



# PCI 9054 Data Book

---





# PCI 9054 Data Book

---

Version 2.1

January 2000

**Website:** <http://www.plxtech.com>

**Email:** [apps@plxtech.com](mailto:apps@plxtech.com)

**Phone:** 408 774-9060

800 759-3735

**FAX:** 408 774-2169

© 2000 PLX Technology, Inc. All rights reserved.

PLX Technology, Inc. retains the right to make changes to this product at any time, without notice. Products may have minor variations to this publication, known as errata. PLX assumes no liability whatsoever, including infringement of any patent or copyright, for sale and use of PLX products.

PLX Technology and the PLX logo are registered trademarks of PLX Technology, Inc.

Other brands and names are the property of their respective owners.

Order Number: 9054-SIL-DB-P1-2.1

Printed in the USA, January 2000

# Contents

<b>Figures</b> .....	<b>xiii</b>
<b>Tables</b> .....	<b>xv</b>
<b>Registers</b> .....	<b>xix</b>
<b>Timing Diagrams</b> .....	<b>xxiii</b>
<b>Preface</b> .....	<b>xxvii</b>
<b>Revision History</b> .....	<b>xxviii</b>
<b>1. Introduction</b> .....	<b>1-1</b>
1.1. Features .....	1-1
1.2. Company and Product Background .....	1-3
1.2.1. PCI 9054 I/O Accelerator .....	1-3
1.2.2. Data Pipe Architecture Technology .....	1-3
1.2.2.1. Dual DMA Channels .....	1-3
1.2.2.2. PCI Initiator (Direct Master) .....	1-3
1.2.2.3. PCI Target (Direct Slave) .....	1-4
1.2.2.4. PCI Messaging .....	1-4
1.2.3. PCI 9054 PCI Applications .....	1-4
1.2.3.1. High Performance Motorola MPC850 or MPC860 PowerQUICC Designs .....	1-4
1.2.3.2. High Performance CompactPCI Adapter Designs .....	1-4
1.2.3.2.1. Hot Swap Capable .....	1-4
1.2.3.2.2. Hot Swap Friendly .....	1-4
1.2.3.3. PCI Bus Embedded Host Design .....	1-5
1.2.4. PCI 9054 Major Features .....	1-5
1.2.5. PCI 9054 Data Assignment Convention .....	1-7
1.2.5.1. PCI 9050/9080 Compatibility .....	1-7
1.2.5.2. Pin Compatibility .....	1-7
1.2.5.3. Register Compatibility .....	1-7
1.2.6. PCI 9054, PCI 9080, and PCI 9050 Comparison .....	1-8
1.2.7. PCI 9054 Signal Listing (M, C, or J Modes) .....	1-9
<b>2. M Mode Bus Operation</b> .....	<b>2-1</b>
2.1. PCI Bus Cycles .....	2-1
2.1.1. PCI Target Command Codes .....	2-1
2.1.2. PCI Master Command Codes .....	2-1
2.1.2.1. DMA Master Command Codes .....	2-1
2.1.2.2. Direct Local-to-PCI Command Codes .....	2-1
2.1.3. PCI Arbitration .....	2-2
2.2. Local Bus Cycles .....	2-2
2.2.1. Local Bus Arbitration .....	2-2
2.2.2. PCI Initiator .....	2-2
2.2.3. PCI Target .....	2-2

- 2.2.4. Wait State Control . . . . . 2-2
  - 2.2.4.1. Wait States—Local Bus . . . . . 2-3
  - 2.2.4.2. Wait States—PCI Bus . . . . . 2-3
- 2.2.5. Burst Mode and Continuous Burst Mode (Bterm “Burst Terminate” Mode) . . . . . 2-3
  - 2.2.5.1. Burst and Bterm Modes . . . . . 2-3
  - 2.2.5.2. Burst-4 Lword Mode . . . . . 2-3
    - 2.2.5.2.1. Continuous Burst Mode (Bterm “Burst Terminate” Mode) . . . . . 2-4
  - 2.2.5.3. Partial Lword Accesses . . . . . 2-4
- 2.2.6. Local Bus Read Accesses . . . . . 2-4
- 2.2.7. Local Bus Write Accesses . . . . . 2-4
- 2.2.8. PCI Target Accesses to 8- or 16-Bit Local Bus . . . . . 2-4
- 2.2.9. Local Bus Data Parity . . . . . 2-4
- 2.3. Big Endian/Little Endian . . . . . 2-5
  - 2.3.1. PCI Bus Little Endian Mode . . . . . 2-5
  - 2.3.2. Local Bus Big/Little Endian Mode . . . . . 2-5
    - 2.3.2.1. 32-Bit Local Bus—Big Endian Mode . . . . . 2-5
    - 2.3.2.2. 16-Bit Local Bus—Big Endian Mode . . . . . 2-6
    - 2.3.2.3. 8-Bit Local Bus—Big Endian Mode . . . . . 2-6
    - 2.3.2.4. Local Bus Big/Little Endian Mode Accesses . . . . . 2-7
- 2.4. Serial EEPROM . . . . . 2-7
  - 2.4.1. Vendor and Device ID Registers . . . . . 2-7
    - 2.4.1.1. Serial EEPROM Initialization . . . . . 2-7
    - 2.4.1.2. Local Initialization . . . . . 2-7
  - 2.4.2. Serial EEPROM Operation . . . . . 2-7
    - 2.4.2.1. Long Serial EEPROM Load . . . . . 2-8
    - 2.4.2.2. Extra Long Serial EEPROM Load . . . . . 2-10
    - 2.4.2.3. New Capabilities Function Support . . . . . 2-10
    - 2.4.2.4. Recommended Serial EEPROMs . . . . . 2-10
    - 2.4.2.5. Serial EEPROM Initialization . . . . . 2-11
  - 2.4.3. Internal Register Access . . . . . 2-11
    - 2.4.3.1. PCI Bus Access to Internal Registers . . . . . 2-11
    - 2.4.3.2. Local Bus Access to Internal Registers . . . . . 2-12
  - 2.4.4. Serial EEPROM Timing Diagrams . . . . . 2-13

**3. M Mode Functional Description . . . . . 3-1**

- 3.1. Reset Operation . . . . . 3-1
  - 3.1.1. PCI Bus Input RST# . . . . . 3-1
  - 3.1.2. Software Reset . . . . . 3-1
- 3.2. PCI 9054 Initialization . . . . . 3-1
- 3.3. Response to FIFO Full or Empty . . . . . 3-1
- 3.4. Direct Data Transfer Modes . . . . . 3-1
  - 3.4.1. PCI Initiator Operation (Local Master-to-PCI Target) . . . . . 3-1
    - 3.4.1.1. PCI Initiator Memory and I/O Decode . . . . . 3-3
    - 3.4.1.2. PCI Initiator FIFOs . . . . . 3-3
    - 3.4.1.3. PCI Initiator Memory Access . . . . . 3-4
    - 3.4.1.4. PCI Initiator I/O Configuration Access . . . . . 3-5
    - 3.4.1.5. PCI Initiator I/O . . . . . 3-5
    - 3.4.1.6. RETRY# Capability . . . . . 3-5
      - 3.4.1.6.1. PCI Initiator Write FIFO Full . . . . . 3-5
      - 3.4.1.6.2. PCI Initiator Delayed Read . . . . . 3-5
    - 3.4.1.7. PCI Initiator Configuration (PCI Configuration Type 0 or Type 1 Cycles) . . . . . 3-5
      - 3.4.1.7.1. PCI Initiator Configuration Cycle Example . . . . . 3-6
    - 3.4.1.8. PCI Initiator PCI Dual Address Cycle . . . . . 3-6
    - 3.4.1.9. PCI Initiator/Target Abort . . . . . 3-7
    - 3.4.1.10. PCI Initiator Memory Write and Invalidate . . . . . 3-7

3.4.2. IDMA/SDMA Operation .....	3-9
3.4.2.1. IDMA Operation .....	3-9
3.4.2.2. SDMA Operation .....	3-9
3.4.3. PCI Target Operation (PCI Master-to-Local Bus Access) .....	3-9
3.4.3.1. PCI Target Lock .....	3-10
3.4.3.2. PCI Target PCI v2.1 Delayed Read Mode .....	3-10
3.4.3.3. PCI Target PCI Read Ahead Mode .....	3-10
3.4.3.4. PCI Target Transfer .....	3-11
3.4.3.5. PCI Target PCI-to-Local Address Mapping .....	3-12
3.4.3.5.1. PCI Target Local Bus Initialization .....	3-12
3.4.3.5.2. PCI Target PCI Initialization .....	3-12
3.4.3.5.3. PCI Target Transfer Size .....	3-14
3.4.3.5.3.1. PCI Target Example .....	3-14
3.4.3.6. PCI Target Priority .....	3-15
3.4.4. Deadlock Conditions .....	3-15
3.4.4.1. Backoff .....	3-16
3.4.4.1.1. Software/Hardware Solution for Systems without Backoff Capability .....	3-16
3.4.4.1.2. Preempt Solution .....	3-16
3.4.4.2. Software Solutions to Deadlock .....	3-16
3.5. DMA Operation .....	3-16
3.5.1. DMA PCI Dual Address Cycle .....	3-17
3.5.2. Block DMA Mode .....	3-17
3.5.2.1. Block DMA PCI Dual Address Cycle .....	3-19
3.5.3. Scatter/Gather DMA Mode .....	3-19
3.5.3.1. Scatter/Gather DMA PCI Dual Address Cycle .....	3-20
3.5.3.2. DMA Clear Count Mode .....	3-20
3.5.4. DMA Memory Write and Invalidate .....	3-20
3.5.4.1. DMA Abort .....	3-20
3.5.5. DMA Priority .....	3-20
3.5.6. DMA Channel 0/1 Interrupts .....	3-22
3.5.7. DMA Data Transfers .....	3-22
3.5.7.1. Local-to-PCI Bus DMA Transfer .....	3-23
3.5.7.2. PCI-to-Local Bus DMA Transfer .....	3-23
3.5.7.3. DMA Local Bus Error Condition .....	3-24
3.5.7.4. DMA Unaligned Transfers .....	3-24
3.5.8. Demand Mode DMA, Channel 0 .....	3-24
3.5.9. End of Transfer (EOT#) Input .....	3-24
3.5.10. DMA Arbitration .....	3-25
3.5.11. Local Bus Latency and Pause Timers .....	3-25
3.6. M Mode Timing Diagrams .....	3-26
3.6.1. M Mode PCI Initiator .....	3-27
3.6.2. M Mode PCI Target .....	3-36
3.6.3. M Mode DMA .....	3-51

**4. C and J Modes Bus Operation .....** **4-1**

4.1. PCI Bus Cycles .....	4-1
4.1.1. PCI Target Command Codes .....	4-1
4.1.2. PCI Master Command Codes .....	4-1
4.1.2.1. DMA Master Command Codes .....	4-1
4.1.2.2. Direct Local-to-PCI Command Codes .....	4-1
4.1.3. PCI Arbitration .....	4-2
4.2. Local Bus Cycles .....	4-2
4.2.1. Local Bus Arbitration .....	4-2
4.2.2. PCI Initiator .....	4-2
4.2.3. PCI Target .....	4-2

4.2.4. Wait State Control . . . . .	4-3
4.2.4.1. Wait States—Local Bus . . . . .	4-3
4.2.4.2. Wait States—PCI Bus . . . . .	4-3
4.2.5. Burst Mode and Continuous Burst Mode (Bterm “Burst Terminate” Mode) . . . . .	4-3
4.2.5.1. Burst and Bterm Modes . . . . .	4-3
4.2.5.2. Burst-4 Lword Mode . . . . .	4-4
4.2.5.2.1. Continuous Burst Mode (Bterm “Burst Terminate” Mode) . . . . .	4-4
4.2.5.3. Partial Lword Accesses . . . . .	4-4
4.2.6. Recovery States (J Mode Only) . . . . .	4-4
4.2.7. Local Bus Read Accesses . . . . .	4-4
4.2.8. Local Bus Write Accesses . . . . .	4-4
4.2.9. PCI Target Accesses to 8- or 16-Bit Local Bus . . . . .	4-4
4.2.10. Local Bus Data Parity . . . . .	4-5
4.3. Big Endian/Little Endian . . . . .	4-5
4.3.1. PCI Bus Little Endian Mode . . . . .	4-5
4.3.2. Local Bus Big/Little Endian Mode . . . . .	4-5
4.3.2.1. 32-Bit Local Bus—Big Endian Mode . . . . .	4-6
4.3.2.2. 16-Bit Local Bus—Big Endian Mode . . . . .	4-6
4.3.2.3. 8-Bit Local Bus—Big Endian Mode . . . . .	4-6
4.3.2.4. Local Bus Big/Little Endian Mode Accesses . . . . .	4-7
4.4. Serial EEPROM . . . . .	4-7
4.4.1. Vendor and Device ID Registers . . . . .	4-7
4.4.1.1. Serial EEPROM Initialization . . . . .	4-7
4.4.1.2. Local Initialization . . . . .	4-7
4.4.2. Serial EEPROM Operation . . . . .	4-7
4.4.2.1. Long Serial EEPROM Load . . . . .	4-8
4.4.2.2. Extra Long Serial EEPROM Load . . . . .	4-10
4.4.2.3. New Capabilities Function Support . . . . .	4-11
4.4.2.4. Recommended Serial EEPROMs . . . . .	4-11
4.4.2.5. Serial EEPROM Initialization . . . . .	4-11
4.4.3. Internal Register Access . . . . .	4-11
4.4.3.1. PCI Bus Access to Internal Registers . . . . .	4-12
4.4.3.2. Local Bus Access to Internal Registers . . . . .	4-12
4.4.4. Serial EEPROM Timing Diagrams . . . . .	4-13
<b>5. C and J Modes Functional Description . . . . .</b>	<b>5-1</b>
5.1. Reset Operation . . . . .	5-1
5.1.1. PCI Bus Input RST# . . . . .	5-1
5.1.2. Software Reset . . . . .	5-1
5.2. PCI 9054 Initialization . . . . .	5-1
5.3. Response to FIFO Full or Empty . . . . .	5-1
5.4. Direct Data Transfer Modes . . . . .	5-1
5.4.1. PCI Initiator Operation (Local Master-to-PCI Target) . . . . .	5-1
5.4.1.1. PCI Initiator Memory and I/O Decode . . . . .	5-4
5.4.1.2. PCI Initiator FIFOs . . . . .	5-4
5.4.1.3. PCI Initiator Memory Access . . . . .	5-4
5.4.1.4. PCI Initiator I/O Configuration Access . . . . .	5-5
5.4.1.5. PCI Initiator I/O . . . . .	5-5
5.4.1.6. PCI Initiator Configuration (PCI Configuration Type 0 or Type 1 Cycles) . . . . .	5-5
5.4.1.6.1. PCI Initiator Configuration Cycle Example . . . . .	5-6
5.4.1.7. PCI Initiator PCI Dual Address Cycle . . . . .	5-6
5.4.1.8. PCI Initiator/Target Abort . . . . .	5-6
5.4.1.9. PCI Initiator Memory Write and Invalidate . . . . .	5-8
5.4.2. PCI Target Operation (PCI Master-to-Local Bus Access) . . . . .	5-8
5.4.2.1. PCI Target Lock . . . . .	5-9
5.4.2.2. PCI Target PCI v2.1 Delayed Read Mode . . . . .	5-9
5.4.2.3. PCI Target PCI Read Ahead Mode . . . . .	5-9

5.4.2.4. PCI Target Transfer	5-9
5.4.2.5. PCI Target PCI-to-Local Address Mapping	5-10
5.4.2.5.1. PCI Target Local Bus Initialization	5-10
5.4.2.5.2. PCI Target PCI Initialization	5-10
5.4.2.5.3. PCI Target Byte Enables (C Mode)	5-12
5.4.2.5.4. PCI Target Byte Enables (J Mode)	5-12
5.4.2.5.4.1. PCI Target Example	5-12
5.4.2.6. PCI Target Priority	5-13
5.4.3. Deadlock Conditions	5-13
5.4.3.1. Backoff	5-13
5.4.3.1.1. Software/Hardware Solution for Systems without Backoff Capability	5-14
5.4.3.1.2. Preempt Solution	5-14
5.4.3.2. Software Solutions to Deadlock	5-14
5.5. DMA Operation	5-14
5.5.1. DMA PCI Dual Address Cycle	5-15
5.5.2. Block DMA Mode	5-15
5.5.2.1. Block DMA PCI Dual Address Cycle	5-16
5.5.3. Scatter/Gather DMA Mode	5-16
5.5.3.1. Scatter/Gather DMA PCI Dual Address Cycle	5-18
5.5.3.2. DMA Clear Count Mode	5-18
5.5.4. DMA Memory Write and Invalidate	5-20
5.5.4.1. DMA Abort	5-20
5.5.5. DMA Priority	5-20
5.5.6. DMA Channel 0/1 Interrupts	5-20
5.5.7. DMA Data Transfers	5-20
5.5.7.1. Local-to-PCI Bus DMA Transfer	5-21
5.5.7.2. PCI-to-Local Bus DMA Transfer	5-21
5.5.7.3. DMA Unaligned Transfers	5-22
5.5.8. Demand Mode DMA, Channel 0	5-22
5.5.9. End of Transfer (EOT#) Input	5-22
5.5.10. DMA Arbitration	5-23
5.5.11. Local Bus Latency and Pause Timers	5-23
5.6. C Mode Timing Diagrams	5-24
5.6.1. C Mode PCI Initiator	5-25
5.6.2. C Mode PCI Target	5-44
5.6.3. C Mode DMA	5-63
5.7. J Mode Timing Diagrams	5-73
5.7.1. J Mode PCI Initiator	5-73
5.7.2. J Mode PCI Target	5-77
5.7.3. J Mode DMA	5-81

**6. PCI Local Interrupts and User I/O** ..... **6-1**

6.1. Interrupts	6-1
6.1.1. PCI Interrupts (INTA#)	6-1
6.1.2. Local Interrupt Input (LINT#)	6-1
6.1.3. Local Interrupt Output (LINT#)	6-2
6.1.4. Master/Target Abort Interrupt	6-2
6.1.5. Mailbox Registers	6-2
6.1.6. Doorbell Registers	6-2
6.1.6.1. Local-to-PCI Doorbell Interrupt	6-3
6.1.6.1.1. M Mode Local-to-PCI Doorbell Interrupt	6-3
6.1.6.1.2. C and J Modes Local-to-PCI Doorbell Interrupt	6-3
6.1.6.2. PCI-to-Local Doorbell Interrupt	6-3
6.1.7. Built-In Self Test Interrupt (BIST)	6-3
6.1.8. DMA Channel 0/1 Interrupts	6-3
6.1.9. All Modes PCI SERR# (PCI NMI)	6-4
6.1.10. M Mode PCI SERR#	6-4

6.1.11. Local NMI	6-4
6.1.12. M Mode Local TEA# (Local NMI)	6-4
6.1.13. C and J Modes Local LSERR# (Local NMI)	6-4
6.2. User Input and Output	6-5

**7. Intelligent I/O (I<sub>2</sub>O) . . . . . 7-1**

7.1. I <sub>2</sub> O-Compatible Message Unit	7-1
7.1.1. Inbound Messages	7-1
7.1.2. Outbound Messages	7-1
7.1.3. I <sub>2</sub> O Pointer Management	7-2
7.1.4. Inbound Free List FIFO	7-2
7.1.5. Inbound Post Queue FIFO	7-3
7.1.6. Outbound Post Queue FIFO	7-3
7.1.7. Outbound Post Queue	7-5
7.1.8. Inbound Free Queue	7-5
7.1.9. Outbound Free List FIFO	7-5
7.1.10. I <sub>2</sub> O Enable Sequence	7-5

**8. PCI Power Management . . . . . 8-1**

8.1. Overview	8-1
8.1.1. PCI Power Management Functional Description	8-1
8.1.2. System Changes Power Mode Example	8-2
8.1.3. Wake-Up Request Example	8-2

**9. CompactPCI Hot Swap . . . . . 9-1**

9.1. Hot Swap	9-1
9.1.1. Overview	9-1
9.1.2. Controlling Connection Processes	9-1
9.1.2.1. Hardware Connection Control	9-1
9.1.2.1.1. Board Slot Control	9-2
9.1.2.1.2. Board Healthy	9-2
9.1.2.1.3. Platform Reset	9-2
9.1.2.2. Software Connection Control	9-3
9.1.2.2.1. Ejector Switch and Blue LED	9-3
9.1.2.2.2. ENUM#	9-3
9.1.2.2.3. Hot Swap Control/Status Register (HS_CSR)	9-3
9.1.2.2.4. Hot Swap Capabilities Register Bit Definition	9-4

**10. PCI Vital Product Data (VPD) . . . . . 10-1**

10.1. Overview	10-1
10.1.1. VPD Capabilities Register	10-1
10.1.2. VPD Serial EEPROM Partitioning	10-1
10.1.3. Sequential Read Only	10-1
10.1.4. Random Read and Write	10-2

<b>11. Registers</b> .....	<b>11-1</b>
11.1. New Register Definitions Summary .....	11-1
11.2. Register Address Mapping .....	11-2
11.2.1. PCI Configuration Registers .....	11-2
11.2.2. Local Configuration Registers .....	11-3
11.2.3. Runtime Registers .....	11-4
11.2.4. DMA Registers .....	11-5
11.2.5. Messaging Queue Registers .....	11-6
11.3. PCI Configuration Registers .....	11-7
11.4. Local Configuration Registers .....	11-19
11.5. Runtime Registers .....	11-29
11.6. DMA Registers .....	11-34
11.7. Messaging Queue Registers .....	11-40
<b>12. Pin Description</b> .....	<b>12-1</b>
12.1. Pin Summary .....	12-1
12.2. Pinout Common to All Bus Modes .....	12-2
12.3. M Bus Mode Pinout .....	12-6
12.4. C Bus Mode Pinout .....	12-9
12.5. J Bus Mode Pinout .....	12-12
12.6. NANDTREE Test Access Method .....	12-15
<b>13. Electrical Specifications</b> .....	<b>13-1</b>
13.1. General Electrical Specifications .....	13-1
13.2. Local Inputs .....	13-3
13.3. Local Outputs .....	13-5
<b>14. Package, Signal, and Pinout Specs</b> .....	<b>14-1</b>
14.1. 176-Pin PQFP .....	14-1
14.2. 225-Pin PBGA .....	14-4
<b>A. General Information</b> .....	<b>A-1</b>
A.1. Ordering Instructions .....	A-1
A.2. United States and International Representatives, and Distributors .....	A-1
A.3. Technical Support .....	A-1
<b>Index</b> .....	<b>Index-1</b>



# Figures

1-1. Typical Adapter Block Diagram	1-1
1-2. PCI 9054 Internal Block Diagram	1-2
1-3. High-Performance MPC850 or MPC860 PowerQUICC Adapter Design	1-3
1-4. High-Performance CompactPCI Adapter	1-5
1-5. High-Performance Embedded Adapter	1-5
2-1. Wait States	2-2
2-2. Big/Little Endian—32-Bit Local Bus	2-6
2-3. Big/Little Endian—16-Bit Local Bus	2-6
2-4. Big/Little Endian—8-Bit Local Bus	2-6
2-5. Serial EEPROM Memory Map	2-11
2-6. PCI 9054 Internal Register Access	2-11
2-7. Address Decode Mode	2-12
3-1. PCI Initiator Access of the PCI Bus	3-3
3-2. PCI Initiator Write	3-4
3-3. PCI Initiator Read	3-4
3-4. Block DMA Mode Initialization (Single Address or Dual Address PCI)	3-7
3-5. Dual Address Timing	3-8
3-6. PCI Target PCI v2.1 Delayed Reads	3-10
3-7. PCI Target PCI 9054 Read Ahead Mode	3-10
3-8. PCI Target Write	3-11
3-9. PCI Target Read	3-11
3-10. Local Bus PCI Target Access	3-13
3-11. DMA, PCI-to-Local Bus	3-18
3-12. DMA, Local-to-PCI Bus	3-18
3-13. Dual Address Timing	3-18
3-14. Scatter/Gather DMA Mode from PCI-to-Local Bus (Control Access from the Local Bus)	3-19
3-15. Scatter/Gather DMA Mode from Local-to-PCI Bus (Control Access from the PCI Bus)	3-19
3-16. Scatter/Gather DMA Mode Descriptor Initialization [PCI SAC/DAC PCI Address (DMADAC0, DMADAC1) Register Dependent]	3-21
3-17. Scatter/Gather DMA Mode Descriptor Initialization [DAC PCI Address (DMAMODE0[18], DMAMODE1[18]) Descriptor Dependent (PCI Address High Added)]	3-21
3-18. Local-to-PCI Bus DMA Data Transfer Operation	3-23
3-19. PCI-to-Local Bus DMA Data Transfer Operation	3-23
4-1. Wait States	4-3
4-2. Big/Little Endian—32-Bit Local Bus	4-6
4-3. Big/Little Endian—16-Bit Local Bus	4-6
4-4. Big/Little Endian—8-Bit Local Bus	4-6
4-5. Serial EEPROM Memory Map	4-11
4-6. PCI 9054 Internal Register Access	4-11
4-7. Address Decode Mode	4-12
5-1. PCI Initiator Access of the PCI Bus	5-3
5-2. PCI Initiator Write	5-4
5-3. PCI Initiator Read	5-4
5-4. Block DMA Mode Initialization (Single Address or Dual Address PCI)	5-7
5-5. Dual Address Timing	5-7

- 5-6. PCI Target PCI v2.1 Delayed Reads ..... 5-9
- 5-7. PCI Target PCI 9054 Read Ahead Mode ..... 5-9
- 5-8. PCI Target Write ..... 5-10
- 5-9. PCI Target Read ..... 5-10
- 5-10. Local Bus PCI Target Access ..... 5-11
- 5-11. DMA, PCI-to-Local Bus ..... 5-15
- 5-12. DMA, Local-to-PCI Bus ..... 5-16
- 5-13. Dual Address Timing ..... 5-17
- 5-14. Scatter/Gather DMA Mode from PCI-to-Local Bus (Control Access from the Local Bus) ..... 5-17
- 5-15. Scatter/Gather DMA Mode from Local-to-PCI Bus (Control Access from the PCI Bus) ..... 5-17
- 5-16. Scatter/Gather DMA Mode Descriptor Initialization [PCI SAC/DAC PCI Address (DMADAC0, DMADAC1) Register Dependent] ..... 5-19
- 5-17. Scatter/Gather DMA Mode Descriptor Initialization [DAC PCI Address (DMAMODE0[18], DMAMODE1[18]) Descriptor Dependent] (PCI Address High Added) ..... 5-19
- 5-18. Local-to-PCI Bus DMA Data Transfer Operation ..... 5-21
- 5-19. PCI-to-Local Bus DMA Data Transfer Operation ..... 5-21
- 6-1. Interrupt and Error Sources ..... 6-1
- 6-2. Mailbox and Doorbell Message Passing ..... 6-2
- 7-1. Typical I<sub>2</sub>O Server/Adapter Card Design ..... 7-1
- 7-2. Driver Architecture Compared ..... 7-1
- 7-3. Circular FIFO Operation ..... 7-4
- 9-1. Redirection of BD\_SEL# ..... 9-2
- 9-2. Board Healthy ..... 9-2
- 9-3. PCI Reset ..... 9-2
- 9-4. Hot Swap Capabilities Register Bit Definition ..... 9-4
- 10-1. VPD Capabilities Register ..... 10-1
- 13-1. PCI 9054 Local Input Setup and Hold Waveform ..... 13-3
- 13-2. PCI 9054 Local Output Delay ..... 13-5
- 13-3. PCI 9054 ALE Output Delay to the Local Clock ..... 13-6
- 14-1. 176-Pin PQFP Package Mechanical Dimensions—Topside and Cross-Section Views ..... 14-1
- 14-2. 176-Pin PQFP PCB Layout Suggested Land Pattern ..... 14-2
- 14-3. 176-Pin PQFP PCI 9054 Pinout ..... 14-3
- 14-4. 225-Pin PBGA Package Mechanical Dimensions—Topside, Underside, and Cross-Section Views ..... 14-4
- 14-5. 225-Pin PBGA PCB Layout Suggested Land Pattern ..... 14-5
- 14-6. 225-Pin PBGA Package Layout (Underside View) ..... 14-6

# Tables

1-1. FIFO Depth	1-5
1-2. Programmable Local Bus Modes	1-6
1-3. PCI 9054 Data Assignment Convention	1-7
1-4. Comparison of PCI 9054, PCI 9080, and PCI 9050	1-8
1-5. PCI 9054 PCI Signal Listing (M, C, or J Modes)	1-9
1-6. PCI 9054 Local Signal Listing (M, C, or J Modes)	1-10
2-1. PCI Target Command Codes	2-1
2-2. DMA Master Command Codes	2-1
2-3. Local-to-PCI Memory Access	2-1
2-4. Local-to-PCI I/O Access	2-1
2-5. Local-to-PCI Configuration Access	2-1
2-6. Local Bus Types (176-Pin PQFP)	2-2
2-7. Local Bus Types (225-Pin PBGA)	2-2
2-8. Burst and Bterm on the Local Bus	2-3
2-9. Burst-4 Lword Mode	2-3
2-10. PCI Bus Little Endian Byte Lanes	2-5
2-11. Byte Number and Lane Cross-Reference	2-5
2-12. Big/Little Endian Program Mode	2-5
2-13. Cycles Reference Tables	2-5
2-14. Upper Lword Lane Transfer	2-5
2-15. Upper Word Lane Transfer	2-6
2-16. Lower Word Lane Transfer	2-6
2-17. Upper Byte Lane Transfer	2-6
2-18. Lower Byte Lane Transfer	2-6
2-19. Serial EEPROM Guidelines	2-8
2-20. Long Serial EEPROM Load Registers	2-9
2-21. Extra Long Serial EEPROM Load Registers	2-10
2-22. New Capabilities Function Support Features	2-10
3-1. Response to FIFO Full or Empty	3-2
3-2. PCI Target Burst Mode Cycle Detection	3-12
3-3. Data Bus TSIZ[0:1] Contents for Single Write Cycles	3-14
3-4. Data Bus TSIZ[0:1] Requirements for Single Read Cycles	3-14
3-5. DMA	3-18
3-6. Normal DMA with EOT Function	3-18
3-7. Demand Mode DMA, Channel 0	3-24
3-8. Any DMA Transfer Channel 0/1 with EOT Functionality	3-25
4-1. PCI Target Command Codes	4-1
4-2. DMA Master Command Codes	4-1
4-3. Local-to-PCI Memory Access	4-1
4-4. Local-to-PCI I/O Access	4-1
4-5. Local-to-PCI Configuration Access	4-1
4-6. Local Bus Types (176-Pin PQFP)	4-2
4-7. Local Bus Types (225-Pin PBGA)	4-2
4-8. Burst and Bterm on the Local Bus	4-3

## Tables

---

4-9. Burst-4 Lword Mode	4-4
4-10. PCI Bus Little Endian Byte Lanes	4-5
4-11. Byte Number and Lane Cross-Reference	4-5
4-12. Big/Little Endian Program Mode	4-5
4-13. Cycles Reference Tables	4-5
4-14. Upper Lword Lane Transfer	4-6
4-15. Upper Word Lane Transfer	4-6
4-16. Lower Word Lane Transfer	4-6
4-17. Upper Byte Lane Transfer	4-6
4-18. Lower Byte Lane Transfer	4-6
4-19. Serial EEPROM Guidelines	4-8
4-20. Long Serial EEPROM Load Registers	4-9
4-21. Extra Long Serial EEPROM Load Registers	4-10
4-22. New Capabilities Function Support Features	4-11
5-1. Response to FIFO Full or Empty	5-2
5-2. DMA Local Burst Mode	5-16
7-1. Queue Starting Address	7-2
7-2. Circular FIFO Summary	7-6
9-1. Hot Swap Control	9-4
11-1. New Registers Definitions Summary (As Compared to the PCI 9080)	11-1
11-2. PCI Configuration Registers	11-2
11-3. Local Configuration Registers	11-3
11-4. Runtime Registers	11-4
11-5. DMA Registers	11-5
11-6. Messaging Queue Registers	11-6
12-1. Pin Type Abbreviations	12-1
12-2. Power and Ground Pins (176-Pin PQFP)	12-2
12-3. Power and Ground Pins (225-Pin PBGA)	12-2
12-4. Serial EEPROM Interface Pins	12-3
12-5. PCI System Bus Interface Pins	12-3
12-6. Local Bus Mode and Processor Independent Interface Pins	12-5
12-7. M Bus Mode Interface Pins	12-6
12-8. C Bus Mode Interface Pins	12-9
12-9. J Bus Mode Interface Pins	12-12
12-10. Sequential Interconnection of the PCI 9054 NANDTREE (Pin Definition)	12-16
13-1. Absolute Maximum Ratings	13-1
13-2. Operating Ranges	13-1
13-3. Capacitance (Sample Tested Only)	13-1
13-4. Thermal Resistance of Packages	13-1
13-5. Electrical Characteristics over Operating Range	13-2
13-6. AC Electrical Characteristics (Local Inputs) over Operating Range (M Mode)	13-3
13-7. AC Electrical Characteristics (Local Inputs) over Operating Range (C and J Modes)	13-4
13-8. AC Electrical Characteristics (Local Outputs) over Operating Range (M Mode)	13-5
13-9. AC Electrical Characteristics (Local Outputs) over Operating Range (C and J Modes)	13-6

14-1. 176-Pin PQFP Package Mechanical Dimensions ..... 14-1  
14-2. 225-Pin PBGA Package Mechanical Dimensions ..... 14-4  
14-3. 225-Pin PBGA PCI 9054 Pinout ..... 14-7  
A-1. Available Packages ..... A-1



# Registers

11-1. (PCIIDR; PCI:00h, LOC:00h) PCI Configuration ID .....	11-7
11-2. (PCICR; PCI:04h, LOC:04h) PCI Command .....	11-7
11-3. (PCISR; PCI:06h, LOC:06h) PCI Status .....	11-8
11-4. (PCIREV; PCI:08h, LOC:08h) PCI Revision ID .....	11-8
11-5. (PCICCR; PCI:09-0Bh, LOC:09-0Bh) PCI Class Code .....	11-9
11-6. (PCICLSR; PCI:0Ch, LOC:0Ch) PCI Cache Line Size .....	11-9
11-7. (PCILTR; PCI:0Dh, LOC:0Dh) PCI Bus Latency Timer .....	11-9
11-8. (PCIHTR; PCI:0Eh, LOC:0Eh) PCI Header Type .....	11-9
11-9. (PCIBISTR; PCI:0Fh, LOC:0Fh) PCI Built-In Self Test (BIST) .....	11-10
11-10. (PCIBAR0; PCI:10h, LOC:10h) PCI Base Address Register for Memory Accesses to Local, Runtime, and DMA .....	11-10
11-11. (PCIBAR1; PCI:14h, LOC:14h) PCI Base Address Register for I/O Accesses to Local, Runtime, and DMA .....	11-11
11-12. (PCIBAR2; PCI:18h, LOC:18h) PCI Base Address Register for Memory Accesses to Local Address Space 0 .....	11-11
11-13. (PCIBAR3; PCI:1Ch, LOC:1Ch) PCI Base Address Register for Memory Accesses to Local Address Space 1 .....	11-12
11-14. (PCIBAR4; PCI:20h, LOC:20h) PCI Base Address .....	11-12
11-15. (PCIBAR5; PCI:24h, LOC:24h) PCI Base Address .....	11-12
11-16. (PCICIS; PCI:28h, LOC:28h) PCI Cardbus CIS Pointer .....	11-12
11-17. (PCISVID; PCI:2Ch, LOC:2Ch) PCI Subsystem Vendor ID .....	11-12
11-18. (PCISID; PCI:2Eh, LOC:2Eh) PCI Subsystem ID .....	11-13
11-19. (PCIERBAR; PCI:30h, LOC:30h) PCI Expansion ROM Base .....	11-13
11-20. (CAP_PTR; PCI:34h, LOC:34h) New Capability Pointer .....	11-14
11-21. (PCIILR; PCI:3Ch, LOC:3Ch) PCI Interrupt Line .....	11-14
11-22. (PCIIPR; PCI:3Dh, LOC:3Dh) PCI Interrupt Pin .....	11-14
11-23. (PCIMGR; PCI:3Eh, LOC:3Eh) PCI Min_Gnt .....	11-14
11-24. (PCIMLR; PCI:3Fh, LOC:3Fh) PCI Max_Lat .....	11-14
11-25. (PMCAPID; PCI:40h, LOC:180h) Power Management Capability ID .....	11-14
11-26. (PMNEXT; PCI:41h, LOC:181h) Power Management Next Capability Pointer .....	11-15
11-27. (PMC; PCI:42h, LOC:182h) Power Management Capabilities .....	11-15
11-28. (PMCSR; PCI:44h, LOC:184h) Power Management Control/Status .....	11-16
11-29. (PMCSR_BSE; PCI:46h, LOC:186h) PMCSR Bridge Support Extensions .....	11-16
11-30. (PMDATA; PCI:47h, LOC:187h) Power Management Data .....	11-17
11-31. (HS_CNTL; PCI:48h, LOC:188h) Hot Swap Control .....	11-17
11-32. (HS_NEXT; PCI:49h, LOC:189h) Hot Swap Next Capability Pointer .....	11-17
11-33. (HS_CSR; PCI:4Ah, LOC:18Ah) Hot Swap Control/Status .....	11-17
11-34. (PVPDCNTL; PCI:4Ch, LOC:18Ch) PCI Vital Product Data Control .....	11-18
11-35. (PVPD_NEXT; PCI:4Dh, LOC:18Dh) PCI Vital Product Data Next Capability Pointer .....	11-18
11-36. (PVPDAD; PCI:4Eh, LOC:18Eh) PCI Vital Product Data Address .....	11-18
11-37. (PVPDATA; PCI:50h, LOC:190h) PCI VPD Data .....	11-18
11-38. (LAS0RR; PCI:00h, LOC:80h) Local Address Space 0 Range Register for PCI-to-Local Bus ...	11-19
11-39. (LAS0BA; PCI:04h, LOC:84h) Local Address Space 0 Local Base Address (Remap) .....	11-19
11-40. (MARBR; PCI:08h or ACh, LOC:88h or 12Ch) Mode/DMA Arbitration .....	11-20
11-41. (BIGEND; PCI:0Ch, LOC:8Ch) Big/Little Endian Descriptor .....	11-21
11-42. (LMISC; PCI:0Dh, LOC:8Dh) Local Miscellaneous Control .....	11-22

## Registers

---

11-43. (PROT_AREA; PCI:0Eh, LOC:8Eh) Serial EEPROM Write-Protected Address Boundary . . . . .	11-22
11-44. (EROMRR; PCI:10h, LOC:90h) Expansion ROM Range . . . . .	11-23
11-45. (EROMBA; PCI:14h, LOC:94h) Expansion ROM Local Base Address (Remap) and BREQo Control . . . . .	11-23
11-46. (LBRD0; PCI:18h, LOC:98h) Local Address Space 0/Expansion ROM Bus Region Descriptor . .	11-24
11-47. (DMRR; PCI:1Ch, LOC:9Ch) Local Range Register for PCI Initiator-to-PCI . . . . .	11-25
11-48. (DMLBAM; PCI:20h, LOC:A0h) Local Bus Base Address Register for PCI Initiator-to-PCI Memory . . . . .	11-25
11-49. (DMLBAI; PCI:24h, LOC:A4h) Local Base Address Register for PCI Initiator-to-PCI I/O Configuration . . . . .	11-25
11-50. (DMPBAM; PCI:28h, LOC:A8h) PCI Base Address (Remap) Register for PCI Initiator-to-PCI Memory . . . . .	11-26
11-51. (DMCFGa; PCI:2Ch, LOC:ACh) PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration . . . . .	11-27
11-52. (LAS1RR; PCI:F0h, LOC:170h) Local Address Space 1 Range Register for PCI-to-Local Bus . .	11-27
11-53. (LAS1BA; PCI:F4h, LOC:174h) Local Address Space 1 Local Base Address (Remap) . . . . .	11-28
11-54. (LBRD1; PCI:F8h, LOC:178h) Local Address Space 1 Bus Region Descriptor . . . . .	11-28
11-55. (DMDAC; PCI:FCh, LOC:17Ch) PCI Initiator PCI Dual Address Cycle . . . . .	11-28
11-56. (MBOX0; PCI:40h or 78h, LOC:C0h) Mailbox Register 0. . . . .	11-29
11-57. (MBOX1; PCI:44h or 7Ch, LOC:C4h) Mailbox Register 1 . . . . .	11-29
11-58. (MBOX2; PCI:48h, LOC:C8h) Mailbox Register 2 . . . . .	11-29
11-59. (MBOX3; PCI:4Ch, LOC:CCh) Mailbox Register 3. . . . .	11-29
11-60. (MBOX4; PCI:50h, LOC:D0h) Mailbox Register 4 . . . . .	11-29
11-61. (MBOX5; PCI:54h, LOC:D4h) Mailbox Register 5 . . . . .	11-29
11-62. (MBOX6; PCI:58h, LOC:D8h) Mailbox Register 6 . . . . .	11-29
11-63. (MBOX7; PCI:5Ch, LOC:DCh) Mailbox Register 7. . . . .	11-30
11-64. (P2LDBELL; PCI:60h, LOC:E0h) PCI-to-Local Doorbell . . . . .	11-30
11-65. (L2PDBELL; PCI:64h, LOC:E4h) Local-to-PCI Doorbell . . . . .	11-30
11-66. (INTCSR; PCI:68h, LOC:E8h) Interrupt Control/Status . . . . .	11-31
11-67. (CNTRL; PCI:6Ch, LOC:ECh) Serial EEPROM Control, PCI Command Codes, User I/O Control, and Init Control . . . . .	11-33
11-68. (PCIHIDR; PCI:70h, LOC:F0h) PCI Hardcoded Configuration ID . . . . .	11-33
11-69. (PCIHREV; PCI:74h, LOC:F4h) PCI Hardcoded Revision ID . . . . .	11-33
11-70. (DMAMODE0; PCI:80h, LOC:100h) DMA Channel 0 Mode . . . . .	11-34
11-71. (DMAPADR0; PCI:84h, LOC:104h) DMA Channel 0 PCI Address . . . . .	11-35
11-72. (DMALADR0; PCI:88h, LOC:108h) DMA Channel 0 Local Address . . . . .	11-35
11-73. (DMASIZ0; PCI:8Ch, LOC:10Ch) DMA Channel 0 Transfer Size (Bytes) . . . . .	11-35
11-74. (DMADPR0; PCI:90h, LOC:110h) DMA Channel 0 Descriptor Pointer . . . . .	11-35
11-75. (DMAMODE1; PCI:94h, LOC:114h) DMA Channel 1 Mode . . . . .	11-36
11-76. (DMAPADR1; PCI:98h, LOC:118h) DMA Channel 1 PCI Address . . . . .	11-37
11-77. (DMALADR1; PCI:9Ch, LOC:11Ch) DMA Channel 1 Local Address. . . . .	11-37
11-78. (DMASIZ1; PCI:A0h, LOC:120h) DMA Channel 1 Transfer Size (Bytes). . . . .	11-37
11-79. (DMADPR1; PCI:A4h, LOC:124h) DMA Channel 1 Descriptor Pointer . . . . .	11-37
11-80. (DMACSR0; PCI:A8h, LOC:128h) DMA Channel 0 Command/Status. . . . .	11-38
11-81. (DMACSR1; PCI:A9h, LOC:129h) DMA Channel 1 Command/Status. . . . .	11-38
11-82. (DMAARB; PCI:ACh, LOC:12Ch) DMA Arbitration. . . . .	11-39
11-83. (DMATHR; PCI:B0h, LOC:130h) DMA Threshold . . . . .	11-39
11-84. (DMADAC0; PCI:B4h, LOC:134h) DMA Channel 0 PCI Dual Address Cycle Address Register . .	11-39

---

11-85. (DMADAC1; PCI:B8h, LOC:138h) DMA Channel 1 PCI Dual Address Cycle Address . . . . .	11-39
11-86. (OPQIS; PCI:30h, LOC:B0h) Outbound Post Queue Interrupt Status . . . . .	11-40
11-87. (OPQIM; PCI:34h, LOC:B4h) Outbound Post Queue Interrupt Mask . . . . .	11-40
11-88. (IQP; PCI:40h) Inbound Queue Port . . . . .	11-40
11-89. (OQP; PCI:44h) Outbound Queue Port . . . . .	11-40
11-90. (MQCR; PCI:C0h, LOC:140h) Messaging Queue Configuration . . . . .	11-41
11-91. (QBAR; PCI:C4h, LOC:144h) Queue Base Address . . . . .	11-41
11-92. (IFHPR; PCI:C8h, LOC:148h) Inbound Free Head Pointer . . . . .	11-41
11-93. (IFTPR; PCI:CCh, LOC:14Ch) Inbound Free Tail Pointer . . . . .	11-41
11-94. (IPHPR; PCI:D0h, LOC:150h) Inbound Post Head Pointer . . . . .	11-42
11-95. (IPTPR; PCI:D4h, LOC:154h) Inbound Post Tail Pointer . . . . .	11-42
11-96. (OFHPR; PCI:D8h, LOC:158h) Outbound Free Head Pointer . . . . .	11-42
11-97. (OFTPR; PCI:DCh, LOC:15Ch) Outbound Free Tail Pointer . . . . .	11-42
11-98. (OPHPR; PCI:E0h, LOC:160h) Outbound Post Head Pointer . . . . .	11-42
11-99. (OPTPR; PCI:E4h, LOC:164h) Outbound Post Tail Pointer . . . . .	11-43
11-100. (QSR; PCI:E8h, LOC:168h) Queue Status/Control . . . . .	11-43



# Timing Diagrams

2-1. Initialization from Serial EEPROM (2K Bit)	2-13
2-2. Initialization from Serial EEPROM (4K Bit)	2-14
2-3. PCI Configuration Write to PCI Configuration Register	2-14
2-4. PCI Configuration Read to PCI Configuration Register	2-15
2-5. PCI Memory Write to Local Configuration Register	2-15
2-6. PCI Memory Read to Local Configuration Register	2-16
2-7. Local Interrupt Asserting PCI Interrupt	2-16
3-1. Local Bus Arbitration (BR#, BG#, BB#, and so forth)	3-26
3-2. PCI Initiator Single Write Cycle, Zero Wait States	3-27
3-3. PCI Initiator Single Read Cycle, One Wait State (WAIT# Asserted for One Clock)	3-28
3-4. PCI Initiator Burst Write Cycle of Four Lwords, Zero Wait States	3-29
3-5. PCI Initiator Burst Read Cycle of Four Lwords, Zero Wait States	3-30
3-6. PCI Initiator Deferred Read Mode (RETRY#)	3-31
3-7. PCI Initiator Burst Read with Read Ahead Mode (Prefetch Counter Set to Eight Lwords)	3-32
3-8. Local Configuration Write to Configuration Register	3-33
3-9. Local Configuration Read from Configuration Register	3-33
3-10. PCI Initiator Burst Write of Six Lwords beyond MPC860 Protocol	3-34
3-11. PCI Initiator Burst Read of Six Lwords beyond MPC860 Protocol	3-35
3-12. PCI Target Single Write Cycle, Zero Wait States	3-36
3-13. PCI Target Single Write Cycle, One Wait State by Delaying TA#	3-37
3-14. Local Bus Single Write Cycle, Zero Wait States, Burst Enabled, 16-Bit Local Bus	3-38
3-15. Local Bus Single Write Cycle, One Wait State, Burst Disabled, 8-Bit Local Bus	3-39
3-16. PCI Target Single Read Cycle, Zero Wait States	3-40
3-17. PCI Target Single Read Cycle, One Wait State Using TA#	3-41
3-18. PCI Target Single Read Cycle, Zero Wait States, 16-Bit Bus	3-42
3-19. PCI Target Single Read Cycle, One Wait State, Burst Disabled, 8-Bit Local Bus	3-43
3-20. PCI Target Burst Write Cycle of Four Lwords, Bterm Disabled, Burst Enabled	3-43
3-21. PCI Target Burst Read Cycle of Four Lwords, Bterm Disabled, Burst Enabled	3-44
3-22. PCI Target Burst Write Cycle of Eight Lwords, Bterm Disabled, Burst Enabled	3-44
3-23. PCI Target Burst Read Cycle of Eight Lwords, Bterm Disabled, Burst Enabled	3-45
3-24. PCI Target Burst Write Cycle of 10 Lwords, Zero Wait States beyond MPC860 Protocol, Bterm Enabled, Burst Enabled	3-45
3-25. PCI Target Burst Read Cycle of 10 Lwords, Zero Wait States beyond MPC860 Protocol, Bterm Enabled, Burst Enabled	3-46
3-26. Initialization from Serial EEPROM (2K Bit)	3-47
3-27. Initialization from Serial EEPROM (4K Bit)	3-48
3-28. PCI Configuration Write to PCI Configuration Register	3-48
3-29. PCI Configuration Read to PCI Configuration Register	3-49
3-30. PCI Memory Write to Local Configuration Register	3-49
3-31. PCI Memory Read to Local Configuration Register	3-50
3-32. Local Interrupt Asserting PCI Interrupt	3-50
3-33. Master Abort Condition During PCI Initiator Read Cycle Causes TEA#	3-51
3-34. DMA PCI-to-Local, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords	3-52
3-35. DMA Local-to-PCI, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords	3-53
3-36. DMA Local-to-PCI, Address Unaligned, Bterm Disabled, Burst Enabled,	

Transfer Size = Six Lwords	3-54
3-37. DMA PCI-to-Local, Address Unaligned, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords	3-55
3-38. DMA Local-to-PCI, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts in the Middle of the Quad-Lword of Data	3-56
3-39. DMA Local-to-PCI, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts at the Last Data of the First Quad-Lword	3-57
3-40. DMA PCI-to-Local, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts in the Middle of the First Quad-Lword of Data	3-58
3-41. DMA PCI-to-Local, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts at the Last Data of the First Quad-Lword	3-59
3-42. DMA Local-to-PCI, Bterm Enabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts at End of Third Local Data beyond MPC860 Protocol	3-60
3-43. DMA PCI-to-Local, Bterm Enabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts at End of Third Local Data beyond MPC860 Protocol	3-61
3-44. Local Bus Latency Timer (Eight Clocks) and Pause Timer (Four Clocks) in DMA Operation	3-62
3-45. Local Bus Latency Timer (Eight Clocks) and Pause Timer (Four Clocks) in DMA Operation beyond MPC860 Protocol	3-62
3-46. DMA PCI-to-Local, Bterm Enabled, Burst Enabled, Transfer Size = 10 Lwords, beyond MPC860 Protocol	3-63
3-47. DMA Local-to-PCI, Bterm Enabled, Burst Enabled, Transfer Size = 10 Lwords, beyond MPC860 Protocol	3-64
3-48. IDMA Single Write Cycle	3-65
4-1. Initialization from Serial EEPROM (2K Bit)	4-13
4-2. Initialization from Serial EEPROM (4K Bit)	4-14
4-3. PCI Configuration Write to PCI Configuration Register	4-14
4-4. PCI Configuration Read to PCI Configuration Register	4-15
4-5. PCI Memory Write to Local Configuration Register	4-15
4-6. PCI Memory Read to Local Configuration Register	4-16
4-7. Local Interrupt Asserting PCI Interrupt	4-16
5-1. Local Bus Arbitration (LHOLD and LHOLDA)	5-24
5-2. PCI Initiator Single Write	5-25
5-3. PCI Initiator Single Read	5-26
5-4. PCI Initiator Memory Write of 12 Lwords with WAIT# Input	5-27
5-5. PCI Initiator Burst Read of Seven Lwords with WAIT# Input	5-28
5-6. PCI Initiator Memory Read of 12 Lwords with Prefetch Counter Set to 16	5-29
5-7. Memory Write and Invalidate with Cache Line Size of Eight	5-30
5-8. PCI Initiator Memory Read with Keep Bus Mode	5-31
5-9. PCI Initiator Memory Read with Drop Bus Mode	5-32
5-10. PCI Bus Request (REQ#) Delay During Direct Master Write (Eight-PCI Clock Delay)	5-33
5-11. PCI Initiator Locked Read Followed by Write and Release (LLOCK# and LOCK#)	5-34
5-12. BREQo and Deadlock	5-35
5-13. Local Bus Write to Configuration Register	5-36
5-14. Local Bus Read to Configuration Register	5-36
5-15. PCI Initiator Configuration Read—Type 1 or Type 0	5-37
5-16. PCI Initiator Configuration Write—Type 1 or Type 0	5-38
5-17. Initialization from Serial EEPROM (2K Bit)	5-39
5-18. Initialization from Serial EEPROM (4K Bit)	5-40
5-19. PCI Configuration Write to PCI Configuration Register	5-41

5-20. PCI Configuration Read to PCI Configuration Register	5-41
5-21. PCI Memory Write to Local Configuration Register	5-42
5-22. PCI Memory Read to Local Configuration Register	5-42
5-23. Local Interrupt Asserting PCI Interrupt	5-43
5-24. PCI Target Single Write (32-Bit Local Bus)	5-44
5-25. PCI Target Burst Cycle Write (16-Bit Local Bus)	5-45
5-26. PCI Target Burst Cycle Write (8-Bit Local Bus)	5-45
5-27. PCI Target Single-Cycle Read (32-Bit Local Bus)	5-46
5-28. PCI Target Single Read with One Wait State Using READY# Input (32-Bit Local Bus)	5-47
5-29. PCI Target Single Read with One Wait State Using Internal Wait State (32-Bit Local Bus)	5-48
5-30. PCI Target Non-Burst Write (32-Bit Local Bus)	5-49
5-31. PCI Target Non-Burst Write (8-Bit Local Bus)	5-50
5-32. PCI Target Non-Burst Local Bus Read	5-51
5-33. PCI Target Burst Write with Bterm Enabled (32-Bit Local Bus)	5-52
5-34. PCI Target Burst Write with Bterm Disabled (32-Bit Local Bus)	5-53
5-35. PCI Target Burst Read with Prefetch Counter Set to 8 (32-Bit Local Bus)	5-54
5-36. PCI Target Burst Read with Prefetch Counter Set to 5 (32-Bit Local Bus)	5-55
5-37. PCI Target Burst Write (32-Bit Local Bus)	5-56
5-38. PCI Target Burst Write (16-Bit Local Bus)	5-57
5-39. PCI Target Burst Write with External Wait States (8-Bit Local Bus)	5-57
5-40. Delayed Read Transaction PCI Specification v2.1	5-58
5-41. PCI Target Read No Flush Mode (Read Ahead Mode), Prefetch Enabled, Prefetch Count Disabled	5-59
5-42. PCI Target Burst Write Suspended by BREQi	5-60
5-43. Locked PCI Target Read Followed by Write and Release (LLOCKo#)	5-61
5-44. PCI Target in BIGEND Local Bus with BIGEND# Input	5-62
5-45. DMA Aligned PCI Address to Aligned Local Address, Bterm Enabled, Burst Enabled	5-63
5-46. DMA Aligned PCI Address to Aligned Local Address, Bterm Disabled, Burst Enabled	5-64
5-47. Scatter/Gather DMA with Descriptor on Local Bus	5-65
5-48. Scatter/Gather DMA from Local-to-PCI with Descriptor on PCI Bus	5-66
5-49. Demand DMA, Terminate with BLAST# (Local-to-PCI)	5-67
5-50. DMA Local-to-PCI, Terminate with EOT#	5-68
5-51. DMA PCI-to-Local, Terminate with EOT#	5-69
5-52. DMA PCI-to-Local with Local Bus Pause and Latency Timers	5-70
5-53. Single-Cycle Demand DMA Mode (PCI-to-Local)	5-71
5-54. Multiple-Cycle Demand DMA Mode (PCI-to-Local)	5-72
5-55. PCI Initiator Single Write	5-73
5-56. PCI Initiator Single Read	5-74
5-57. PCI Initiator Burst Write of 10 Lwords, Zero Wait States	5-75
5-58. PCI Initiator Burst Read of 10 Lwords, Zero Wait States	5-76
5-59. PCI Target Single Write (32-Bit Local Bus)	5-77
5-60. PCI Target Single Read (32-Bit Local Bus)	5-78
5-61. PCI Target Burst Write with Bterm Enabled (32-Bit Local Bus)	5-79
5-62. PCI Target Burst Read with Prefetch Enabled (32-Bit Local Bus), Prefetch Counter Set to 8	5-80
5-63. DMA Aligned PCI Address to Aligned Local Address, Bterm Enabled, Burst Enabled	5-81
5-64. DMA Aligned Local Address to Aligned PCI Address, Bterm Enabled, Burst Enabled	5-82



# PREFACE

The information contained in this document is subject to change without notice. Although an effort has been made to keep the information accurate, there may be misleading or even incorrect statements made herein.

The following is a list of additional documentation to provide the reader with more information about the PCI 9054 and related subjects:

- *PCI Local Bus Specification, Revision 2.1*  
PCI Special Interest Group  
2575 NE Kathryn #17, Hillsboro, OR, 97124, USA  
Tel: 800 433-5177 (domestic only) or 503 693-6232, Fax: 503 693-8344, <http://www.pcisig.com>
- *PCI Hot-Plug Specification, Revision 1.0*  
PCI Special Interest Group  
2575 NE Kathryn #17, Hillsboro, OR, 97124, USA  
Tel: 800 433-5177 (domestic only) or 503 693-6232, Fax: 503 693-8344, <http://www.pcisig.com>
- *PCI Power Management Interface Specification, Revision 1.0, June 30, 1997*  
PCI Special Interest Group  
2575 NE Kathryn #17, Hillsboro, OR, 97124, USA  
Tel: 800 433-5177 (domestic only) or 503 693-6232, Fax: 503 693-8344, <http://www.pcisig.com>
- *PICMG 2.1 R1.0 CompactPCI® Hot Swap Specification*  
PCI Industrial Computer Manufacturers Group (PICMG)  
c/o Rogers Communications, 401 Edgewater Place, Suite 500, Wakefield, MA 01880, USA  
Tel: 781 246-9318, Fax: 781 224-1239, <http://www.picmg.org>
- *Intelligent I/O (I<sub>2</sub>O) Architecture Specification Revision 1.5*  
I<sub>2</sub>O Special Interest Group  
404 Balboa Street, San Francisco, CA, 94118, USA  
Tel: 415 750-8352, Fax: 415 751-4829, <http://www.i2osig.org>

# REVISION HISTORY

Date	Revision	Comments
09/16/1997	0.30	Revised version 0.2 Red Book to include PCI 9054 special functions.
01/21/1998	0.90	Added special functions: Power Management, Vital Product Data (VPD), and CompactPCI Hot Swap functionality; PCI Hot Plug; MPC850 or MPC860 Operation mode; 176-pin PQFP and 225-pin PBGA pinouts. Portions of the CompactPCI <i>Specification</i> used for the CompactPCI Hot Swap section.
02/13/1998	0.90	Incorporated changes from January and February review meetings.
06/26/1998	0.91	Incorporated engineering feedback and PCI 9054 flier text. Updated Local timing specifications.
09/02/1998	0.92	<p>Changed title to "data book" and removed references to inches in Tables 14-1 and 14-2.</p> <p>Added serial EEPROM information to Table 1-4.</p> <p>Added information to BB# description in Tables 1-5 and 12-7.</p> <p>Changed TA# information for PCI Target and DMA modes in Section 2.2.4.1.</p> <p>Added cross-references to other sections for Bust Forever mode in Tables 2-8 and 2-9.</p> <p>Added burst information to Section 2.2.5.1.</p> <p>Added VPD information relating to long serial EEPROM loads to Section 2.4.2.1.</p> <p>Added "Register Bits Affected" information to Tables 2-20 and 2-21.</p> <p>Added port size information to Table 3-3.</p> <p>Changed DMA PCI DAC information in Section 3-5.1.</p> <p>Added Section 5.5.2.1, "Block DMA PCI Dual Address Cycle."</p> <p>Added Section 5.5.3.1, "Scatter/Gather DMA PCI Dual Address Cycle."</p> <p>Changed captions for Figures 1-3, 5/16, and 5-17.</p> <p>Added recommendation to not assert EOT# during a descriptor load on the Local Bus to Sections 3.5.9 and 5.5.9.</p> <p>Clarified INTCSR[11]=0 information in Section 6.1.3.</p> <p>Added further VPD information to Section 10 (specifically 10.1 and 10.14).</p> <p>Changed PCIBAR register information and corrected the serial EEPROM information for 3Ch in Table 11-2.</p> <p>Changed the PCI:0Ch and LOC:8Ch information in Table 11-3.</p> <p>Changed the Read information from "Yes" to "PCI" in Tables 11-40 through 11-43 (PVPDCNTL, PVPD_NEXT, PVPDAD, and PVPDATA).</p> <p>Changed CNTRL[27] description to "Reserved" in Table 11-73.</p> <p>Swapped pinouts for TSIZ[0:1] and LBE[3:2]#:</p> <ul style="list-style-type: none"> <li>M Mode— TSIZ0 is now pin 92 (PQFP) and pin N14 (PBGA).</li> <li>          TSIZ1 is now pin 91 (PQFP) and pin P15 (PBGA).</li> <li>C and J Modes—LBE3# is now pin 92 (PQFP) and pin N14 (PBGA).</li> <li>                  LBE2# is now pin 91 (PQFP) and pin P15 (PBGA).</li> </ul> <p>Corrected minor typographical errors.</p>

Date	Revision	Comments
11/12/1998	1.0	<p>Separated entire book by placing M mode and C and J modes into their own sections.</p> <p>Globally changed “negate” to “de-assert.”</p> <p>Changed “Half Word” to “Word” and “Word” to “Lword” in M mode sections.</p> <p>Globally changed “driven” to “asserted” in multiple tables.</p> <p>Globally added a slash to “PCI Initiator/Target Abort.”</p> <p>Added “PCI” to “Dual Address Cycle” text references.</p> <p>Changed “PCI command 7h” to “PCI command code = 7h.”</p> <p>Updated timing diagrams for all modes.</p> <p>Changed “Arbitration” to “Control Access” in Scatter/Gather DMA Mode from Local-to-PCI Bus figure captions.</p> <p>Corrected inputs and outputs for DMA 0/1 Local and PCI FIFOs in the PCI 9054 Internal Block Diagram.</p> <p>Changed “Local Initialization” and “Serial EEPROM” section descriptions.</p> <p>Added new Section 1.5, “PCI 9054 Data Assignment Convention.”</p> <p>Split signal listings into PCI and Local in Tables 1-5 and 1-6.</p> <p>Added exception for MWI mode to Sections 2.1.2 and 4.1.2.</p> <p>Changed assertion and bus ownership information in Sections 2.2.1 and 4.2.1.</p> <p>Local Bus cycle description changed in Sections 2.2.2 and 4.2.2. Further changes to Section 4.2.2 added regarding use of BLAST#.</p> <p>Added New Capabilities function support information, Sections 2.4.2.3 and 4.4.2.3.</p> <p>Added Hot Swap Next Capability Pointer information to Tables 2-21 and 4-21.</p> <p>Changed Serial EEPROM Memory map information in Figures 2-5 and 4-5.</p> <p>Updated Response to FIFO Full or Empty information in Tables 3-1 and 5-1.</p> <p>Corrected references to Local Configuration and PCI Configuration DMA registers in Sections 3.1.1 and 5.1.2.</p> <p>Added Master Enable (PCICR) and PCI Command Code (CNTRL) to bullet list in Sections 3.4.1 and 5.4.1.</p> <p>Added Continuous Burst information to Section 3.4.1.3 (M mode only).</p> <p>Added Lword information to FIFO description in Sections 3.4.3 and 5.4.2.</p> <p>Miscellaneous changes made to DMA Operation information in Sections 3.5, 3.5.2 and 5.5, 5.5.2.</p> <p>Changed DMA descriptor “Note” text in Sections 3.5.3 and 5.5.3.</p> <p>Added new Sections 3.5.3.2 and 5.5.3.2, “DMA Clear Count Mode.”</p> <p>Changed current descriptor and Channel Done bit text in Sections 3.5.6, 5.5.6, and 6.1.8.</p> <p>Descriptor Pointer and Memory Descriptor Block(s) text changes applied to Figures 3-16, 3-17, 5-16, and 5-17.</p> <p>Changed sections describing “End of Transfer (EOT#) Input” and “Local Bus Latency and Pause Timers” to third-level headings. Moved “DMA Arbitration” section to appear after “End of Transfer (EOT#) Input.”</p> <p>Added “PCI Initiator” to “Read Ahead mode” text and changed “Multiple-Cycle reads” to “Burst-Cycle reads” and “are set” to “are asserted” to the “Read” portion of Sections 3.4.1.3 and 5.4.1.3.</p> <p>Added DMA Local Bus Timer information to Sections 3.5.11 and 5.5.11.</p> <p>Removed text describing generation of a new ADS# from Section 4.2.5.1.</p> <p>Changed “End of Chain bits are programmed” to “End of Chain bits are detected” in Sections 3.5.3 and 5.5.3.</p> <p>Changed Doorbell register information in Figure 6-1.</p> <p>Swapped sequence of Sections 6.1 and 6.2.</p> <p>Applied significant revisions to Section 8, including System Changes Power Mode and Wake-Up Request examples.</p> <p>Combined “Hot Swap Switch” and “Status LED” sections into new Section 9.1.2.2.1, “Ejector Switch and Blue LED.” Added new content to section.</p> <p>Corrected register “Value after Reset” values in Section 11.</p> <p>Revised timing information in Tables 13-6 through 13-9, AC Electrical Characteristics for Local inputs and outputs.</p> <p>Corrected information for B9, C6, D10, and P6 in PBGA Pinout in Table 14-3.</p> <p>Added new Section 15, “Ordering Instructions.”</p>

## Revision History

Date	Revision	Comments
04/1999	1.1	<p>Restyled headers and footers.            Minor typographical changes.            Signal angle corrections to various timing diagrams.            All references to CS# changed to CCS#.            Changed "Direct Slave" to "PCI Target" and "Direct Master" to "PCI Initiator."            Dates changed to reflect new revision.            First para in preface changed to reflect that the document is subject to change without notice.            Changed the following timing diagrams: "Initialization from Serial EEPROM (2K)" and "Initialization from Serial EEPROM (4K)" (all occurrences). Also changed Timing Diagrams 5-13, 5-14, and 5-32.            C and J mode graphics, changed "LD" references to "LA/LAD."            Section 2.2.5.1, changed two Lwords to four Lwords in the case of 16- and 8-bit Local Bus bursts.            Section 2.2.8, changed C and J mode signal name to M mode equivalent (regarding Transfer Size bits, TSIZ[0:1]).            Section 4.2.1, added LHOLDA asserted.            Sections 4.4.3.2 and 5.4.3.1, changed M mode signal names to their C and J mode equivalents.            Figure 5-12, added BLAST#.            Section 5.5.9, changed second para DMAMODE0[14]=1 and DMAMODE1[14]=1 to [15]=1. Third para changed DMAMODE0[14]=0 and DMAMODE1[14]=0 to [15]=0 and added "depending on the Local Bus width."            Timing Diagram 5-32 title changed to read "PCI Target Non-Burst Local Bus Read."            Section 6, entirely restructured.            Table 11-47, changed bit 4 description to reference C and J Modes and M Mode byte lane descriptions separately.            Table 11-86, changed Channel 1 to Channel 0 in Bit 2 description.            Section 12 Pin summary note regarding pull-up and pull-down resistors, corrected M, C, and J mode pin references.            Table 12-7, changed BB# pin type from "TP" to "OC."            Note to Table 13-5, changed I<sub>2</sub>O to "I<sub>2</sub>O simultaneously switching outputs."            Tables 13-6 and 13-7, changed Worst Case values from "2" to "1" for all signals listed.            Section 14, added "PQFP" and "PBGA" suggested land patterns for PCB layout.            Section 16, created new section, listing US and international representatives and distributors.</p>
06/1999	2.0	<p>PCI 9054_AB revision information added.            Merged Features and Company and Product Background sections into one to streamline and be consistent.            Minor typographical changes.            Table 1-6, Local Signal Listing (M, C, or J Modes) TA# input and output reversed.            Following text moved from Section 2.2.4.1 to Section 4.2.4.1 and TA# changed to READY#. "In PCI Target and DMA modes, the TA# signal has no effect until the wait state counter—(LBRD0[21:18, 5:2]), (LBRD1[5:2]), (DMAMODE0[5:2]), and/or (DMAMODE1[5:2])—reaches zero. TA# then controls the number of wait states by being de-asserted in the middle of the data transaction."            Section 2.2.5.1, "BDIP# output is not supported in this case" changed to BDIP# output is supported in burst forever..."            Section 2.2.5.2.1, added "...PCI 9054_AB supports the BDIP# signal for continuous bursts greater than four Lwords, which differs from MPC850/860 protocol..."            Section 2.2.9, added statement regarding parity being checked for Direct Slave reads, Direct Master write and DMA Local Bus reads.            Section 2.4.2 and 4.4.2, added information on the PCI 9054_AB revision offering the ability to manually access the EEPROM.            Section 2.4.2.1 and Section 4.4.2.1, added bit to read back the serial EEPROM value.            Tables 3-3 and 3-4, added bit range to TSIZ heading.            Table 3-5, revised table to reflect correct information.            Section 6.1.12, added statement regarding the PCI 9054 tolerating TEA# input assertion only during PCI Target or DMA transactions.            Section 9.1.2.2.3, added statement regarding the PCI 9054 Hot Swap register (HS_CSR) being accessible for writes from the Local Bus.            Register 11-33, revised write status in bits 1, 6, and 7.            Register 11-67, revised bit 27 to a read bit, not reserved.            Table 13-5, revised ICC Max range to 200.            Table 14-2, revised 225-Pin PBGA Package Mechanical Dimensions.</p>

Date	Revision	Comments
08/1999	2.0	<p>Global, changed "kilobit" to "KB."</p> <p>Minor text changes, additions and deletions throughout book.</p> <p>Section 2.2.5.3 Lword in some instances changed to quad-word.</p> <p>Section 2.2.9 "Direct Slave" changed to "PCI Target," and "Direct Master" changed to "PCI Initiator."</p> <p>Section 2.4.2 "32 bits" changed to "33 bits." In fifth para, output changed to high.</p> <p>Swapped Sections 3.4.1.8 and 3.5.1.</p> <p>Figure 3-10, added Space 1, now reads "Space 0/1"</p> <p>Tables 3-3 and 3-4, added "Single" in title to read "Single Write Cycles" and "Single Read Cycles."</p> <p>Figure 3-17, added "(PCI Address High Added)."</p> <p>Section 3.5.4.1, deleted first two methods.</p> <p>Section 3.5.9, supplemented with BTERM# disable and enable statements.</p> <p>TDs 3-2 through 3-7, revised GNT# signal.</p> <p>TD 3-10, revised GNT#, and TD 3-11, revised GNT# and BDIP# signals.</p> <p>TDs 3-14, 3-15, 3-18, and 3-19, supplemented LA[0:31] signal.</p> <p>TDs 3-24 and 3-25, revised BDIP# signal.</p> <p>TD 3-33, revised TEA# signal.</p> <p>TDs 3-46 and 3-47, revised BDIP# signal.</p> <p>TD 3-48, revised MDREQ# signal.</p> <p>Section 4.2.2, supplemented text in last para.</p> <p>Section 4.4.2 "32 bits" changed to "33 bits." In fifth para, output changed to high.</p> <p>Swapped Sections 5.4.1.7 and 5.5.1.</p> <p>Figure 5-10, added Space 1, now reads "Space 0/1"</p> <p>Figure 5-17, added "(PCI Address High Added)."</p> <p>Section 5.5.4.1, deleted first two methods.</p> <p>TD 5-4, revised LD[31:0] and WAIT# (input) signals.</p> <p>Figure 6-2, changed LINTo# to LINT#.</p> <p>Section 9.1.2.2.2, third para revised.</p> <p>9.1.2.2.3.1, value of "0x00" changed to "0x06."</p> <p>Register 11-20, bits 7:0, write changed from "No" to "Local."</p> <p>Sections 15 and 16 changed to Appendix A and Appendix B.</p>
10/1999	2.0	<p>Published Addendum for page 12-1 replacement, to include missing information previously included in the v1.0 release.</p>
01/2000	2.1	<p>Section 1.2.4, Keep Bus Mode paras, changed "Initiator" to "Target."</p> <p>Table 1-4, deleted "Tolerant" and "Signaling."</p> <p>Table 2-14, changed byte numbers high to low.</p> <p>Section 2.4.2, para 4, last sentence, changed "EEDO" to "EEDI" and "EEDI" to "EEDO."</p> <p>Table 2-21, added "/Status" to Serial EEPROM Offset 54h.</p> <p>Section 3.4.1.9, removed "and BI#" from first sentence of third para.</p> <p>Figure 5-12, reversed last arrow on left side "AD (addr &amp; data)."</p> <p>Section 5.5.7.3, removed last line of third para, "All reads are Lwords."</p> <p>TD 5-25 and 5-26, Changed "Single-Cycle" to "Burst Cycle."</p> <p>6.1.5., added sentence about clearing mailbox registry.</p> <p>Table 11-2, replaced missing Local Access column.</p> <p>Register 11-40, Bit 29, deleted two "ands."</p> <p>Register 11-45, Bit 4, changed to all modes, added "/RETRY#."</p> <p>Register 11-67, Bit 17, changed "Writing" to "Reading."</p> <p>Section 12.1, Notes, changed.</p> <p>Sections 13.2 and 13.3, deleted note.</p> <p>Consolidated Appendices A and B into one appendix.</p>



## 1 INTRODUCTION

### 1.1 FEATURES

- PCI Specification version 2.2 (v2.2) compliant 32-bit, 33-MHz Bus Master Interface Controller with PCI Power Management features for adapters and embedded systems
- General Purpose Bus Master Interface featuring advanced Data Pipe Architecture™ technology, which includes two DMA engines, programmable Target and Initiator Data Transfer modes and PCI messaging functions
- PCI v2.2 Vital Product Data (VPD) configuration support
- PCI Dual Address Cycle (DAC) support
- PCI Hot Plug and CompactPCI Hot Swap compliant
- I<sub>2</sub>O™ v1.5-Ready Messaging Unit
- Two independent DMA channels for Local Bus memory to and from PCI Host Bus Data transfers
- Supports Type 0 and Type 1 Configuration cycles
- Programmable Burst Management
- Programmable Interrupt Generator
- Six programmable FIFOs for zero wait state burst operation
- PCI ↔ Local Data transfers up to 132 MB/s
- 3.3V, 5V tolerant PCI and Local signaling supports Universal PCI Adapter designs, 3.3V core, low-power CMOS in 176-pin PQFP and 225-pin PBGA
- Supports Local Bus Direct-Connect to the Motorola® MPC850 or MPC860 PowerQUICC™, Intel® i960 family and IBM® PPC401 CPUs and similar bus protocol devices
- Programmable Local Bus runs up to 50 MHz and supports non-multiplexed 32-bit address/data, multiplexed 32-bit, and slave accesses of 8-, 16-, or 32-bit Local Bus devices
- Serial EEPROM interface
- Three PCI-to-Local Address spaces
- Programmable Local Bus wait states
- Programmable prefetch counter
- Local Bus runs asynchronously to the PCI Bus
- Eight 32-bit Mailbox and two 32-bit Doorbell registers
- Performs Big Endian ↔ Little Endian conversion
- PCI-to-Local Delayed Read mode
- Local-to-PCI Deferred Read mode (M mode only)
- Flexible 3.3V, 5V Tolerant Local Bus operation up to 50 MHz
- Industrial Temp Range operation

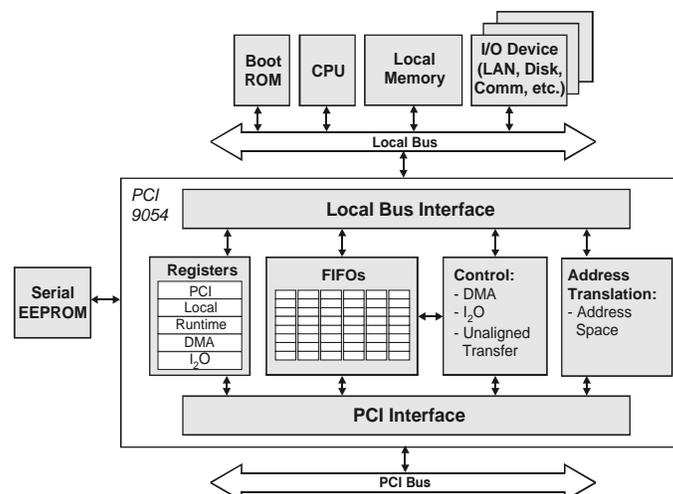


Figure 1-1. Typical Adapter Block Diagram

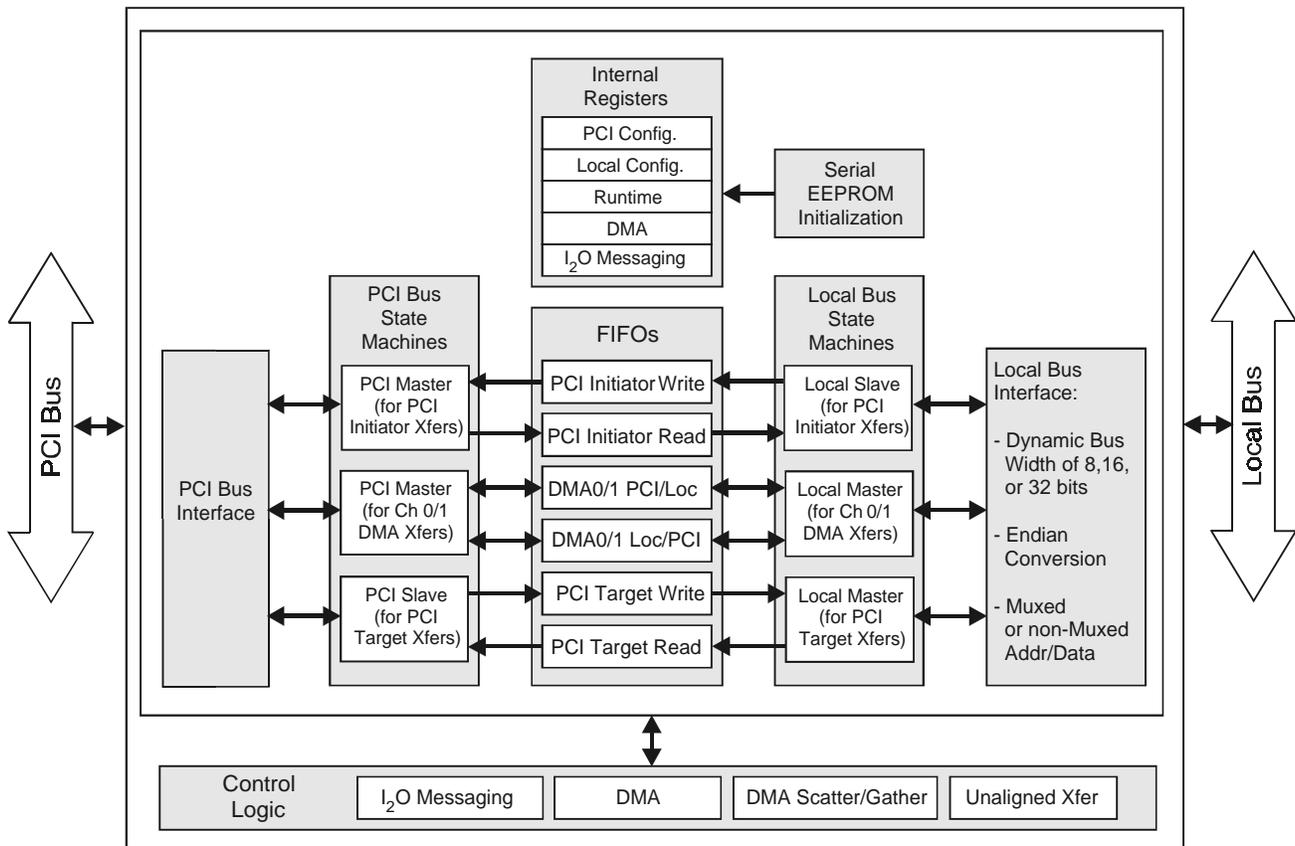


Figure 1-2. PCI 9054 Internal Block Diagram

## 1.2 COMPANY AND PRODUCT BACKGROUND

PLX Technology, Inc., the world leader in PCI-to-Local Bus I/O Accelerator chips, supports OEM customers in a wide variety of PCI applications. Customer applications include PC workstations and servers, PCI add-in boards, embedded PCI communication systems (such as routers and switches), and industrial PCI implementations (such as CompactPCI, PMC, and Passive Backplane PCI).

PLX Technology, Inc., is an active participant in industry standard committees, including the PCI SIG®, I<sub>2</sub>O SIG®, and PICMG®, and maintains active developer technology and cross-marketing partnerships with industry leaders, such as Intel, IBM, Hewlett-Packard, Motorola, Integrated Systems, WindRiver, and others.

Focused on providing complete solutions for PCI implementations, PLX provides design assistance to customers in the form of Reference Design Kits (RDK) and Software Development Kits (SDK). Depending upon the application, these kits may include reference boards, API libraries, software debug tools, and sample device drivers with source, enabling customers to quickly bring new designs to production. New tools, application notes, FAQs, and information updates are constantly added to our website ([www.plxtech.com](http://www.plxtech.com)) for the convenience of PLX customers. Our expertise and total solutions for the PCI interface allow customers to focus on adding value in their designs without worrying about the complexities of implementing PCI, I<sub>2</sub>O, and CompactPCI.

### 1.2.1 PCI 9054 I/O Accelerator

The PCI 9054, a 32-bit 33-MHz PCI Bus Master I/O Accelerator, is the most advanced general-purpose bus Master device available. It offers a robust PCI Specification v2.2 implementation enabling Burst transfers up to 132 MB/second. The PCI 9054 incorporates the industry leading PLX Data Pipe Architecture™ technology, including DMA engines, programmable PCI Initiator and Target Data-Transfer modes, and PCI messaging functions.

## 1.2.2 Data Pipe Architecture Technology

### 1.2.2.1 Dual DMA Channels

- Dual independent channels provide flexible prioritization scheme
- Programmable Burst length, including unlimited burst
- Direct hardware control of DMA
  - Demand mode DMA operation
  - Block mode or Scatter/Gather operation
  - End of Transfer (EOT) signal
- Shuttle mode DMA channel support provides automatic invalidation of used DMA descriptors
- Unaligned transfer support
- Supports PCI Bus Mastering from the Local Slave-only devices
- Scatter/Gather list management
  - Descriptors can be found in PCI Bus or Local Bus memory
  - Allows independent Scatter/Gather ring management

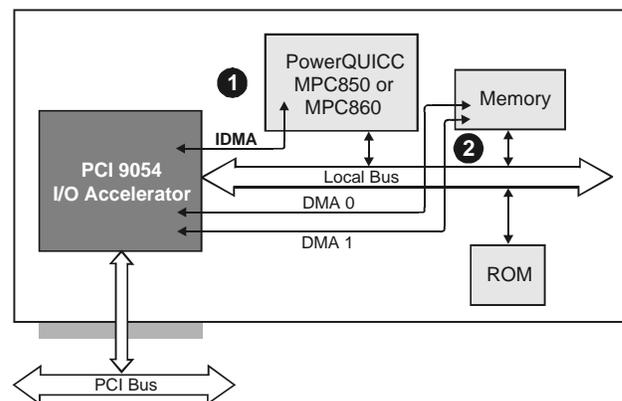


Figure 1-3. High-Performance MPC850 or MPC860 PowerQUICC Adapter Design

### 1.2.2.2 PCI Initiator (Direct Master)

- Type 0 and Type 1 Configuration cycles
- Supports all PCI Memory and I/O cycles
- Initiator Read prefetching
- Burst length control-programmable threshold pointer
- Unaligned transfer control
- Big/Little Endian conversion

### 1.2.2.3 PCI Target (Direct Slave)

- Multiple independent address spaces
- Dynamic Local Bus width control
- Target Read prefetching
- Big/Little Endian conversion
- Local Bus priority control
- PCI Latency Timer

### 1.2.2.4 PCI Messaging

- Complete Messaging Unit with mailbox and doorbell registers
- Queue management pointers that can be used for message passing under the I<sup>2</sup>O protocol or a custom protocol

## 1.2.3 PCI 9054 PCI Applications

### 1.2.3.1 High Performance Motorola MPC850 or MPC860 PowerQUICC Designs

A key application for the PCI 9054 is Motorola MPC850- or MPC860-based adapters for telecom and networking applications. These applications include high performance communications such as WAN/LAN controller cards, high-speed modem cards, Frame Relay cards, and routers and switches. The PCI 9054 simplifies designs by providing an industry-leading enhanced direct-connect interface to the MPC850 or MPC860 processor. The flexible PCI 9054 3.3V, 5V tolerant I/O buffers, combined with a Local Bus operation up to 50 MHz, is ideally suited for current and future PowerQUICC processors. The PCI 9054 also provides support for the MPC850 or MPC860 IDMA channel for movement of data between the internal MPC850 or MPC860 I/O and the PCI Bus. In addition, the PCI 9054 also makes use of the advanced Data Pipe Architecture technology, allowing unlimited Burst capability, as shown in Figure 1-3.

1. For PowerQUICC IDMA operation, the PCI 9054 transfers data to the PCI Bus under the control of the IDMA handshake protocol.
2. At the same time, the PCI 9054 Data Pipe Architecture technology DMA can be operated bidirectionally, with the PCI 9054 as the Master for both buses, to manage transfers of data from the Local Bus to the PCI Bus or from the PCI Bus to

the Local Bus. This is a prime example of how the PCI 9054 provides superior general purpose Bus Master performance and provides designers using the PowerQUICC processor with greater flexibility in implementing multiple simultaneous I/O transfers. The PCI 9054 has unlimited bursting capability, which enhances any MPC850 or MPC860 PowerQUICC design.

### 1.2.3.2 High Performance CompactPCI Adapter Designs

Another key application for the PCI 9054 is CompactPCI adapters for telecom and networking applications. These applications include high performance communications such as WAN/LAN controller cards, high-speed modem cards, Frame Relay cards, and telephony cards for telecom switches and remote-access systems.

The PCI 9054 has integrated key features to enable live-insertion of Hot Swap CompactPCI adapters. The PCI 9054 PICMG v2.1-compatible Hot Swap *Friendly* PCI interface includes both Hot Swap *Capable* and Hot Swap *Friendly* features.

#### 1.2.3.2.1 Hot Swap Capable

- PCI Specification v2.1 or better
- Vcc tolerant from early power
- Tolerant of asynchronous reset
- Tolerant of precharge voltage
- Limited I/O pin leakage at precharge voltage

#### 1.2.3.2.2 Hot Swap Friendly

- Incorporates the Hot Swap Control/Status register (HS\_CSR)
- Incorporates an Extended Capability Pointer (ECP) mechanism
- Incorporates added resources for software control of ENUM#, the ejector switch, and the status LED, which indicates insertion and removal to the user

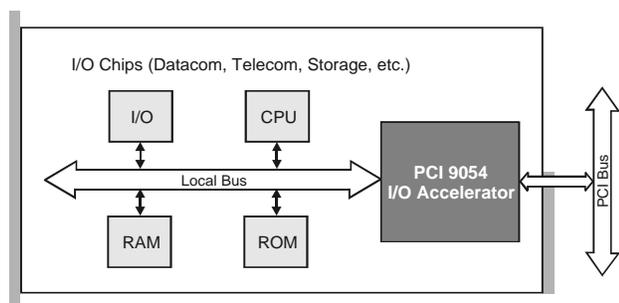


Figure 1-4. High-Performance CompactPCI Adapter

### 1.2.3.3 PCI Bus Embedded Host Design

Another application for the PCI 9054 is PCI Host-embedded system designs, such as network switches and routers, printer engines, set-top boxes, and industrial equipment. In this configuration, the PCI 9054 Data Pipe Architecture technology allows high-performance transfer modes. In addition, the PCI 9054 supports both Type 0 and Type 1 PCI Configuration cycles that allow the PCI 9054 to configure other PCI devices or cards in the system.

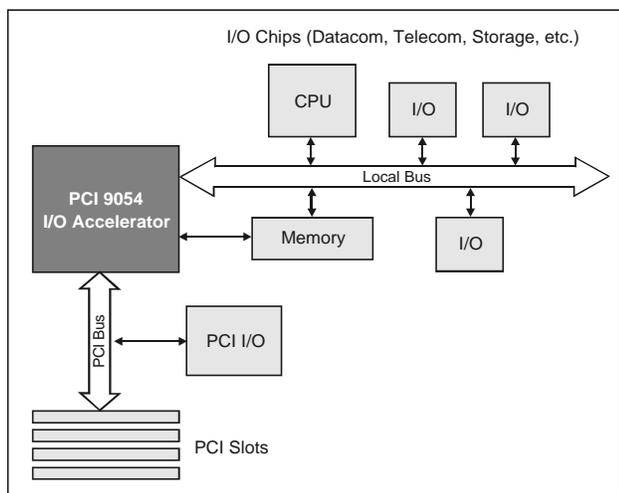


Figure 1-5. High-Performance Embedded Adapter

### 1.2.4 PCI 9054 Major Features

**PCI v2.1 and v2.2 Compliant.** Compliant with all aspects of PCI Specification v2.1 and v2.2, including PCI Power Management features. Supports four power states for PCI Power Management functions—D<sub>0</sub>, D<sub>1</sub>, D<sub>2</sub>, and D<sub>3hot</sub>—and the Power Management Event interrupt (PME#).

**VPD Support.** Fully supports the Vital Product Data (VPD) PCI extension, which provides an alternate access method other than Expansion ROM for VPD.

**PCI Dual-Address Cycle (DAC) Support (64-bit Address Space).** Supports PCI Dual Address Cycle beyond the low 4-GB Address space. PCI DAC can be used during PCI 9054 PCI Bus Master operation (DMA, PCI Initiator).

**PCI Hot Plug and CompactPCI Hot Swap Compliant.** Compliant with PCI Hot Plug and CompactPCI Hot Swap adapter specifications.

**I<sub>2</sub>O Ready Messaging Unit.** Incorporates the I<sub>2</sub>O Ready Messaging Unit, which enables the adapter or embedded system to communicate with other I<sub>2</sub>O-supported devices. The I<sub>2</sub>O Messaging Unit is fully compatible with the PCI extension of I<sub>2</sub>O Specification v1.5.

**Dual Independently Programmable DMA Controllers with Programmable FIFOs.** Provides two independently programmable DMA controllers with shared programmable FIFOs. Each channel supports Block and Scatter/Gather DMA modes, as well as End of Transfer (EOT) mode. The PCI 9054 supports Demand Mode DMA for DMA Channel 0.

**PCI Host Capability.** In PCI Initiator mode, the PCI 9054 can assert Type 1 and Type 0 PCI Configuration cycles.

**Six Programmable FIFOs for Zero Wait State Burst Operation.** The following table enumerates the FIFO depth.

Table 1-1. FIFO Depth

FIFO	Length
PCI Initiator Read	16 Lwords
PCI Initiator Write	32 Lwords
PCI Target Read	16 Lwords
PCI Target Write	32 Lwords
DMA Read	32 Lwords
DMA Write	32 Lwords

**PCI ↔ Local Data Transfers up to 132 MB/sec.**

**5 Volt Tolerant Operation.** The PCI 9054 requires 3.3V Vcc. It provides 3.3V signaling with 5V I/O tolerance on both the PCI and Local Buses.

**Local Bus Direct Interface.** Supports Local Bus direct interface to the Motorola MPC850 or MPC860 family, the Intel i960 family, the IBM PPC401 family, and other similar bus-protocol devices.

**Programmable Local Bus.** Runs up to 50 MHz and supports non-multiplexed 32-bit address/data, multiplexed 32-bit, and Slave accesses of 8-, 16-, or 32-bit Local Bus devices.

**Serial EEPROM Interface.** Contains an optional serial EEPROM interface (optional only if a Local processor is being used) that can be used to load configuration information. This is useful for loading information that is unique to a particular adapter (such as the Network ID or the Vendor ID).

**Three PCI-to-Local Address Spaces.** The PCI 9054 supports three PCI-to-Local Address spaces when the PCI 9054 is in PCI Target or PCI Slave mode. These spaces (Space 0, Space 1, and Expansion ROM spaces) allow any PCI Bus Master to access the Local Memory spaces with programmable wait states, bus width, burst capabilities, and so forth.

**Programmable Prefetch Counter.** The PCI 9054 can be programmed to prefetch data during PCI Target and PCI Initiator prefetches (known or unknown size). To perform burst reads, prefetching must be enabled. The prefetch size can be programmed to match the Master burst length, or can be used as Read Ahead mode data. The PCI 9054 reads single data (8, 16, or 32 bit) if the Master initiates a single cycle; otherwise, the PCI 9054 prefetches the programmed size.

**Mailbox Registers.** Contains eight 32-bit Mailbox registers that may be accessed from the PCI or Local Bus.

**Doorbell Registers.** Includes two 32-bit doorbell registers. One asserts interrupts from the PCI Bus to the Local Bus. The other asserts interrupts from the Local Bus to the PCI Bus.

**Big/Little Endian Conversion.** Supports dynamic switching between Big Endian (Address Invariance) and Little Endian (Data Invariance) operations for PCI Target, PCI Initiator, DMA, and internal register accesses on the Local Bus.

The PCI 9054 supports on-the-fly Endian conversion for Space 0, Space 1, and Expansion ROM space. The Local Bus can be Big/Little Endian by using the BIGEND# input pin or programmable internal register configuration. When BIGEND# is asserted, it overrides the internal register configuration during PCI Initiator and internal register accesses on the Local Bus.

**Note:** *The PCI Bus is always Little Endian.*

**Programmable Local Bus Modes.** The PCI 9054 is a PCI Bus Master interface chip that connects a PCI Bus to one of three Local Bus types (M, C, or J mode), selected through mode pins.

The PCI 9054 may be connected to any Local Bus with a similar design with little or no glue logic. Table 1-2 lists the three modes.

**Table 1-2. Programmable Local Bus Modes**

Mode	Description
M	32-bit address/32-bit data, non-multiplexed direct connect interface to MPC850 or MPC860
C	32-bit address/32-bit data, non-multiplexed
J	32-bit address/32-bit data, multiplexed

**Clock.** The Local Bus interface runs from a Local clock to provide the necessary internal clocks. This clock runs asynchronously to the PCI clock.

**Read Ahead Mode.** Supports Read Ahead mode, where prefetched data can be read from the PCI 9054 internal PCI Target Read FIFO instead of from the Local Bus. The address must be subsequent to the previous address and 32-bit aligned (next address = current address + 4). This feature allows for increased bandwidth and reduced data latency.

**M Mode of Operation.** The PCI 9054 is designed with a seamless interface to the Motorola MPC850 or MPC860 RISC processors. The PCI 9054 communicates with the MPC850 or MPC860 using five possible Data-Transfer modes:

- Configuration Register Access
- PCI Initiator Operation
- PCI Target Operation
- DMA Operation
- IDMA/SDMA Operation

**C and J Modes of Operation.** The PCI 9054 is designed with a seamless interface to the Intel i960 and the IBM PPC401 family RISC processors. The PCI 9054 communicates with these processors using four possible Data-Transfer modes:

- Configuration Register Access
- PCI Initiator Operation
- PCI Target Operation
- DMA Operation

**Interrupt Generator.** Can assert PCI and Local interrupts from external and internal sources.

**Unaligned DMA Transfer Support.** Can transfer data on any byte-boundary combination of the PCI and Local Address spaces.

**Keep Bus Mode (M Mode).** The PCI 9054 can be programmed to keep the PCI Bus by generating wait state(s) if the PCI Target Write FIFO becomes full. The PCI 9054 can also be programmed to keep the Local Bus (BB# asserted) if the PCI Target Write FIFO becomes empty or the PCI Target Read FIFO becomes full. The Local Bus is dropped in either case when the Local Bus Latency Timer is enabled and expires.

**Keep Bus Mode (C and J Modes).** The PCI 9054 can be programmed to keep the PCI Bus by generating wait state(s) if the PCI Target Write FIFO becomes full. The PCI 9054 can also be programmed to keep the Local Bus (LHOLD asserted) if the PCI Target Write FIFO becomes empty or the PCI Target Read FIFO becomes full. The Local Bus is dropped in either case when the Local Bus Latency Timer is enabled and expires.

**New Capabilities Structure.** Supports New Capabilities registers to define additional capabilities of the PCI functions.

**Posted Memory Writes.** Supports the Posted Memory Writes (PMW) for maximum performance and to avoid potential deadlock situations.

**RST# Timing.** Supports response to first configuration accesses after de-assertion of RST# under  $2^{25}$  clocks.

**Subsystem ID and Subsystem Vendor ID.** Contains Subsystem ID and Subsystem Vendor ID in the PCI Configuration Register Space in addition to System and Vendor IDs. The PCI 9054 also contains a permanent Vendor ID (10B5h) and Device ID (9054h).

**Direct Bus Master.** Supports PCI accesses from a Local Bus Master. Burst transfers are supported for memory-mapped devices and single-transfers are supported for memory-mapped and I/O devices. The PCI 9054 also supports PCI Bus interlock (LOCK#) cycles.

**PCI Target.** Supports Burst Memory-Mapped and Single I/O-Mapped accesses to the Local Bus. The Read and Write FIFOs enable high-performance bursting.

### 1.2.5 PCI 9054 Data Assignment Convention

Table 1-3 describes the PCI 9054 data assignment convention.

**Table 1-3. PCI 9054 Data Assignment Convention**

Data Width	PCI 9054 Convention
1 byte (8 bits)	Byte
2 bytes (16 bits)	Word
4 bytes (32 bits)	Lword

#### 1.2.5.1 PCI 9050/9080 Compatibility

#### 1.2.5.2 Pin Compatibility

The PCI 9054 is *not* pin compatible with the PCI 9080 *nor* with the PCI 9050.

#### 1.2.5.3 Register Compatibility

All registers implemented in the PCI 9080 are implemented in the PCI 9054. The PCI 9054 includes many new bit definitions and several new registers. (Refer to Section 11.1 for details.)

The PCI 9054 is *not* register-compatible with the PCI 9050.

## 1.2.6 PCI 9054, PCI 9080, and PCI 9050 Comparison

Table 1-4. Comparison of PCI 9054, PCI 9080, and PCI 9050

Feature	PCI 9054	PCI 9080	PCI 9050
Package Size/Type	176 PQFP, 225 PBGA	208 PQFP	160 PQFP
Number of DMA Channels	2	2	0
Local Address Spaces	3	3	5
PCI Initiator Mode	Yes	Yes	No
Mailbox Registers	Eight 32-bit	Eight 32-bit	Eight 32 bit
Doorbell Registers	Two 32-bit	Two 32-bit	Two 32 bit
Number of FIFOs	6	8	—
FIFO Depth—PCI Target Write and PCI Initiator Write	32 Lwords (128 bytes)	32 Lwords (128 bytes)	—
FIFO Depth—PCI Target Read and PCI Initiator Read	16 Lwords (64 bytes)	16 Lwords (64 bytes)	2
FIFO Depth—DMA Channel 0	32 Lwords (128 bytes) Single bidirectional Read/Write FIFO	32 Lwords (128 bytes) Read and Write FIFOs	N/A
FIFO Depth—DMA Channel 1	16 Lwords (64 bytes) Single bidirectional Read/Write FIFO	16 Lwords (64 bytes) Read and Write FIFOs	N/A
LLOCKo# Pin for Lock Cycles	Yes	Yes	Yes
WAIT# Pin for Wait State Generation	Yes	Yes	No
BPCLKo Pin; Buffered PCI Clock	No	Yes	Yes
DREQ0# and DACK0# Pins for Demand Mode DMA Support	Yes (One channel only)	Yes	No
Register Addresses	Identical to the PCI 9080 except the PCI 9054 contains additional registers related to added functionality	—	—
Big Endian ↔ Little Endian Conversion	Yes	Yes	Yes
PCI Specification v2.1 Deferred Reads	Yes	Yes	Yes
PCI Specification v2.2 PCI Power Management, PCI Hot Plug Compliant, CompactPCI Hot Swap Compliant	Yes	No	No
PCI v2.2 VPD Support	Yes	No	No
Programmable Prefetch Counter	Yes	Yes	Yes
Memory Write and Invalidate Cycle	Yes	Yes	No
Additional Device and Vendor ID Registers	Yes	Yes	No
I <sub>2</sub> O Messaging Unit	Yes	Yes	No
Core and Local Bus Vcc	3.3V	5V	5V
PCI Bus Vcc	3.3V	3.3/5V	5V
3.3V PCI Bus and Local Bus Signaling	Yes	Yes (if PCI Vcc is 3.3V)	No
5V Tolerant PCI Bus and Local Bus	Yes	Yes (if PCI Vcc is 5V)	Yes
Serial EEPROM Support	2K bit, 4K bit devices	1K bit, 2K bit devices	—
Serial EEPROM Read Control	Reads allowed via Vital Product Data Function (refer to Section 10)	Reads allowed via Serial EEPROM Control Register (CNTRL)	1K bit device

## 1.2.7 PCI 9054 Signal Listing (M, C, or J Modes)

Table 1-5. PCI 9054 PCI Signal Listing (M, C, or J Modes)

Symbol	Bus Mode	Signal Name	Total Pins	Function
AD[31:0]	All	Address and Data	32	All multiplexed on the same PCI pins. The Bus transaction consists of an Address phase, followed by one or more Data phases. The PCI 9054 supports both Read and Write bursts.
C/BE[3:0]#	All	Bus Command and Byte Enables	4	All multiplexed on the same PCI pins. During the Address phase of a transaction, defines the bus command. During the Data phase, used as byte enables. Refer to the PCI spec for further detail if needed.
DEVSEL#	All	Device Select	1	When actively driven, indicates the driving device has decoded its address as the Target of the current access. As an input, indicates whether any device on the bus is selected.
FRAME#	All	Cycle Frame	1	Driven by the current Master to indicate beginning and duration of an access. FRAME# is asserted to indicate a bus transaction is beginning. While FRAME# is asserted, Data transfers continue. When FRAME# is de-asserted, the transaction is in the final Data phase.
GNT#	All	Grant	1	Indicates to the agent that access to the bus is granted. Every Master has its own REQ# and GNT#.
IDSEL	All	Initialization Device Select	1	Used as a chip select during Configuration Read and Write transactions.
INTA#	All	Interrupt A	1	PCI Interrupt request.
IRDY#	All	Initiator Ready	1	Indicates ability of the initiating agent (Bus Master) to complete the current Data phase of the transaction.
LOCK#	All	Lock	1	Indicates an atomic operation that may require multiple transactions to complete.
PAR	All	Parity	1	Even parity across AD[31:0] and C/BE[3:0]#. All PCI agents require parity generation. PAR is stable and valid one clock after the Address phase. For Data phases, PAR is stable and valid one clock after either IRDY# is asserted on a Write transaction or TRDY# is asserted on a Read transaction. Once PAR is valid, it remains valid until one clock after current Data phase completes.
PCLK	All	Clock	1	Provides timing for all transactions on PCI and is an input to every PCI device. The PCI 9054 operates up to 33 MHz.
PERR#	All	Parity Error	1	Reports data parity errors during all PCI transactions, except during a special cycle.
PME#	All	Power Management Event	1	Wake-up event interrupt.
REQ#	All	Request	1	Indicates to arbiter that this agent must use the bus. Every Master has its own GNT# and REQ#.
RST#	All	Reset	1	Used to bring PCI-specific registers, sequencers, and signals to a consistent state.
SERR#	All	Systems Error	1	Reports address parity errors, data parity errors on the Special Cycle command, or any other system error where the result is catastrophic.
STOP#	All	Stop	1	Indicates the current Target is requesting that the Master stop the current transaction.
TRDY#	All	Target Ready	1	Indicates ability of the Target agent (selected device) to complete the current Data phase of the transaction.

Table 1-6. PCI 9054 Local Signal Listing (M, C, or J Modes)

Symbol	Bus Mode	Signal Name	Total Pins	Function
ADS#	C, J	Address Strobe	1	Indicates valid address and start of new Bus access. Asserted for first clock of Bus access.
ALE	J	Address Latch Enable	1	Asserted during Address phase and de-asserted before Data phase and before next LCLK rising edge.
BB#	M	Bus Busy	1	As an input, the PCI 9054 monitors this signal to determine whether an external Master has ended a Bus cycle. As an output, the PCI 9054 asserts this signal after an external arbiter has granted ownership of the Local Bus and BB# is inactive from another Master. Signal requires an external pull-up resistor value of 510W be applied to guarantee a fast transition to the inactive state when the PCI 9054 relinquishes ownership of the Local Bus.
BDIP#	M	Burst Data in Progress	1	As an input, driven by the Bus Master during a Burst transaction. The Master de-asserts before the last Data phase on the bus. As an output, driven by the PCI 9054 during the Data phase of a Burst transaction. The PCI 9054 de-asserts before the last Burst Data phase on the bus.
BG#	M	Bus Grant	1	Asserted by the Local Bus arbiter in response to BR#. Indicates the requesting Master is next.
BI#	M	Burst Inhibit	1	Whenever BR# is asserted, indicates that the Target device does not support Burst transactions.
BIGEND#/ WAIT#	All M	Big Endian Select WAIT Input/Output Select ( <i>WAIT# is available at this location only in M mode</i> )	1	Multiplexed input/output pin. Can be asserted during the Local Bus Address phase of a PCI Initiator transfer or Configuration register access to specify use of Big Endian Byte ordering. Big Endian Byte order for PCI Initiator transfers or Configuration register accesses is also programmable through the Configuration registers. If wait is selected, then PCI 9054 issues WAIT# when it is a Master on the Local Bus and has internal wait states setup. As a Slave, the PCI 9054 accepts WAIT# as an input from the Bus Master.
BLAST#	C, J	Burst Last	1	Signal driven by the current Local Bus Master to indicate the last transfer in a Bus access.
BR#	M	Bus Request	1	Asserted by the Master to request use of the Local Bus. The Local Bus arbiter asserts BG# when the Master is next in line for bus ownership.
BREQi	C, J	Bus Request	1	Asserted to indicate a Local Bus Master requires the bus. If enabled through the PCI 9054 Configuration registers, the PCI 9054 releases the bus during a DMA transfer if this signal is asserted.
BREQo	C, J	Bus Request Out	1	Asserted to indicate the PCI 9054 requires the bus to perform a PCI Target PCI-to-Local Bus access while a PCI Initiator access is pending on the Local Bus. Can be used with external logic to assert backoff to a Local Bus Master. Operational parameters are set up through the PCI 9054 Configuration registers.
BTERM#	C, J	Burst Terminate	1	As input to the PCI 9054: For processors that burst up to four Lwords. If the Bterm Mode bit is disabled through the PCI 9054 Configuration registers, the PCI 9054 also bursts up to four Lwords. If enabled, the PCI 9054 continues to burst until a BTERM# input is asserted. BTERM# is a Ready input that breaks up a Burst cycle and causes another Address cycle to occur. Used in conjunction with the PCI 9054 programmable wait state generator. As output from the PCI 9054: Asserted, along with READY#, to request break up of a burst and start of a new Address cycle (PCI Aborts only).

Table 1-6. PCI 9054 Local Signal Listing (M, C, or J Modes) (Continued)

Symbol	Bus Mode	Signal Name	Total Pins	Function
BURST#	M	Burst	1	As an input, driven by the Master along with address and data indicating a Burst transfer is in progress. As an output, driven by the PCI 9054 along with address and data indicating a Burst transfer is in progress.
CCS#	All	Configuration Register Select	1	Internal PCI 9054 registers are selected when CCS# is asserted low.
DACK0#	All			See USERi/DACK0#/LLOCKi#.
DEN#	J	Data Enable	1	Used in conjunction with DT/R# to provide control for data transceivers attached to the Local Bus.
DMPAF	All			See MDREQ#/DMPAF/EOT#.
DP[0:3]	M	Data Parity	4	Parity is even for each of up to four byte lanes on the Local Bus. Parity is checked for writes or reads to the PCI 9054. Parity is asserted for reads from or writes by the PCI 9054.
DP[3:0]	C, J			
DREQ0#	All			See USERo/DREQ0#/LLOCKo#.
DT/R#	J	Data Transmit/Receive	1	Used in conjunction with DEN# to provide control for data transceivers attached to the Local Bus. When asserted, indicates the PCI 9054 receives data.
EECS	All	Serial EEPROM Chip Select	1	Serial EEPROM Chip Select.
EEDI/ EEDO	All	Serial EEPROM Data IN/ Serial EEPROM Data OUT	1	Multiplexed Write/Read data to a serial EEPROM pin.
EESK	All	Serial Data Clock	1	Serial EEPROM clock pin.
ENUM#	All	Enumeration	1	Interrupt output asserted when an adapter using PCI 9054 has been freshly inserted or is ready to be removed from a PCI slot.
EOT#	All			See MDREQ#/DMPAF/EOT#.
LA[28:2]	J	Local Address Bus	27	Carries the middle 27 bits of the physical address bus. During bursts, it is incremented to indicate successive Data cycles. The lowest two bits, LA[3:2], carry the Word address of the 32-bit Memory address. All bits are incremented during a Burst access.
LA[0:31]	M	Address Bus	32	Carries the 32 bits of the physical Address Bus.
LA[31:2]	C		30	Carries the upper 30 bits of physical Address Bus. During bursts, LA[31:2] increment to indicate successive Data cycles.
LAD[31:0]	J	Address/Data Bus	32	During an Address phase, the bus carries the upper 30 bits of the physical Address Bus. During a Data phase, the bus carries 32 bits of data.

Table 1-6. PCI 9054 Local Signal Listing (M, C, or J Modes) (Continued)

Symbol	Bus Mode	Signal Name	Total Pins	Function
LBE[3:0]#	C	Byte Enables	4	<p>Encoded, based on the bus-width configuration, as follows:</p> <p><b>32-Bit Bus</b> The four byte enables indicate which of the four bytes are active during a Data cycle: BE3# Byte Enable 3—LD[31:24] BE2# Byte Enable 2—LD[23:16] BE1# Byte Enable 1—LD[15:8] BE0# Byte Enable 0—LD[7:0]</p> <p><b>16-Bit Bus</b> BE3#, BE1# and BE0# are encoded to provide BHE#, LA1, and BLE#, respectively: BE3# Byte High Enable (BHE#)—LD[15:8] BE2# not used BE1# Address bit 1 (LA1) BE0# Byte Low Enable (BLE#)—LD[7:0]</p> <p><b>8-Bit Bus</b> BE1# and BE0# are encoded to provide LA1 and LA0, respectively: BE3# not used BE2# not used BE1# Address bit 1 (LA1) BE0# Address bit 0 (LA0)</p>
	J			<p>Encoded, based on the bus-width configuration, as follows:</p> <p><b>32-Bit Bus</b> The four byte enables indicate which of the four bytes are active during a Data cycle: BE3# Byte Enable 3—LAD[31:24] BE2# Byte Enable 2—LAD[23:16] BE1# Byte Enable 1—LAD[15:8] BE0# Byte Enable 0—LAD[7:0]</p> <p><b>16-Bit Bus</b> BE3#, BE1# and BE0# are encoded to provide BHE#, LAD1, and BLE#, respectively: BE3# Byte High Enable (BHE#)—LAD[15:8] BE2# not used BE1# Address bit 1 (LAD1) BE0# Byte Low Enable (BLE#)—LAD[7:0]</p> <p><b>8-Bit Bus</b> BE1# and BE0# are encoded to provide LAD1 and LAD0, respectively: BE3# not used BE2# not used BE1# Address bit 1 (LAD1) BE0# Address bit 0 (LAD0)</p>
LCLK	All	Local processor Clock	1	Local clock input.
LD[0:31]	M	Data Bus	32	Carries 8-, 16-, or 32-bit data quantities, depending upon bus-width configuration. All Master accesses to the PCI 9054 are 32 bits only.
LD[31:0]	C			
LEDon/ LEDin	All	LEDon/LEDin	1	As an output, acts as the Hot Swap board indicator LED. As an input, monitors the CompactPCI board latch status.
LFRAME#	All	PCI Buffered FRAME# Signal	1	Could be used to monitor PCI Bus activity. <i>Available only on the PBGA package.</i>
LHOLD	C, J	Hold Request	1	Asserted to request use of the Local Bus. The Local Bus arbiter asserts LHOLDA when control is granted.

Table 1-6. PCI 9054 Local Signal Listing (M, C, or J Modes) (Continued)

Symbol	Bus Mode	Signal Name	Total Pins	Function															
LHOLDA	C, J	Hold Acknowledge	1	Asserted by the Local Bus arbiter when control is granted in response to LHOLD. Bus should not be granted to the PCI 9054 unless requested by LHOLD.															
LINT#	All	Local Interrupt	1	As an input to the PCI 9054, when asserted low, causes a PCI interrupt. As an output, a synchronous level output that remains asserted as long as an interrupt condition exists. If edge level interrupt is required, disabling and then enabling Local interrupts through INTCSR creates an edge if an interrupt condition still exists or a new interrupt condition occurs.															
LLOCKi#	All			See USERi/DACK0#/LLOCKi#.															
LLOCKo#	All			See USERo/DREQ0#/LLOCKo#.															
LRESETo#	All	Local Bus Reset Out	1	Asserted when the PCI 9054 chip is reset. Can be used to drive RESET# input of a Local processor.															
LSERR#	C, J	System Error Interrupt Output	1	Synchronous level output asserted when the PCI Bus Target Abort bit is set (PCISR[11]=1) or Received Master Abort bit is set (PCISR[13]=1). If edge level interrupt is required, disabling and then enabling LSERR# interrupts through the interrupt/control status creates an edge if an interrupt condition still exists or a new interrupt condition occurs.															
LW/R#	C, J	Write/Read	1	Asserted low for reads and high for writes.															
MDREQ#/ DMPAF/ EOT#	M All All	IDMA Data Transfer Request (MDREQ# is available at this location in M mode only) PCI Initiator Programmable Almost Full End of Transfer for Current DMA Channel	1	Multiplexed input or output pin. MDREQ#: IDMA M mode Data transfer request start. Always asserted, indicating Data transfer should start. De-asserted only when the PCI Initiator FIFO becomes full. Programmable through a Configuration register. DMPAF: PCI Initiator Write FIFO Almost Full status output. Programmable through a Configuration register. EOT#: Terminates the current DMA transfer. <b>Note:</b> EOT# serves as a general purpose EOT. Before asserting EOT#, user should be aware of DMA channel activity.															
MODE[1:0]	All	Bus Mode	2	Selects the PCI 9054 bus operation mode: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Mode 0</th> <th>Mode 1</th> <th>Bus Mode</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>M</td> </tr> <tr> <td>1</td> <td>0</td> <td>J</td> </tr> <tr> <td>0</td> <td>1</td> <td>Reserved</td> </tr> <tr> <td>0</td> <td>0</td> <td>C</td> </tr> </tbody> </table>	Mode 0	Mode 1	Bus Mode	1	1	M	1	0	J	0	1	Reserved	0	0	C
Mode 0	Mode 1	Bus Mode																	
1	1	M																	
1	0	J																	
0	1	Reserved																	
0	0	C																	
RD/WR#	M	Read/Write	1	Asserted high for reads and low for writes.															
READY#	C, J	Ready Input/Output	1	When the PCI 9054 is a Bus Master, indicates that Read data on the bus is valid or that a Write Data transfer is complete. Used in conjunction with the PCI 9054 programmable wait state generator. When a Local Bus access is made to the PCI 9054, indicates that Read data on the bus is valid or that a Write Data transfer is complete.															
RETRY#	M	Retry	1	Driven by the PCI 9054 when it is a Slave to indicate a Local Master must back off and restart the cycle. In Deferred Read mode, indicates the Master should return for requested data.															
TA#	M	Transfer Acknowledge	1	As an output, when a Local Bus access is made to the PCI 9054, indicates a Write Data transfer can complete or that Read data on the bus is valid. As an input, when the PCI 9054 is a Bus Master, indicates a Write Data transfer is complete or that Read data on the bus is valid.															

Table 1-6. PCI 9054 Local Signal Listing (M, C, or J Modes) (Continued)

Symbol	Bus Mode	Signal Name	Total Pins	Function
TEA#	M	Transfer Error Acknowledge	1	Driven by the Target device, indicating an error condition has occurred during a Bus cycle.
TEST	All	Test Pin	1	Pulled high for test and low for normal operation. When pulled high: All outputs except USER <sub>o</sub> /DREQ <sub>0</sub> #/LLOCK <sub>o</sub> # and LED <sub>o</sub> /LED <sub>in</sub> are placed in tri-state. USER <sub>o</sub> /DREQ <sub>0</sub> #/LLOCK <sub>o</sub> # provides NANDTREE output.
TS#	M	Address Strobe	1	Indicates the valid address and start of new Bus access. Asserted for the first clock of a Bus access.
TSIZ[0:1]	M	Transfer Size	2	Driven by current Master along with the address, indicating the data-transfer size. Refer to Section 3.4.3.5.3 for more information.
USER <sub>i</sub> / DACK <sub>0</sub> #/ LLOCK <sub>i</sub> #	All	User Input Demand Mode DMA Acknowledge Local Lock Input	1	Multiplexed input/output pin. USER <sub>i</sub> : General-purpose input that can be read by way of the PCI 9054 Configuration registers. DACK <sub>0</sub> #: When a channel is programmed through the Configuration registers to operate in Demand mode, this output indicates a DMA transfer is being executed. DACK <sub>0</sub> # corresponds to PCI 9054 DMA Ch 0. LLOCK <sub>i</sub> #: Indicates an atomic operation that may require multiple transactions to complete. Used by the PCI 9054 for direct Local access to the PCI Bus.
USER <sub>o</sub> / DREQ <sub>0</sub> #/ LLOCK <sub>o</sub> #	All	User Output Demand DMA Request Local Lock Output	1	Multiplexed input/output pin. USER <sub>o</sub> : General-purpose output controlled from the PCI 9054 Configuration registers. DREQ <sub>0</sub> #: When a channel is programmed through the Configuration registers to operate in Demand mode, this input serves as a DMA request. DREQ <sub>0</sub> # corresponds to PCI 9054 DMA Ch 0. LLOCK <sub>o</sub> #: Indicates an atomic operation for a PCI Target PCI-to-Local Bus access may require multiple transactions to complete.
VDD	All	Power (+3.3V)	15	Three-volt power supply pins for core and I/O buffers. Liberal .01 to .1 μF decoupling capacitors should be placed near the PCI 9054.
VSS	All	Ground	12	Ground pins.
WAIT#	C, J	Wait Input/Output	1	As an input, can be asserted to cause the PCI 9054 to insert wait states for Local PCI Initiator accesses to the PCI Bus. Can be thought of as a Ready input from an external Master for PCI Initiator accesses. As an output, asserted by the PCI 9054 when internal wait state generator causes wait states. Can be thought of as an output providing PCI 9054 Ready status.
WAIT#	M	Wait Input/Output	1	See BIGEND#/WAIT#.

## 2 M MODE BUS OPERATION

### 2.1 PCI BUS CYCLES

The PCI 9054 is compliant with PCI Specification v2.2. Refer to PCI Specification v2.2 for specific PCI Bus functions.

#### 2.1.1 PCI Target Command Codes

As a Target, the PCI 9054 allows access to the PCI 9054 internal registers and the Local Bus, using the commands listed in Table 2-1.

All Read or Write accesses to the PCI 9054 can be Byte, Word, or Lword (longword) accesses, defined as 32 bit. All memory commands are aliased to basic memory commands. All I/O accesses to the PCI 9054 are decoded to an Lword boundary. Byte enables are used to determine which bytes are read or written. An I/O access with illegal byte enable combinations is terminated with a Target Abort.

**Table 2-1. PCI Target Command Codes**

Command Type	Code (C/BE[3:0]#)
I/O Read	0010 (2h)
I/O Write	0011 (3h)
Memory Read	0110 (6h)
Memory Write	0111 (7h)
Configuration Read	1010 (Ah)
Configuration Write	1011 (Bh)
Memory Read Multiple	1100 (Ch)
PCI Dual Address Cycle	1101 (Dh)
Memory Read Line	1110 (Eh)
Memory Write and Invalidate	1111 (Fh)

#### 2.1.2 PCI Master Command Codes

The PCI 9054 can access the PCI Bus to perform DMA or PCI Initiator Local-to-PCI Bus transfers. During a PCI Initiator or DMA transfer, the command code assigned to the PCI 9054 internal register location (CNTRL[15:0]) is used as the PCI command code (except for Memory Write and Invalidate mode for DMA cycles where (DMPBAM[9]=1). Table 2-2 through Table 2-5 list various PCI Master Command codes.

**Notes:** Programmable internal registers determine PCI command codes when the PCI 9054 is the Master. DMA cannot perform I/O or configuration accesses.

#### 2.1.2.1 DMA Master Command Codes

DMA controllers of the PCI 9054 can assert the Memory cycles listed in Table 2-2.

**Table 2-2. DMA Master Command Codes**

Command Type	Code (C/BE[3:0]#)
Memory Read	0110 (6h)
Memory Write	0111 (7h)
Memory Read Multiple	1100 (Ch)
PCI Dual Address Cycle	1101 (Dh)
Memory Read Line	1110 (Eh)
Memory Write and Invalidate	1111 (Fh)

#### 2.1.2.2 Direct Local-to-PCI Command Codes

For direct Local-to-PCI Bus accesses, the PCI 9054 asserts the cycles listed in Table 2-3 through Table 2-5.

**Table 2-3. Local-to-PCI Memory Access**

Command Type	Code (C/BE[3:0]#)
Memory Read	0110 (6h)
Memory Write	0111 (7h)
Memory Read Multiple	1100 (Ch)
PCI Dual Address Cycle	1101 (Dh)
Memory Read Line	1110 (Eh)
Memory Write and Invalidate	1111 (Fh)

**Table 2-4. Local-to-PCI I/O Access**

Command Type	Code (C/BE[3:0]#)
I/O Read	0010 (2h)
I/O Write	0011 (3h)

**Table 2-5. Local-to-PCI Configuration Access**

Command Type	Code (C/BE[3:0]#)
Configuration Memory Read	1010 (Ah)
Configuration Memory Write	1011 (Bh)

### 2.1.3 PCI Arbitration

The PCI 9054 asserts REQ# to request the PCI Bus. The PCI 9054 can be programmed using the PCI Request Mode bit (MARBR[23]) to de-assert REQ# when it asserts FRAME# during a Bus Master cycle, or to keep REQ# asserted for the entire Bus Master cycle. The PCI 9054 always de-asserts REQ# for a minimum of two PCI clocks between Bus Master ownership that includes a Target disconnect.

The PCI Initiator Write Delay bits (DMPBAM[15:14]) can be programmed to delay the PCI 9054 from asserting PCI REQ# during a PCI Initiator Write cycle. DMPBAM can be programmed to wait 0, 4, 8, or 16 PCI Bus clocks after the PCI 9054 has received its first Write data from the Local Bus Master and is ready to begin the PCI Write transaction. This function is useful in applications where a Local Master is bursting and a Local Bus clock is slower than the PCI Bus clock. This allows Write data to accumulate in the PCI 9054 PCI Initiator Write FIFO, which provides for better use of the PCI Bus.

## 2.2 LOCAL BUS CYCLES

The PCI 9054 interfaces a PCI Host bus to several Local Bus types, as listed in Table 2-6 and Table 2-7. It operates in one of three modes, selected through MODE[1:0] (PQFP—Pins 157 and 156; PBGA—Pins B7 and E8), corresponding to M, J, and C bus types.

Table 2-6. Local Bus Types (176-Pin PQFP)

Pin 157	Pin 156	Mode	Bus Type
1	1	M	32-bit non-multiplexed
1	0	Reserved	—
0	1	J	32-bit multiplexed
0	0	C	32-bit non-multiplexed

Table 2-7. Local Bus Types (225-Pin PBGA)

Pin B7	Pin E8	Mode	Bus Type
1	1	M	32-bit non-multiplexed
1	0	Reserved	—
0	1	J	32-bit multiplexed
0	0	C	32-bit non-multiplexed

In M mode, the PCI 9054 provides a direct connection to the MPC850 or MPC860 address and data lines, regardless of the PCI 9054 Little Endian or Big Endian modes.

### 2.2.1 Local Bus Arbitration

The PCI 9054 asserts BR# to request the Local Bus. It owns the Local Bus when BG# is asserted. Upon receiving BG#, the PCI 9054 waits for BB# to de-assert. The PCI 9054 then asserts BB# at the next rising edge of the Local clock after acknowledging BB# is de-asserted (no other device is acting as the Local Bus Master). The PCI 9054 continues to assert BB# while acting as the Local Bus Master (*that is*, it holds the bus until instructed to release BB#) when the Local Bus Latency Timer is enabled and expires (MARBR[7:0]) or the transaction is complete.

**Note:** The Local Bus Pause Timer applies only to DMA operation. It does **not** apply to PCI Target operation.

### 2.2.2 PCI Initiator

Local Bus cycles can be Single or Burst cycles. As a Local Bus Target, the PCI 9054 allows access to the PCI 9054 internal registers and the PCI Bus.

Local Bus PCI Initiator accesses to the PCI 9054 must be for a 32-bit non-pipelined bus. Non-32-bit PCI Initiator accesses to the PCI 9054 require simple external logic (latch array to combine data into a 32-bit bus).

### 2.2.3 PCI Target

The PCI Bus Master reads from and writes to the Local Bus (the PCI 9054 is a PCI Bus Target and a Local Bus Master).

### 2.2.4 Wait State Control

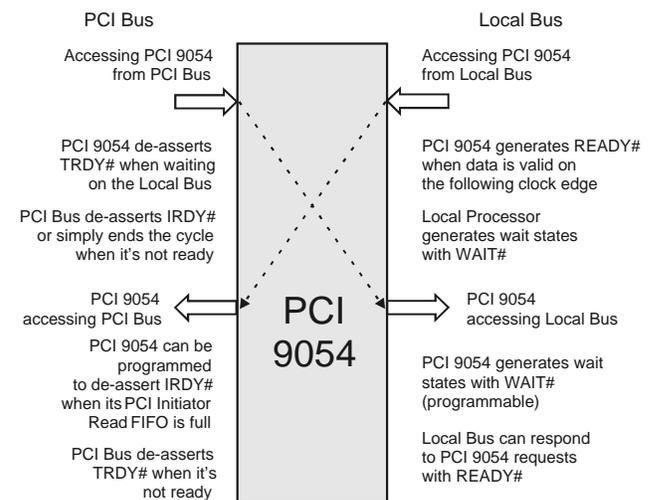


Figure 2-1. Wait States

**Note:** The figure represents a sequence of Bus cycles.

The TA# signal overwrites the programmable wait state counter, and could be used to introduce additional wait states.

### 2.2.4.1 Wait States—Local Bus

In PCI Initiator mode, when accessing the PCI 9054 registers, the PCI 9054 acts as a Local Bus Slave. The PCI 9054 asserts external wait states with the TA# signal.

In PCI Target and DMA modes, the PCI 9054 acts as a Local Bus Master. The Internal Wait States bit(s) (LBRD0[21:18, 5:2], (LBRD1[5:2]), DMAMODE0[5:2], and/or DMAMODE1[5:2]) can be used to program the number of internal wait states between the first address-to-data (and subsequent data-to-data in Burst mode).

In PCI Target and DMA modes, if TA# is enabled and active, it continues the Data transfer, regardless of the wait state counter.

### 2.2.4.2 Wait States—PCI Bus

To insert PCI Bus wait state(s), the PCI Bus Master throttles IRDY# and the PCI Bus Slave throttles TRDY#.

## 2.2.5 Burst Mode and Continuous Burst Mode (Bterm “Burst Terminate” Mode)

**Note:** In the following sections, Bterm refers to the PCI 9054 internal register bit, and BTERM# refers to the PCI 9054 external signal.

### 2.2.5.1 Burst and Bterm Modes

**Table 2-8. Burst and Bterm on the Local Bus**

Mode	Burst	Bterm	Result
Single Cycle	0	0	One TS# per data (default).
Single Cycle	0	1	One TS# per data.
Burst-4	1	0	One TS# per four data (recommended for MPC850 or MPC860).
Burst Forever	1	1	PCI Target or DMA—One TS# per Burst data or until BI# is asserted (refer to Section 2.2.5.2.1).

**Note:** BI# is supported in Burst-4 mode. Refer to the MPC850 or MPC860 data manual.

On the Local Bus, BTERM# is not supported, but the Bterm bit can be used to gain maximum performance and data throughput.

- If the Burst Mode bit is enabled, but the Bterm Mode bit is disabled, then the PCI 9054 bursts four Lwords. BDIP# is de-asserted at the last Lword transfer before its completion (LA[3:2]=11) and a new TS# is asserted at the first Lword (LA[3:2]=00) of the next burst.
- If the Burst Mode and Bterm Mode bits are both enabled, then the PCI 9054 bursts until the transfer counter counts to “0”, the Local Latency Timer is enabled and expires, the EOT function is introduced, or DREQ0# is de-asserted during DMA transactions. For PCI Target transactions, the PCI 9054 bursts until BI# is asserted, implying a new TS# is required, or the Local Latency Timer is enabled and expires. The PCI 9054 does not release bus ownership during BI# assertion. BDIP# output is supported in Burst Forever mode with a different behavior than MPC860 protocol. Refer to Section 2.2.5.2.

**Notes:** If Address Increment is disabled, the DMA transaction bursts beyond four Lwords.

If the Bterm Mode bit is disabled, the PCI 9054 performs the following:

- **32-bit Local Bus**—Bursts up to four Lwords
- **16-bit Local Bus**—Bursts up to four Lwords
- **8-bit Local Bus**—Bursts up to four Lwords

In every case, it transfers 16 bytes of data.

### 2.2.5.2 Burst-4 Lword Mode

If the Burst Mode bit is enabled and the Bterm Mode bit is disabled, bursting can start only on a 16-byte boundary and continue up to the next 16-byte address boundary. After data before the boundary is transferred, the PCI 9054 asserts a new Address cycle (TS#).

**Table 2-9. Burst-4 Lword Mode**

Bus Width	Burst
32 bit	Four Lwords or up to a quad Lword boundary (LA3, LA2 = 11)
16 bit	Eight words or up to a quad Lword boundary (LA2, LA1 = 11)
8 bit	Sixteen bytes or up to a quad Lword boundary (LA1, LA0 = 11)

### 2.2.5.2.1 Continuous Burst Mode (Bterm “Burst Terminate” Mode)

If both the Burst and Bterm Mode bits are enabled, the PCI 9054 can operate beyond the Burst-4 Lword mode.

Bterm mode enables the PCI 9054 to perform long bursts to special external M-mode interface devices that can accept bursts of longer than four Lwords. The PCI 9054 asserts one Address cycle and continues to burst data. The external address is incremented during bursts. If a device requires a new Address cycle, it can assert BI# input anywhere after the first Data phase to cause the PCI 9054 to assert a new Address cycle (TS#). The BI# input acknowledges the current Data transfer and requests that a new Address cycle be asserted (TS#), for the next Data transfer. If the Bterm Mode bit is enabled, the PCI 9054 de-asserts BURST# only if its Read FIFO is full, its Write FIFO is empty, or if a transfer is complete. If the transfer starts on a non quad-word aligned address, the PCI 9054 single cycles the data until the next quad-word aligned address and bursters forever the remainder of the data.

The PCI 9054\_AB supports the BDIP# signal for continuous bursts greater than four Lwords, which differs from MPC850/860 protocol. When Bterm and Burst functions are enabled for PCI Target and/or DMA transactions, and Slow Terminate mode is enabled for DMA, the PCI 9054 drives the BDIP# signal high until one data before the last burst transfer data. On the last data transfer, the PCI 9054 asserts BDIP# low, indicating the last transfer of the burst transaction. BDIP# is driven low for as long as data is valid on the bus and is de-asserted on the following clock, along with the BURST# signal.

During Burst Forever Write transactions in M mode, the PCI 9054 passes all bytes from the PCI Bus to the Local Bus, if C/BE# begins to toggle on the nonquad-aligned address by keeping TSIZ[0:1] at a constant value of 0 and issues TS# for the toggled address. However, if C/BE# toggles on the quad-word-aligned address, the PCI 9054 begins the Local Bus Burst and toggles TSIZ[0:1], along with TS#, for all data that follows when a burst resumes. It is recommended to keep all bytes enabled during a PCI Write Burst transaction.

### 2.2.5.3 Partial Lword Accesses

Lword accesses, in which not all byte enables are asserted, are broken into Single-Cycle accesses. Burst start addresses can be any quad-word boundary. The PCI 9054 first performs a Single cycle, if the Burst Start Address in a PCI Target or DMA transfer is not aligned to a quad-word or Lword boundary. It then starts to burst on the quad-word boundary if there is remaining data that is not a whole Lword during DMA (*for example*, it results in a Single cycle at the end).

### 2.2.6 Local Bus Read Accesses

For all Single-Cycle Local Bus Read accesses, the PCI 9054 reads only bytes corresponding to byte enables requested by the PCI Initiator. For all Burst Read cycles, the PCI 9054 passes all the bytes and can be programmed to:

- Prefetch
- Perform Read Ahead mode
- Generate internal wait states
- Enable external wait control (TA# input)
- Enable type of Burst mode to perform

### 2.2.7 Local Bus Write Accesses

For Local Bus writes, only bytes specified by a PCI Bus Master or the PCI 9054 DMA controller are written.

### 2.2.8 PCI Target Accesses to 8- or 16-Bit Local Bus

Direct PCI access to an 8- or 16-bit Local Bus results in the PCI Bus Lword being broken into multiple Local Bus transfers. For each transfer, byte enables are encoded to provide Transfer Size bits (TSIZ[0:1]).

### 2.2.9 Local Bus Data Parity

Generation or use of Local Bus data parity is optional. Signals on the data parity pins do not affect operation of the PCI 9054. The PCI Bus parity checking and generation is independent of the Local Bus parity checking and generation. PCI Bus parity checking may result in assertion of PERR#, a PCI Bus system

error (SERR#), or other means of PCI Bus transfer termination as a result of the parity error on the PCI data address, command code, and byte enables. The Local Bus Parity Check is passive and only provides parity information to the Local processor during PCI Initiator, PCI Target, and DMA transfers.

There is one data parity pin for each byte lane of the PCI 9054 data bus (DP[0:3]). “Even data parity” is asserted for each lane during Local Bus reads from the PCI 9054 and during PCI 9054 Master writes to the Local Bus.

Even data parity is checked during Local Bus writes to the PCI 9054 and during PCI 9054 reads from the Local Bus. Parity is checked for each byte lane with an asserted byte enable. If a parity error is detected, TEA# is asserted in the Clock cycle following the data being checked.

Parity is checked for PCI Target reads, PCI Initiator writes, and DMA Local Bus reads. The PCI 9054 sets a status bit and asserts an interrupt (TEA#) in the clock cycle following data being checked if a parity error is detected. However, the Data Parity Error Status bit and interrupt are never set or asserted unless the TA# signal is active and asserted low. This applies only when the TA# signal is disabled in the PCI 9054 register. A workaround for this is to disable the TA# Enable bit and externally pull TA# low.

## 2.3 BIG ENDIAN/LITTLE ENDIAN

### 2.3.1 PCI Bus Little Endian Mode

PCI Bus is a Little Endian bus (*that is*, the address is invariant and data is Lword-aligned to the lowermost byte lane).

**Table 2-10. PCI Bus Little Endian Byte Lanes**

Byte Number	Byte Lane
0	AD[7:0]
1	AD[15:8]
2	AD[23:16]
3	AD[31:24]

### 2.3.2 Local Bus Big/Little Endian Mode

The PCI 9054 Local Bus can be programmed to operate in Big or Little Endian mode.

**Table 2-11. Byte Number and Lane Cross-Reference**

Byte Number		Byte Lane
Big Endian	Little Endian	
3	0	LD[31:24]
2	1	LD[23:16]
1	2	LD[15:8]
0	3	LD[7:0]

**Table 2-12. Big/Little Endian Program Mode**

BIGEND# Pin	BIGEND Register (1=Big, 0=Little)	Endian Mode
0	0	Big
0	1	Big
1	0	Little
1	1	Big

Table 2-13 lists registers which contain information about the following cycles.

**Table 2-13. Cycles Reference Tables**

Cycles	Register Bits
Local access to the Configuration registers	BIGEND[0]
PCI Initiator, Memory, and I/O	BIGEND[1]
PCI Target	BIGEND[2], Space 0, and BIGEND[3], Expansion ROM

In Big Endian mode, the PCI 9054 transposes data byte lanes. Data is transferred as listed in Table 2-14 through Table 2-18.

#### 2.3.2.1 32-Bit Local Bus—Big Endian Mode

Data is Lword aligned to uppermost byte lane (Data Invariance).

**Table 2-14. Upper Lword Lane Transfer**

Burst Order	Byte Lane
First transfer	Byte 3 appears on Local Data [31:24]
	Byte 2 appears on Local Data [23:16]
	Byte 1 appears on Local Data [15:8]
	Byte 0 appears on Local Data [7:0]

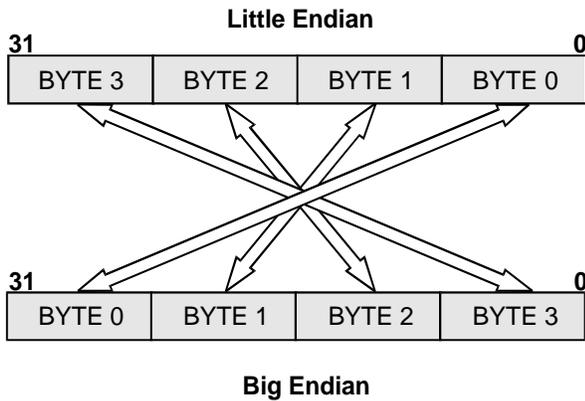


Figure 2-2. Big/Little Endian—32-Bit Local Bus

### 2.3.2.2 16-Bit Local Bus—Big Endian Mode

For a 16-bit Local Bus, the PCI 9054 can be programmed to use upper or lower word lanes.

Table 2-15. Upper Word Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [31:24]
	Byte 1 appears on Local Data [23:16]
Second transfer	Byte 2 appears on Local Data [31:24]
	Byte 3 appears on Local Data [23:16]

Table 2-16. Lower Word Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [15:8]
	Byte 1 appears on Local Data [7:0]
Second transfer	Byte 2 appears on Local Data [15:8]
	Byte 3 appears on Local Data [7:0]

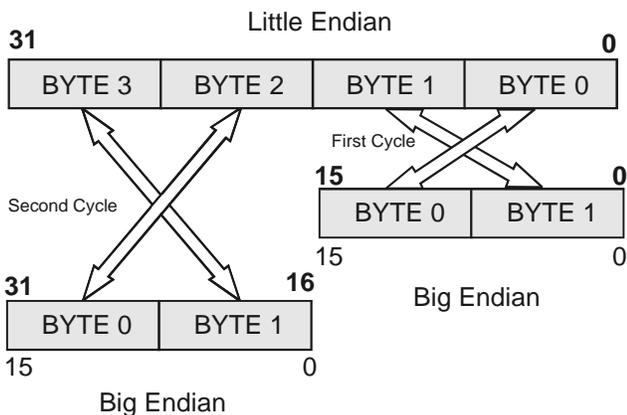


Figure 2-3. Big/Little Endian—16-Bit Local Bus

### 2.3.2.3 8-Bit Local Bus—Big Endian Mode

For an 8-bit Local Bus, the PCI 9054 can be programmed to use upper or lower byte lanes.

Table 2-17. Upper Byte Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [31:24]
Second transfer	Byte 1 appears on Local Data [31:24]
Third transfer	Byte 2 appears on Local Data [31:24]
Fourth transfer	Byte 3 appears on Local Data [31:24]

Table 2-18. Lower Byte Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [7:0]
Second transfer	Byte 1 appears on Local Data [7:0]
Third transfer	Byte 2 appears on Local Data [7:0]
Fourth transfer	Byte 3 appears on Local Data [7:0]

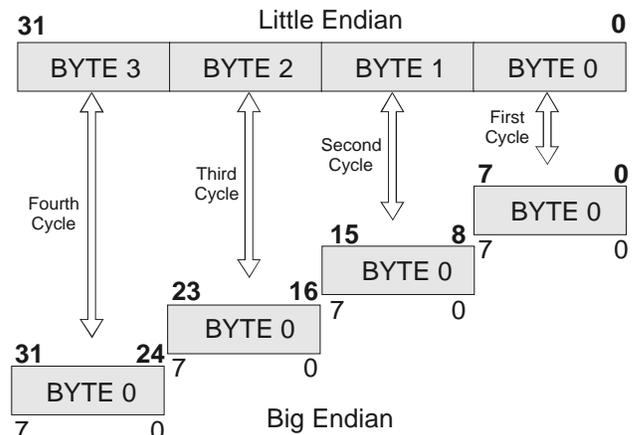


Figure 2-4. Big/Little Endian—8-Bit Local Bus

### 2.3.2.4 Local Bus Big/Little Endian Mode Accesses

For each of the following transfer types, the PCI 9054 Local Bus can be independently programmed to operate in Little Endian or Big Endian mode:

- Local Bus accesses to the PCI 9054 Configuration registers
- PCI Target PCI accesses to Local Address Space 0
- PCI Target PCI accesses to Local Address Space 1
- PCI Target PCI accesses to the Expansion ROM
- DMA Channel 0 accesses to the Local Bus
- DMA Channel 1 accesses to the Local Bus
- PCI Initiator accesses to the PCI Bus

For Local Bus accesses to the Internal Configuration registers and PCI Initiator accesses, use BIGEND# to dynamically change the Endian mode.

**Notes:** *The PCI Bus is always Little Endian. Only byte lanes are swapped, not individual bits.*

## 2.4 SERIAL EEPROM

Functional operation described can be modified through the PCI 9054 programmable internal registers.

### 2.4.1 Vendor and Device ID Registers

Three Vendor and Device ID registers are supported:

- **PCIIDR**—Contains normal Device and Vendor IDs. Can be loaded from the serial EEPROM or Local processor(s).
- **PCISVID**—Contains Subsystem and Subvendor IDs. Can be loaded from the serial EEPROM or Local processor(s).
- **PCIHIDR**—Contains hardcoded PLX Vendor and Device IDs.

#### 2.4.1.1 Serial EEPROM Initialization

During serial EEPROM initialization, the PCI 9054 responds to PCI Target accesses with a Retry. During serial EEPROM initialization, the PCI 9054 responds to a Local processor access by delaying acknowledgement of the cycle (TA#).

#### 2.4.1.2 Local Initialization

The PCI 9054 issues a Retry to all PCI accesses until the Local Init Status bit (LMISC[2]) is set. This bit can be programmed three different ways:

1. By the Local processor, through the Local Configuration register.
2. By the serial EEPROM, during a serial EEPROM load, if the Local processor does not set this bit or if this bit is missing.
3. If the Local processor and/or the serial EEPROM are missing, the serial EEPROM remains blank and the PCI 9054 reverts to the default values and sets this bit (refer to Table 2-19).

### 2.4.2 Serial EEPROM Operation

After reset, the PCI 9054 attempts to read the serial EEPROM to determine its presence. An active Start bit set to 0 indicates a serial EEPROM is present. The PCI 9054 supports 93CS56L (2K bit) or 93CS66L (4K bit). (Refer to manufacturer's data sheet for the particular serial EEPROM being used.) The first Lword is then checked to verify that the serial EEPROM is programmed. If the first Lword (33 bits) is all ones, a blank serial EEPROM is present. If the first Lword (33 bits) is all zeros, no serial EEPROM is present. For both conditions, the PCI 9054 reverts to the default values. (Refer to Table 2-19.) CNTRL[28] is set to 1 if programmed (real or random data if a serial EEPROM is detected).

The 3.3V serial EEPROM clock (EESK) is derived from the PCI clock. The PCI 9054 generates the serial EEPROM clock by internally dividing the PCI clock by 132.

The serial EEPROM can be read or written from the PCI or Local Buses. The Serial EEPROM Control Register bits (CNTRL[28:24]) control the PCI 9054 pins that enable reading or writing of serial EEPROM data bits. (Refer to manufacturer's data sheet for the particular serial EEPROM being used.)

The PCI 9054\_AB revision provides the ability to manually access the serial EEPROM. This may be accomplished by using bits 24 through 27 of the CNTRL register (EESK, EECS, and EEDI/EEDO, controlled by software). Bit 24 is used to generate EESK (clock), bit 25 controls the chip select, and bit 26 sets/clears EEDI. Bit 27, when read, returns the value of EEDO.

Setting bits 24, 25, and 26 to 1, causes the output to go high. A pull-up resistor is required on EEDO to go high when bit 26 is set. When reading the serial EEPROM, bit 26 must be set to 1.

To perform the read, the basic approach is to set the EECS and EEDO bits (bits 25 and 26) to the desired level and then toggle EESK high and low until done. *For example*, reading the serial EEPROM at location 0 involves the following steps:

1. Clear EESK, EEDO and EECS bits.
2. Toggle EESK high, then low.
3. Set EECS high.
4. Toggle EESK high, then low.
5. Set EEDO bit high (start bit).
6. Toggle EESK high, then low.
7. Repeat step 6.
8. Clear EEDO.
9. Toggle EESK bit high, then low eight times (clock in Local Address 0).
10. Set EEDO to float the EEDO pin for reading.
11. Toggle EESK high, then low 16 times (clock in one word from serial EEPROM).
12. After each clock pulse, read bit 27 and save.
13. Clear EECS bit.
14. Toggle EESK high, then low.
15. Read is now complete.

The serial EEPROM can also be read or written, using the VPD function (refer to Section 10).

The PCI 9054 has two serial EEPROM load options:

- **Long Load Mode**—Default. The PCI 9054 loads 17 Lwords from the Serial EEPROM and the Extra Long Load bit (LBRD0[25])
- **Extra Long Load Mode**—The PCI 9054 loads 22 Lwords if the Serial EEPROM and the Extra Long Load bit (LBRD0[25]) is set to 1 during a Long Load

Table 2-19. Serial EEPROM Guidelines

Local Processor	Serial EEPROM	System Boot Condition
None	None	The PCI 9054 uses default values. The EEDI/EEDO pin must be <b>pulled low</b> —a 1 K-ohm resistor is required (rather than pulled high, which is typically done for this pin). If the PCI 9054 detects all zeros, it reverts to default values.
None	Programmed	Boot with serial EEPROM values. The Local Init Status bit (LMISC[2]) must be set by the serial EEPROM.
None	Blank	The PCI 9054 detects a blank device and reverts to default values.
Present	None	The Local processor programs the PCI 9054 registers, then sets the Local Init Status bit (LMISC[2]=done). <i>Note: Some systems may hang if PCI Target reads and writes take too long (during initialization, the PCI Host also performs PCI Target accesses). The value of the PCI Target Retry Delay Clocks (LBRD0[31:28]) may resolve this.</i>
Present	Programmed	Load serial EEPROM, but the Local processor can reprogram the PCI 9054. Either the Local processor or the serial EEPROM must set the Local Init Status bit (LMISC[2]=done).
Present	Blank	The PCI 9054 detects a blank serial EEPROM and reverts to default values. <i>Notes: In some systems, the Local processor may be too late to reconfigure the PCI 9054 registers before the BIOS configures them. The serial EEPROM can be programmed through the PCI 9054 after the system boots in this condition.</i>

### 2.4.2.1 Long Serial EEPROM Load

The registers listed in Table 2-20 are loaded from the serial EEPROM after a reset is de-asserted if the Serial EEPROM Extra Long Load bit is not set (LBRD0[25]=0). The serial EEPROM is organized in words (16 bit). The PCI 9054 first loads the Most Significant Word bits (MSW[31:16]), starting from the most significant bit ([31]). The PCI 9054 then loads the Least Significant Word bits (LSW[15:0]), starting again from the most significant bit ([15]). Therefore, the PCI 9054 loads the Device ID, Vendor ID, class code, and so forth.

The serial EEPROM values can be programmed using a Data I/O programmer. The values can also be programmed using the PCI 9054 VPD function (refer to Section 10) or through the Serial EEPROM Control register (CNTRL).

The CNTRL register allows programming of the serial EEPROM, one bit at a time. To read back the value

from the serial EEPROM, the CNTRL[27] bit (refer to Section 2.4.2) or the VPD function should be utilized. With full utilization of VPD, the designer can perform reads and writes from/to the serial EEPROM, 32 bits at a time. Values should be programmed in the order listed in Table 2-20. The 34, 16-bit words listed in the table are stored sequentially in the serial EEPROM.

**Table 2-20. Long Serial EEPROM Load Registers**

Serial EEPROM Offset	Description	Register Bits Affected
0h	Device ID	PCIIDR[31:16]
2h	Vendor ID	PCIIDR[15:0]
4h	Class Code	PCICCR[23:8]
6h	Class Code / Revision	PCICCR[7:0] / PCIREV[7:0]
8h	Maximum Latency / Minimum Grant	PCIMLR[7:0] / PCIMGR[7:0]
Ah	Interrupt Pin / Interrupt Line Routing	PCIIPR[7:0] / PCIILR[7:0]
Ch	MSW of Mailbox 0 (User Defined)	MBOX0[31:16]
Eh	LSW of Mailbox 0 (User Defined)	MBOX0[15:0]
10h	MSW of Mailbox 1 (User Defined)	MBOX1[31:16]
12h	LSW of Mailbox 1 (User Defined)	MBOX1[15:0]
14h	MSW of Range for PCI-to-Local Address Space 0	LASORR[31:16]
16h	LSW of Range for PCI-to-Local Address Space 0	LASORR[15:0]
18h	MSW of Local Base Address (Remap) for PCI-to-Local Address Space 0	LASOBA[31:16]
1Ah	LSW of Local Base Address (Remap) for PCI-to-Local Address Space 0	LASOBA[15:0]
1Ch	MSW of Mode/DMA Arbitration Register	MARBR[31:16]
1Eh	LSW of Mode/DMA Arbitration Register	MARBR[15:0]
20h	MSW of Serial EEPROM Write-Protected Address	PROT_AREA[15:0]
22h	LSW of Local Miscellaneous Control Register / LSW of Local Bus Big/Little Endian Descriptor Register	LMISC[7:0] / BIGEND[7:0]
24h	MSW of Range for PCI-to-Local Expansion ROM	EROMRR[31:16]
26h	LSW of Range for PCI-to-Local Expansion ROM	EROMRR[15:0]
28h	MSW of Local Base Address (Remap) for PCI-to-Local Expansion ROM	EROMBA[31:16]
2Ah	LSW of Local Base Address (Remap) for PCI-to-Local Expansion ROM	EROMBA[15:0]
2Ch	MSW of Bus Region Descriptors for PCI-to-Local Accesses	LBRD0[31:16]
2Eh	LSW of Bus Region Descriptors for PCI-to-Local Accesses	LBRD0[15:0]
30h	MSW of Range for PCI Initiator-to-PCI	DMRR[31:16]
32h	LSW of Range for PCI Initiator-to-PCI	DMRR[15:0]
34h	MSW of Local Base Address for PCI Initiator-to-PCI Memory	DMLBAM[31:16]
36h	LSW of Local Base Address for PCI Initiator-to-PCI Memory	DMLBAM[15:0]
38h	MSW of Local Bus Address for PCI Initiator-to-PCI I/O Configuration	DMLBAI[31:16]
3Ah	LSW of Local Bus Address for PCI Initiator-to-PCI I/O Configuration	DMLBAI[15:0]
3Ch	MSW of PCI Base Address (Remap) for PCI Initiator-to-PCI	DMPBAM[31:16]
3Eh	LSW of PCI Base Address (Remap) for PCI Initiator-to-PCI	DMPBAM[15:0]
40h	MSW of PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration	DMCFGA[31:16]
42h	LSW of PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration	DMCFGA[15:0]

### 2.4.2.2 Extra Long Serial EEPROM Load

The registers listed in the Local Address Space 0/Expansion ROM Bus Region Descriptor register (LBRD0) are loaded from serial EEPROM after a reset is de-asserted if the Serial EEPROM Extra Long Load bit is set (LBRD0[25]=1). The serial EEPROM is organized in words (16 bit). The PCI 9054 first loads the Most Significant Word bits ([31:16]), starting from the most significant bit ([31]). It then loads the Least Significant Word bits ([15:0]), restarting from the most

significant bit ([15]). Therefore, the PCI 9054 loads Device ID, Vendor ID, class code, and so forth.

The serial EEPROM values can be programmed using a Data I/O programmer. The values can also be programmed using the PCI 9054 VPD function or through the Serial EEPROM Control register (CNTRL).

Values should be programmed in the order listed in Table 2-21. The 44 16-bit words listed in Table 2-20 and Table 2-21 should be stored sequentially in the serial EEPROM.

Table 2-21. Extra Long Serial EEPROM Load Registers

Serial EEPROM Offset	Description	Register Bits Affected
44h	Subsystem ID	PCISID[15:0]
46h	Subsystem Vendor ID	PCISVID[15:0]
48h	MSW of Range for PCI-to-Local Address Space 1 (1 MB)	LAS1RR[31:16]
4Ah	LSW of Range for PCI-to-Local Address Space 1 (1 MB)	LAS1RR[15:0]
4Ch	MSW of Local Base Address (Remap) for PCI-to-Local Address Space 1	LAS1BA[31:16]
4Eh	LSW of Local Base Address (Remap) for PCI-to-Local Address Space 1	LAS1BA[15:0]
50h	MSW of Bus Region Descriptors (Space 1) for PCI-to-Local Accesses	LBRD1[31:16]
52h	LSW of Bus Region Descriptors (Space 1) for PCI-to-Local Accesses	LBRD1[15:0]
54h	MSW of Hot Swap Control/Status	<b>Reserved</b>
56h	LSW of Hot Swap Control / Hot Swap Next Capability Pointer	HS_NEXT[7:0] / HS_CNTRL[7:0]

### 2.4.2.3 New Capabilities Function Support

The New Capabilities Function Support includes PCI Power Management, Hot Swap, and VPD features, as listed in Table 2-22.

Table 2-22. New Capabilities Function Support Features

New Capability Function	PCI Register Offset Location
First (Power Management)	40'h, if the New Capabilities Function Support bit (PCISR[4]) is enabled (PCISR[4] is enabled, by default).
Second (Hot Swap)	48'h, which is pointed to from PMNEXT[7:0].
Third (VPD)	4C'h, which is pointed to from HS_NEXT[7:0]. Because PVPD_NEXT[7:0] defaults to zero, this indicates that VPD is the last New Capability Function Support feature of the PCI 9054.

### 2.4.2.4 Recommended Serial EEPROMs

The PCI 9054 is designed to use either a 2K bit (NM93CS56L or compatible) or 4K bit (NM93CS66L or compatible) device.

**Note:** The PCI 9054 does not support serial EEPROMs that do not support sequential reads and writes (such as the NM93C56L).

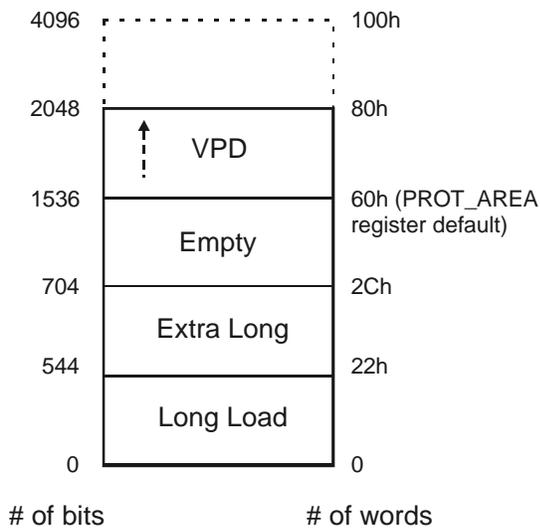


Figure 2-5. Serial EEPROM Memory Map

### 2.4.2.5 Serial EEPROM Initialization

During serial EEPROM initialization, the PCI 9054 responds to PCI Target accesses with a Retry. During serial EEPROM initialization, the PCI 9054 responds to a Local processor access by delaying acknowledgement of the cycle (TA#).

### 2.4.3 Internal Register Access

The PCI 9054 provides several internal registers, which allow for maximum flexibility in the bus interface design and performance. These registers are accessible from the PCI and Local Buses and include the following:

- PCI and Local Configuration registers
- DMA registers
- Mailbox registers
- PCI-to-Local and Local-to-PCI Doorbell registers
- Messaging Queue registers (I<sub>2</sub>O)
- Power Management registers
- Hot Swap registers
- VPD registers

Figure 2-6 illustrates how these registers are accessed.

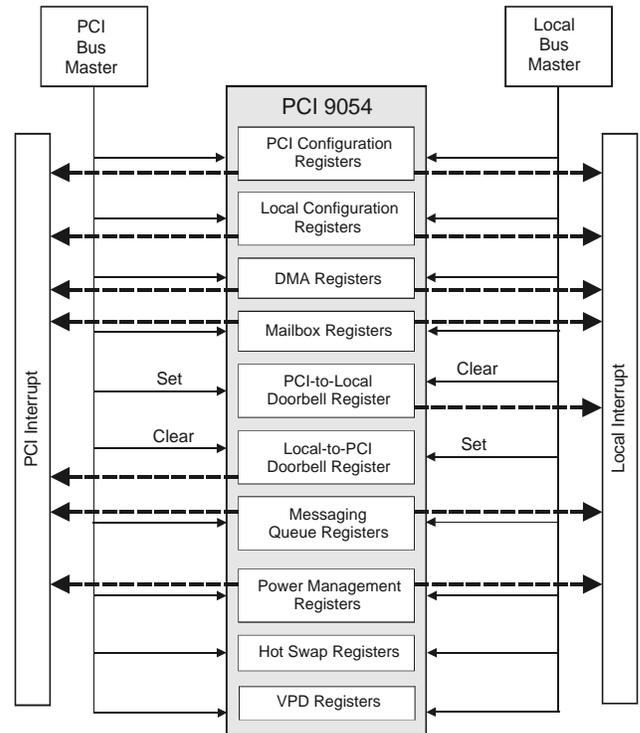


Figure 2-6. PCI 9054 Internal Register Access

#### 2.4.3.1 PCI Bus Access to Internal Registers

The PCI 9054 PCI Configuration registers can be accessed from the PCI Bus with a Configuration Type 0 cycle.

All other PCI 9054 internal registers can be accessed by a Memory cycle, with the PCI Bus address that matches the base address specified in PCI Base Address 0 (PCIBAR0[31:8]) for the PCI 9054 Memory-Mapped Configuration register. These registers can also be accessed by an I/O cycle, with the PCI Bus address matching the base address specified in PCI Base Address 1 for the PCI 9054 I/O-Mapped Configuration register (PCIBAR1).

All PCI Read or Write accesses to the PCI 9054 registers can be Byte, Word, or Lword accesses. All PCI Memory accesses to the PCI 9054 registers can be Burst or Non-Burst accesses. The PCI 9054 responds with a PCI disconnect for all Burst I/O accesses (PCIBAR1[31:8]) to the PCI 9054 Internal registers.

### 2.4.3.2 Local Bus Access to Internal Registers

The Local processor can access all PCI 9054 internal registers through an external chip select. The PCI 9054 responds to a Local Bus access when the PCI 9054 Configuration Chip Select input (CCS#) is asserted low. Figure 2-7 illustrates how the Configuration Chip Select logic works.

**Notes:** CCS# must be decoded while TS# is low.  
Accesses must be for a 32-bit non-pipelined bus.

Local Read or Write accesses to the PCI 9054 internal registers can be Byte, Word, or Lword accesses. Local accesses to the PCI 9054 internal registers can be Burst or Non-Burst accesses.

The PCI 9054 TA# signal indicates that Data transfer is complete.

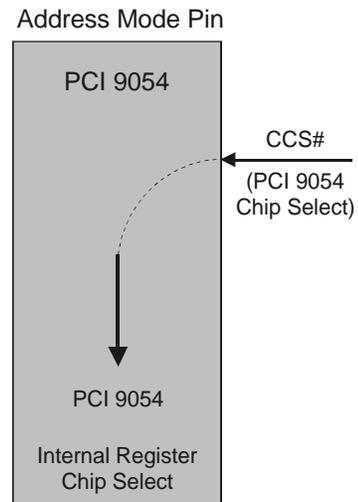
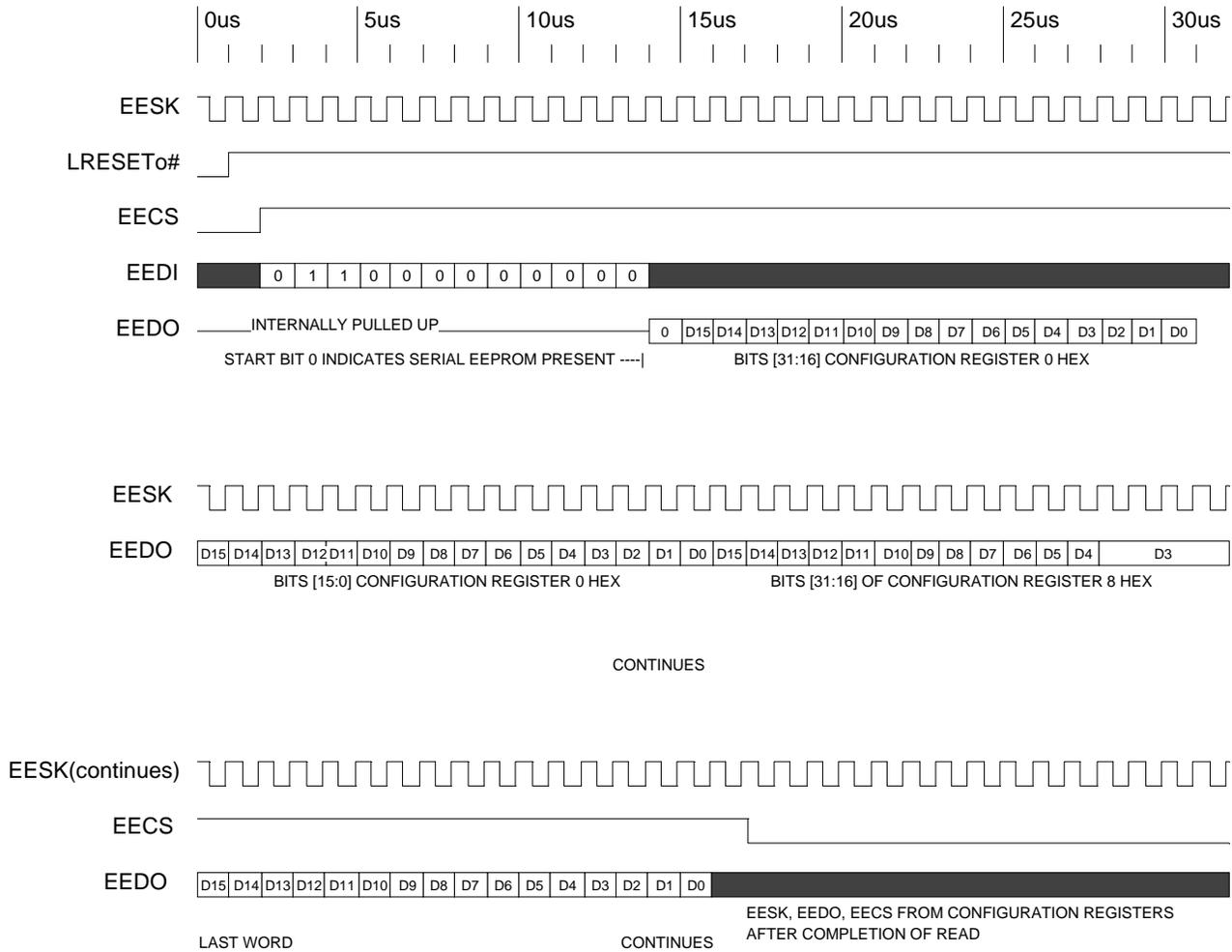


Figure 2-7. Address Decode Mode

### 2.4.4 Serial EEPROM Timing Diagrams

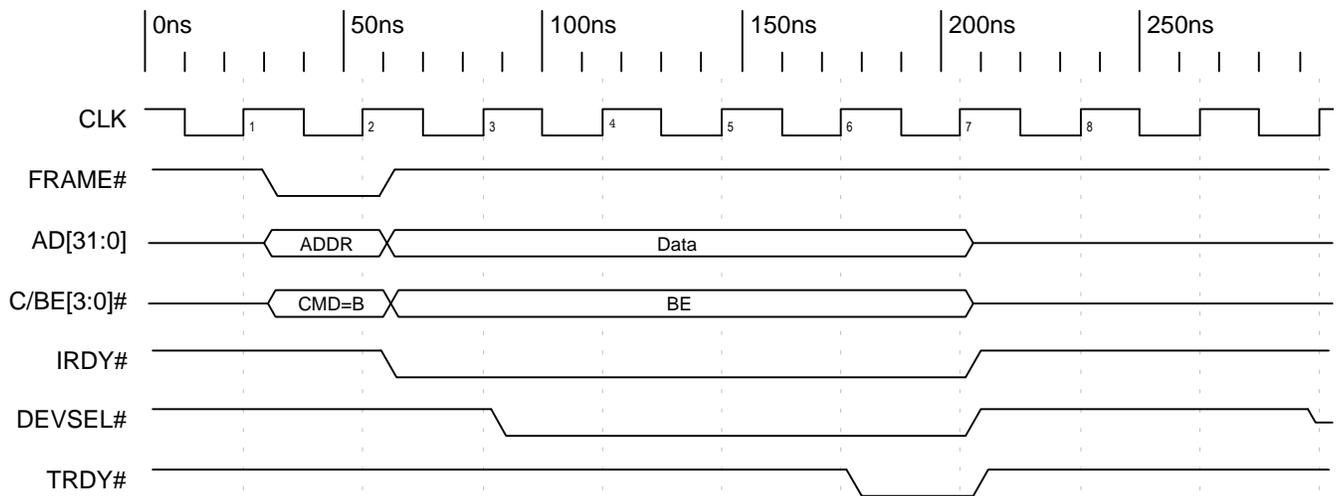


**Timing Diagram 2-1. Initialization from Serial EEPROM (2K Bit)**

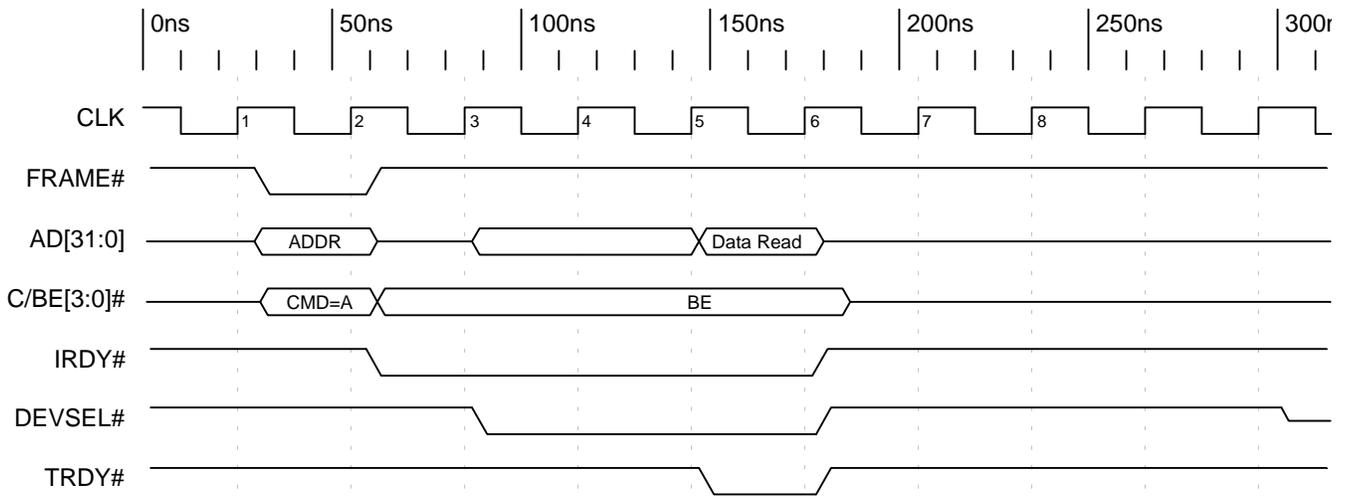
Section 2—M Bus Op



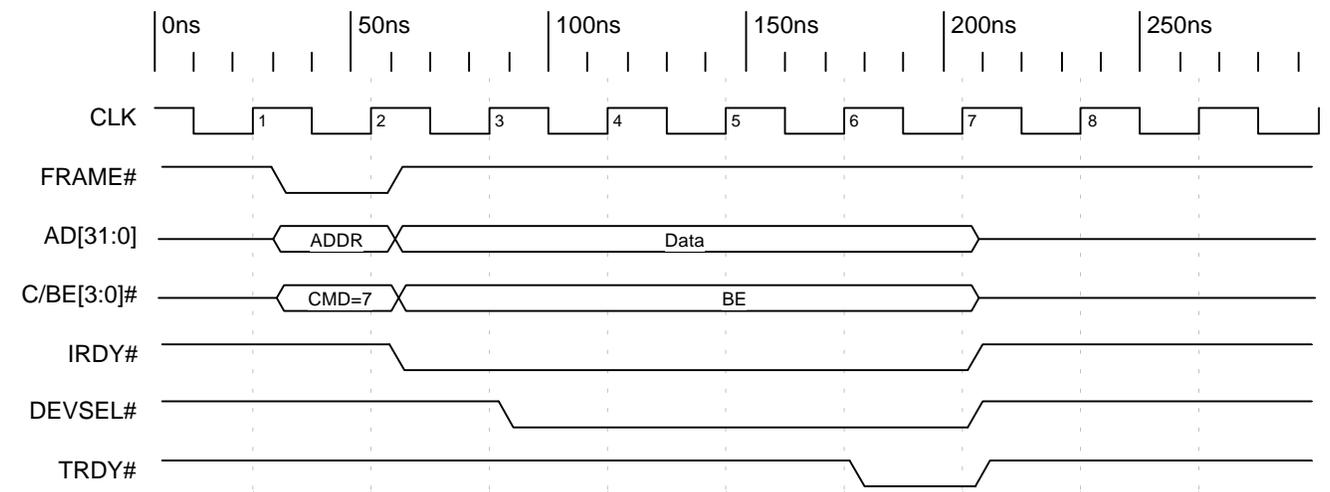
Timing Diagram 2-2. Initialization from Serial EEPROM (4K Bit)



Timing Diagram 2-3. PCI Configuration Write to PCI Configuration Register

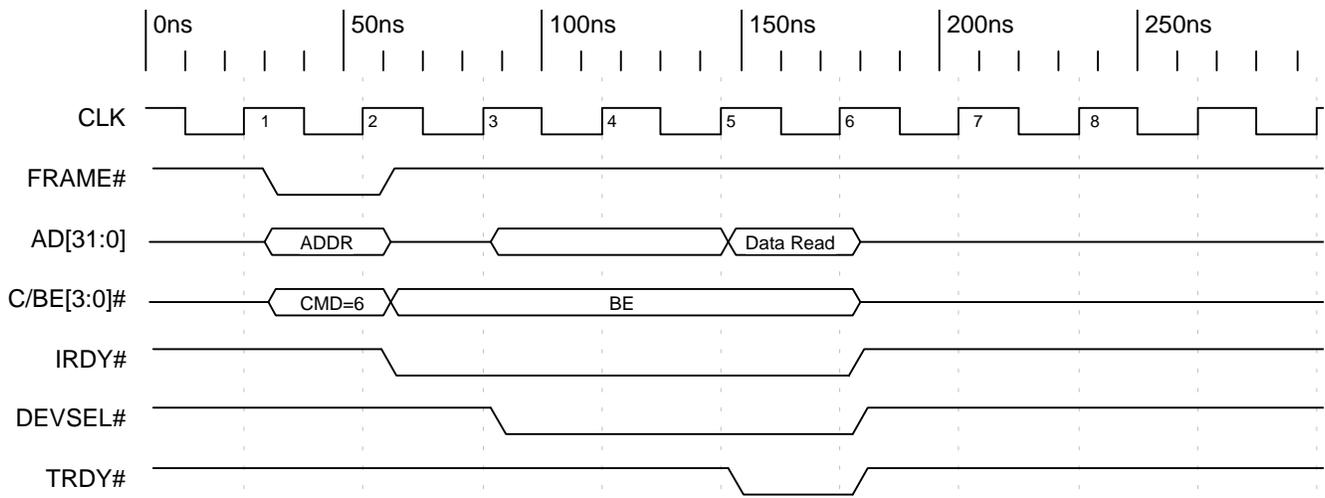


Timing Diagram 2-4. PCI Configuration Read to PCI Configuration Register

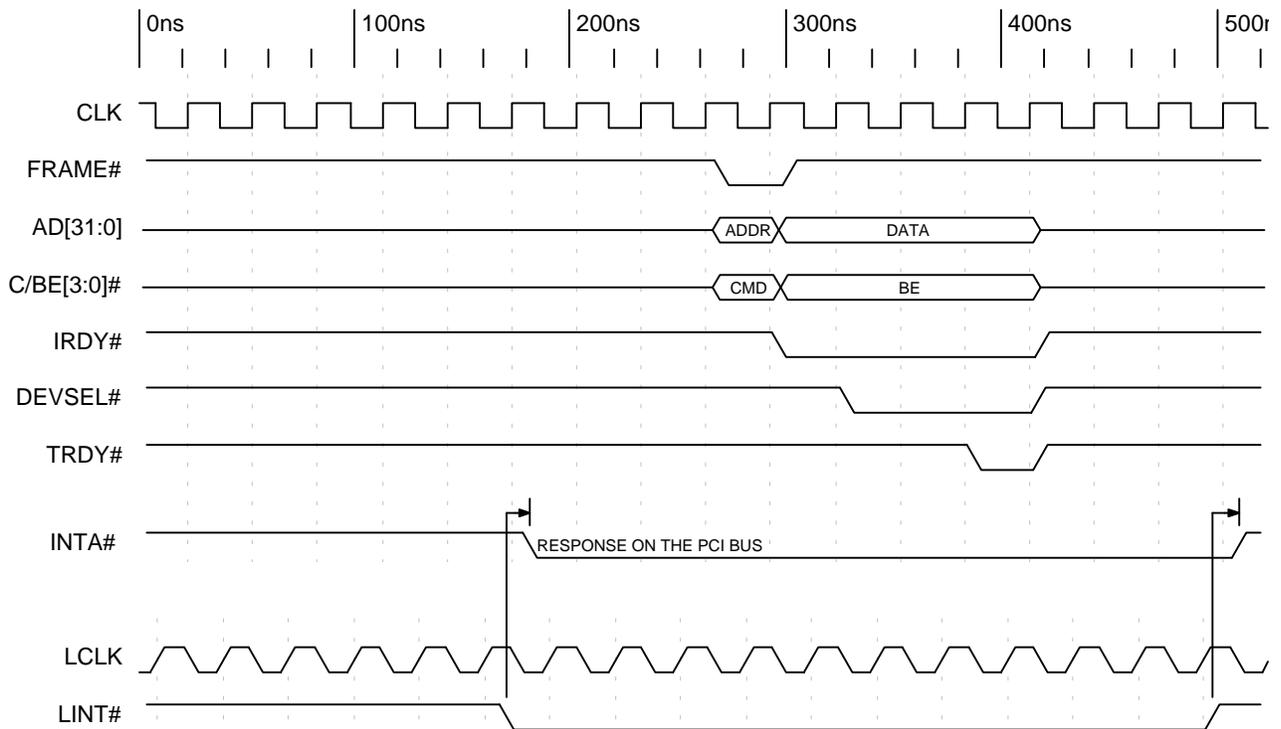


Timing Diagram 2-5. PCI Memory Write to Local Configuration Register

Section 2—M Bus Op



Timing Diagram 2-6. PCI Memory Read to Local Configuration Register



Timing Diagram 2-7. Local Interrupt Asserting PCI Interrupt

## 3 M MODE FUNCTIONAL DESCRIPTION

The functional operation described in this chapter can be modified through the PCI 9054 programmable internal registers.

### 3.1 RESET OPERATION

#### 3.1.1 PCI Bus Input RST#

PCI Bus RST# input pin is a PCI Host reset. It causes all PCI Bus outputs to float, resets the entire PCI 9054 and causes the Local reset LRESETo# signal to be asserted.

#### 3.1.2 Software Reset

A Host on the PCI Bus can set the PCI Adapter Software Reset bit (CNTRL[30]=1) to reset the PCI 9054 and assert LRESETo# output. All Local Configuration registers are reset; however, the PCI Configuration DMA and Shared Runtime registers and the Local Init Status bit (LMISC[2]) are not reset. When the Software Reset bit (CNTRL[30]) is set, the PCI 9054 responds to PCI accesses, but not to Local Bus accesses. The PCI 9054 remains in this reset condition until the PCI Host clears the bit. The serial EEPROM is reloaded if the Reload Configuration Registers bit is set (CNTRL[29]=1).

**Note:** *The Local Bus cannot clear this reset bit because the Local Bus is in a reset state, even if the Local processor does not use LRESETo# to reset.*

### 3.2 PCI 9054 INITIALIZATION

The PCI 9054 Configuration registers can be programmed by an optional serial EEPROM and/or by a Local processor, as listed in Table 2-19. The serial EEPROM can be reloaded by setting the Reload Configuration Registers bit (CNTRL[29]).

The PCI 9054 retries all PCI cycles until the Local Init Status bit is set to “done” (LMISC[2]=1).

**Note:** *The PCI Host processor can also access Internal Configuration registers after the Local Init Status bit is set.*

If a PCI Host is present, the Master Enable, Memory Space, and I/O Space bits (PCICR[2:0]) are programmed by that Host after initialization completes (LMISC[2]=1).

### 3.3 RESPONSE TO FIFO FULL OR EMPTY

Table 3-1 lists the PCI 9054 response to full and empty FIFOs.

### 3.4 DIRECT DATA TRANSFER MODES

The PCI 9054 supports three direct transfer modes:

- **PCI Initiator**—Local CPU accesses PCI memory or I/O
- **PCI Target**—PCI Master accesses Local memory or I/O
- **DMA**—PCI 9054 DMA controller reads/writes PCI memory to/from Local memory

#### 3.4.1 PCI Initiator Operation (Local Master-to-PCI Target)

The PCI 9054 supports a direct access of the PCI Bus by the Local processor or an intelligent controller. Master mode must be enabled in the PCI Command register. The following registers define Local-to-PCI accesses:

- PCI Initiator Memory and I/O Range (DMRR)
- Local Base Address for PCI Initiator-to-PCI Memory (DMLBAM)
- Local Base Address for PCI Initiator-to-PCI I/O and Configuration (DMLBAI)
- PCI Base Address (DMPBAM)
- PCI Initiator Configuration (DMCFGA)
- PCI Initiator PCI Dual Address Cycles (DMDAC)
- Master Enable (PCICR)
- PCI Command Code (CNTRL)

Table 3-1. Response to FIFO Full or Empty

Mode	Direction	FIFO	PCI Bus	Local Bus
PCI Initiator Write	Local-to-PCI	Full	Normal	De-assert TA#, RETRY# <sup>1</sup>
		Empty	De-assert REQ# (off the PCI Bus)	Normal
PCI Initiator Read	PCI-to-Local	Full	De-assert REQ# or throttle IRDY# <sup>2</sup>	Normal
		Empty	Normal	De-assert TA#
PCI Target Write	PCI-to-Local	Full	Disconnect or throttle TRDY# <sup>3</sup>	Normal
		Empty	Normal	De-assert BB# <sup>4</sup>
PCI Target Read	Local-to-PCI	Full	Normal	De-assert BB# <sup>4</sup>
		Empty	Throttle TRDY# <sup>3</sup>	Normal
DMA	Local-to-PCI	Full	Normal	De-assert BB# <sup>4</sup>
		Empty	De-assert REQ#	Normal
	PCI-to-Local	Full	De-assert REQ#	Normal
		Empty	Normal	De-assert BB# <sup>4</sup>

- <sup>1</sup>. Issue RETRY# depends upon the PCI Initiator Write FIFO Almost Full RETRY# Output Enable bit (LMISC[6]).
- <sup>2</sup>. Throttle IRDY# depends upon the PCI Initiator PCI Read Mode bit (DMPBAM[4]).
- <sup>3</sup>. Throttle TRDY# depends upon the PCI Target Write Mode bit (LBRD0[27]).
- <sup>4</sup>. BB# de-assert depends upon the Local Bus PCI Target Release Bus Mode bit (MARBR[21]).

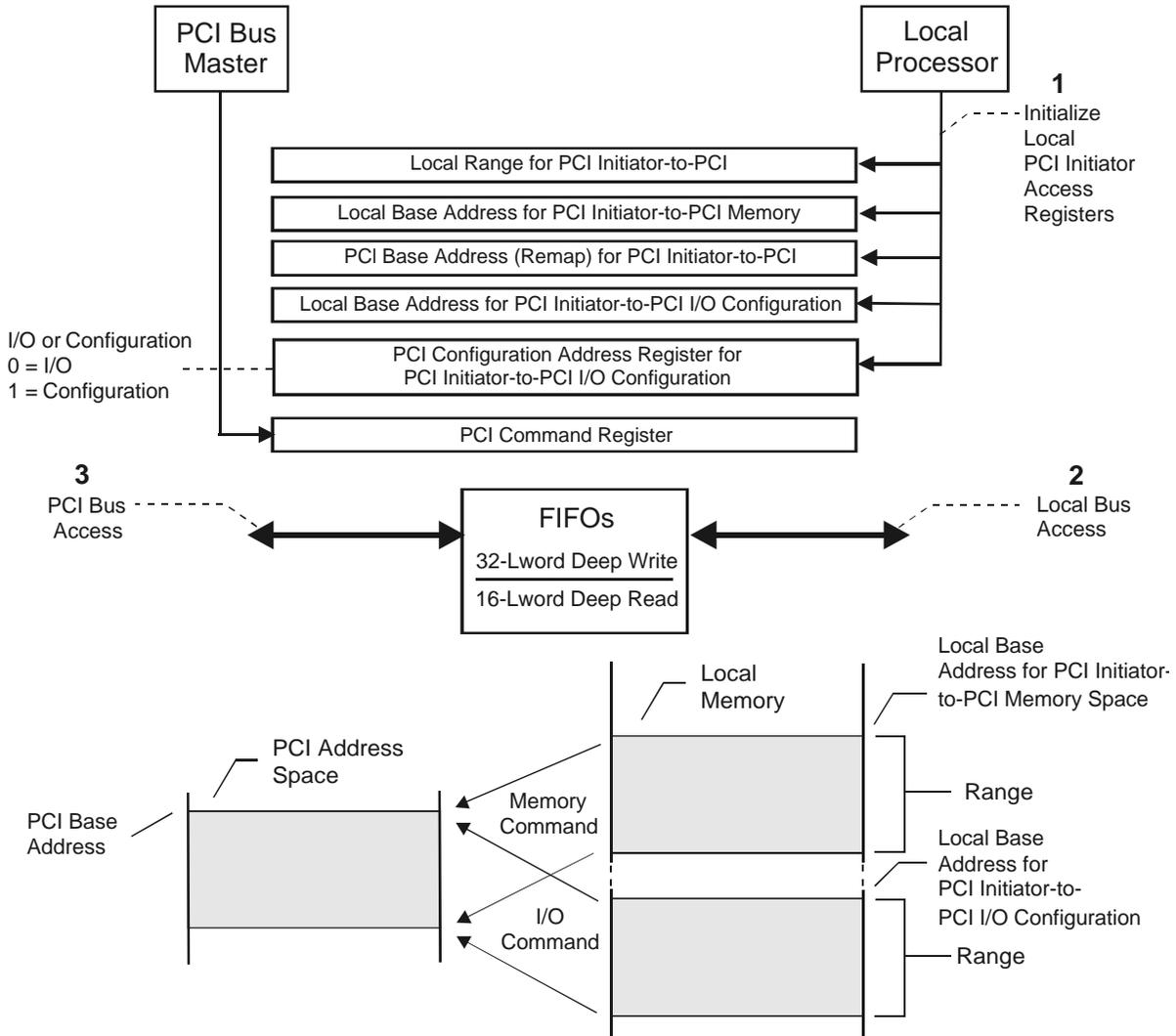


Figure 3-1. PCI Initiator Access of the PCI Bus

### 3.4.1.1 PCI Initiator Memory and I/O Decode

The Range register and the Local Base Address specifies the Local Address bits to use for decoding a Local-to-PCI access (PCI Initiator). The range of Memory or I/O space must be a power of 2 and the Range register value must be the inverse of the range value. In addition, the Local Base Address must be a multiple of the range value.

Any Local Master Address starting from the PCI Initiator Local Base Address (Memory or I/O) to the range value is recognized as a PCI Initiator access by the PCI 9054. All PCI Initiator cycles are then decoded as PCI Memory, I/O, or Configuration Type 0 or 1.

Moreover, a PCI Initiator Memory or I/O cycle is remapped according to the Remap register value. The Remap Register value must be a multiple of the PCI Initiator Range value (not the Range register value).

The PCI 9054 can only accept Memory cycles from a Local processor. The Local Base Address and/or the range determine whether PCI Memory or PCI I/O transactions occur.

### 3.4.1.2 PCI Initiator FIFOs

For PCI Initiator Memory access to the PCI Bus, the PCI 9054 has a 32-Lword (128-byte) Write FIFO and a 16-Lword (64-byte) Read FIFO. The FIFOs enable the Local Bus to operate independent of the PCI Bus and

Section 3—M Func Desc

allows high-performance bursting on the PCI and Local Buses. In a PCI Initiator write, the Local processor (Master) writes data to the PCI Bus (Slave). In a PCI Initiator read, the Local processor (Master) reads data from the PCI Bus (Slave). The FIFOs that function during a PCI Initiator write and read are illustrated in Figure 3-2 and Figure 3-3.

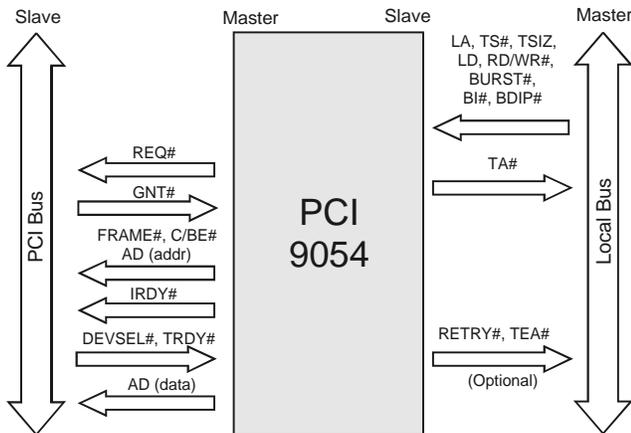


Figure 3-2. PCI Initiator Write

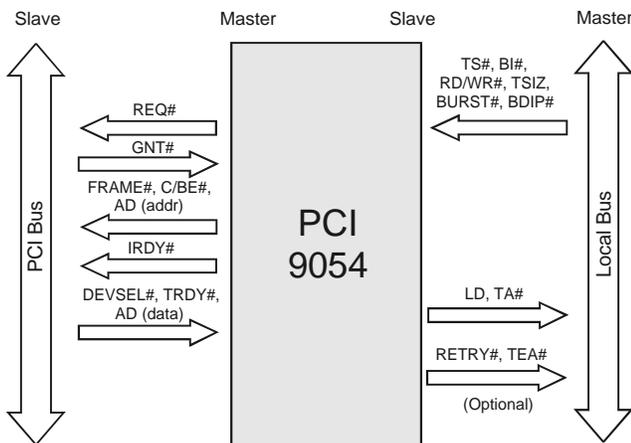


Figure 3-3. PCI Initiator Read

**Note:** The figures represent a sequence of Bus cycles.

### 3.4.1.3 PCI Initiator Memory Access

The MPC850 or MPC860 transfers data through a Single or Burst Read/Write Memory transaction, or through SDMA channels to the PCI 9054 and PCI Bus. The MPC850 or MPC860 IDMA/SDMA accesses to the PCI 9054 appear as PCI Initiator operations. (Refer to Section 3.4.2 for further information.)

Transactions are initiated by the MPC850 or MPC860 (a Local Bus Master) when the memory address on the Local Bus matches the Memory space decoded for PCI Initiator operations. Upon a Local Bus Read, the PCI 9054 becomes a PCI Bus Master, arbitrates for the PCI Bus, and reads data from the PCI Target directly into the PCI Initiator Read FIFO. When sufficient data is placed into the FIFO, it asserts the Transfer Acknowledge (TA#) signal onto the Local Bus to indicate that the requested data is on the Local Bus.

The Local processor can read or write to PCI memory. The PCI 9054 converts the Local Read/Write access. The Local Address space starts from PCI Initiator Local Base Address up to the range. Remap (PCI Base Address) defines PCI starting address.

An MPC850 or MPC860 Single cycle causes a Single-Cycle PCI transaction. An MPC850 or MPC860 Burst cycle asserts a Burst PCI Cycle transaction. Bursts are limited to 16 bytes (four Lwords) in the MPC850 or MPC860.

The PCI 9054 supports bursts beyond the 16-byte boundary (Continuous Burst) when the BDIP# input signal remains asserted beyond a 16-byte boundary by an external Local Bus Master. To finish, continuing burst and the external Master should de-assert the BDIP# signal on the last Data phase.

**Writes**—Upon a Local Bus Write, the Local Bus Master writes data to the PCI Initiator Write FIFO. When the first data is in the FIFO, the PCI 9054 becomes the PCI Bus Master, arbitrates for the PCI Bus, and writes data to the PCI Target device. The PCI 9054 continues to accept writes and returns TA# until the Write FIFO is full. It then holds off TA# until space becomes available in the Write FIFO. A programmable PCI Initiator “almost full” status output is provided (MDREQ#/DMPAF/EOT#). The PCI 9054 asserts RETRY# whenever the PCI Initiator Write FIFO is full, implying that the Local Master can relinquish the bus and finish the Write operation at a later time (LMISC[6]).

**Reads**—The PCI 9054 holds off TA# while gathering an Lword from the PCI Bus. Programmable Prefetch modes are available if prefetch is enabled: prefetch, 4, 8, 16, or continuous until the PCI Initiator cycle ends. The Read cycle is terminated when the Local BDIP# input is de-asserted. Unused Read data is flushed from the FIFO.

The PCI 9054 does not prefetch Read data for single cycle PCI Initiator reads (Local BURST# input is not asserted during the first Data phase). In this case, the PCI 9054 reads a single PCI Lword unless PCI Initiator Read Ahead mode is enabled.

For PCI Initiator Single-Cycle reads, the PCI 9054 sets the corresponding PCI Bus byte enables from the Local Bus address and the TSIZ[0:1] signal.

For Burst-Cycle reads, the PCI 9054 reads entire Lwords (all PCI Bus byte enables are asserted).

If the PCI Initiator Prefetch Limit bit is enabled (DMPBAM[11]=1), the PCI 9054 does not prefetch past a 4 KB boundary. Also, the Local Bus must not cross a 4 KB boundary during a Burst read.

The PCI 9054 never prefetches beyond the region specified for PCI Initiator accesses.

#### 3.4.1.4 PCI Initiator I/O Configuration Access

When a Local PCI Initiator I/O access to the PCI Bus occurs, the PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration Enable bit (DMCFGA[31]) determines whether an I/O or Configuration access is to be made to the PCI Bus.

Local Burst accesses are broken into single PCI I/O (address/data) cycles. The PCI 9054 does not prefetch Read data for I/O and configuration reads.

For PCI Initiator I/O or Configuration cycles, the PCI 9054 asserts the same PCI Bus byte enables as set on the Local Bus.

#### 3.4.1.5 PCI Initiator I/O

If the Configuration Enable bit is cleared (DMCFGA[31]=0), a single I/O access is made to the PCI Bus. The Local Address, Remapped Decode Address Bits, and Local byte enables are encoded to provide the address and are output with an I/O read or write command during a PCI Address cycle.

When the I/O Remap Select bit is set (DMPBAM[13]=1), the PCI Address bits [31:16] are forced to 0 for the 64 KB I/O address limit.

For writes, data is loaded into the Write FIFO and TA# is returned to the Local Bus. For reads, the PCI 9054 holds off TA# while receiving an Lword from the PCI Bus.

### 3.4.1.6 RETRY# Capability

#### 3.4.1.6.1 PCI Initiator Write FIFO Full

The PCI 9054 supports the PCI Initiator Write FIFO full condition. When enabled (LMISC[6]=1), the PCI 9054 asserts the RETRY# signal to the Local Bus Master to relinquish ownership of the bus and return to finish the initial write at a later time.

In a PCI Initiator Write FIFO full condition, the PCI 9054 asserts the RETRY# signal. Otherwise, the PCI Initiator Write transfer goes through successfully.

#### 3.4.1.6.2 PCI Initiator Delayed Read

The PCI 9054 supports Deferred PCI Initiator Read transactions. When the M Mode PCI Initiator Deferred Read Enable bit is set (LMISC[4]=1), the PCI 9054 asserts RETRY# and prefetches Read data every time the Local Master requests a read. During a PCI data prefetch, the Local Master is capable of doing other transactions and free to return for requested data at a later time. When Deferred PCI Initiator Read mode is disabled, the Local Master must “keep” the Local Bus and wait for the requested data (TA# is not asserted until data is available to the Local Bus).

#### 3.4.1.7 PCI Initiator Configuration (PCI Configuration Type 0 or Type 1 Cycles)

If the Configuration Enable bit (DMCFGA[31]) is set, a Configuration access is made to the PCI Bus. In addition to enabling configuration of this bit, the user must provide all register information. The Register Number and Device Number bits (DMCFGA[7:2] and DMCFGA[15:11], respectively) must be modified and a new Configuration Read/Write cycle must be performed before accessing other registers or devices.

If the PCI Configuration Address register selects a Type 0 command, bits [10:0] of the register are copied to address bits [10:0]. Bits [15:11] (device number) are translated into a single bit being set in the PCI Address bits [31:11]. The PCI Address bits [31:11] can be used as a device select. For a Type 1 command, bits [23:0] are copied from the register to bits [23:0] of the PCI address. The PCI Address bits [31:24] are set to 0. A configuration read or write command code is output with the address during the PCI Address cycle (refer to the DMCFGA register).

For writes, Local data is loaded into the Write FIFO and TA# is returned. For reads, the PCI 9054 holds off TA# while gathering an Lword from the PCI Bus.

### 3.4.1.7.1 PCI Initiator Configuration Cycle Example

To perform a Type 0 Configuration cycle to PCI device on AD[21]:

1. The PCI 9054 must be configured to allow PCI Initiator access to the PCI Bus. The PCI 9054 must also be set to respond to I/O Space accesses. These bits must be set (PCICR[2:0]=111b).

In addition, PCI Initiator memory and I/O access must be enabled (DMPBAM[1:0]=11).

2. The Local Memory map selects the PCI Initiator range. For this example, use a range of 1 MB:

$$1 \text{ MB} = 2^{20} = 000\text{FFFFFFh}$$

3. The value to program into the Range register is the inverse of 000FFFFFFh (FFF00000h):

$$\text{DMRR} = \text{FFF00000h}$$

4. The Local Memory map determines the Local Base Address for the PCI Initiator-to-PCI I/O Configuration register. For this example, use 40000000h:

$$\text{DMLBAI} = 40000000\text{h}$$

5. The PCI Address (Remap) for PCI Initiator-to-PCI Memory register must enable the PCI Initiator I/O access. The PCI Initiator I/O Access Enable bit must be set (DMPBAM[1]=1).
6. The user must know which PCI device and PCI Configuration register the PCI Configuration cycle is accessing. This example assumes the IDSEL signal of the Target PCI device is connected to AD[21] (logical device #10=0Ah). Also access PCIBAR0 (the fourth register, counting from 0; use Table 11-2 for reference). Set DMCFGA[31, 23:0] as follows:

Bit	Description	Value
1:0	Configuration Type 0.	00b
7:2	Register Number. Fourth register. Must program a "4" into this value, beginning with bit 2.	000100b
10:8	Function Number.	000b
15:11	Device Number n-11, where n is the value in AD[n]=21-11 = 10.	01010b
23:16	Bus Number.	00000000b
31	Configuration Enable.	1

After these registers are configured, a simple Local Master Memory cycle to the I/O base address is necessary to generate a PCI Configuration Read or Write cycle. Offset to the base address is not necessary because the register offset for the read or write is specified in the Configuration register. The PCI 9054 takes the Local Bus Master Memory cycle and checks for the Configuration Enable bit (DMCFGA[31]). If set, the PCI 9054 converts the current cycle to a PCI Configuration cycle, using the DMCFGA register and the Write/Read signal (RD/WR#).

The Register Number and Device Number bits (DMCFGA[7:2] and DMCFGA[15:11], respectively) must be modified and a new Configuration Read/Write cycle must be performed before accessing other registers or devices.

### 3.4.1.8 PCI Initiator PCI Dual Address Cycle

The PCI 9054 supports PCI Dual Address Cycle (DAC) when it is a PCI Bus Master using the DMDAC register for PCI Initiator transactions. The DAC command is used to transfer a 64-bit address to devices that support 64-bit addressing when the address is not in the low 4-GB Address space. The PCI 9054 performs the address portion of a DAC in two PCI clock periods, where the first PCI address is a Lo-Addr with the command (C/BE[3:0]#) "D" and the second PCI address will be a Hi-Addr with the command (C/BE[3:0]#) "6" or "7", depending upon it being a PCI Read or a PCI Write cycle. Whenever the DMDAC register contains a value of 0x00000000, the PCI 9054 performs a Single Address Cycle (SAC) on the PCI Bus. (Refer to Figure 3-4.)

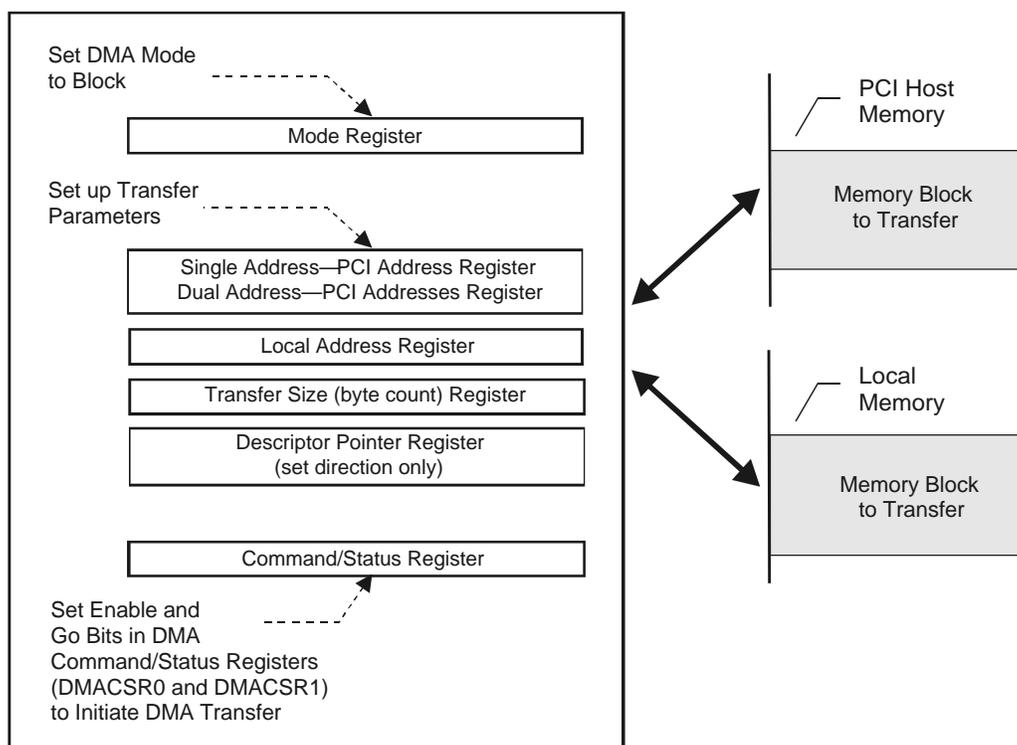


Figure 3-4. Block DMA Mode Initialization (Single Address or Dual Address PCI)

### 3.4.1.9 PCI Initiator/Target Abort

The PCI 9054 PCI Initiator/Target Abort logic enables a Local Bus Master to perform a PCI Initiator Bus poll of devices to determine whether devices exist (typically when the Local Bus performs Configuration cycles to the PCI Bus). When a PCI Master device attempts to access and does not receive DEVSEL# within six PCI clocks, it results in a Master Abort. The Local Bus Master must clear the Received Master Abort bit or Target Abort bit (PCISR[13 or 11]=0, respectively) and continue by processing the next task.

If a PCI Master/Target Abort, or Retry Time-Out is encountered during a transfer, the PCI 9054 asserts TEA# if enabled [(INTCSR[1:0]=1), which can be used as a Non-Maskable Interrupt (NMI)]. If a Local Bus Master is waiting for TA#, it is asserted along with TEA#. The interrupt handler of the Local Bus Master can take the appropriate application-specific action. It can then clear the Target Abort bit (PCISR[11]) to clear the TEA# interrupt and re-enable PCI Initiator transfers.

If a Local Bus Master is attempting a Burst read from a nonresponding PCI device (Master/Target Abort), it

receives TA# for the first cycle only. In addition, the PCI 9054 asserts TEA# if the Enable Local Bus TEA# bits are enabled (INTCSR[1:0], which can be used as an NMI). If the Local processor cannot terminate its Burst cycle, it may cause the Local processor to hang. The Local Bus must then be reset from the PCI Bus. If the Local Bus Master cannot terminate its cycle with TEA# output, it should not perform Burst cycles when attempting to determine whether a PCI device exists.

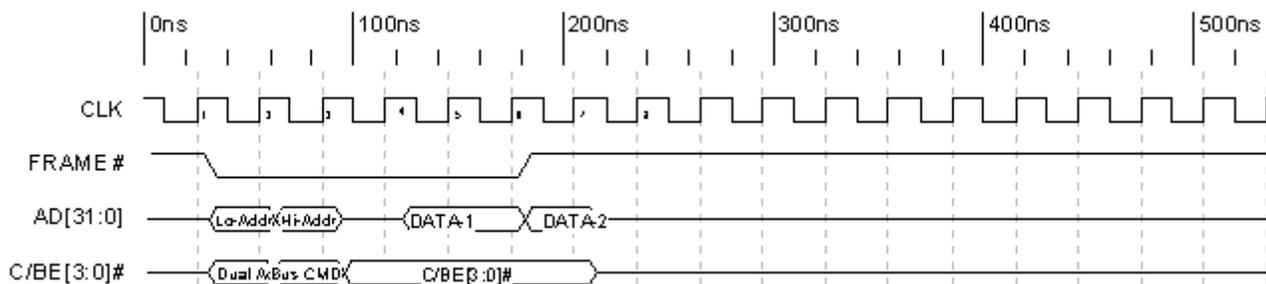
### 3.4.1.10 PCI Initiator Memory Write and Invalidate

The PCI 9054 can be programmed to perform Memory Write and Invalidate cycles to the PCI Bus for PCI Initiator transfers, as well as DMA transfers (refer to Section 3.5.4). The PCI 9054 supports Memory Write and Invalidate transfers for cache line sizes of 8 or 16 Lwords. Size is specified in the System Cache Line Size bits (PCICLSR[7:0]). If a size other than 8 or 16 is specified, the PCI 9054 performs Write transfers rather than Memory Write and Invalidate transfers.

PCI Initiator Memory Write and Invalidate transfers are enabled when the Invalidate Enable and the Memory Write and Invalidate Enable bits are set (DMPBAM[9]) and (PCICR[4]), respectively.

In Memory Write and Invalidate mode, if the start address of the PCI Initiator transfer is on a cache line boundary, the PCI 9054 waits until the number of Lwords required for the specified cache line size are written from the Local Bus before starting a PCI Memory Write and Invalidate access. This ensures a complete cache line write can complete in one PCI Bus ownership.

If the start address is not on a cache line boundary, the PCI 9054 starts a normal PCI Write access (PCI command code = 7h). The PCI 9054 terminates a cycle at a cache line boundary if it is performing a normal write or if it is performing a Memory Write and Invalidate cycle and another cache line of data is not available. If an entire cache line is available by the time PCI 9054 regains use of the PCI Bus, the PCI 9054 resumes Memory Write and Invalidate cycles. Otherwise, it continues with a normal write. If a Target disconnects before a cache line is completed, the PCI 9054 completes the remainder of that cache line, using normal writes.



**Figure 3-5. Dual Address Timing**

## 3.4.2 IDMA/SDMA Operation

### 3.4.2.1 IDMA Operation

The PCI 9054 supports the MPC850 or MPC860 Independent DMA (IDMA) mode, using the MDREQ# signal and operating in PCI Initiator mode. In M mode, this signal is connected to the MPC850 or MPC860 DREQ[1:0] input pins. After programming the MPC850 or MPC860 IDMA channel, the PCI 9054 uses PCI Initiator mode to transfer data between the PCI Bus and the MPC850 or MPC860 internal dual-port RAM (or external memory). The data count is controlled by the IDMA Byte counter and throttled by the PCI 9054 MDREQ# signal. When the PCI 9054 FIFO is nearly full, MDREQ# is de-asserted to the MPC850 or MPC860, indicating that it should inhibit transferring further data (the FIFO threshold count in the PCI 9054 must be set to a value of at least five Lwords below the full capacity of the FIFO—27 Lwords) (DMPBAM[10, 8:5]). The Retry function can be used to communicate to the Local Bus Master that it should relinquish ownership of the Local Bus.

**Note:** The PCI Initiator Write FIFO Retry Enable bit (LMISC[6]) can be disabled to prevent assertion of the RETRY# signal.

In IDMA reads (PCI 9054 to the Local Bus), the MDREQ# signal is always asserted (indicating data is available), although the Read FIFO is empty. Any Local Bus read of the PCI Bus causes the PCI 9054 to become a PCI Bus Master and fill the PCI Initiator Read FIFO buffer. When sufficient data is in the FIFO, the PCI 9054 completes the Local Bus cycle by asserting Transfer Acknowledge (TA#).

After the IDMA has transferred all required bytes (MPC850 or MPC860 Byte counter decrements to zero), the MPC850 or MPC860 generate an internal interrupt, which in turn should execute the code to disable the IDMA channel (the MDREQ# input signal may still be asserted by the PCI 9054). The SDACK[1:0] signal from the MPC850 or MPC860 is not used by the PCI 9054 (no connection).

Refer to Section 3.4.1 for more information about PCI Initiator Data transfers.

### 3.4.2.2 SDMA Operation

The PCI 9054 supports the MPC850 or MPC860 Serial DMA (SDMA) mode, using PCI Initiator mode. No handshake signals are required to perform the SDMA operation.

The Retry function can be used to communicate to the Local Bus Master it should relinquish ownership of the Local Bus. The PCI Initiator Write FIFO Retry Enable bit (LMISC[6]) can be disabled to prevent assertion of the RETRY# signal.

**Note:** The PCI Initiator Write FIFO can be programmed to identify the full status condition (DMPBAM[10, 8:5]). The FIFO Full Status Flag is in MARBR[30].

## 3.4.3 PCI Target Operation (PCI Master-to-Local Bus Access)

The PCI 9054 supports both Burst Memory-Mapped Transfer accesses and I/O-Mapped, Single-Transfer accesses to the Local Bus from the PCI Bus through a 16-Lword (64-byte) PCI Target Read FIFO and a 32-Lword (128-byte) PCI Target Write FIFO. The PCI Base Address registers are provided to set up the location of the adapter in the PCI memory and the I/O space. In addition, Local mapping registers allow address translation from the PCI Address Space to the Local Address Space. Three spaces are available:

- Space 0
- Space 1
- Expansion ROM space

Expansion ROM space is intended to support a bootable ROM device for the Host.

For Single-Cycle PCI Target reads, the PCI 9054 reads a single Local Bus Lword or partial Lword. The PCI 9054 disconnects after one transfer for all PCI Target I/O accesses.

For the highest data-transfer rate, the PCI 9054 supports posted writes and can be programmed to prefetch data during a PCI Burst read. The Prefetch size, when enabled, can be from one to 16 Lwords or until the PCI Bus stops requesting. When the PCI 9054 prefetches, if enabled, it drops the Local Bus after reaching the prefetch counter. In Continuous Prefetch mode, the PCI 9054 prefetches as long as FIFO space is available and stops prefetching when the PCI Bus terminates the request. If Read prefetching is disabled, the PCI 9054 disconnects after one Read transfer.

In addition to Prefetch mode, the PCI 9054 supports Read Ahead mode (refer to Section 3.4.3.3).

Each Local space can be programmed to operate in an 8-, 16-, or 32-bit Local Bus width. The PCI 9054 has an internal wait state generator and external wait state input, TA#. TA# can be disabled or enabled with the Internal Configuration registers.

With or without wait state(s), the Local Bus, independent of the PCI Bus, can:

- Burst as long as data is available (Continuous Burst mode)
- Burst four Lwords at a time (recommended)
- Perform a Continuous Single cycle

A Burst cycle from the PCI Bus through the PCI 9054 asserts an MPC850 or MPC860 Burst transaction if the following is true:

- The address is quad-Lword aligned,
- A FIFO contains at least four Lwords, and
- All PCI Bus byte enables are set for writes only and ignored for reads

### 3.4.3.1 PCI Target Lock

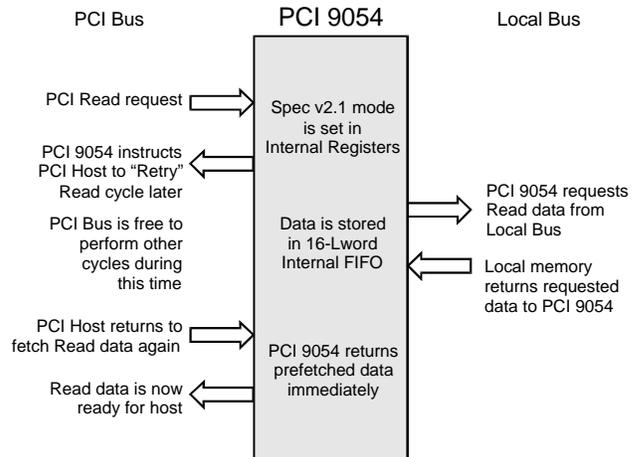
The PCI 9054 supports direct PCI-to-Local-Bus exclusive accesses (locked atomic operations). A PCI-locked operation to the Local Bus results in the entire address Space 0, Space 1, and Expansion ROM space being locked until they are released by the PCI Bus Master. Locked operations are enabled or disabled with the PCI Target LOCK# Enable bit (MARBR[22]) for PCI-to-Local accesses.

### 3.4.3.2 PCI Target PCI v2.1 Delayed Read Mode

The PCI 9054 can be programmed through the PCI Specification v2.1 Mode bit (MARBR[24]=1) to perform delayed reads, as specified in PCI Specification v2.1.

In addition to delayed reads, the PCI 9054 supports the following PCI Specification v2.1 functions:

- No write while a read is pending (PCI Retry for writes)
- Write and flush pending read

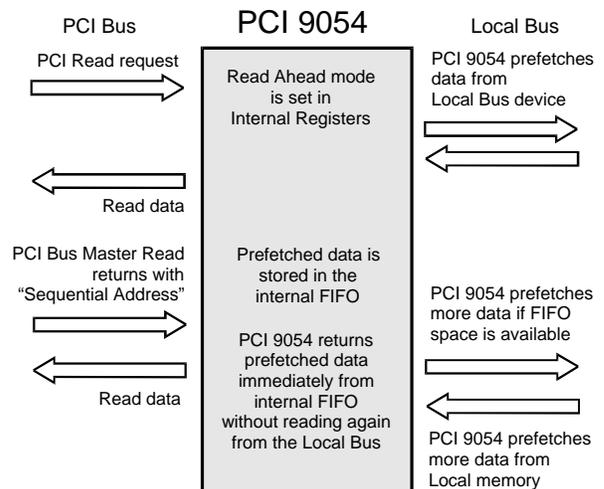


**Figure 3-6. PCI Target PCI v2.1 Delayed Reads**

**Note:** The figure represents a sequence of Bus cycles.

### 3.4.3.3 PCI Target PCI Read Ahead Mode

The PCI 9054 also supports Read Ahead mode, where prefetched data can be read from the internal FIFO of the PCI 9054 instead of from the Local Bus. The address must be subsequent to the previous address and 32-bit aligned (next address = current address + 4). Read Ahead mode functions with or without PCI Delayed Read mode.



**Figure 3-7. PCI Target PCI 9054 Read Ahead Mode**

**Note:** The figure represents a sequence of Bus cycles.

### 3.4.3.4 PCI Target Transfer

A PCI Bus Master addressing the Memory space decoded for the Local Bus initiates transactions. Upon a PCI Read/Write, the PCI 9054 becomes a Local Bus Master and arbitrates for the Local Bus.

The PCI 9054 then reads data into the PCI Target Read FIFO or writes data to the Local Bus.

The PCI Target or PCI Initiator preempts DMA; however, the PCI Target does not preempt the PCI Initiator (refer to Section 3.4.4.1).

The PCI 9054 can be programmed to “keep” the PCI Bus by generating a wait state(s) and de-asserting TRDY# if the Write FIFO becomes full. The PCI 9054 can also be programmed to “keep” the Local Bus and continue asserting BB# if the PCI Target Write FIFO becomes empty or the PCI Target Read FIFO becomes full. In either case, the Local Bus is dropped when the Local Bus Latency Timer is enabled and expires (MARBR[7:0]).

For PCI Target writes, the PCI Bus writes data to the Local Bus. PCI Target is the “Command from the PCI Host,” which has the highest priority.

For PCI Target reads, the PCI Bus Master reads data from the Local Bus Slave.

The PCI 9054 supports on-the-fly Endian conversion for Space 0, Space 1, and Expansion ROM space. The Local Bus can be Big/Little Endian (Address/Data Invariance) by using the programmable internal register configuration.

**Note:** The PCI Bus is always Little Endian.

During PCI Target transactions, the MPC850 or MPC860 user has the option to use the PCI 9054 for maximum Burst transfers, using the BTERM# Input Enable bit(s) (LBRD0[23,7], LBRD1[7], and DMAMODE0[7]).

In PCI Target transfers, each PCI Target space (Space 0, Space 1, and Expansion ROM) has its own BTERM# Input Enable bit (the BTERM# input signal becomes the BI# signal in M mode). Space 0 is in LBRD0[7], Space 1 is in LBRD1[7], and Expansion ROM is in LBRD0[23].

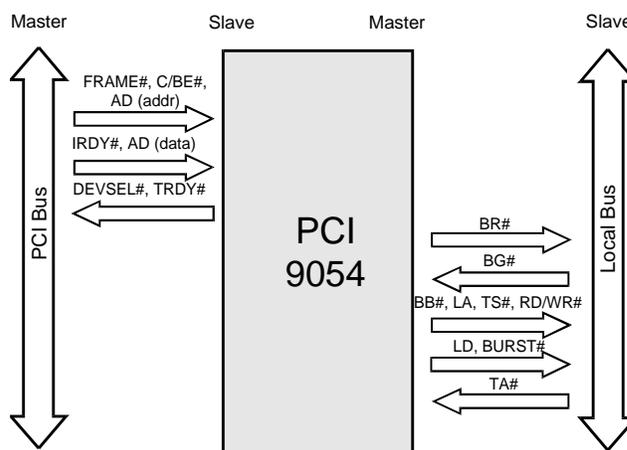


Figure 3-8. PCI Target Write

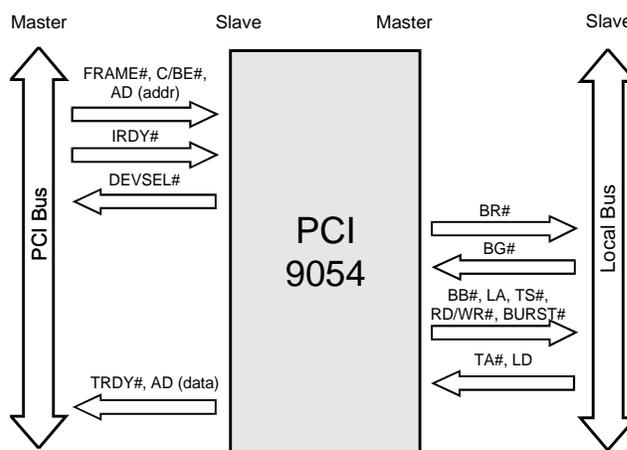


Figure 3-9. PCI Target Read

**Note:** The figures represent a sequence of Bus cycles.

When the Bterm Mode bit is enabled, the PCI 9054 continues to burst on the Local Bus until the BI# signal is asserted for one CLK cycle any time after the first Data phase, implying a new Address cycle (TS#) is needed if there is more data to transfer. If the BI# signal is asserted on the first Data phase, the Burst transfer is broken into Single-Cycle transactions.

When the Bterm Mode bit is enabled and the BI# signal asserted for one CLK cycle any time after the First data phase, this implies that a new Address cycle (TS#) is needed for more data to transfer.

Section 3—M Func Desc

Regardless of the Bterm mode setting, if the BI# signal is asserted on the first Data phase, Single-Cycle transfers are performed until the quad-word boundary is reached.

**Table 3-2. PCI Target Burst Mode Cycle Detection**

Burst Enable Bit	BTERM# Input Enable Bit	BI# Signal	Result
1	0	Not asserted	Burst 16 bytes (MPC850 or MPC860 compatible)
1	0	Asserted during first Data phase	Single cycle
1	1	Asserted after first Data phase	Burst until BI# is asserted for one CLK cycle
0	X	X	Single cycle

**Caution:** The MPC850 and MPC860 do not support bursting more than 16 bytes. The BTERM# Input Enable bits should be set only for Local Bus Masters that support continuous bursting.

**Note:** “X” is “Don’t Care.”

The PCI 9054 supports Local Bus error conditions using TEA#. TEA# may be asserted by a device on the Local Bus, either before or simultaneously with TA#. In either case, the PCI 9054 tries to complete the current transaction by transferring data and then asserting TS# for every address that follows, waiting for another TA# or TEA# to be issued (used to flush PCI Target FIFOs). After acknowledging TEA# is asserted, the PCI 9054 asserts PCI SERR# and sets an error flag, using the Signaled System Error bit (PCISR[14]=1). When set, this indicates a catastrophic error occurred on the Local Bus. SERR# may be masked off by resetting the TEA# Input Interrupt Mask bit (LMISC[5]=0).

The PCI 9054 Local Bus Latency Timer (MARBR[7:0]) can be used to better utilize the Local Bus.

### 3.4.3.5 PCI Target PCI-to-Local Address Mapping

**Note:** Not applicable in I<sub>2</sub>O mode.

Three Local Address spaces—Space 0, Space 1, and Expansion ROM—are accessible from the PCI Bus. Each is defined by a set of three registers:

- Local Address Range (LAS0RR, LAS1RR, and/or EROMRR)
- Local Base Address (LAS0BA, LAS1BA, and/or EROMBA)
- PCI Base Address (PCIBAR2, PCIBAR3, and/or PCIERBAR)

A fourth register, the Bus Region Descriptor register for PCI-to-Local Accesses (LBRD0 and/or LBRD1), defines the Local Bus characteristics for the PCI Target regions (refer to Figure 3-10).

Each PCI-to-Local Address space is defined as part of reset initialization, as described in Section 3.4.3.5.1. These Local Bus characteristics can be modified at any time before actual data transactions.

#### 3.4.3.5.1 PCI Target Local Bus Initialization

**Range**—Specifies which PCI Address bits to use for decoding a PCI access to Local Bus space. Each bit corresponds to a PCI Address bit. Bit 31 corresponds to address bit 31. Write 1 to all bits that must be included in decode and 0 to all others.

**Remap PCI-to-Local Addresses into a Local Address Space**—Bits in this register remap (replace) the PCI Address bits used in decode as the Local Address bits.

**Local Bus Region Descriptor**—Specifies the Local Bus characteristics.

#### 3.4.3.5.2 PCI Target PCI Initialization

After a PCI reset, the software determines how much address space is required by writing all ones (1) to a PCI Base Address register and then reading back the value. The PCI 9054 returns zeroes (0) in the Don’t Care Address bits, effectively specifying the address space required. The PCI software then maps the Local Address space into the PCI Address space by programming the PCI Base Address register. (Refer to Figure 3-10.)

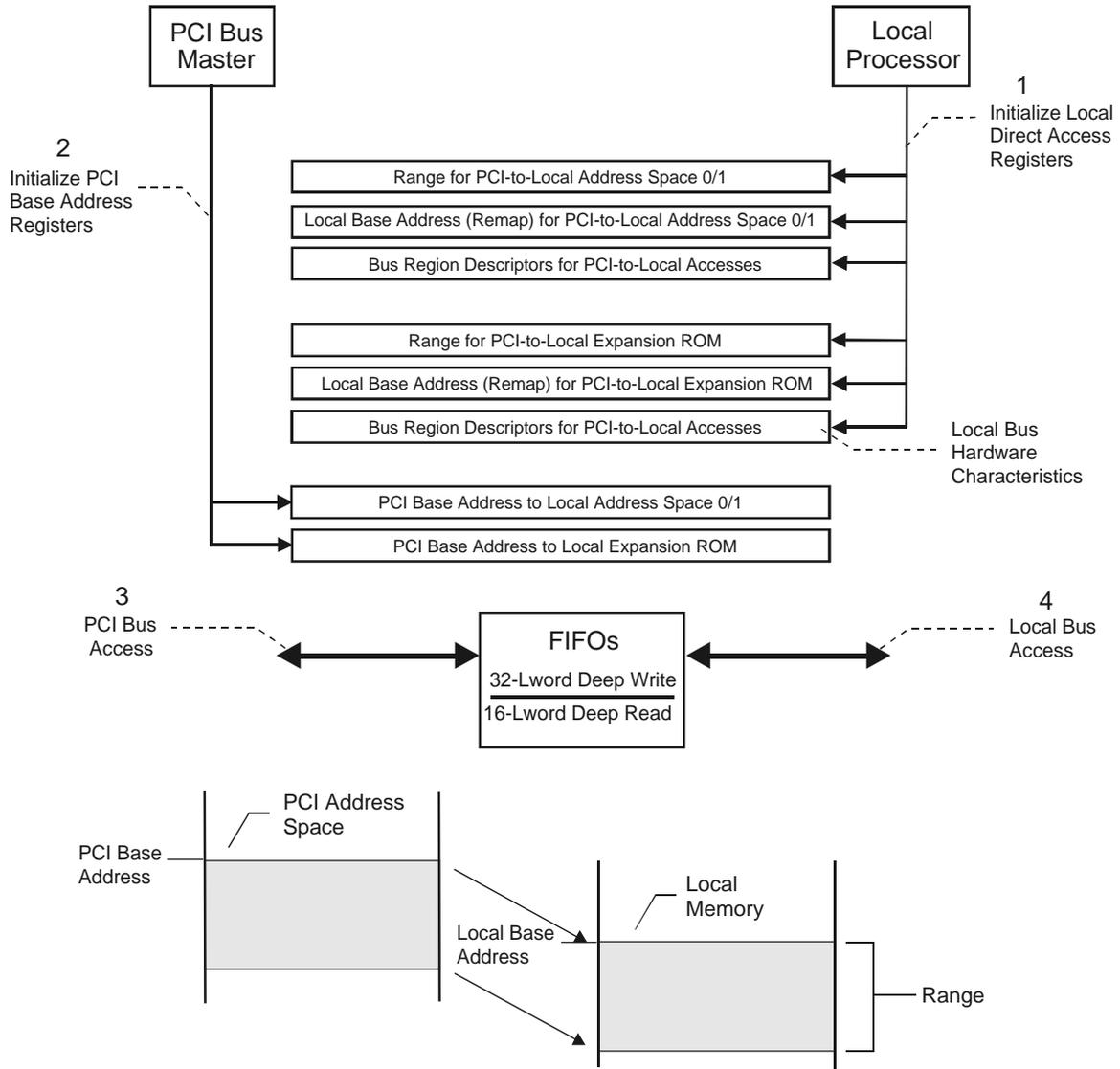


Figure 3-10. Local Bus PCI Target Access

### 3.4.3.5.3 PCI Target Transfer Size

The TSIZ[0:1] pins correspond to the data-transfer size on the Local Bus.

Table 3-3. Data Bus TSIZ[0:1] Contents for Single Write Cycles

Transfer Size	TSIZ [0:1]		Address		External Data Bus Pattern For 32-, 16-, and 8-Bit Port Sizes			
			LA30	LA31	LD[0:7]	LD[8:15]	LD[16:23]	LD[24:31]
Byte	0	1	0	0	OP0	—	—	—
	0	1	0	1	OP1	OP1	—	—
	0	1	1	0	OP2	—	OP2	—
	0	1	1	1	OP3	OP3	—	OP3
Word	1	0	0	0	OP0	OP1	—	—
	1	0	1	0	OP2	OP3	OP2	OP3
Lword	0	0	0	0	OP0	OP1	OP2	OP3

Table 3-4. Data Bus TSIZ[0:1] Requirements for Single Read Cycles

Transfer Size	TSIZ [0:1]		Address		32-Bit Port Size				16-Bit Port Size		8-Bit Port Size
			LA30	LA31	LD[0:7]	LD[8:15]	LD[16:23]	LD[24:31]	LD[0:7]	LD[8:15]	LD[0:7]
Byte	0	1	0	0	OP0	—	—	—	OP0	—	OP0
	0	1	0	1	—	OP1	—	—	—	OP1	OP1
	0	1	1	0	—	—	OP2	—	OP2	—	OP2
	0	1	1	1	—	—	—	OP3	—	OP3	OP3
Word	1	0	0	0	OP0	OP1	—	—	OP0	OP1	OP0
	1	0	1	0	—	—	OP2	OP3	OP2	OP3	OP2
Lword	0	0	0	0	OP0	OP1	OP2	OP3	OP0	OP1	OP0

#### 3.4.3.5.3.1 PCI Target Example

A 1 MB Local Address Space, 12300000h through 123FFFFFFh, is accessible from the PCI Bus at PCI addresses 78900000h through 789FFFFFFh.

- a. Local initialization software sets the Range and Local Base Address registers as follows:
  - **Range**—FFF00000h (1 MB, decode the upper 12 PCI Address bits)
  - **Local Base Address (Remap)**—123XXXXXh (Local Base Address for PCI-to-Local accesses [Space Enable bit(s) must be set to be recognized by the PCI Host (LAS0BA[0]=1, LAS1BA[0]=1)])

- b. PCI Initialization software writes all ones to the PCI Base Address, then reads it back again.
  - The PCI 9054 returns a value of FFF00000h. The PCI software then writes to the PCI Base Address register(s).
  - **PCI Base Address**—789XXXXXh (PCI Base Address for Access to the Local Address Space registers, PCIBAR2 and PCIBAR3).

For a PCI Direct access to the Local Bus, the PCI 9054 has a 32-Lword (128-byte) Write FIFO and a 16-Lword (64-byte) Read FIFO. The FIFOs enable the Local Bus to operate independent of the PCI Bus. The PCI 9054 can be programmed to return a Retry response or to throttle TRDY# for any PCI Bus transaction attempting to write to the PCI 9054 Local Bus when the FIFO is full.

For PCI Read transactions from the Local Bus, the PCI 9054 holds off TRDY# while gathering data from the Local Bus. For Read accesses mapped to PCI Memory space, the PCI 9054 prefetches up to 16 Lwords (has Continuous Prefetch mode) from the Local Bus. Unused Read data is flushed from the FIFO. For Read accesses mapped to PCI I/O space, the PCI 9054 does not prefetch Read data. Rather, it breaks each read of a Burst cycle into a Single Address/Data cycle on the Local Bus.

The PCI Target Retry Delay Clocks bits (LBRD0[31:28]) can be used to program the period of time in which the PCI 9054 holds off TRDY#. The PCI 9054 issues a Retry to the PCI Bus Transaction Master when the programmed time period expires. This occurs when the PCI 9054 cannot gain control of the Local Bus and return TRDY# within the programmed time period.

### 3.4.3.6 PCI Target Priority

PCI Target accesses have a higher priority than DMA accesses, thereby preempting DMA transfers. During a DMA transfer, if the PCI 9054 detects a pending PCI Target access, it releases the Local Bus within two Data transfers. The PCI 9054 resumes operation after the PCI Target access completes.

When the PCI 9054 DMA controller owns the Local Bus, its BR# output and BG# input are asserted. When a PCI Target access occurs, the PCI 9054 releases the Local Bus within two Lword transfers by de-asserting BB# and floating the Local Bus outputs. After the PCI 9054 acknowledges that BG# is de-asserted, it requests the Local Bus for a PCI Target transfer by asserting BR#. When the PCI 9054 receives BG#, it drives the bus and performs the PCI Target transfer. Upon completing a PCI Target transfer, the PCI 9054 releases the Local Bus by de-asserting BB# and floating the Local Bus outputs. After the PCI 9054 acknowledges that BG# is de-asserted and the Local Bus Pause Timer is set to zero, it requests a DMA transfer from the Local Bus by re-asserting BR#. When it receives BG#, it drives the bus and continues the DMA transfer.

### 3.4.4 Deadlock Conditions

Deadlock can occur when a PCI Bus Master must access the PCI 9054 Local Bus at the same time a Master on the PCI 9054 Local Bus must access the PCI Bus.

There are two types of deadlock:

- **Partial Deadlock**—A Local Bus Master is performing a Direct Bus Master access to a PCI Bus device other than the PCI Bus device concurrently trying to access the Local Bus
- **Full Deadlock**—A Local Bus Master is performing a Direct Bus Master access to the same PCI Bus device concurrently trying to access the Local Bus

This applies only to PCI Initiator and PCI Target accesses through the PCI 9054. Deadlock does not occur in transfers through the PCI 9054 DMA channels or the PCI 9054 internal registers (such as mailboxes).

For partial deadlock, the PCI access to the Local Bus times out (the PCI Target Retry Delay Clock (LBRD0[31:28]), which is programmable through the Local Bus Region Descriptor register) and the PCI 9054 responds with a PCI Retry. The PCI Specification requires that a PCI Master release its request for the PCI Bus (de-assert REQ#) for a minimum of two PCI clocks after receiving a Retry. This allows the PCI Bus arbiter to grant the PCI Bus to the PCI 9054 so that it can complete its PCI Initiator access and free up the Local Bus. Possible solutions are described in the following sections for cases in which the PCI Bus arbiter does not function as described (PCI Bus architecture dependent), waiting for a time out is undesirable, or a full deadlock condition exists.

When a full deadlock occurs, the only solution is to backoff the Local Bus Master.

### 3.4.4.1 Backoff

The PCI 9054 Local RETRY# signal indicates whether a possible deadlock condition exists. The PCI 9054 starts the Backoff timer (programmable through registers) when it detects one of the following conditions:

- A PCI Bus Master is attempting to access memory or an I/O device on the Local Bus and is not gaining access (*for example*, BG# is not received).
- A Local Bus Master is performing a Direct Bus Master Read access to the PCI Bus. Or, a Local Bus Master is performing a Direct Bus Master Write access to the PCI Bus and the PCI 9054 PCI Initiator Write FIFO cannot accept another Write cycle.

If the Local Bus Backoff Enable bit is enabled (EROMBA[4]=1), the Backoff Timer expires, and the PCI 9054 has not received BG#, the PCI 9054 asserts RETRY#. External bus logic can use this signal to perform backoff.

The Backoff cycle is device/bus architecture dependent. The external logic (arbiter) can assert the necessary signals to cause the Local Bus Master to release the Local Bus (backoff). After the Local Bus Master backs off, it can grant the bus to the PCI 9054 by asserting BG#.

Once RETRY# is asserted, TA# for current Data cycle is never asserted (the Local Bus Master must perform a backoff). When the PCI 9054 detects BG#, it proceeds with the PCI Master-to-Local-Bus access. When this access completes and the PCI 9054 releases the Local Bus, external logic can then release the backoff and the Local Bus Master can resume the cycle interrupted by the Backoff cycle. The PCI 9054 Write FIFO retains all data acknowledged (*that is*, last data for which TA# was asserted).

After the backoff condition ends, the Local Bus Master restarts the last cycle with TS#. For writes, data following TS# should be the data the PCI 9054 did not acknowledge prior to the Backoff cycle (*for example*, the last data for which TA# is not asserted).

All PCI Read cycles completed before the Local Bus was backed off remain in the PCI Initiator Read FIFO. Therefore, if the Local Bus Master returns with the same last cycle, the cycle is acknowledged with the data currently in the FIFO (the FIFO data is not read twice).

A new PCI read is performed if the resumed Local Bus cycle is not the same as the Backed Off cycle.

#### 3.4.4.1.1 Software/Hardware Solution for Systems without Backoff Capability

For adapters that do not support backoff, a possible deadlock solution is as follows.

PCI Host software, external Local Bus hardware, general-purpose output USERo and general-purpose input USERi can be used to prevent deadlock. USERo can be asserted to request that the external arbiter not grant the bus to any Local Bus Master except the PCI 9054. Status output from the Local arbiter can be connected to the general purpose input USERi to indicate that no Local Bus Master owns the Local Bus, or the PCI Host to determine that no Local Bus Master that currently owns the Local Bus can read input. The PCI Host can then perform PCI Target access. When the Host finishes, it de-asserts USERo.

#### 3.4.4.1.2 Preempt Solution

For devices that support preempt, USERo can be used to preempt the current Local Bus Master device. When USERo is asserted, the current Local Bus Master device completes its current cycle and releases the Local Bus, de-asserting BB#.

### 3.4.4.2 Software Solutions to Deadlock

Both PCI Host and Local Bus software can use a combination of mailbox registers, doorbell registers, interrupts, direct Local-to-PCI accesses and direct PCI-to-Local accesses to avoid deadlock.

## 3.5 DMA OPERATION

The PCI 9054 supports two independent DMA channels capable of transferring data from the:

- Local-to-PCI Bus
- PCI-to-Local Bus

Each channel consists of a DMA controller and a dedicated, bidirectional FIFO. Both channels support Block transfers, and Scatter/Gather transfers, with or without End of Transfer (EOT#). Only DMA Channel 0 supports Demand mode DMA transfers. Master mode must be enabled with the Master Enable bit (PCICR[2]) before the PCI 9054 can become a PCI

Bus Master. In addition, both DMA channels can be programmed to:

- Operate in 8-, 16-, or 32-bit Local Bus width
- Use zero to 15 internal wait states (Local Bus)
- Enable/disable internal wait states (Local Bus)
- Enable/disable Local Bus Burst capability
- Limit Local Bus bursts to four (BTERM# enable/disable)
- Hold Local address constant (Local Target is FIFO) or increment
- Perform PCI Memory Write and Invalidate (command code = Fh) or normal PCI Memory Write (command code = 7h)
- Pause Local transfer with/without BLAST# (DMA Fast/Slow termination)
- Assert PCI interrupt (INTA#) or Local interrupt (LINT#) when DMA transfer is complete or Terminal Count is reached during Scatter/Gather DMA mode transfers
- Operate in DMA Clear Count mode (only if the descriptor is in Local memory)

The PCI 9054 also supports PCI Dual Address with the upper 32-bit registers (DMADAC0 and DMADAC1).

The Local Bus Latency Timer determines the number of Local clocks the PCI 9054 can burst data before relinquishing the Local Bus. The Local Pause Timer sets how soon the DMA channel can request the Local Bus.

### 3.5.1 DMA PCI Dual Address Cycle

The PCI 9054 supports PCI Dual Address Cycle (DAC) when it is a PCI Bus Master using the DMADAC0 and DMADAC1 registers for Block DMA transactions. Scatter/Gather DMA can utilize the DAC function via the DMADAC0 and DMADAC1 registers or DMAMODE0[18] and DMAMODE1[18]. The DAC command is used to transfer a 64-bit address to devices that support 64-bit addressing when the address is above the 4-GB Address space. The PCI 9054 performs a DAC within two PCI clock periods, when the first PCI address is a Lo-Addr, with the command (C/BE[3:0]#) “D”, and the second PCI address is a Hi-Addr, with the command (C/BE[3:0]#) “6” or “7”, depending upon whether it is a PCI Read or PCI Write cycle.

### 3.5.2 Block DMA Mode

The Host processor or the Local processor sets the Local and PCI starting addresses, transfer byte count, and transfer direction. The Host or Local processor then sets the DMA Start bit to initiate a transfer. The PCI 9054 requests the PCI and Local Buses and transfers data. Once the transfer completes, the PCI 9054 sets the Channel Done bit(s) (DMACSR0[4]=1 and/or DMACSR1[4]=1) and, if enabled, asserts an interrupt(s) (DMAMODE0[10] and/or DMAMODE1[10]) to the Local processor or the PCI Host (programmable). The Channel Done bit(s) can be polled, instead of interrupt generation, to indicate the DMA transfer status.

DMA registers are accessible from the PCI and Local Buses (refer to Figure 3-4 on page 3-7).

During DMA transfers, the PCI 9054 is a Master on both the PCI and Local Buses. For simultaneous access, PCI Target or PCI Initiator has a higher priority than DMA.

The PCI 9054 releases the PCI Bus if one of the following conditions occur (refer to Figure 3-11 on page 3-18 and Figure 3-12 on page 3-18):

- FIFO is full (PCI-to-Local Bus)
- FIFO is empty (Local-to-PCI Bus)
- Terminal count is reached
- PCI Bus Latency Timer expires (PCILTR[7:0])—normally programmed by the Host PCI BIOS—and PCI GNT# de-asserts
- PCI Host asserts STOP#

The PCI 9054 releases the Local Bus if one of the following conditions occurs:

- FIFO is empty (PCI-to-Local Bus)
- FIFO is full (Local-to-PCI Bus)
- Terminal count is reached
- Local Bus Latency Timer is enabled and expires (MARBR[7:0])
- Special cycle BI# input is asserted
- PCI Target request is pending

During DMA transactions, users have the option of using the Burst Forever BTERM# Input Enable bit(s) (DMAMODE0[7] and/or DMAMODE1[7]) if the External Memory Controller is provided. Used in conjunction with the Fast/Slow Terminate Mode Select bit(s) (DMAMODE0[15] and/or DMAMODE1[15]).

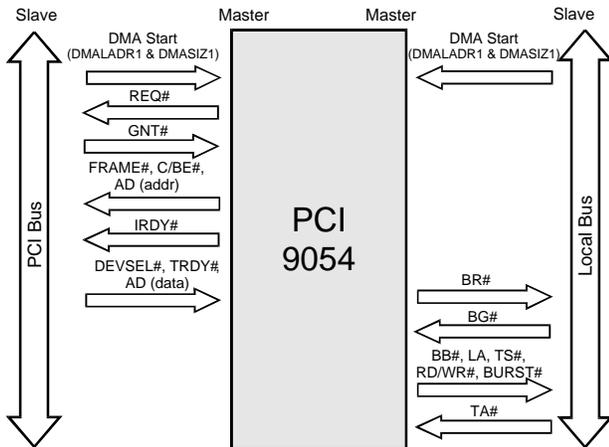


Figure 3-11. DMA, PCI-to-Local Bus

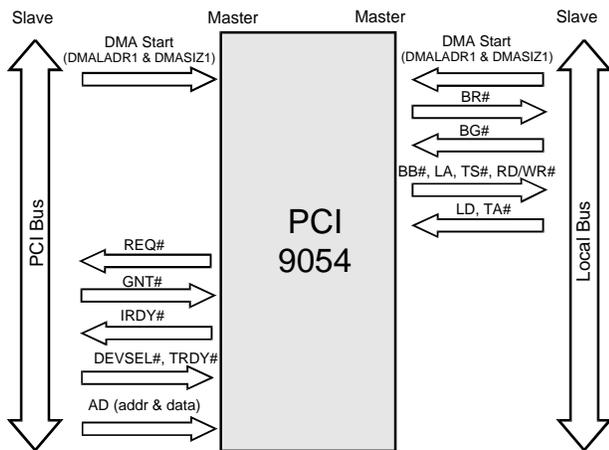


Figure 3-12. DMA, Local-to-PCI Bus

**Note:** The figures represent a sequence of Bus cycles.

Table 3-5. DMA

BTERM# Input Enable Bit(s)	Fast/Slow Terminate Mode Select Bit(s)	PCI 9054 BDIP# Output
Enabled (1)	Disabled (1)	BDIP# is not asserted. Burst forever or until BI# asserts for one CLK cycle.
Enabled (1)	Enabled (0)	BDIP# is asserted on the last data transfer, or until BI# asserts for one CLK cycle. (Refer to Section 2.2.5.2.1.)
Disabled (0)	Disabled (1)	BDIP# is not asserted. Burst forever.
Disabled (0)	Enabled (0)	BDIP# is asserted by the PCI 9054. Burst up to 16 bytes (MPC850 or MPC860 compatible).

Table 3-6. Normal DMA with EOT Function

BTERM# Enable Bit(s)	Fast/Slow Terminate Mode Select Bit(s)	PCI 9054 BDIP# Output
Enabled (1)	Disabled (1)	BDIP# is not asserted. Immediate transfer terminated by EOT#.
Enabled (1)	Enabled (0)	
Disabled (0)	Disabled (1)	
Disabled (0)	Enabled (0)	BDIP# is asserted by the PCI 9054. Transfers up to the nearest 16-byte boundary, then terminates (MPC850 or MPC860 compatible).

**Note:** If the Burst Enable bit is set to 0, the PCI 9054 performs Single-Cycle transfers on the Local Bus.

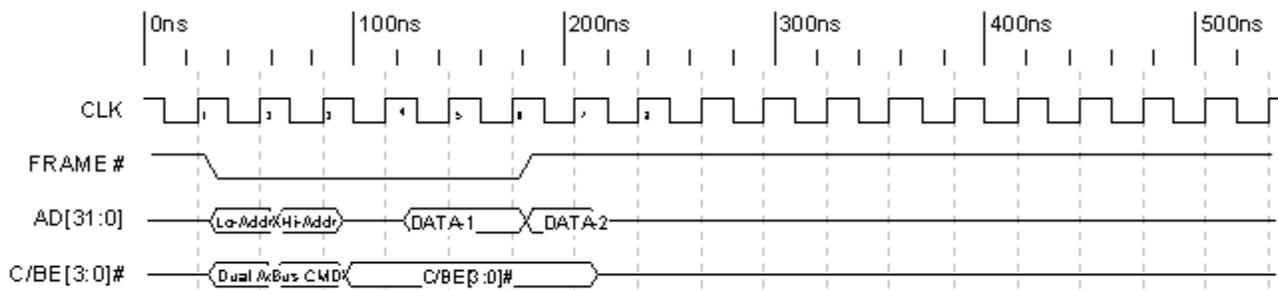


Figure 3-13. Dual Address Timing

### 3.5.2.1 Block DMA PCI Dual Address Cycle

The PCI 9054 supports the DAC feature in Block DMA mode. Whenever the DMADAC0 or DMADAC1 registers contain a value of 0x00000000, the PCI 9054 performs a Single Address Cycle (SAC) on the PCI Bus. Any other value causes a Dual Address to appear on the PCI Bus. (Refer to Figure 3-13.)

### 3.5.3 Scatter/Gather DMA Mode

In Scatter/Gather DMA mode, the Host processor or Local processor sets up descriptor blocks in Local or Host memory composed of PCI and Local addresses, transfer count, transfer direction, and address of next descriptor block (refer to Figure 3-14 and Figure 3-15). The Host or Local processor then:

- Enables the Scatter/Gather mode bit(s) (DMAMODE0[9]=1 and/or DMAMODE1[9]=1)
- Sets up the address of initial descriptor block in the PCI 9054 Descriptor Pointer register(s) (DMADPR0 and/or DMADPR1)
- Initiates the transfer by setting a control bit(s) (DMACSR0[1:0] and/or DMACSR1[1:0])

The PCI 9054 loads the first descriptor block and initiates the Data transfer. The PCI 9054 continues to load descriptor blocks and transfer data until it detects the End of Chain bit(s) (DMADPR0[1] and/or DMADPR1[1]) is set (these bits are part of each descriptor). When the End of Chain bit(s) is detected, the PCI 9054 completes the current descriptor block and sets the DMA Done bit(s) (DMACSR0[4] and/or DMACSR1[4]). If the End of Chain bit(s) is detected, the PCI 9054 asserts a PCI interrupt (INTA#) and/or Local interrupt (LINT#).

The PCI 9054 can also be programmed to assert PCI or Local interrupts after each descriptor is loaded, then finish transferring.

If Scatter/Gather descriptors are in Local memory, the DMA controller can be programmed to clear the transfer size at completion of each DMA, using the DMA Clear Count Mode bit(s) (DMAMODE0[16] and/or DMAMODE1[16]).

**Notes:** In Scatter/Gather DMA mode, the descriptor includes the PCI and Local Address Space, transfer size, and next descriptor pointer. It also includes a DAC value if DMADPR0[18] and/or DMAMODE1[18] is enabled. Otherwise, the register (DMADAC0 and DMADAC1) values are used.

The Descriptor Pointer register(s) (DMADPR0 and/or DMADPR1) contains end of chain (bit 1), direction of transfer (bit 3), next descriptor address (bits [31:4]), interrupt after terminal count (bit 2), and next descriptor location (bit 0) bits.

The Local Bus width must be the same as Local Memory bus width. A DMA descriptor can be on the Local memory or the PCI memory, or both (for example, one descriptor on Local memory, another descriptor on PCI memory and vice-versa).

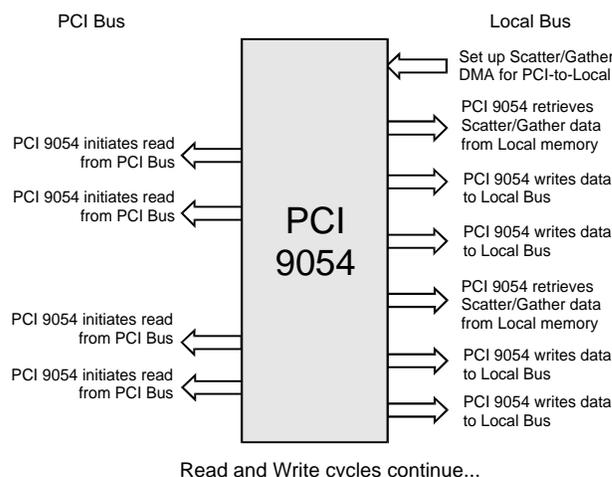


Figure 3-14. Scatter/Gather DMA Mode from PCI-to-Local Bus (Control Access from the Local Bus)

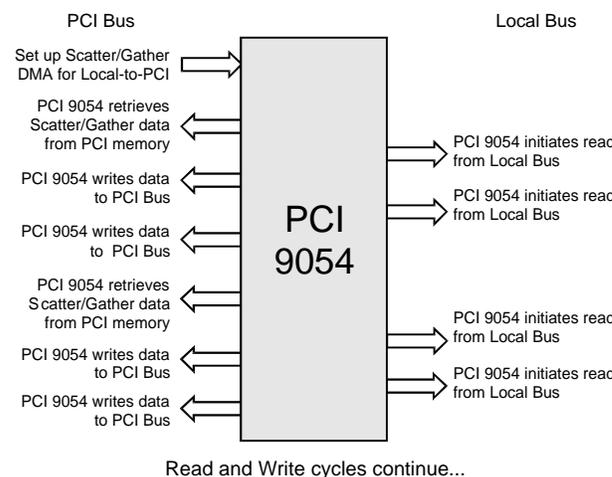


Figure 3-15. Scatter/Gather DMA Mode from Local-to-PCI Bus (Control Access from the PCI Bus)

**Note:** The figures represent a sequence of Bus cycles.

Section 3—M Func Desc

### 3.5.3.1 Scatter/Gather DMA PCI Dual Address Cycle

The PCI 9054 supports the DAC feature in Scatter/Gather DMA mode for Data transfers only. The descriptor blocks should reside below the 4-GB Address space.

The PCI 9054 offers three different options of how PCI DAC Scatter/Gather DMA is utilized. Assuming the descriptor blocks are located on the PCI Bus:

- DMADAC0 and/or DMADAC1 contain(s) a non-zero value. DMAMODE0[18] and/or DMAMODE1[18] is set to 0. The PCI 9054 performs a Single Address Cycle (SAC) four-Lword descriptor block load from PCI memory and DMA transfer with DAC on the PCI Bus. (Refer to Figure 3-16.)
- DMADAC0 and/or DMADAC1 contain(s) an 0x00000000 value. DMAMODE0[18] and/or DMAMODE1[18] is set to 1. The PCI 9054 performs a SAC five-Lword descriptor block load from PCI memory and DMA transfer with DAC on the PCI Bus. (Refer to Figure 3-17.)
- DMADAC0 and/or DMADAC1 contain(s) a non-zero value. DMAMODE0[18] and/or DMAMODE1[18] is set to 1. The PCI 9054 performs a SAC five-Lword descriptor block load from PCI memory and DMA transfer with DAC on the PCI Bus. The fifth descriptor overwrites the value of the DMADAC0 and/or DMADAC1 register. (Refer to Figure 3-17.)

### 3.5.3.2 DMA Clear Count Mode

The PCI 9054 supports DMA Clear Count mode (Write-Back feature, DMAMODE0[16] and DMAMODE1[16]). This feature allows users to control the data transfer blocks during Scatter/Gather DMA operations. The PCI 9054 clears the Transfer Size descriptor to zero by writing to a descriptor-memory location at the end of each transfer chain. This feature works only if DMA descriptors are on the Local Bus.

### 3.5.4 DMA Memory Write and Invalidate

The PCI 9054 can be programmed to perform Memory Write and Invalidate cycles to the PCI Bus for DMA transfers, as well as PCI Initiator transfers (refer to Section 3.4.1.10). The PCI 9054 supports Memory Write and Invalidate transfers for cache line sizes of 8 or 16 Lwords. Size is specified in the System Cache Line Size bits (PCICLSR[7:0]). If a size other than 8 or

16 is specified, the PCI 9054 performs Write transfers rather than Memory Write and Invalidate transfers.

DMA Memory Write and Invalidate transfers are enabled when the DMA controller Memory Write and Invalidate Enable bit(s) (DMAMODE0[13] and/or DMAMODE1[13]) and the Memory Write and Invalidate Enable bit (PCICR[4]) are set.

In Memory Write and Invalidate mode, the PCI 9054 waits until the number of Lwords required for specified cache line size are read from the Local Bus before starting the PCI access. This ensures a complete cache line write can complete in one PCI Bus ownership. If a Target disconnects before a cache line completes, the PCI 9054 completes the remainder of that cache line, using normal writes before resuming Memory Write and Invalidate transfers. If a Memory Write and Invalidate cycle is in progress, the PCI 9054 continues to burst if another cache line is read from the Local Bus before the cycle completes. Otherwise, the PCI 9054 terminates the burst and waits for the next cache line to be read from the Local Bus. If the final transfer is not a complete cache line, the PCI 9054 completes the DMA transfer, using normal writes.

### 3.5.4.1 DMA Abort

DMA transfers can be aborted, in addition to the EOT# signal, as follows:

1. Clear the DMA Channel Enable bit(s) (DMACSR0[0]=0 and/or DMACSR1[0]=0).
2. Abort DMA by setting the Channel Abort bit(s) (DMACSR0[2]=1 and/or DMACSR1[2]=1).
3. Wait until the Channel Done bit(s) is set (DMACSR0[4]=1 and/or DMACSR1[4]=1).

**Note:** One to two Data transfers occur after the Abort bit is set. Aborting when no DMA cycles are in progress causes the next DMA to abort.

### 3.5.5 DMA Priority

The DMA Channel Priority bit (MARBR[20:19]) can be used to specify the following priorities:

- Rotating (MARBR[20:19]=00)
- DMA Channel 0 (MARBR[20:19]=01)
- DMA Channel 1 (MARBR[20:19]=10)

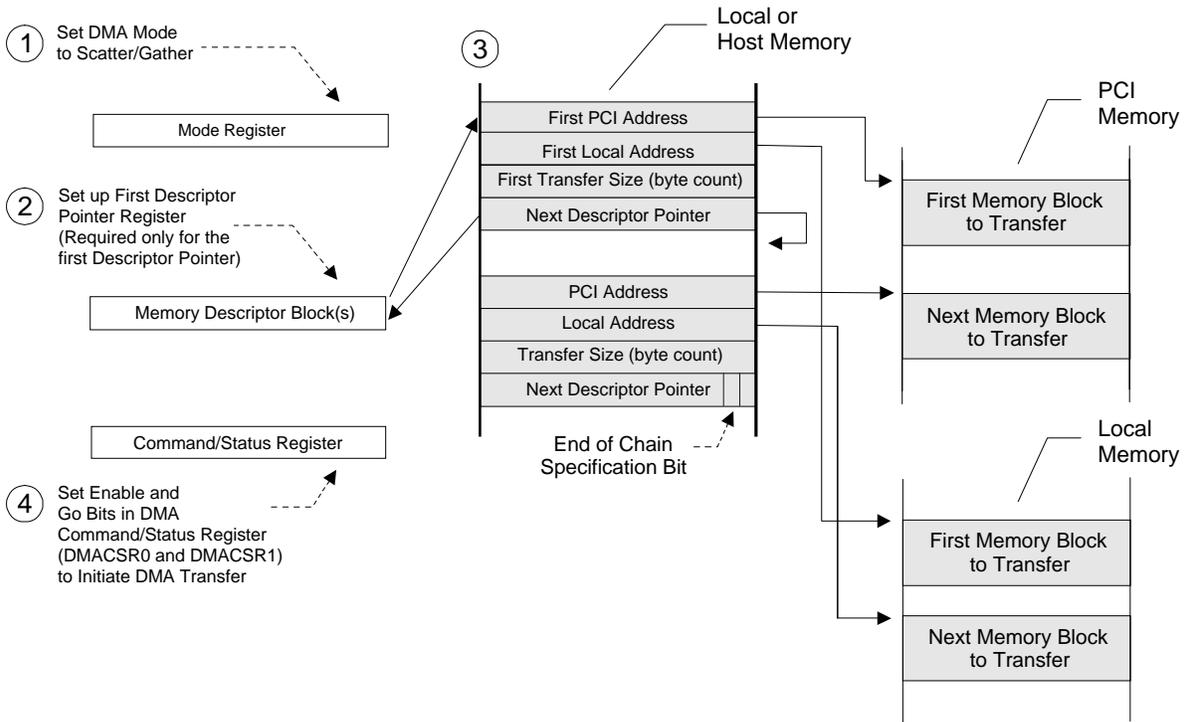


Figure 3-16. Scatter/Gather DMA Mode Descriptor Initialization [PCI SAC/DAC PCI Address (DMADAC0, DMADAC1) Register Dependent]

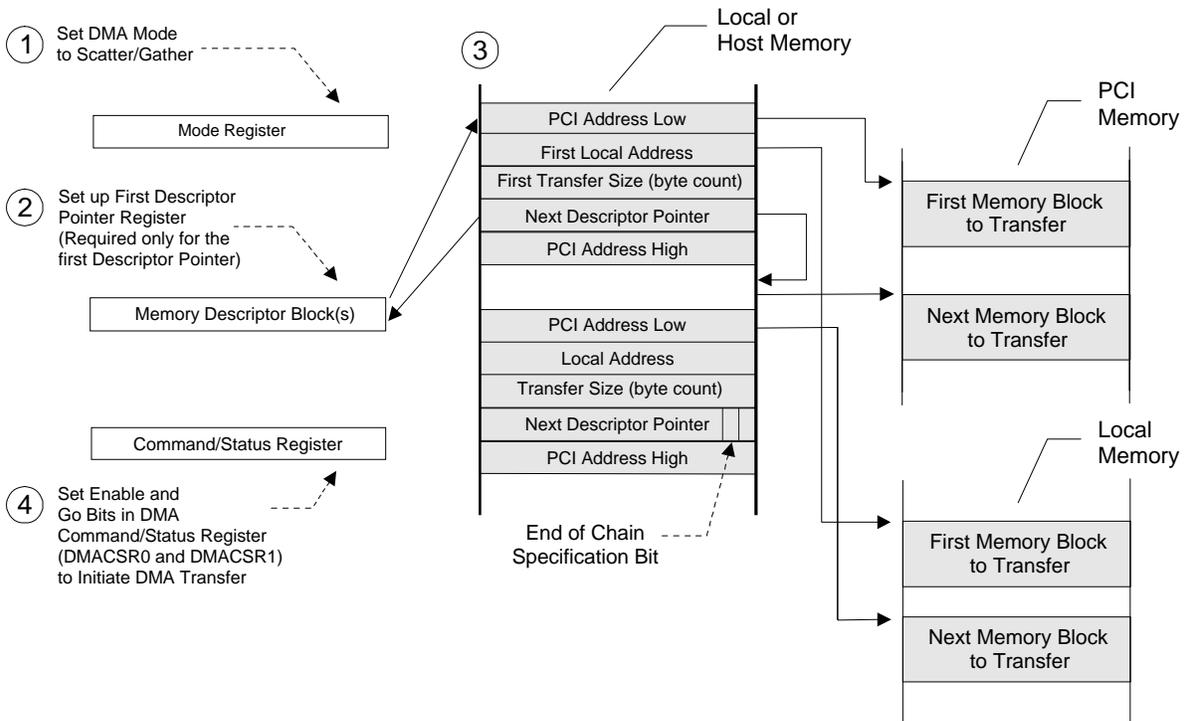


Figure 3-17. Scatter/Gather DMA Mode Descriptor Initialization [DAC PCI Address (DMAMODE0[18], DMAMODE1[18]) Descriptor Dependent (PCI Address High Added)]

### 3.5.6 DMA Channel 0/1 Interrupts

A DMA channel can assert a PCI Bus or Local Bus interrupt when done (transfer complete) or after a transfer is complete for the current descriptor in Scatter/Gather DMA mode. The DMA Channel Interrupt Select bit(s) determine whether to assert a PCI (DMAMODE0[17]=1 and/or DMAMODE1[17]=1) or Local (DMAMODE0[17]=0 and/or DMAMODE1[17]=0) interrupt. The Local or PCI processor can read the DMA Channel 0 Interrupt Active bits to determine whether a DMA Channel 0 (INTCSR[21]) or DMA Channel 1 (INTCSR[22]) interrupt is pending.

The Channel Done bit(s) (DMACSR0[4] and/or DMACSR1[4]) can be used to determine whether an interrupt is:

- DMA Done interrupt
- Transfer complete for current descriptor interrupt

The Done Interrupt Enable bit(s) (DMAMODE0[10] and/or DMAMODE1[10]) enable a Done interrupt. In Scatter/Gather DMA mode, a bit in the Next Descriptor Pointer register of the channel (loaded from Local memory) specifies whether to assert an interrupt at the end of the transfer for the current descriptor.

A DMA Channel interrupt is cleared by the Channel Clear Interrupt bit(s) (DMACSR0[3]=1 and/or DMACSR1[3]=1).

### 3.5.7 DMA Data Transfers

The PCI 9054 DMA controller can be programmed to transfer data from the Local-to-PCI Bus or from the PCI-to-Local Bus.

### 3.5.7.1 Local-to-PCI Bus DMA Transfer

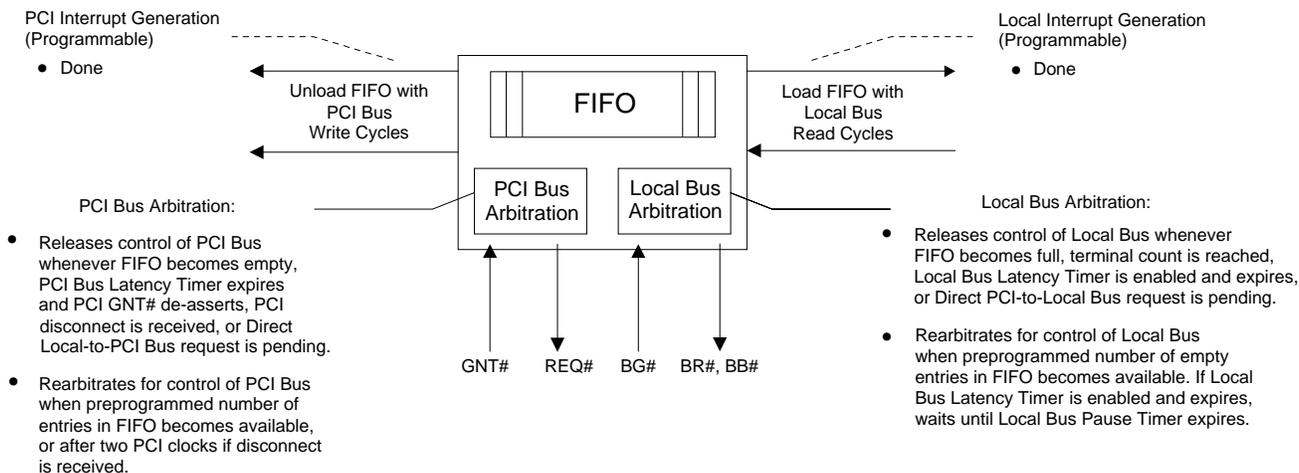


Figure 3-18. Local-to-PCI Bus DMA Data Transfer Operation

### 3.5.7.2 PCI-to-Local Bus DMA Transfer

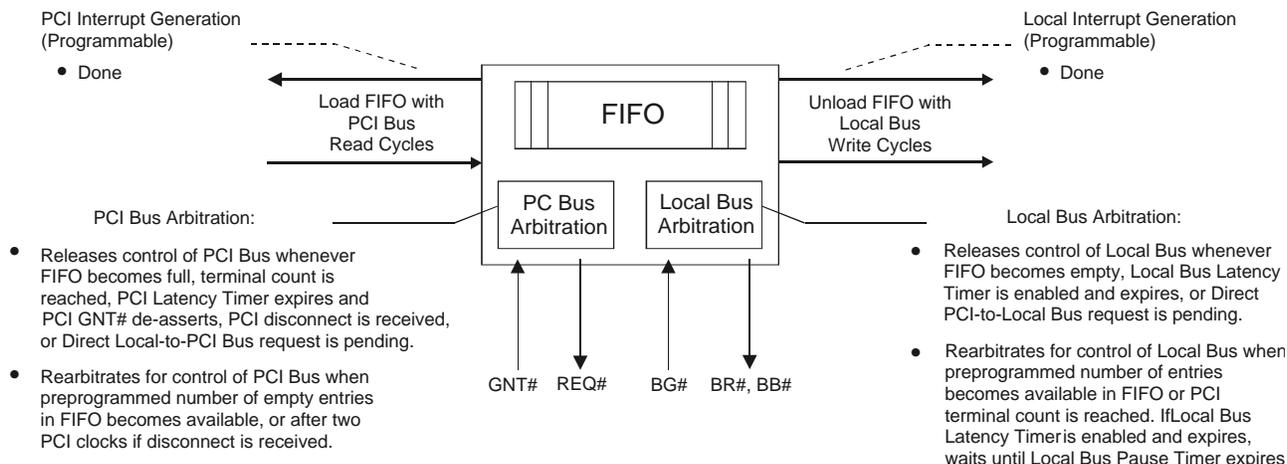


Figure 3-19. PCI-to-Local Bus DMA Data Transfer Operation

Section 3—M Func Desc

### 3.5.7.3 DMA Local Bus Error Condition

The PCI 9054 supports Local Bus error conditions with the TEA# signal. TEA# may be asserted by a device on the Local Bus, either before or simultaneously with TA#. In either case, the PCI 9054 attempts to finish the current transaction by transferring data and then asserting TS# for every address that follows, waiting for another TA# or TEA# to be issued to flush the FIFOs. After sensing TEA# is asserted, the PCI 9054 asserts PCI SERR# and sets the Signaled System Error bit (PCISR[14], indicating a catastrophic error occurred on the Local Bus. SERR# may be masked by resetting the TEA# Input Interrupt Mask bit (LMISC[5]=0).

The PCI 9054 Local Bus Latency Timer (MARBR[7:0]), as well as the Local Bus Pause Timer (MARBR[15:8]), can be used to better utilize the Local Bus.

### 3.5.7.4 DMA Unaligned Transfers

For unaligned Local-to-PCI transfers, the PCI 9054 reads a partial Lword from the Local Bus. It continues to perform a Single-Cycle read (Lwords) from the Local Bus until the nearest 16-byte boundary. If the Burst Mode bit is enabled, the PCI 9054 bursts thereafter. Lwords are assembled, aligned to the PCI Bus address, and loaded into the FIFO until the nearest 16-byte boundary.

For PCI-to-Local transfers, Lwords are read from the PCI Bus and loaded into the FIFO. On the Local Bus, Lwords are assembled from the FIFO, aligned to the Local Bus address and single cycle written to the Local Bus until the nearest 16-byte boundary. If burst functionality is enabled, the PCI 9054 bursts thereafter.

### 3.5.8 Demand Mode DMA, Channel 0

The Fast/Slow Terminate Mode Select bit(s) (DMAMODE0[15] and/or DMAMODE1[15]) determines the number of Lwords to transfer after the DMA controller DREQ0# input is de-asserted.

If BDIP# output is not required to be de-asserted before the last Lword of a DMA transfer (bit [15]=1), the DMA controller releases the data bus after it receives an external TA# or the internal wait state counter decrements to 0 for the current Lword. If the DMA controller is currently bursting data, which is not the last Data phase for the burst, BDIP# is not asserted before the last Lword of the DMA transfer.

If BDIP# output must be de-asserted before the last Lword of the DMA transfer (bit [15]=0), the DMA controller continues transferring data up to the nearest 16-byte boundary. If DREQ0# is de-asserted during the Address phase of the first transfer in PCI 9054 Local Bus ownership (TS#, BG# asserted), the DMA controller completes a 16-byte transfer. If DREQ0# is de-asserted during a Data-Transfer phase, one Lword before the last 16-byte transfer, the PCI 9054 finishes the transfer and performs an additional 16-byte transfer to satisfy BDIP# de-assertion protocol. (Refer to Table 3-7.)

Table 3-7. Demand Mode DMA, Channel 0

BTERM# Input Enable Bit(s)	Fast/Slow Terminate Mode Select Bit(s)	PCI 9054 BDIP# Output
Enabled (1)	Disabled (1)	BDIP# is not asserted. Immediate transfer terminated by EOT#.
Enabled (1)	Enabled (0)	
Disabled (0)	Disabled (1)	
Disabled (0)	Enabled (0)	BDIP# asserted by the PCI 9054. Transfers up to the nearest 16-byte boundary, then terminates (MPC850 or MPC860 compatible).

### 3.5.9 End of Transfer (EOT#) Input

The DMA EOT# Enable bit(s) (DMAMODE0[14] and DMAMODE1[14]) determines the number of Lwords to transfer after a DMA controller asserts EOT# input. EOT# input should be asserted only when the PCI 9054 owns a bus. (Refer to Table 3-8.)

If BDIP# output is not required to be de-asserted before the last Lword of the DMA transfer (DMAMODE0[15]=1 and/or DMAMODE1[15]=1), and the DMA EOT# Enable bit(s) is set (DMAMODE0[14]=1 and/or DMAMODE1[14]=1), the DMA controller releases the data bus and terminates DMA after receiving an external TA# signal. Or, the internal wait state counter decrements to 0 for the current Lword when EOT# is asserted.

If BDIP# output must be de-asserted before the last Lword of the DMA transfer (DMAMODE0[15]=0 and/or DMAMODE1[15]=0), the DMA controller transfers data up to the nearest 16-byte boundary if EOT# is asserted and enabled (DMAMODE0[14]=1 and/or DMAMODE1[14]=1).

When the BTERM# Enable bit is disabled, Fast/Slow Terminate is enabled, and EOT# is asserted during the Data-Transfer phase of the last four bytes of a 16-byte transfer, the PCI 9054 completes the transfer and performs an additional 16-byte transfer to satisfy the BDIP# de-assertion protocol. Otherwise, it completes the current 16-byte transfer.

When the BTERM# Enable bit is enabled, or the BTERM# Enable bit is disabled and Fast/Slow Terminate is disabled, the DMA controller terminates a transfer on an Lword boundary after EOT# is asserted. For an 8-bit bus, the PCI 9054 terminates after transferring the last byte for the Lword. For a 16-bit bus, the PCI 9054 terminates after transferring the last word for the Lword. In Single-Cycle mode (burst disabled), the transfer is terminated at the next Lword boundary after EOT# occurs. The exception to this is when EOT# occurs on the last four bytes of the Transfer Count setting.

During the descriptor loading on the Local Bus, assertion of EOT# causes a complete descriptor load and no subsequent Data transfer; however, this is not recommended. This has no effect when the descriptor is loaded from the PCI Bus.

**Table 3-8. Any DMA Transfer Channel 0/1 with EOT Functionality**

BTERM# Enable Bit(s)	Fast/Slow Terminate Mode Select Bit(s)	PCI 9054 BDIP# Output
Enabled (1)	Disabled (1)	BDIP# is not asserted. Transfer is immediately terminated by EOT# or paused by DREQ# at Lword boundary.
Enabled (1)	Enabled (0)	
Disabled (0)	Disabled (1)	
Disabled (0)	Enabled (0)	BDIP# asserted by the PCI 9054. Transfers up to the nearest 16-byte boundary, then terminates (MPC850 or MPC860 compatible).

### 3.5.10 DMA Arbitration

The PCI 9054 asserts BR# when it needs to be the Local Bus Master. Upon receiving BG#, the PCI 9054 waits for BB# to be de-asserted. The PCI 9054 then asserts BB# at the next rising edge of the Local clock after sensing that BB# is de-asserted (no other device is acting as Local Bus Master). The PCI 9054 continues to assert BB# while acting as the Local Bus Master (*that is*, it holds the bus until instructed to release BB#) under the following conditions:

- Local Bus Latency Timer is enabled and expires (MARBR[7:0])
- PCI Target access is pending
- EOT# input is received (if enabled)

The DMA controller releases control of the PCI Bus when one of the following conditions occurs:

- FIFOs are full or empty
- PCI Bus Latency Timer expires (PCILTR[7:0])—and loses the PCI GNT# signal
- Target disconnect response is received

The DMA controller de-asserts its PCI Bus request (REQ#) for a minimum of two PCI clocks.

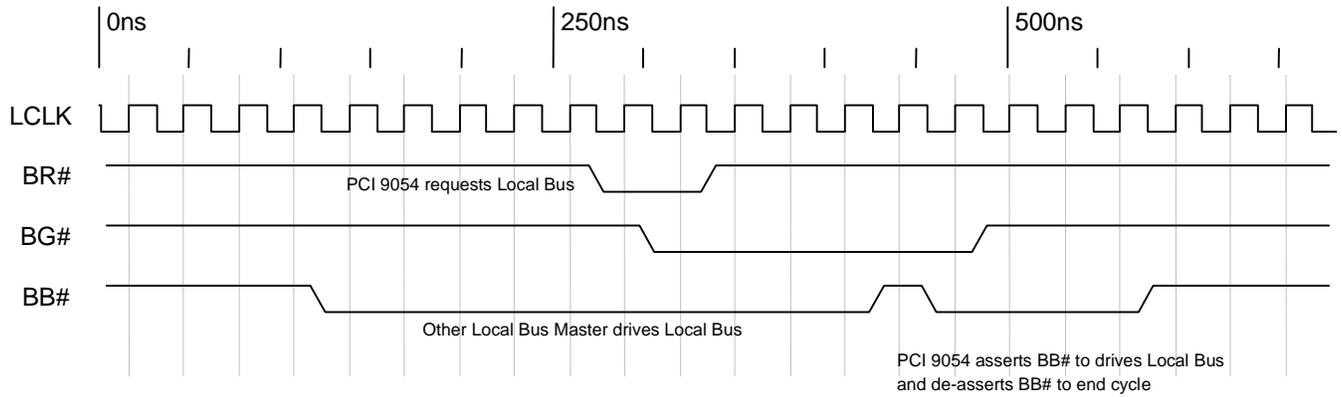
### 3.5.11 Local Bus Latency and Pause Timers

The Local Bus Latency and Pause Timers are programmable with the Mode/DMA Arbitration register (MARBR[7:0, 15:8]). If the Local Bus Latency Timer is enabled and expires (MARBR[7:0]), the PCI 9054 completes an Lword transfer up to the nearest 16-byte boundary and releases the Local Bus, de-asserting BB#. After the programmable Pause Timer expires (MARBR[15:8]), it arbitrates for the bus by asserting BR#. When it receives BG#, it asserts BB# and continues to transfer until the FIFO is empty for a Local-to-PCI transfer or full for a PCI-to-Local transfer.

The DMA transfer can be paused by writing a 0 to the Channel Enable bit. To acknowledge the disable, the PCI 9054 gets at least one data from the bus before it stops. However, this is not recommended during a burst.

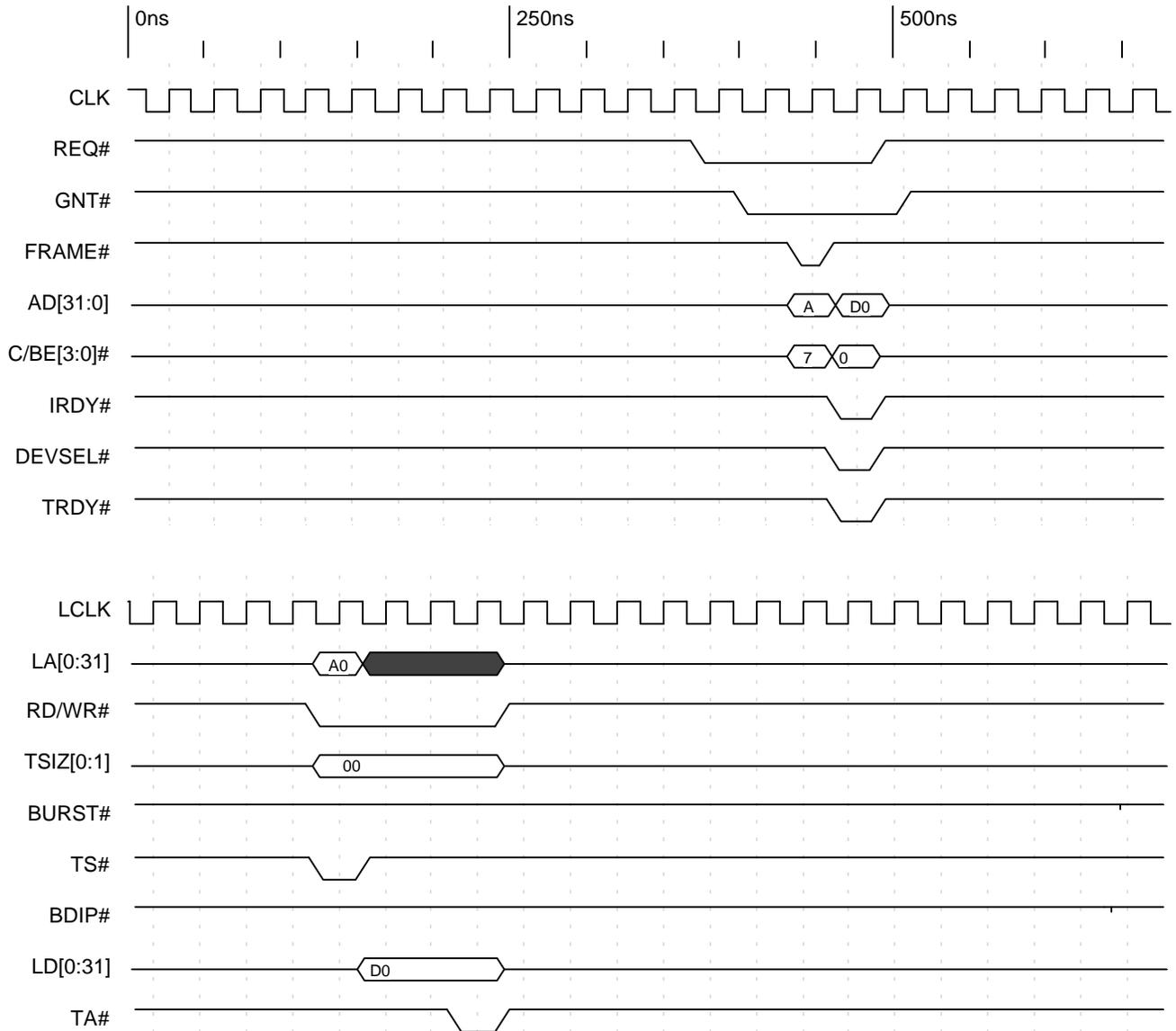
The DMA Local Bus Timer starts after the Local Bus is granted to the PCI 9054 and the Local Pause Timer starts after BB# is de-asserted.

### 3.6 M MODE TIMING DIAGRAMS



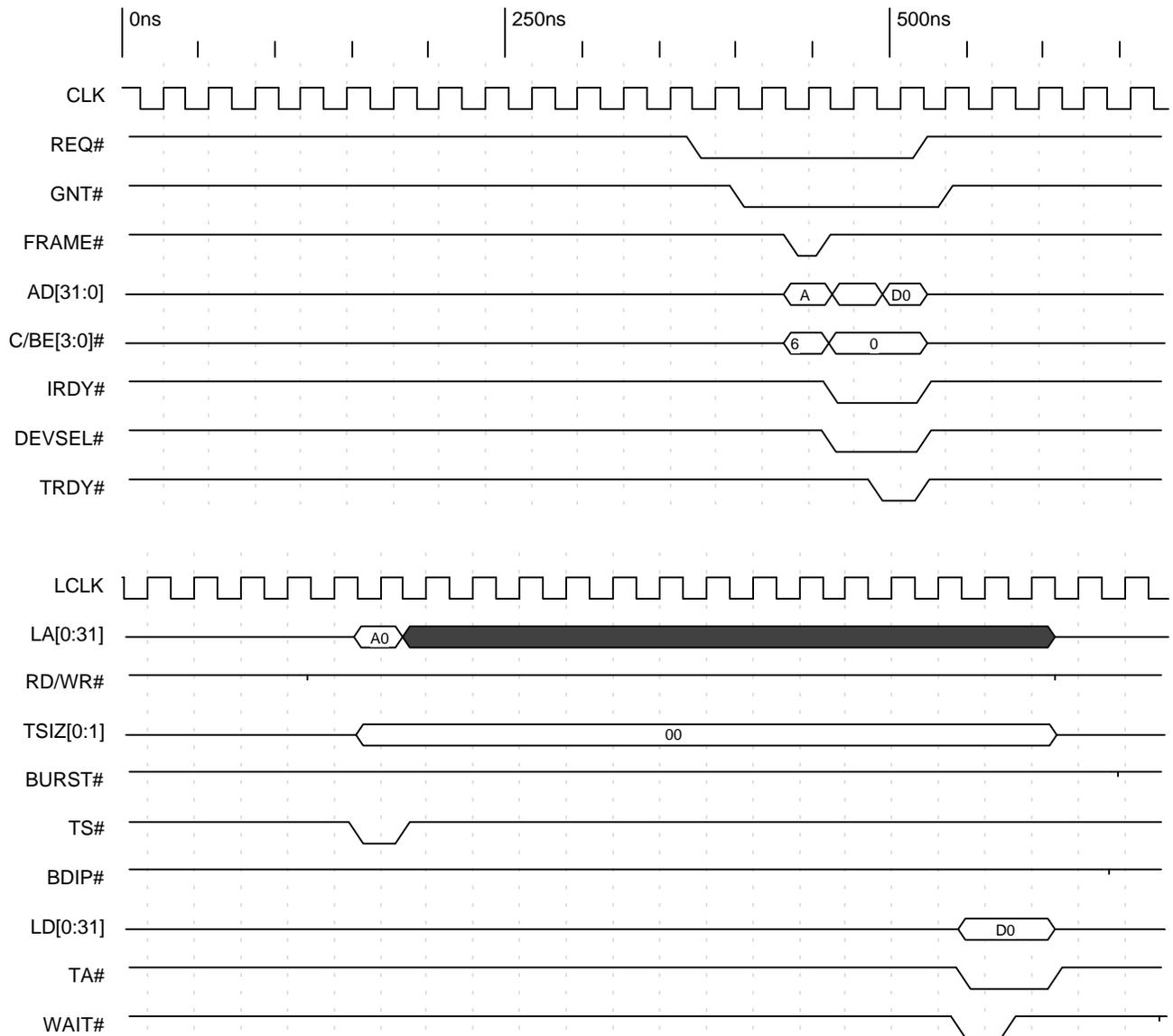
Timing Diagram 3-1. Local Bus Arbitration (BR#, BG#, BB#, and so forth)

3.6.1 M Mode PCI Initiator

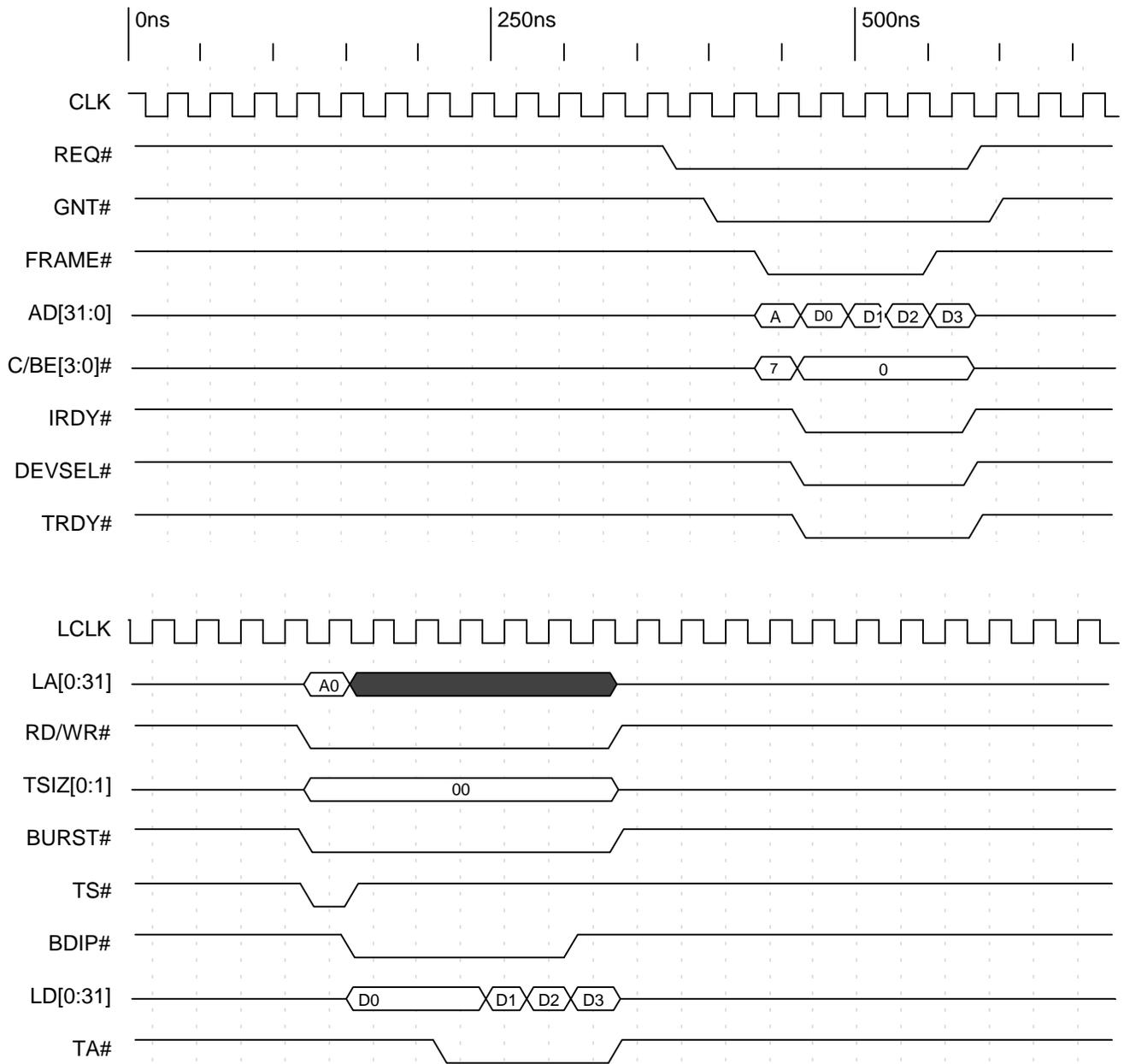


Timing Diagram 3-2. PCI Initiator Single Write Cycle, Zero Wait States

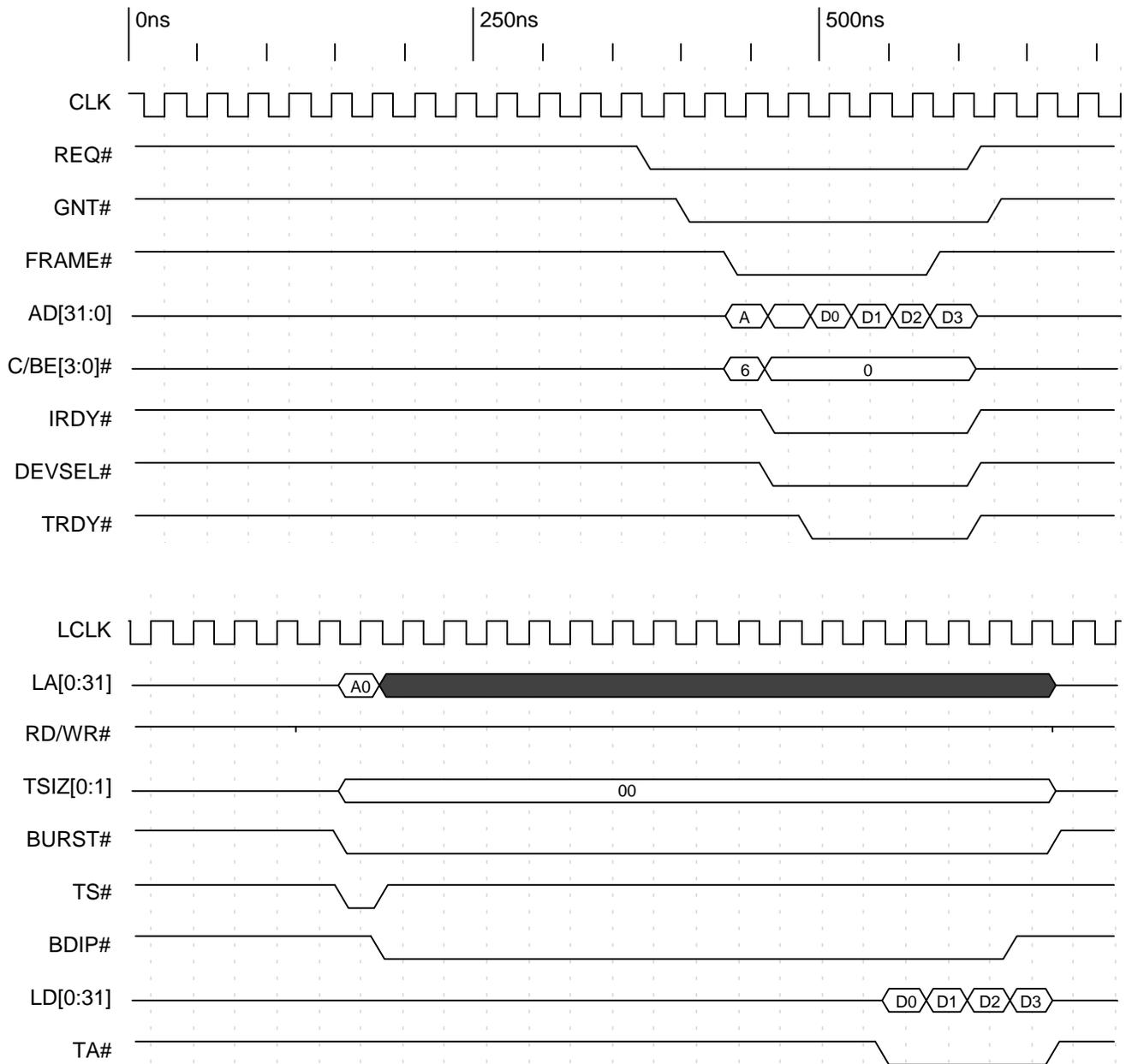
Section 3—M Func Desc



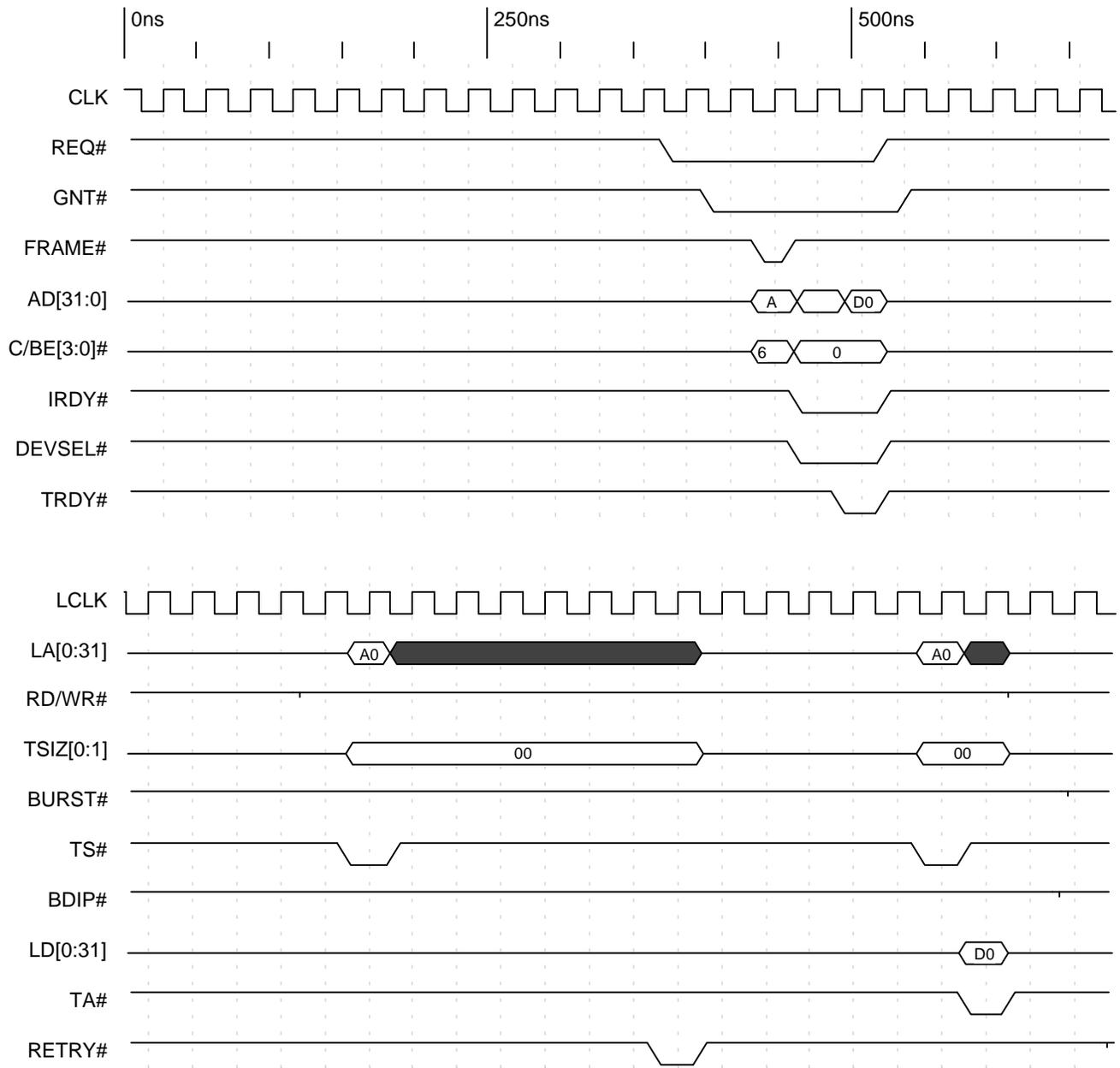
Timing Diagram 3-3. PCI Initiator Single Read Cycle, One Wait State (WAIT# Asserted for One Clock)



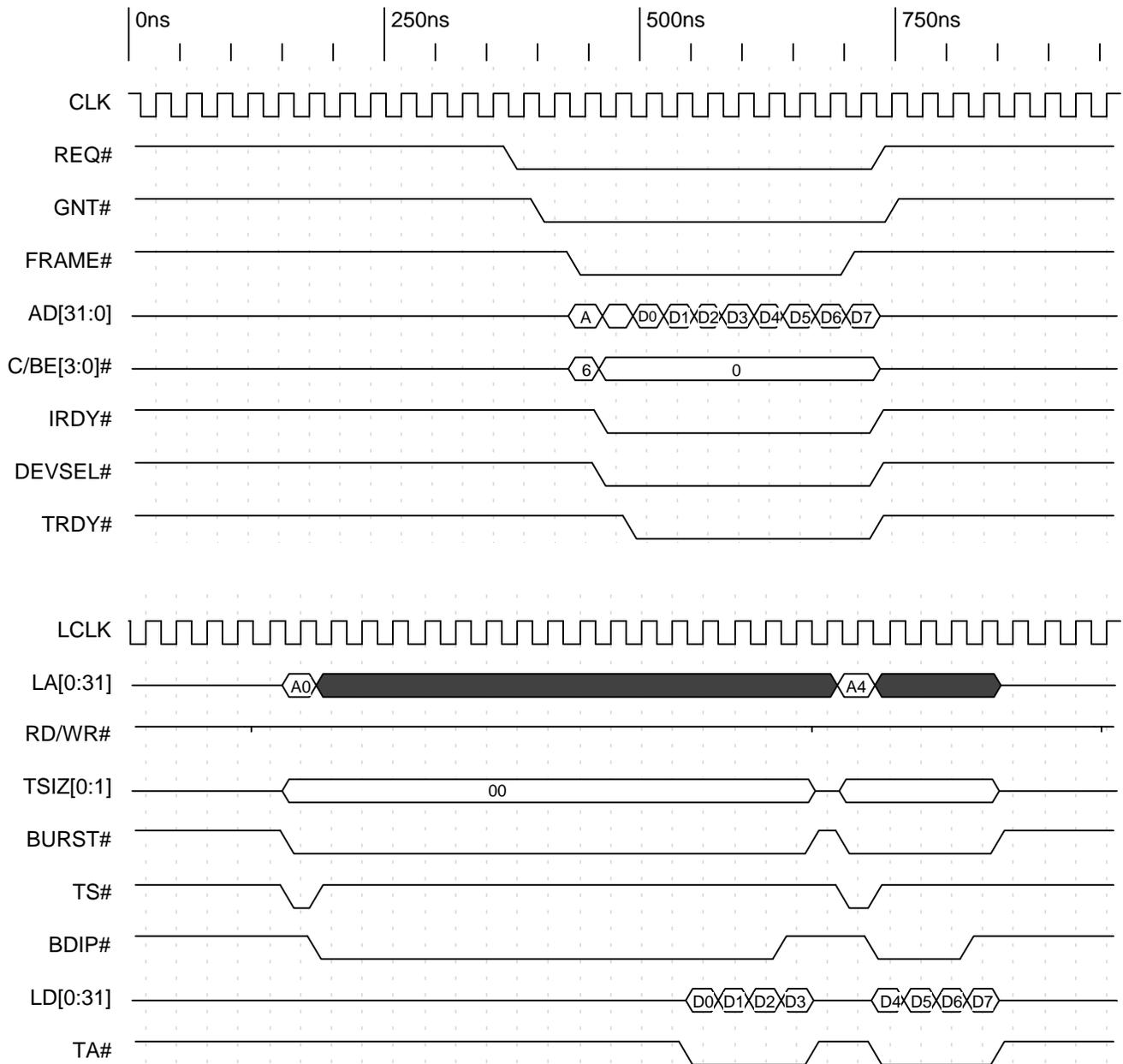
Timing Diagram 3-4. PCI Initiator Burst Write Cycle of Four Lwords, Zero Wait States



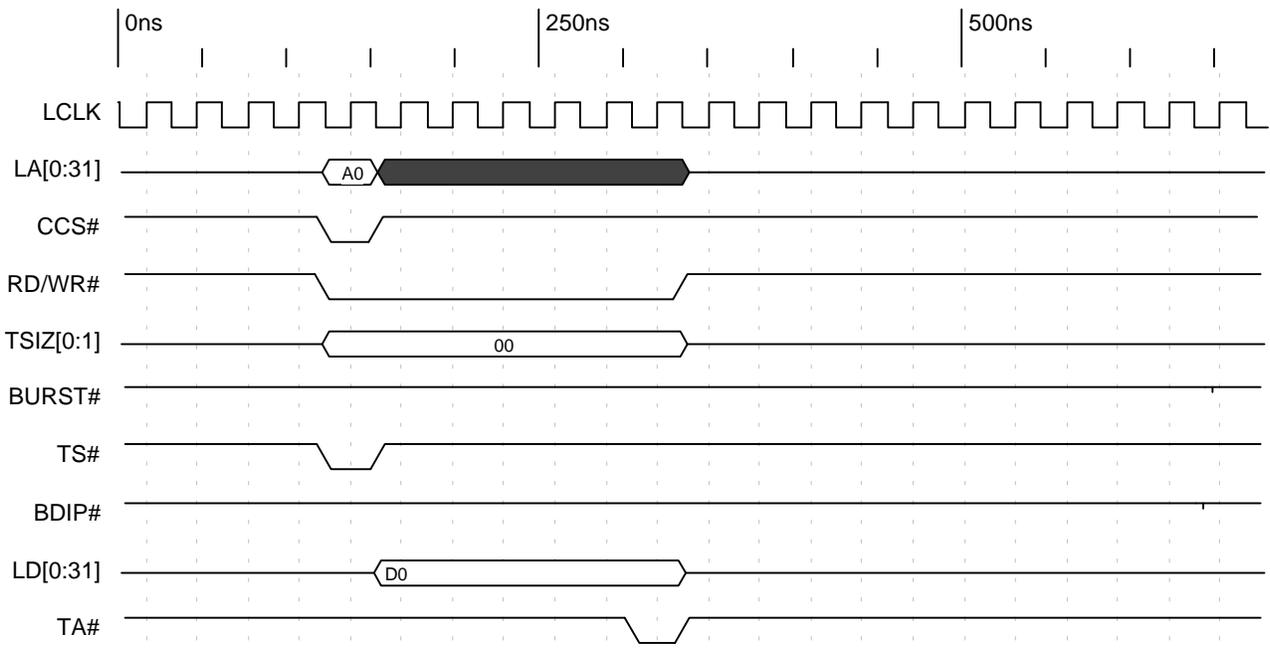
Timing Diagram 3-5. PCI Initiator Burst Read Cycle of Four Lwords, Zero Wait States



Timing Diagram 3-6. PCI Initiator Deferred Read Mode (RETRY#)

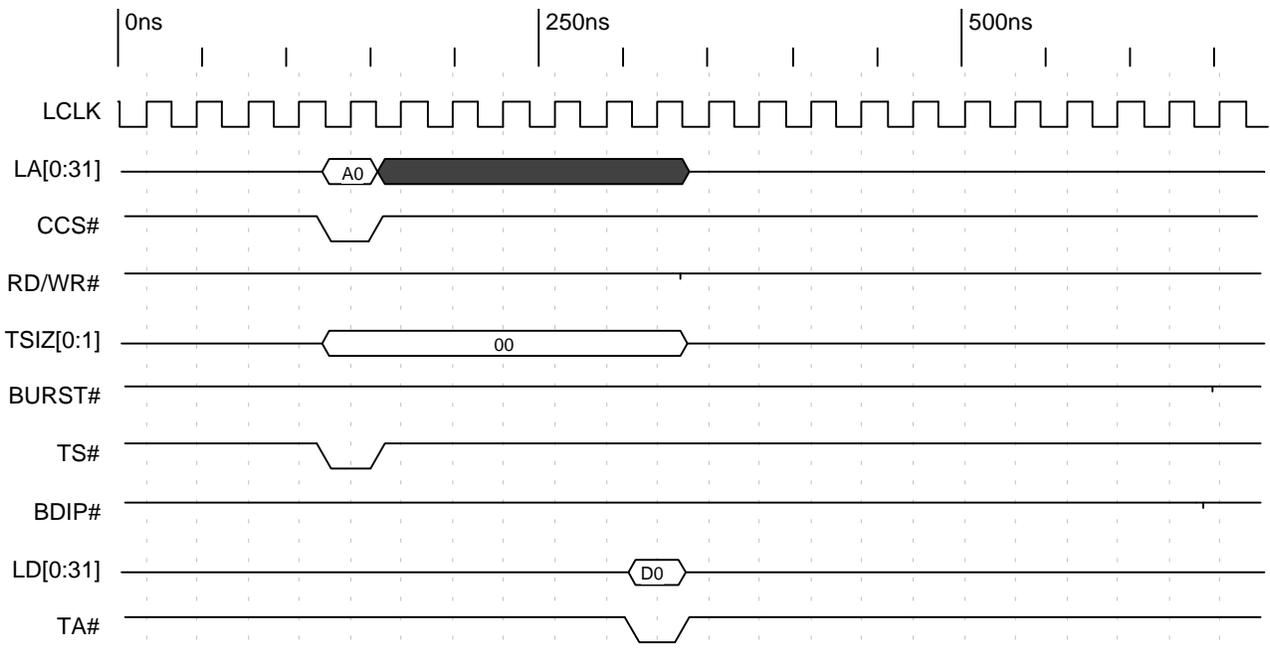


Timing Diagram 3-7. PCI Initiator Burst Read with Read Ahead Mode (Prefetch Counter Set to Eight Lwords)



It takes a minimum of five clocks for the PCI 9054 to assert TA#

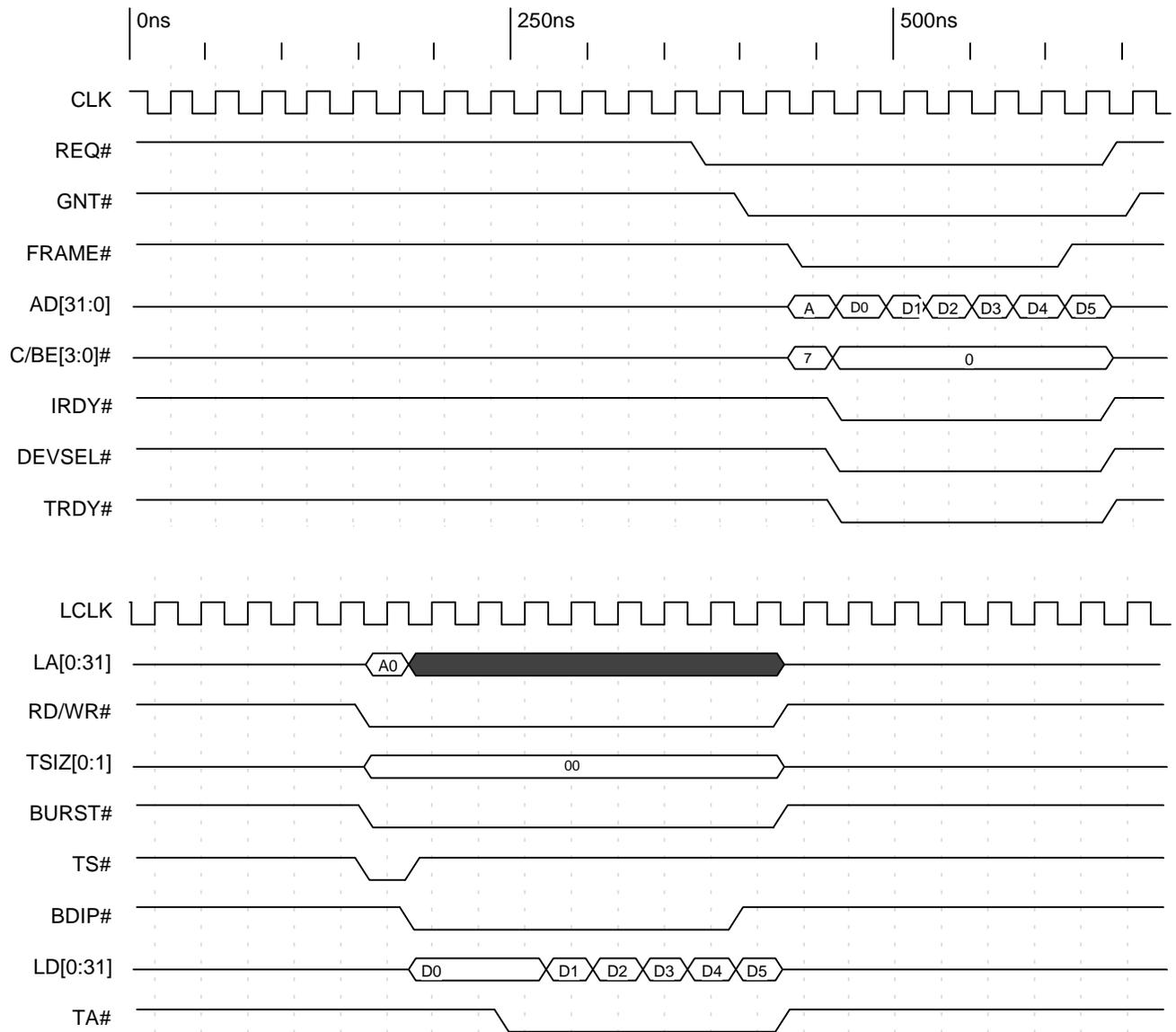
Timing Diagram 3-8. Local Configuration Write to Configuration Register



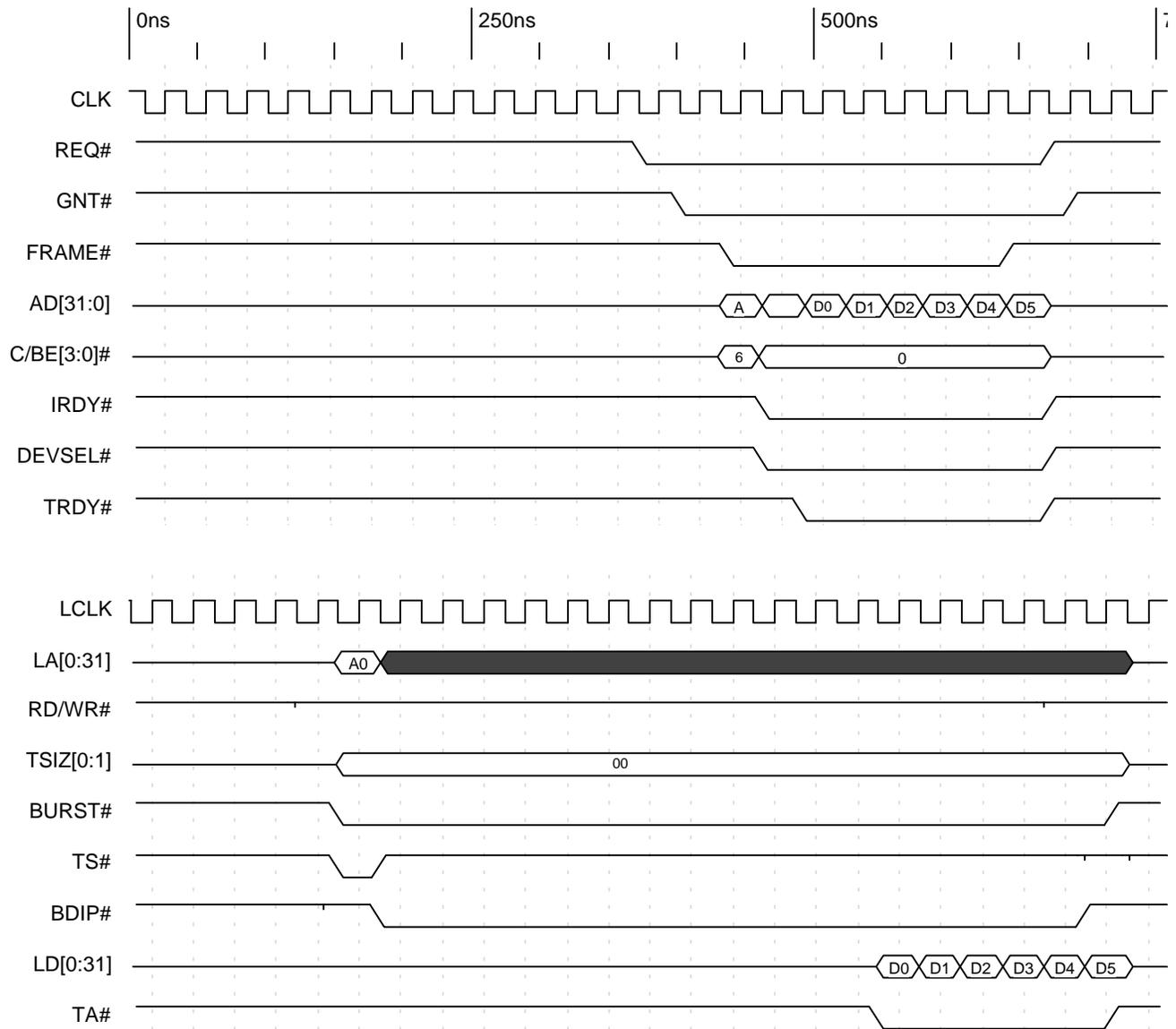
It takes a minimum of five clocks for the PCI 9054 to assert TA#

Timing Diagram 3-9. Local Configuration Read from Configuration Register

Section 3—M Func Desc

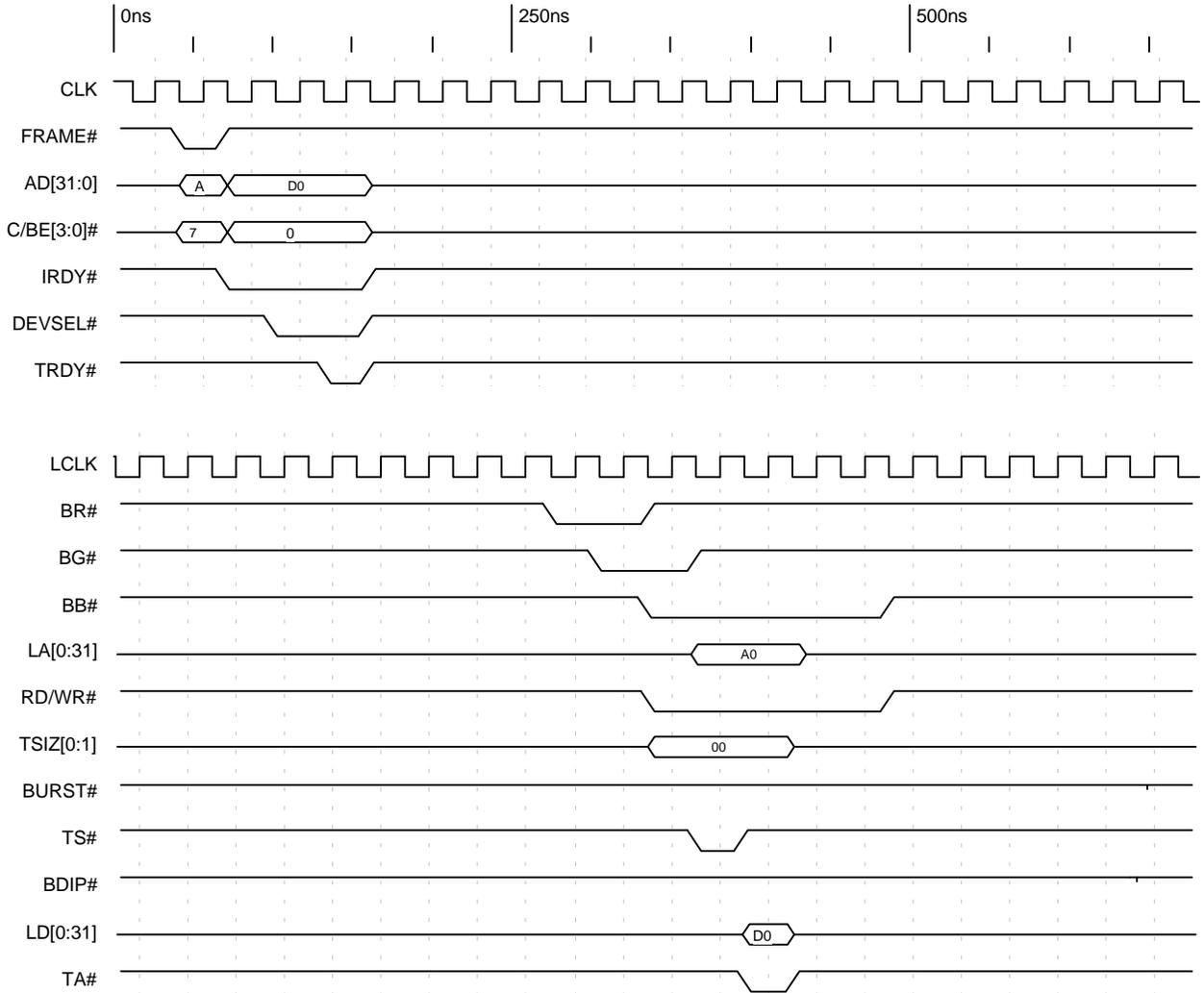


Timing Diagram 3-10. PCI Initiator Burst Write of Six Lwords beyond MPC860 Protocol

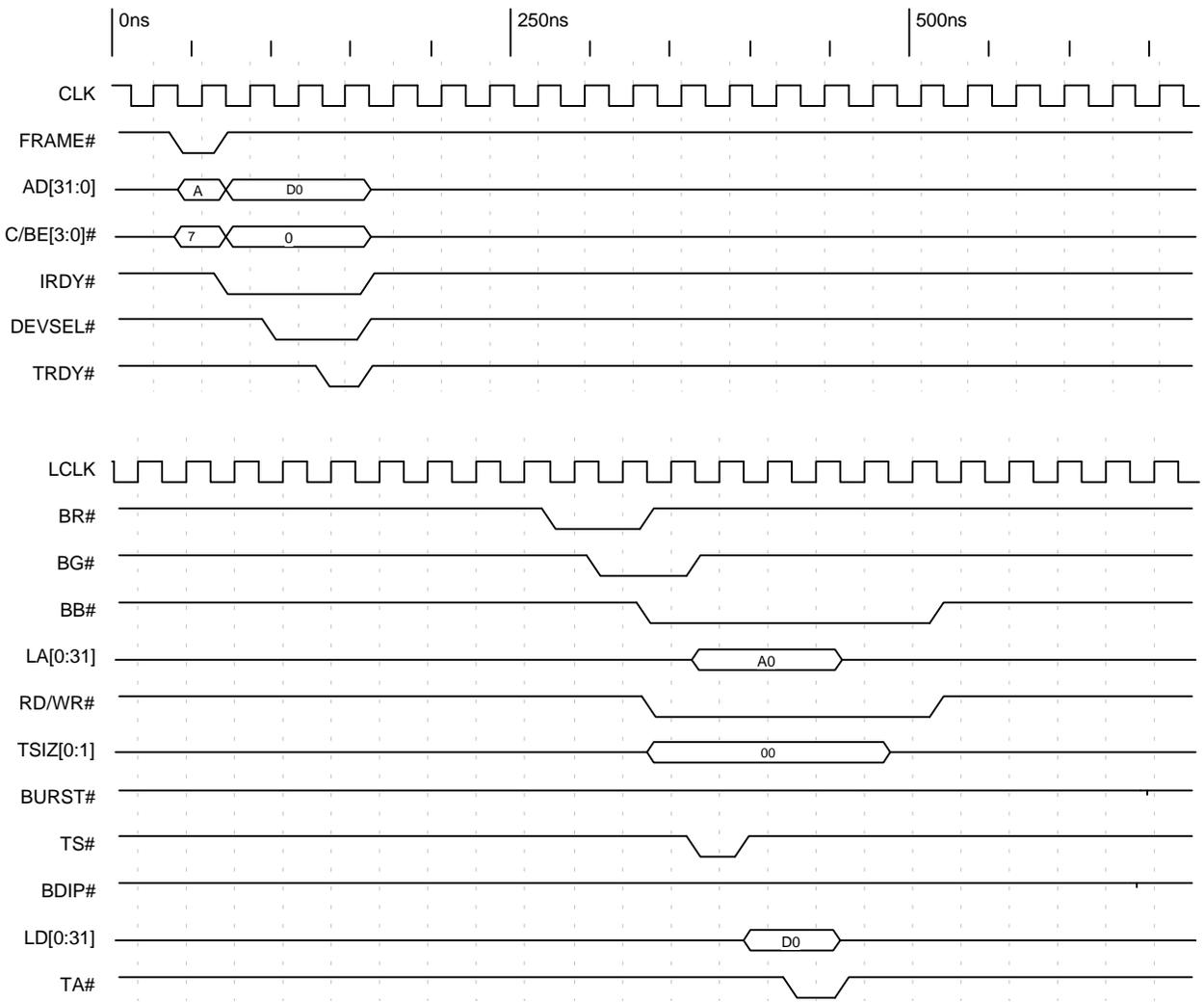


Timing Diagram 3-11. PCI Initiator Burst Read of Six Lwords beyond MPC860 Protocol

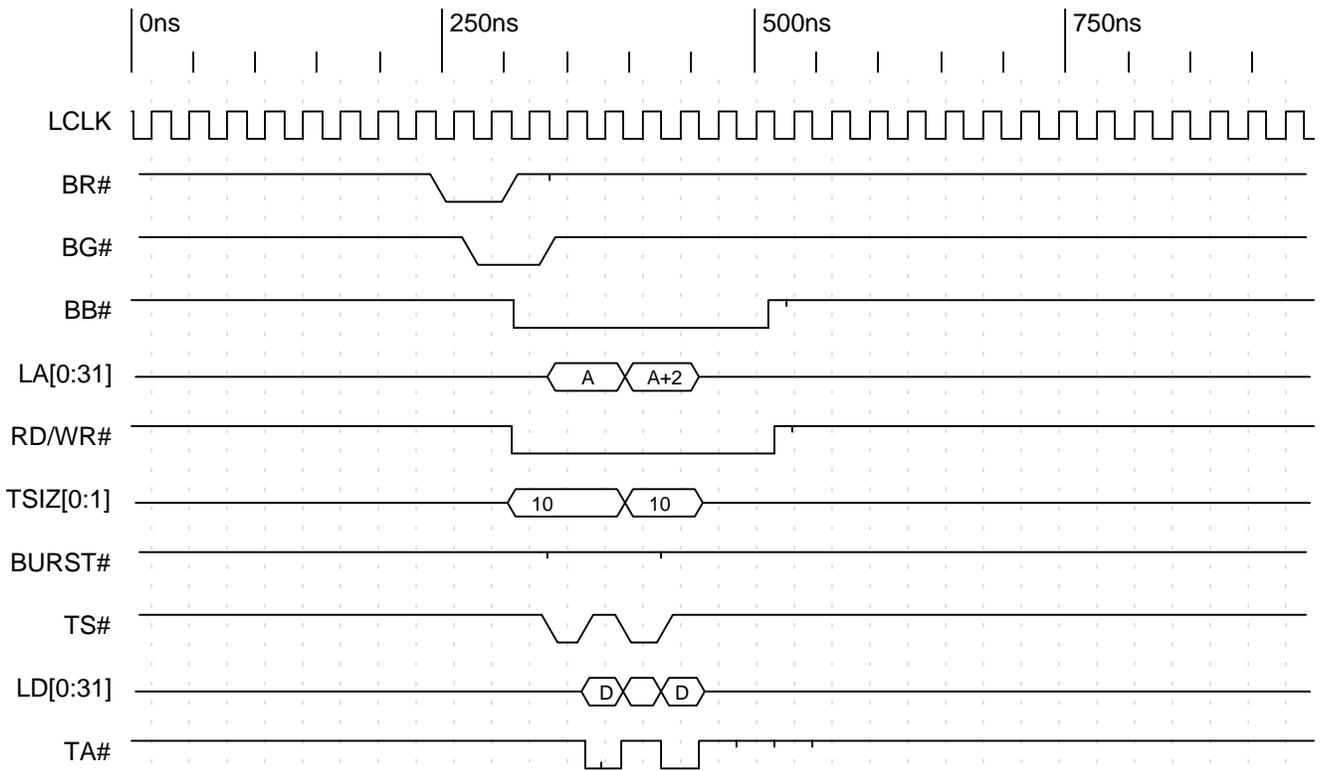
3.6.2 M Mode PCI Target



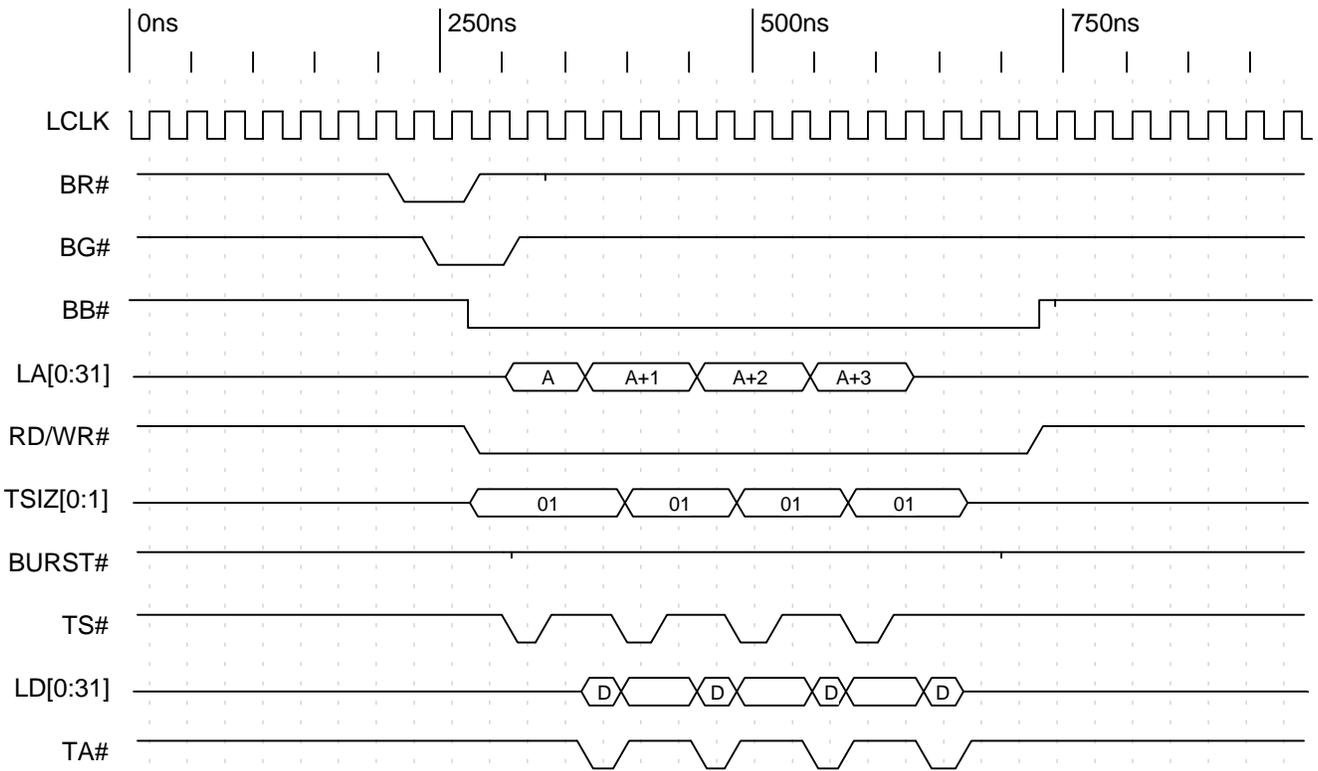
Timing Diagram 3-12. PCI Target Single Write Cycle, Zero Wait States



Timing Diagram 3-13. PCI Target Single Write Cycle, One Wait State by Delaying TA#

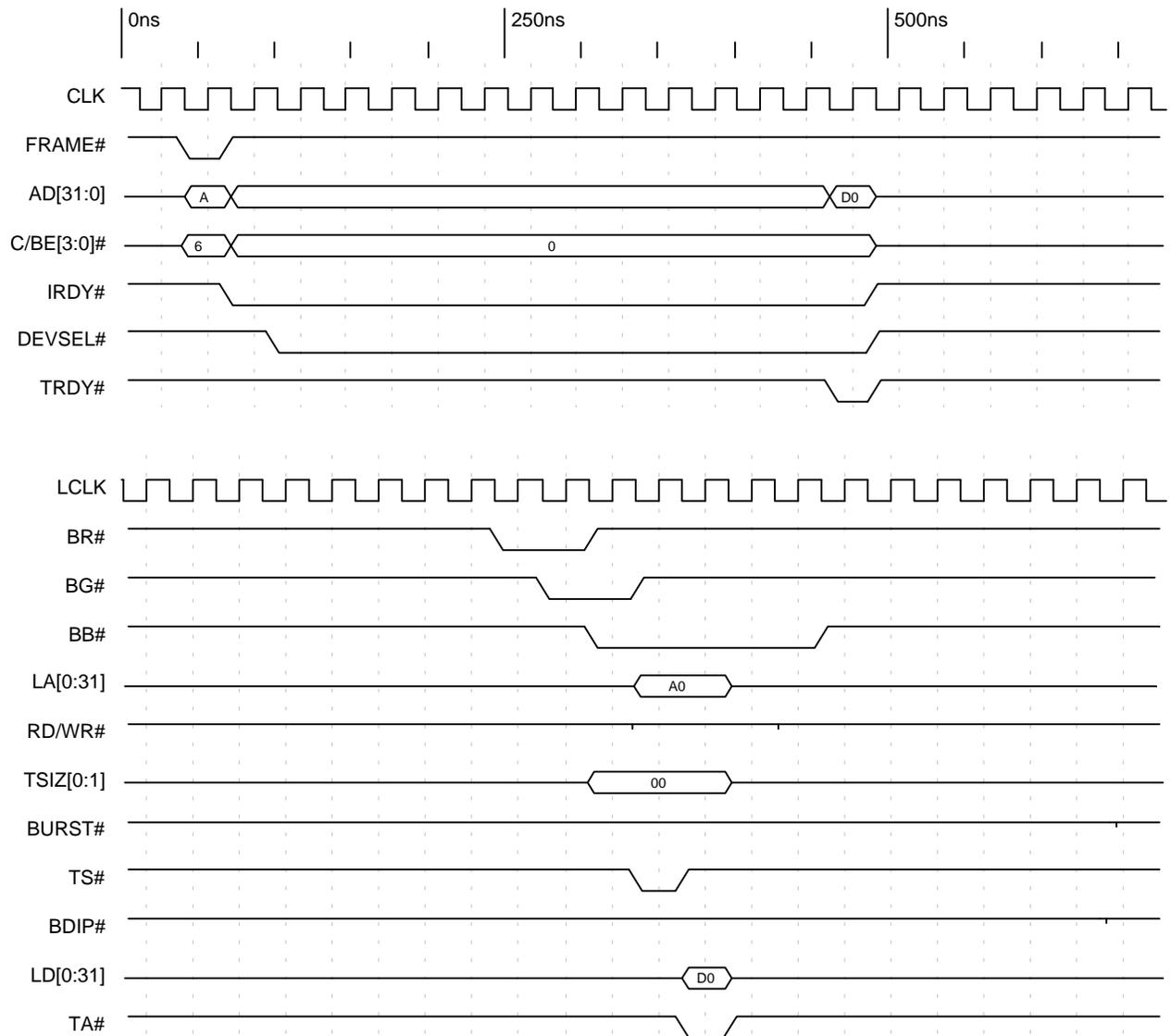


Timing Diagram 3-14. Local Bus Single Write Cycle, Zero Wait States, Burst Enabled, 16-Bit Local Bus

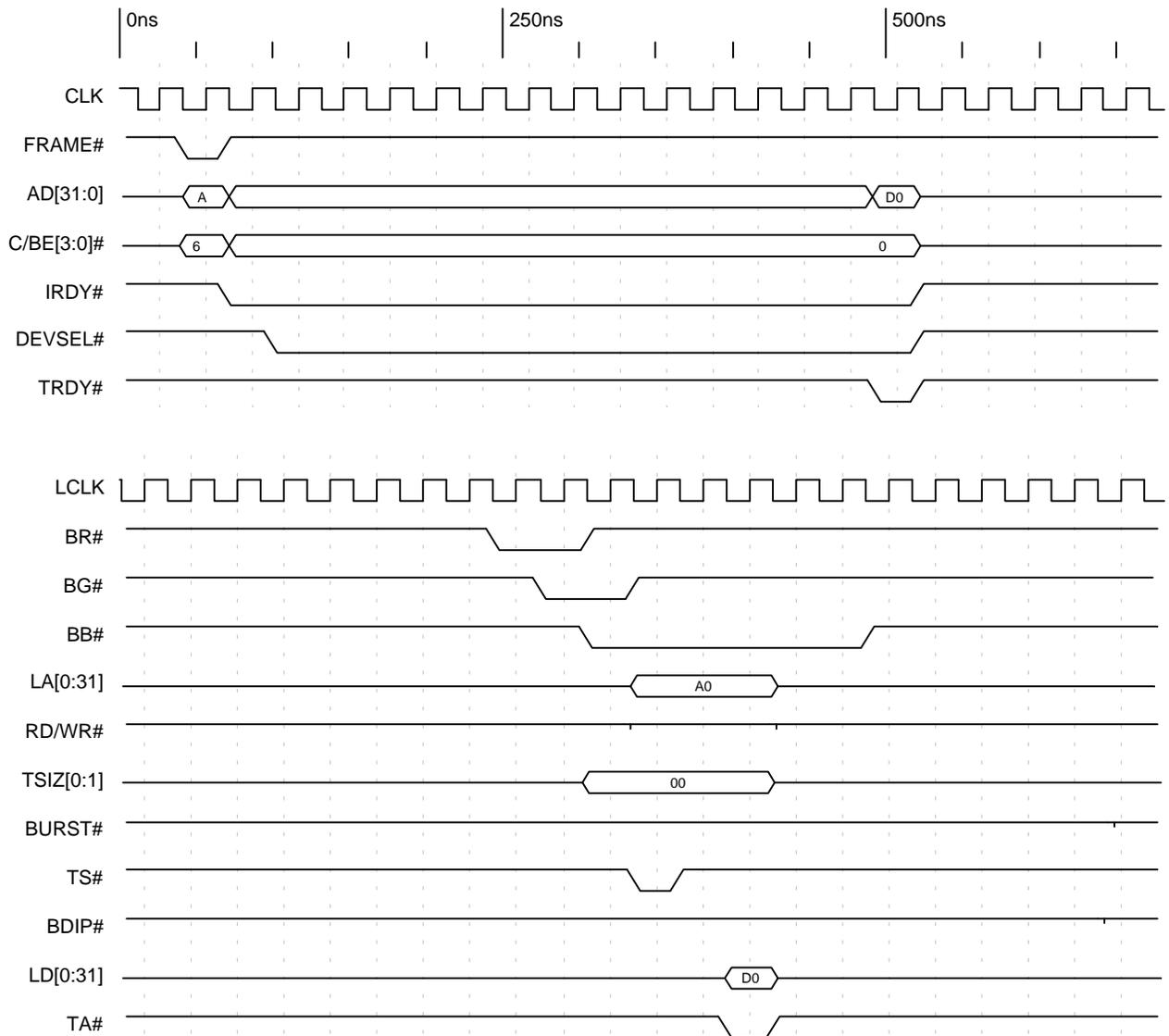


Timing Diagram 3-15. Local Bus Single Write Cycle, One Wait State, Burst Disabled, 8-Bit Local Bus

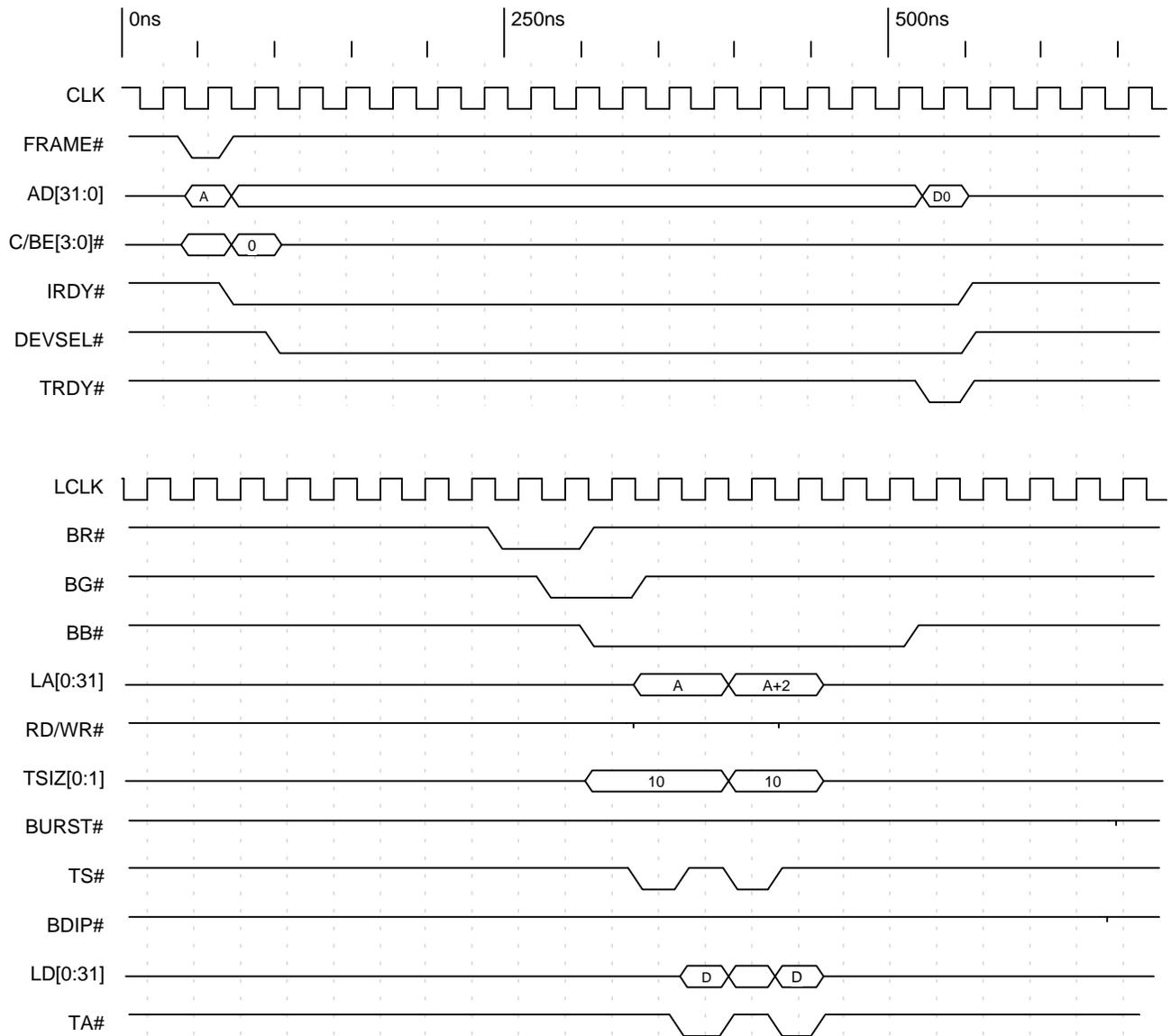
Section 3—M Func Desc



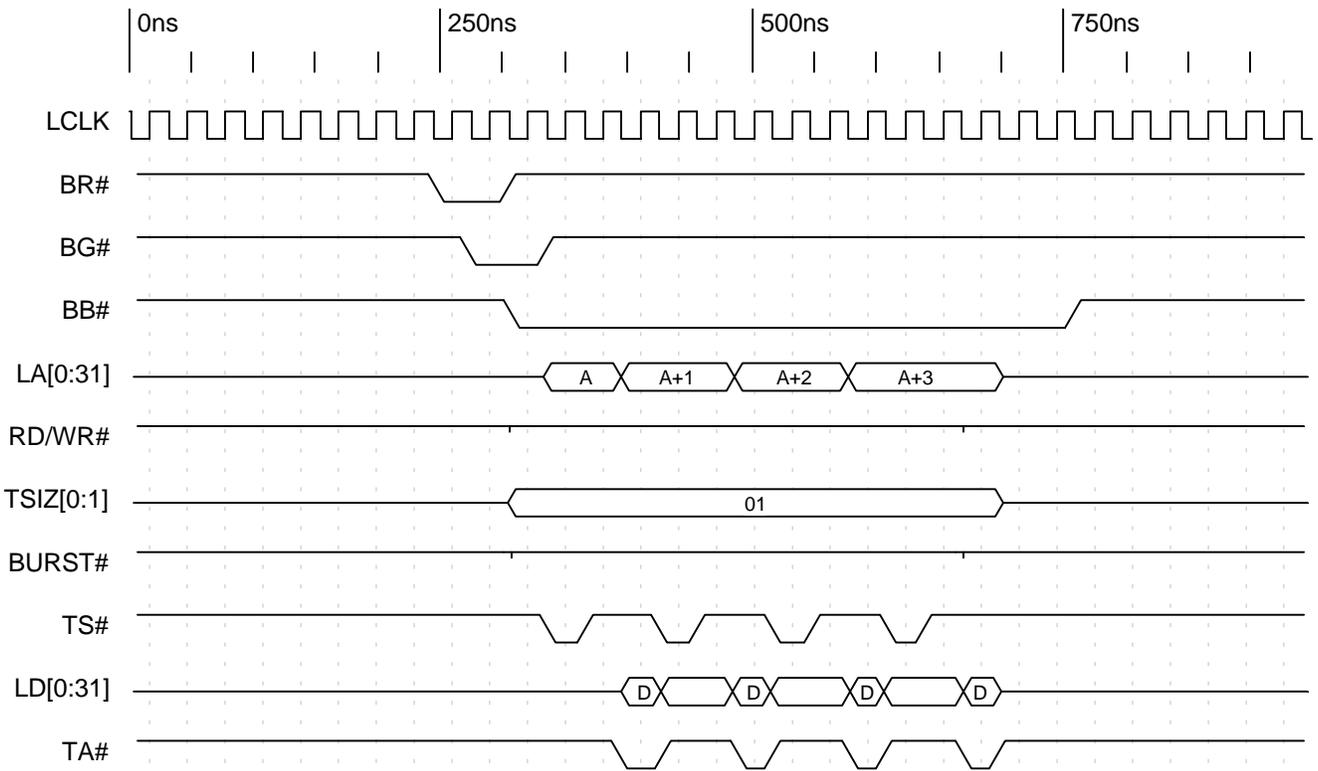
Timing Diagram 3-16. PCI Target Single Read Cycle, Zero Wait States



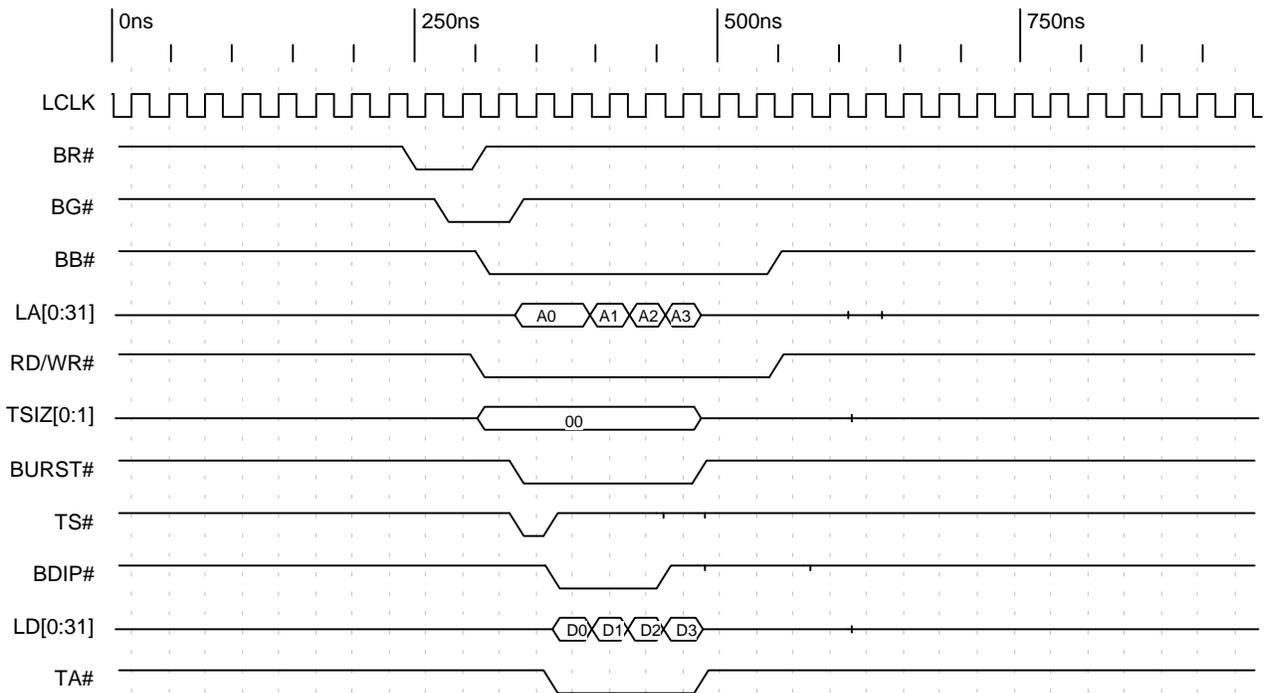
Timing Diagram 3-17. PCI Target Single Read Cycle, One Wait State Using TA#



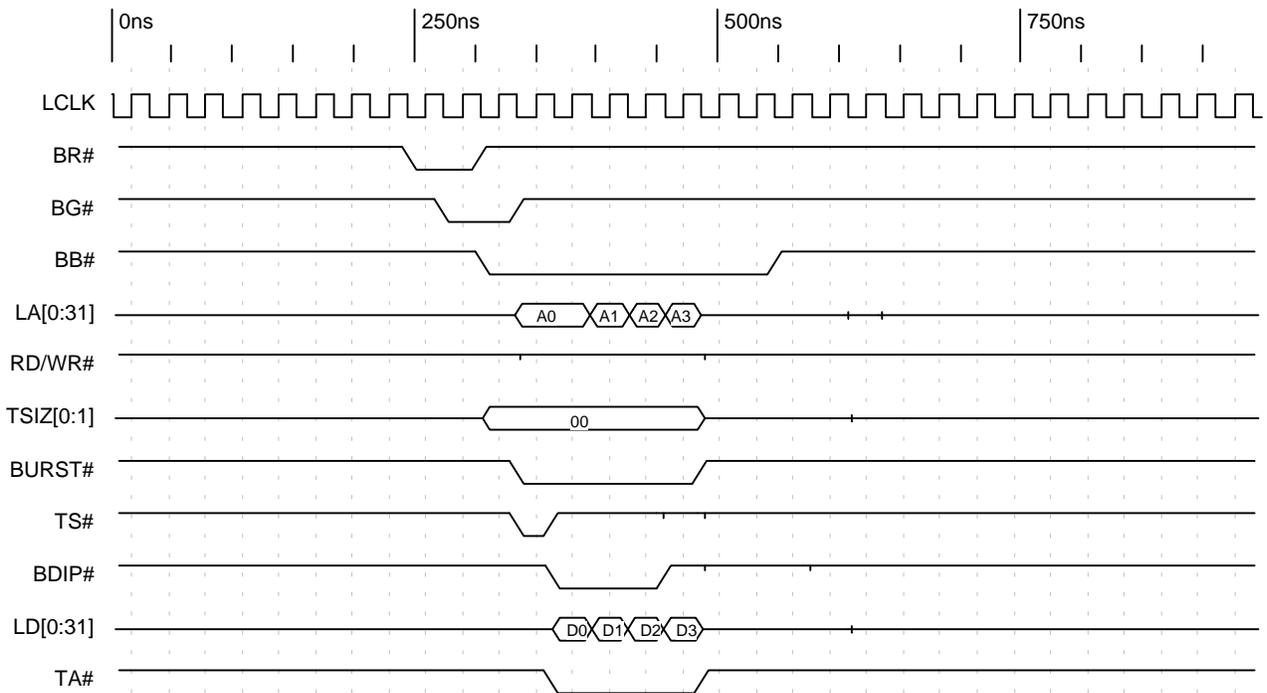
Timing Diagram 3-18. PCI Target Single Read Cycle, Zero Wait States, 16-Bit Bus



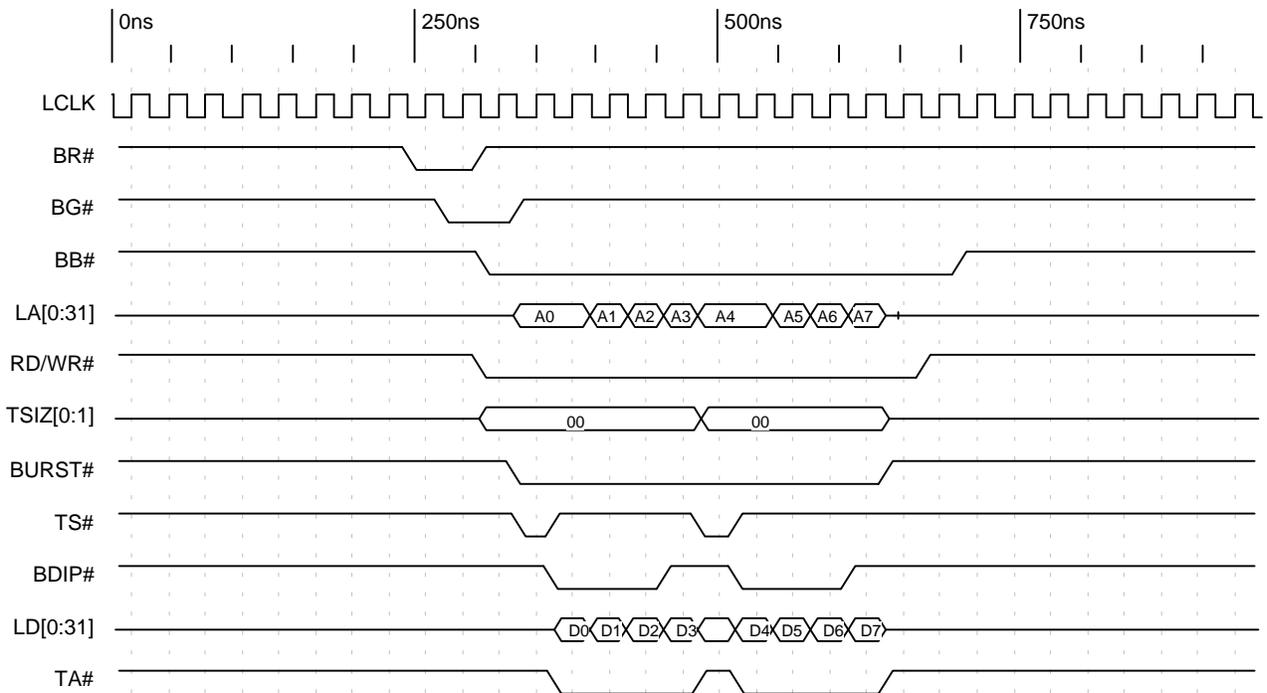
Timing Diagram 3-19. PCI Target Single Read Cycle, One Wait State, Burst Disabled, 8-Bit Local Bus



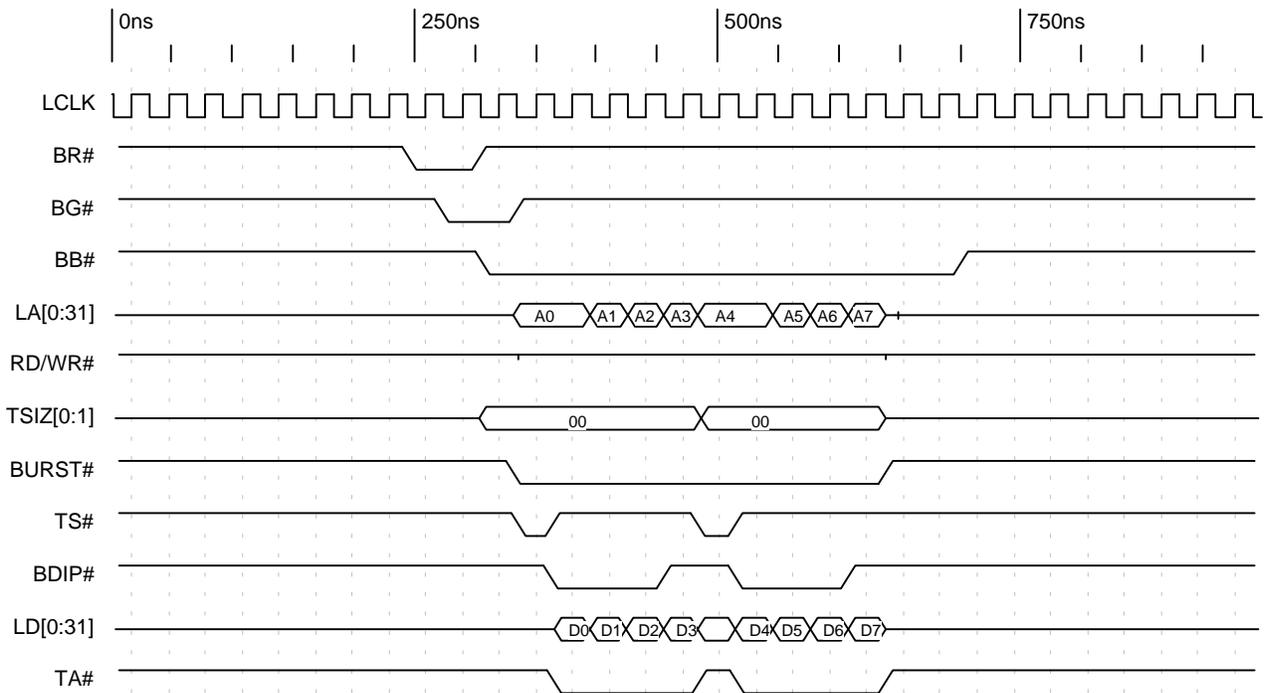
Timing Diagram 3-20. PCI Target Burst Write Cycle of Four Lwords, Bterm Disabled, Burst Enabled



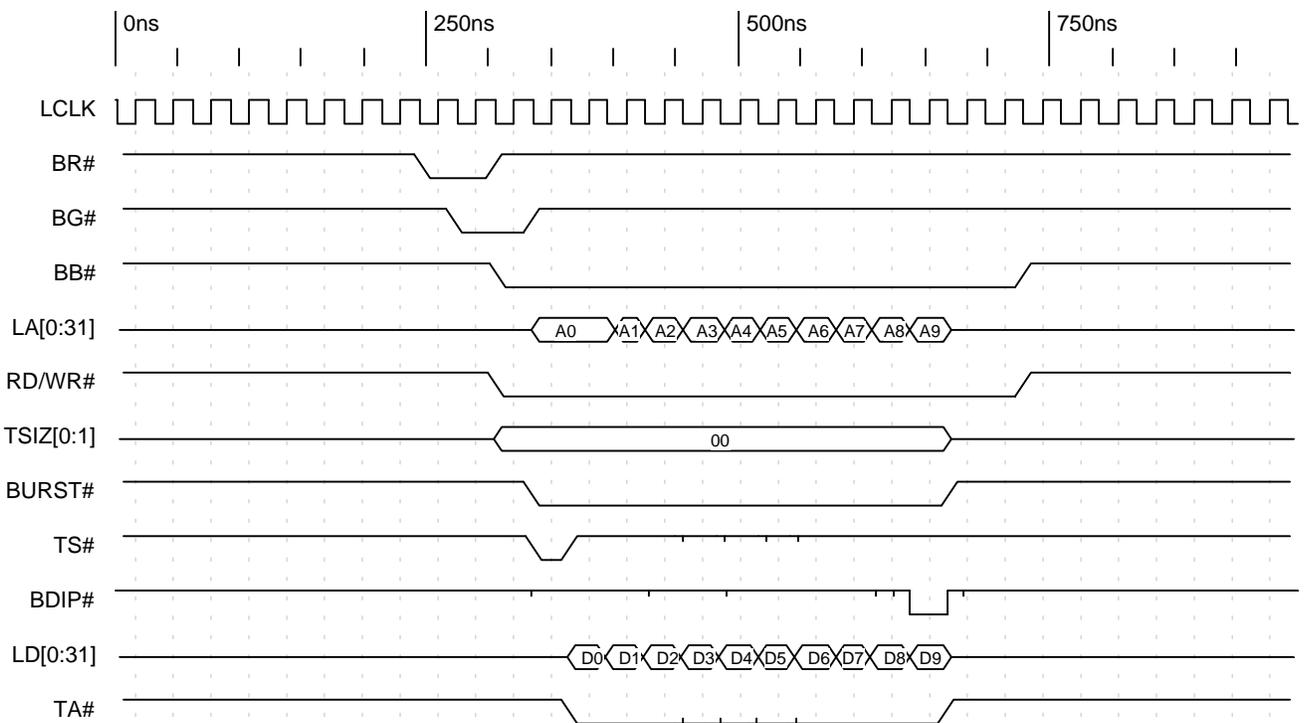
Timing Diagram 3-21. PCI Target Burst Read Cycle of Four Lwords, Bterm Disabled, Burst Enabled



Timing Diagram 3-22. PCI Target Burst Write Cycle of Eight Lwords, Bterm Disabled, Burst Enabled

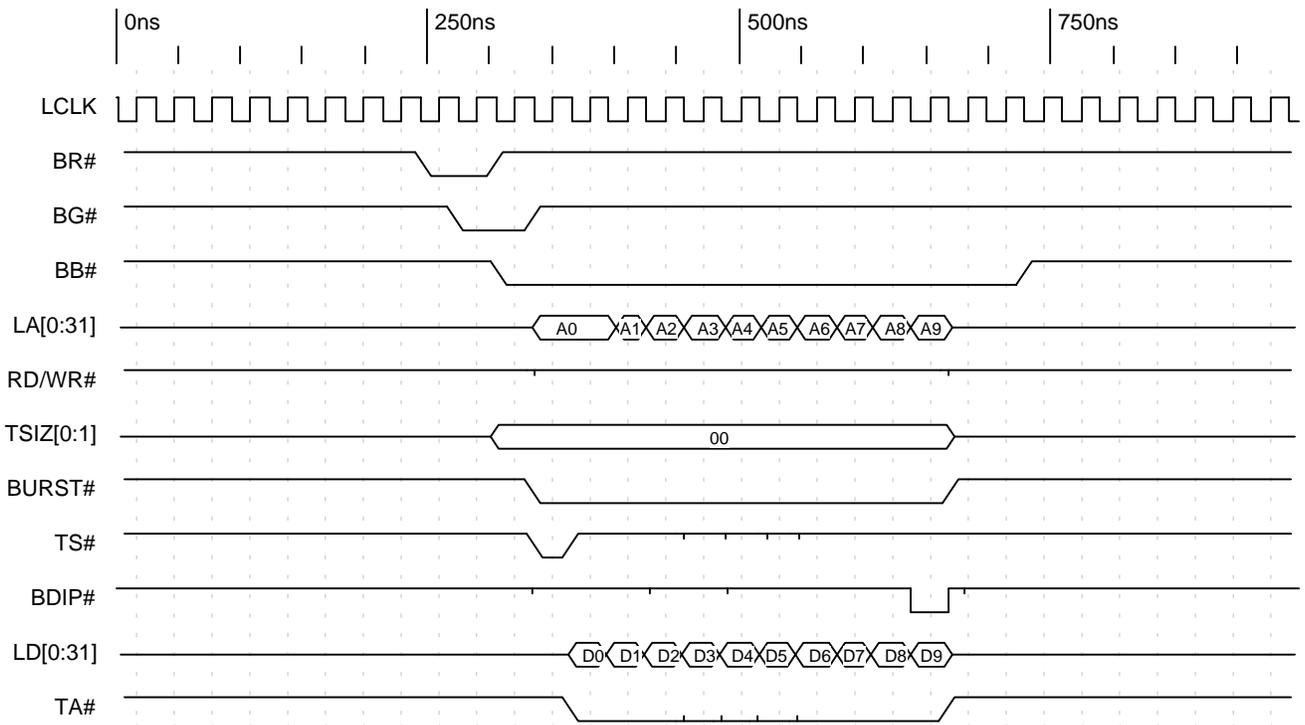


Timing Diagram 3-23. PCI Target Burst Read Cycle of Eight Lwords, Bterm Disabled, Burst Enabled

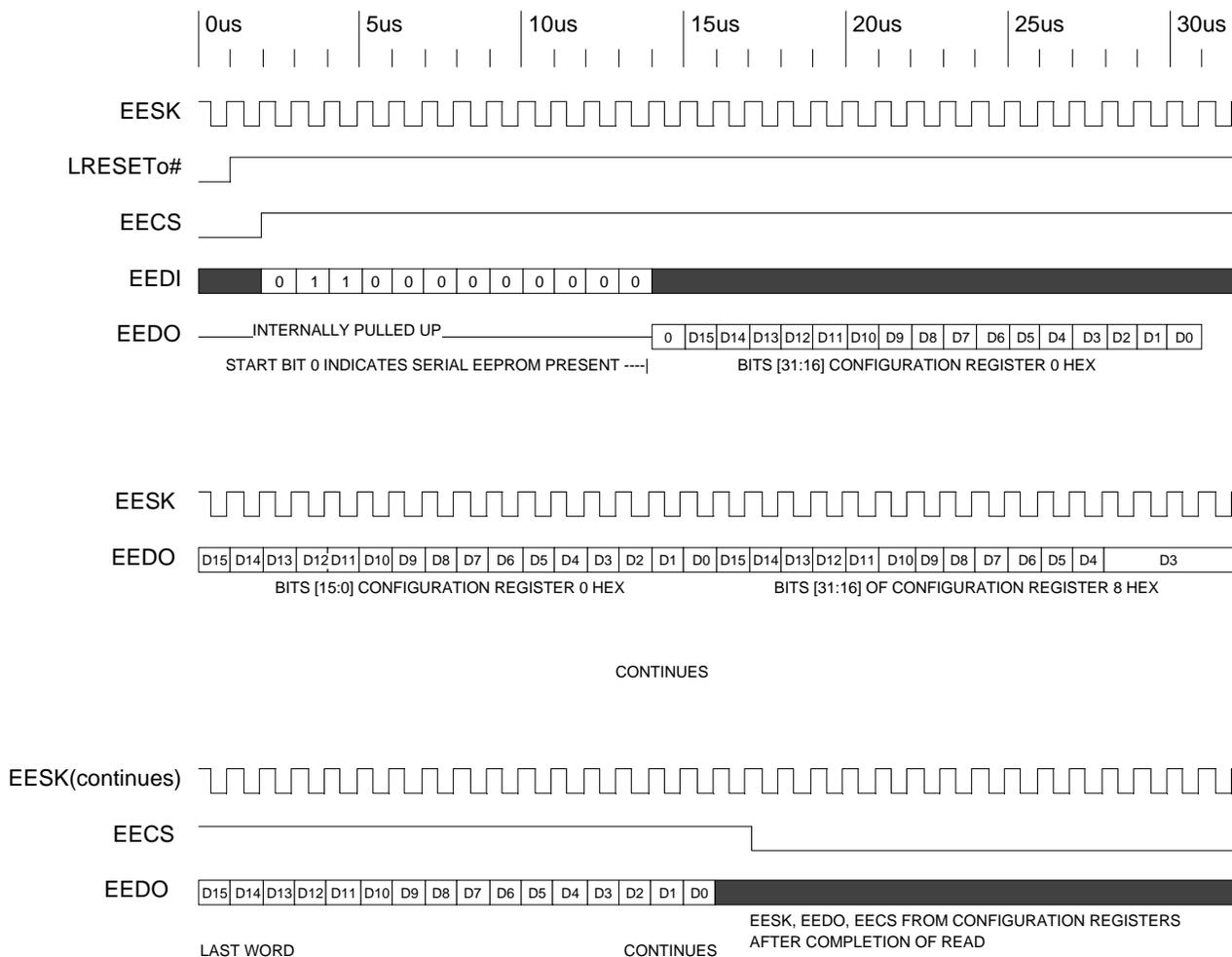


Timing Diagram 3-24. PCI Target Burst Write Cycle of 10 Lwords, Zero Wait States beyond MPC860 Protocol, Bterm Enabled, Burst Enabled

Section 3—M Func Desc

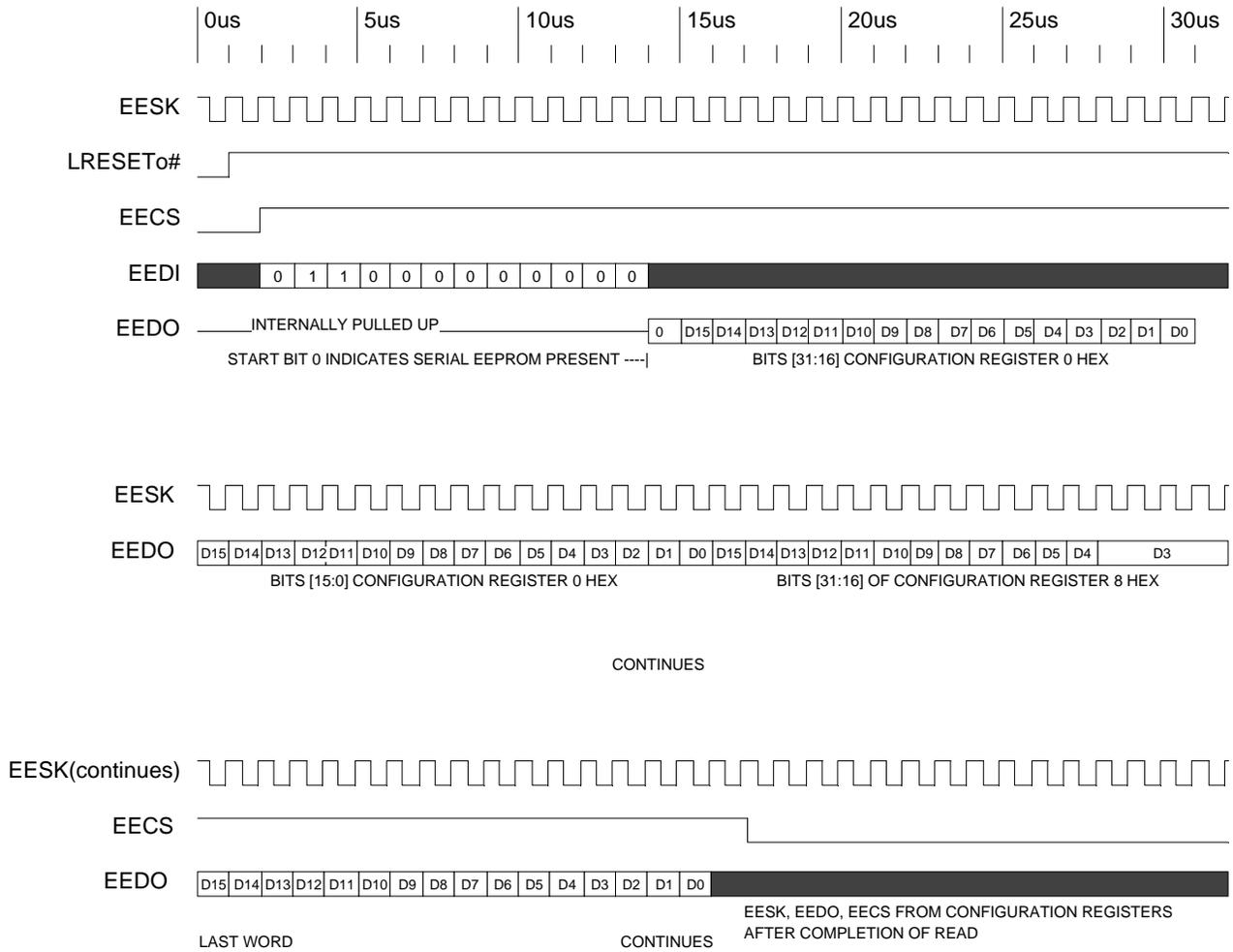


Timing Diagram 3-25. PCI Target Burst Read Cycle of 10 Lwords, Zero Wait States beyond MPC860 Protocol, Bterm Enabled, Burst Enabled

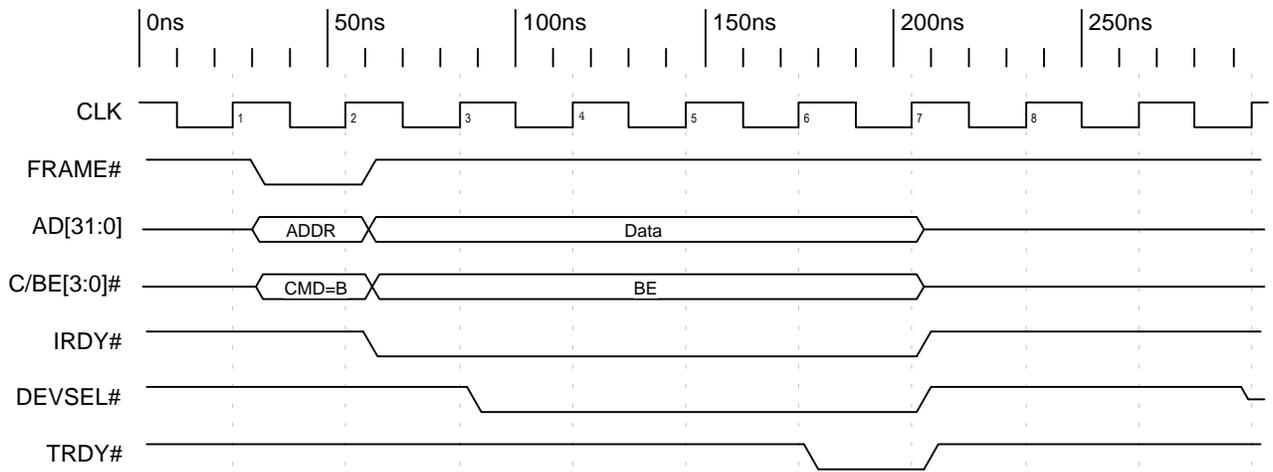


Timing Diagram 3-26. Initialization from Serial EEPROM (2K Bit)

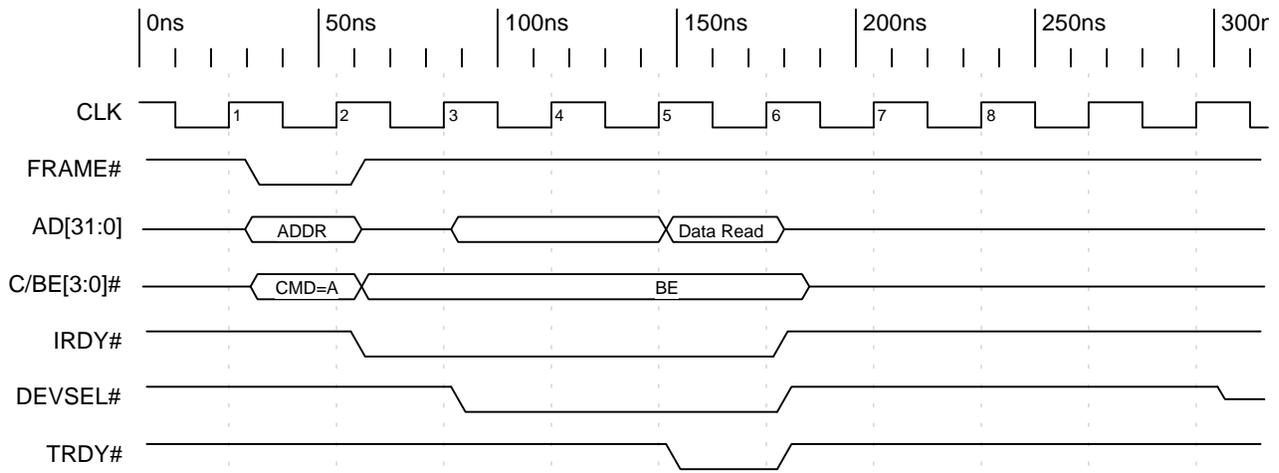
Section 3—M Func Desc



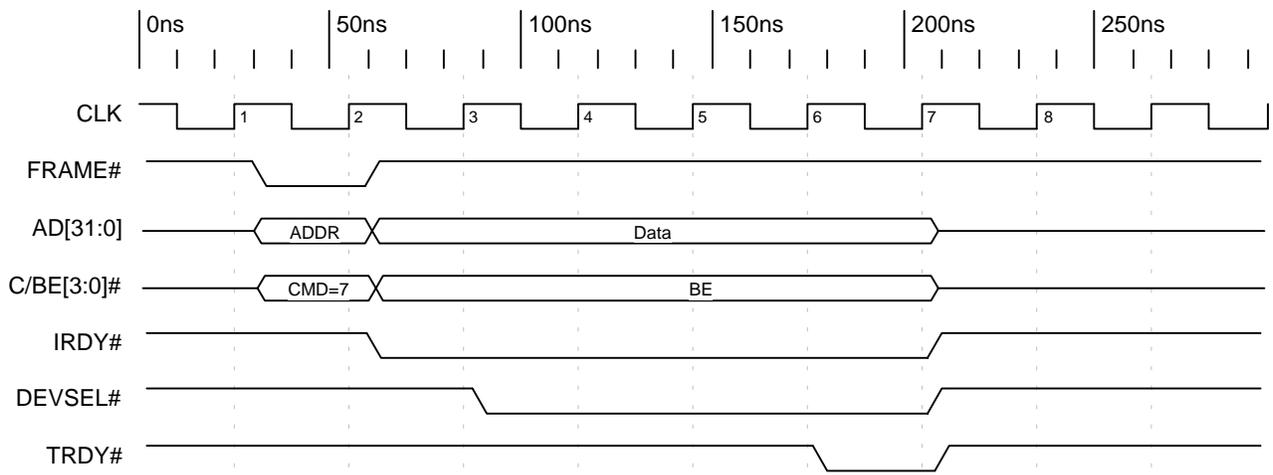
Timing Diagram 3-27. Initialization from Serial EEPROM (4K Bit)



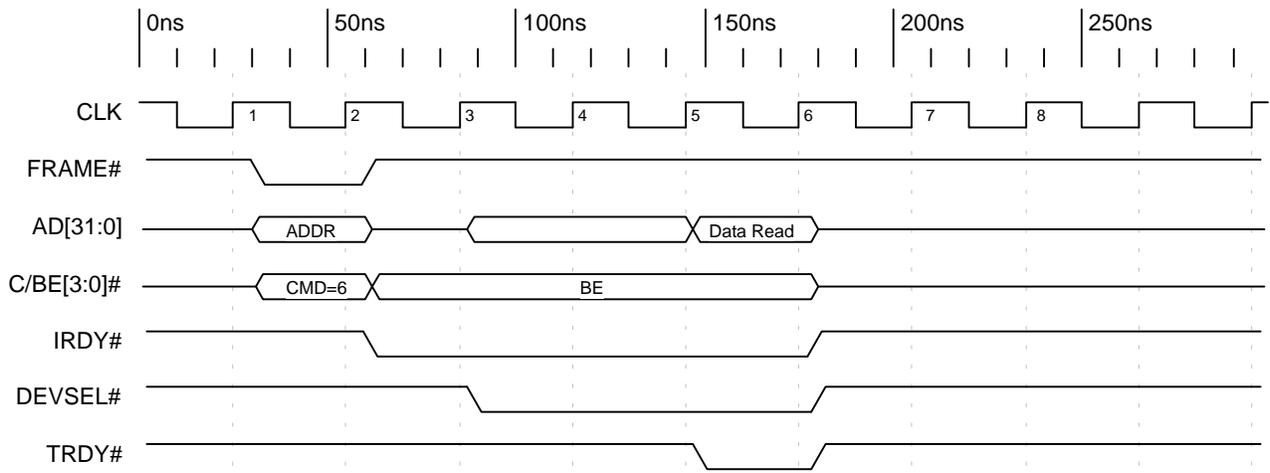
Timing Diagram 3-28. PCI Configuration Write to PCI Configuration Register



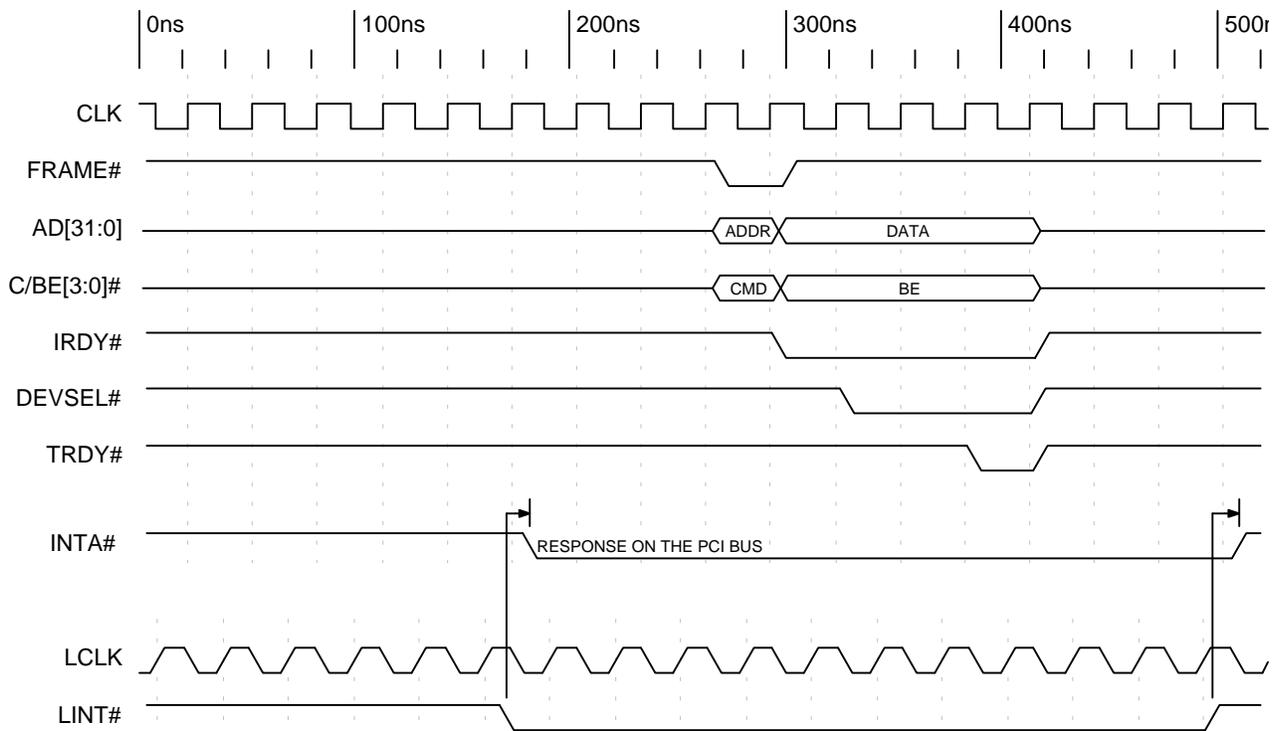
Timing Diagram 3-29. PCI Configuration Read to PCI Configuration Register



Timing Diagram 3-30. PCI Memory Write to Local Configuration Register

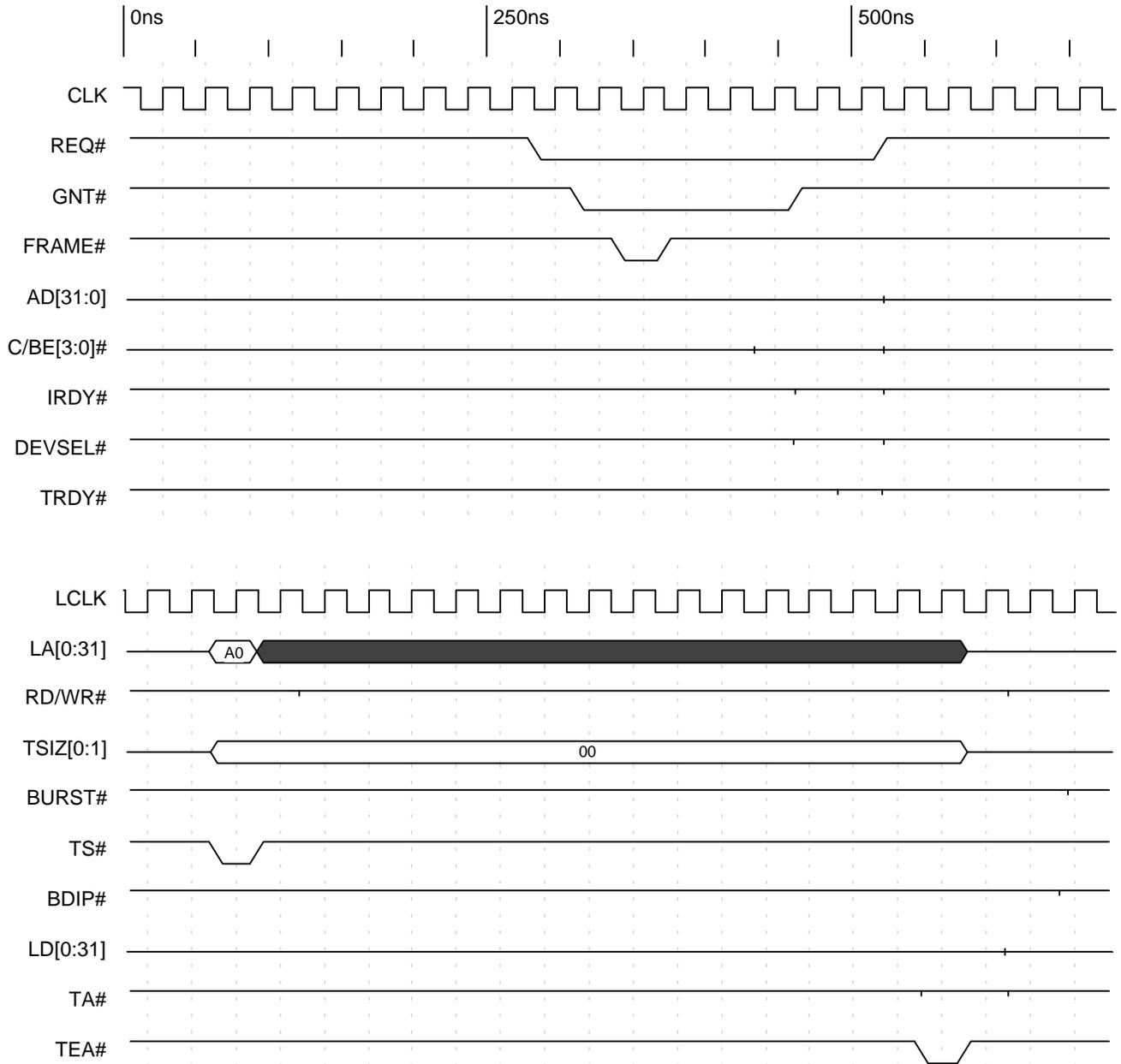


Timing Diagram 3-31. PCI Memory Read to Local Configuration Register



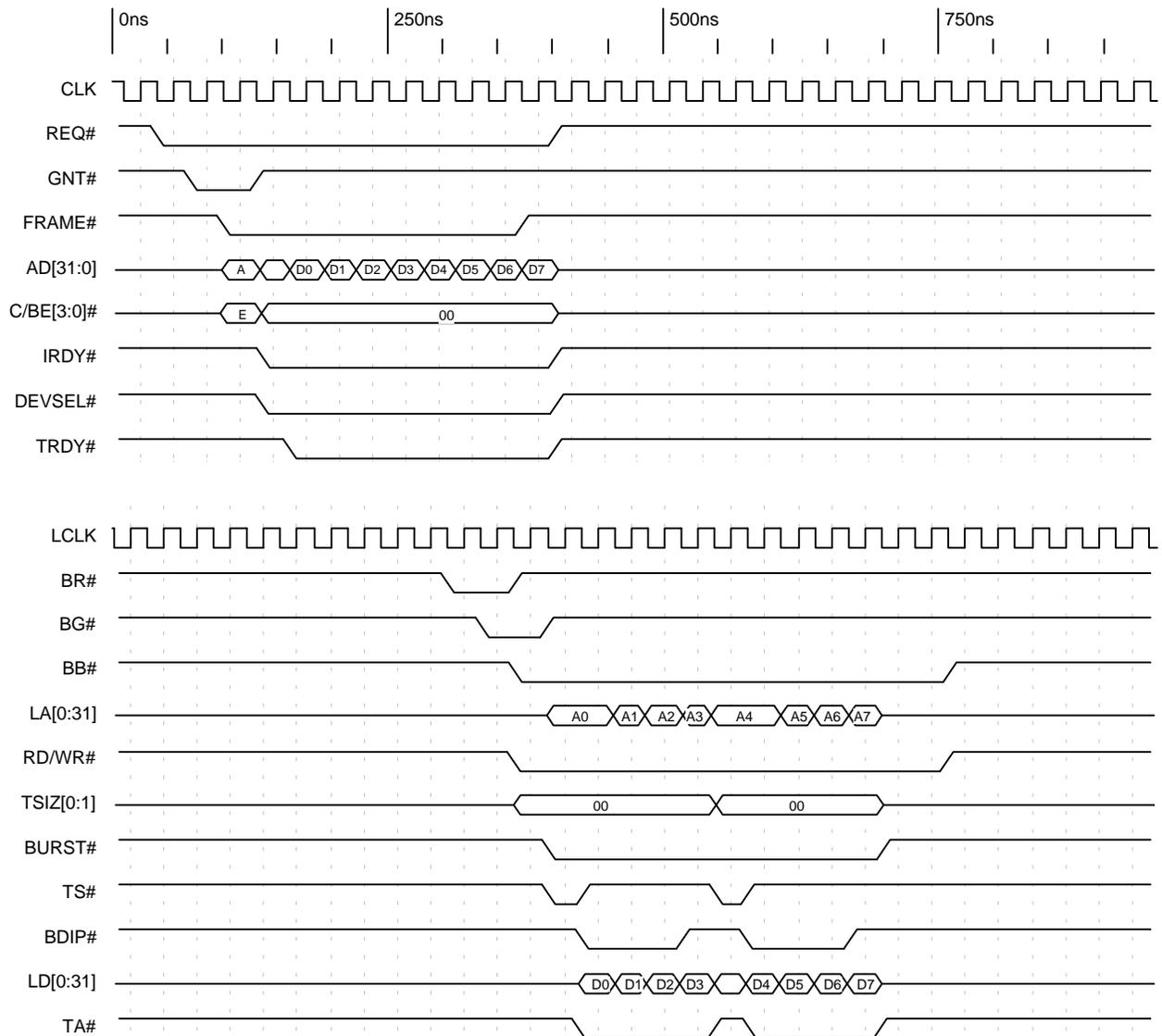
Timing Diagram 3-32. Local Interrupt Asserting PCI Interrupt

### 3.6.3 M Mode DMA

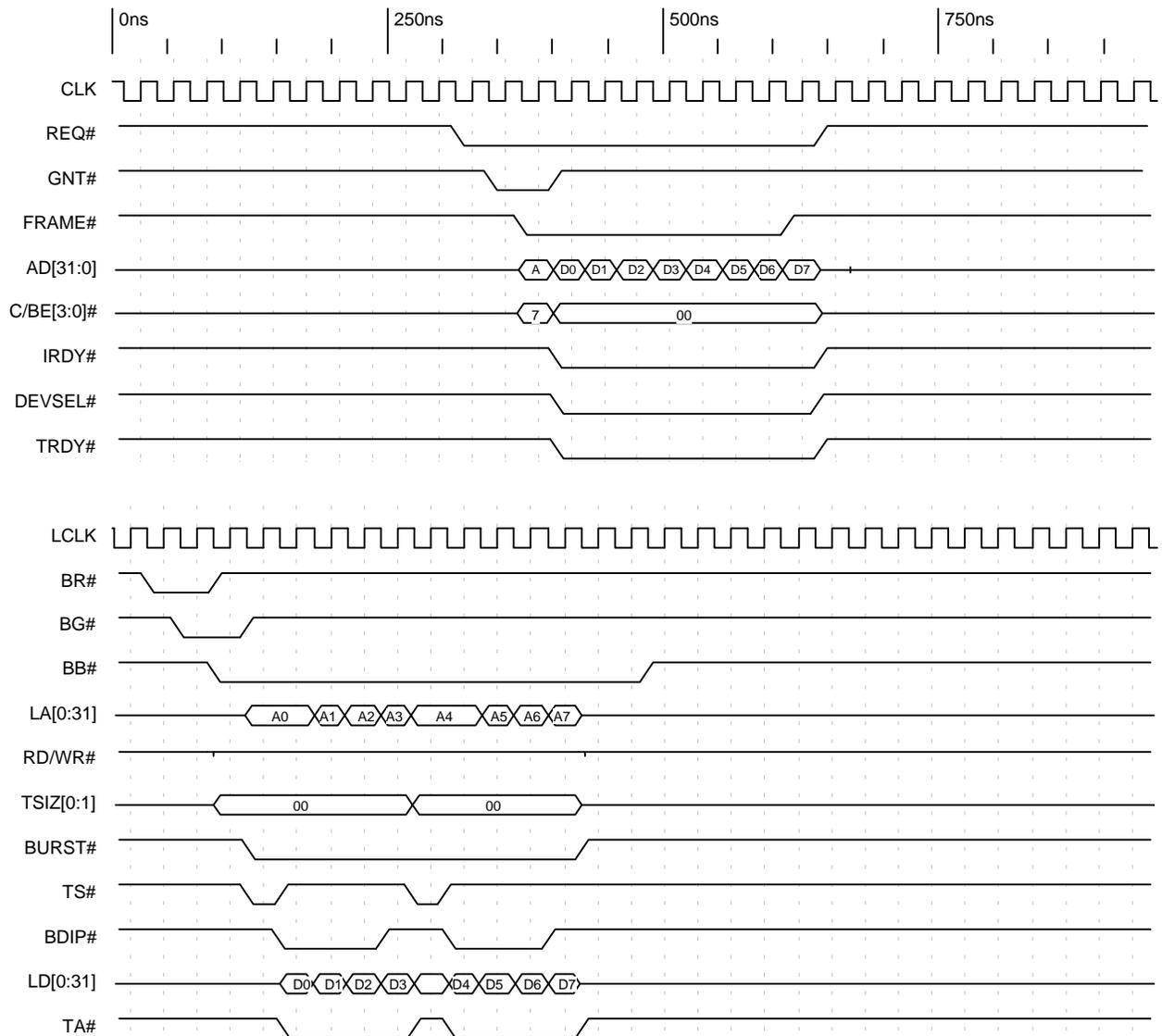


Timing Diagram 3-33. Master Abort Condition During PCI Initiator Read Cycle Causes TEA#

Section 3—M Func Desc

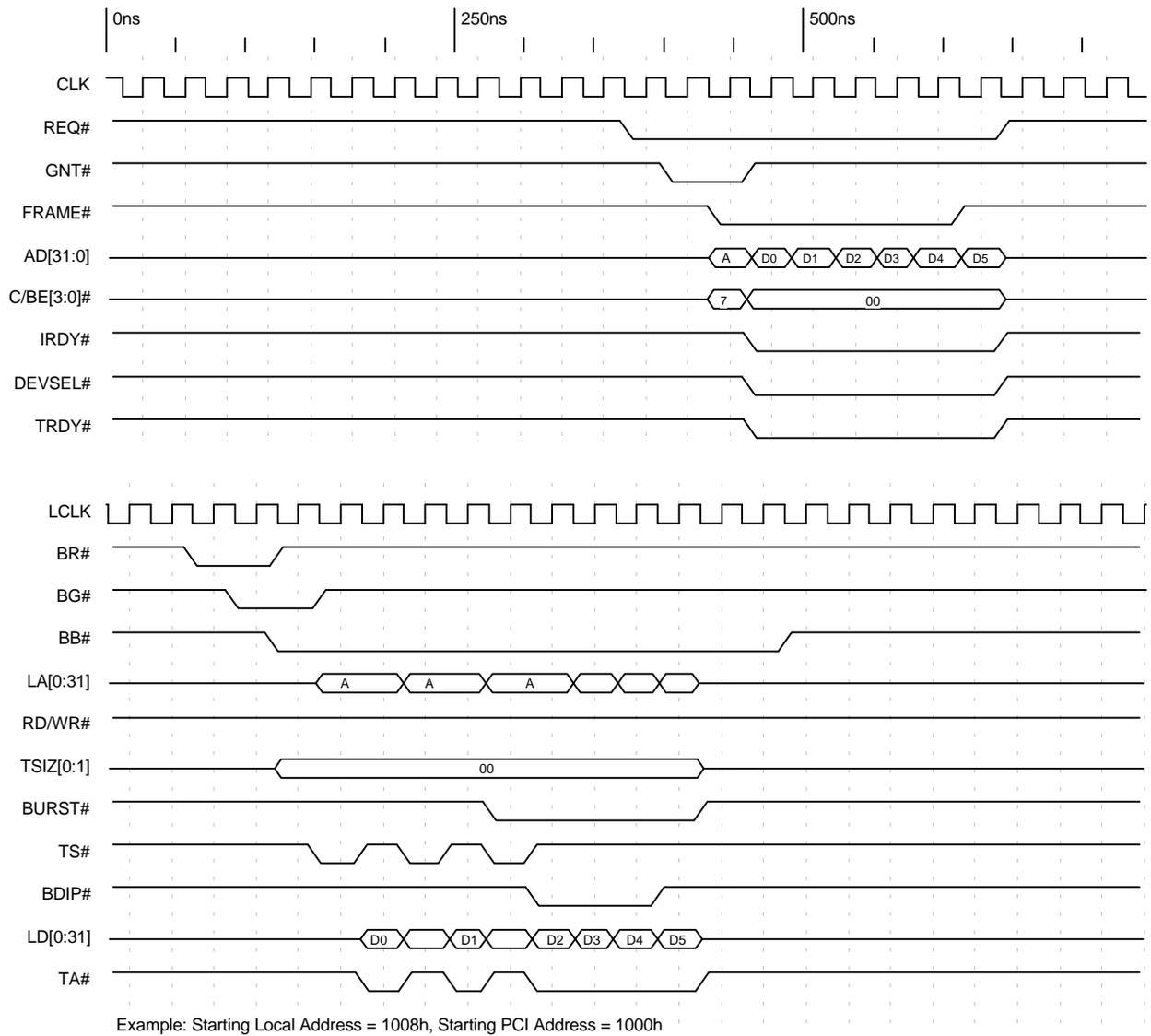


Timing Diagram 3-34. DMA PCI-to-Local, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords

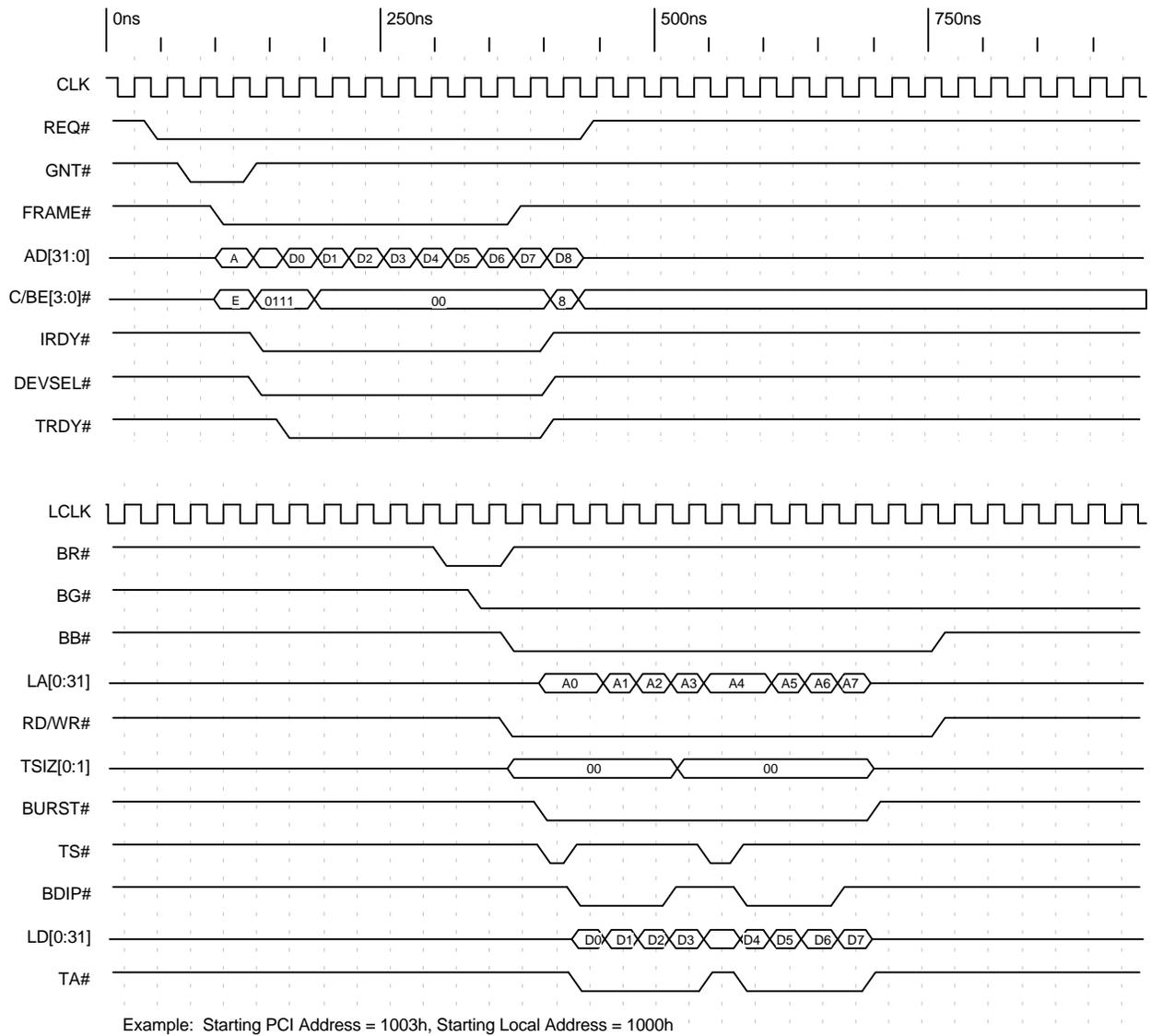


Timing Diagram 3-35. DMA Local-to-PCI, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords

Section 3—M Func Desc

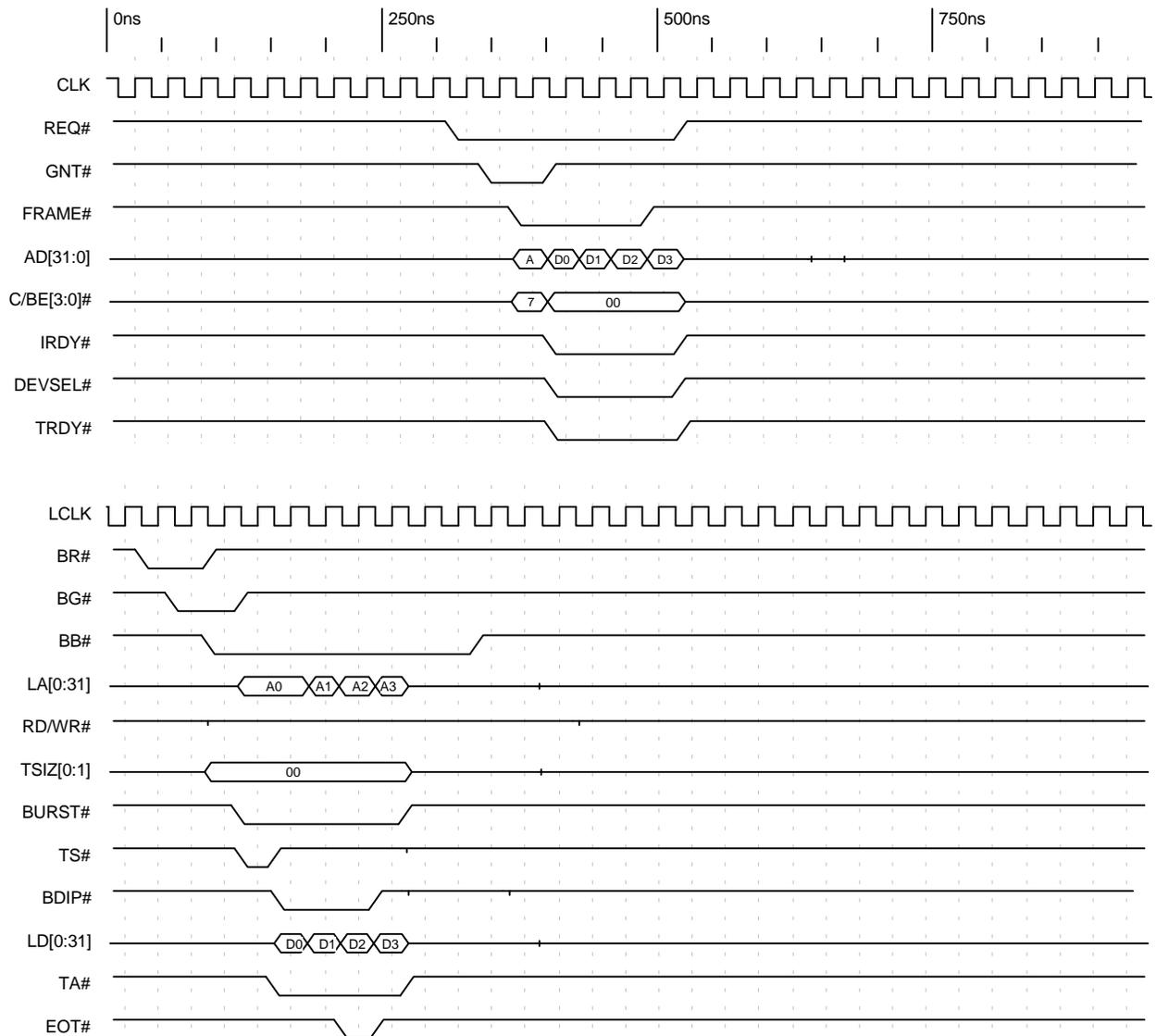


**Timing Diagram 3-36. DMA Local-to-PCI, Address Unaligned, Bterm Disabled, Burst Enabled, Transfer Size = Six Lwords**

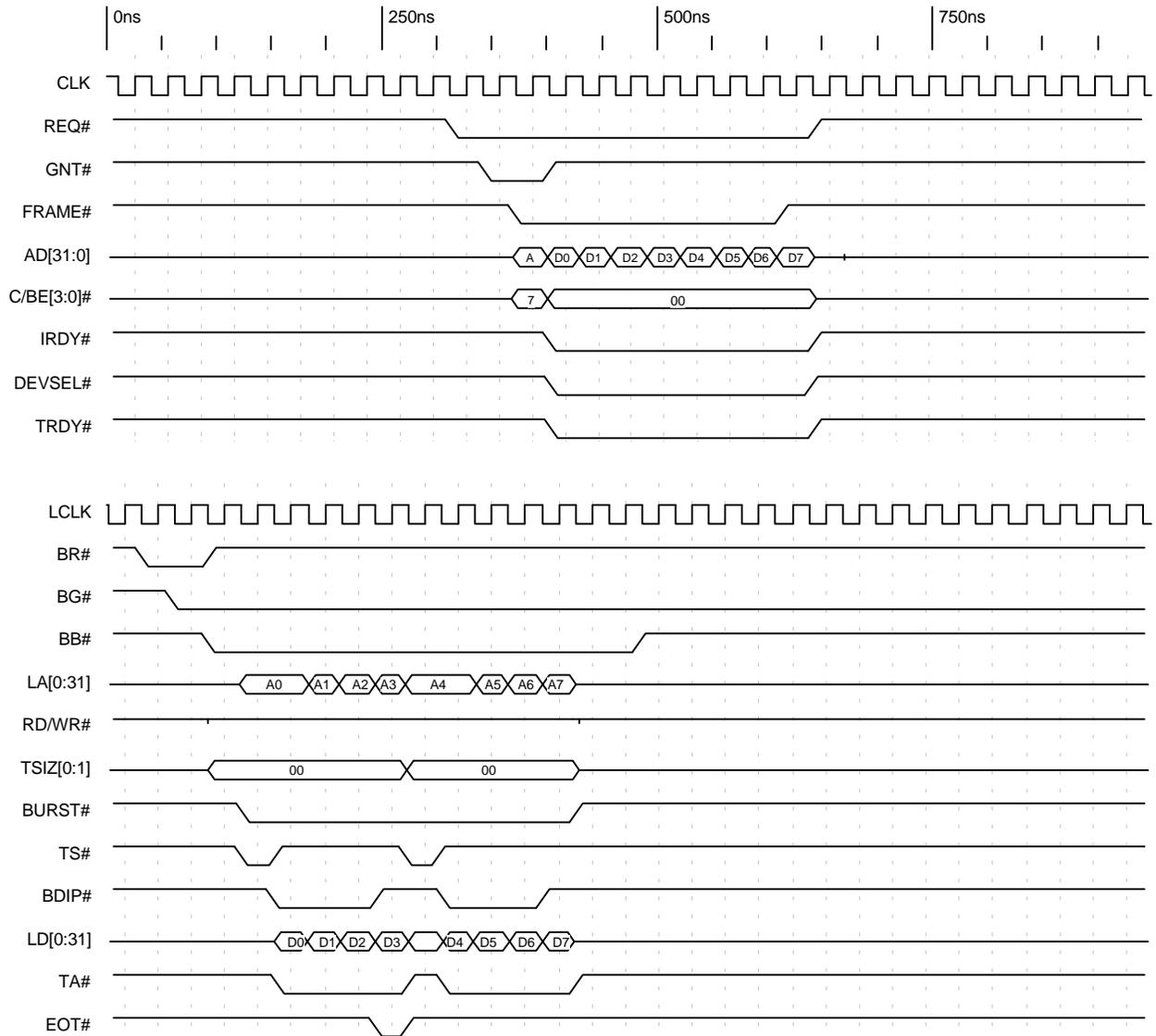


Timing Diagram 3-37. DMA PCI-to-Local, Address Unaligned, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords

Section 3—M Func Desc

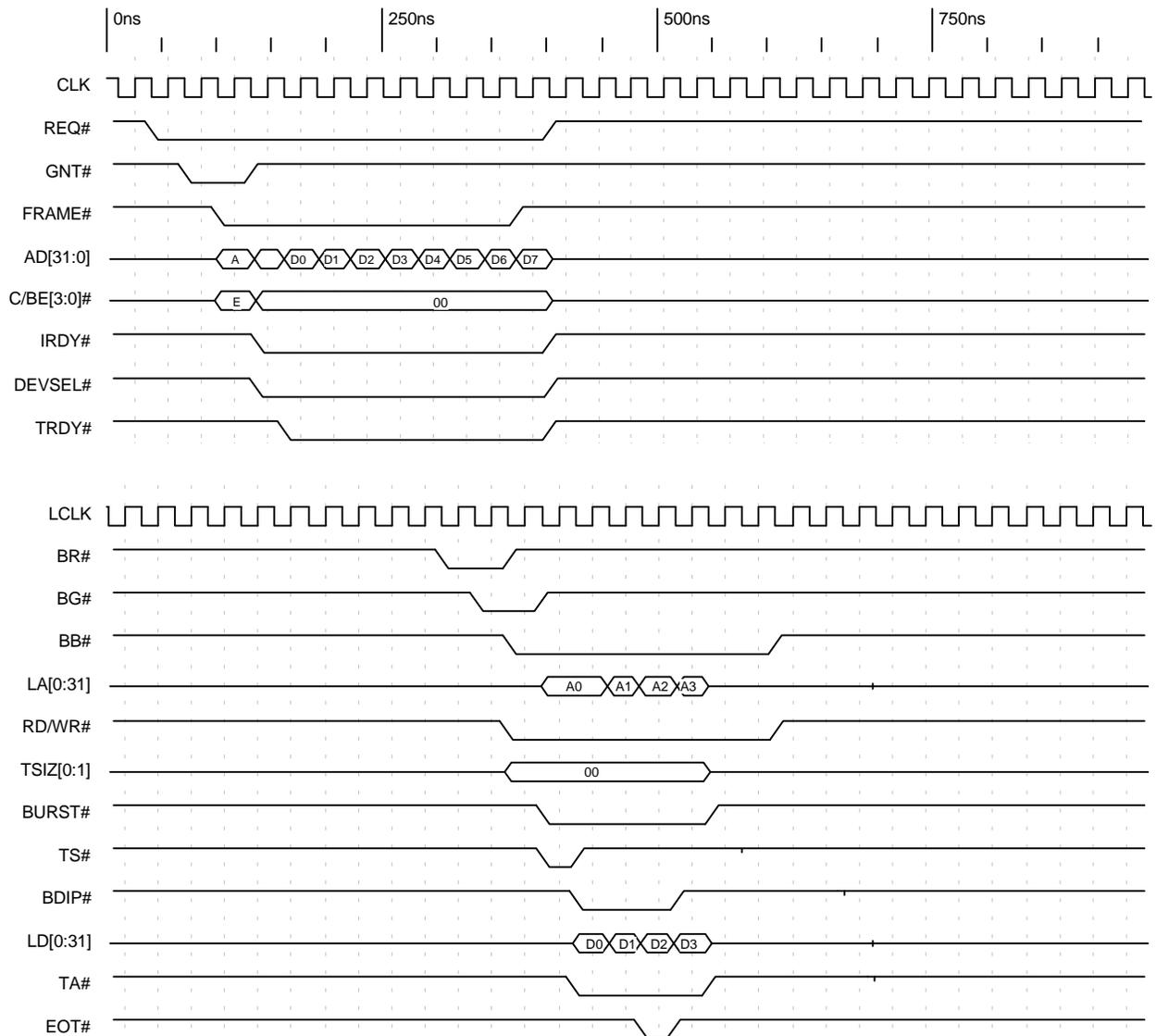


Timing Diagram 3-38. DMA Local-to-PCI, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts in the Middle of the Quad-Lword of Data

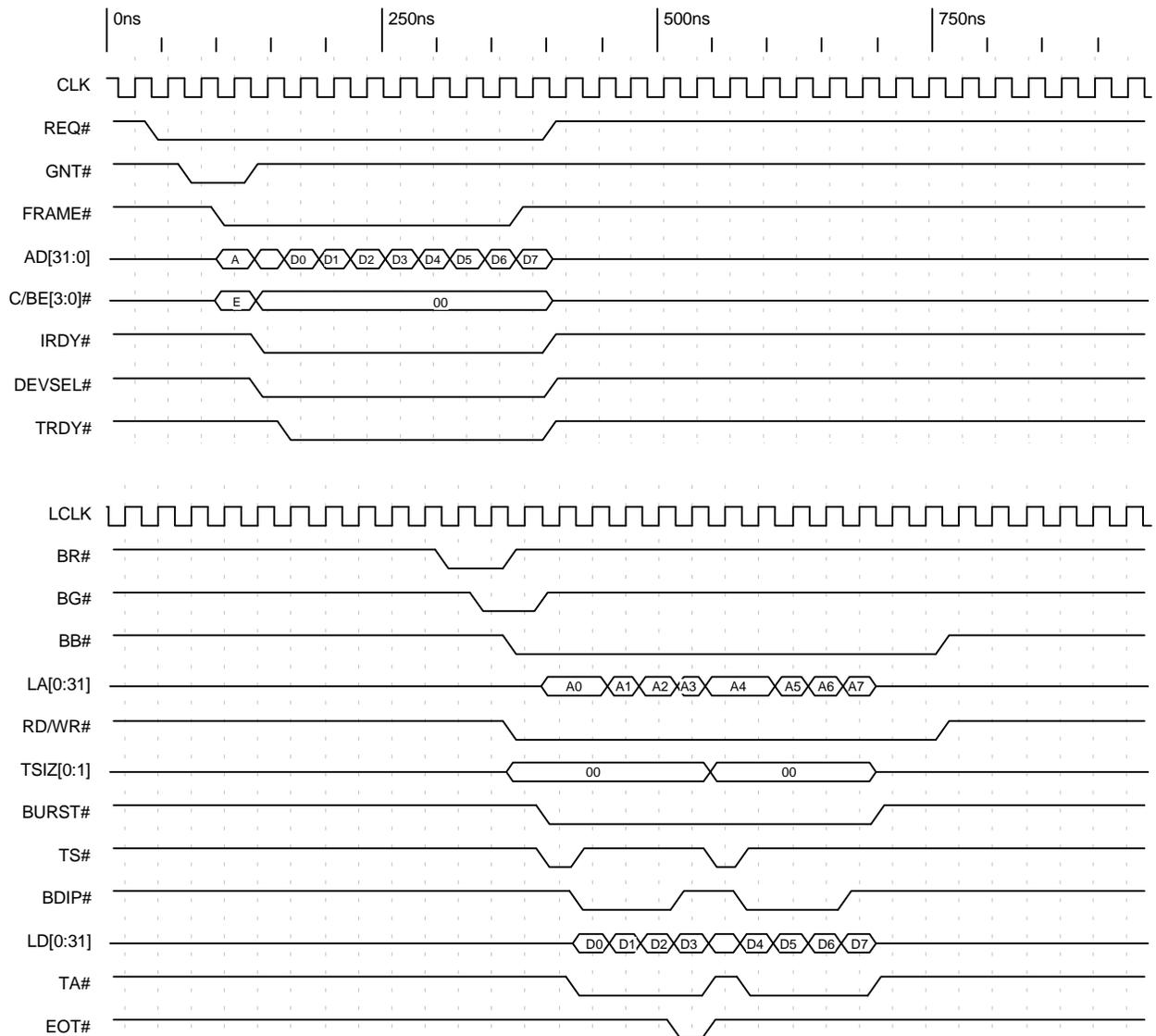


Timing Diagram 3-39. DMA Local-to-PCI, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts at the Last Data of the First Quad-Lword

Section 3—M Func Desc

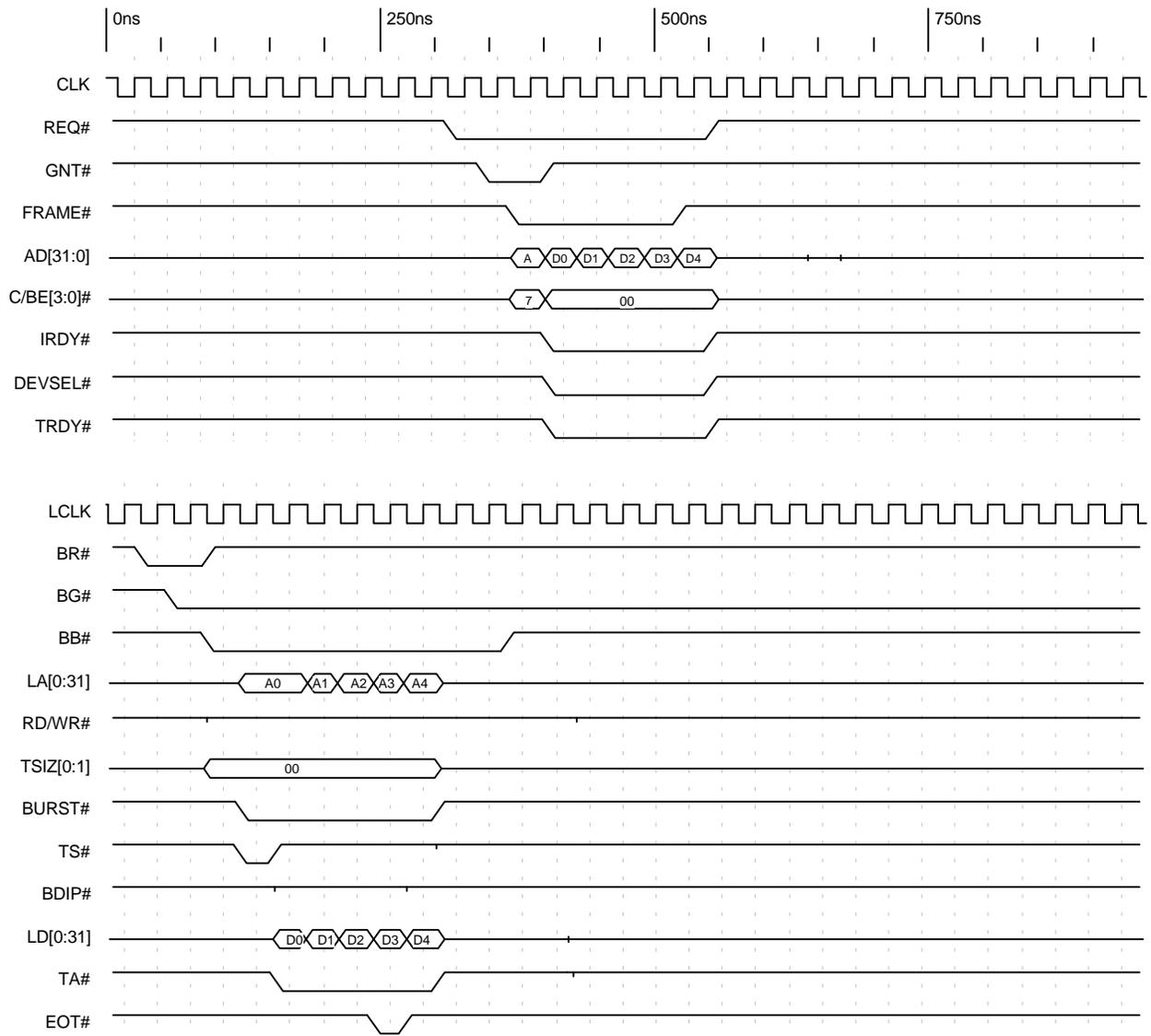


Timing Diagram 3-40. DMA PCI-to-Local, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts in the Middle of the First Quad-Lword of Data

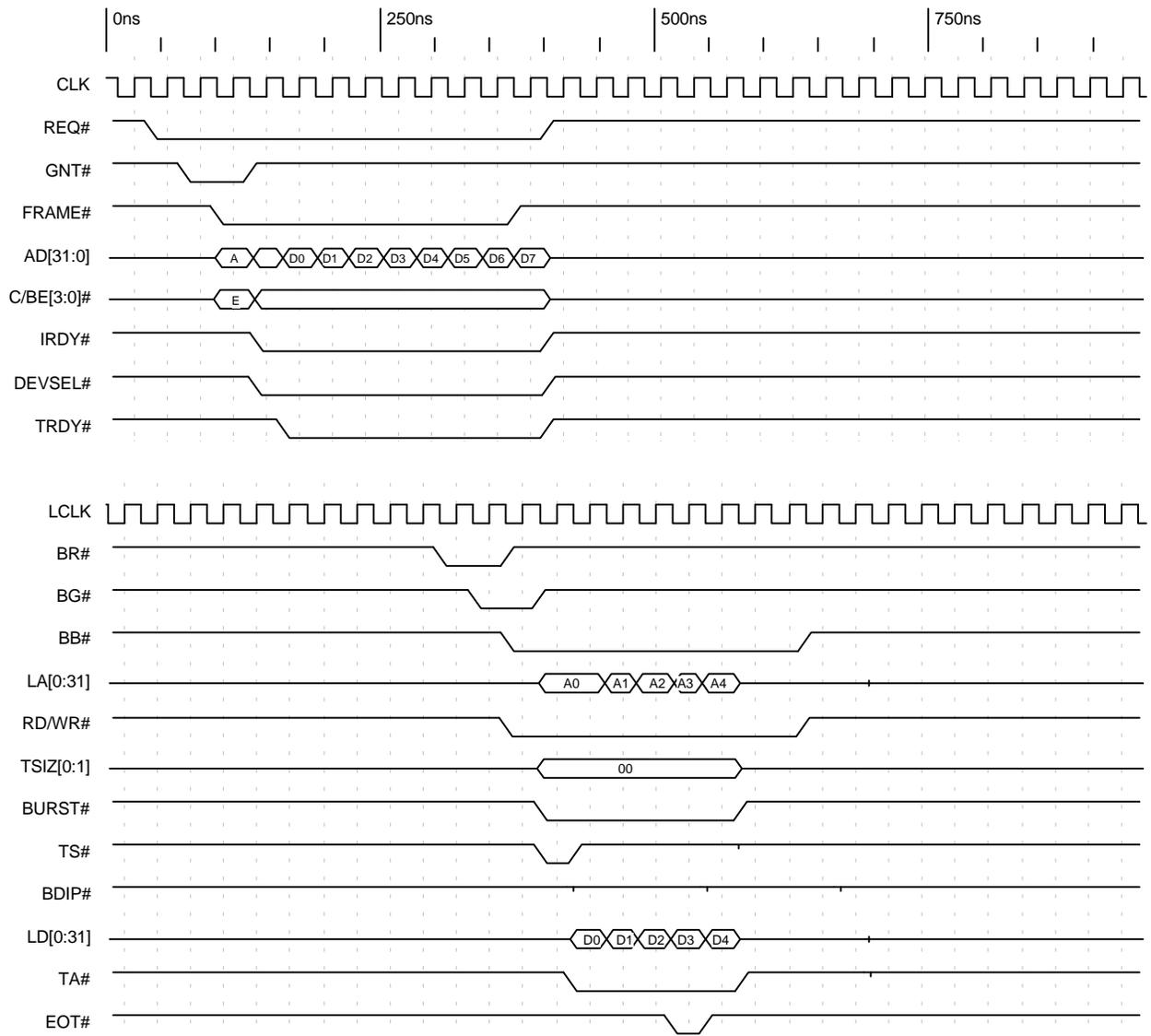


Timing Diagram 3-41. DMA PCI-to-Local, Bterm Disabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts at the Last Data of the First Quad-Lword

Section 3—M Func Desc

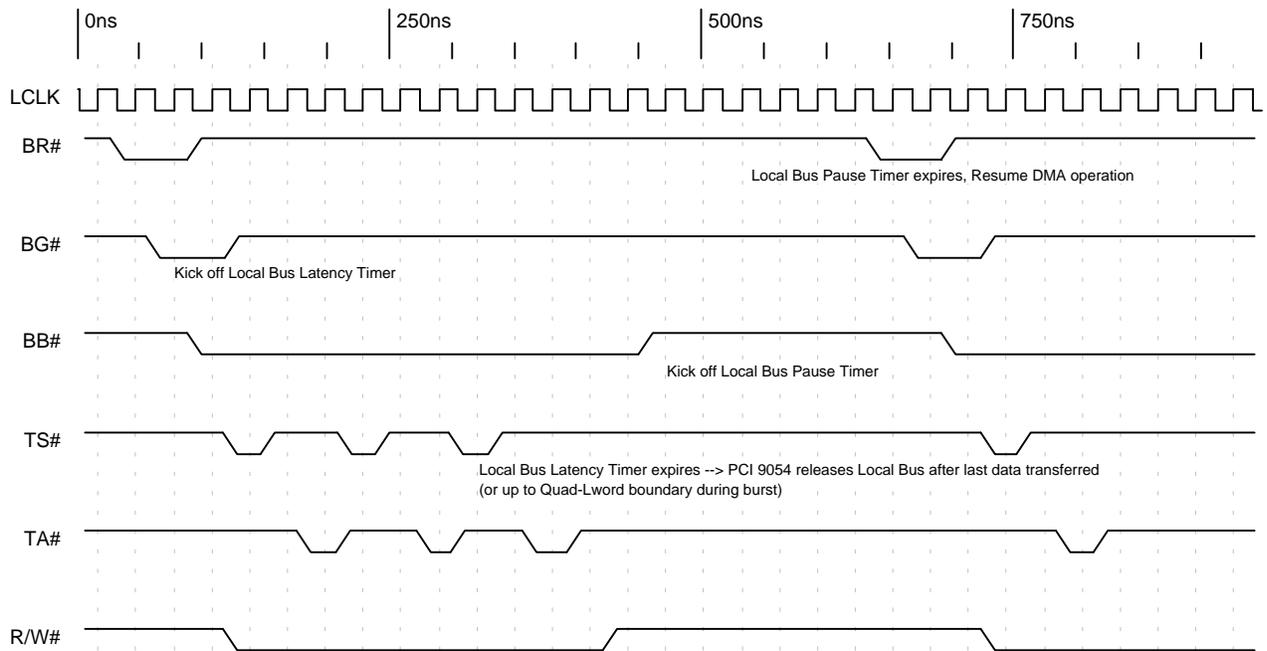


Timing Diagram 3-42. DMA Local-to-PCI, Bterm Enabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts at End of Third Local Data beyond MPC860 Protocol

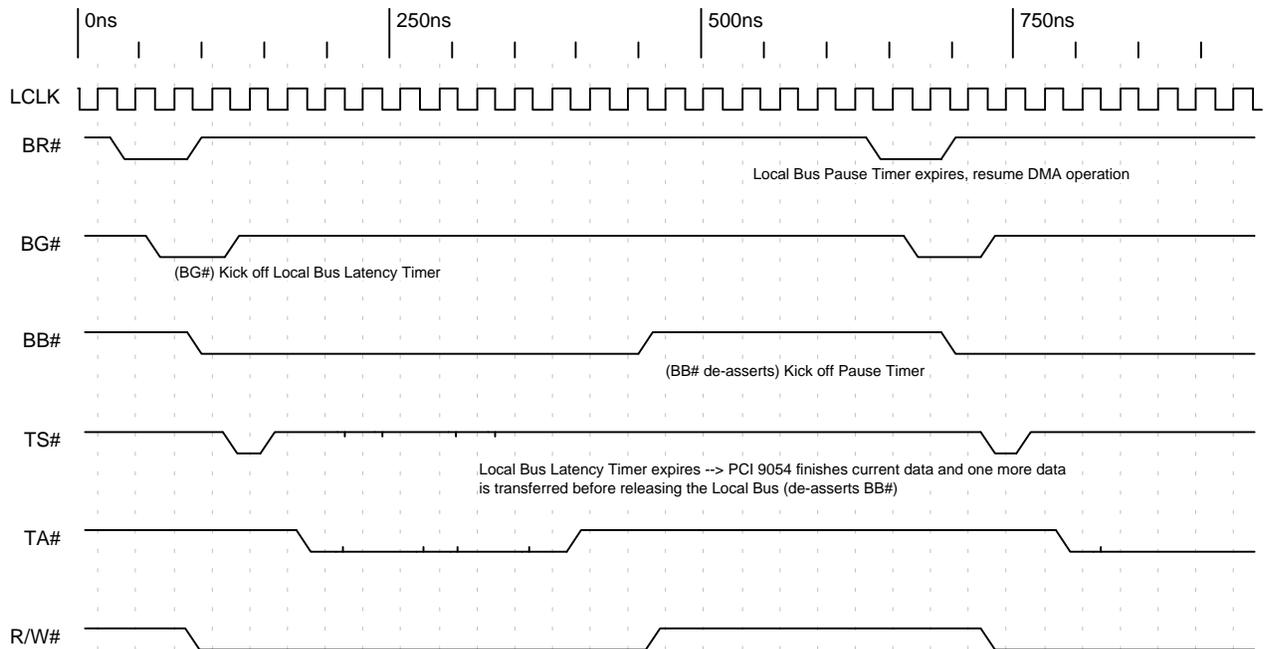


Timing Diagram 3-43. DMA PCI-to-Local, Bterm Enabled, Burst Enabled, Transfer Size = Eight Lwords, EOT# Asserts at End of Third Local Data beyond MPC860 Protocol

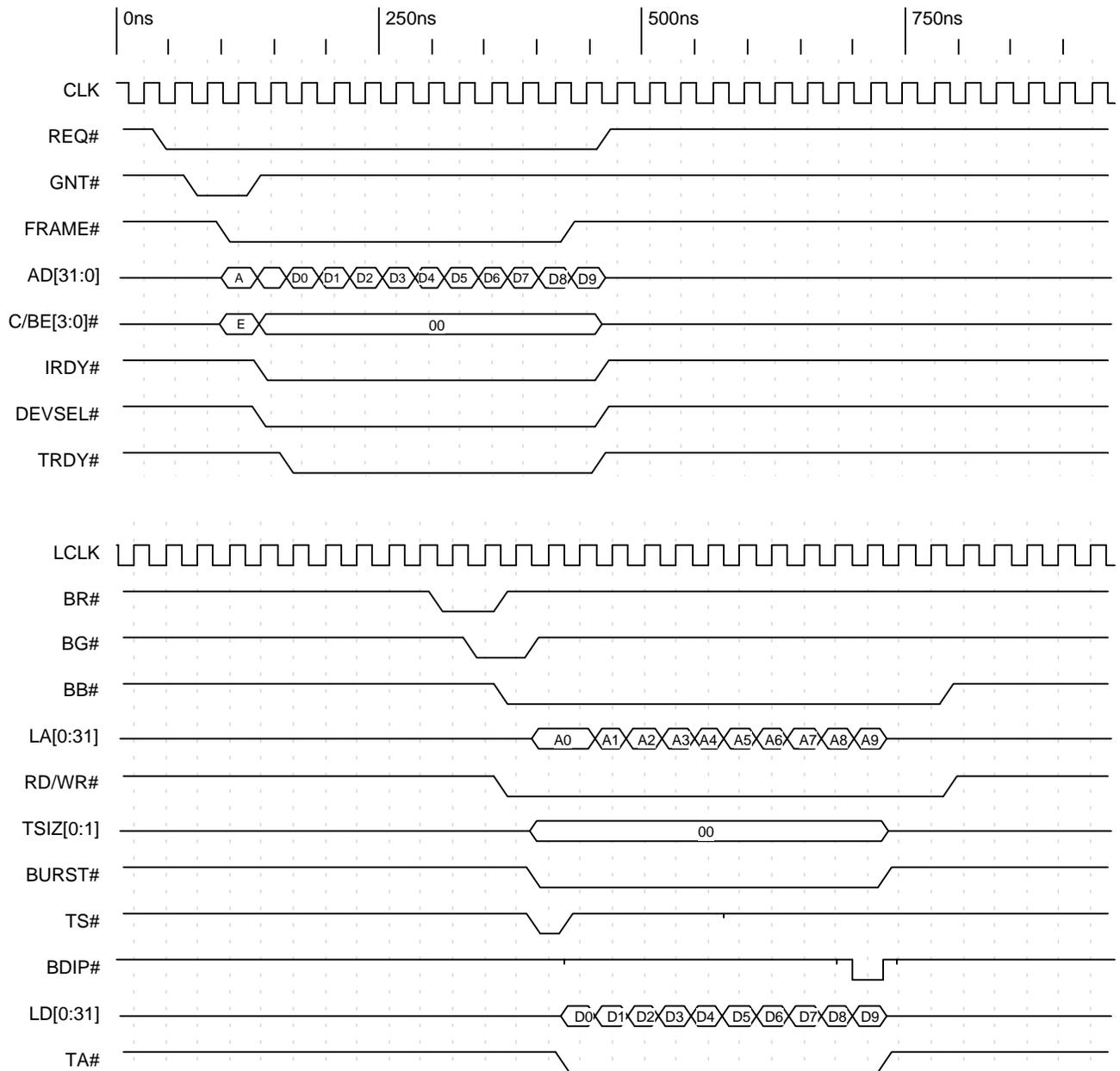
Section 3—M Func Desc



Timing Diagram 3-44. Local Bus Latency Timer (Eight Clocks) and Pause Timer (Four Clocks) in DMA Operation

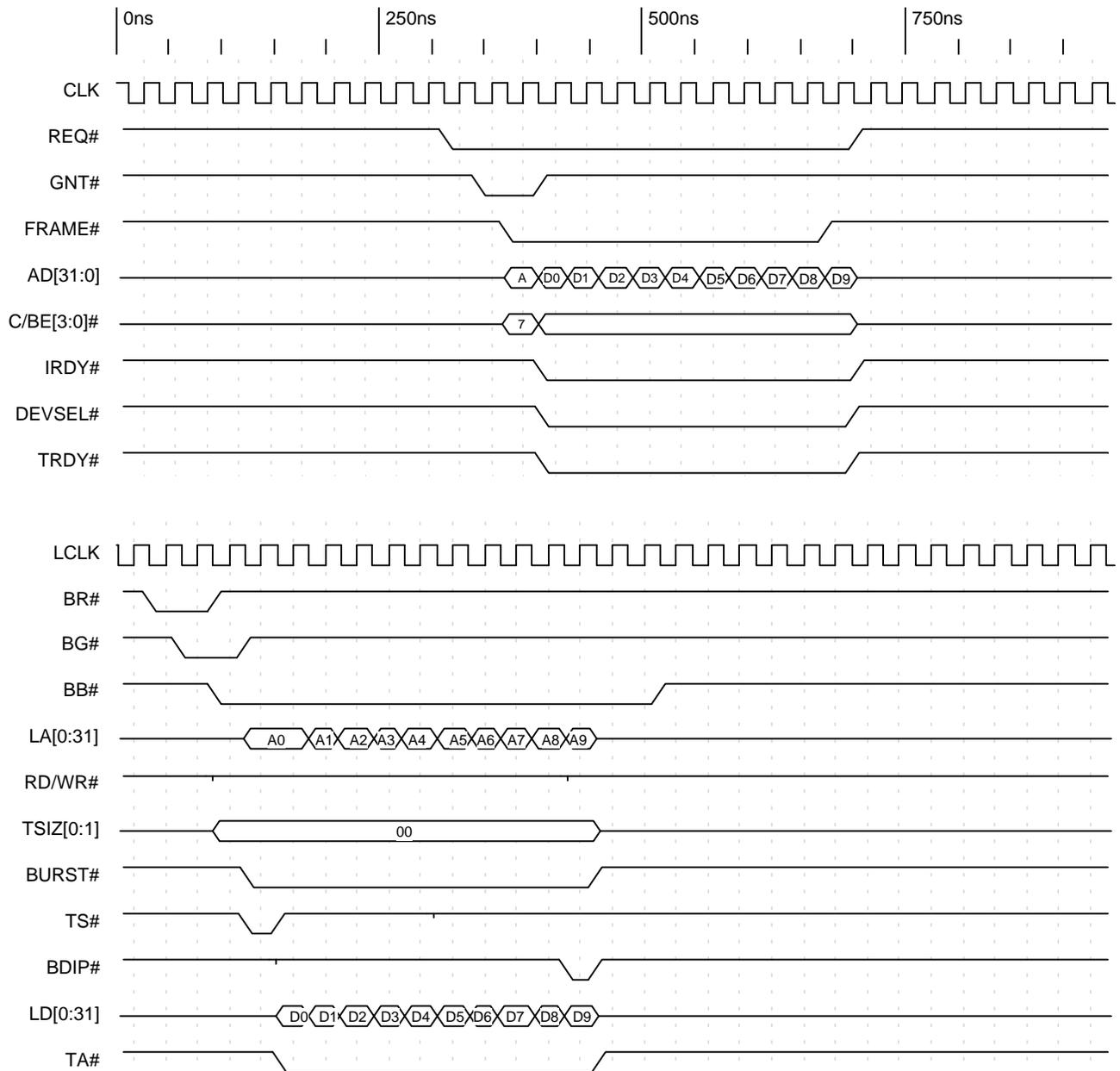


Timing Diagram 3-45. Local Bus Latency Timer (Eight Clocks) and Pause Timer (Four Clocks) in DMA Operation beyond MPC860 Protocol

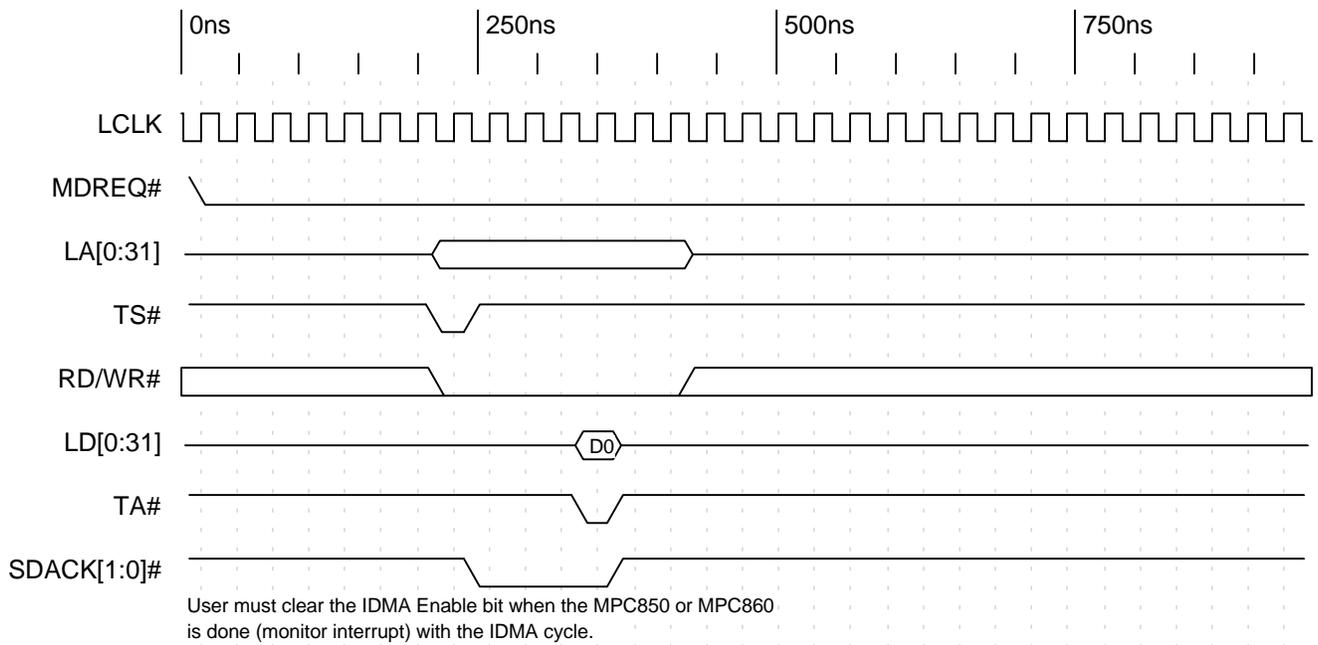


Timing Diagram 3-46. DMA PCI-to-Local, Bterm Enabled, Burst Enabled, Transfer Size = 10 Lwords, beyond MPC860 Protocol

Section 3—M Func Desc



Timing Diagram 3-47. DMA Local-to-PCI, Bterm Enabled, Burst Enabled, Transfer Size = 10 Lwords, beyond MPC860 Protocol



**Timing Diagram 3-48. IDMA Single Write Cycle**

**Notes:** The PCI 9054 treats the IDMA function from the MPC850 or MPC860 the same as a PCI Initiator cycle. The MPC850 or MPC860 starts IDMA cycle when the IDMA Enable bit is set in the MPC850 or MPC860 respective register. The PCI 9054 does not look at SDACK[1:0]# because the pins do not exist in the PCI 9054 (not connected).

Section 3—M Func Desc



# 4 C AND J MODES BUS OPERATION

## 4.1 PCI BUS CYCLES

The PCI 9054 is compliant with PCI Specification v2.2. Refer to PCI Specification v2.2 for specific PCI Bus functions.

### 4.1.1 PCI Target Command Codes

As a Target, the PCI 9054 allows access to the PCI 9054 internal registers and the Local Bus, using the commands listed in Table 4-1.

All Read or Write accesses to the PCI 9054 can be Byte, Word, or Lword (longword) accesses, defined as 32 bit. All memory commands are aliased to basic memory commands. All I/O accesses to the PCI 9054 are decoded to an Lword boundary. Byte enables are used to determine which bytes are read or written. An I/O access with illegal byte enable combinations is terminated with a Target Abort.

**Table 4-1. PCI Target Command Codes**

Command Type	Code (C/BE[3:0]#)
I/O Read	0010 (2h)
I/O Write	0011 (3h)
Memory Read	0110 (6h)
Memory Write	0111 (7h)
Configuration Read	1010 (Ah)
Configuration Write	1011 (Bh)
Memory Read Multiple	1100 (Ch)
PCI Dual Address Cycle	1101 (Dh)
Memory Read Line	1110 (Eh)
Memory Write and Invalidate	1111 (Fh)

### 4.1.2 PCI Master Command Codes

The PCI 9054 can access the PCI Bus to perform DMA or PCI Initiator Local-to-PCI Bus transfers. During a PCI Initiator or DMA transfer, the command code assigned to the PCI 9054 internal register location (CNTRL[15:0]) is used as the PCI command code (except for Memory Write and Invalidate mode for DMA cycles where (DMPBAM[9]=1). Table 4-2 through Table 4-5 lists various PCI Master Command codes.

**Notes:** Programmable internal registers determine PCI command codes when the PCI 9054 is the Master. DMA cannot perform I/O or configuration accesses.

### 4.1.2.1 DMA Master Command Codes

DMA controllers of the PCI 9054 can assert the Memory cycles listed in Table 4-2.

**Table 4-2. DMA Master Command Codes**

Command Type	Code (C/BE[3:0]#)
Memory Read	0110 (6h)
Memory Write	0111 (7h)
Memory Read Multiple	1100 (Ch)
PCI Dual Address Cycle	1101 (Dh)
Memory Read Line	1110 (Eh)
Memory Write and Invalidate	1111 (Fh)

### 4.1.2.2 Direct Local-to-PCI Command Codes

For direct Local-to-PCI Bus accesses, the PCI 9054 asserts the cycles listed in Table 4-3 through Table 4-5.

**Table 4-3. Local-to-PCI Memory Access**

Command Type	Code (C/BE[3:0]#)
Memory Read	0110 (6h)
Memory Write	0111 (7h)
Memory Read Multiple	1100 (Ch)
PCI Dual Address Cycle	1101 (Dh)
Memory Read Line	1110 (Eh)
Memory Write and Invalidate	1111 (Fh)

**Table 4-4. Local-to-PCI I/O Access**

Command Type	Code (C/BE[3:0]#)
I/O Read	0010 (2h)
I/O Write	0011 (3h)

**Table 4-5. Local-to-PCI Configuration Access**

Command Type	Code (C/BE[3:0]#)
Configuration Memory Read	1010 (Ah)
Configuration Memory Write	1011 (Bh)

### 4.1.3 PCI Arbitration

The PCI 9054 asserts REQ# to request the PCI Bus. The PCI 9054 can be programmed using the PCI Request Mode bit (MARBR[23]) to de-assert REQ# when it asserts FRAME# during a Bus Master cycle, or to keep REQ# asserted for the entire Bus Master cycle. The PCI 9054 always de-asserts REQ# for a minimum of two PCI clocks between Bus Master ownership that includes a Target disconnect.

The PCI Initiator Write Delay bits (DMPBAM[15:14]) can be programmed to delay the PCI 9054 from asserting PCI REQ# during a PCI Initiator Write cycle. DMPBAM can be programmed to wait 0, 4, 8, or 16 PCI Bus clocks after the PCI 9054 has received its first Write data from the Local Bus Master and is ready to begin the PCI Write transaction. This function is useful in applications where a Local Master is bursting and a Local Bus clock is slower than the PCI Bus clock. This allows Write data to accumulate in the PCI 9054 PCI Initiator Write FIFO, which provides for better use of the PCI Bus.

## 4.2 LOCAL BUS CYCLES

The PCI 9054 interfaces a PCI Host bus to several Local Bus types, as listed in Table 4-6 and Table 4-7. It operates in one of three modes, selected through MODE[1:0] (PQFP—Pins 157 and 156; PBGA—Pins B7 and E8), corresponding to three bus types—M, J, and C.

Table 4-6. Local Bus Types (176-Pin PQFP)

Pin 157	Pin 156	Mode	Bus Type
1	1	M	32-bit non-multiplexed
1	0	Reserved	—
0	1	J	32-bit multiplexed
0	0	C	32-bit non-multiplexed

Table 4-7. Local Bus Types (225-Pin PBGA)

Pin B7	Pin E8	Mode	Bus Type
1	1	M	32-bit non-multiplexed
1	0	Reserved	—
0	1	J	32-bit multiplexed
0	0	C	32-bit non-multiplexed

### 4.2.1 Local Bus Arbitration

The PCI 9054 asserts L\_HOLD to request the Local Bus. It owns the Local Bus when L\_HOLD and L\_HOLD\_A are asserted. When the PCI 9054 acknowledges BREQ\_i assertion during DMA or PCI Target Write transfers, it releases the Local Bus within two Lword transfers by de-asserting L\_HOLD and floating the Local Bus outputs if either of the following conditions exist:

- BREQ\_i is asserted and enabled
- Gating is enabled and the Local Bus Latency Timer is enabled and expires (MARBR[27, 7:0])

The Local Arbiter can now grant the Local Bus to another Local Master. After the PCI 9054 acknowledges that L\_HOLD\_A is de-asserted and the Local Bus Pause Timer is zero, it re-asserts L\_HOLD to request the Local Bus. When the PCI 9054 receives L\_HOLD\_A, it drives the bus and continues the transfer.

**Note:** The Local Bus Pause Timer applies only to DMA operation. It does **not** apply to PCI Target operation.

### 4.2.2 PCI Initiator

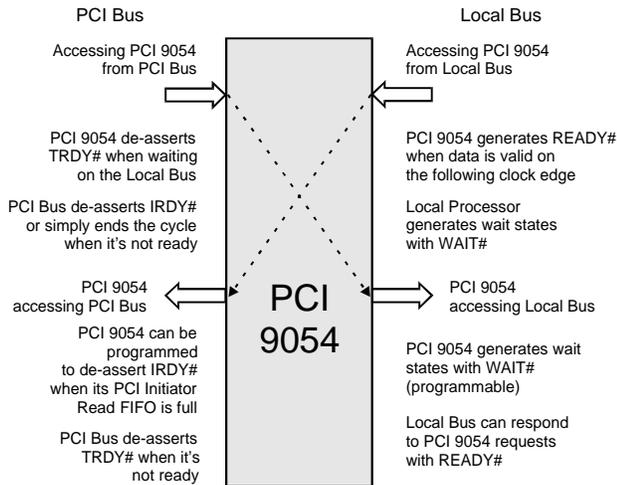
Local Bus cycles can be Single or Burst cycles. The BLAST# signal is used to determine if a Single or Burst cycle is to be performed. If BLAST# is asserted at the beginning of the first Data phase, on which the PCI 9054 performs a Single PCI Bus cycle. Otherwise, the PCI 9054 performs a Burst PCI Bus cycle and BLAST# is used to end the cycle. As a Local Bus Target, the PCI 9054 allows access to the PCI 9054 internal registers and the PCI Bus. Non-32-bit PCI Initiator accesses to the PCI 9054 require simple external logic (latch array to combine data into a 31-bit bus).

Local Bus PCI Initiator accesses to the PCI 9054 must be for a 32-bit non-pipelined bus.

### 4.2.3 PCI Target

The PCI Bus Master reads from and writes to the Local Bus (the PCI 9054 is a PCI Bus Target and a Local Bus Master).

### 4.2.4 Wait State Control



**Figure 4-1. Wait States**

**Note:** The figure represents a sequence of Bus cycles.

If READY# mode is disabled, the external READY# input signal has no effect on wait states for a Local access. Wait states between Data cycles are asserted internally by a wait state counter. The wait state counter is initialized with its Configuration register value at the start of each data access.

If READY# mode is enabled, it has no effect until the wait state counter reaches 0. READY# then controls the number of additional wait states.

BTERM# input is not sampled until the wait state counter reaches 0. BTERM# overrides READY# when BTERM# is enabled and asserted.

#### 4.2.4.1 Wait States—Local Bus

In PCI Initiator mode and when accessing the PCI 9054 registers, the PCI 9054 acts as a Local Bus Slave. The PCI 9054 asserts wait states by delaying the READY# signal. The Local processor asserts wait states with the WAIT# signal.

In PCI Target and DMA modes, the PCI 9054 acts as a Local Bus Master. The PCI 9054 inserts internal wait states with the WAIT# signal. The Local processor asserts external wait states by delaying the READY# signal.

The Internal Wait State bit(s) (LBRD0[21:18, 5:2], (LBRD1[5:2]), DMAMODE0[5:2], and/or DMAMODE1

[5:2]) can be used to program the number of internal wait states between the first address-to-data (and subsequent data-to-data in Burst mode).

During PCI Initiator accesses, WAIT# signal must be asserted during the ADS phase for the PCI 9054 to sample the wait state phase.

In PCI Target and DMA modes, the READY# signal has no effect until the wait state counter—(LBRD0[21:18, 5:2]), (LBRD1[5:2]), (DMAMODE0[5:2]), and/or (DMAMODE1[5:2])—reaches zero. READY# then controls the number of wait states by being de-asserted in the middle of the data transaction.

#### 4.2.4.2 Wait States—PCI Bus

The PCI Bus Master throttles IRDY# and the PCI Bus Slave throttles TRDY# to assert PCI Bus wait state(s).

### 4.2.5 Burst Mode and Continuous Burst Mode (Bterm “Burst Terminate” Mode)

**Note:** In the following sections, Bterm refers to the PCI 9054 internal register bit and BTERM# refers to the PCI 9054 external signal.

#### 4.2.5.1 Burst and Bterm Modes

**Table 4-8. Burst and Bterm on the Local Bus**

Mode	Burst	Bterm	Result
Single Cycle	0	0	One ADS# per data (default).
Single Cycle	0	1	One ADS# per data.
Burst-4	1	0	One ADS# per four data (recommended for i960 and PPC401 family).
Burst Forever	1	1	One ADS# per BTERM# (refer to Section 4.2.5.2.1).

On the Local Bus, BLAST# and BTERM# perform the following:

- If the Burst Mode bit is enabled, but the Bterm Mode bit is disabled, then the PCI 9054 bursts (up to a quad word boundary) four Lwords. BLAST# is asserted at the beginning of the fourth Lword Data phase (LA[3:2]=11) and a new ADS# is asserted at the first Lword (LA[3:2]=00) of the next burst.

- If BTERM# is enabled and asserted, the PCI 9054 terminates the Burst cycle of the end of the current Data phase without generating BLAST#. The PCI 9054 generates a new burst transfer starting with a new ADS#, terminating it normally using BLAST#.
- BTERM# input is valid only when the PCI 9054 is Master of the Local Bus (PCI Target or DMA modes).
- As an input, BTERM# is asserted by external logic. It instructs the PCI 9054 to break up a Burst cycle.
- BTERM# is used to indicate a memory access is crossing a page boundary or requires a new Address cycle.

**Notes:** If Address Increment is disabled, the DMA transfer bursts beyond four Lwords.

If the Bterm Mode bit is disabled, the PCI 9054 performs the following:

- **32-bit Local Bus**—Bursts up to four Lwords
- **16-bit Local Bus**—Bursts up to two Lwords
- **8-bit Local Bus**—Bursts up to one Lword

In every case, it performs four transactions.

#### 4.2.5.2 Burst-4 Lword Mode

If the Burst Mode bit is enabled and the Bterm Mode bit is disabled, bursting can start on any Lword boundary and continue up to a 16-byte address boundary. After data up to the boundary is transferred, the PCI 9054 asserts a new Address cycle (ADS#).

**Table 4-9. Burst-4 Lword Mode**

Bus Width	Burst
32 bit	Four Lwords or up to a quad-Lword boundary (LA3, LA2 = 11)
16 bit	Four words or up to a quad-word boundary (LA2, LA1 = 11)
8 bit	Four bytes or up to a quad-byte boundary (LA1, LA0 = 11)

##### 4.2.5.2.1 Continuous Burst Mode (Bterm “Burst Terminate” Mode)

If both the Burst and Bterm Mode bits are enabled, the PCI 9054 can operate beyond the Burst-4 Lword mode.

Bterm mode enables PCI 9054 to perform long bursts to devices that can accept bursts of longer than four Lwords. The PCI 9054 asserts one Address cycle and continues to burst data. If a device requires a new Address cycle (ADS#), it can assert BTERM# input to cause the PCI 9054 to assert a new Address cycle. BTERM# input acknowledges current Data transfer and requests that a new Address cycle be asserted (ADS#). The new address is for the next Data transfer. If the Bterm Mode bit is

enabled and the BTERM# signal is asserted, the PCI 9054 asserts BLAST# only if its Read FIFO is full, its Write FIFO is empty, or if a transfer is complete.

#### 4.2.5.3 Partial Lword Accesses

Lword accesses in which not all byte enables asserted will be broken into Single-Cycle accesses. Burst start addresses can be any Lword boundary. If the Burst Start Address in a PCI Target or DMA transfer is not aligned to an Lword boundary, the PCI 9054 first performs a Single cycle. It then starts to burst on the Lword boundary if there is remaining data that is not a whole Lword during DMA (for example, it will result in a Single cycle at the end).

#### 4.2.6 Recovery States (J Mode Only)

In J mode, the PCI 9054 inserts one recovery state between the last Data transfer and the next Address cycle.

**Note:** The PCI 9054 does not support the i960J function that uses READY# input to add recovery states. No additional recovery states are added if READY# input remains asserted during the last Data cycle.

#### 4.2.7 Local Bus Read Accesses

For all Single-Cycle Local Bus Read accesses, the PCI 9054 reads only bytes corresponding to byte enables requested by the PCI Initiator. For all Burst Read cycles, the PCI 9054 passes all the bytes and can be programmed to:

- Prefetch
- Perform Read Ahead mode
- Generate internal wait states
- Enable external wait control (READY# input)
- Enable type of Burst mode to perform

#### 4.2.8 Local Bus Write Accesses

For Local Bus writes, only bytes specified by a PCI Bus Master or the PCI 9054 DMA controller are written.

#### 4.2.9 PCI Target Accesses to 8- or 16-Bit Local Bus

Direct PCI access to an 8- or 16-bit Local Bus results in the PCI Bus Lword being broken into multiple Local Bus transfers. For each transfer, byte enables are encoded as in the i960C to provide Local Address bits LA[1:0].

### 4.2.10 Local Bus Data Parity

Generation or use of Local Bus data parity is optional. Signals on the data parity pins do not affect operation of the PCI 9054. The PCI Bus parity checking and generation is independent of the Local Bus parity checking and generation. PCI Bus parity checking may result in assertion of PERR#, a PCI Bus system error (SERR#), or other means of PCI Bus transfer termination as a result of the parity error on the PCI data address, command code, and byte enables. The Local Bus Parity Check is passive and only provides parity information to the Local processor during PCI Initiator, PCI Target, and DMA transfers.

There is one data parity pin for each byte lane of the PCI 9054 data bus (DP[3:0]). “Even data parity” is asserted for each lane during Local Bus reads from the PCI 9054 and during PCI 9054 Master writes to the Local Bus.

Even data parity is checked during Local Bus writes to the PCI 9054 and during PCI 9054 reads from the Local Bus. Parity is checked for each byte lane with an asserted byte enable. If a parity error is detected, LSERR# is asserted in the Clock cycle following the data being checked.

Parity is checked for PCI Target reads, PCI Initiator writes, and DMA Local Bus reads. The PCI 9054 sets a status bit and asserts an interrupt (LSERR#) in the clock cycle following data being checked if a parity error is detected. However, the Data Parity Error Status bit and interrupt are never set or asserted unless the READY# signal is active and asserted low. This applies only when the READY# signal is disabled in the PCI 9054 register. A workaround for this is to disable the READY# Enable bit and externally pull READY# low.

## 4.3 BIG ENDIAN/LITTLE ENDIAN

### 4.3.1 PCI Bus Little Endian Mode

PCI Bus is a Little Endian bus (*that is*, the address is invariant and data is Lword-aligned to the lowermost byte lane).

Table 4-10. PCI Bus Little Endian Byte Lanes

Byte Number	Byte Lane
0	AD[7:0]
1	AD[15:8]
2	AD[23:16]
3	AD[31:24]

### 4.3.2 Local Bus Big/Little Endian Mode

The PCI 9054 Local Bus can be programmed to operate in Big or Little Endian mode.

Table 4-11. Byte Number and Lane Cross-Reference

Mode	Byte Number		Byte Lane
	Big Endian	Little Endian	
C	3	0	LD[7:0]
	2	1	LD[15:8]
	1	2	LD[23:16]
	0	3	LD[31:24]
J	3	0	LAD[7:0]
	2	1	LAD[15:8]
	1	2	LAD[23:16]
	0	3	LAD[31:24]

Table 4-12. Big/Little Endian Program Mode

BIGEND# Pin	BIGEND Register (1=Big, 0=Little)	Endian Mode
0	0	Big
0	1	Big
1	0	Little
1	1	Big

Table 4-13 lists registers for information about the following cycles.

Table 4-13. Cycles Reference Tables

Cycles	Register Bits
Local access to the Configuration registers	BIGEND[0]
PCI Initiator, Memory, and I/O	BIGEND[1]
PCI Target	BIGEND[2], Space 0, and BIGEND[3], Expansion ROM

In Big Endian mode, the PCI 9054 transposes data byte lanes. Data is transferred as listed in Table 4-14 through Table 4-19.

### 4.3.2.1 32-Bit Local Bus—Big Endian Mode

Data is Lword aligned to uppermost byte lane (Data Invariance).

Table 4-14. Upper Lword Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [31:24]
	Byte 1 appears on Local Data [23:16]
	Byte 2 appears on Local Data [15:8]
	Byte 3 appears on Local Data [7:0]

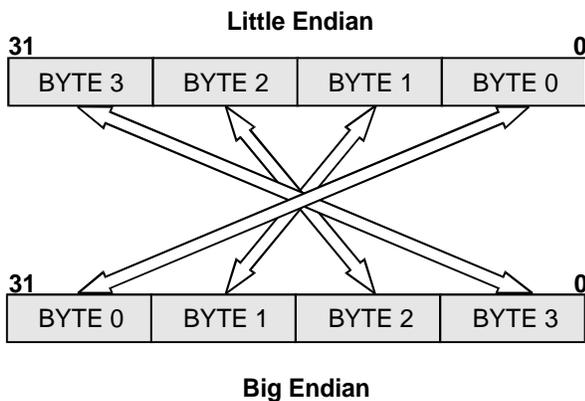


Figure 4-2. Big/Little Endian—32-Bit Local Bus

### 4.3.2.2 16-Bit Local Bus—Big Endian Mode

For a 16-bit Local Bus, the PCI 9054 can be programmed to use upper or lower word lanes.

Table 4-15. Upper Word Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [31:24]
	Byte 1 appears on Local Data [23:16]
Second transfer	Byte 2 appears on Local Data [31:24]
	Byte 3 appears on Local Data [23:16]

Table 4-16. Lower Word Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [15:8]
	Byte 1 appears on Local Data [7:0]
Second transfer	Byte 2 appears on Local Data [15:8]
	Byte 3 appears on Local Data [7:0]

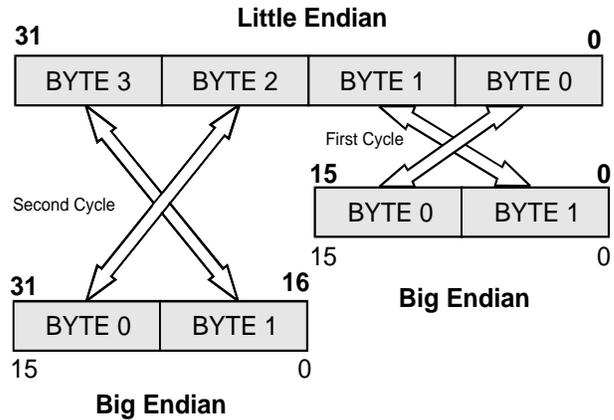


Figure 4-3. Big/Little Endian—16-Bit Local Bus

### 4.3.2.3 8-Bit Local Bus—Big Endian Mode

For an 8-bit Local Bus, the PCI 9054 can be programmed to use upper or lower byte lanes.

Table 4-17. Upper Byte Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [31:24]
Second transfer	Byte 1 appears on Local Data [31:24]
Third transfer	Byte 2 appears on Local Data [31:24]
Fourth transfer	Byte 3 appears on Local Data [31:24]

Table 4-18. Lower Byte Lane Transfer

Burst Order	Byte Lane
First transfer	Byte 0 appears on Local Data [7:0]
Second transfer	Byte 1 appears on Local Data [7:0]
Third transfer	Byte 2 appears on Local Data [7:0]
Fourth transfer	Byte 3 appears on Local Data [7:0]

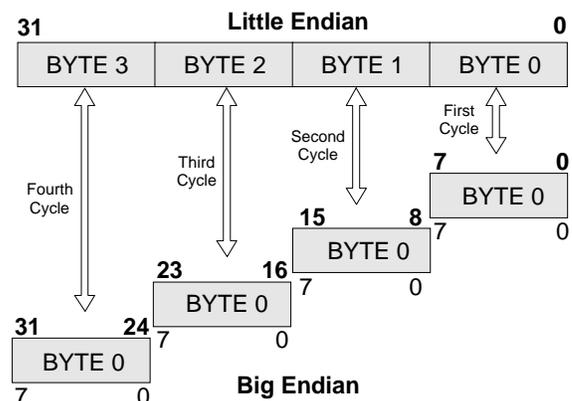


Figure 4-4. Big/Little Endian—8-Bit Local Bus

### 4.3.2.4 Local Bus Big/Little Endian Mode Accesses

For each of the following transfer types, the PCI 9054 Local Bus can be independently programmed to operate in Little Endian or Big Endian mode:

- Local Bus accesses to the PCI 9054 Configuration registers
- PCI Target PCI accesses to Local Address Space 0
- PCI Target PCI accesses to Local Address Space 1
- PCI Target PCI accesses to the Expansion ROM
- DMA Channel 0 accesses to the Local Bus
- DMA Channel 1 accesses to the Local Bus
- PCI Initiator accesses to the PCI Bus

For Local Bus accesses to the Internal Configuration registers and PCI Initiator accesses, use BIGEND# to dynamically change the Endian mode.

**Notes:** *The PCI Bus is always Little Endian. Only byte lanes are swapped, not individual bits.*

## 4.4 SERIAL EEPROM

Functional operation described can be modified through the PCI 9054 programmable internal registers.

### 4.4.1 Vendor and Device ID Registers

Three Vendor and Device ID registers are supported:

- **PCIIDR**—Contains normal Device and Vendor IDs. Can be loaded from the serial EEPROM or Local processor(s).
- **PCISVID**—Contains Subsystem and Subvendor IDs. Can be loaded from the serial EEPROM or Local processor(s).
- **PCIHIDR**—Contains hardcoded PLX Vendor and Device IDs.

#### 4.4.1.1 Serial EEPROM Initialization

During serial EEPROM initialization, the PCI 9054 responds to PCI Target accesses with a Retry. During serial EEPROM initialization, the PCI 9054 responds to a Local processor access by delaying acknowledgment of the cycle (READY#).

### 4.4.1.2 Local Initialization

The PCI 9054 issues a Retry to all PCI accesses until the Local Init Status bit (LMISC[2]) is set.

This bit can be programmed three different ways:

1. By the Local processor, through the Local Configuration register.
2. By the serial EEPROM, during a serial EEPROM load, if the Local processor does not set this bit or if this bit is missing.
3. If the Local processor and/or the serial EEPROM are missing, the serial EEPROM remains blank and the PCI 9054 reverts to the default values and sets this bit (refer to Table 4-19. on page 4-8).

### 4.4.2 Serial EEPROM Operation

After reset, the PCI 9054 attempts to read the serial EEPROM to determine its presence. An active Start bit set to 0 indicates a serial EEPROM is present. The PCI 9054 supports 93CS56L (2K bits) or 93CS66L (4K bits). (Refer to manufacturer's data sheet for the particular serial EEPROM being used.) The first Lword is then checked to verify that the serial EEPROM is programmed. If the first Lword (33 bits) is all ones, a blank serial EEPROM is present. If the first Lword (33 bits) is all zeros, no serial EEPROM is present. For both conditions, the PCI 9054 reverts to the default values. (Refer to Table 4-19.) CNTRL[28] is set to 1 if programmed (real or random data if a serial EEPROM is detected).

The 3.3V serial EEPROM clock (EESK) is derived from the PCI clock. The PCI 9054 generates the serial EEPROM clock by internally dividing the PCI clock by 132.

The serial EEPROM can be read or written from the PCI or Local Buses. The Serial EEPROM Control Register bits (CNTRL[28:24]) control the PCI 9054 pins that enable reading or writing of serial EEPROM data bits. (Refer to manufacturer's data sheet for the particular serial EEPROM being used.)

The PCI 9054\_AB revision, provides the ability to manually access the serial EEPROM. This may be accomplished by using bits 24 through 27 of the CNTRL register (EESK, EECS, and EEDI/EEDO controlled by software). Bit 24 is used to generate EESK (clock), bit 25 controls the chip select, and bit 26 sets/clears EEDO. Bit 27, when read, returns the value of EEDI.

Setting bits 24, 25, and 26 to 1, causes the output to go high. A pull-up resistor is required on EEDO to go high when bit 26 is set. When reading the EEPROM, bit 26 must be set to a 1.

To perform the read, the basic approach is to set the EECS and EEDO bits (bits 25 and 26) to the desired level and then toggle EESK high and low until done. *For example*, reading the serial EEPROM at location 0 involves the following steps:

1. Clear EESK, EEDO and EECS bits.
2. Toggle EESK high, then low.
3. Set EECS high.
4. Toggle EESK high, then low.
5. Set EEDO bit high (start bit).
6. Toggle EESK high, then low.
7. Repeat step 6.
8. Clear EEDO.
9. Toggle EESK bit high, then low eight times (clock in Local Address 0).
10. Set EEDO to float the EEDO pin for reading.
11. Toggle EESK high, then low 16 times (clock in one word from serial EEPROM).
12. After each clock pulse, read bit 27 and save.
13. Clear EECS bit.
14. Toggle EESK high, then low.
15. Read is now complete.

The serial EEPROM can also be read or written, using the VPD function (refer to Section 10).

The PCI 9054 has two serial EEPROM load options:

- **Long Load Mode**—Default. The PCI 9054 loads 17 Lwords from the Serial EEPROM and the Extra Long Load bit (LBRD0[25])
- **Extra Long Load Mode**—The PCI 9054 loads 22 Lwords if the Serial EEPROM and the Extra Long Load bit (LBRD0[25]) is set to 1 during a Long Load

Table 4-19. Serial EEPROM Guidelines

Local Processor	Serial EEPROM	System Boot Condition
None	None	The PCI 9054 uses default values. The EEDI/EEDO pin must be <b>pulled low</b> —a 1 K-ohm resistor is required (rather than pulled high, which is typically done for this pin). If the PCI 9054 detects all zeros, it reverts to default values.
None	Programmed	Boot with serial EEPROM values. The Local Init Status bit (LMISC[2]) must be set by the serial EEPROM.
None	Blank	The PCI 9054 detects a blank device and reverts to default values.
Present	None	The Local processor programs the PCI 9054 registers, then sets the Local Init Status bit (LMISC[2]=done). <i>Note: Some systems may hang if PCI Target reads and writes take too long (during initialization, the PCI Host also performs PCI Target accesses). The value of the PCI Target Retry Delay Clocks (LBRD0[31:28]) may resolve this.</i>
Present	Programmed	Load serial EEPROM, but the Local processor can reprogram the PCI 9054. Either the Local processor or the serial EEPROM must set the Local Init Status bit (LMISC[2]=done).
Present	Blank	The PCI 9054 detects a blank serial EEPROM and reverts to default values. <i>Notes: In some systems, the Local processor may be too late to reconfigure the PCI 9054 registers before the BIOS configures them. The serial EEPROM can be programmed through the PCI 9054 after the system boots in this condition.</i>

#### 4.4.2.1 Long Serial EEPROM Load

The registers listed in Table 4-20 are loaded from the serial EEPROM after a reset is de-asserted if the Serial EEPROM Extra Long Load bit is not set (LBRD0[25]=0). The serial EEPROM is organized in words (16 bit). The PCI 9054 first loads the Most Significant Word bits (MSW[31:16]), starting from the most significant bit [31]. The PCI 9054 then loads the Least Significant Word bits (LSW[15:0]), starting again from the most significant bit [15]. Therefore, the PCI 9054 loads the Device ID, Vendor ID, class code, and so forth.

The serial EEPROM values can be programmed using a Data I/O programmer. The values can also be programmed using the PCI 9054 VPD function (refer to Section 10) or through the Serial EEPROM Control register (CNTRL).

The CNTRL register allows programming of the serial EEPROM, one bit at a time. To read back the value from the serial EEPROM, the CNTRL[27] bit (refer to

Section 4.4.2) or the VPD function should be utilized. With full utilization of VPD, the designer can perform reads and writes from/to the serial EEPROM, 32 bits at a time. Values should be programmed in the order listed in Table 4-20. The 34, 16-bit words listed in the table are stored sequentially in the serial EEPROM.

**Table 4-20. Long Serial EEPROM Load Registers**

Serial EEPROM Offset	Description	Register Bits Affected
0h	Device ID	PCIIDR[31:16]
2h	Vendor ID	PCIIDR[15:0]
4h	Class Code	PCICCR[23:8]
6h	Class Code / Revision	PCICCR[7:0] / PCIREV[7:0]
8h	Maximum Latency / Minimum Grant	PCIMLR[7:0] / PCIMGR[7:0]
Ah	Interrupt Pin / Interrupt Line Routing	PCIIPR[7:0] / PCIILR[7:0]
Ch	MSW of Mailbox 0 (User Defined)	MBOX0[31:16]
Eh	LSW of Mailbox 0 (User Defined)	MBOX0[15:0]
10h	MSW of Mailbox 1 (User Defined)	MBOX1[31:16]
12h	LSW of Mailbox 1 (User Defined)	MBOX1[15:0]
14h	MSW of Range for PCI-to-Local Address Space 0	LASORR[31:16]
16h	LSW of Range for PCI-to-Local Address Space 0	LASORR[15:0]
18h	MSW of Local Base Address (Remap) for PCI-to-Local Address Space 0	LASOBA[31:16]
1Ah	LSW of Local Base Address (Remap) for PCI-to-Local Address Space 0	LASOBA[15:0]
1Ch	MSW of Mode/DMA Arbitration Register	MARBR[31:16]
1Eh	LSW of Mode/DMA Arbitration Register	MARBR[15:0]
20h	MSW of Serial EEPROM Write-Protected Address	PROT_AREA[15:0]
22h	LSW of Local Miscellaneous Control Register / LSW of Local Bus Big/Little Endian Descriptor Register	LMISC[7:0] / BIGEND[7:0]
24h	MSW of Range for PCI-to-Local Expansion ROM	EROMRR[31:16]
26h	LSW of Range for PCI-to-Local Expansion ROM	EROMRR[15:0]
28h	MSW of Local Base Address (Remap) for PCI-to-Local Expansion ROM	EROMBA[31:16]
2Ah	LSW of Local Base Address (Remap) for PCI-to-Local Expansion ROM	EROMBA[15:0]
2Ch	MSW of Bus Region Descriptors for PCI-to-Local Accesses	LBRD0[31:16]
2Eh	LSW of Bus Region Descriptors for PCI-to-Local Accesses	LBRD0[15:0]
30h	MSW of Range for PCI Initiator-to-PCI	DMRR[31:16]
32h	LSW of Range for PCI Initiator-to-PCI	DMRR[15:0]
34h	MSW of Local Base Address for PCI Initiator-to-PCI Memory	DMLBAM[31:16]
36h	LSW of Local Base Address for PCI Initiator-to-PCI Memory	DMLBAM[15:0]
38h	MSW of Local Bus Address for PCI Initiator-to-PCI I/O Configuration	DMLBAI[31:16]
3Ah	LSW of Local Bus Address for PCI Initiator-to-PCI I/O Configuration	DMLBAI[15:0]
3Ch	MSW of PCI Base Address (Remap) for PCI Initiator-to-PCI	DMPBAM[31:16]
3Eh	LSW of PCI Base Address (Remap) for PCI Initiator-to-PCI	DMPBAM[15:0]
40h	MSW of PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration	DMCFGA[31:16]
42h	LSW of PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration	DMCFGA[15:0]

### 4.4.2.2 Extra Long Serial EEPROM Load

The registers listed in the Local Address Space 0/Expansion ROM Bus Region Descriptor register (LBRD0) are loaded from serial EEPROM after a reset is de-asserted if the Serial EEPROM Extra Long Load bit is set (LBRD0[25]=1). The serial EEPROM is organized in words (16 bit). The PCI 9054 first loads the Most Significant Word bits [31:16], starting from the most significant bit [31]. It then loads the Least Significant Word bits [15:0], restarting from the most

significant bit [15]. Therefore, the PCI 9054 loads Device ID, Vendor ID, class code, and so forth.

The serial EEPROM values can be programmed using a Data I/O programmer. The values can also be programmed using the PCI 9054 VPD function or through the Serial EEPROM Control register (CNTRL). Values should be programmed in the order listed in Table 4-21. The 44 16-bit words listed in Table 4-20 and Table 4-21 should be stored sequentially in the serial EEPROM.

Table 4-21. Extra Long Serial EEPROM Load Registers

Serial EEPROM Offset	Description	Register Bits Affected
44h	Subsystem ID	PCISID[15:0]
46h	Subsystem Vendor ID	PCISVID[15:0]
48h	MSW of Range for PCI-to-Local Address Space 1 (1 MB)	LAS1RR[31:16]
4Ah	LSW of Range for PCI-to-Local Address Space 1 (1 MB)	LAS1RR[15:0]
4Ch	MSW of Local Base Address (Remap) for PCI-to-Local Address Space 1	LAS1BA[31:16]
4Eh	LSW of Local Base Address (Remap) for PCI-to-Local Address Space 1	LAS1BA[15:0]
50h	MSW of Bus Region Descriptors (Space 1) for PCI-to-Local Accesses	LBRD1[31:16]
52h	LSW of Bus Region Descriptors (Space 1) for PCI-to-Local Accesses	LBRD1[15:0]
54h	MSW of Hot Swap Control	<b>Reserved</b>
56h	LSW of Hot Swap Control / Hot Swap Next Capability Pointer	HS_NEXT[7:0] / HS_CNTL[7:0]

### 4.4.2.3 New Capabilities Function Support

The New Capabilities Function Support includes PCI Power Management, Hot Swap, and VPD features, as listed in Table 4-22.

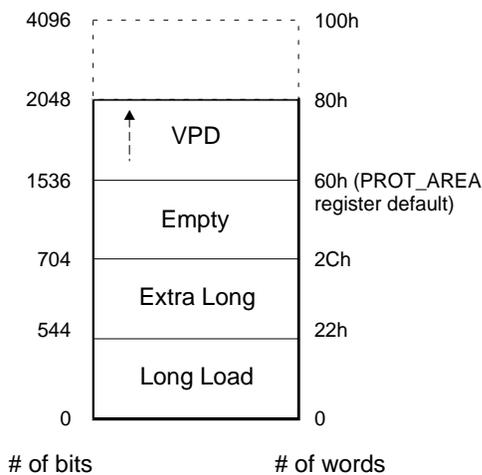
**Table 4-22. New Capabilities Function Support Features**

New Capability Function	PCI Register Offset Location
First (Power Management)	40'h, if the New Capabilities Function Support bit (PCISR[4]) is enabled (PCISR[4] is enabled, by default).
Second (Hot Swap)	48'h, which is pointed to from PMNEXT[7:0].
Third (VPD)	4C'h, which is pointed to from HS_NEXT[7:0]. Because PVPD_NEXT[7:0] defaults to zero, this indicates that VPD is the last New Capability Function Support feature of the PCI 9054.

### 4.4.2.4 Recommended Serial EEPROMs

The PCI 9054 is designed to use either a 2K bit (NM93CS56L or compatible) or 4K bit (NM93CS66L or compatible) device.

**Note:** The PCI 9054 does not support serial EEPROMs that do not support sequential reads and writes (such as the NM93C56L).



**Figure 4-5. Serial EEPROM Memory Map**

### 4.4.2.5 Serial EEPROM Initialization

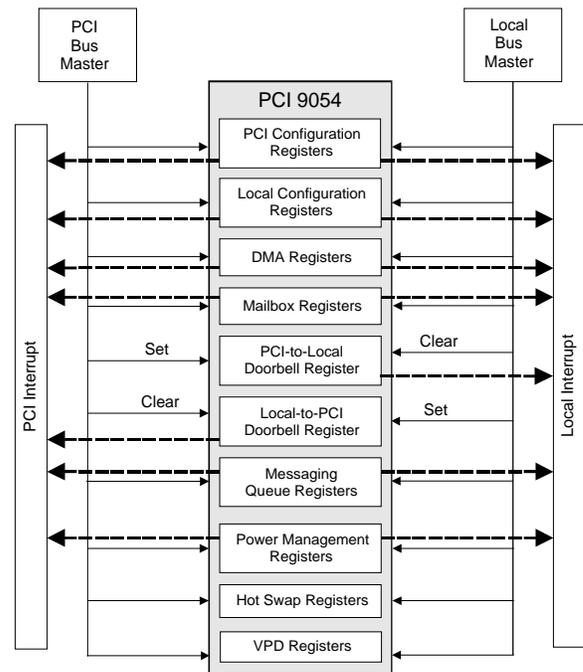
During serial EEPROM initialization, the PCI 9054 responds to PCI Target accesses with a Retry. During serial EEPROM initialization, the PCI 9054 responds to a Local processor access by delaying acknowledgment of the cycle (READY#).

### 4.4.3 Internal Register Access

The PCI 9054 provides several internal registers, which allow for maximum flexibility in the bus-interface design and performance. These registers are accessible from the PCI and Local Buses and include the following:

- PCI and Local Configuration registers
- DMA registers
- Mailbox registers
- PCI-to-Local and Local-to-PCI Doorbell registers
- Messaging Queue registers (I<sub>2</sub>O)
- Power Management registers
- Hot Swap registers
- VPD registers

Figure 4-6 illustrates how these registers are accessed.



**Figure 4-6. PCI 9054 Internal Register Access**

Section 4—C, J Bus Op

### 4.4.3.1 PCI Bus Access to Internal Registers

The PCI 9054 PCI Configuration registers can be accessed from the PCI Bus with a Configuration Type 0 cycle.

All other PCI 9054 internal registers can be accessed by a Memory cycle, with the PCI Bus address that matches the base address specified in PCI Base Address 0 (PCIBAR0[31:8]) for the PCI 9054 Memory-Mapped Configuration register. These registers can also be accessed by an I/O cycle, with the PCI Bus address matching the base address specified in PCI Base Address 1 for the PCI 9054 I/O-Mapped Configuration register.

All PCI Read or Write accesses to the PCI 9054 registers can be Byte, Word, or Lword accesses. All PCI Memory accesses to the PCI 9054 registers can be Burst or Non-Burst accesses. The PCI 9054 responds with a PCI disconnect for all Burst I/O accesses (PCIBAR1[31:8]) to the PCI 9054 Internal registers.

### 4.4.3.2 Local Bus Access to Internal Registers

The Local processor can access all PCI 9054 internal registers through an external chip select. The PCI 9054 responds to a Local Bus access when the PCI 9054 Configuration Chip Select input (CCS#) is asserted low. Figure 4-7 illustrates how the Configuration Chip Select logic works.

**Notes:** CCS# must be decoded while ADS# is low. Accesses must be for a 32-bit non-pipelined bus.

Local Read or Write accesses to the PCI 9054 internal registers can be Byte, Word, or Lword accesses. Local accesses to the PCI 9054 internal registers can be Burst or Non-Burst accesses.

The PCI 9054 READY# signal indicates that Data transfer is complete.

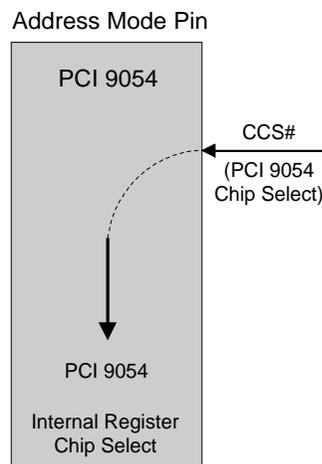
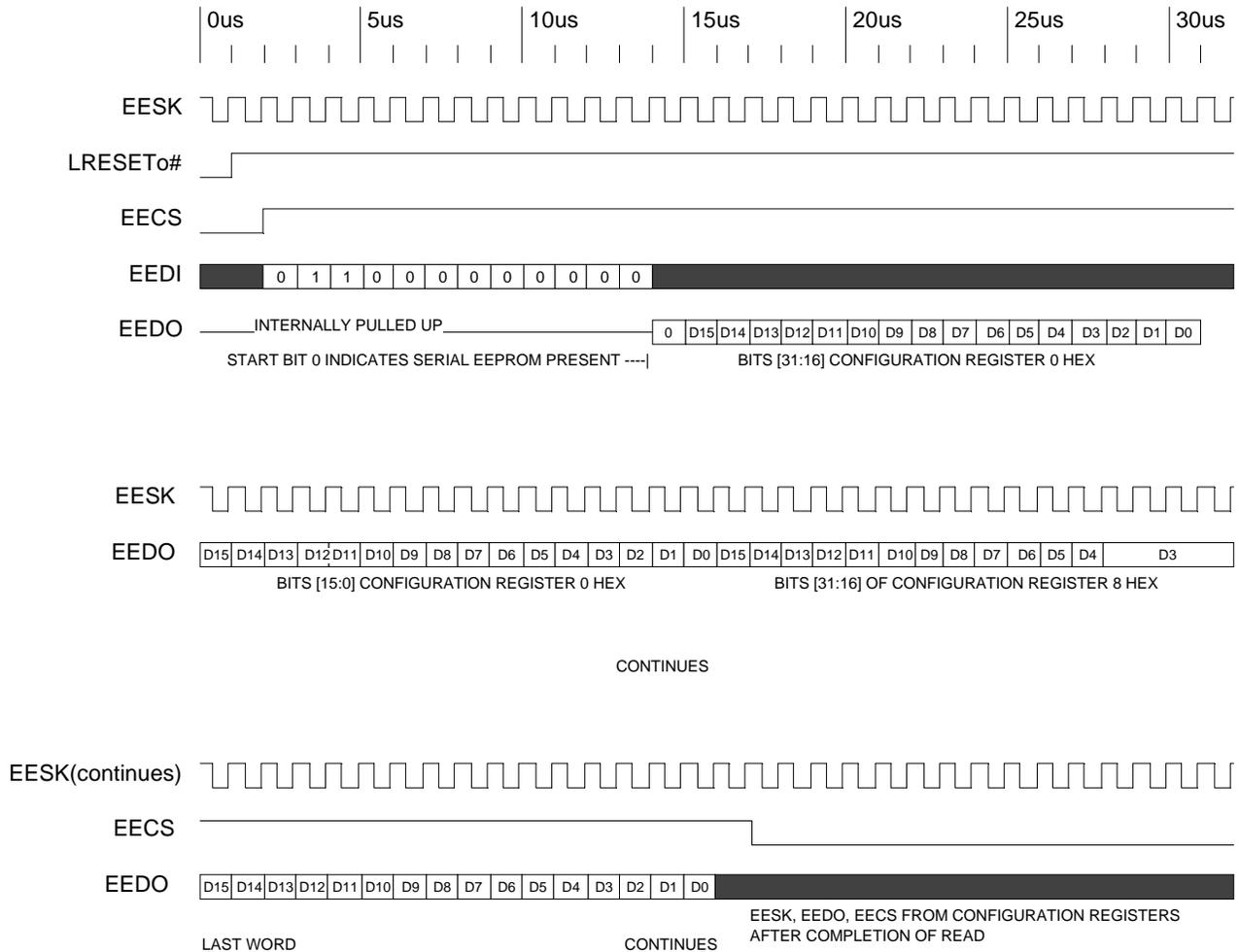


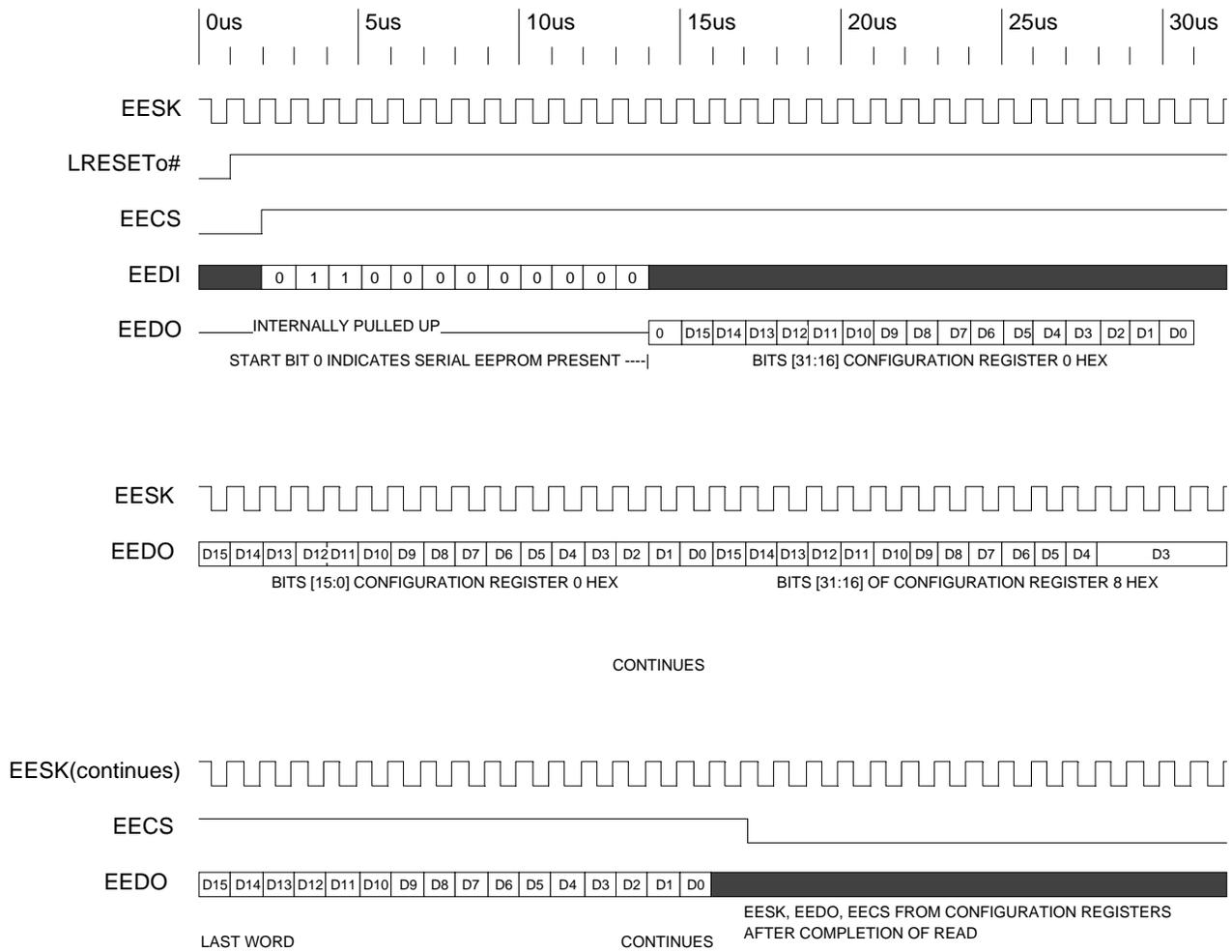
Figure 4-7. Address Decode Mode

### 4.4.4 Serial EEPROM Timing Diagrams

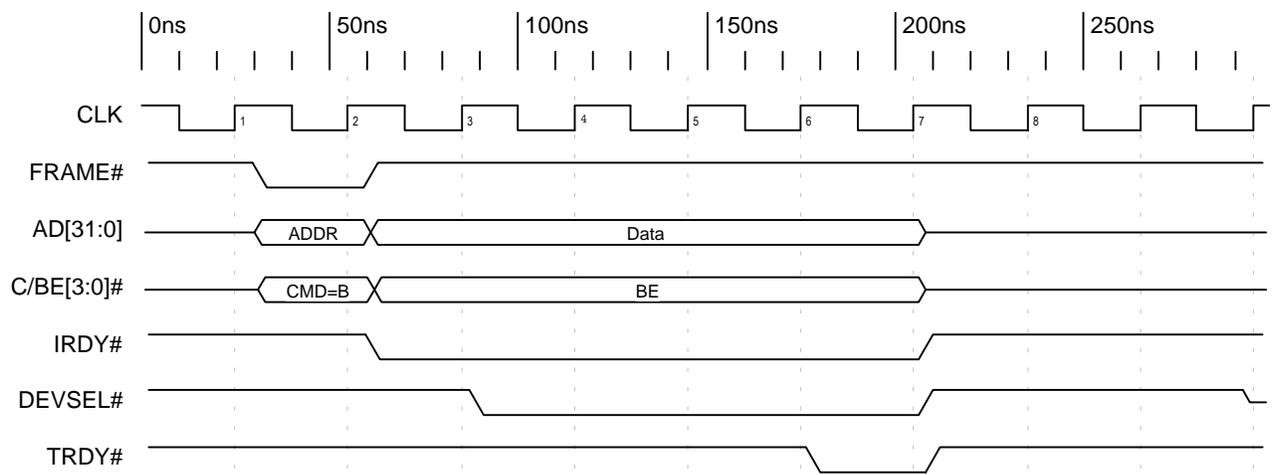


Timing Diagram 4-1. Initialization from Serial EEPROM (2K Bit)

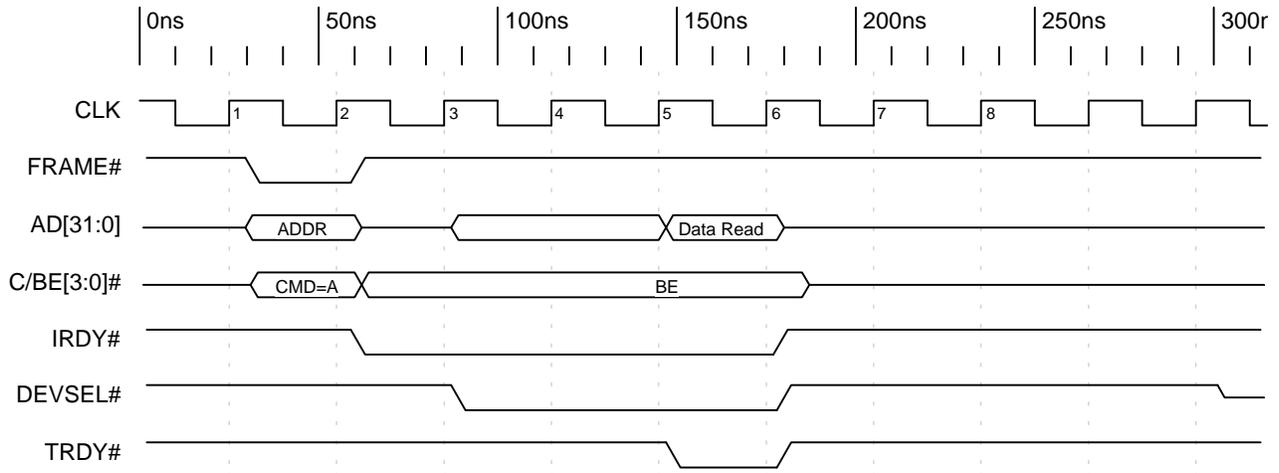
Section 4—C, J Bus Op



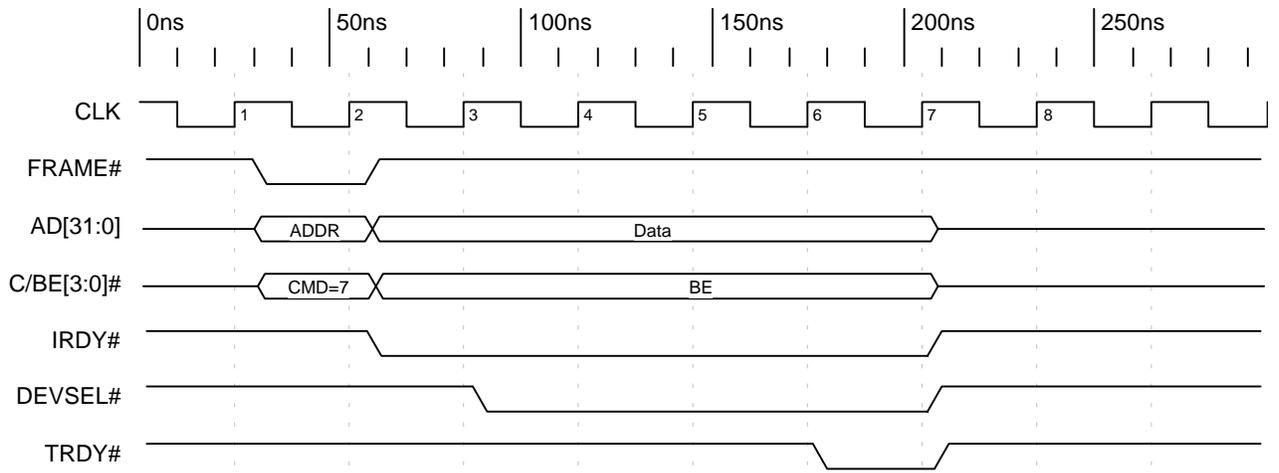
Timing Diagram 4-2. Initialization from Serial EEPROM (4K Bit)



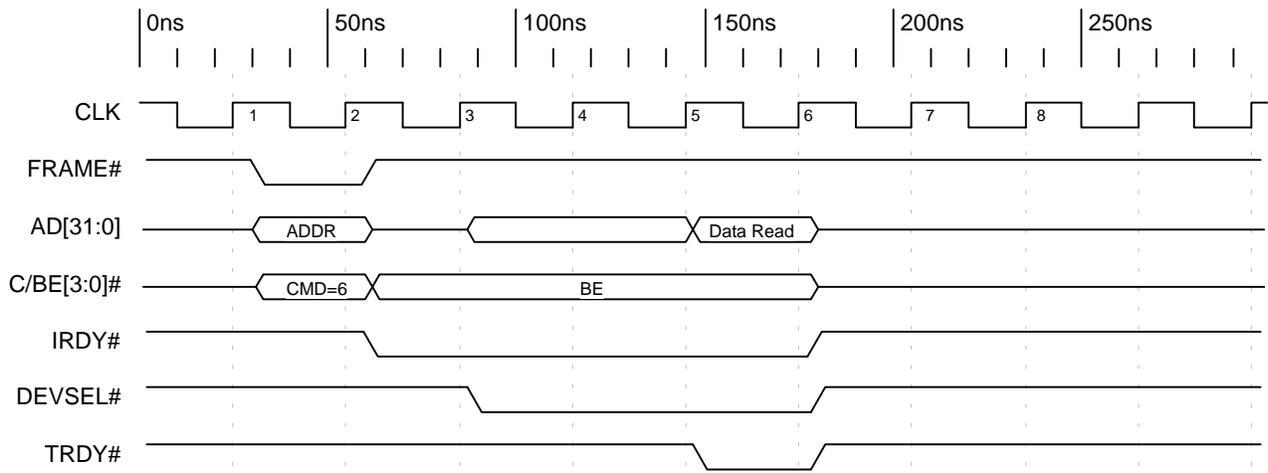
Timing Diagram 4-3. PCI Configuration Write to PCI Configuration Register



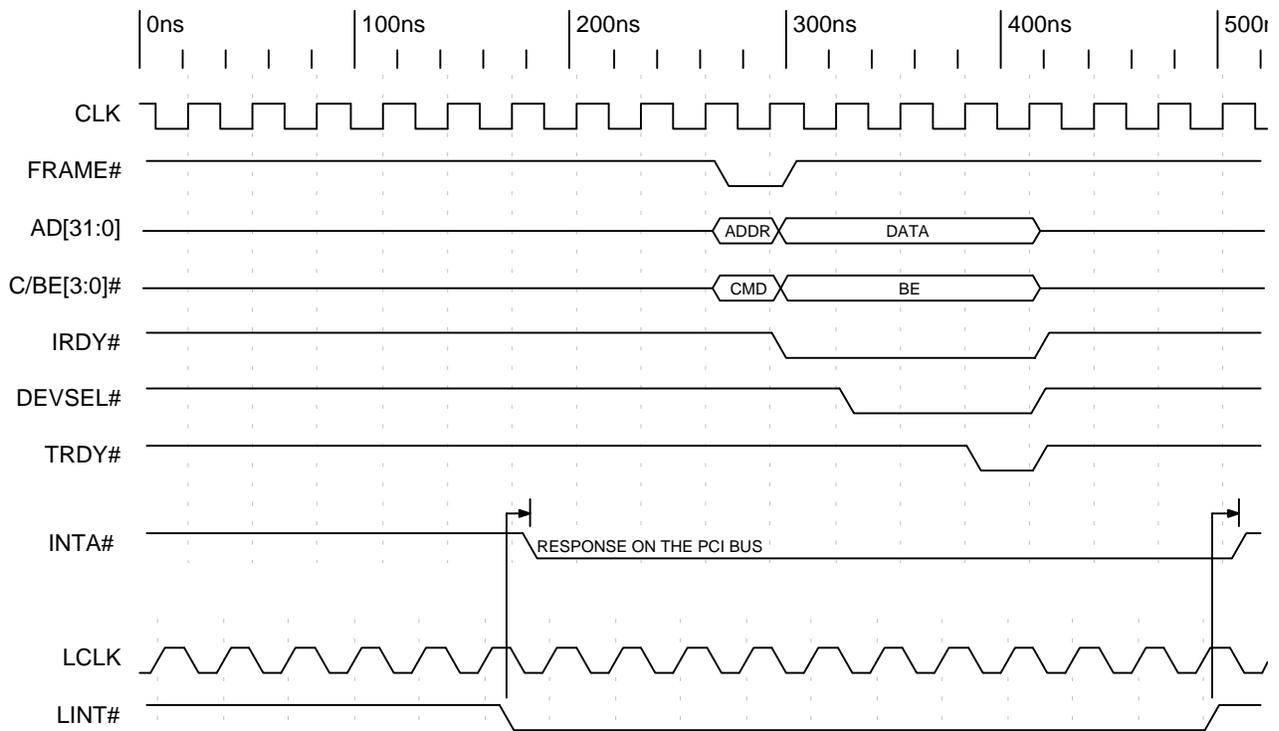
Timing Diagram 4-4. PCI Configuration Read to PCI Configuration Register



Timing Diagram 4-5. PCI Memory Write to Local Configuration Register



Timing Diagram 4-6. PCI Memory Read to Local Configuration Register



Timing Diagram 4-7. Local Interrupt Asserting PCI Interrupt

# 5 C AND J MODES FUNCTIONAL DESCRIPTION

The functional operation described can be modified through the PCI 9054 programmable internal registers.

## 5.1 RESET OPERATION

### 5.1.1 PCI Bus Input RST#

PCI Bus RST# input pin is a PCI Host reset. It causes all PCI Bus outputs to float, resets the entire PCI 9054 and causes the Local reset LRESETo# signal to be asserted.

### 5.1.2 Software Reset

A Host on the PCI Bus can set the PCI Adapter Software Reset bit (CNTRL[30]=1) to reset the PCI 9054 and assert LRESETo# output. All Local Configuration registers are reset; however, the PCI Configuration DMA and Shared Runtime registers and the Local Init Status bit (LMISC[2]) are not reset. When the Software Reset bit (CNTRL[30]) is set, the PCI 9054 responds to PCI accesses, but not to Local Bus accesses. The PCI 9054 remains in this reset condition until the PCI Host clears the bit. The serial EEPROM is reloaded if the Reload Configuration Registers bit is set (CNTRL[29]=1).

**Note:** *The Local Bus cannot clear this reset bit because the Local Bus is in a reset state, even if the Local processor does not use LRESETo# to reset.*

## 5.2 PCI 9054 INITIALIZATION

The PCI 9054 Configuration registers can be programmed by an optional serial EEPROM and/or by a Local processor, as listed in Table 4-19. The serial EEPROM can be reloaded by setting the Reload Configuration Registers bit (CNTRL[29]).

The PCI 9054 retries all PCI cycles until the Local Init Status bit is set to “done” (LMISC[2]=1).

**Note:** *The PCI Host processor can also access Internal Configuration registers after the Local Init Status bit is set.*

If a PCI Host is present, the Master Enable, Memory Space, and I/O Space bits (PCICR[2:0]) are programmed by that Host after initialization completes (LMISC[2]=1).

## 5.3 RESPONSE TO FIFO FULL OR EMPTY

Table 5-1 lists the response of the PCI 9054 to full and empty FIFOs.

## 5.4 DIRECT DATA TRANSFER MODES

The PCI 9054 supports three direct transfer modes:

- **PCI Initiator**—Local CPU accesses PCI memory or I/O
- **PCI Target**—PCI Master accesses Local memory or I/O
- **DMA**—PCI 9054 DMA controller reads/writes PCI memory to/from Local memory

### 5.4.1 PCI Initiator Operation (Local Master-to-PCI Target)

The PCI 9054 supports a direct access of the PCI Bus by the Local processor or an intelligent controller. Master mode must be enabled in the PCI Command register. The following registers define Local-to-PCI accesses:

- PCI Initiator Memory and I/O Range (DMRR)
- Local Base Address for PCI Initiator to PCI Memory (DMLBAM)
- Local Base Address for PCI Initiator to PCI I/O and Configuration (DMLBAI)
- PCI Base Address (DMPBAM)
- PCI Initiator Configuration (DMCFGA)
- PCI Initiator PCI Dual Address Cycles (DMDAC)
- Master Enable (PCICR)
- PCI Command Code (CNTRL)

Table 5-1. Response to FIFO Full or Empty

Mode	Direction	FIFO	PCI Bus	Local Bus
PCI Initiator Write	Local-to-PCI	Full	Normal	De-assert READY#
		Empty	De-assert REQ# (off PCI Bus)	Normal
PCI Initiator Read	PCI-to-Local	Full	De-assert REQ# or throttle IRDY# <sup>1</sup>	Normal
		Empty	Normal	De-assert READY#
PCI Target Write	PCI-to-Local	Full	Disconnect or throttle TRDY# <sup>2</sup>	Normal
		Empty	Normal	De-assert LHOLD, assert BLAST# <sup>3</sup>
PCI Target Read	Local-to-PCI	Full	Normal	De-assert LHOLD, assert BLAST# <sup>3</sup>
		Empty	Throttle TRDY# <sup>2</sup>	Normal
DMA	Local-to-PCI	Full	Normal	De-assert LHOLD, assert BLAST# <sup>3</sup>
		Empty	De-assert REQ#	Normal
	PCI-to-Local	Full	De-assert REQ#	Normal
		Empty	Normal	De-assert LHOLD, assert BLAST# <sup>3</sup>

<sup>1</sup> Throttle IRDY# depends on the PCI Initiator PCI Read Mode bit (DMPBAM[4]).

<sup>2</sup> Throttle TRDY# depends on the PCI Target Write Mode bit (LBRD0[27]).

<sup>3</sup> LHOLD de-assert depends upon the Local Bus PCI Target Release Bus Mode bit (MARBR[21]).

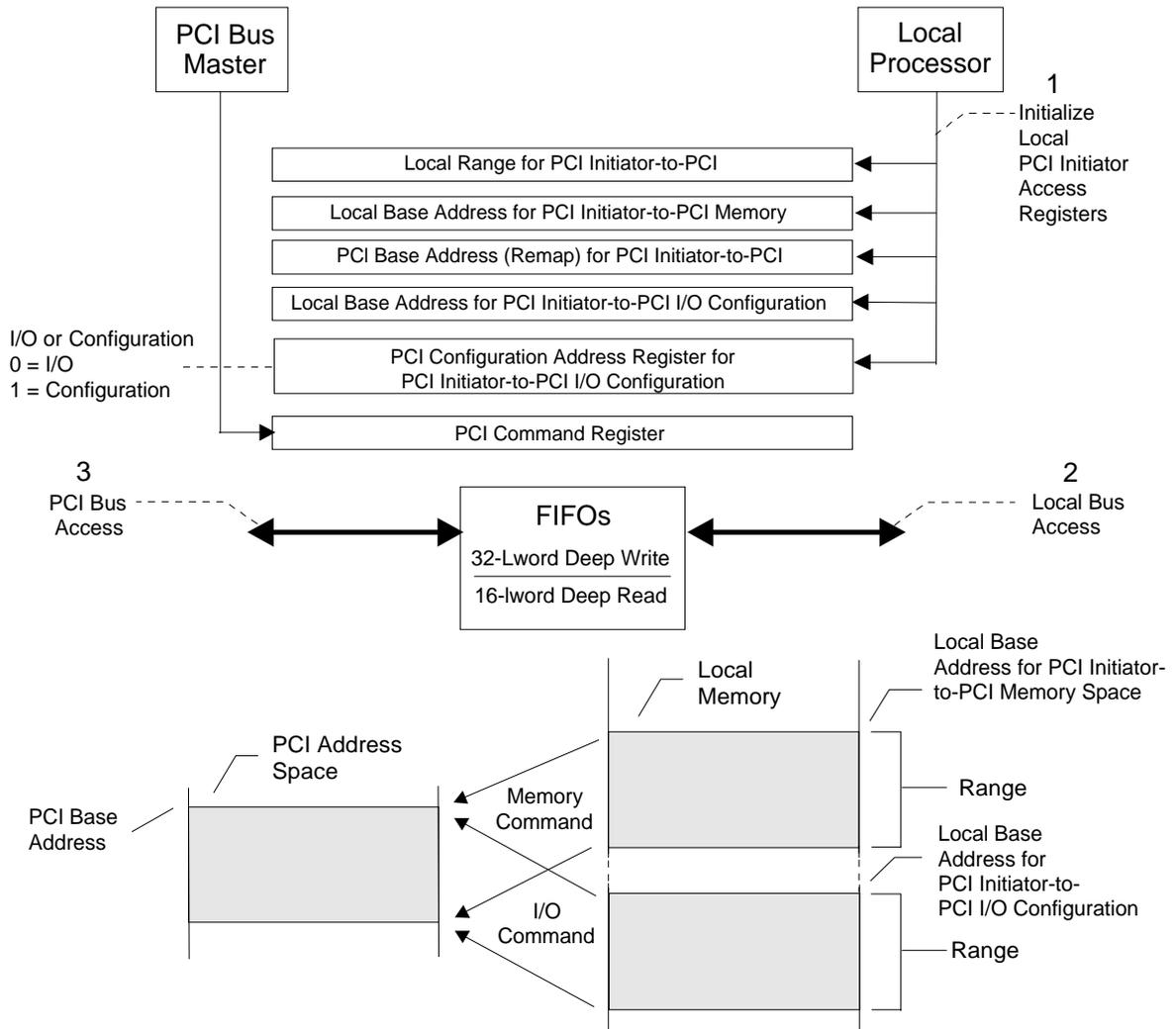


Figure 5-1. PCI Initiator Access of the PCI Bus

### 5.4.1.1 PCI Initiator Memory and I/O Decode

The Range register and the Local Base Address specifies the Local Address bits to use for decoding a Local-to-PCI access (PCI Initiator). The range of memory or I/O space must be a power of 2 and the Range register value must be the inverse of the Range value. In addition, the Local Base Address must be a multiple of the range value.

Any Local Master Address starting from the PCI Initiator Local Base Address (Memory or I/O) to the range value is recognized as a PCI Initiator access by the PCI 9054. All PCI Initiator cycles are then decoded as PCI Memory, I/O, or Configuration Type 0 or 1. Moreover, a PCI Initiator memory or I/O cycle is remapped according to the Remap register value. The Remap Register value must be a multiple of the PCI Initiator Range value (not the Range register value).

The PCI 9054 can only accept Memory cycles from the Local processor. The Local Base Address and/or the range determine whether PCI Memory or PCI I/O transactions occur.

### 5.4.1.2 PCI Initiator FIFOs

For PCI Initiator Memory access to the PCI Bus, the PCI 9054 has a 32-Lword (128-byte) Write FIFO and a 16-Lword (64-byte) Read FIFO. The FIFOs enable the Local Bus to operate independent of the PCI Bus and allows high-performance bursting on the PCI and Local Buses. In a PCI Initiator write, the Local processor (Master) writes data to the PCI Bus (Slave). In a PCI Initiator read, the Local processor (Master) reads data from the PCI Bus (Slave). The FIFOs that function during a PCI Initiator write and read are illustrated in Figure 5-2 and Figure 5-3.

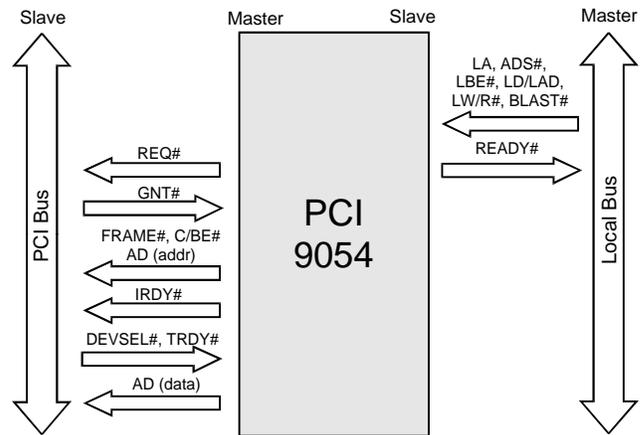


Figure 5-2. PCI Initiator Write

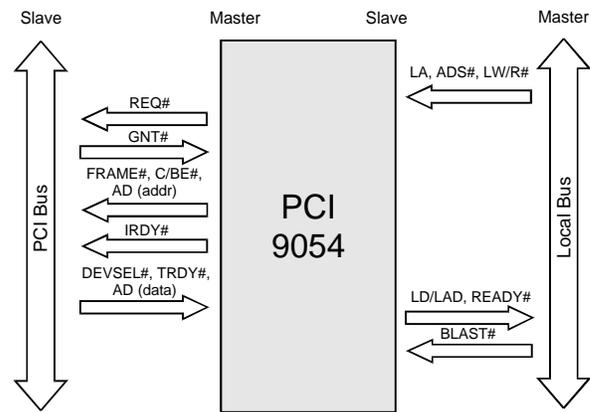


Figure 5-3. PCI Initiator Read

**Note:** The figures represent a sequence of Bus cycles.

### 5.4.1.3 PCI Initiator Memory Access

The Local processor can read or write to the PCI memory. The PCI 9054 converts the Local Read/Write access. The Local Address space starts from the PCI Initiator Local Base Address up to the range. Remap (PCI Base Address) defines the PCI starting address.

**Writes**—PCI 9054 continues to accept writes and returns READY# until the Write FIFO is full. It then holds off READY# until space becomes available in the Write FIFO. A programmable PCI Initiator FIFO “almost full” status output is provided (DMPAF).

**Reads**—PCI 9054 holds off READY# while gathering an Lword from the PCI Bus. Programmable prefetch modes are available if prefetch is enabled: prefetch, 4, 8, 16, or continuous until the PCI Initiator cycle ends. The Read cycle is terminated when the Local BLAST# input is asserted. Unused Read data is flushed from the FIFO.

The PCI 9054 does not prefetch Read data for Single-Cycle PCI Initiator reads (Local BLAST# input asserted during the first Data phase). In this case, the PCI 9054 reads a single PCI Lword unless PCI Initiator Read Ahead mode is enabled.

For PCI Initiator Single-Cycle reads, the PCI 9054 sets the same PCI Bus byte enables as set on the Local Bus.

For Burst-Cycle reads, the PCI 9054 reads entire Lwords (all PCI Bus byte enables are asserted).

If the PCI Initiator Prefetch Limit bit is enabled (DMPBAM[11]=1), the PCI 9054 does not prefetch past a 4 KB boundary. Also, the Local Bus must not cross a 4 KB boundary during a Burst read.

The PCI 9054 never prefetches beyond the region specified for PCI Initiator accesses.

#### 5.4.1.4 PCI Initiator I/O Configuration Access

When a Local PCI Initiator I/O access to the PCI Bus occurs, the PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration Enable bit (DMCFGA[31]) determines whether an I/O or Configuration access is to be made to the PCI Bus.

Local Burst accesses are broken into single PCI I/O (address/data) cycles. The PCI 9054 does not prefetch Read data for I/O and configuration reads.

For PCI Initiator I/O or Configuration cycles, the PCI 9054 asserts the same PCI Bus byte enables as set on the Local Bus.

#### 5.4.1.5 PCI Initiator I/O

If the Configuration Enable bit is cleared (DMCFGA[31]=0), a Single I/O access is made to the PCI Bus. The Local Address, Remapped Decode Address bits, and Local byte enables are encoded to provide the address and are output with an I/O Read or Write command during a PCI Address cycle.

When the I/O Remap Select bit is set (DMPBAM[13]=1), the PCI Address bits [31:16] are forced to 0 for the 64 KB I/O address limit.

For writes, data is loaded into the Write FIFO and READY# is returned to the Local Bus. For reads, the PCI 9054 holds off READY# while receiving an Lword from the PCI Bus.

#### 5.4.1.6 PCI Initiator Configuration (PCI Configuration Type 0 or Type 1 Cycles)

If the Configuration Enable bit (DMCFGA[31]) is set, a Configuration access is made to the PCI Bus. In addition to enabling configuration of this bit, the user must provide all register information. The Register Number and Device Number bits (DMCFGA[7:2] and DMCFGA[15:11], respectively) must be modified and a new Configuration Read/Write cycle must be performed before accessing other registers or devices.

If the PCI Configuration Address register selects a Type 0 command, bits [10:0] of the register are copied to address bits [10:0]. Bits [15:11] (device number) are translated into a single bit being set in the PCI Address bits [31:11]. The PCI Address bits [31:11] can be used as a device select. For a Type 1 command, bits [23:0] are copied from the register to bits [23:0] of the PCI address. The PCI Address bits [31:24] are set to 0. A Configuration Read or Write command code is output with the address during the PCI Address cycle (refer to the DMCFGA register).

For writes, Local data is loaded into the Write FIFO and READY# is returned. For reads, the PCI 9054 holds off READY# while gathering an Lword from the PCI Bus.

### 5.4.1.6.1 PCI Initiator Configuration Cycle Example

To perform a Type 0 Configuration cycle to PCI device on AD[21]:

1. The PCI 9054 must be configured to allow PCI Initiator access to the PCI Bus. The PCI 9054 must also be set to respond to I/O space accesses. These bits must be set (PCICR[2:0]=111b).

In addition, PCI Initiator memory and I/O access must be enabled (DMPBAM[1:0]=11).

2. The Local memory map selects the PCI Initiator range. For this example, use a range of 1 MB:

$$1 \text{ MB} = 2^{20} = 000FFFFFFh$$

The value to program into the Range register is the inverse of 000FFFFFFh (FFF00000h):

$$\text{DMRR} = \text{FFF00000h}$$

3. The Local memory map determines the Local Base Address for the PCI Initiator-to-PCI I/O Configuration register. For this example, use 40000000h:

$$\text{DMLBAI} = 40000000h$$

4. The PCI Address (Remap) for PCI Initiator-to-PCI Memory register must enable the PCI Initiator I/O access. The PCI Initiator I/O Access Enable bit must be set (DMPBAM[1]=1).
5. The user must know which PCI device and PCI Configuration register the PCI Configuration cycle is accessing. This example assumes the IDSEL signal of the Target PCI device is connected to AD[21] (logical device #10=0Ah). Also access PCIBAR0 (the fourth register, counting from 0; use Table 11-2 for reference). Set DMCFG A[31, 23:0] as follows:

Bit	Description	Value
1:0	Configuration Type 0.	00b
7:2	Register Number. Fourth register. Must program a "4" into this value, beginning with bit 2.	000100b
10:8	Function Number.	000b
15:11	Device Number n-11, where n is the value in AD[n]=21-11 = 10.	01010b
23:16	Bus Number.	00000000b
31	Configuration Enable.	1

After these registers are configured, a simple Local Master Memory cycle to the I/O base address is necessary to generate a PCI Configuration Read

or Write cycle. Offset to the base address is not necessary because the register offset for the read or write is specified in the Configuration register. The PCI 9054 takes the Local Bus Master Memory cycle and checks for the Configuration Enable bit (DMCFG A[31]). If set, the PCI 9054 converts the current cycle to a PCI Configuration cycle, using the DMCFG A register and the Write/Read signal (LW/R#).

The Register Number and Device Number bits (DMCFG A[7:2] and DMCFG A[15:11], respectively) must be modified and a new Configuration Read/Write cycle must be performed before accessing other registers or devices.

### 5.4.1.7 PCI Initiator PCI Dual Address Cycle

The PCI 9054 supports PCI Dual Address Cycle (DAC) when it is a PCI Bus Master using the DMDAC register for PCI Initiator transactions. The DAC command is used to transfer a 64-bit address to devices that support 64-bit addressing when the address is not in the low 4-GB Address space. The PCI 9054 performs the address portion of a DAC in two PCI clock periods, where the first PCI address is a Lo-Addr with the command (C/BE[3:0]#) "D" and the second PCI address will be a Hi-Addr with the command (C/BE[3:0]#) "6" or "7", depending upon it being a PCI Read or a PCI Write cycle. Whenever the DMDAC register contains a value of 0x00000000, the PCI 9054 performs a Single Address Cycle (SAC) on the PCI Bus. (Refer to Figure 5-4.)

### 5.4.1.8 PCI Initiator/Target Abort

The PCI 9054 PCI Initiator/Target Abort logic enables a Local Bus Master to perform a PCI Initiator Bus poll of devices to determine whether devices exist (typically when the Local Bus performs Configuration cycles to the PCI Bus). When a PCI Master device attempts to access and does not receive DEVSEL# within six PCI clocks, it results in a Master Abort. The Local Bus Master must clear the Received Master Abort bit or Target Abort bit (PCISR[13 or 11]=0, respectively) and continue by processing the next task.

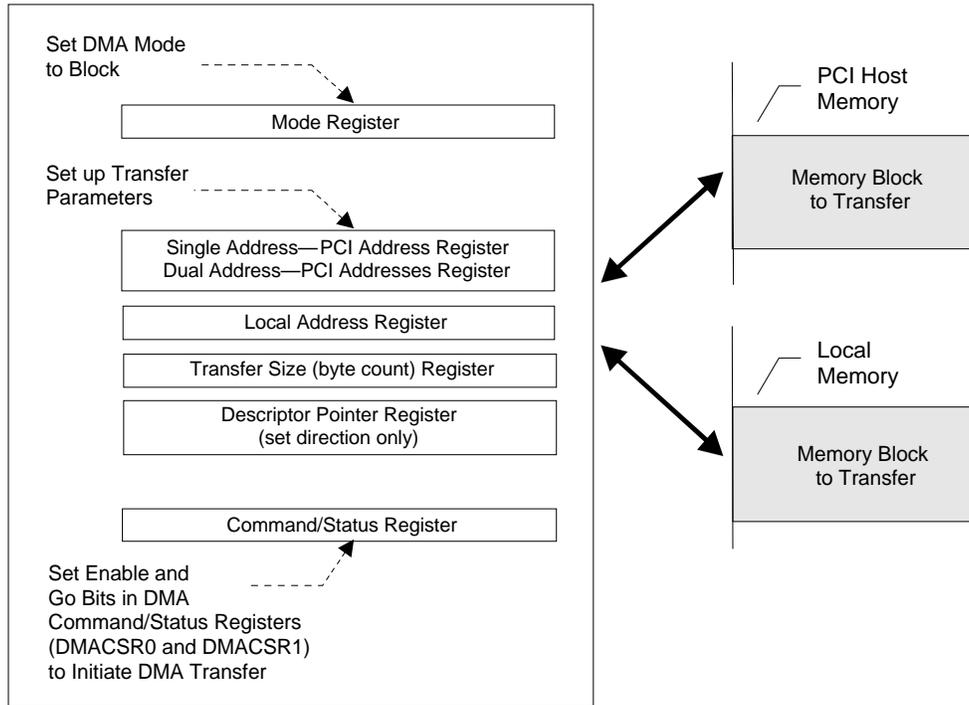


Figure 5-4. Block DMA Mode Initialization (Single Address or Dual Address PCI)

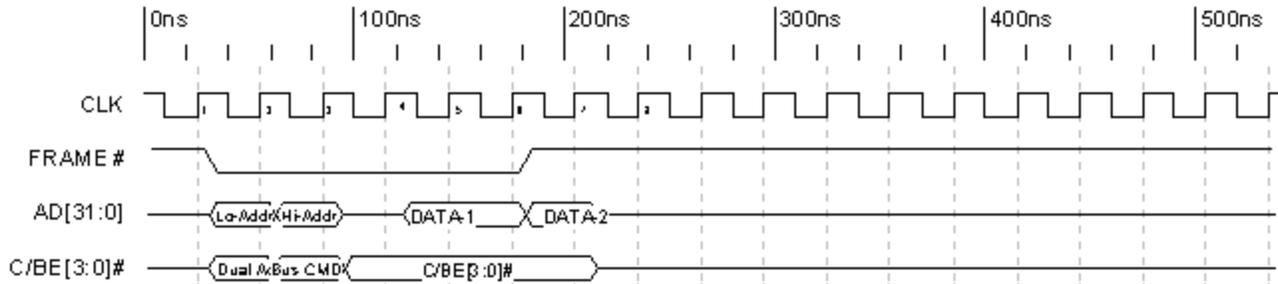


Figure 5-5. Dual Address Timing

If a PCI Master/Target Abort, or Retry Time-Out is encountered during a transfer, the PCI 9054 asserts LSERR# if enabled [(INTCSR[1:0]=1), which can be used as an NMI]. If a Local Bus Master is waiting for READY#, it is asserted along with BTERM#. The Local Master's interrupt handler can take the appropriate application-specific action. It can then clear the Target Abort bit (PCISR[11]) to de-assert the LSERR# interrupt and re-enable PCI Initiator transfers.

If a Local Bus Master is attempting a Burst read from a nonresponding PCI device (Master/Target Abort), it receives READY# and BTERM# for the first cycle only. In addition, the PCI 9054 asserts LSERR# if the Enable Local Bus LSERR# bits are enabled (INTCSR[1:0], which can be used as an NMI). If the Local processor cannot terminate its Burst cycle, it may cause the Local processor to hang. The Local Bus must then be reset from the PCI Bus. If a Local Bus Master cannot terminate its cycle with BTERM# output, it should not perform Burst cycles when attempting to determine whether a PCI device exists.

### 5.4.1.9 PCI Initiator Memory Write and Invalidate

The PCI 9054 can be programmed to perform Memory Write and Invalidate cycles to the PCI Bus for PCI Initiator transfers, as well as DMA transfers (refer to Section 5.5.4). The PCI 9054 supports Memory Write and Invalidate transfers for cache line sizes of 8 or 16 Lwords. Size is specified in the System Cache Line Size bits (PCICLSR[7:0]). If a size other than 8 or 16 is specified, the PCI 9054 performs Write transfers rather than Memory Write and Invalidate transfers.

PCI Initiator Memory Write and Invalidate transfers are enabled when the Invalidate Enable and the Memory Write and Invalidate Enable bits are set (DMPBAM[9] and PCICR[4], respectively).

In Memory Write and Invalidate mode, if the start address of the PCI Initiator transfer is on a cache line boundary, the PCI 9054 waits until the number of Lwords required for the specified cache line size are written from the Local Bus before starting a PCI Memory Write and Invalidate access. This ensures a complete cache line write can complete in one PCI Bus ownership.

If the start address is not on a cache line boundary, the PCI 9054 starts a normal PCI Write access (PCI command code = 7h). The PCI 9054 terminates a cycle at a cache line boundary if it is performing a normal write or if it is performing a Memory Write and Invalidate cycle and another cache line of data is not available. If an entire cache line is available by the time PCI 9054 regains use of the PCI Bus, the PCI 9054 resumes Memory Write and Invalidate cycles. Otherwise, it continues with a normal write. If a Target disconnects before a cache line is completed, the PCI 9054 completes the remainder of that cache line, using normal writes.

### 5.4.2 PCI Target Operation (PCI Master-to-Local Bus Access)

The PCI 9054 supports both Burst Memory-Mapped Transfer accesses and I/O-Mapped, Single-Transfer accesses to the Local Bus from the PCI Bus through a 16-Lword (64-byte) PCI Target Read FIFO and a 32-Lword (128-byte) PCI Target Write FIFO. The PCI Base Address registers are provided to set up the location of the adapter in the PCI memory and the I/O

space. In addition, Local mapping registers allow address translation from the PCI Address Space to the Local Address Space.

Three spaces are available:

- Space 0
- Space 1
- Expansion ROM space

Expansion ROM space is intended to support a bootable ROM device for the Host.

For Single-Cycle PCI Target reads, the PCI 9054 reads a single Local Bus Lword or partial Lword. The PCI 9054 disconnects after one transfer for all PCI Target I/O accesses.

For the highest data-transfer rate, the PCI 9054 supports posted writes and can be programmed to prefetch data during a PCI Burst Read. The Prefetch size, when enabled, can be from one to 16 Lwords or until the PCI Bus stops requesting. When the PCI 9054 prefetches, if enabled, it drops the Local Bus after reaching the prefetch counter. In Continuous Prefetch mode, the PCI 9054 prefetches as long as FIFO space is available and stops prefetching when the PCI Bus terminates the request. If Read prefetching is disabled, the PCI 9054 disconnects after one Read transfer.

In addition to Prefetch mode, the PCI 9054 supports Read Ahead mode (refer to Section 5.4.2.3).

Each Local space can be programmed to operate in an 8-, 16-, or 32-bit Local Bus width. The PCI 9054 has an internal wait state generator and external wait state input, READY#. READY# can be disabled or enabled with the Internal Configuration registers.

With or without wait state(s), the Local Bus, independent of the PCI Bus, can:

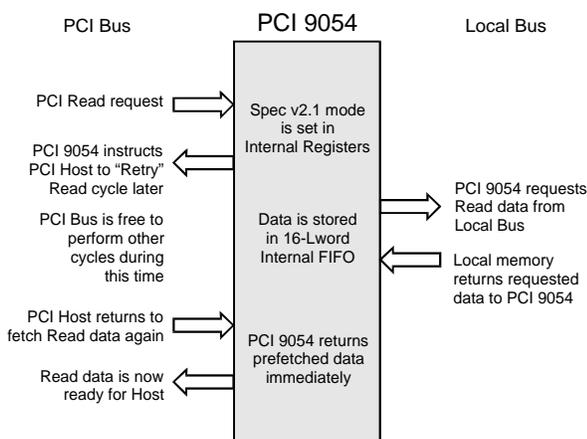
- Burst as long as data is available (Continuous Burst mode)
- Burst four Lwords at a time (recommended)
- Perform a continuous Single cycle

### 5.4.2.1 PCI Target Lock

The PCI 9054 supports direct PCI-to-Local-Bus Exclusive accesses (locked atomic operations). A PCI-locked operation to the Local Bus results in the entire address Space 0, Space 1, and Expansion ROM space being locked until they are released by the PCI Bus Master. Locked operations are enabled or disabled with the PCI Target LOCK# Enable bit (MARBR[22]) for PCI-to-Local accesses.

### 5.4.2.2 PCI Target PCI v2.1 Delayed Read Mode

The PCI 9054 can be programmed through the PCI Specification v2.1 Mode bit (MARBR[24]=1) to perform delayed reads, as specified in PCI Specification v2.1.



**Figure 5-6. PCI Target PCI v2.1 Delayed Reads**

**Note:** The figure represents a sequence of Bus cycles.

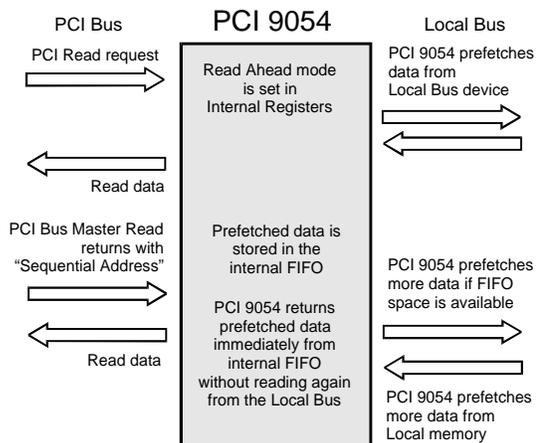
In addition to delayed reads, the PCI 9054 supports the following PCI Specification v2.1 functions:

- No write while a read is pending (PCI Retry for writes)
- Write and flush pending read

### 5.4.2.3 PCI Target PCI Read Ahead Mode

The PCI 9054 also supports Read Ahead mode, where prefetched data can be read from the internal FIFO of the PCI 9054 instead of from the Local Bus. The address must be subsequent to the previous

address and 32-bit aligned (next address = current address + 4). Read Ahead mode functions with or without PCI Delayed Read mode.



**Figure 5-7. PCI Target PCI 9054 Read Ahead Mode**

**Note:** The figure represents a sequence of Bus cycles.

### 5.4.2.4 PCI Target Transfer

A PCI Bus Master addressing the Memory space decoded for the Local Bus initiates transactions. Upon a PCI Read/Write, the PCI 9054 becomes a Local Bus Master and arbitrates for the Local Bus.

The PCI 9054 then reads data into the PCI Target Read FIFO or writes data to the Local Bus.

The PCI Target or PCI Initiator preempts DMA; however, the PCI Target does not preempt the PCI Initiator (refer to Section 5.4.3.1).

The PCI 9054 can be programmed to "keep" the PCI Bus by generating a wait state(s) and de-asserting TRDY# if the Write FIFO becomes full. The PCI 9054 can also be programmed to "keep" the Local Bus and continue asserting LHOLD if the PCI Target Write FIFO becomes empty or the PCI Target Read FIFO becomes full. In either case, the Local Bus is dropped when the Local Bus Latency Timer is enabled and expires (MARBR[7:0]).

For PCI Target writes, the PCI Bus writes data to the Local Bus. The PCI Target is the "Command from the PCI Host," which has highest priority.

For PCI Target reads, the PCI Bus Master reads data from the Local Bus Slave.

The PCI 9054 supports on-the-fly Endian conversion for Space 0, Space 1, and Expansion ROM space. The Local Bus can be Big/Little Endian by using the programmable internal register configuration.

**Note:** The PCI Bus is always Little Endian.

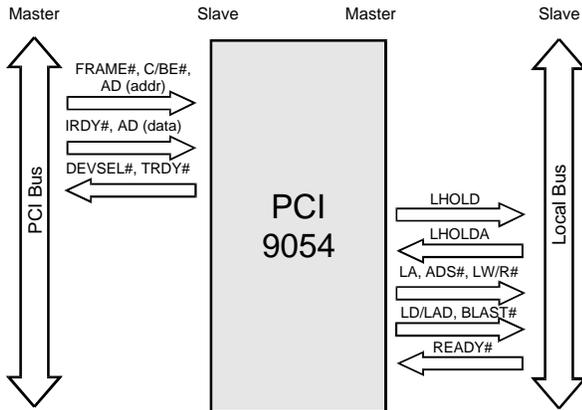


Figure 5-8. PCI Target Write

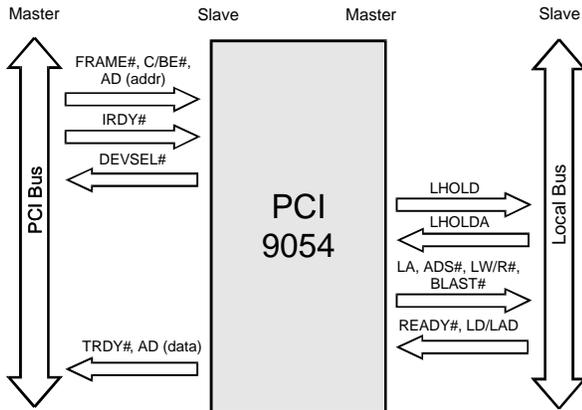


Figure 5-9. PCI Target Read

**Note:** The figures represent a sequence of Bus cycles.

### 5.4.2.5 PCI Target PCI-to-Local Address Mapping

**Note:** Not applicable in I2O mode.

Three Local Address spaces—Space 0, Space 1, and Expansion ROM—are accessible from the PCI Bus. Each is defined by a set of three registers:

- Local Address Range (LAS0RR, LAS1RR, and/or EROMRR)
- Local Base Address (LAS0BA, LAS1BA, and/or EROMBA)
- PCI Base Address (PCIBAR2, PCIBAR3, and/or PCIERBAR)

A fourth register, the Bus Region Descriptor register for PCI-to-Local Accesses (LBRD0 and/or LBRD1), defines the Local Bus characteristics for the PCI Target regions (refer to Figure 5-10).

Each PCI-to-Local Address space is defined as part of reset initialization, as described in Section 5.4.2.5.1. These Local Bus characteristics can be modified at any time before actual data transactions.

#### 5.4.2.5.1 PCI Target Local Bus Initialization

**Range**—Specifies which PCI Address bits to use for decoding a PCI access to Local Bus space. Each bit corresponds to a PCI Address bit. Bit 31 corresponds to address bit 31. Write 1 to all bits that must be included in decode and 0 to all others.

**Remap PCI-to-Local Addresses into a Local Address Space**—Bits in this register remap (replace) the PCI Address bits used in decode as the Local Address bits.

**Local Bus Region Descriptor**—Specifies the Local Bus characteristics.

#### 5.4.2.5.2 PCI Target PCI Initialization

After a PCI reset, the software determines how much address space is required by writing all ones (1) to a PCI Base Address register and then reading back the value. The PCI 9054 returns zeroes (0) in the Don't Care Address bits, effectively specifying the address space required. The PCI software then maps the Local Address space into the PCI Address space by programming the PCI Base Address register. (Refer to Figure 5-10.)

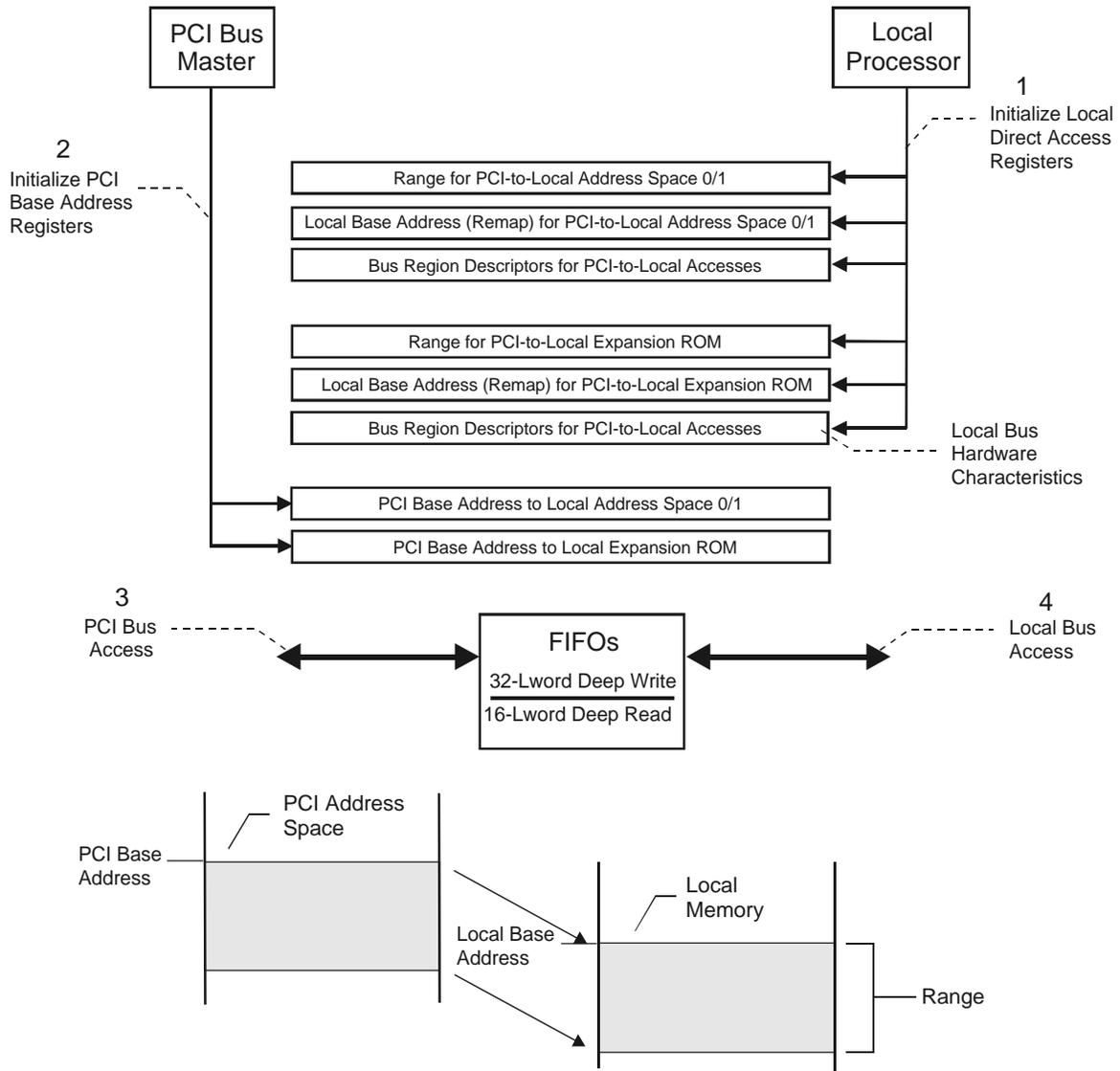


Figure 5-10. Local Bus PCI Target Access

### 5.4.2.5.3 PCI Target Byte Enables (C Mode)

During a PCI Target transfer, each of three spaces (Space 0, Space 1, and Expansion ROM spaces) can be programmed to operate in an 8-, 16-, or 32-bit Local Bus width by encoding the Local Byte Enables (LBE[3:0]#).

LBE[3:0]# (PQFP—Pins 91-94; PBGA—Pins P15, N14, L11, and M13) are encoded, based on the configured bus width, as follows:

**32-Bit Bus**—The four-byte enables indicate which of the four bytes are active during a Data cycle:

- BE3# Byte Enable 3—LD[31:24]
- BE2# Byte Enable 2—LD[23:16]
- BE1# Byte Enable 1—LD[15:8]
- BE0# Byte Enable 0—LD[7:0]

**16-Bit Bus**—BE3#, BE1# and BE0# are encoded to provide BHE#, LA1, and BLE#, respectively:

- BE3# Byte High Enable (BHE#)—LD[15:8]
- BE2# not used
- BE1# Address bit 1 (LA1)
- BE0# Byte Low Enable (BLE#)—LD[7:0]

**8-Bit Bus**—BE1# and BE0# are encoded to provide LA1 and LA0, respectively:

- BE3# not used
- BE2# not used
- BE1# Address bit 1 (LA1)
- BE0# Address bit 0 (LA0)

### 5.4.2.5.4 PCI Target Byte Enables (J Mode)

During a PCI Target transfer, each of three spaces (Space 0, Space 1, and Expansion ROM spaces) can be programmed to operate in an 8-, 16-, or 32-bit Local Bus width by encoding the Local Byte Enables (LBE[3:0]#).

LBE[3:0]# (PQFP—Pins 91-94; PBGA—Pins P15, N14, L11, and M13) are encoded, based on the configured bus width, as follows:

**32-Bit Bus**—The four-byte enables indicate which of the four bytes are active during a Data cycle:

- BE3# Byte Enable 3—LAD[31:24]
- BE2# Byte Enable 2—LAD[23:16]
- BE1# Byte Enable 1—LAD[15:8]

- BE0# Byte Enable 0—LAD[7:0]

**16-Bit Bus**—BE3#, BE1# and BE0# are encoded to provide BHE#, LAD1, and BLE#, respectively:

- BE3# Byte High Enable (BHE#)—LAD[15:8]
- BE2# not used
- BE1# Address bit 1 (LAD1)
- BE0# Byte Low Enable (BLE#)—LAD[7:0]

**8-Bit Bus**—BE1# and BE0# are encoded to provide LAD1 and LAD0, respectively:

- BE3# not used
- BE2# not used
- BE1# Address bit 1 (LAD1)
- BE0# Address bit 0 (LAD0)

#### 5.4.2.5.4.1 PCI Target Example

A 1 MB Local Address Space, 12300000h through 123FFFFFFh, is accessible from the PCI Bus at PCI addresses 78900000h through 789FFFFFFh.

- a. Local initialization software sets the Range and Local Base Address registers as follows:
  - **Range**—FFF00000h (1 MB, decode the upper 12 PCI Address bits)
  - **Local Base Address (Remap)**—123XXXXXh (Local Base Address for PCI-to-Local accesses) [Space Enable bit(s) must be set to be recognized by the PCI Host (LAS0BA[0]=1, LAS1BA[0]=1)]
- b. PCI Initialization software writes all ones to the PCI Base Address, then reads it back again.
  - The PCI 9054 returns a value of FFF00000h. The PCI software then writes to the PCI Base Address register(s).
  - **PCI Base Address**—789XXXXXh (PCI Base Address for Access to the Local Address Space registers, PCIBAR2 and PCIBAR3).

For a PCI Direct access to the Local Bus, the PCI 9054 has a 32-Lword (128-byte) Write FIFO and a 16-Lword (64-byte) Read FIFO. The FIFOs enable the Local Bus to operate independent of the PCI Bus. The PCI 9054 can be programmed to return a Retry response or to throttle TRDY# for any PCI Bus transaction attempting to write to the PCI 9054 Local Bus when the FIFO is full.

For PCI Read transactions from the Local Bus, the PCI 9054 holds off TRDY# while gathering data from the Local Bus. For Read accesses mapped to PCI Memory space, the PCI 9054 prefetches up to 16

Lwords (has Continuous Prefetch mode) from the Local Bus. Unused Read data is flushed from the FIFO. For Read accesses mapped to PCI I/O space, the PCI 9054 does not prefetch Read data. Rather, it breaks each read of a Burst cycle into a Single Address/Data cycle on the Local Bus.

The PCI Target Retry Delay Clocks bits (LBRD0[31:28]) can be used to program the period of time in which the PCI 9054 holds off TRDY#. The PCI 9054 issues a Retry to the PCI Bus Transaction Master when the programmed time period expires. This occurs when the PCI 9054 cannot gain control of the Local Bus and return TRDY# within the programmed time period.

#### 5.4.2.6 PCI Target Priority

PCI Target accesses have a higher priority than DMA accesses, thereby preempting DMA transfers. During a DMA transfer, if the PCI 9054 detects a pending PCI Target access, it releases the Local Bus within two Data transfers. The PCI 9054 resumes operation after the PCI Target access completes.

When the PCI 9054 DMA controller owns the Local Bus, its LHOLD output and LHOLDA input are asserted. When a PCI Target access occurs, the PCI 9054 releases the Local Bus within two Lword transfers by de-asserting LHOLD and floating the Local Bus outputs. After the PCI 9054 acknowledges that LHOLDA is de-asserted, it requests the Local Bus for a PCI Target transfer by asserting LHOLD. When the PCI 9054 receives LHOLDA, it drives the bus and performs the PCI Target transfer. Upon completing a PCI Target transfer, the PCI 9054 releases the Local Bus by de-asserting LHOLD and floating the Local Bus outputs. After the PCI 9054 samples LHOLDA is de-asserted and the Local Bus Pause Timer is set to zero, it requests a DMA transfer from the Local Bus by re-asserting LHOLD. When it receives LHOLDA, it drives the bus and continues the DMA transfer.

#### 5.4.3 Deadlock Conditions

Deadlock can occur when a PCI Bus Master must access the PCI 9054 Local Bus at the same time a Master on the PCI 9054 Local Bus must access the PCI Bus.

There are two types of deadlock:

- **Partial Deadlock**—A Local Bus Master is performing a Direct Bus Master access to a PCI Bus device other than the PCI Bus device concurrently trying to access the Local Bus
- **Full Deadlock**—A Local Bus Master is performing a Direct Bus Master access to the same PCI Bus device concurrently trying to access the Local Bus

This applies only to PCI Initiator and PCI Target accesses through the PCI 9054. Deadlock does not occur in transfers through the PCI 9054 DMA channels or the PCI 9054 internal registers (such as mailboxes).

For partial deadlock, the PCI access to the Local Bus times out (the PCI Target Retry Delay Clock (LBRD0[31:28]), which is programmable through the Local Bus Region Descriptor register) and the PCI 9054 responds with a PCI Retry. The PCI Specification requires that a PCI Master release its request for the PCI Bus (de-assert REQ#) for a minimum of two PCI clocks after receiving a Retry. This allows the PCI Bus arbiter to grant the PCI Bus to the PCI 9054 so that it can complete its PCI Initiator access and free up the Local Bus. Possible solutions are described in the following sections for cases in which the PCI Bus arbiter does not function as described (PCI Bus architecture dependent), waiting for a time out is undesirable, or a full deadlock condition exists.

When a full deadlock occurs, the only solution is to backoff the Local Bus Master.

#### 5.4.3.1 Backoff

The PCI 9054 BREQo signal indicates whether a possible deadlock condition exists. The PCI 9054 starts the Backoff Timer (programmable through registers) when it detects the following conditions:

- A PCI Bus Master is attempting to access memory or an I/O device on the Local Bus and is not gaining access (*for example*, LHOLDA is not received).
- A Local Bus Master is performing a Direct Bus Master Read access to the PCI Bus. Or, a Local Bus Master is performing a Direct Bus Master Write access to the PCI Bus and the PCI 9054 PCI Initiator Write FIFO cannot accept another Write cycle.

If the Local Bus Backoff Enable bit is enabled (EROMBA[4]=1), the Backoff Timer expires, and the PCI 9054 has not received LHOLDA, the PCI 9054

asserts BREQo. External bus logic can use this signal to perform backoff.

The Backoff cycle is device/bus architecture dependent. External logic (an arbiter) can assert the necessary signals necessary to cause a Local Bus Master to release a Local Bus (backoff). After the Local Bus Master backs off, it can grant the bus to the PCI 9054 by asserting LHOLDA.

Once BREQo is asserted, READY# for the current Data cycle is never asserted (the Local Bus Master must perform backoff). When the PCI 9054 detects LHOLDA, it proceeds with the PCI Master-to-Local-Bus access. When this access completes and the PCI 9054 releases the Local Bus, external logic can release the backoff and the Local Bus Master can resume the cycle interrupted by the Backoff cycle. The PCI 9054 Write FIFO retains all data it acknowledged (*that is*, the last data for which READY# was asserted).

After the backoff condition ends, the Local Bus Master restarts the last cycle with ADS#. For writes, data following ADS# should be the data the PCI 9054 did not acknowledge prior to the Backoff cycle (*for example*, the last data for which READY# is not asserted).

If a PCI Read cycle completes when the Local Bus is backed off, the Local Bus Master receives that data if the Local Master restarts the same last cycle (data is not read twice). A new read is performed if the resumed Local Bus cycle is not the same as the Backed Off cycle.

#### 5.4.3.1.1 Software/Hardware Solution for Systems without Backoff Capability

For adapters that do not support backoff, a possible deadlock solution is as follows.

The PCI Host software can use PCI Host software, external Local Bus hardware, general purpose output USERo and general purpose input USERi to prevent deadlock. USERo can be asserted to request that the external arbiter not grant the bus to any Local Bus Master except the PCI 9054. Status output from the Local arbiter can be connected to the general purpose input USERi to indicate that no Local Bus Master owns the Local Bus, or the PCI Host to determine that no Local Bus Master that currently owns the Local Bus can read input. The PCI Host can then perform PCI Target access. When the Host finishes, it de-asserts USERo.

#### 5.4.3.1.2 Preempt Solution

For devices that support preempt, USERo can be used to preempt the current Bus Master device. When USERo is asserted, the current Local Bus Master device completes its current cycle and releases the Local Bus, de-asserting LHOLD.

#### 5.4.3.2 Software Solutions to Deadlock

Both PCI Host and Local Bus software can use a combination of mailbox registers, doorbell registers, interrupts, direct Local-to-PCI accesses and direct PCI-to-Local accesses to avoid deadlock.

### 5.5 DMA OPERATION

The PCI 9054 supports two independent DMA channels capable of transferring data from the:

- Local-to-PCI Bus
- PCI-to-Local Bus

Each channel consists of a DMA controller and a dedicated bidirectional FIFO. Both channels support Block transfers, Scatter/Gather transfers, with or without End of Transfer (EOT#). Only DMA Channel 0 supports Demand mode DMA transfers. Master mode must be enabled with the Master Enable bit (PCICR[2]) before the PCI 9054 can become a PCI Bus Master. In addition, both DMA channels can be programmed to:

- Operate in 8-, 16-, or 32-bit Local Bus width
- Use zero to 15 internal wait states (Local Bus)
- Enable/disable internal wait states (Local Bus)
- Enable/disable Local Bus Burst capability
- Limit Local Bus bursts to four (BTERM# enable/disable)
- Hold Local address constant (Local Target is FIFO) or increment
- Perform PCI Memory Write and Invalidate (command code = Fh) or normal PCI Memory Write (command code = 7h)
- Pause Local transfer with/without BLAST# (DMA Fast/Slow termination)
- Assert PCI interrupt (INTA#) or Local interrupt (LINT#) when DMA transfer is complete or Terminal Count is reached during Scatter/Gather DMA mode transfers
- Operate in DMA Clear Count mode (only if the descriptor is in Local memory)

The PCI 9054 also supports PCI Dual Address with the upper 32-bit registers (DMADAC0 and DMADAC1). The Local Bus Latency Timer determines the number of Local clocks the PCI 9054 can burst data before relinquishing the Local Bus. The Local Pause Timer sets how soon the DMA channel can request the Local Bus.

### 5.5.1 DMA PCI Dual Address Cycle

The PCI 9054 supports PCI Dual Address Cycle (DAC) when it is a PCI Bus Master, using the DMADAC0 and DMADAC1 registers for Block DMA transactions. Scatter/Gather DMA can utilize the DAC function via the DMADAC0 and DMADAC1 registers or DMAMODE0[18] and DMAMODE1[18]. The DAC command is used to transfer a 64-bit address to devices that support 64-bit addressing when the address is above the 4-GB Address space. The PCI 9054 performs a DAC within two PCI clock periods, where the first PCI address is a Lo-Addr, with the command (C/BE[3:0]#) "D", and the second PCI address is a Hi-Addr, with the command (C/BE[3:0]#) "6" or "7", depending upon whether it is a PCI Read or PCI Write cycle.

### 5.5.2 Block DMA Mode

The Host processor or the Local processor sets the Local and PCI starting addresses, transfer byte count, and transfer direction. The Host or Local processor then sets the DMA Start bit to initiate a transfer. The PCI 9054 requests the PCI and Local Buses and transfers data. Once the transfer completes, the PCI 9054 sets the Channel Done bit(s) (DMACSR0[4]=1 and/or DMACSR1[4]=1) and, if enabled, asserts an interrupt(s) (DMAMODE0[10] and/or DMAMODE1[10]) to the Local processor or the PCI Host (programmable). The Channel Done bit(s) can be polled, instead of interrupt generation, to indicate the DMA transfer status.

DMA registers are accessible from the PCI and Local Buses (refer to Figure 5-4 on page 5-7).

During DMA transfers, the PCI 9054 is a Master on both the PCI and Local Buses. For simultaneous access, PCI Target or PCI Initiator has a higher priority than DMA.

The PCI 9054 releases the PCI Bus if one of the following conditions occur (refer to Figure 5-11 and Figure 5-12):

- FIFO is full (PCI-to-Local Bus)
- FIFO is empty (Local-to-PCI Bus)
- Terminal count is reached
- PCI Bus Latency Timer expires (PCILTR[7:0])—normally programmed by the Host PCI BIOS—and PCI GNT# de-asserts
- PCI Host asserts STOP#

The PCI 9054 releases the Local Bus if one of the following conditions occurs:

- FIFO is empty (PCI-to-Local Bus)
- FIFO is full (Local-to-PCI Bus)
- Terminal count is reached
- Local Bus Latency Timer is enabled and expires (MARBR[7:0])
- Special cycle BREQi# is asserted
- PCI Target request is pending

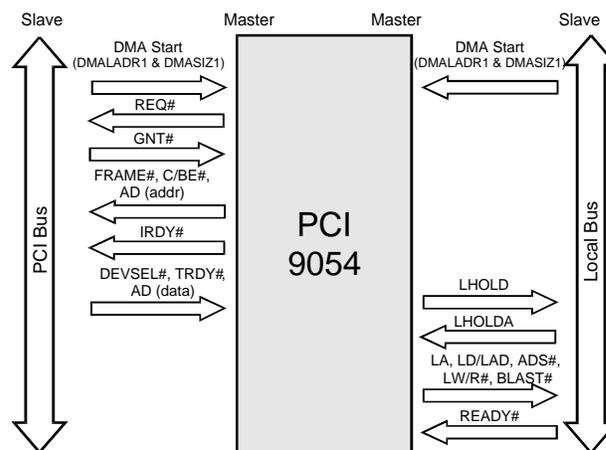


Figure 5-11. DMA, PCI-to-Local Bus

**Note:** The figure represents a sequence of Bus cycles.

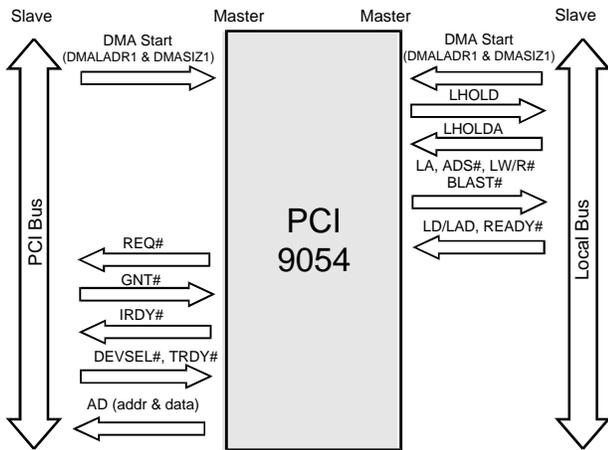


Figure 5-12. DMA, Local-to-PCI Bus

Note: The figure represents a sequence of Bus cycles.

Table 5-2. DMA Local Burst Mode

Burst Enable Bit	BTERM# Enable Bit	Result
0	X	Single cycle
1	0	Burst up to four Data cycles
1	1	Burst forever (terminate when BTERM# is asserted or transfer is completed)

Note: "X" is "Don't Care."

### 5.5.2.1 Block DMA PCI Dual Address Cycle

The PCI 9054 supports the DAC feature in Block DMA mode. Whenever the DMADAC0 or DMADAC1 registers contain a value of 0x00000000, the PCI 9054 performs a Single Address Cycle (SAC) on the PCI Bus. Any other value causes a Dual Address to appear on the PCI Bus. (Refer to Figure 5-13.)

### 5.5.3 Scatter/Gather DMA Mode

In Scatter/Gather DMA mode, the Host processor or Local processor sets up descriptor blocks in Local or Host memory composed of PCI and Local addresses, transfer count, transfer direction, and address of next descriptor block (refer to Figure 5-14 and Figure 5-15). The Host or Local processor then:

- Enables the Scatter/Gather mode bit(s) (DMAMODE0[9]=1 and/or DMAMODE1[9]=1)
- Sets up the address of initial descriptor block in the PCI 9054 Descriptor Pointer register(s) (DMADPR0 and/or DMADPR1)
- Initiates the transfer by setting a control bit(s) (DMACSR0[1:0] and/or DMACSR1[1:0])

The PCI 9054 loads the first descriptor block and initiates the Data transfer. The PCI 9054 continues to load descriptor blocks and transfer data until it detects the End of Chain bit(s) (DMADPR0[1] and/or DMADPR1[1]) is set (these bits are part of each descriptor). When the End of Chain bit(s) is detected, the PCI 9054 completes the current descriptor block and sets the DMA Done bit(s) (DMACSR0[4] and/or DMACSR1[4]). If the End of Chain bit(s) is detected, the PCI 9054 asserts a PCI interrupt (INTA#) and/or Local interrupt (LINT#).

The PCI 9054 can also be programmed to assert PCI or Local interrupts after each descriptor is loaded, then finish transferring.

If Scatter/Gather descriptors are in Local memory, the DMA controller can be programmed to clear the transfer size at completion of each DMA, using the DMA Clear Count Mode bit(s) (DMAMODE0[16] and/or DMAMODE1[16]).

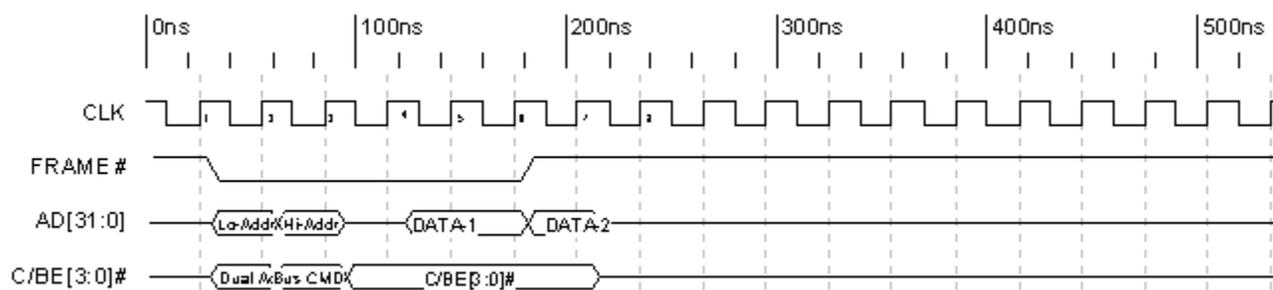


Figure 5-13. Dual Address Timing

**Notes:** In Scatter/Gather DMA mode, the descriptor includes the PCI and Local Address Space, transfer size, and next descriptor pointer. It also includes a DAC value if DMADPR0[18] and/or DMAMODE1[18] is enabled. Otherwise, the register (DMADAC0 and DMADAC1) values are used. The Descriptor Pointer register(s) (DMADPR0 and/or DMADPR1) contains end of chain (bit 1), direction of transfer (bit 3), next descriptor address (bits [31:4]),

interrupt after terminal count (bit 2), and next descriptor location (bit 0) bits.

The Local Bus width must be the same as Local Memory bus width. A DMA descriptor can be on the Local memory or the PCI memory, or both (for example, one descriptor on Local memory, another descriptor on PCI memory and vice-versa).

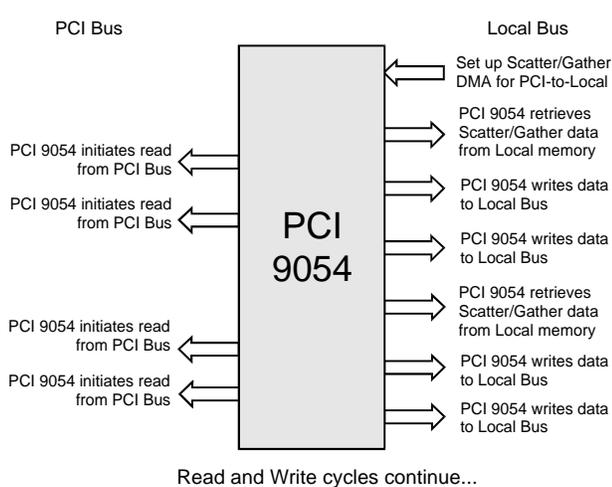


Figure 5-14. Scatter/Gather DMA Mode from PCI-to-Local Bus (Control Access from the Local Bus)

**Note:** The figures represent a sequence of Bus cycles.

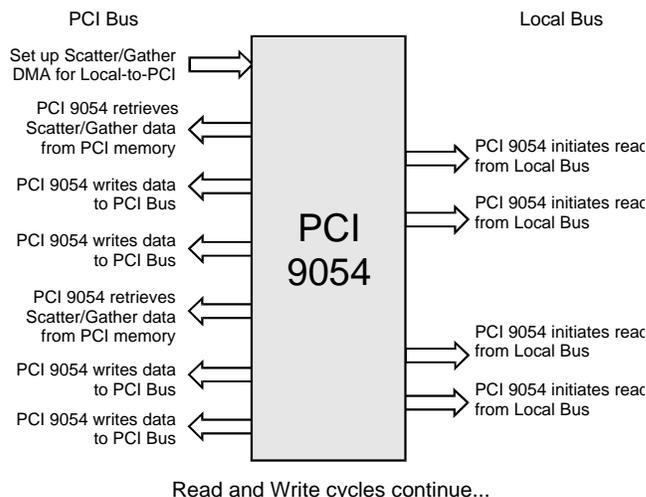


Figure 5-15. Scatter/Gather DMA Mode from Local-to-PCI Bus (Control Access from the PCI Bus)

### 5.5.3.1 Scatter/Gather DMA PCI Dual Address Cycle

The PCI 9054 supports the PCI DAC feature in Scatter/Gather DMA mode for Data transfers only. The descriptor blocks should reside below the 4-GB Address space. The PCI 9054 offers three different options of how PCI DAC Scatter/Gather DMA is utilized. Assuming the descriptor blocks are located on the PCI Bus:

- DMADAC0 and/or DMADAC1 contain(s) a non-zero value. DMAMODE0[18] and/or DMAMODE1[18] is set to 0. The PCI 9054 performs a Single Address Cycle (SAC) four-Lword descriptor block load from PCI memory and DMA transfer with DAC on the PCI Bus. (Refer to Figure 5-16.)
- DMADAC0 and/or DMADAC1 contain(s) an 0x00000000 value. DMAMODE0[18] and/or DMAMODE1[18] is set to 1. The PCI 9054 performs a SAC five-Lword descriptor block load from PCI memory and DMA transfer with PCI DAC on the PCI Bus. (Refer to Figure 5-17.)

- DMADAC0 and/or DMADAC1 contain(s) a non-zero value. DMAMODE0[18] and/or DMAMODE1[18] is set to 1. The PCI 9054 performs a SAC five-Lword descriptor block load from PCI memory and DMA transfer with DAC on the PCI Bus. The fifth descriptor overwrites the value of the DMADAC0 and/or DMADAC1 register. (Refer to Figure 5-17.)

### 5.5.3.2 DMA Clear Count Mode

The PCI 9054 supports DMA Clear Count mode (Write-Back feature, DMAMODE0[16] and DMAMODE1[16]). This feature allows users to control the data transfer blocks during Scatter/Gather DMA operations. The PCI 9054 clears the Transfer Size descriptor to zero by writing to a descriptor-memory location at the end of each transfer chain. This feature works only if DMA descriptors are on the Local Bus.

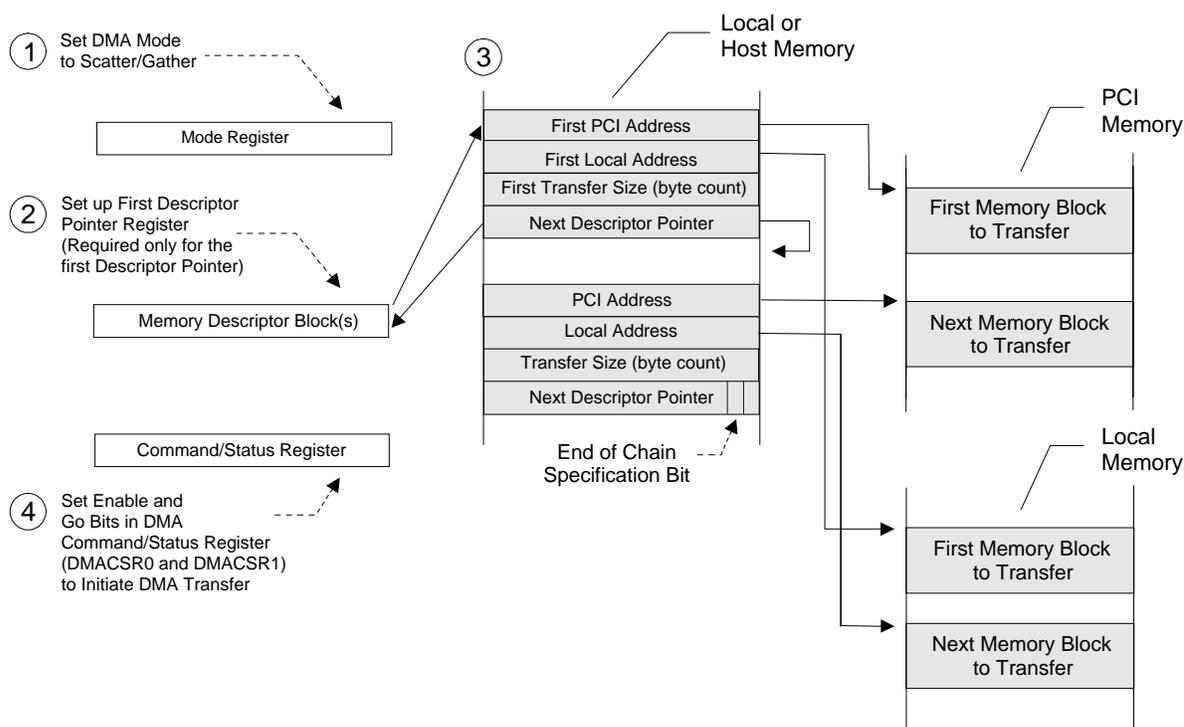


Figure 5-16. Scatter/Gather DMA Mode Descriptor Initialization [PCI SAC/DAC PCI Address (DMADAC0, DMADAC1) Register Dependent]

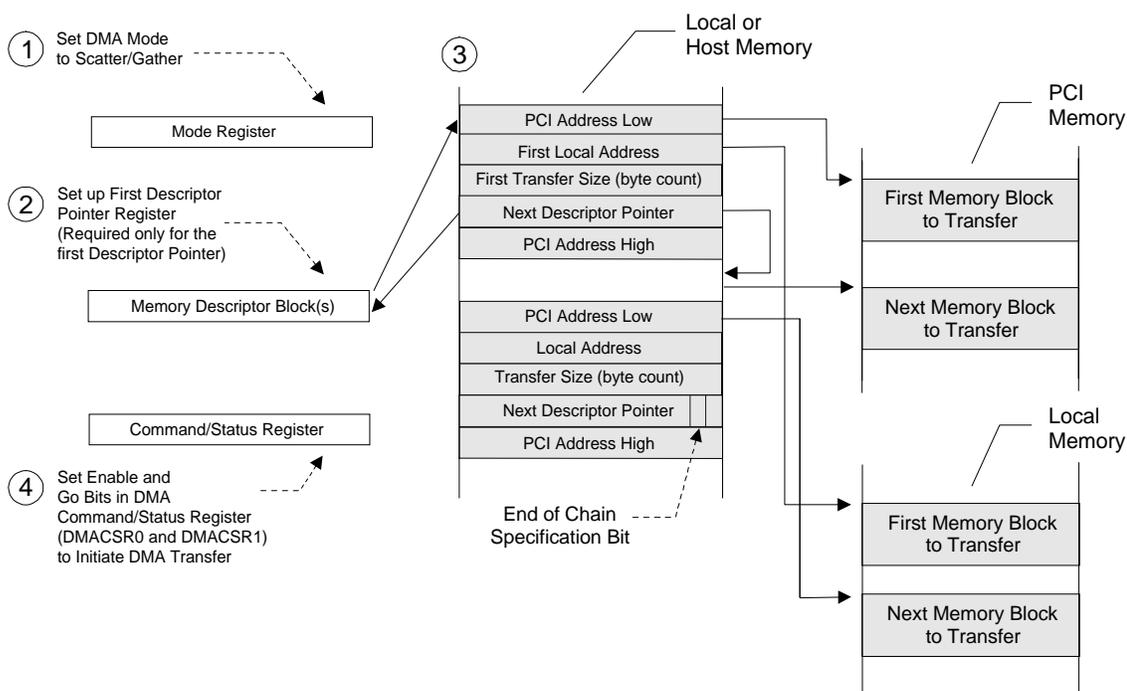


Figure 5-17. Scatter/Gather DMA Mode Descriptor Initialization [DAC PCI Address (DMAMODE0[18], DMAMODE1[18]) Descriptor Dependent] (PCI Address High Added)

## 5.5.4 DMA Memory Write and Invalidate

The PCI 9054 can be programmed to perform Memory Write and Invalidate cycles to the PCI Bus for DMA transfers, as well as PCI Initiator transfers (refer to Section 5.4.1.9). The PCI 9054 supports Memory Write and Invalidate transfers for cache line sizes of 8 or 16 Lwords. Size is specified in the System Cache Line Size bits (PCICLSR[7:0]). If a size other than 8 or 16 is specified, the PCI 9054 performs Write transfers rather than Memory Write and Invalidate transfers.

DMA Memory Write and Invalidate transfers are enabled when the DMA controller Memory Write and Invalidate Enable bit(s) (DMAMODE0[13] and/or DMAMODE1[13]) and the Memory Write and Invalidate Enable bit (PCICR[4]) are set.

In Memory Write and Invalidate mode, the PCI 9054 waits until the number of Lwords required for specified cache line size are read from the Local Bus before starting the PCI access. This ensures a complete cache line write can complete in one PCI Bus ownership. If a Target disconnects before a cache line completes, the PCI 9054 completes the remainder of that cache line, using normal writes before resuming Memory Write and Invalidate transfers. If a Memory Write and Invalidate cycle is in progress, the PCI 9054 continues to burst if another cache line is read from the Local Bus before the cycle completes. Otherwise, the PCI 9054 terminates the burst and waits for the next cache line to be read from the Local Bus. If the final transfer is not a complete cache line, the PCI 9054 completes the DMA transfer, using normal writes.

### 5.5.4.1 DMA Abort

DMA transfers can be aborted, in addition to the EOT# signal, as follows:

1. Clear the DMA Channel Enable bit(s) (DMACSR0[0]=0 and/or DMACSR1[0]=0).
2. Abort DMA by setting the Channel Abort bit(s) (DMACSR0[2]=1 and/or DMACSR1[2]=1).
3. Wait until the Channel Done bit(s) is set (DMACSR0[4]=1 and/or DMACSR1[4]=1).

**Note:** One to two Data transfers occur after the Abort bit is set. Aborting when no DMA cycles are in progress causes the next DMA to abort.

## 5.5.5 DMA Priority

The DMA Channel Priority bit (MARBR[20:19]) can be used to specify the following priorities:

- Rotating (MARBR[20:19]=00)
- DMA Channel 0 (MARBR[20:19]=01)
- DMA Channel 1 (MARBR[20:19]=10)

## 5.5.6 DMA Channel 0/1 Interrupts

A DMA channel can assert a PCI Bus or Local Bus interrupt when done (transfer complete) or after a transfer is complete for the current descriptor in Scatter/Gather DMA mode. The DMA Channel Interrupt Select bit(s) determine whether to assert a PCI (DMAMODE0[17]=1 and/or DMAMODE1[17]=1) or Local (DMAMODE0[17]=0 and/or DMAMODE1[17]=0) interrupt. The Local or PCI processor can read the DMA Channel 0 Interrupt Active bits to determine whether a DMA Channel 0 (INTCSR[21]) or DMA Channel 1 (INTCSR[22]) interrupt is pending.

The Channel Done bit(s) (DMACSR0[4] and/or DMACSR1[4]) can be used to determine whether an interrupt is:

- DMA Done interrupt
- Transfer complete for current descriptor interrupt

The Done Interrupt Enable bit(s) (DMAMODE0[10] and/or DMAMODE1[10]) enable a Done interrupt. In Scatter/Gather DMA mode, a bit in the Next Descriptor Pointer register of the channel (loaded from Local memory) specifies whether to assert an interrupt at the end of the transfer for the current descriptor.

A DMA Channel interrupt is cleared by the Channel Clear Interrupt bit(s) (DMACSR0[3]=1 and/or DMACSR1[3]=1).

## 5.5.7 DMA Data Transfers

The PCI 9054 DMA controller can be programmed to transfer data from the Local-to-PCI Bus or from the PCI-to-Local Bus.

### 5.5.7.1 Local-to-PCI Bus DMA Transfer

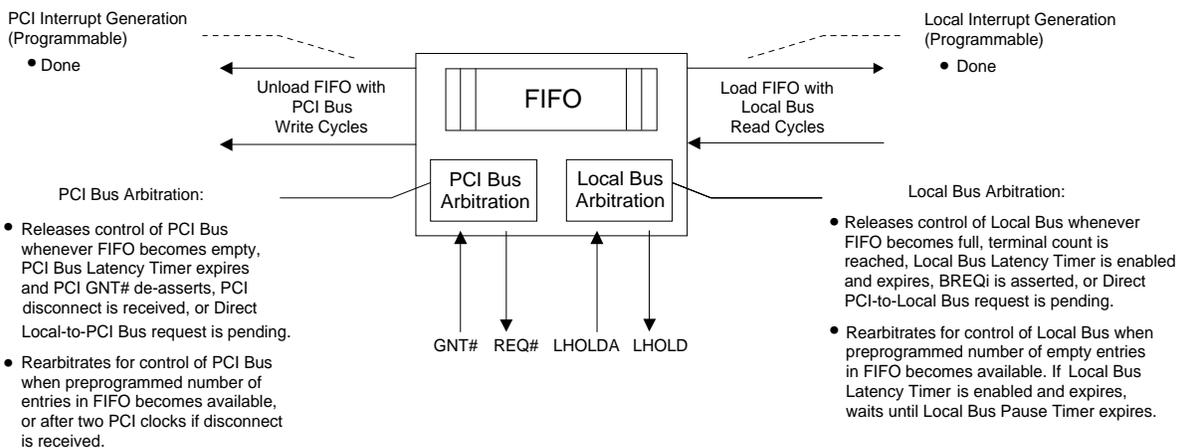


Figure 5-18. Local-to-PCI Bus DMA Data Transfer Operation

### 5.5.7.2 PCI-to-Local Bus DMA Transfer

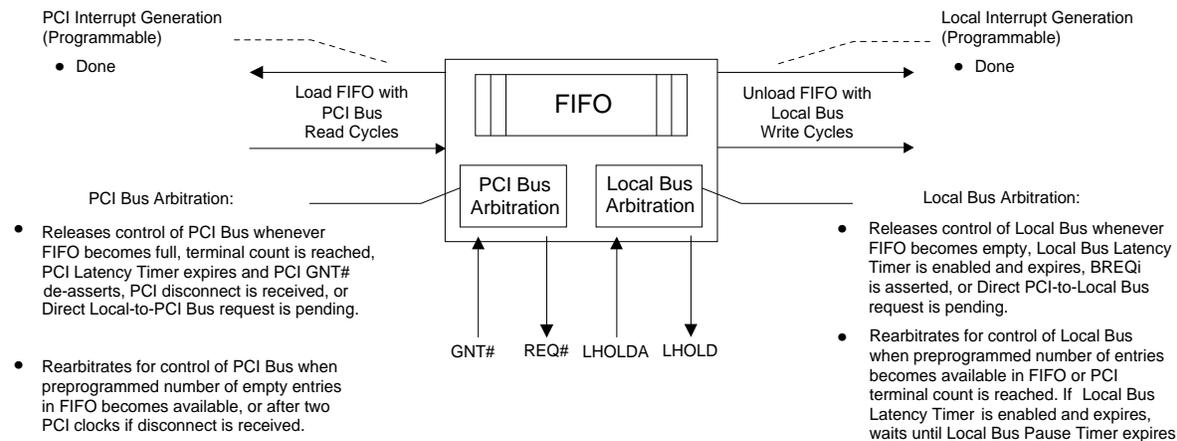


Figure 5-19. PCI-to-Local Bus DMA Data Transfer Operation

### 5.5.7.3 DMA Unaligned Transfers

For unaligned Local-to-PCI transfers, the PCI 9054 reads a partial Lword from the Local Bus. It continues to read Lwords from the Local Bus. Lwords are assembled, aligned to the PCI Bus address, and loaded into the FIFO.

For PCI-to-Local transfers, Lwords are read from the PCI Bus and loaded into the FIFO. On the Local Bus, Lwords are assembled from the FIFO, aligned to the Local Bus address and written to the Local Bus.

On both the PCI and Local Buses, the byte enables for writes determine LA[1:0] for the start of a transfer. For the last transfer, byte enables specify the bytes to be written.

### 5.5.8 Demand Mode DMA, Channel 0

The Fast/Slow Terminate Mode Select bit(s) (DMAMODE0[15] and/or DMAMODE1[15]) determines the number of Lwords to transfer after the DMA controller DREQ0# input is de-asserted.

If BLAST# output is not required for the last Lword of a DMA transfer (bit [15]=1), the DMA controller releases the data bus after it receives an external READY# or the internal wait state counter decrements to 0 for the current Lword. If the DMA controller is currently bursting data, which is not the last Data phase for the Burst, BLAST# is not asserted.

If BLAST# output is required for the last Lword of the DMA transfer (bit [15]=0), the DMA controller transfers one or two Lwords. If DREQ0# is de-asserted during the Address phase of the first transfer in the PCI 9054 Local Bus ownership (ADS#, LHOLDA asserted), the DMA controller completes current Lword. If DREQ0# is de-asserted during any phase other than the Address phase of the first transfer in the PCI 9054 Local Bus ownership, the DMA controller completes the current Lword, and one additional Lword (this allows BLAST# output to be asserted during the final Lword). If the DMA FIFO is full or empty after the Data phase in which DREQ0# is de-asserted, the second Lword is not transferred.

DREQ0# controls only the number of Lword transfers. For an 8-bit bus, the PCI 9054 releases the bus after transferring the last byte for the Lword. For a 16-bit bus, the PCI 9054 releases the bus after transferring the last word for the Lword. (Refer to the timing diagrams in Section 5.6 for C mode or Section 5.7 for J mode.)

### 5.5.9 End of Transfer (EOT#) Input

The DMA EOT# Enable bit(s) (DMAMODE0[14] and/or DMAMODE1[14]) determines the number of Lwords to transfer after a DMA controller asserts EOT# input. EOT# input should be asserted only when the PCI 9054 owns a bus.

If BLAST# output is not required for the last Lword of the DMA transfer (DMAMODE0[15]=1 and/or DMAMODE1[15]=1), the DMA controller releases the data bus and terminates DMA after it receives an external READY#. Or, the internal wait state counter decrements to 0 for the current Lword. If the DMA controller is currently bursting data that is not the last Data phase for the burst, BLAST# output is not asserted.

If BLAST# output is required for last Lword of the DMA transfer (DMAMODE0[15]=0 and/or DMAMODE1[15]=0), the DMA controller transfers one or two Lwords, depending on the Local Bus width. If EOT# is asserted, the DMA controller completes the current Lword and one additional Lword (this allows BLAST# output to be asserted during the final Lword). If the DMA FIFO is full or empty after the Data phase in which EOT# is asserted, the second Lword is not transferred.

The DMA controller terminates a transfer on an Lword boundary after EOT# is asserted. For an 8-bit bus, the PCI 9054 terminates after transferring the last byte for the Lword. For a 16-bit bus, the PCI 9054 terminates after transferring the last word for the Lword.

During the descriptor loading on the Local Bus, assertion of EOT# causes a complete descriptor load and no subsequent Data transfer; however, this is not recommended. This has no effect when the descriptor is loaded from the PCI Bus.

### 5.5.10 DMA Arbitration

The PCI 9054 DMA controller releases control of the Local Bus (de-asserts LHOLD) when one of the following conditions occur:

- Local Bus Latency Timer is enabled and expires (MARBR[7:0])
- BREQi is asserted (BREQi can be enabled or disabled, or gated with a Local Bus Latency Timer before the PCI 9054 releases the Local Bus)
- PCI Target access is pending
- EOT# input is received (if enabled)

The DMA controller releases control of the PCI Bus when one of the following conditions occurs:

- FIFOs are full or empty
- PCI Bus Latency Timer expires (PCILTR [7:0])—and loses the PCI GNT# signal
- Target disconnect response is received

The DMA controller de-asserts its PCI Bus request (REQ#) for a minimum of two PCI clocks.

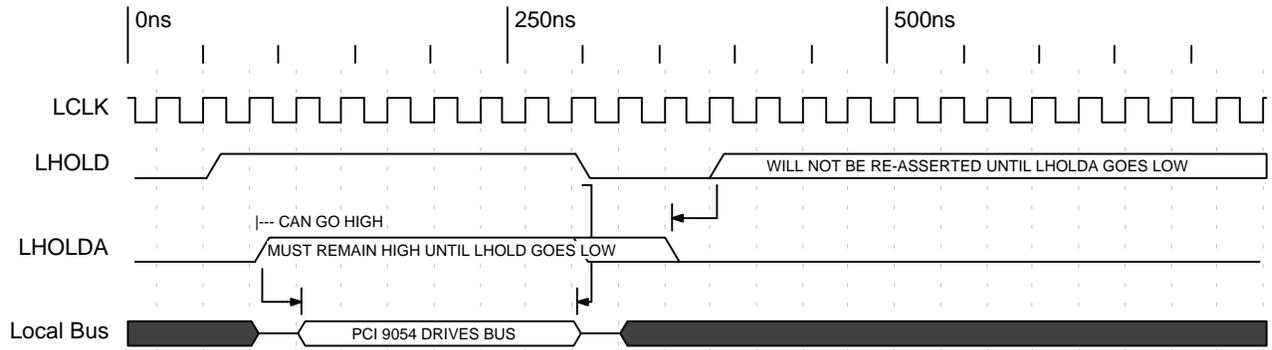
### 5.5.11 Local Bus Latency and Pause Timers

The Local Bus Latency and Pause Timers are programmable with the Mode/DMA Arbitration register (MARBR[7:0, 15:8]). If the Local Bus Latency Timer is enabled and expires (MARBR[7:0]), the PCI 9054 completes the current Lword transfer and releases LHOLD. After its programmable Pause Timer expires (MARBR[7:0]), it reasserts LHOLD. It continues to transfer when it receives LHOLDA. The PCI Bus transfer continues until the FIFO is empty for a Local-to-PCI transfer or full for a PCI-to-Local transfer.

The DMA transfer can be paused by writing a 0 to the Channel Enable bit. To acknowledge the disable, the PCI 9054 gets at least one data from the bus before it stops. However, this is not recommended during a burst.

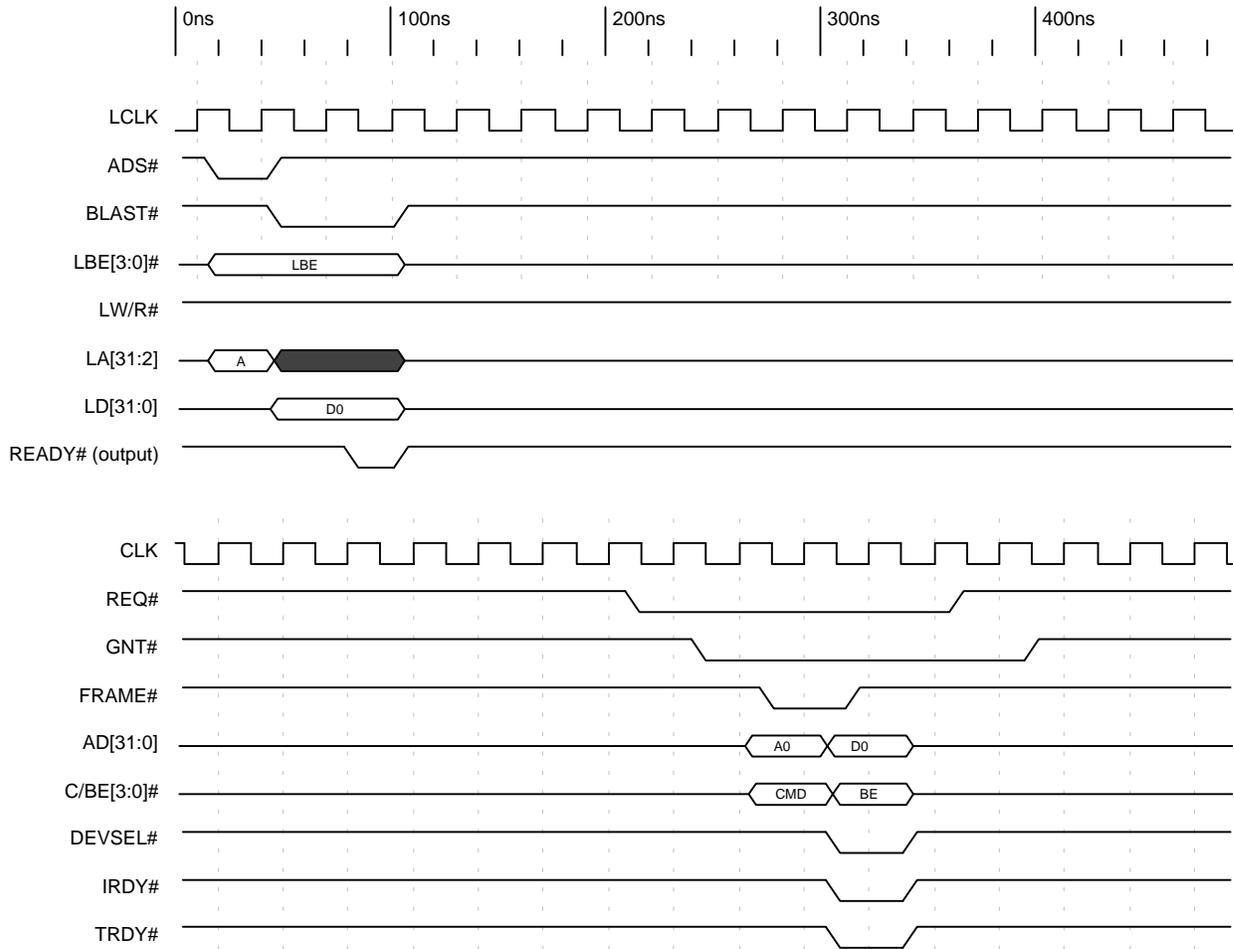
The DMA Local Bus Timer starts after the Local Bus is granted to the PCI 9054 and the Local Pause Timer starts after LHOLDA is de-asserted.

### 5.6 C MODE TIMING DIAGRAMS

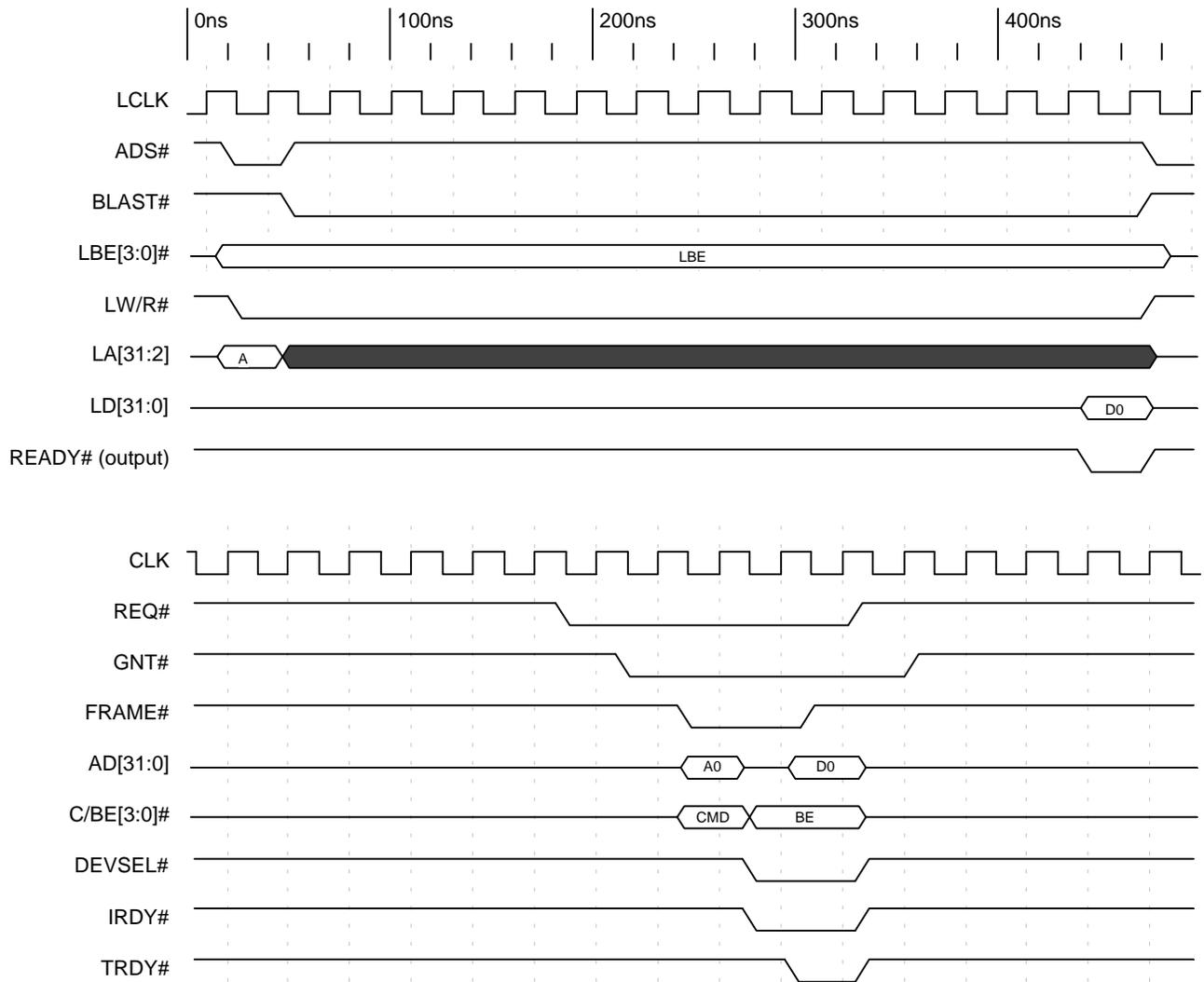


Timing Diagram 5-1. Local Bus Arbitration (LHOLD and LHOLDA)

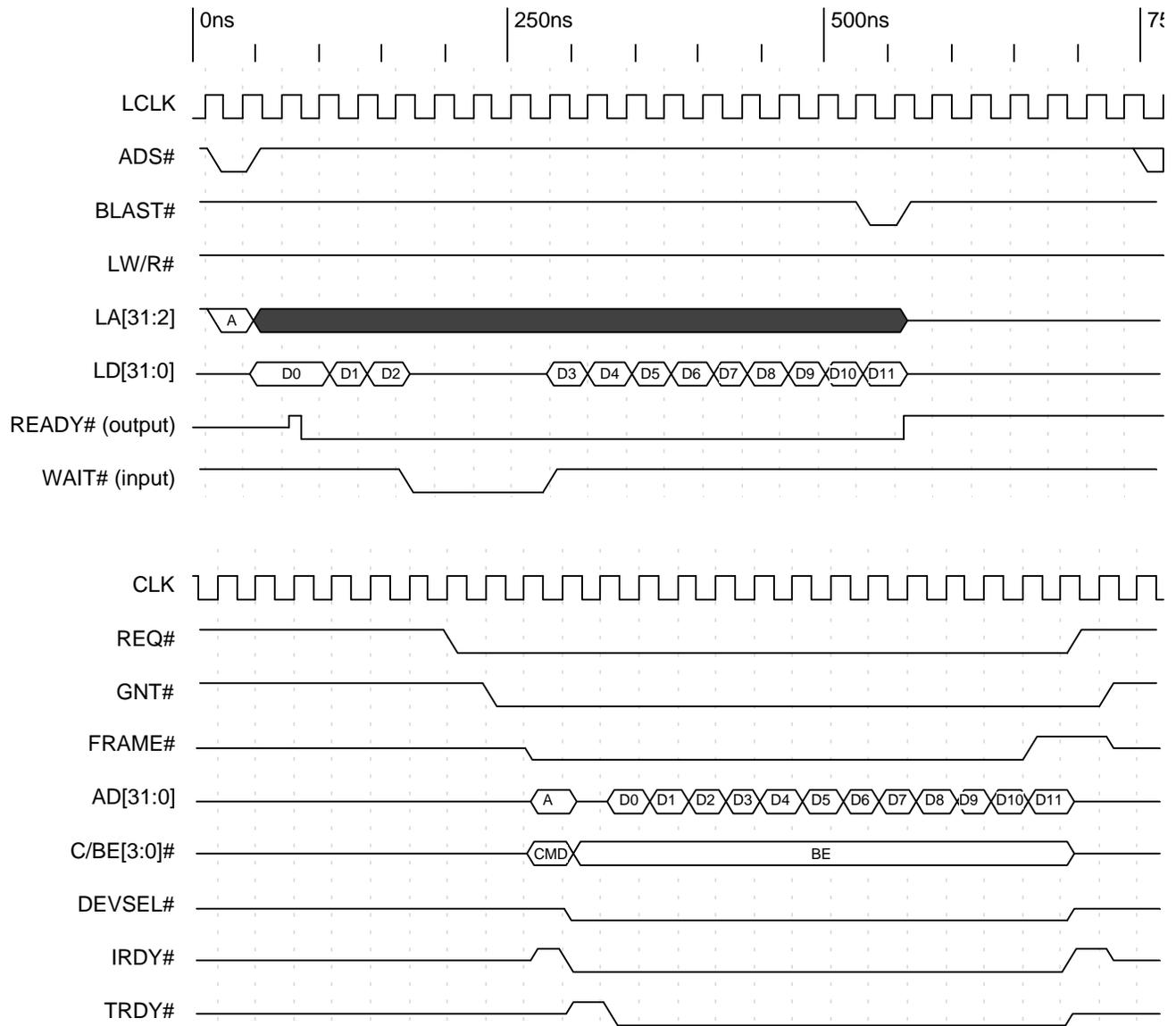
5.6.1 C Mode PCI Initiator



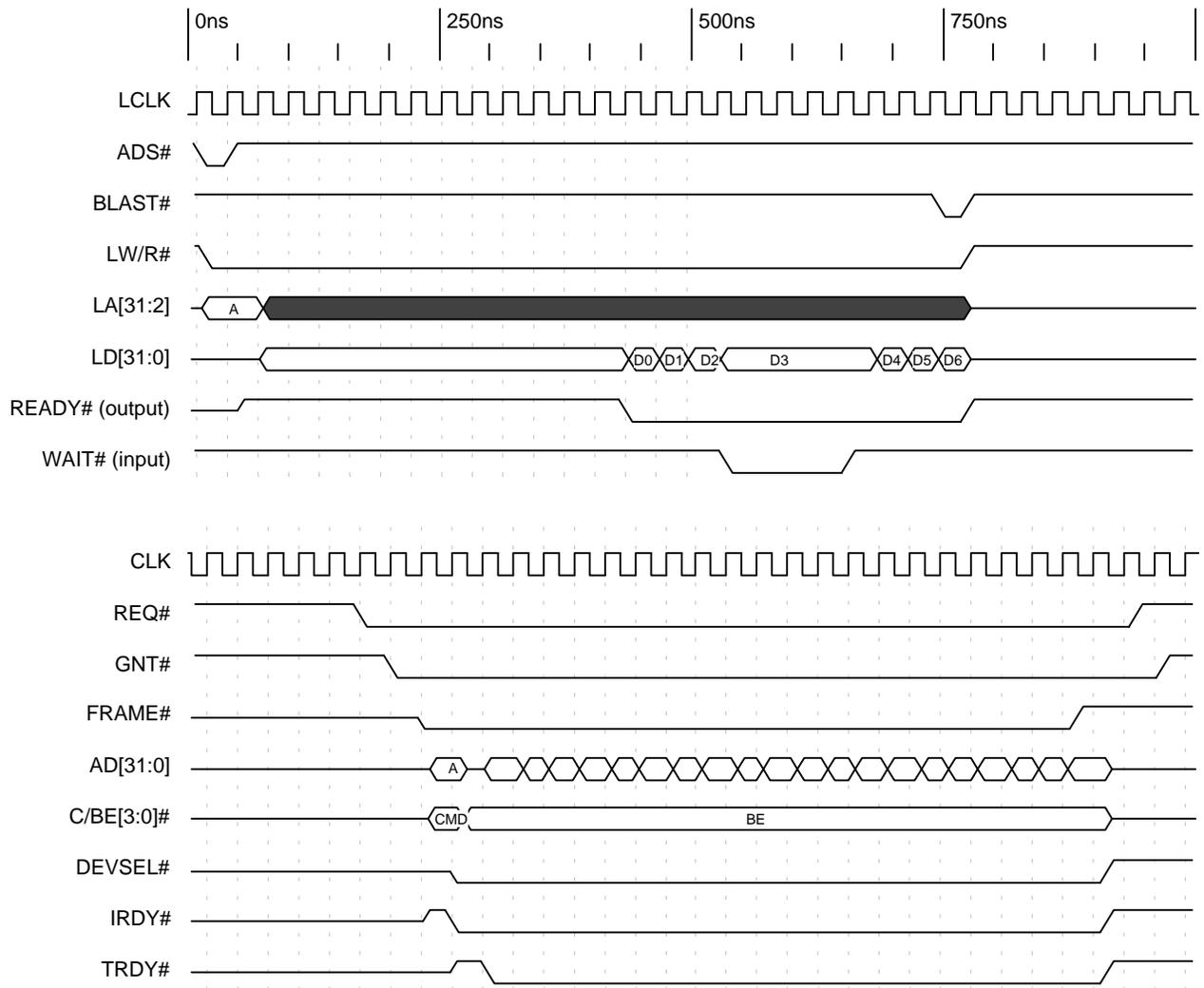
Timing Diagram 5-2. PCI Initiator Single Write



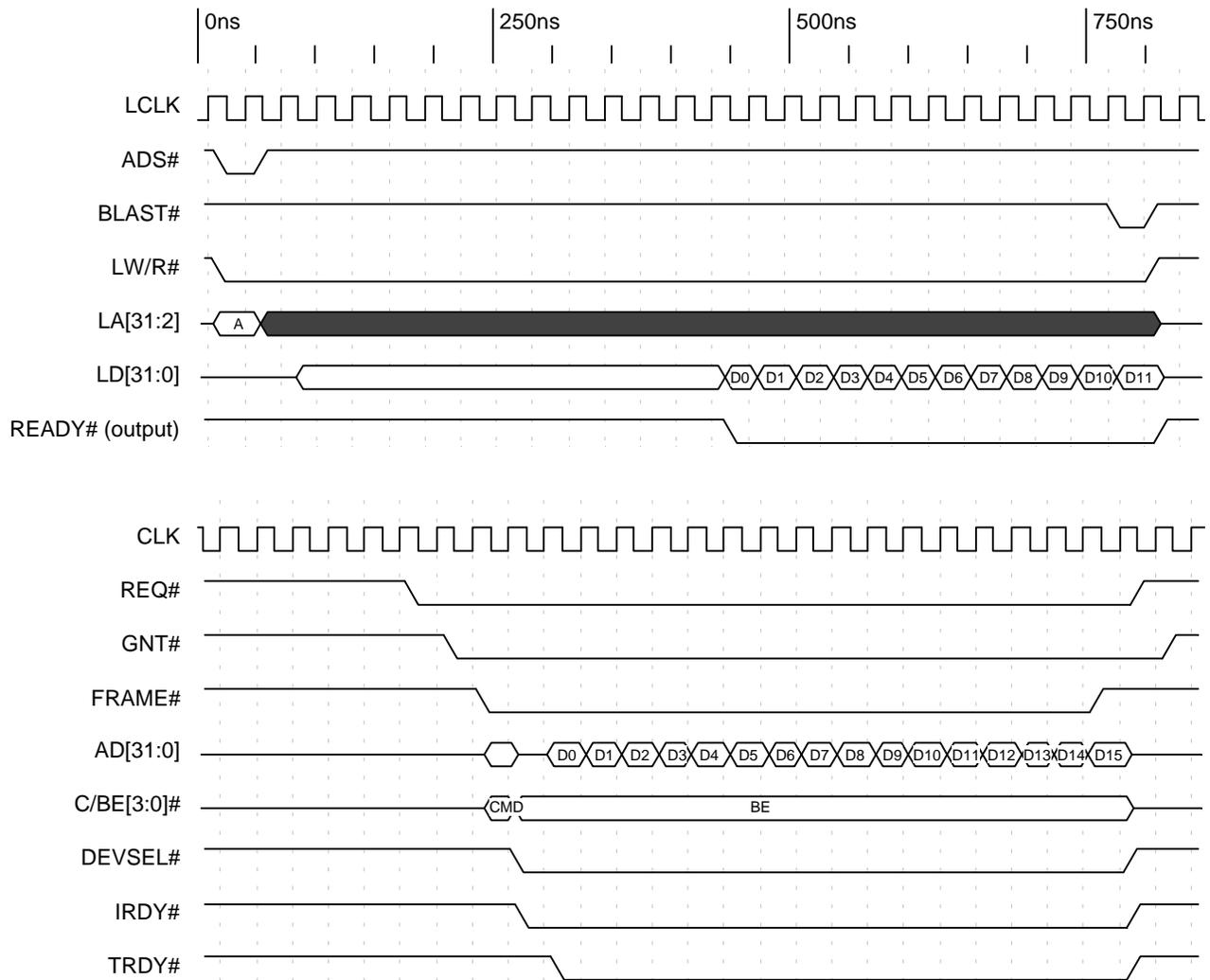
Timing Diagram 5-3. PCI Initiator Single Read



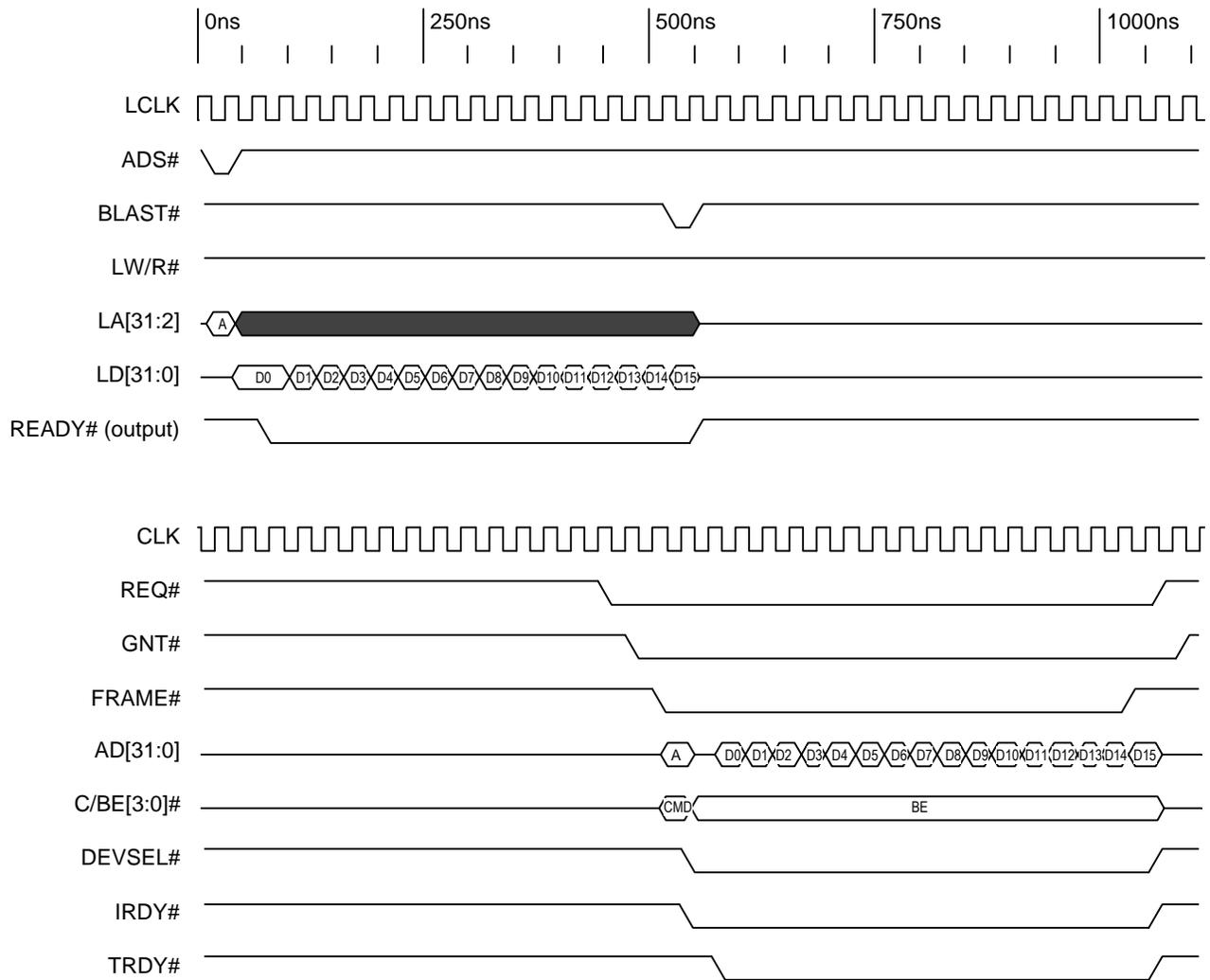
Timing Diagram 5-4. PCI Initiator Memory Write of 12 Lwords with WAIT# Input



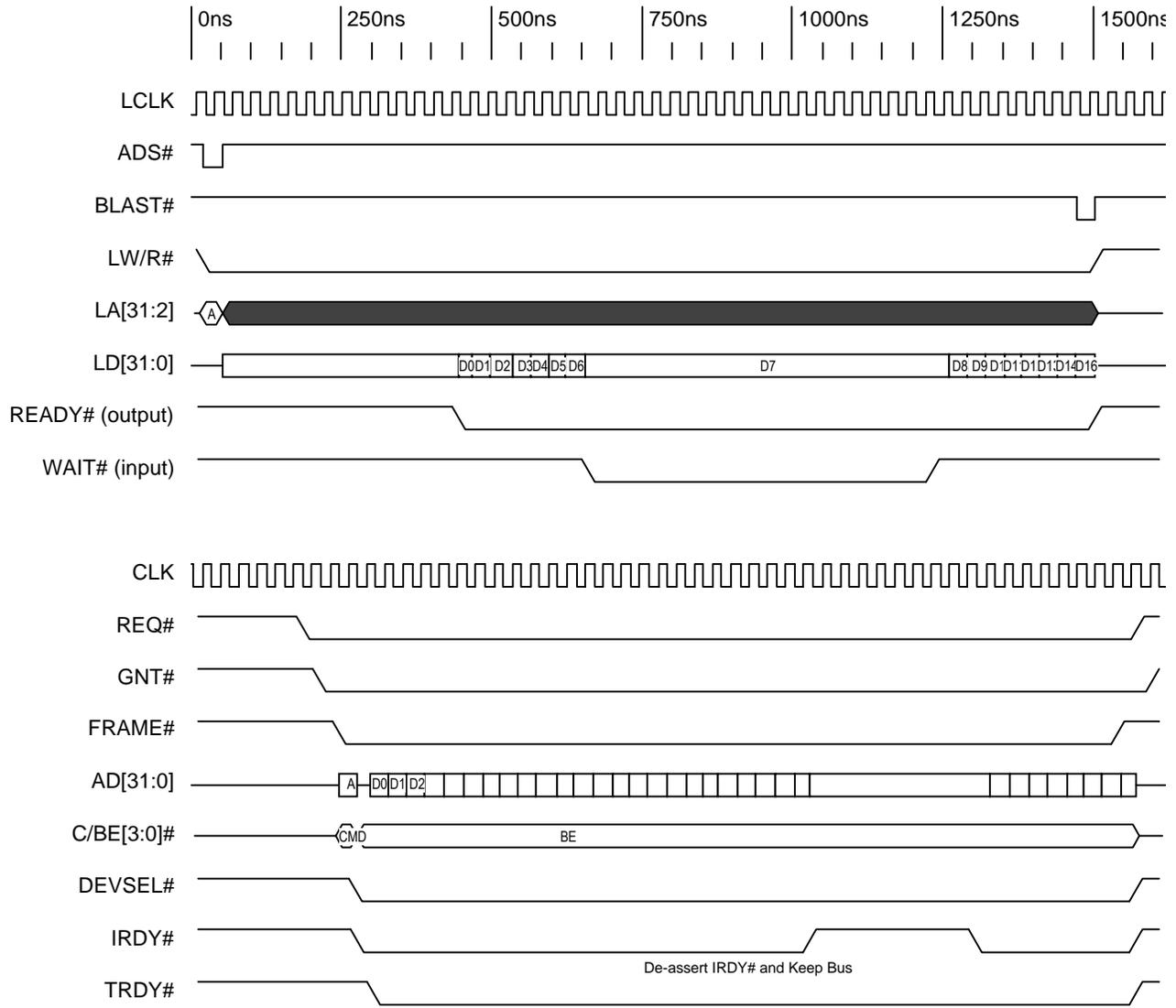
Timing Diagram 5-5. PCI Initiator Burst Read of Seven Lwords with WAIT# Input



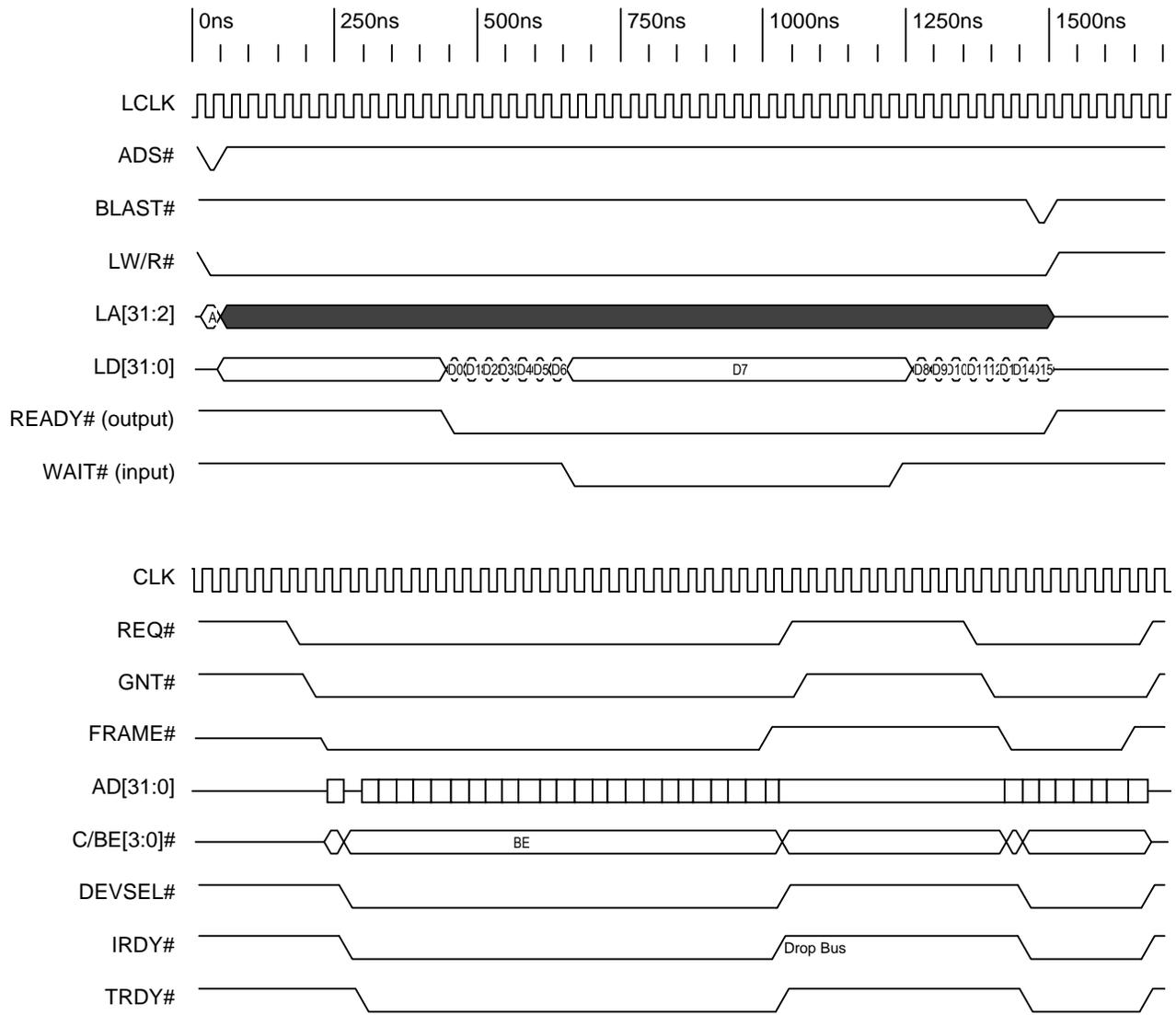
Timing Diagram 5-6. PCI Initiator Memory Read of 12 Lwords with Prefetch Counter Set to 16



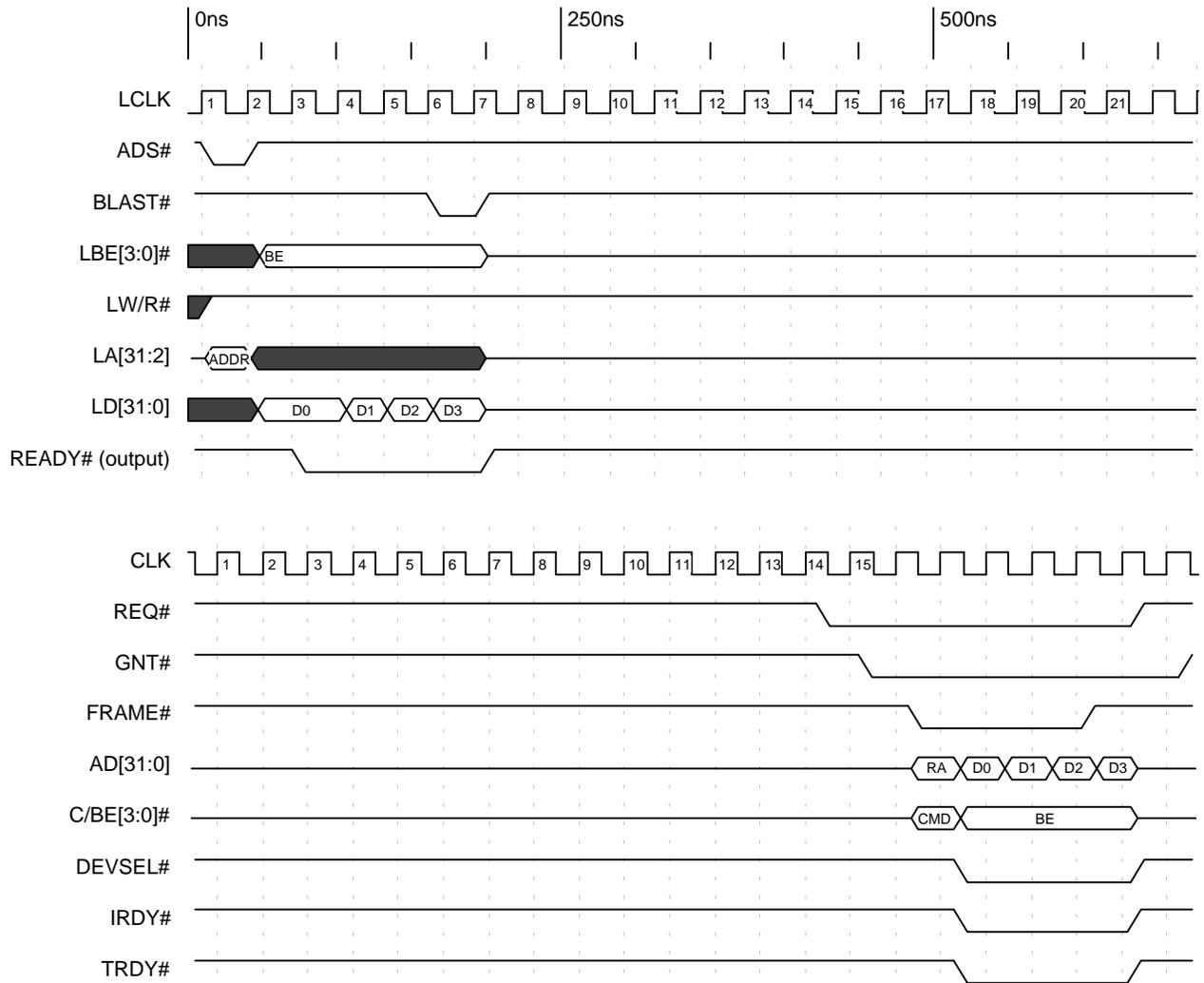
Timing Diagram 5-7. Memory Write and Invalidate with Cache Line Size of Eight



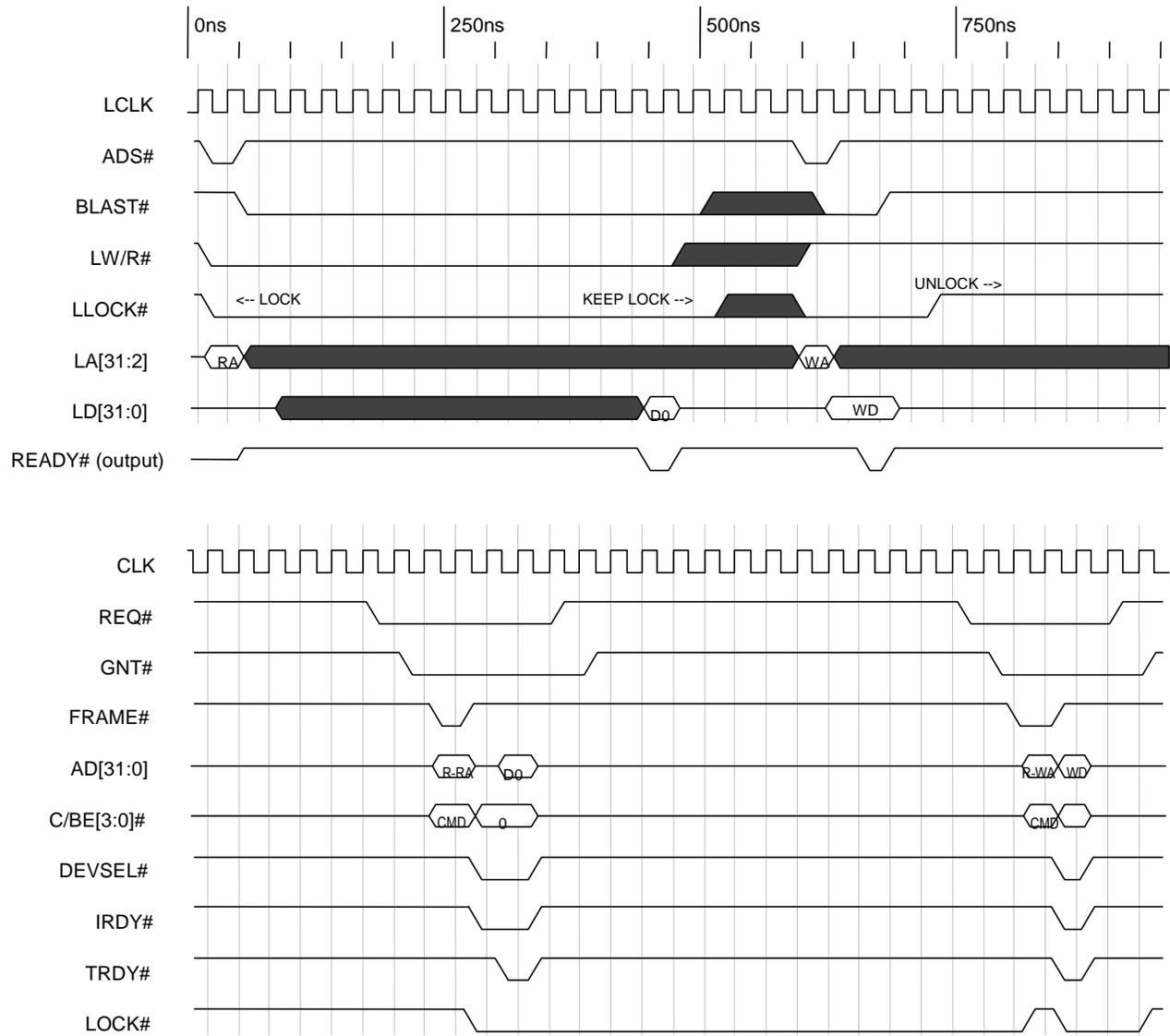
Timing Diagram 5-8. PCI Initiator Memory Read with Keep Bus Mode



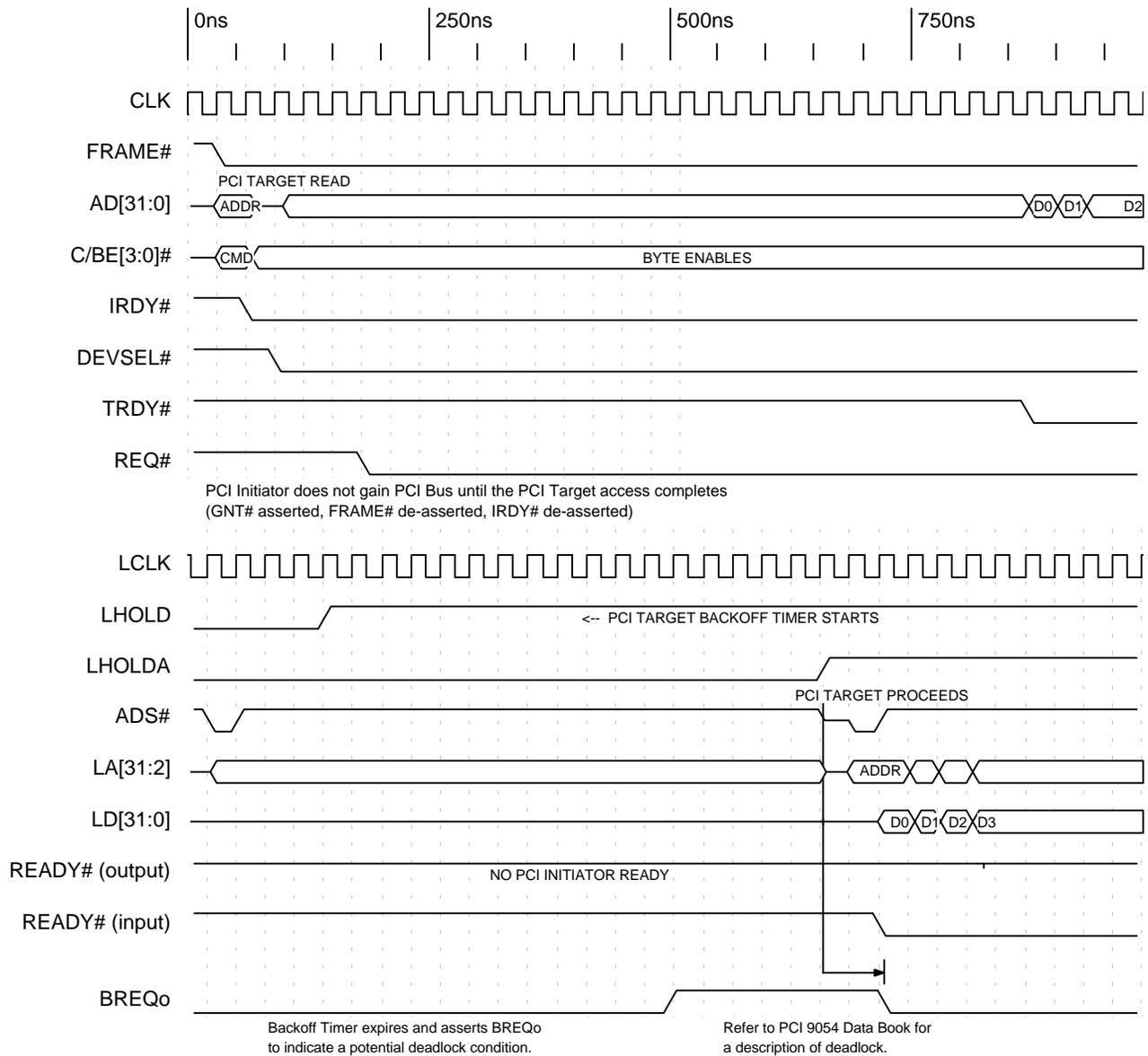
Timing Diagram 5-9. PCI Initiator Memory Read with Drop Bus Mode



Timing Diagram 5-10. PCI Bus Request (REQ#) Delay During Direct Master Write (Eight-PCI Clock Delay)

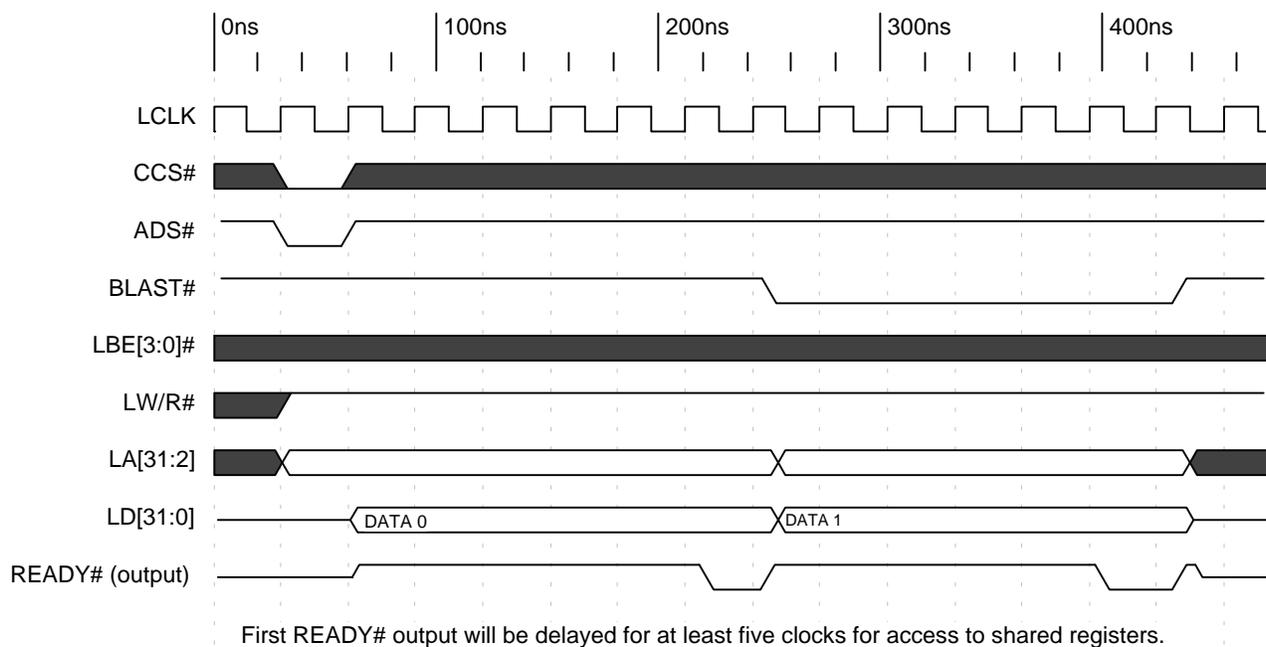


Timing Diagram 5-11. PCI Initiator Locked Read Followed by Write and Release (LLOCK# and LOCK#)

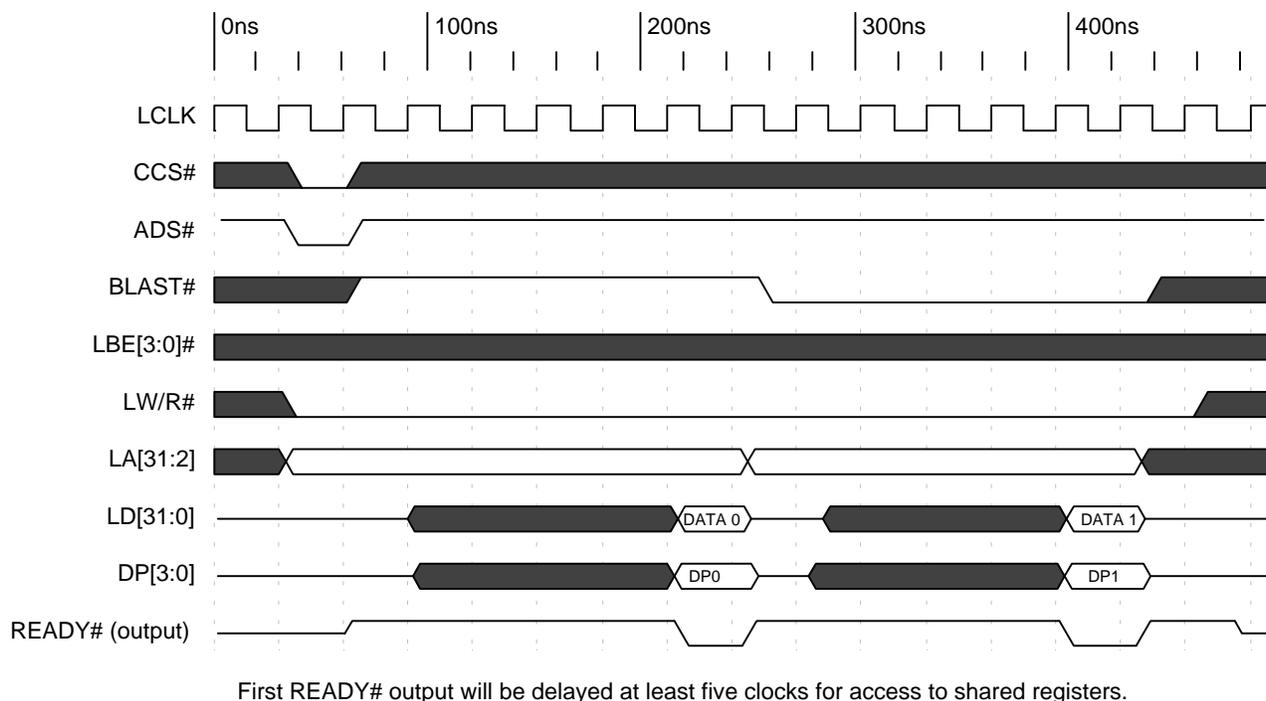


Note: For partial deadlock, PCI Target Retry Delay Clock bits (LBRD0[31:28]) can be used to issue Retrys to the PCI Master attempting the PCI Target access.

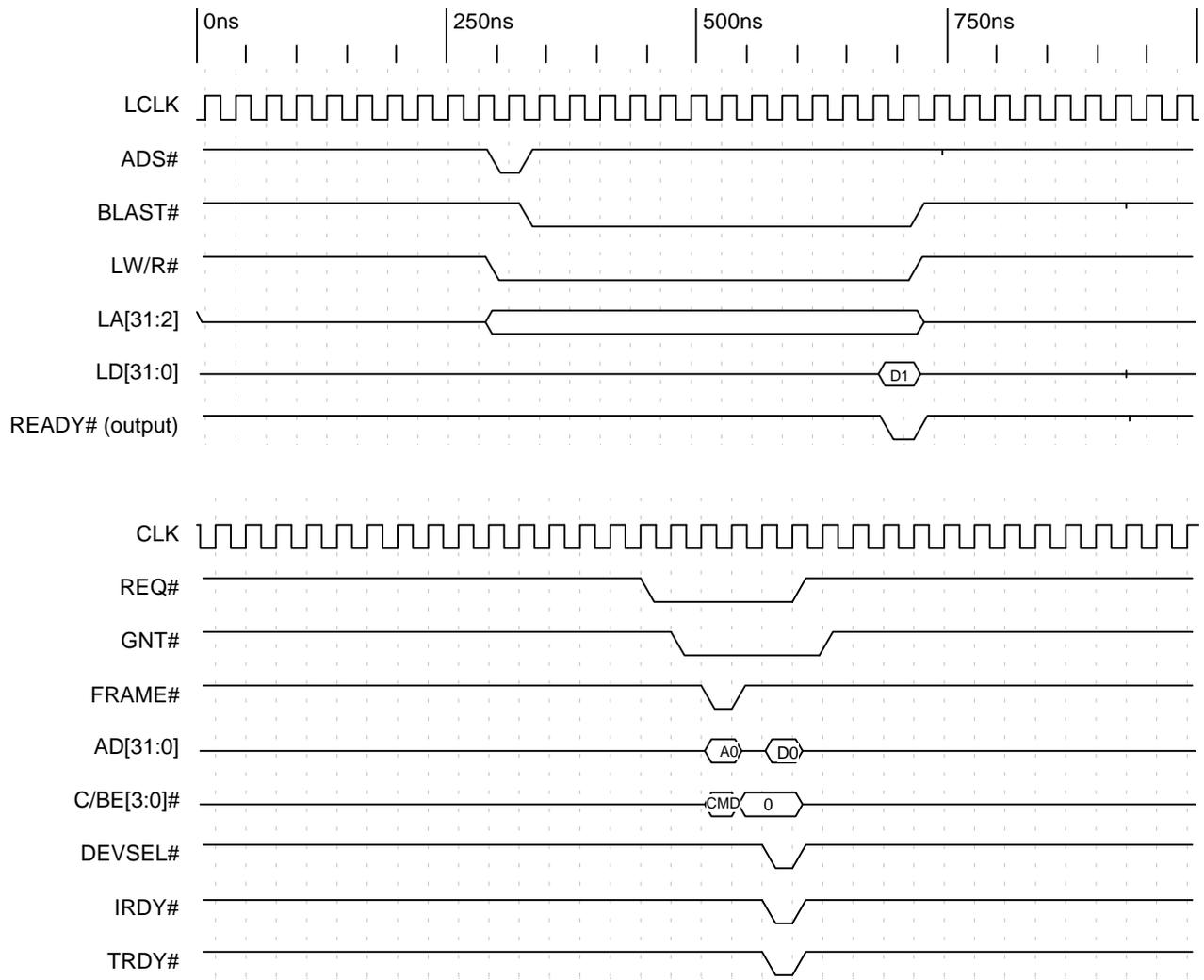
Timing Diagram 5-12. BREQo and Deadlock



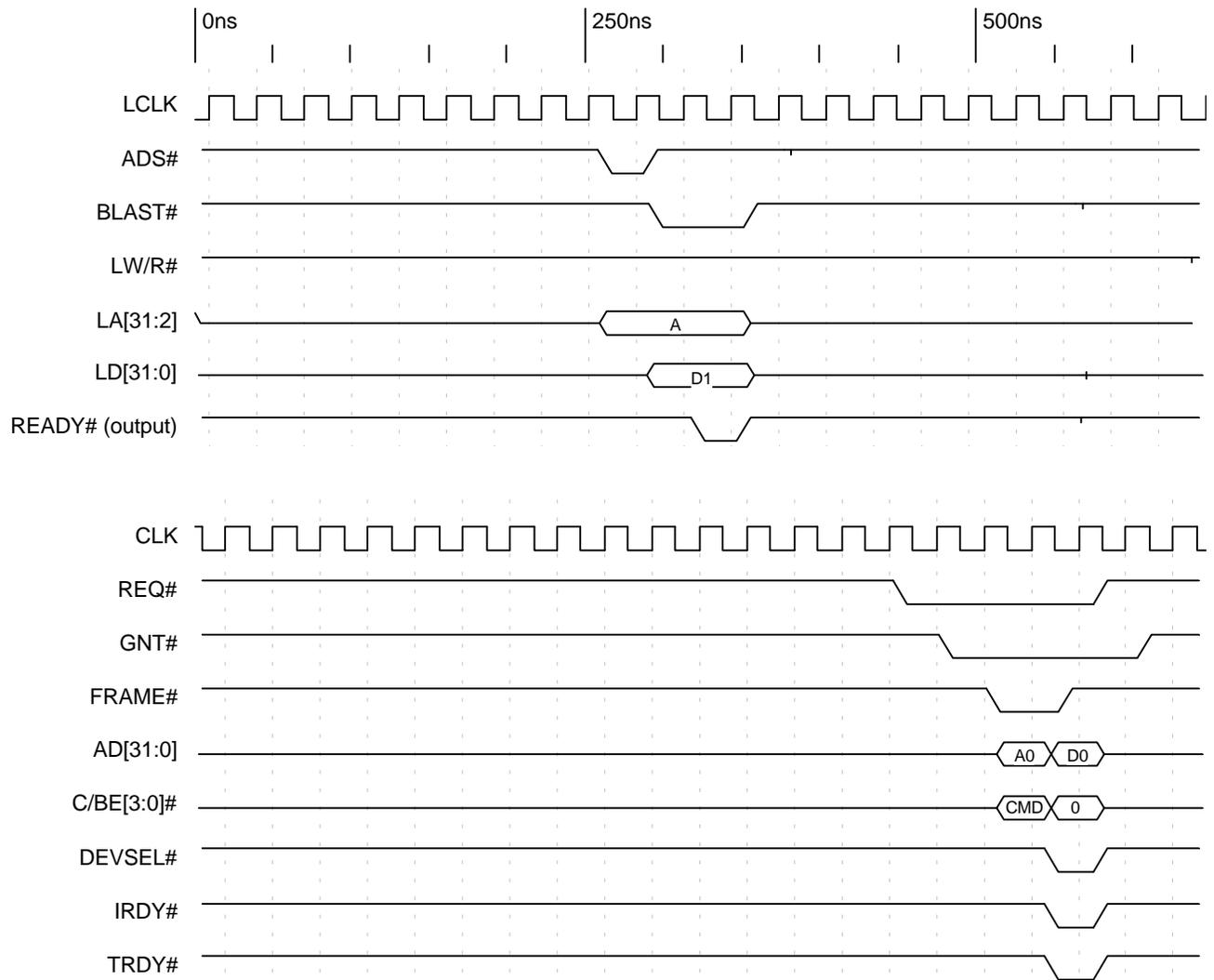
**Timing Diagram 5-13. Local Bus Write to Configuration Register**



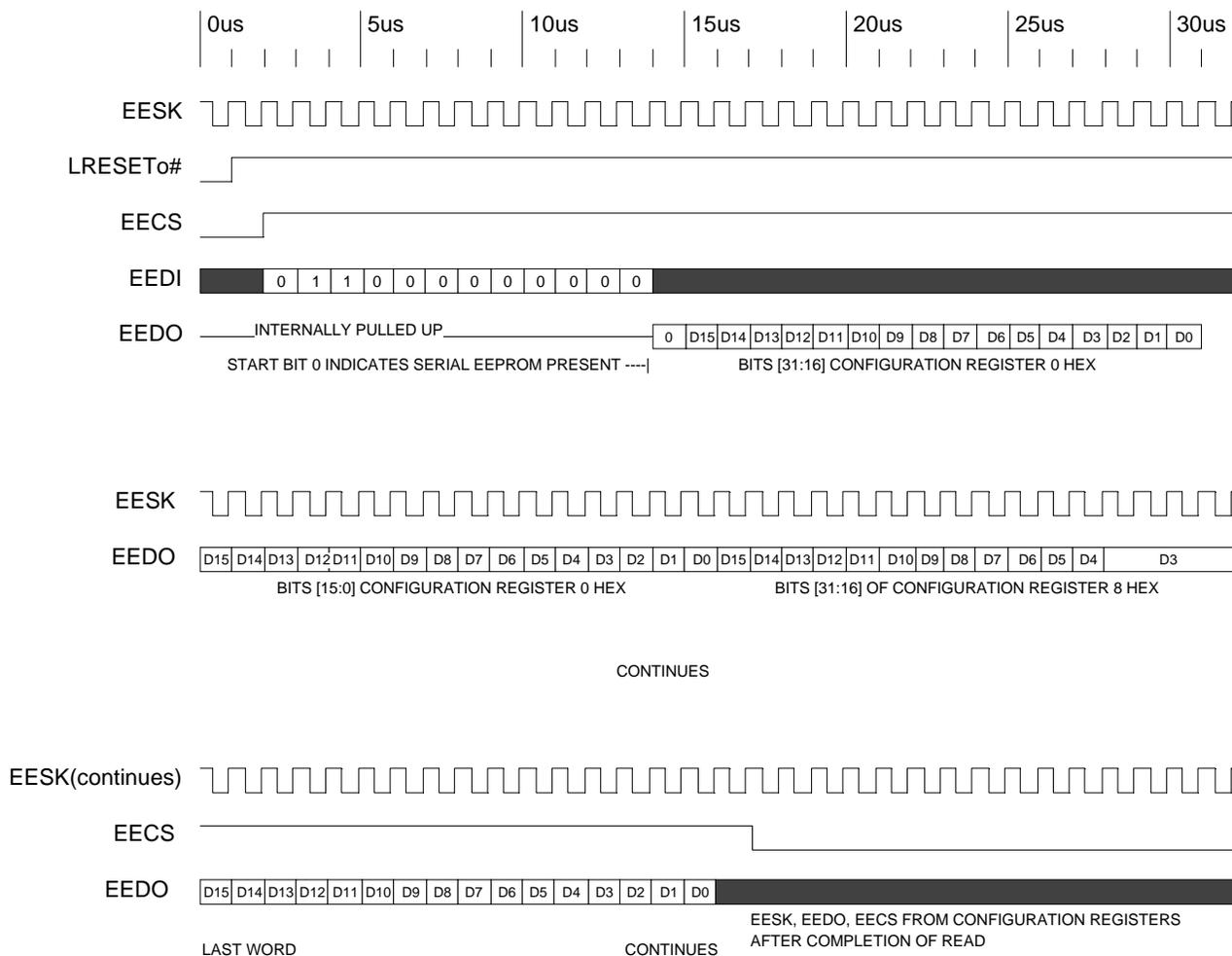
**Timing Diagram 5-14. Local Bus Read to Configuration Register**



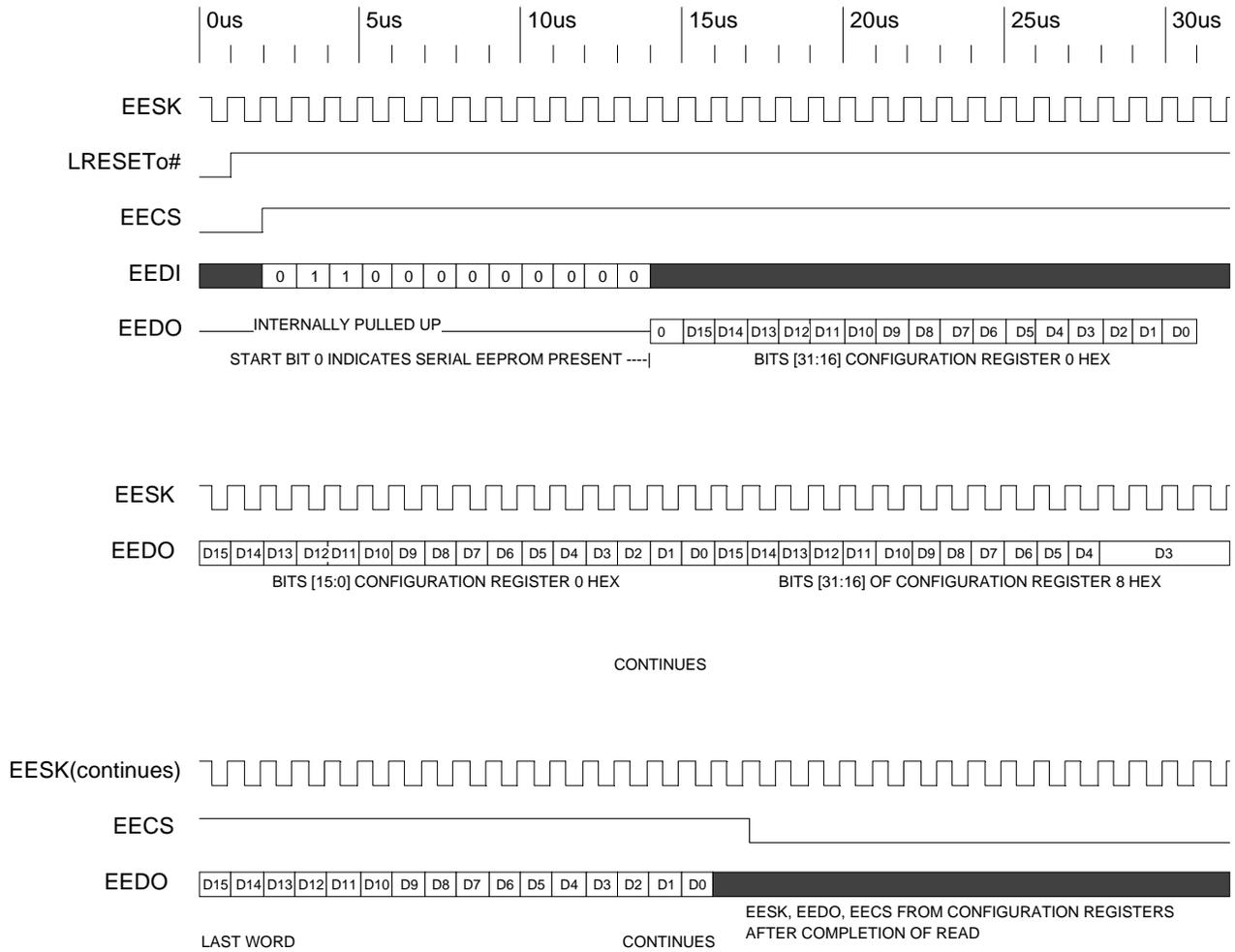
Timing Diagram 5-15. PCI Initiator Configuration Read—Type 1 or Type 0



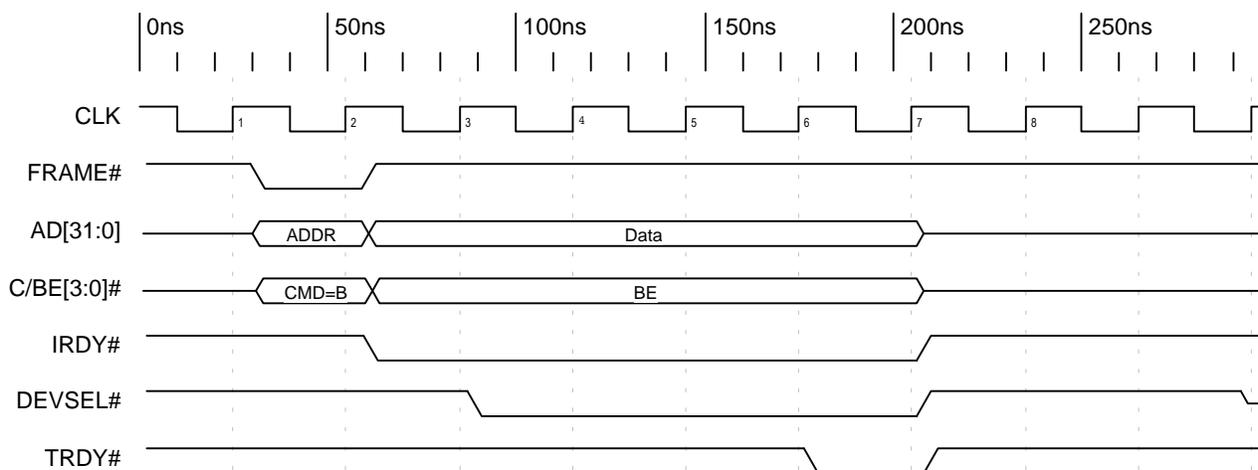
Timing Diagram 5-16. PCI Initiator Configuration Write—Type 1 or Type 0



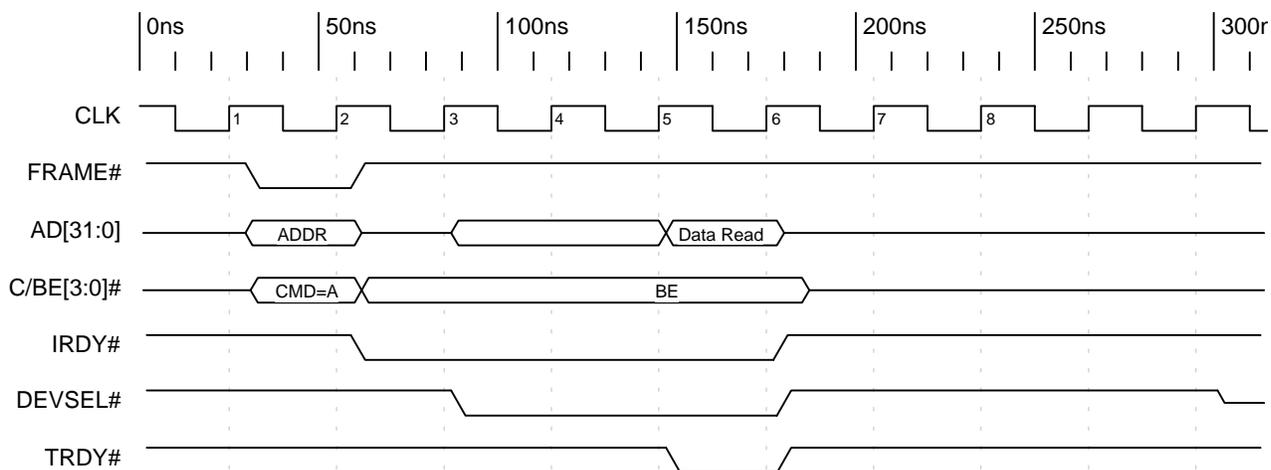
Timing Diagram 5-17. Initialization from Serial EEPROM (2K Bit)



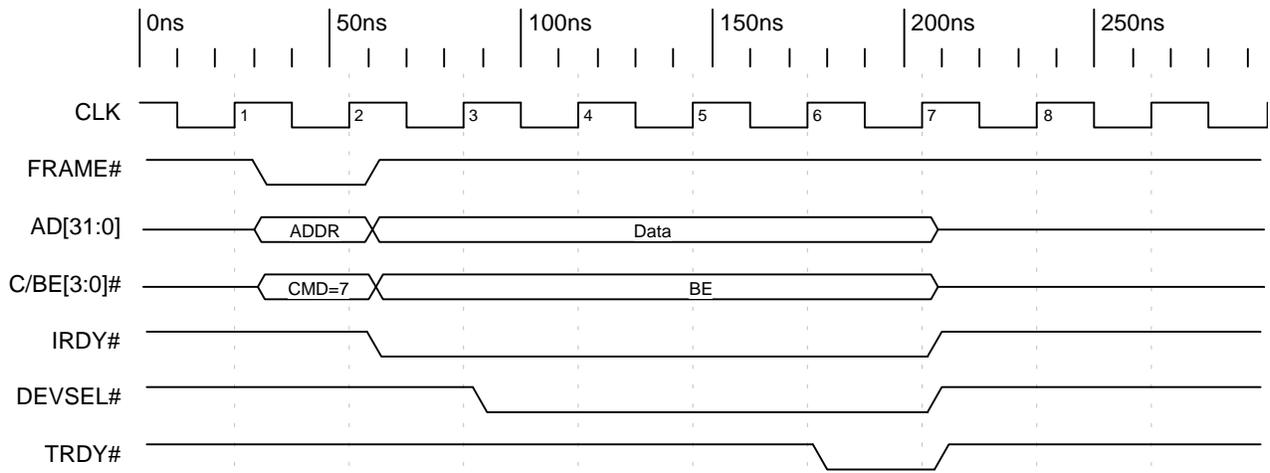
Timing Diagram 5-18. Initialization from Serial EEPROM (4K Bit)



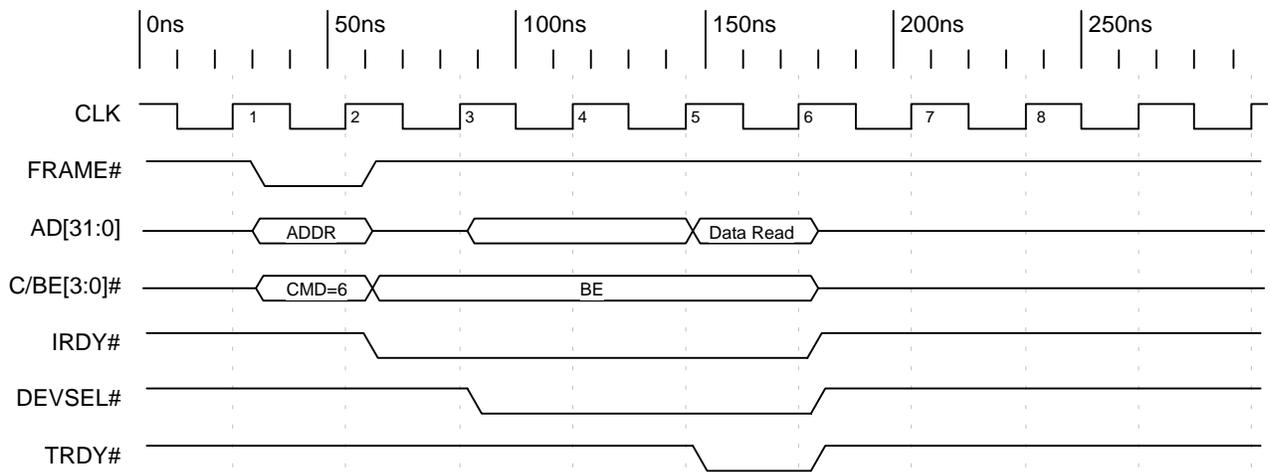
Timing Diagram 5-19. PCI Configuration Write to PCI Configuration Register



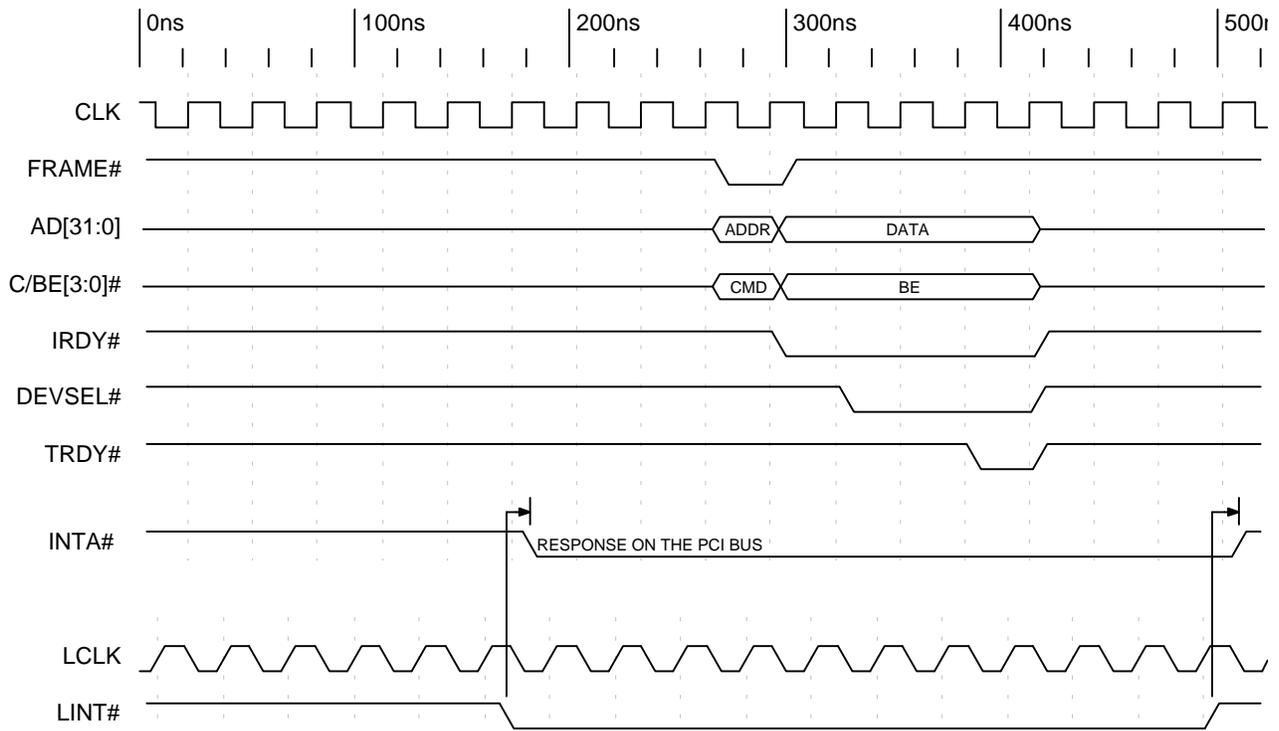
Timing Diagram 5-20. PCI Configuration Read to PCI Configuration Register



Timing Diagram 5-21. PCI Memory Write to Local Configuration Register

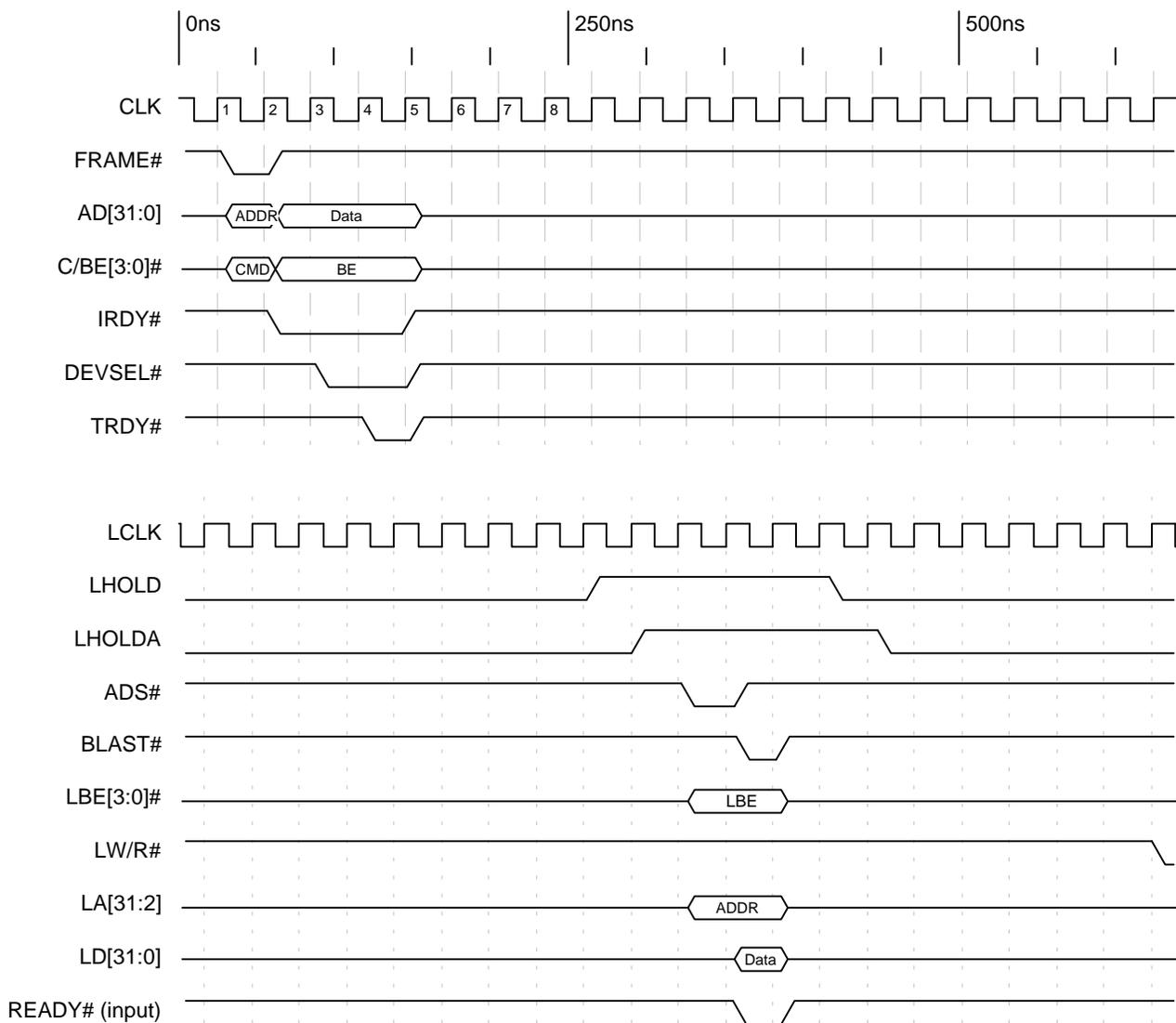


Timing Diagram 5-22. PCI Memory Read to Local Configuration Register

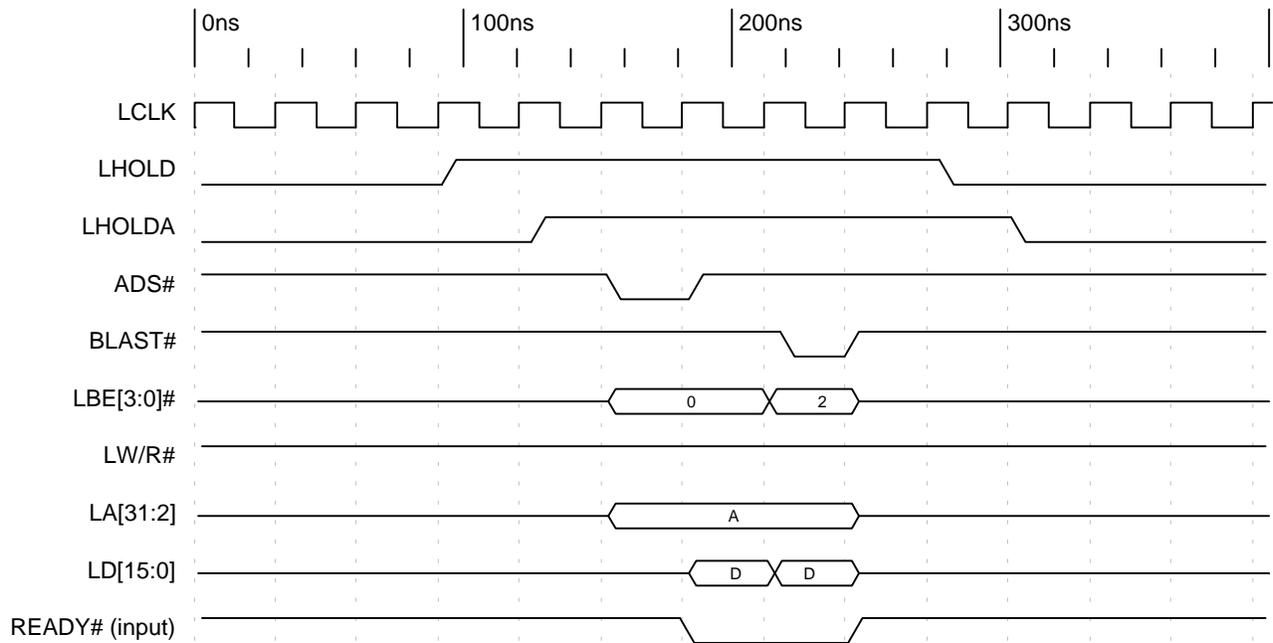


Timing Diagram 5-23. Local Interrupt Asserting PCI Interrupt

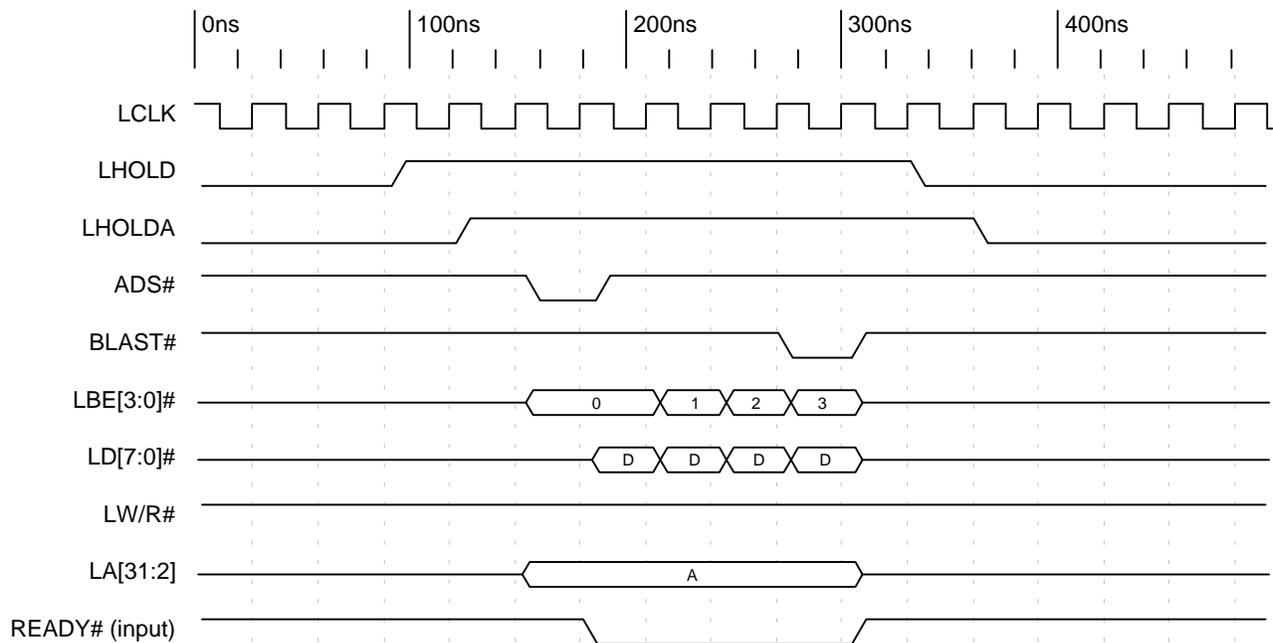
### 5.6.2 C Mode PCI Target



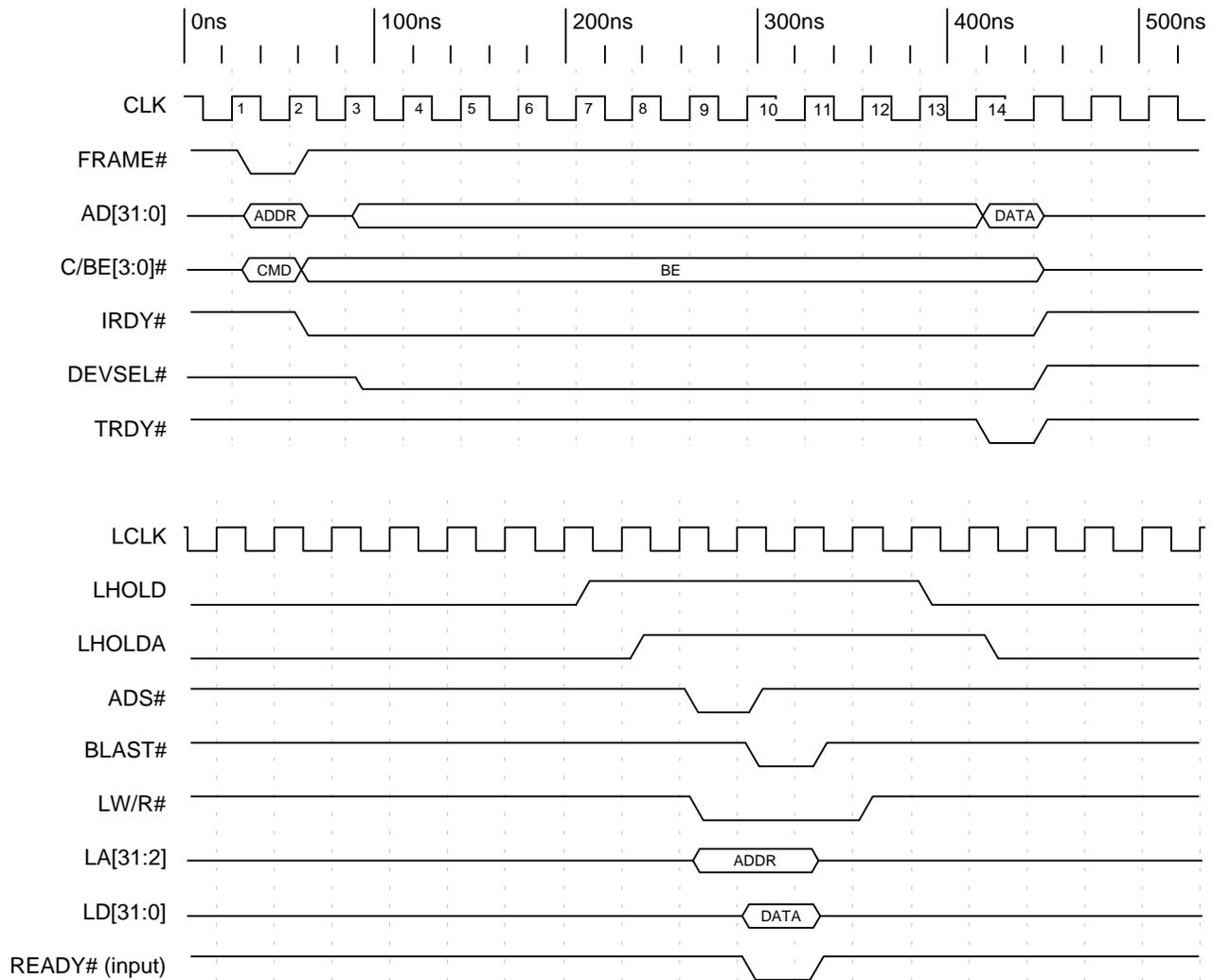
Timing Diagram 5-24. PCI Target Single Write (32-Bit Local Bus)



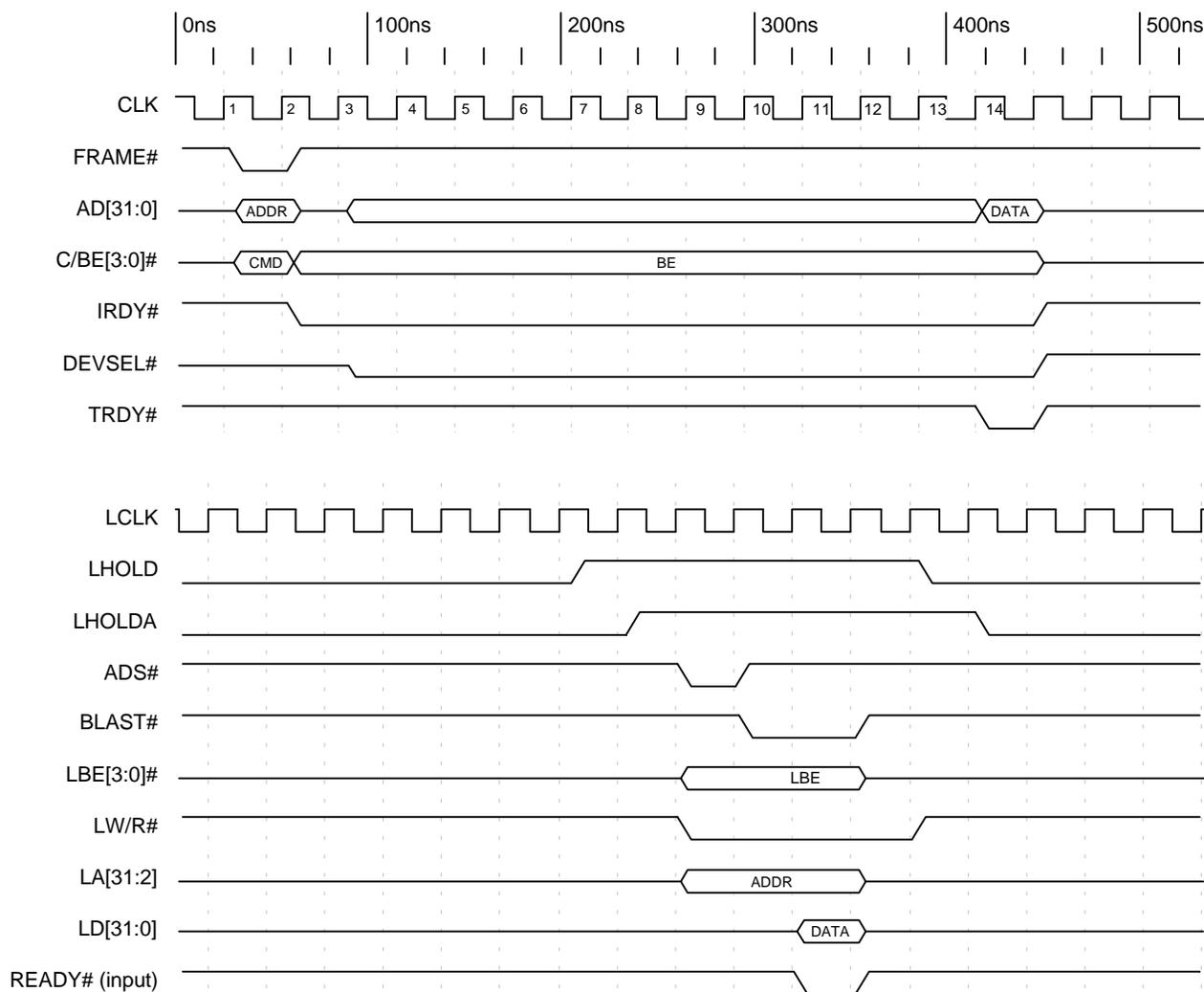
Timing Diagram 5-25. PCI Target Burst Cycle Write (16-Bit Local Bus)



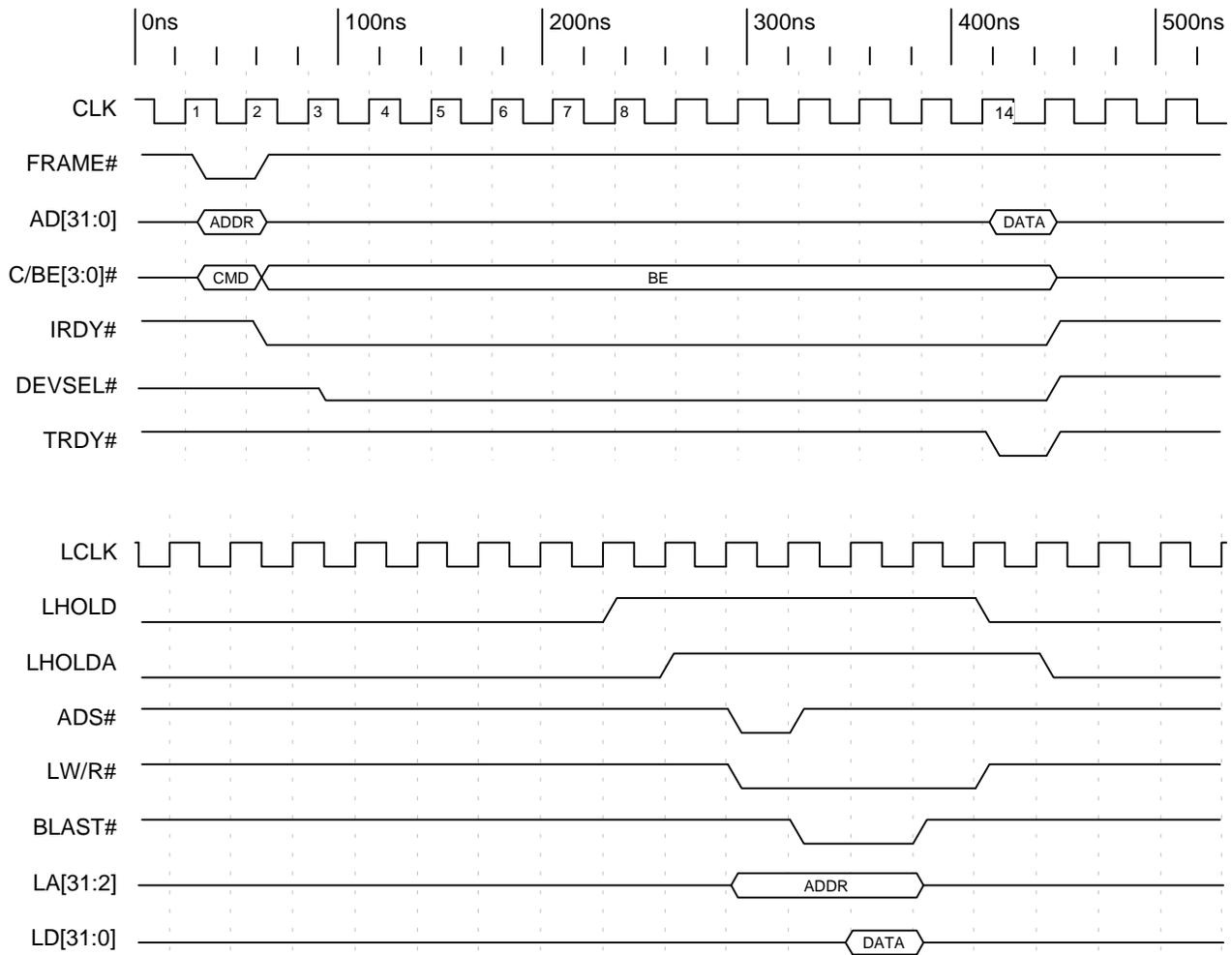
Timing Diagram 5-26. PCI Target Burst Cycle Write (8-Bit Local Bus)



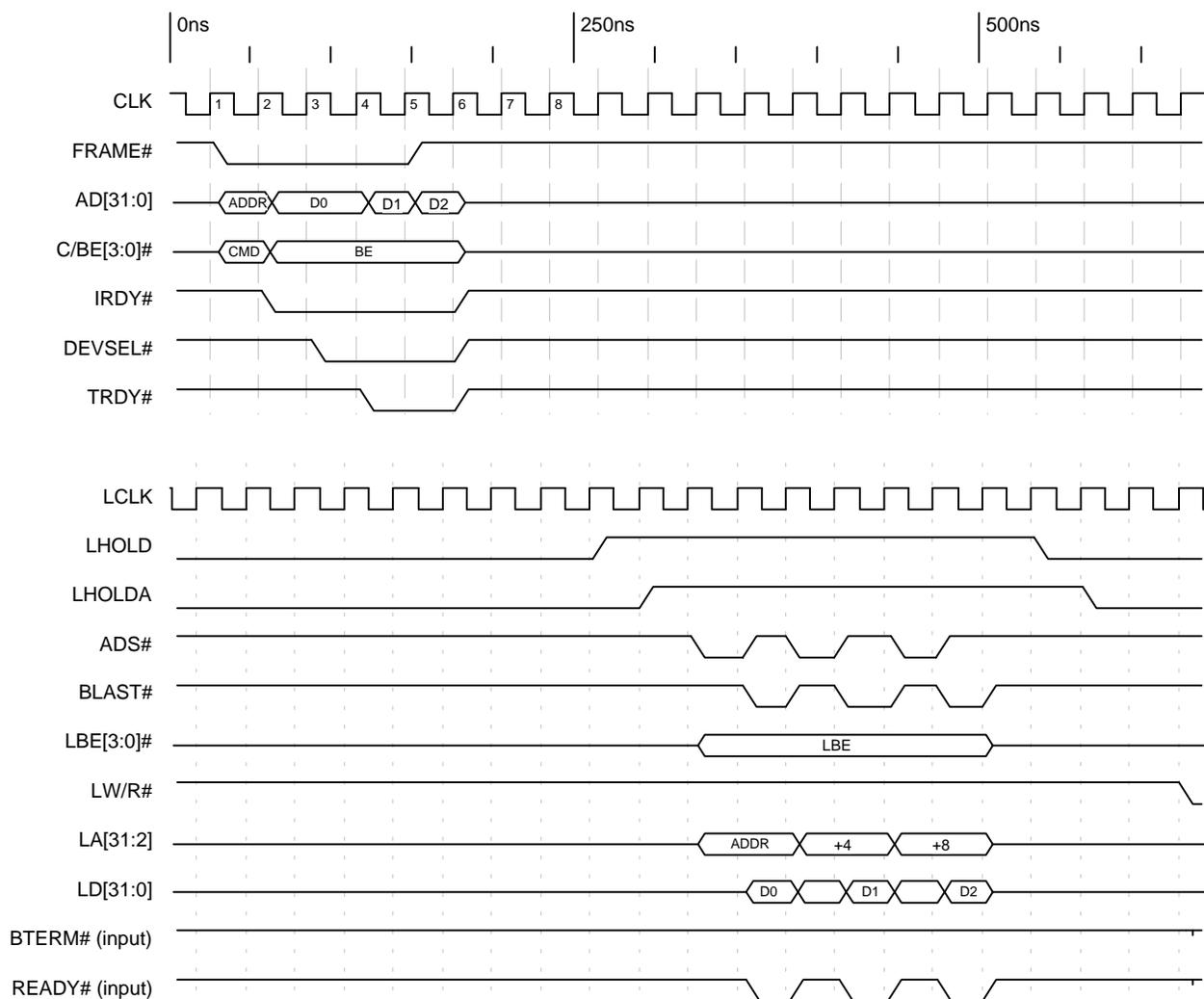
Timing Diagram 5-27. PCI Target Single-Cycle Read (32-Bit Local Bus)



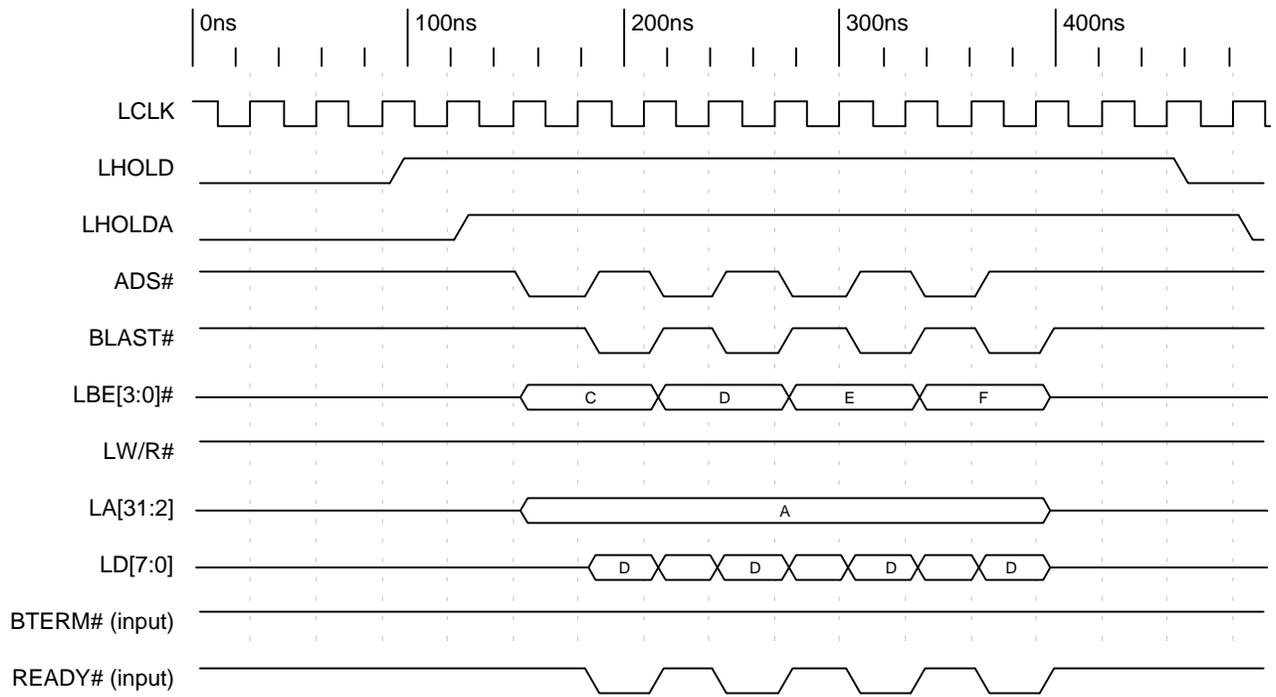
Timing Diagram 5-28. PCI Target Single Read with One Wait State Using READY# Input (32-Bit Local Bus)



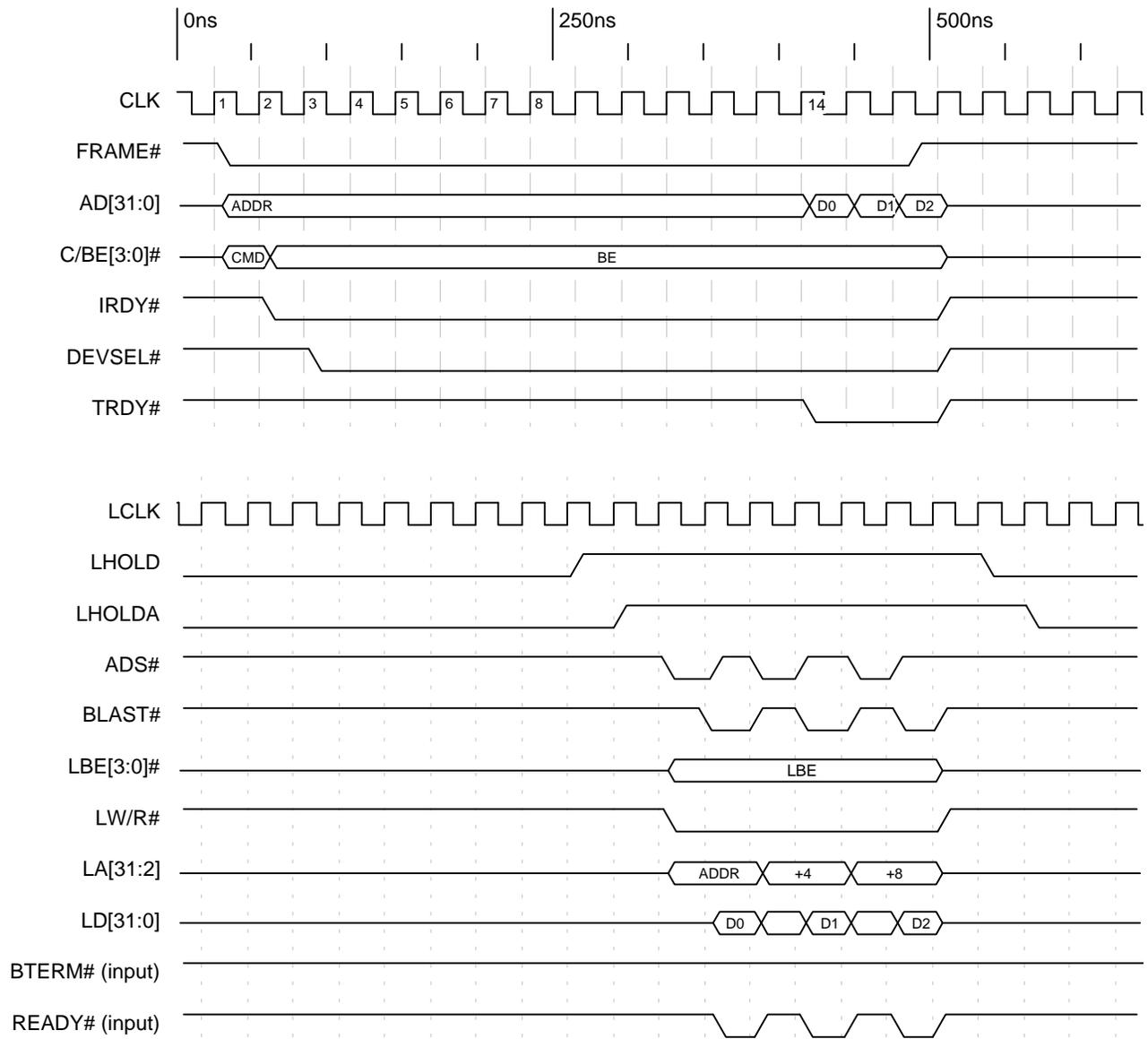
Timing Diagram 5-29. PCI Target Single Read with One Wait State Using Internal Wait State (32-Bit Local Bus)



Timing Diagram 5-30. PCI Target Non-Burst Write (32-Bit Local Bus)

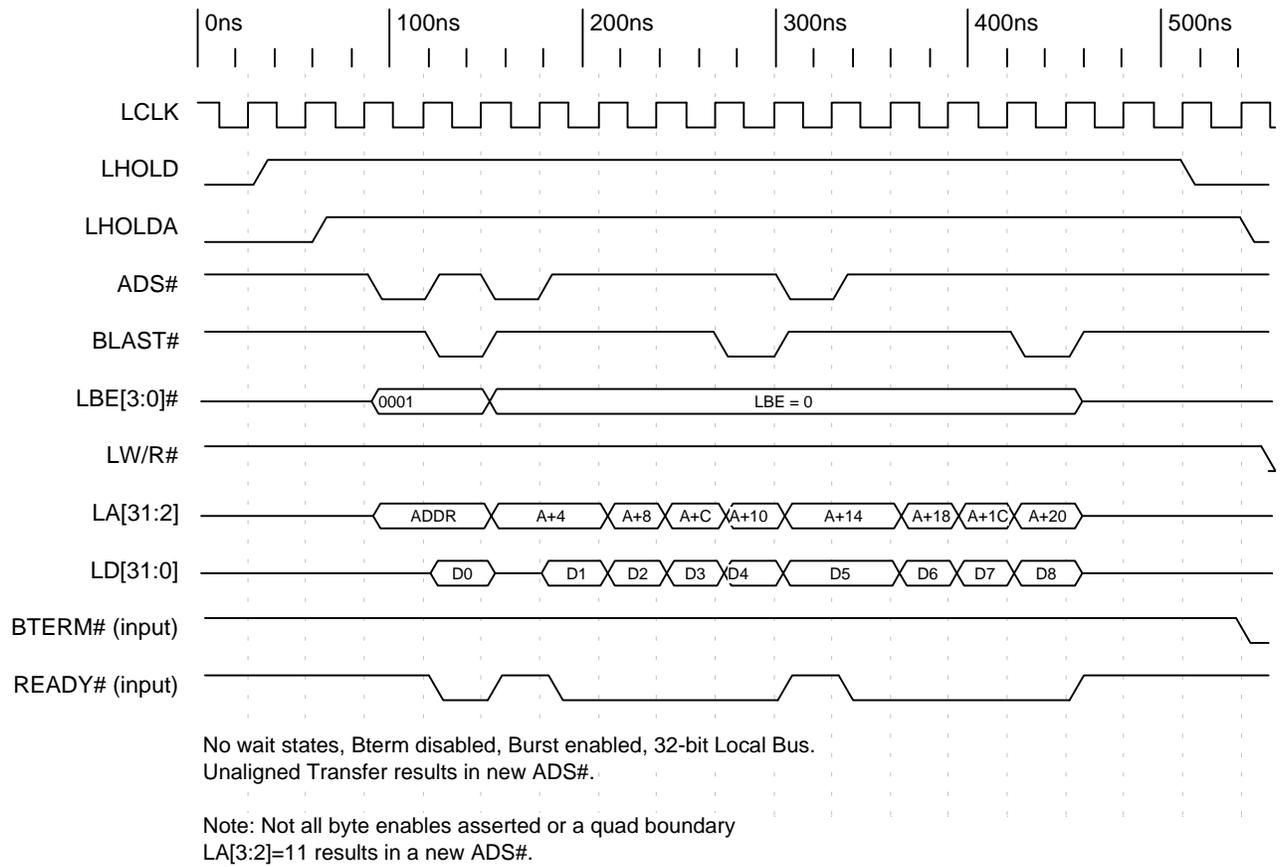


Timing Diagram 5-31. PCI Target Non-Burst Write (8-Bit Local Bus)

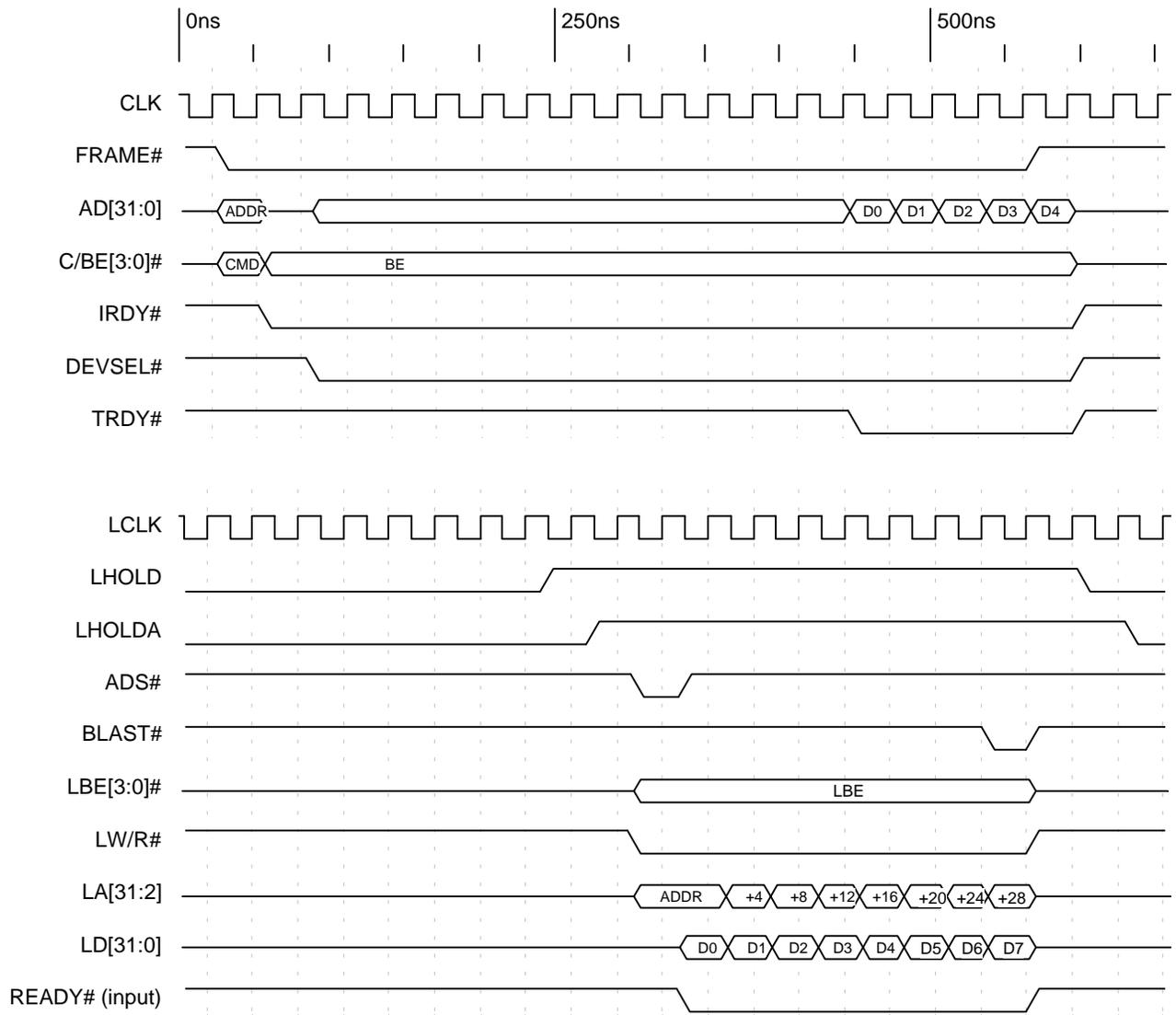


Timing Diagram 5-32. PCI Target Non-Burst Local Bus Read

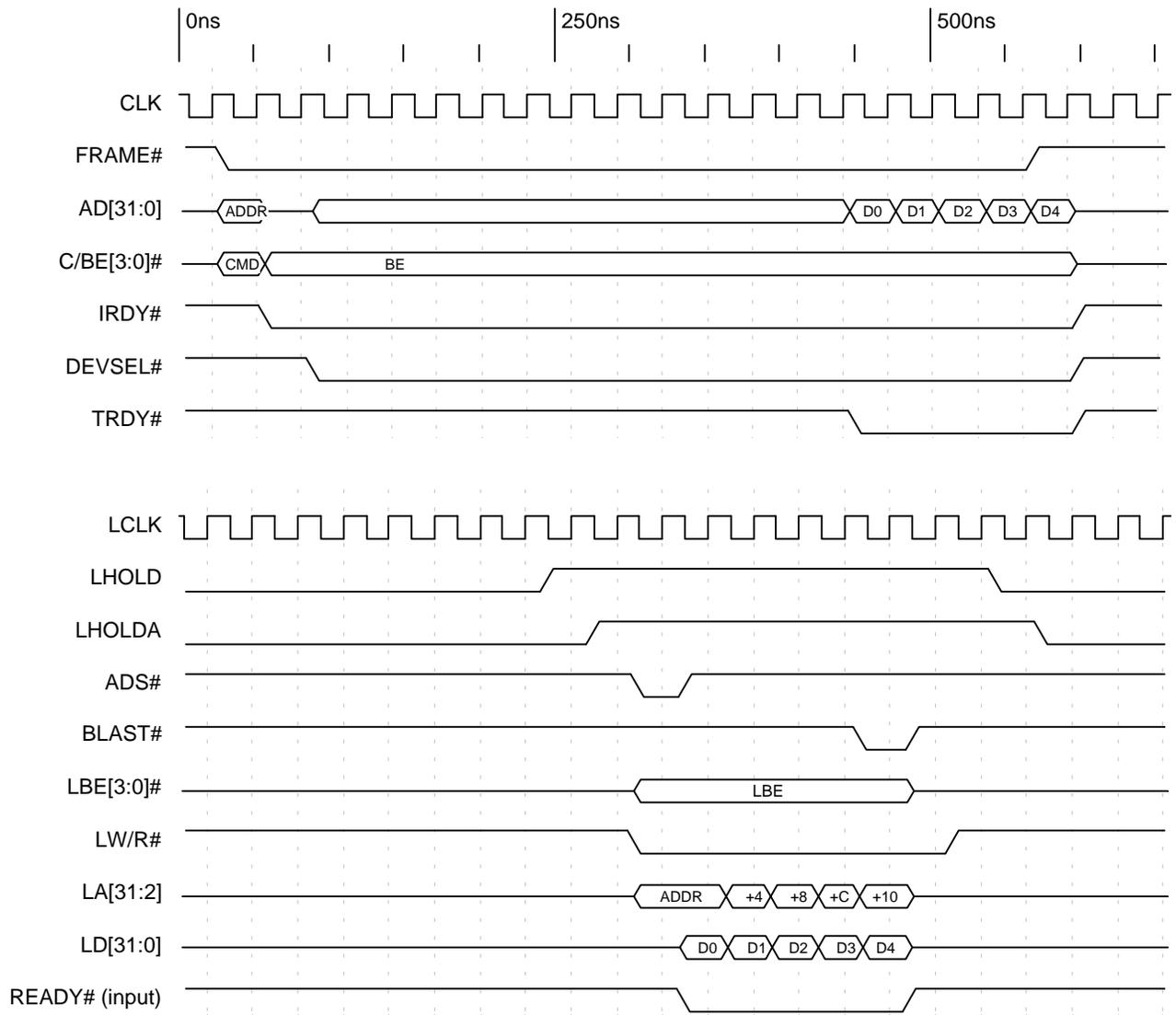




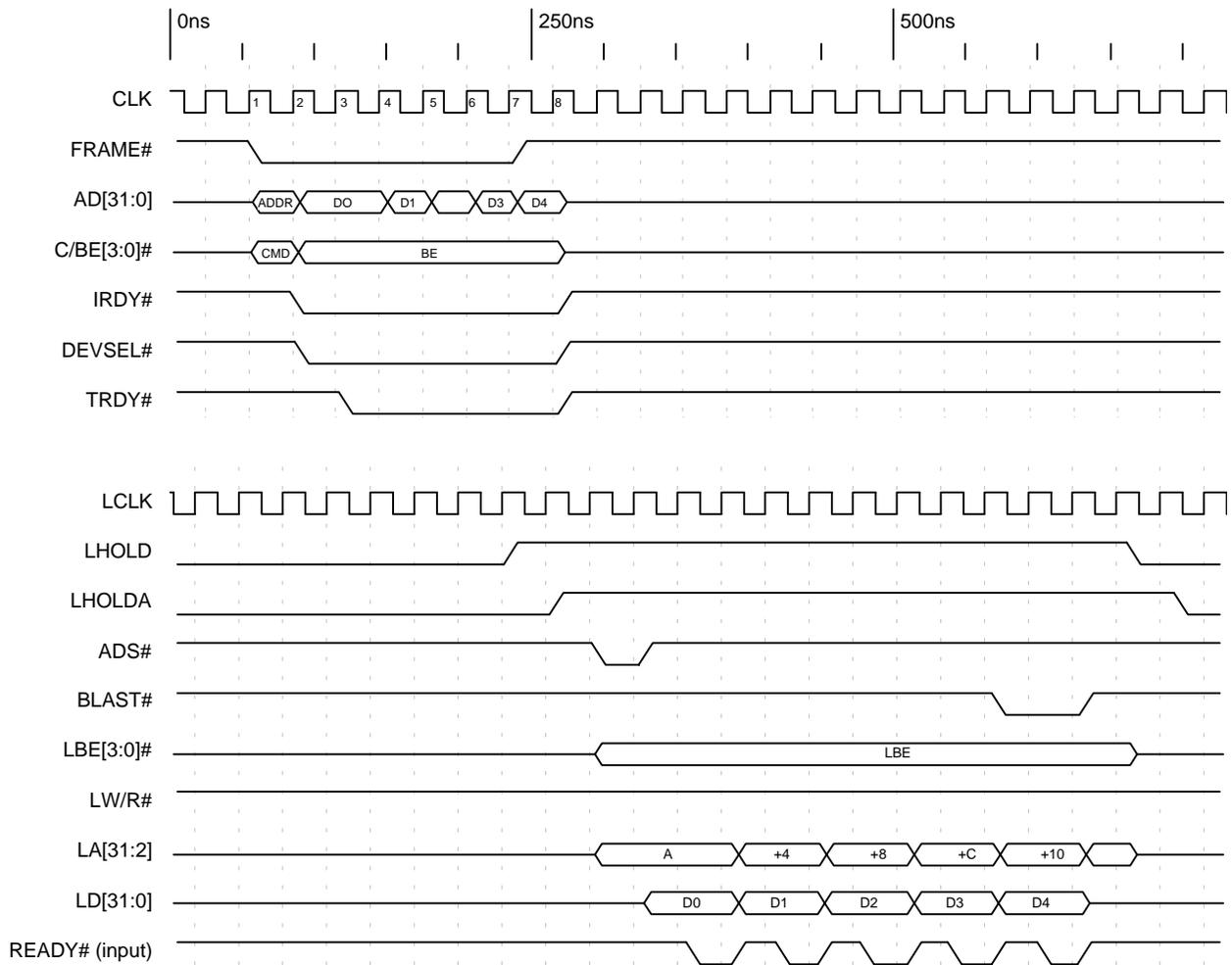
Timing Diagram 5-34. PCI Target Burst Write with Bterm Disabled (32-Bit Local Bus)



Timing Diagram 5-35. PCI Target Burst Read with Prefetch Counter Set to 8 (32-Bit Local Bus)

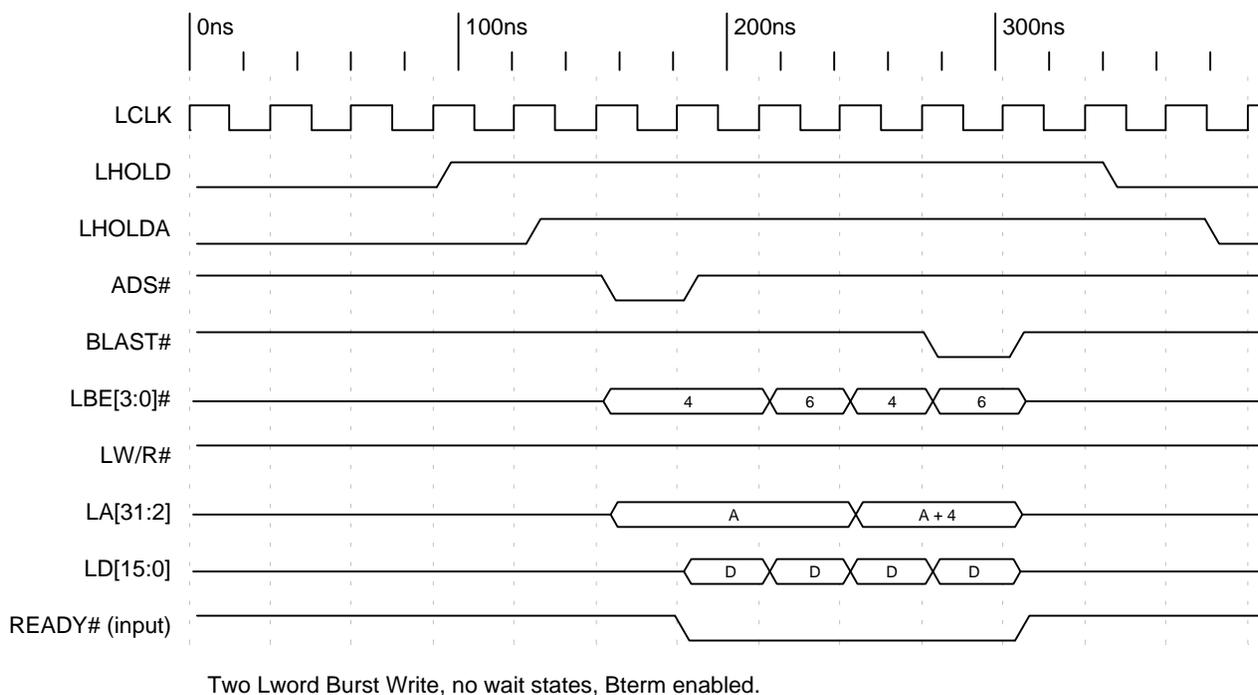


Timing Diagram 5-36. PCI Target Burst Read with Prefetch Counter Set to 5 (32-Bit Local Bus)

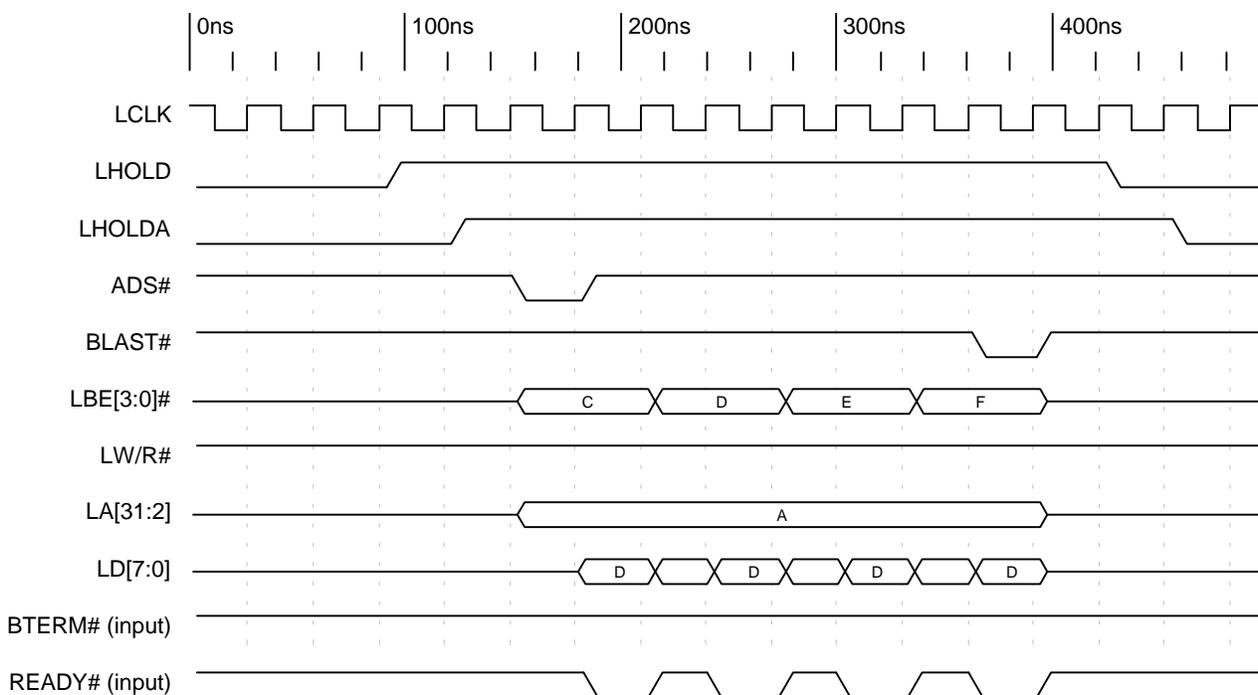


Five Lwords, one external wait state, Bterm enabled, Burst enabled.

**Timing Diagram 5-37. PCI Target Burst Write (32-Bit Local Bus)**

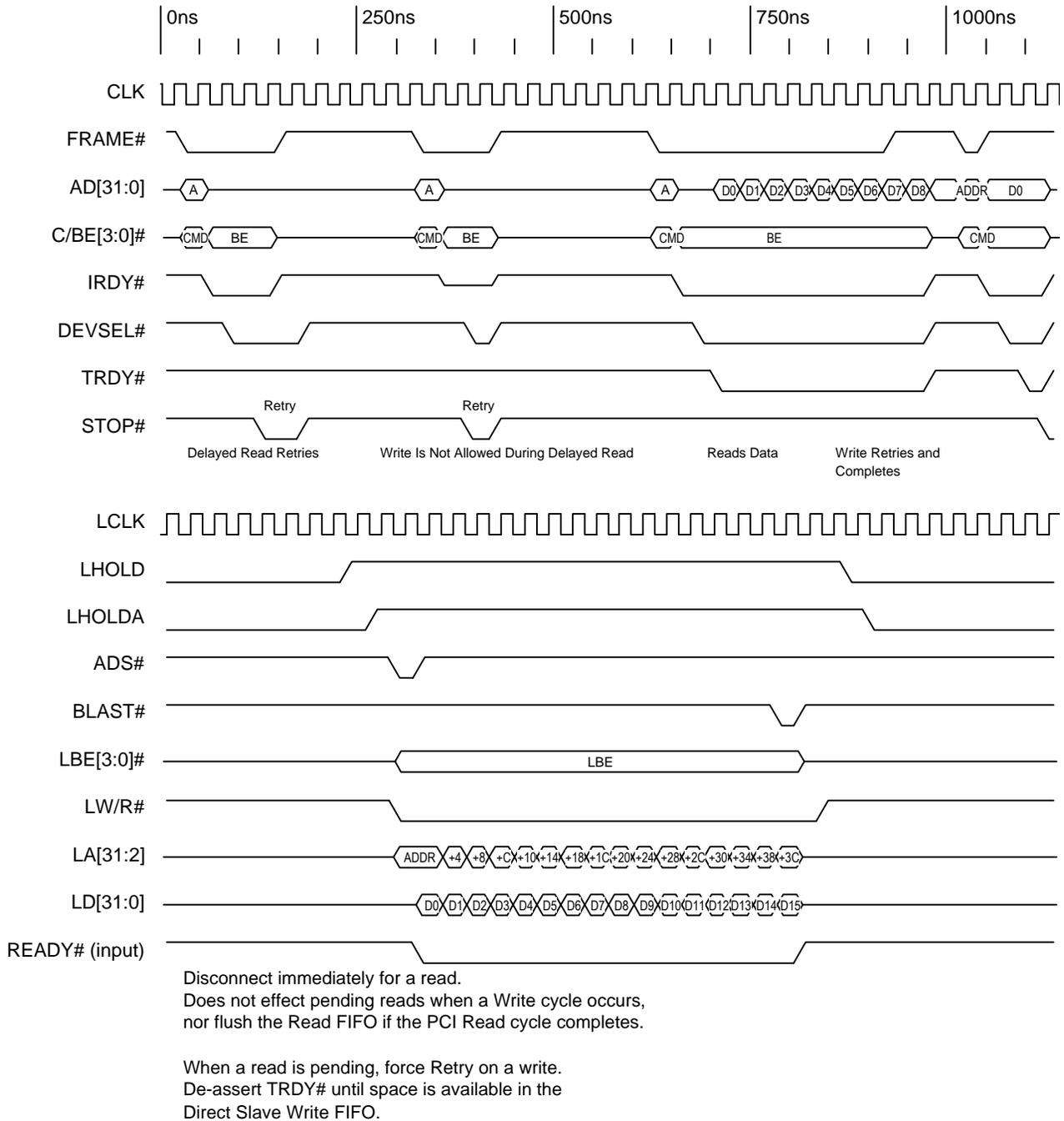


Timing Diagram 5-38. PCI Target Burst Write (16-Bit Local Bus)

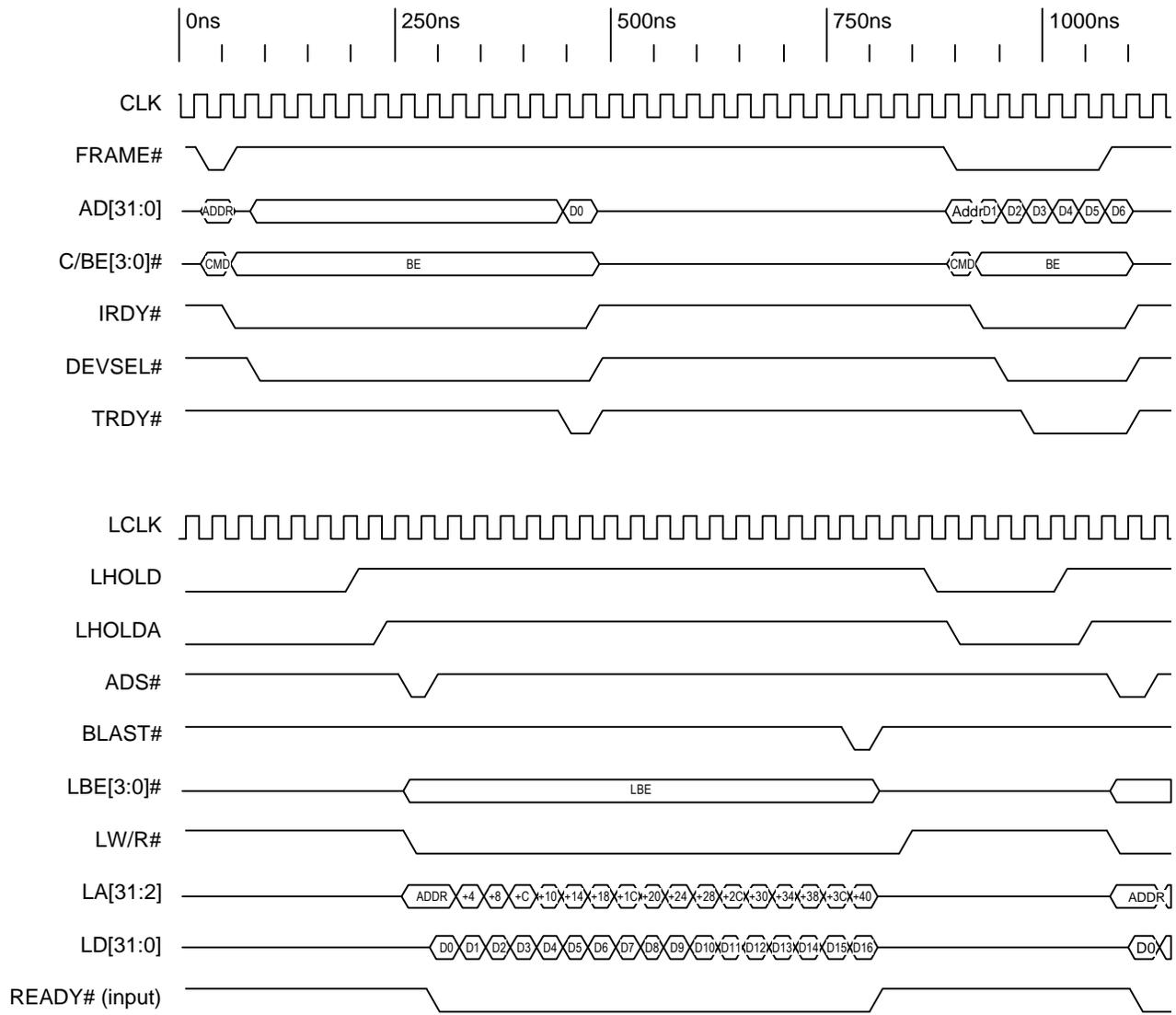


Timing Diagram 5-39. PCI Target Burst Write with External Wait States (8-Bit Local Bus)

Section 5—C, J Func Desc

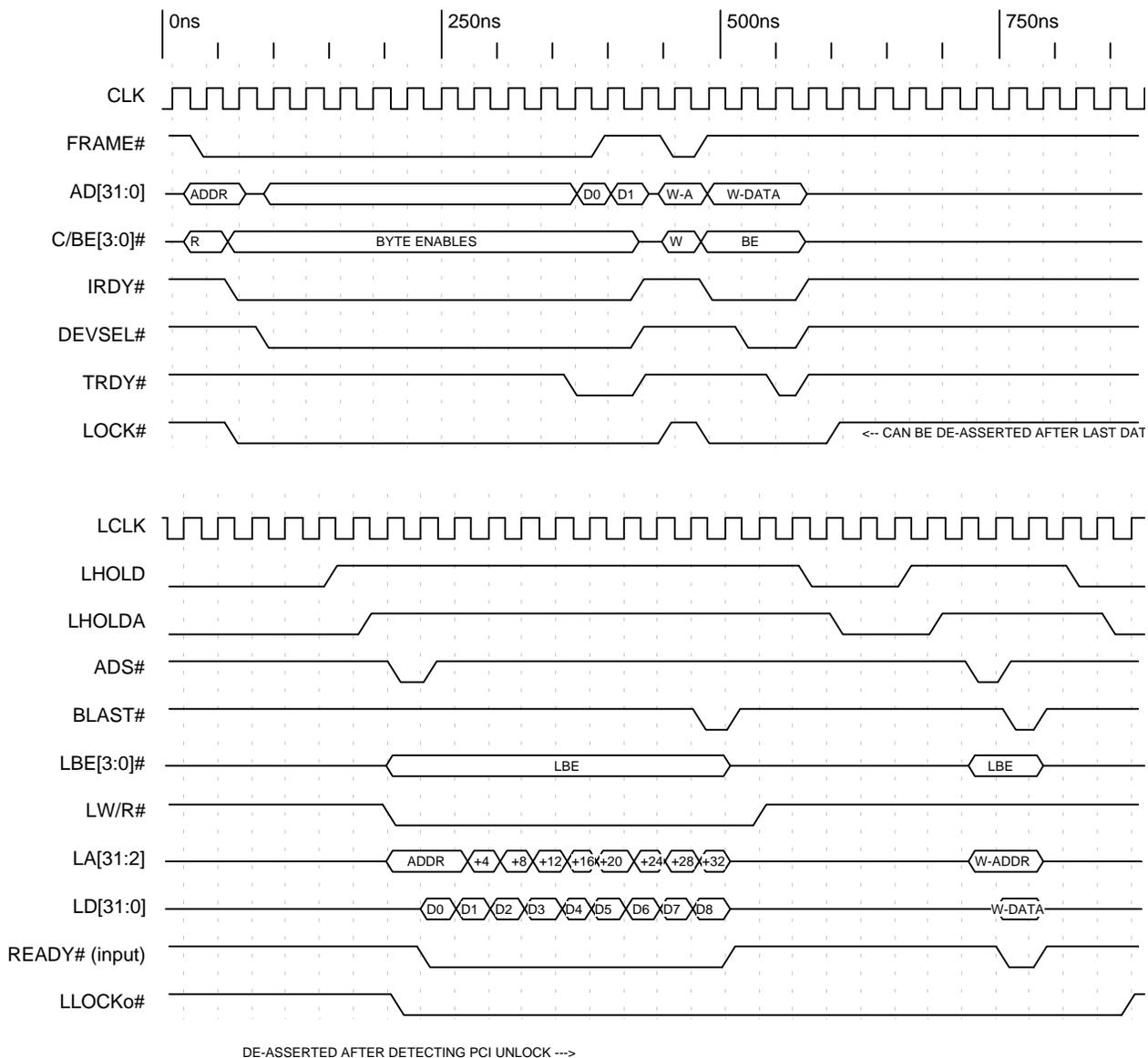


Timing Diagram 5-40. Delayed Read Transaction PCI Specification v2.1

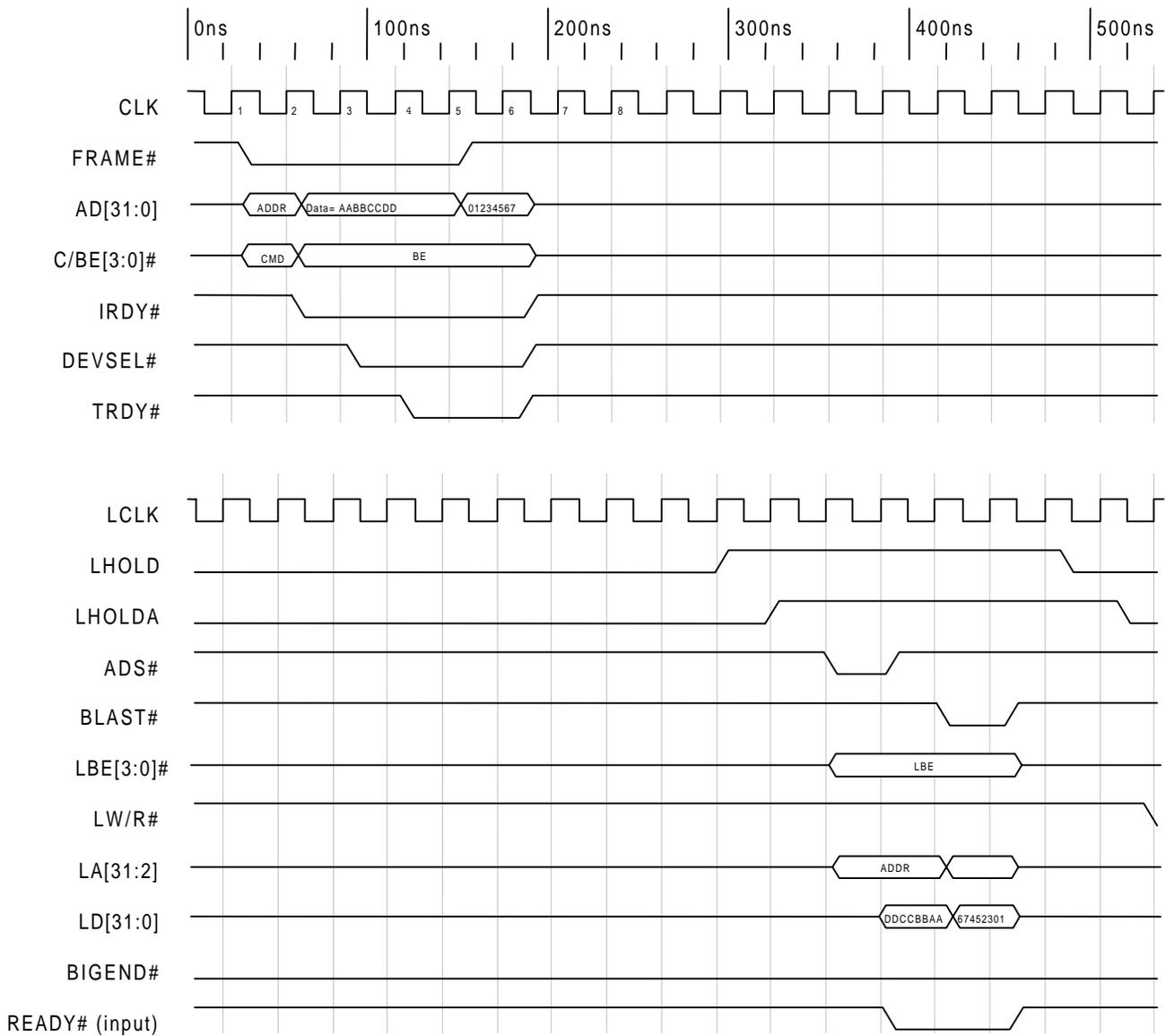


Timing Diagram 5-41. PCI Target Read No Flush Mode (Read Ahead Mode), Prefetch Enabled, Prefetch Count Disabled



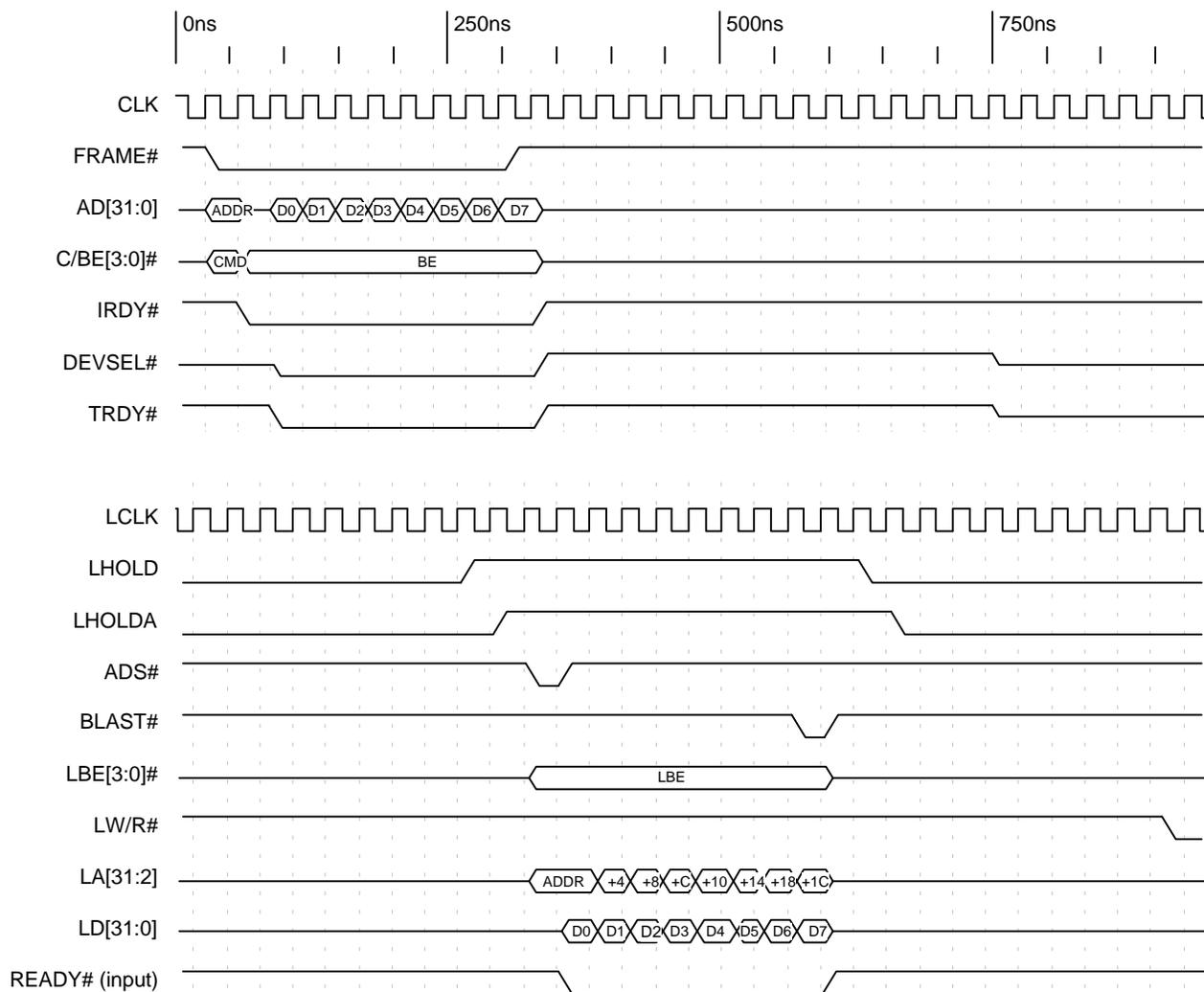


Timing Diagram 5-43. Locked PCI Target Read Followed by Write and Release (LLOCKo#)

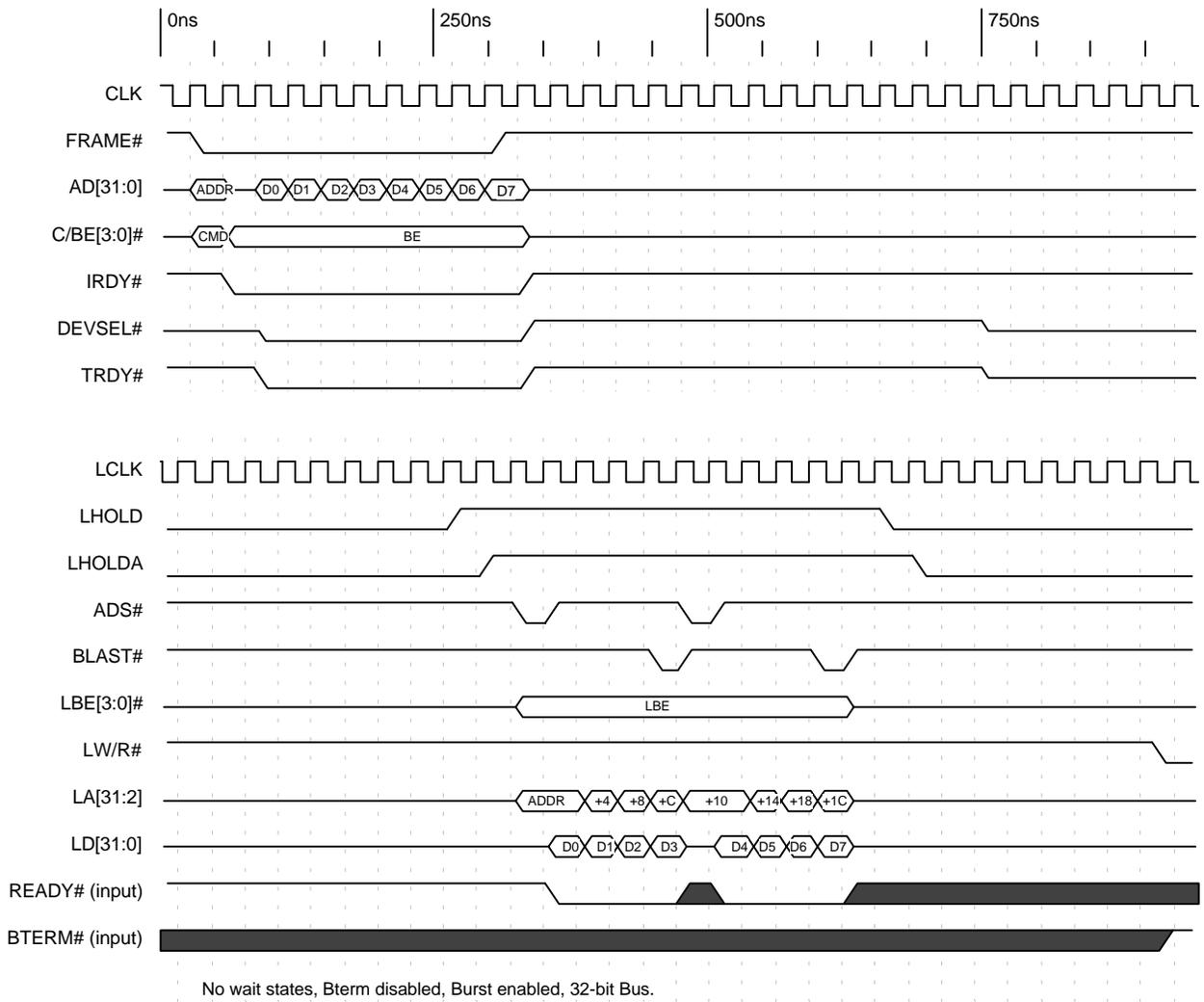


Timing Diagram 5-44. PCI Target in BIGEND Local Bus with BIGEND# Input

### 5.6.3 C Mode DMA

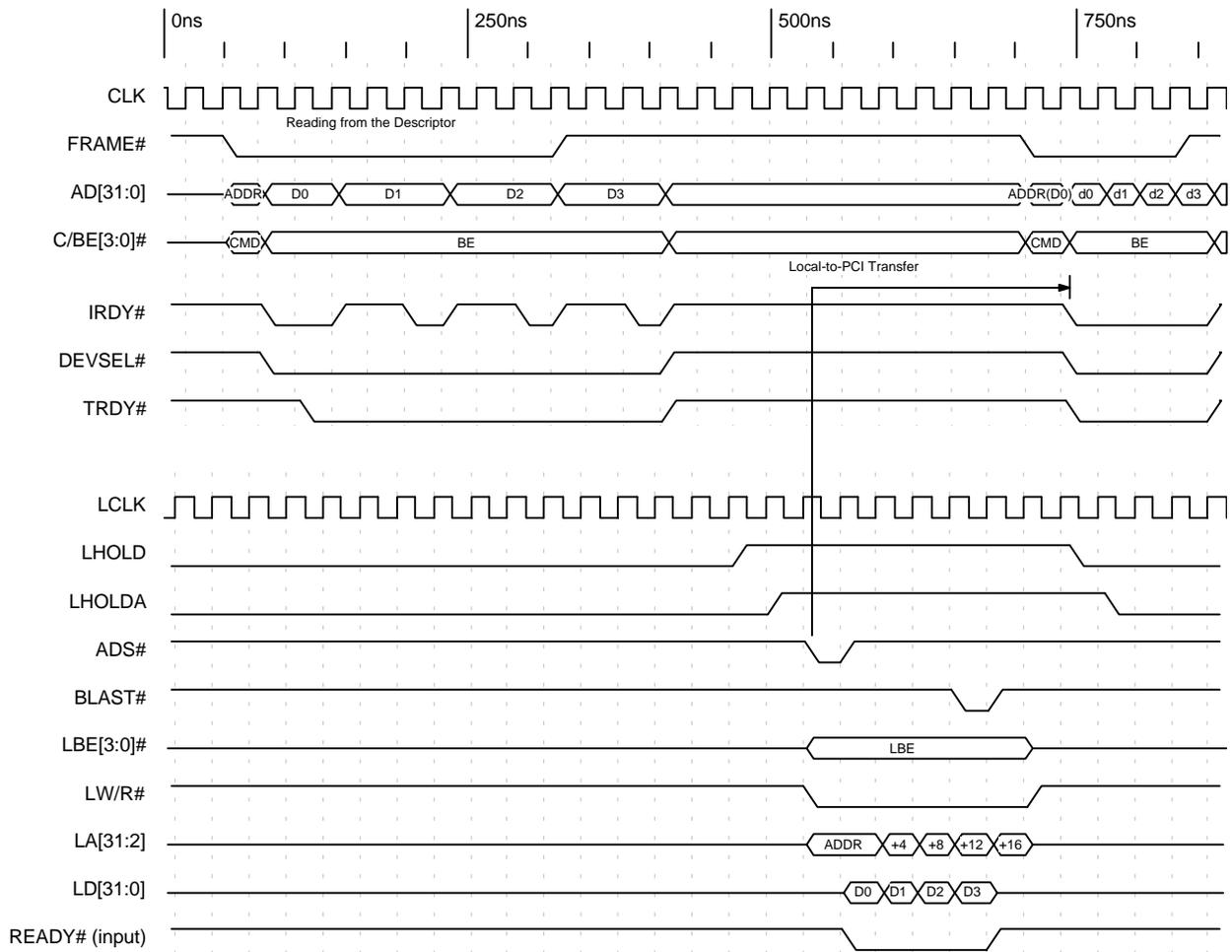


Timing Diagram 5-45. DMA Aligned PCI Address to Aligned Local Address, Bterm Enabled, Burst Enabled

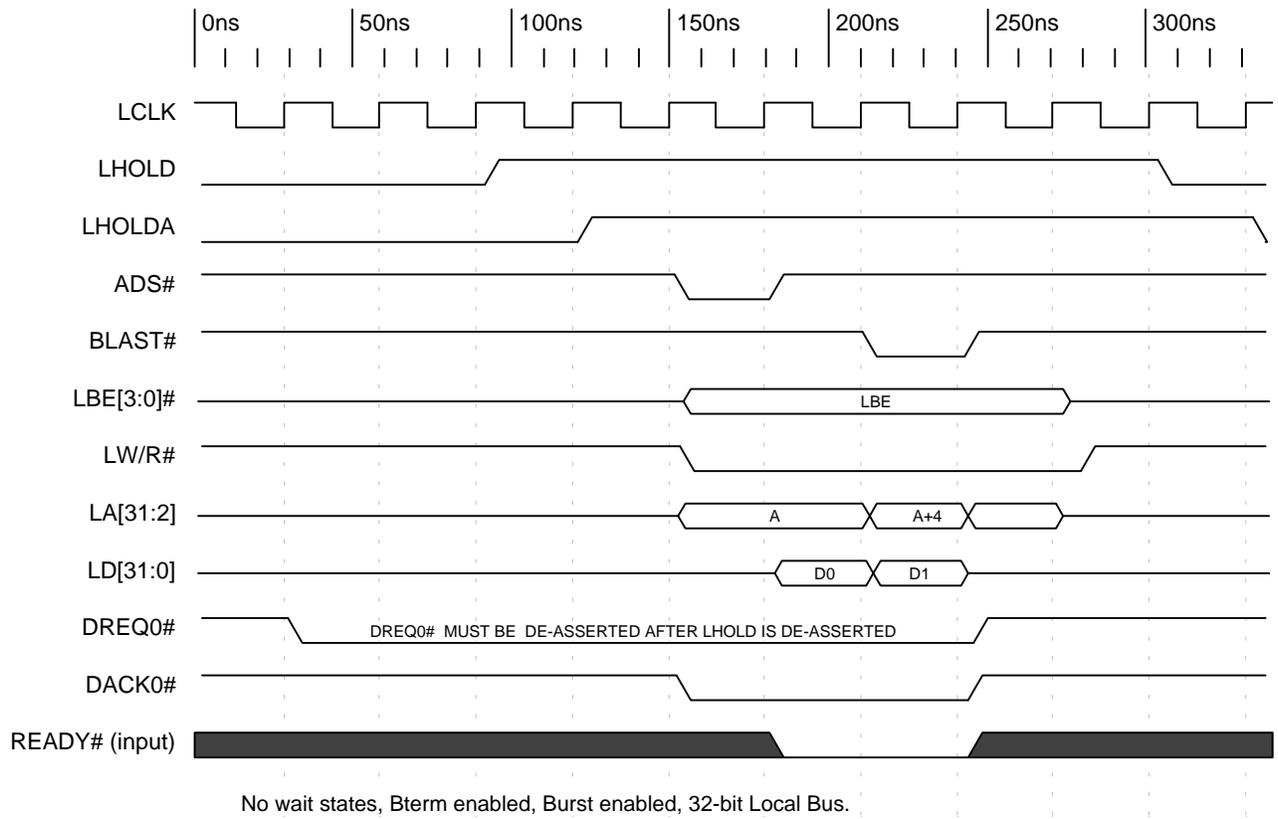


Timing Diagram 5-46. DMA Aligned PCI Address to Aligned Local Address, Bterm Disabled, Burst Enabled

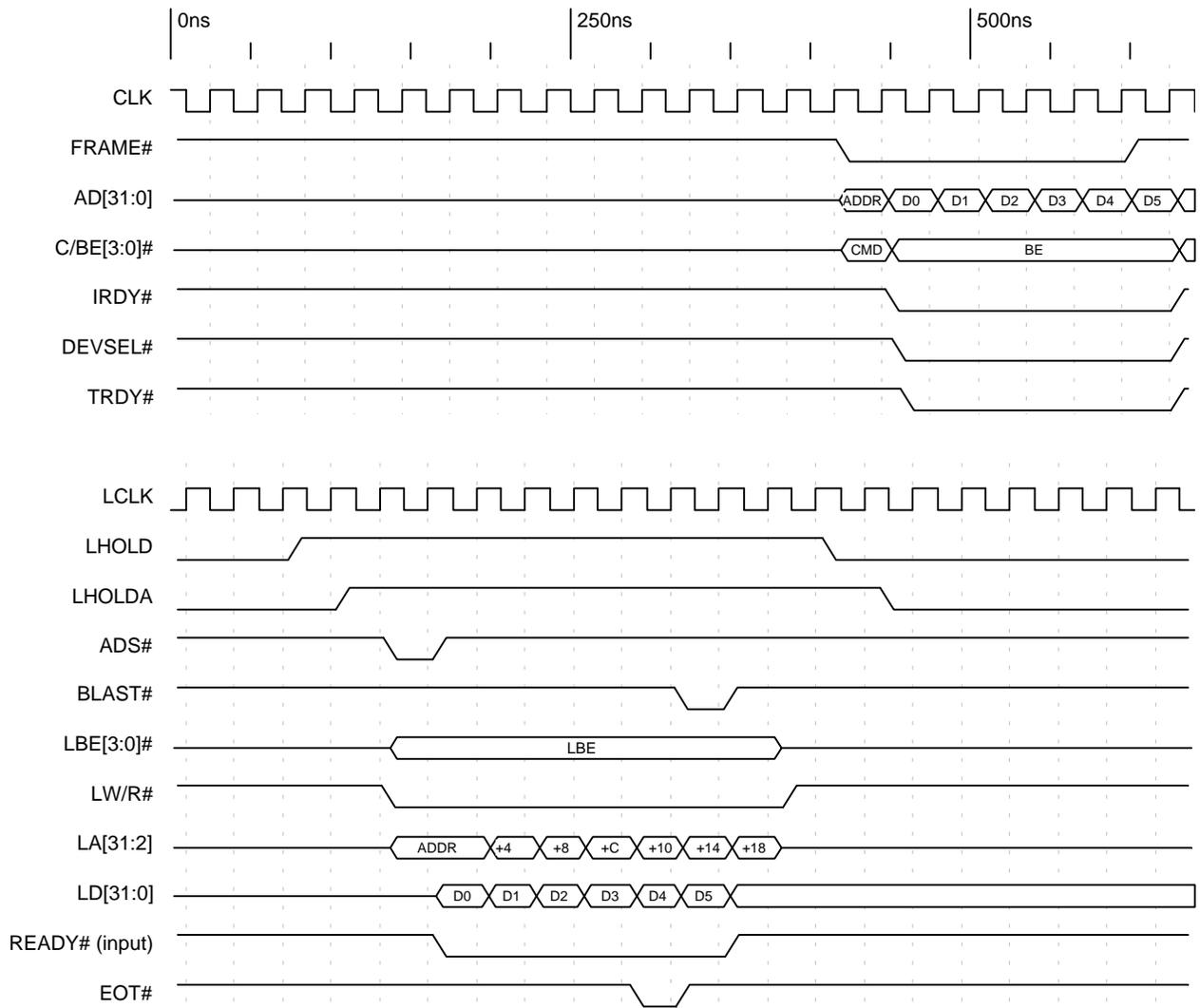




Timing Diagram 5-48. Scatter/Gather DMA from Local-to-PCI with Descriptor on PCI Bus

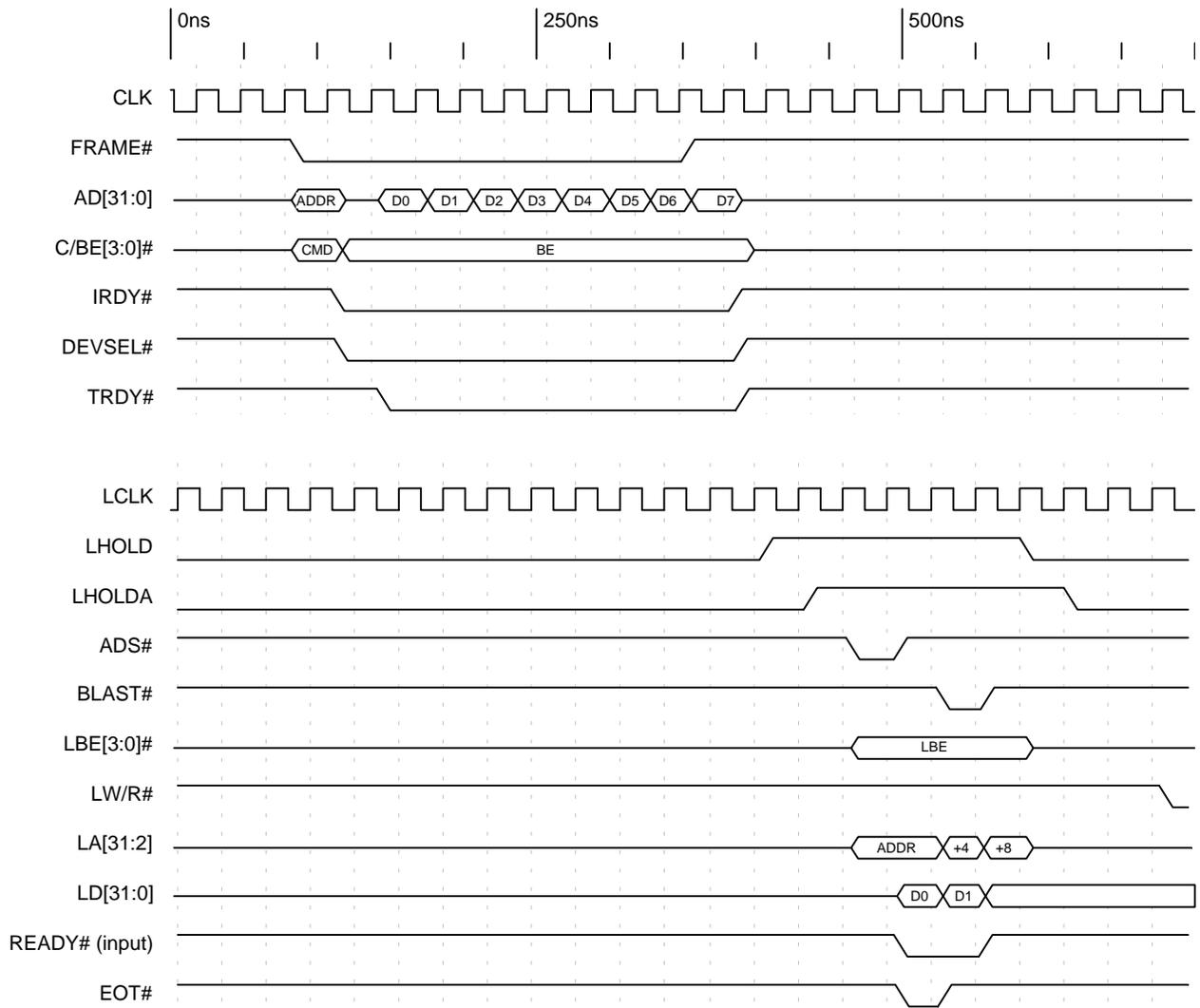


Timing Diagram 5-49. Demand DMA, Terminate with BLAST# (Local-to-PCI)



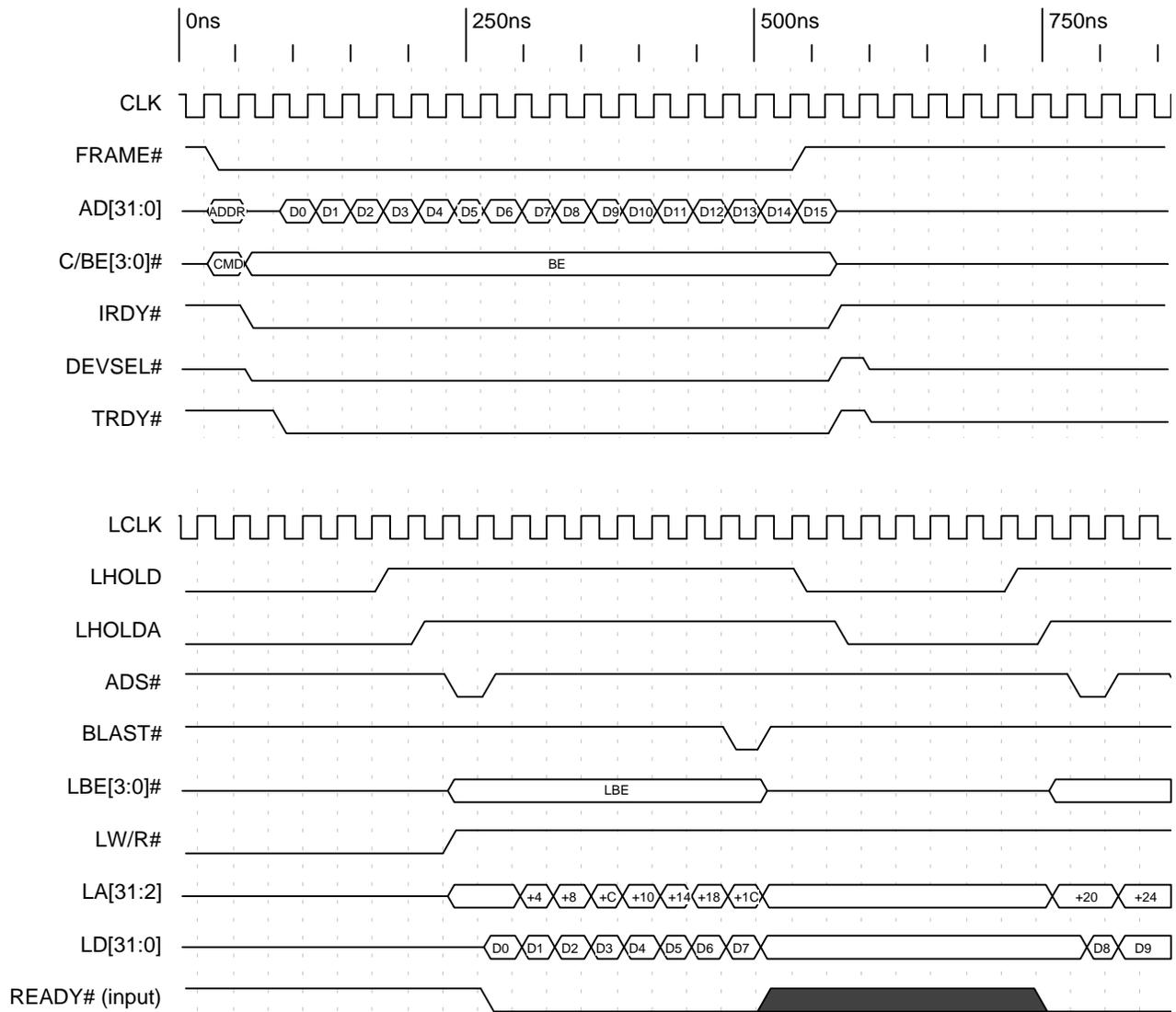
The Fast/Slow Terminate Mode Select bit is not set. If this bit is set, there is no BLAST# and D5 is not transferred (see DMAMODE[15] and/or DMAMODE1[15]).

Timing Diagram 5-50. DMA Local-to-PCI, Terminate with EOT#



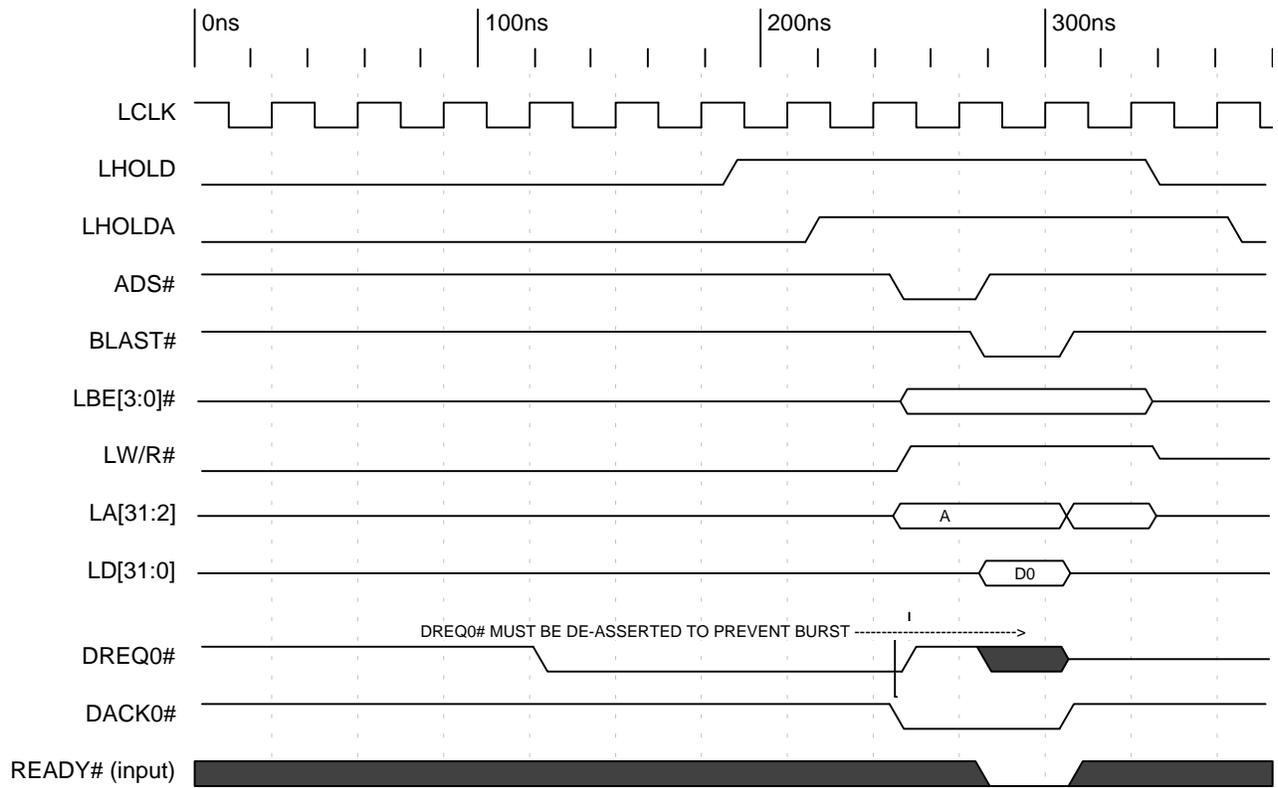
The Fast/Slow Terminate Mode Select bit is not set. If this bit is set, there is no BLAST# and D1 is not transferred (see DMAMODE[15] and/or DMAMODE1[15]).

Timing Diagram 5-51. DMA PCI-to-Local, Terminate with EOT#

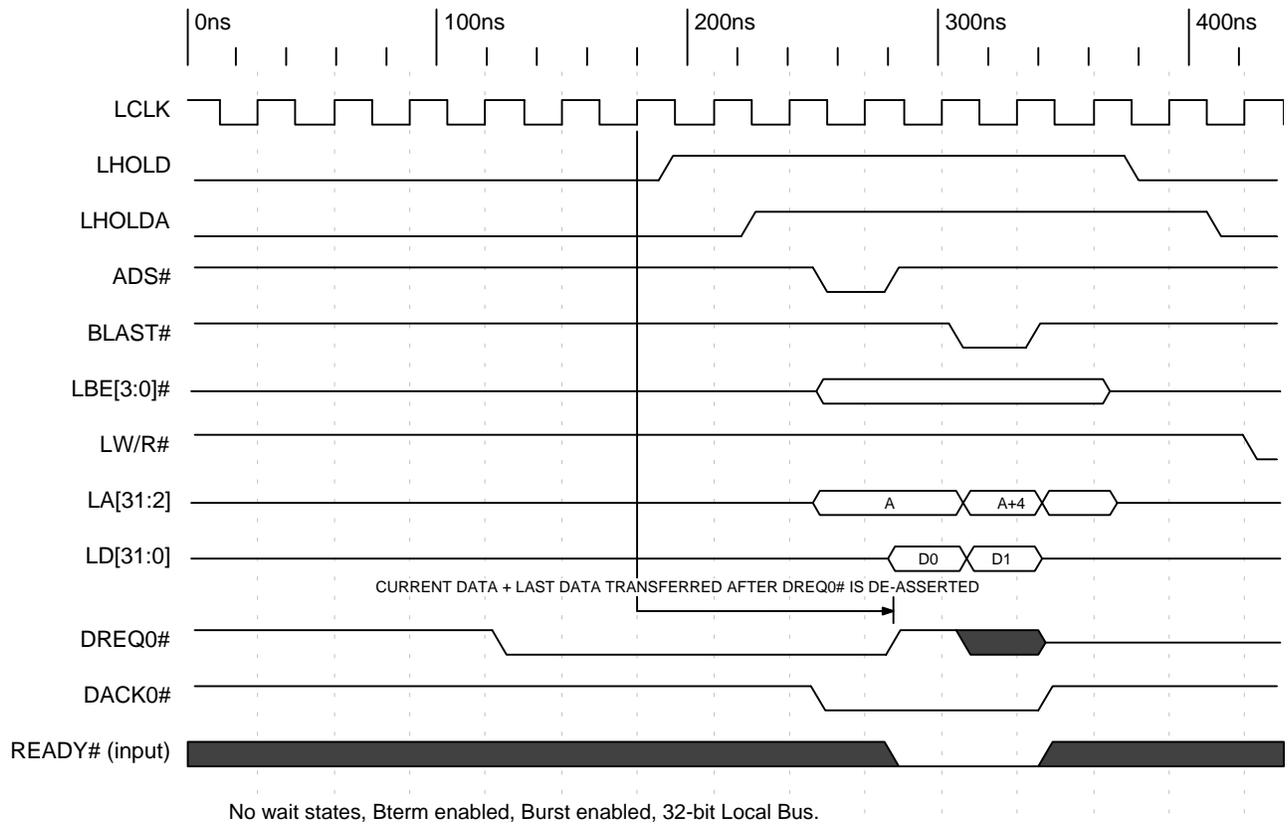


Local Bus Latency Timer = 7 CLK, Local Bus Pause Timer = 4 CLK.  
 The PCI 9054 has internally added another clock to the Local Bus Pause Timer.  
 The Local Bus Latency Timer starts after ADS# is active.  
 The Local Bus Pause Timer starts after LHOLDA is de-asserted.

Timing Diagram 5-52. DMA PCI-to-Local with Local Bus Pause and Latency Timers



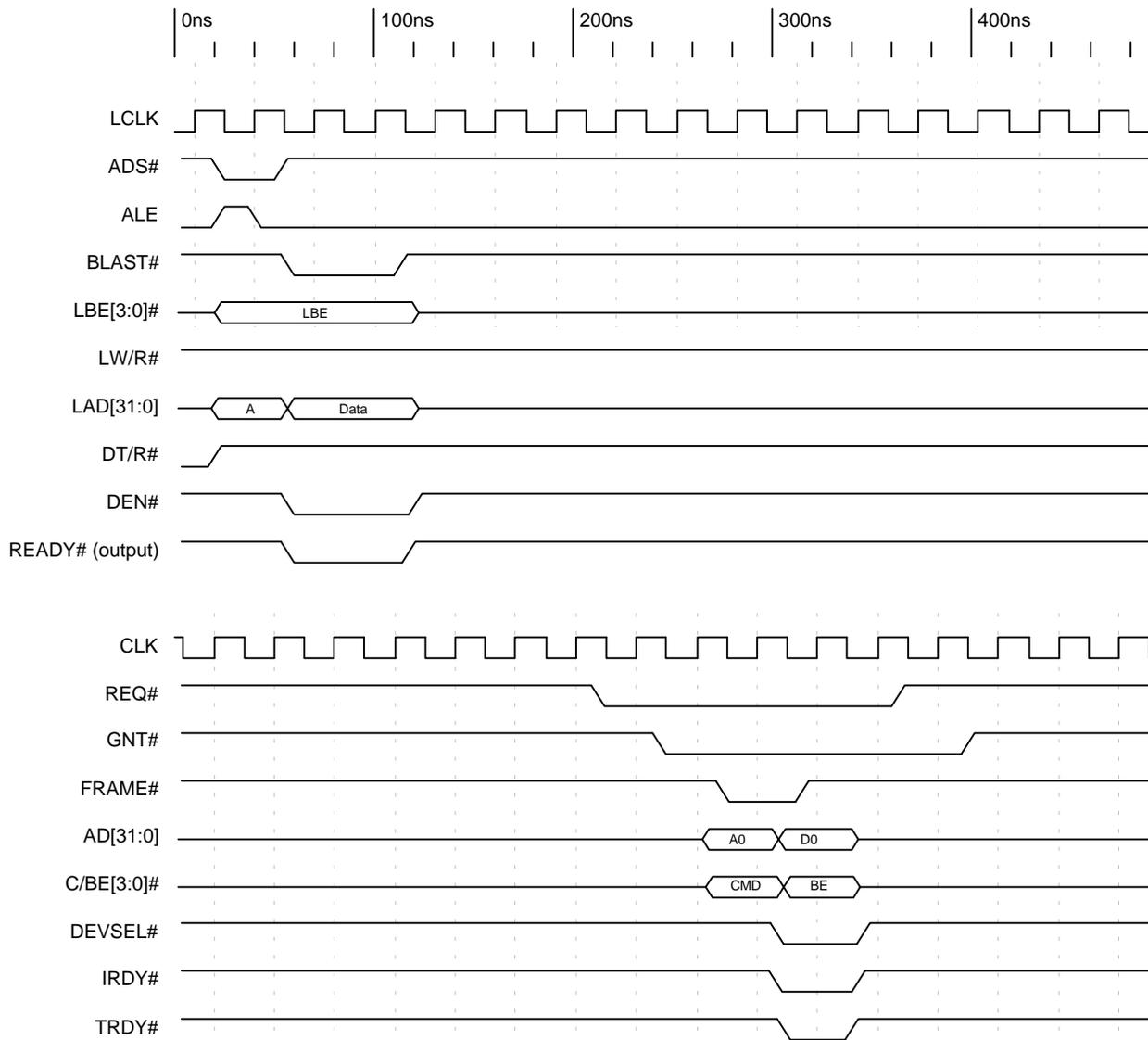
Timing Diagram 5-53. Single-Cycle Demand DMA Mode (PCI-to-Local)



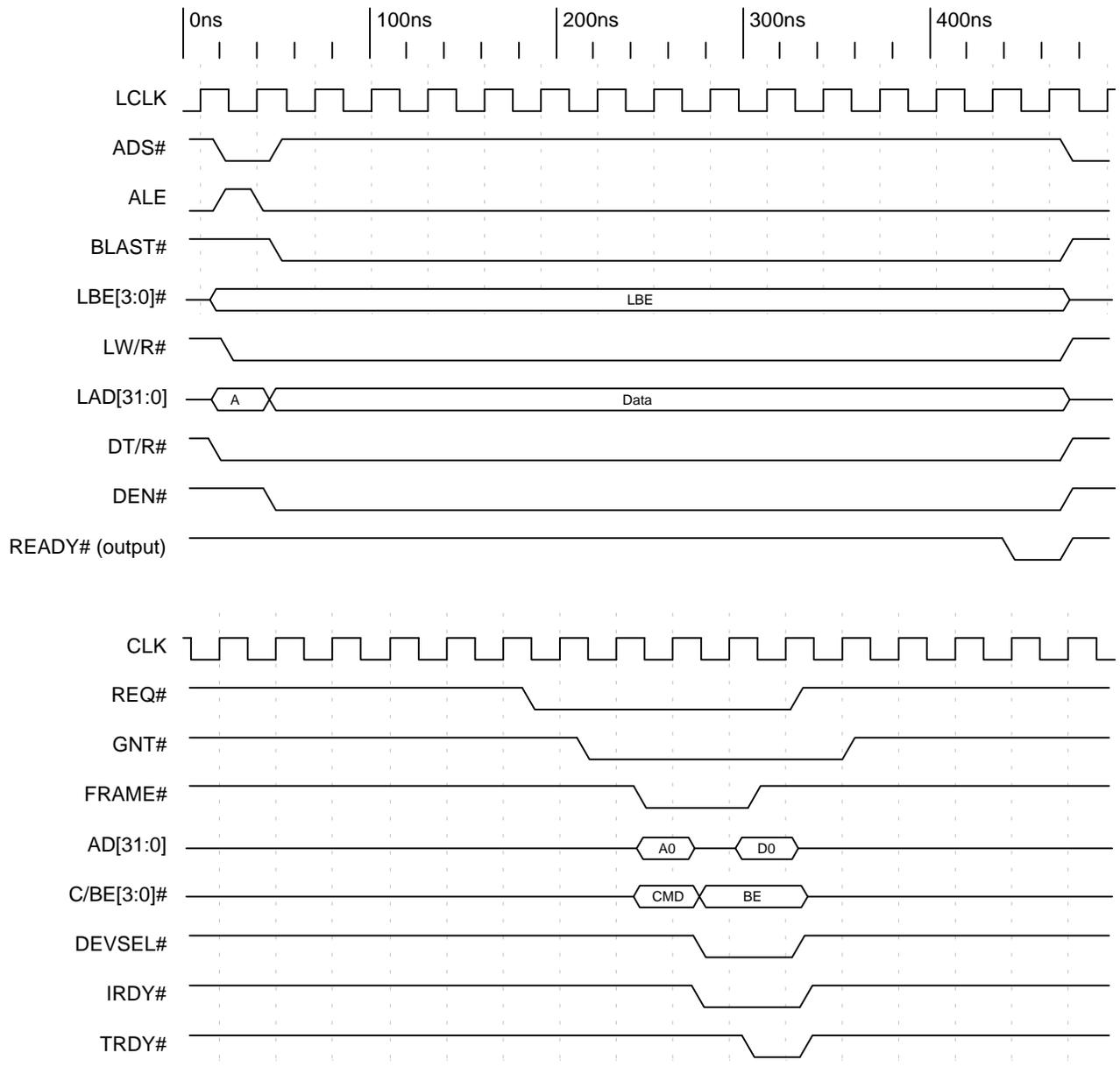
Timing Diagram 5-54. Multiple-Cycle Demand DMA Mode (PCI-to-Local)

## 5.7 J MODE TIMING DIAGRAMS

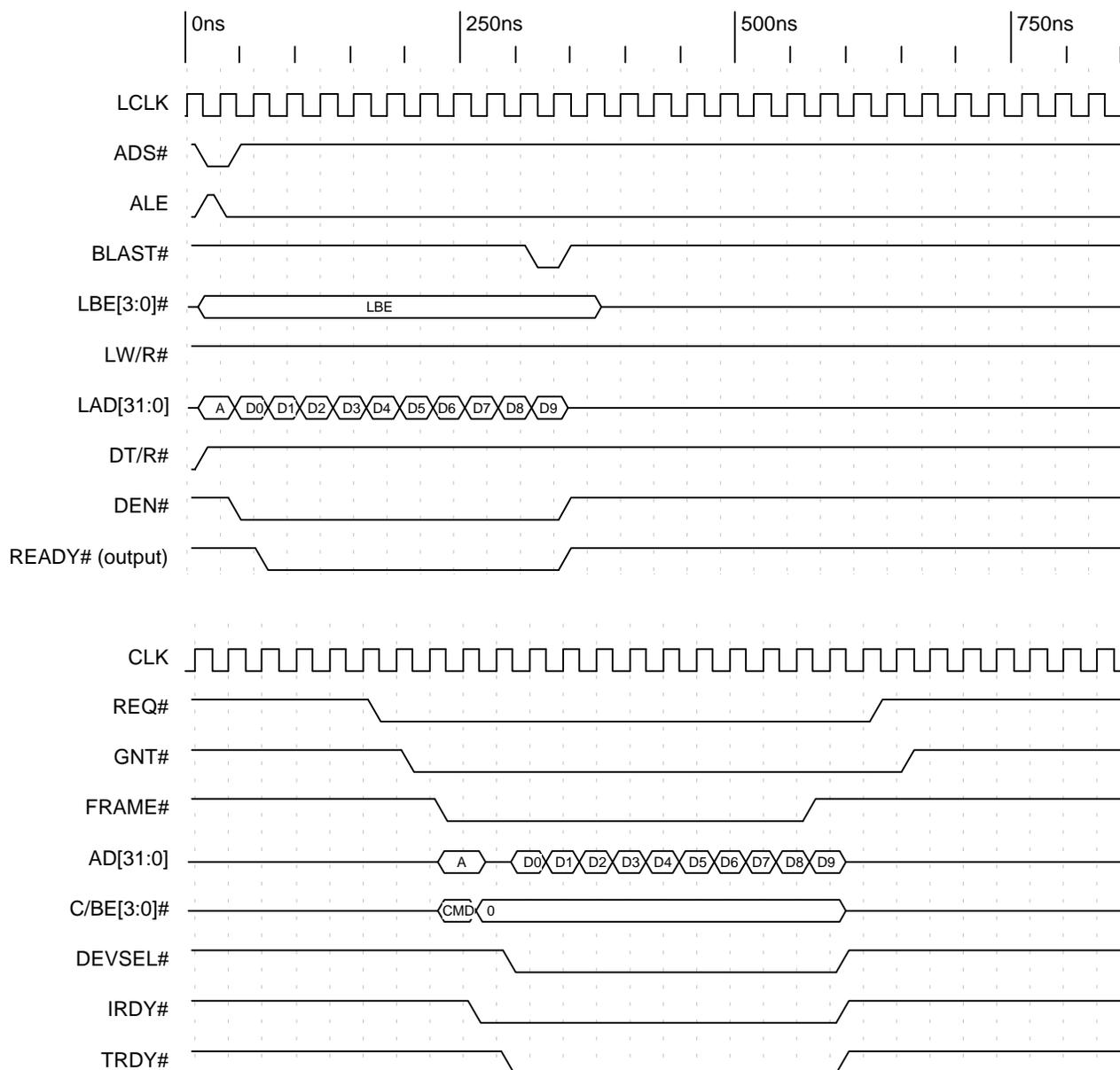
### 5.7.1 J Mode PCI Initiator



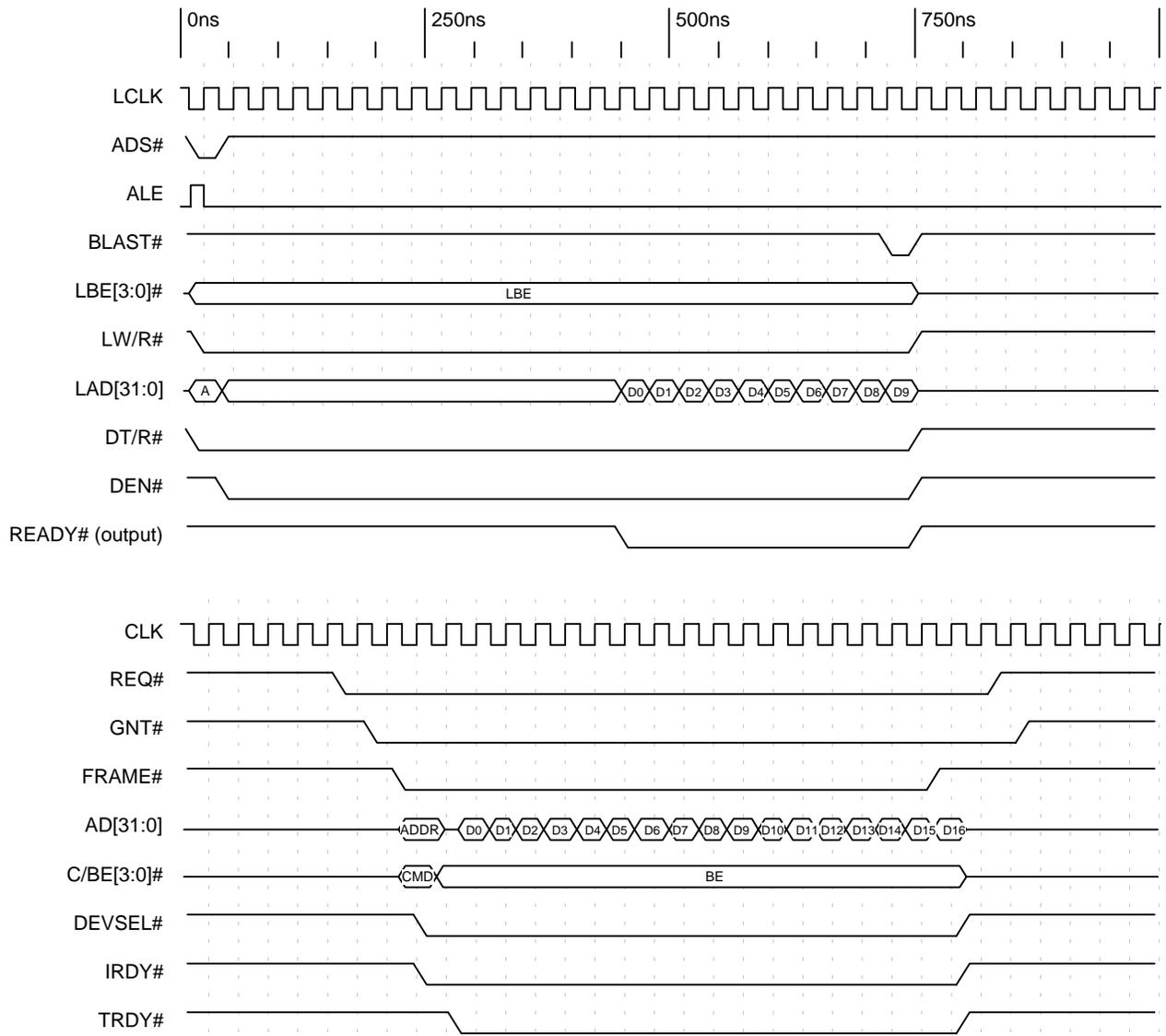
Timing Diagram 5-55. PCI Initiator Single Write



Timing Diagram 5-56. PCI Initiator Single Read

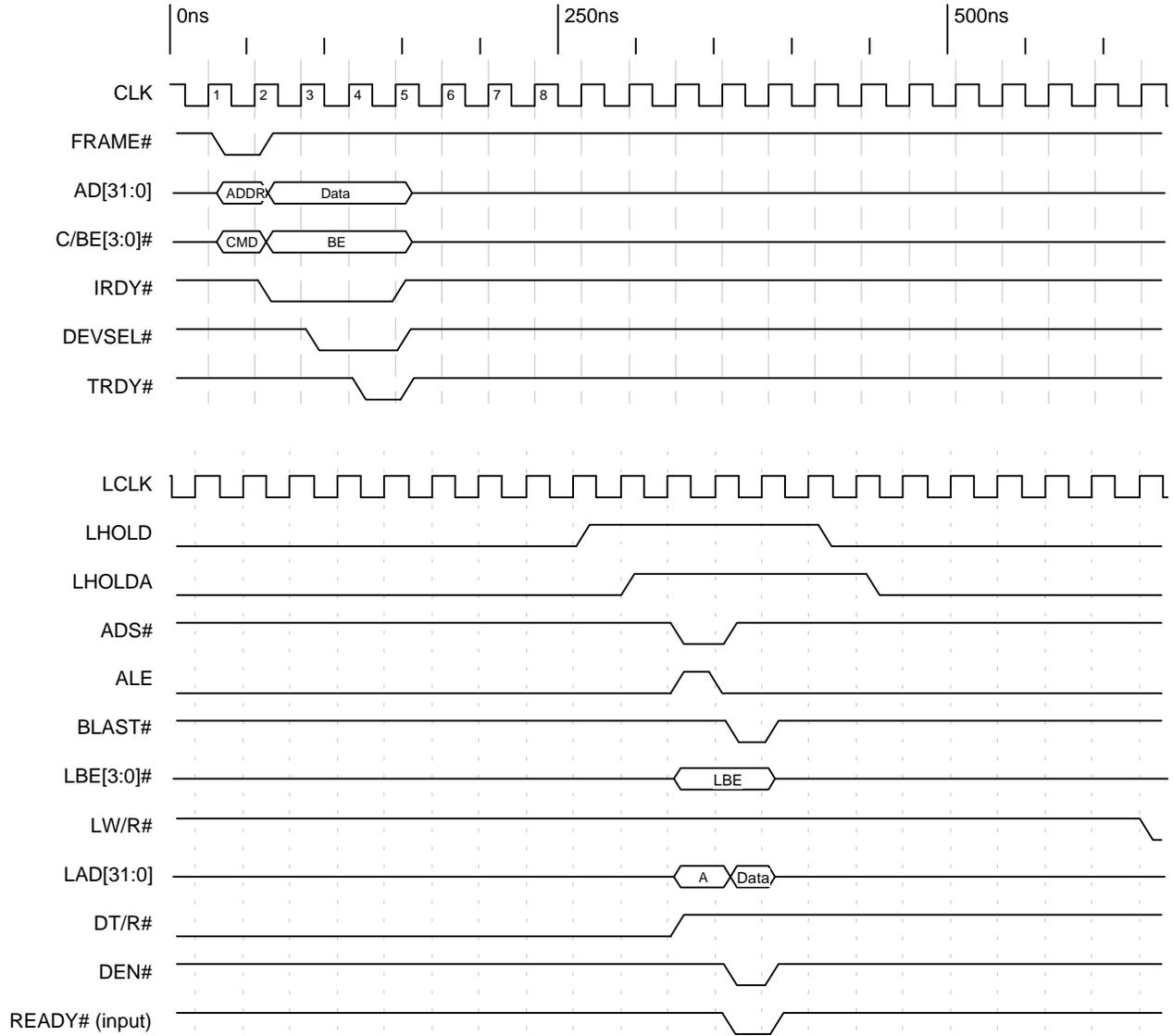


Timing Diagram 5-57. PCI Initiator Burst Write of 10 Lwords, Zero Wait States

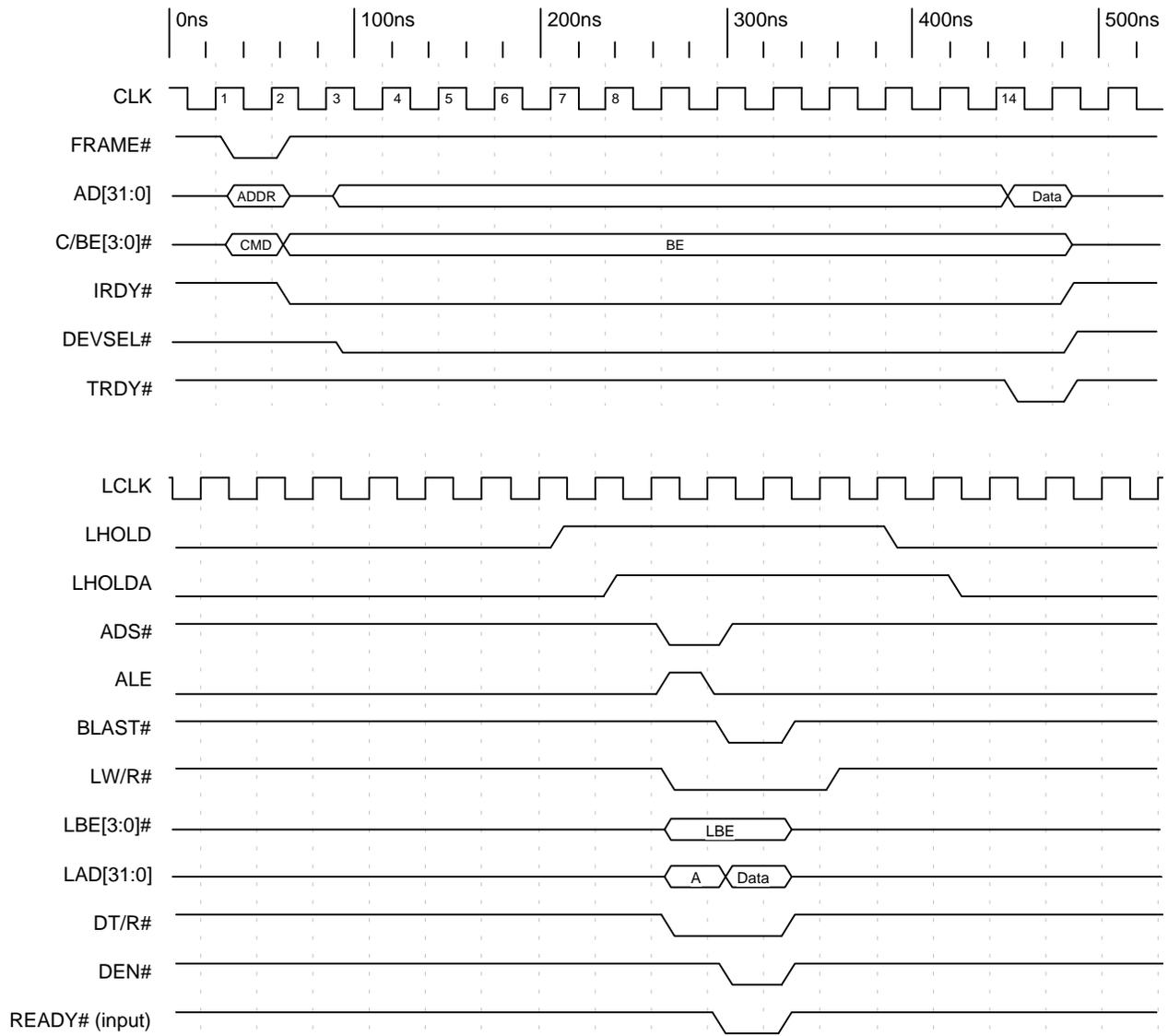


Timing Diagram 5-58. PCI Initiator Burst Read of 10 Lwords, Zero Wait States

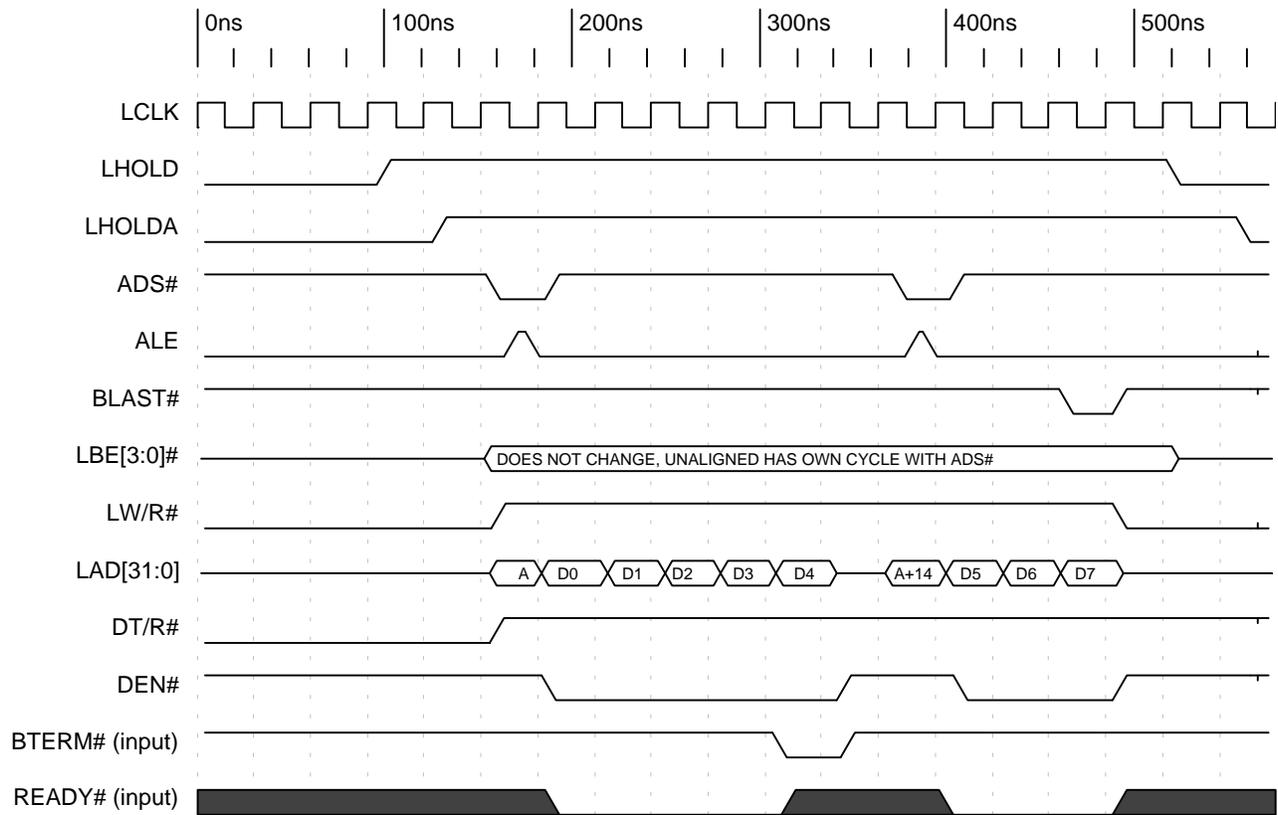
5.7.2 J Mode PCI Target



Timing Diagram 5-59. PCI Target Single Write (32-Bit Local Bus)



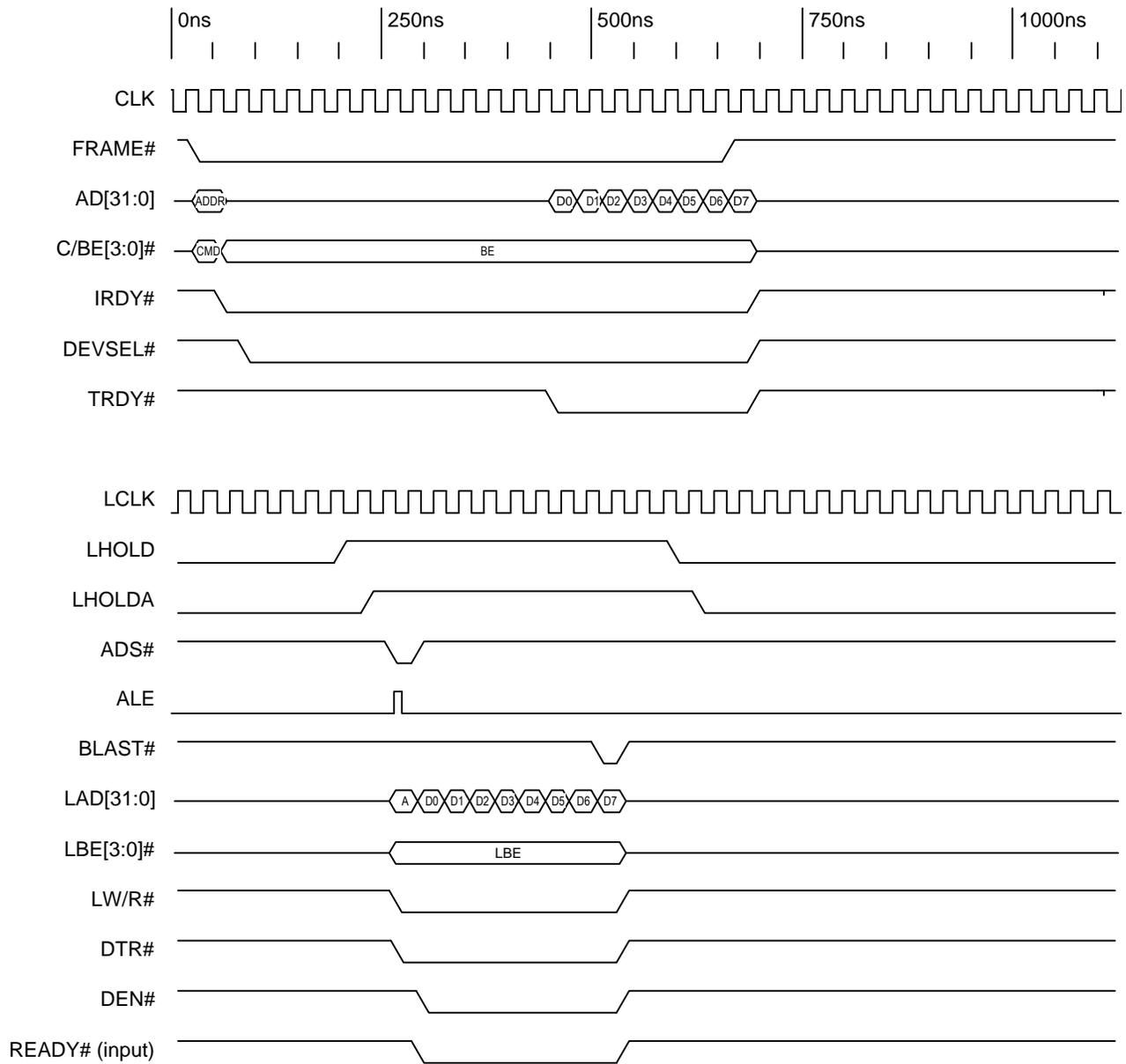
Timing Diagram 5-60. PCI Target Single Read (32-Bit Local Bus)



Eight Lword Burst, no wait states, BTERM# (input) enabled, Burst enabled, 32-bit Local Bus.

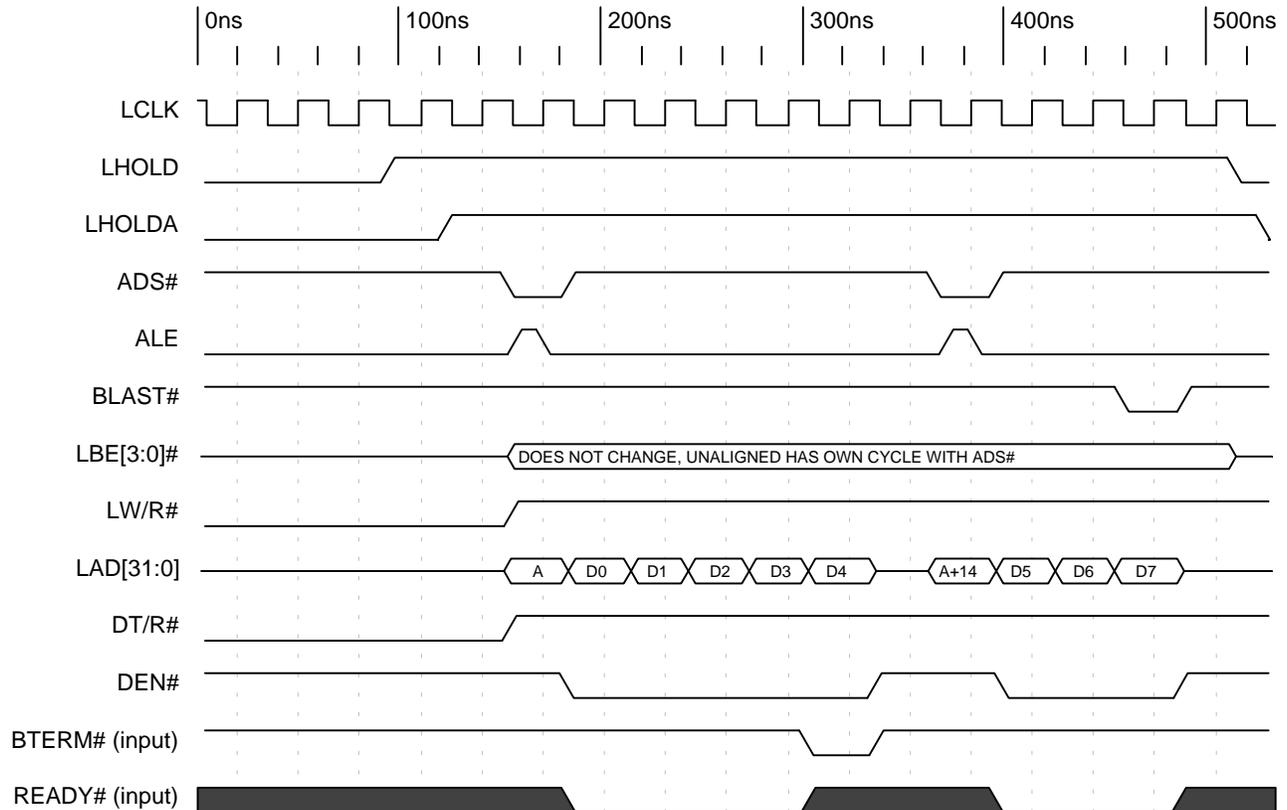
Note: If BTERM# (input) is disabled, a new ADS# cycle starts every quad-Lword boundary. BTERM# (input) replaces READY# (input) when asserted.

Timing Diagram 5-61. PCI Target Burst Write with Bterm Enabled (32-Bit Local Bus)



Timing Diagram 5-62. PCI Target Burst Read with Prefetch Enabled (32-Bit Local Bus), Prefetch Counter Set to 8

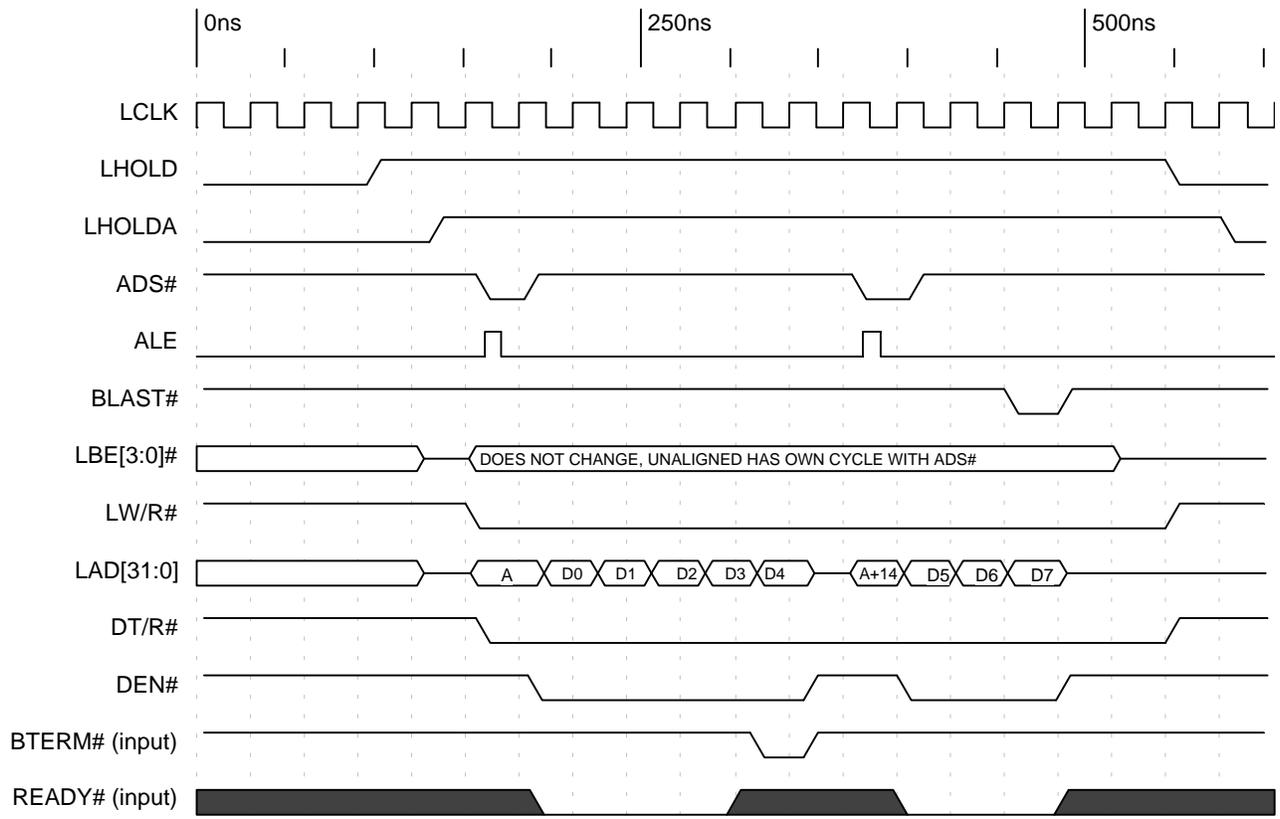
5.7.3 J Mode DMA



Eight Lword burst, no wait states, BTERM# (input) enabled,  
Burst enabled, 32-bit Local Bus

Note: If BTERM# (input) is disabled, a new ADS#  
cycle starts every quad-Lword boundary.

Timing Diagram 5-63. DMA Aligned PCI Address to Aligned Local Address, Bterm Enabled, Burst Enabled



Eight Lword Burst, no wait states, BTERM# (input) enabled, Burst enabled, 32-bit Local Bus.

Note: If BTERM# (input) is disabled, a new ADS# cycle starts every quad-Lword boundary.

Timing Diagram 5-64. DMA Aligned Local Address to Aligned PCI Address, Bterm Enabled, Burst Enabled

# 6 PCI LOCAL INTERRUPTS AND USER I/O

## 6.1 INTERRUPTS

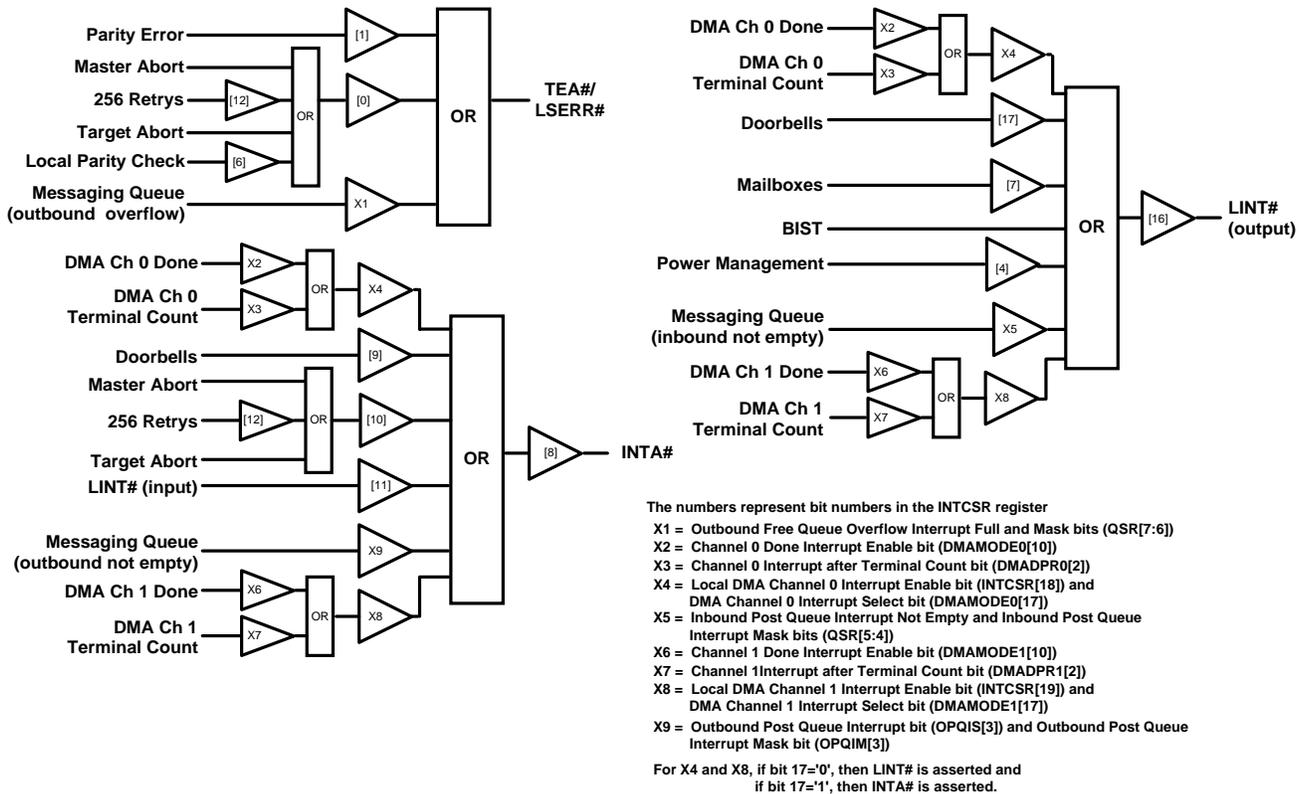


Figure 6-1. Interrupt and Error Sources

### 6.1.1 PCI Interrupts (INTA#)

A PCI 9054 PCI Interrupt (INTA#) can be asserted by one of the following:

- Local-to-PCI Doorbell register
- Local Interrupt input
- Master/Target Abort Status condition
- DMA Ch 0/Ch 1 Done
- DMA Ch 0/Ch 1 Terminal Count is reached
- Messaging Outbound Post Queue not empty
- 256 consecutive PCI Retrys

INTA#, or individual sources of an interrupt, can be enabled or disabled with the PCI 9054 Interrupt Control/Status register (INTCSR). This register also provides the interrupt status of each interrupt source.

The PCI 9054 PCI Bus interrupt is a level output. Disabling an interrupt enable bit or clearing the cause(s) of the interrupt can clear an interrupt.

### 6.1.2 Local Interrupt Input (LINT#)

Asserting the Local Bus input LINT# can assert a PCI Bus interrupt. The PCI Host processor can read the PCI 9054 Interrupt Control/Status register (INTCSR) to determine whether an interrupt is pending as a result of LINT# being asserted (INTCSR[15]).

The interrupt remains asserted as long as LINT# input is asserted and the Local Interrupt input is enabled. The PCI Host processor can take adapter-specific action to cause the Local Bus to release LINT#.

If the PCI Interrupt Enable bit is cleared (INTCSR[8]=0), the PCI interrupt (INTA#) is de-asserted; however, the Local interrupts (LINT#) and the status bit remain active.

### 6.1.3 Local Interrupt Output (LINT#)

The PCI 9054 Local Interrupt (LINT#) output can be asserted by one of the following:

- PCI-to-Local Doorbell/Mailbox register access
- PCI BIST interrupt
- DMA Ch 0/Ch 1 Done interrupt
- DMA Ch 0/Ch 1 Terminal Count is reached
- DMA Abort Interrupt or Messaging Outbound Post Queue is not empty

LINT#, or individual sources of an interrupt, can be enabled or disabled with the PCI 9054 Interrupt Control/Status register (INTCSR). This register also provides interrupt status for each interrupt source.

The PCI 9054 Local interrupt is a level output. Interrupts can be cleared by disabling the Interrupt Enable bit of a source or by clearing the cause of an interrupt.

The Local Interrupt Input Enable bit must be disabled (INTCSR[11]=0) when LINT# output is active; otherwise, the PCI interrupt becomes active. This could result in an unwanted PCI interrupt.

### 6.1.4 Master/Target Abort Interrupt

The PCI 9054 sets the Received Master Abort or Target Abort bit (PCISR[13, 11]=1, respectively) when it detects a Master or Target Abort. These status bits cause the PCI INTA# to be asserted if interrupts are enabled.

Interrupt remains set as long as the Receive Master Abort or Target Abort bits remain set and the Master/Target Abort interrupt is enabled. Use PCI Type 0 Configuration or Local accesses to clear the Received Master Abort and Target Abort interrupt bits (PCISR[13, 11]=0, respectively).

The Interrupt Control/Status Register bit(s) (INTCSR[26:24]) are latched at the time of a Master or Target Abort interrupt. These bits provide information such as which device was the Master when an abort occurred. The PCI 9054 updates these bits only when an abort occurs.

### 6.1.5 Mailbox Registers

The PCI 9054 has eight 32-bit Mailbox registers that can be written to and read from both the PCI and Local Buses. These registers can be used to pass command and status information directly between the PCI and Local Bus devices.

A Local interrupt can be asserted, if enabled (INTCSR[3] and INTCSR[16]), when the PCI Host writes to one of the first four Mailbox registers (MBOX0, MBOX1, MBOX2, or MBOX3).

To clear the Mailbox registry, the destination bus should read the values currently in the Mailbox registers.

### 6.1.6 Doorbell Registers

The PCI 9054 has two 32-bit Doorbell Interrupt/Status registers. One is assigned to the PCI Bus interface. The other is assigned to the Local Bus interface.

A Local processor can assert a PCI Bus interrupt by writing any number other than all zeroes to the Local-to-PCI Doorbell register bits (P2LDBELL[31:0]).

A PCI Host can assert a Local Bus interrupt by writing any number other than all zeroes to the PCI-to-Local Doorbell register bits (L2PDBELL[31:0]). The PCI Interrupt and Local Interrupt remain asserted until all bits are cleared to zero.

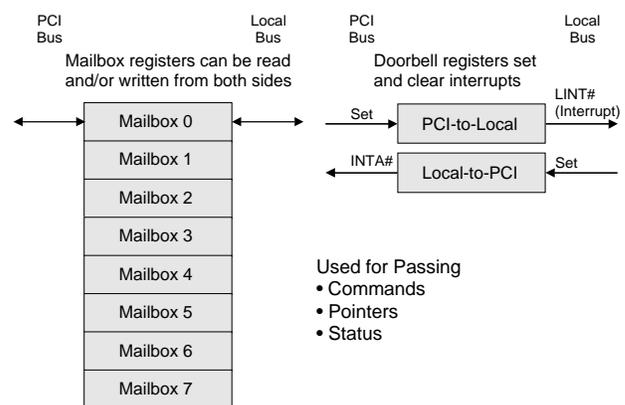


Figure 6-2. Mailbox and Doorbell Message Passing

### 6.1.6.1 Local-to-PCI Doorbell Interrupt

A Local Bus Master can assert a PCI Bus interrupt by writing to the Local-to-PCI Doorbell Register bit(s) (L2PDBELL[31:0]). The PCI Host processor can read the PCI Doorbell Interrupt Active bit to determine whether a PCI Doorbell interrupt is pending (INTCSR[13]), and if so, read the PCI 9054 Local-to-PCI Doorbell register.

Each bit in the Local-to-PCI Doorbell register is individually controlled. The Local Bus can only set bits in the Local-to-PCI Doorbell register. From Local Bus, writing 1 to any bit position sets that bit and writing 0 has no effect. Bits in the Local-to-PCI Doorbell register can only be cleared from the PCI Bus. From the PCI Bus, writing 1 to any bit position clears that bit and writing 0 has no effect.

Interrupts remain set as long as any Local-to-PCI Doorbell register bits are set and the PCI Doorbell Interrupt Enable bit (INTCSR[9]) is set.

#### 6.1.6.1.1 M Mode Local-to-PCI Doorbell Interrupt

To prevent race conditions from occurring when the PCI Bus is accessing the Local-to-PCI Doorbell register (or any Configuration register), the PCI 9054 automatically de-asserts TA# output to prevent Local Bus configuration accesses.

#### 6.1.6.1.2 C and J Modes Local-to-PCI Doorbell Interrupt

To prevent race conditions from occurring when the PCI Bus is accessing the Local-to-PCI Doorbell register (or any Configuration register), the PCI 9054 automatically de-asserts READY# output to prevent Local Bus configuration accesses.

### 6.1.6.2 PCI-to-Local Doorbell Interrupt

A PCI Bus Master can assert a Local Bus interrupt by writing to the PCI-to-Local Doorbell Register bit(s) (P2LDBELL[31:0]). The Local processor can read the Local Doorbell Interrupt Active bit to determine whether a Local doorbell interrupt is pending (P2LDBELL[20]), and if so, read the PCI 9054 PCI-to-Local Doorbell register.

Each bit in the PCI-to-Local Doorbell register is individually controlled. The PCI Bus only sets bits in the PCI-to-Local Doorbell register. From the PCI Bus, writing 1 to any bit position sets that bit and writing 0 to a bit position has no effect. Bits in the PCI-to-Local Doorbell register can only be cleared from the Local Bus. From the Local Bus, writing 1 to any bit position clears that bit and writing 0 has no effect.

**Note:** If the Local Bus cannot clear a Doorbell Interrupt, do not use the PCI-to-Local Doorbell register.

Interrupts remain set as long as any PCI-to-Local Doorbell register bits are set and the Local Doorbell Interrupt Enable bit is set (INTCSR[17]=1).

To prevent race conditions when the Local Bus is accessing the PCI-to-Local Doorbell register (or any Configuration register), the PCI 9054 automatically issues a Retry to the PCI Bus.

### 6.1.7 Built-In Self Test Interrupt (BIST)

A PCI Bus Master can assert a Local Bus interrupt by performing a PCI Type 0 Configuration write to a bit in the PCI BIST register. A Local processor can read the BIST Interrupt Active bit (INTCSR[23]) to determine whether a BIST interrupt is pending.

Interrupts remain set as long as the bit is set and the PCI BIST Interrupt Enable bit is set (PCIBISTR[6]=1). The Local Bus then resets the bit when BIST completes. The PCI Host software may fail the device if the bit is not reset after two seconds.

**Note:** The PCI 9054 does not have an internal BIST.

### 6.1.8 DMA Channel 0/1 Interrupts

A DMA channel can assert a PCI Bus or Local Bus interrupt when done (transfer complete) or after a transfer is complete for the current descriptor in Scatter/Gather DMA mode. The DMA Channel Interrupt Select bit(s) determine whether to assert a PCI (DMAMODE0[17]=1 and/or DMAMODE1[17]=1) or Local (DMAMODE0[17]=0 and/or DMAMODE1[17]=0) interrupt. The Local or PCI processor can read the DMA Channel 0 Interrupt Active bits to determine whether a DMA Channel 0 (INTCSR[21]) or DMA Channel 1 (INTCSR[22]) interrupt is pending.

The Channel Done bit(s) (DMACSR0[4] and/or DMACSR1[4]) can be used to determine whether an interrupt is one of the following:

- DMA Done interrupt
- Transfer complete for current descriptor interrupt

The Done Interrupt Enable bit(s) (DMAMODE0[10] and/or DMAMODE1[10]) enable a Done interrupt. In Scatter/Gather DMA mode, a bit in the Next Descriptor Pointer register of the channel (loaded from Local memory) specifies whether to assert an interrupt at the end of the transfer for the current descriptor.

A DMA Channel interrupt is cleared by the Channel Clear Interrupt bit(s) (DMACSR0[3]=1 and/or DMACSR1[3]=1).

### 6.1.9 All Modes PCI SERR# (PCI NMI)

The PCI 9054 asserts an SERR# pulse if parity checking is enabled (PCICR[6]=1) and it detects an address or 1 is written to the Generate PCI Bus SERR# Interrupt bit (INTCSR[2]) with a current value of 0.

SERR# output can be enabled or disabled with the SERR# Enable bit (PCICR[8]).

### 6.1.10 M Mode PCI SERR#

The PCI 9054 also asserts SERR# if the Local Bus responds with TEA# to the PCI 9054. The TEA# Input Interrupt Mask bit (LMISC[5]) masks out the SERR# interrupt assertion process.

### 6.1.11 Local NMI

If the Parity Error Response bit is set (PCICR[6]=1), the PCI 9054 sets the Master Data Parity Error Detected bit (PCISR[8]=1) when the following three conditions are met:

- The PCI 9054 asserted PERR# or acknowledged PERR# was asserted
- The PCI 9054 was the Bus Master for the operation in which the error occurred
- The Parity Error Response bit is set (PCICR[6]=1)

The PCI 9054 sets the Detected Parity Error bit (PCISR[15]=1) if it detects one of the following conditions:

- The PCI 9054 detected a parity error during a PCI Address phase
- The PCI 9054 detected a data parity error when it was the Target of a write
- The PCI 9054 detected a data parity error when performing Master Read operation

### 6.1.12 M Mode Local TEA# (Local NMI)

A TEA# interrupt is asserted if the following occurs:

- PCI Bus Target Abort bit is set (PCISR[11]=1) or Received Master Abort bit is set (PCISR[13]=1)
- Detected Parity Error bit is set (PCISR[15]=1)
- PCI Initiator Local Data Parity Check Error Status bit is set (INTCSR[7]=1)
- Messaging Outbound Free queue overflows

The Enable Local Bus TEA# bit (INTCSR[0]) can be used to enable or disable TEA# for an abort or parity error. TEA# is a level output that remains asserted as long as the Abort or Parity Error Status bits are set.

The PCI 9054 tolerates TEA# input assertion only during PCI Target or DMA transactions. The PCI 9054 does not sample TEA# assertion during PCI Initiator transactions.

### 6.1.13 C and J Modes Local LSERR# (Local NMI)

An LSERR# interrupt is asserted if the following conditions occur:

- PCI Bus Target Abort bit is set (PCISR[11]=1) or Received Master Abort bit is set (PCISR[13]=1)
- Detected Parity Error bit is set (PCISR[15]=1)
- PCI Initiator Local Data Parity Check Error Status bit is set (INTCSR[7]=1)
- Messaging Outbound Free queue overflows

The Enable Local Bus LSERR# bit (INTCSR[0]) can be used to enable or disable LSERR# for an abort or parity error. LSERR# is a level output that remains asserted as long as the Abort or Parity Error Status bits are set.

## 6.2 USER INPUT AND OUTPUT

The PCI 9054 supports user input and output pins, USERi and USERo (PQFP—pins 159-158; PBGA—pins C7 and A7), respectively. Both are multiplexed with other functional pins. The default PCI 9054 condition are the USERi and USERo functions. USERi is selected when CNTRL[18]=1. USERo is selected when CNTRL[19]=1. User output data can be logged by writing to the General Purpose Output bit (CNTRL[16]). User input data can be read from the General Purpose Input bit (CNTRL[17]).

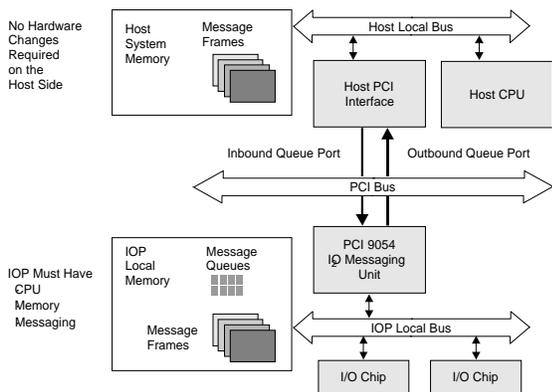


# 7 INTELLIGENT I/O (I<sub>2</sub>O)

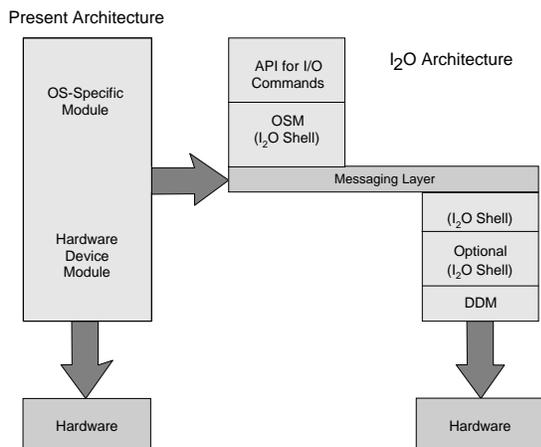
## 7.1 I<sub>2</sub>O-COMPATIBLE MESSAGE UNIT

The I<sub>2</sub>O-compatible Messaging Unit supplies two paths for messages, two inbound FIFOs to receive messages from the primary PCI Bus, and two outbound FIFOs to pass messages to the primary PCI Bus. Refer to *I<sub>2</sub>O Architecture Specification v1.5* for details.

Figure 7-1 and Figure 7-2 illustrate I<sub>2</sub>O architecture.



**Figure 7-1. Typical I<sub>2</sub>O Server/Adapter Card Design**



**Figure 7-2. Driver Architecture Compared**

### 7.1.1 Inbound Messages

Inbound messages reside in a pool of message frames (minimum 64-byte frames) allocated in the shared Local Bus (IOP) memory. The inbound message queue is comprised of a pair of rotating

FIFOs implemented in Local memory. The Inbound Free List FIFO holds the message frame addresses (MFA) of available message frames in Local memory. The Inbound Post Queue FIFO holds the MFA of all currently posted messages.

External PCI agents, through the Inbound Queue Port location in PCI Address space access inbound circular FIFOs. The Inbound Queue Port, when read by an external PCI agent, returns the Inbound Free List FIFO MFA. The external PCI agent places a message frame into the Inbound Post Queue FIFO by writing its MFA to the Inbound Queue Port location.

### 7.1.2 Outbound Messages

Outbound messages reside in a pool of message frames (minimum 64-byte frames) allocated in the shared PCI Bus (Host System) memory. The Outbound message queue is comprised of a pair of rotating FIFOs implemented in Local memory. The Outbound Free List FIFO holds the message frame addresses (MFA) of available message frames in system memory. The Outbound Post Queue FIFO holds the MFA of all currently posted messages.

External PCI agents, through the Outbound Queue Port location in PCI Address space access outbound circular FIFOs. The Outbound Queue Port, when read by an external PCI agent, returns the Outbound Post Queue FIFO MFA. The External PCI agent places free message frames into the Outbound Free List FIFO by writing the free MFA into the Outbound Queue Port location.

Memory for the circular FIFOs themselves must be allocated in Local (IOP) memory. The base address of the queue is contained in the Queue Base Address bits (QBAR[31:20]). Each FIFO entry is a 32-bit data value. Each read and write of the queue must be a single 32-bit access.

Circular FIFOs range in size from 4-KB to 64-KB entries. All four FIFOs must be the same size and contiguous. Therefore, the total amount of Local memory needed for circular FIFOs ranges from 64 KB to 1 MB. A FIFO size is specified in the Circular Queue Size bits (MQCR[5:1]).

The starting address of each FIFO is based on the Queue Base Address and the FIFO Size, as listed in Table 7-1.

**Table 7-1. Queue Starting Address**

FIFO	Starting Address
Inbound Free List	QBAR
Inbound Post List	QBAR + (1 * FIFO Size)
Outbound Post List	QBAR + (2 * FIFO Size)
Outbound Free List	QBAR + (3 * FIFO Size)

### 7.1.3 I<sub>2</sub>O Pointer Management

The FIFOs always reside in shared Local (IOP) memory and are allocated and initialized by the IOP. Before setting the Queue Enable bit (MQCR[0]=1), the Local processor must initialize the following registers, with the initial offset according to the configured FIFO size:

- Inbound Post and Free Head Pointer registers (IPHPR)
- Inbound Post and Free Tail Pointer registers (IPTPR)
- Outbound Post and Free Head Pointer registers (OFHPR)
- Outbound Post and Free Tail Pointer registers (OFTPR)

The Messaging Unit automatically adds the Queue Base Address to offset in each head and tail pointer register. The software can then enable I<sub>2</sub>O. After initialization, the Local software should not write to the pointers managed by the MU hardware.

Empty flags are set if the queues are disabled (MQCR[0]=0) or head and tail pointers are equal. This occurs independent of how the head and tail pointers are set.

An empty flag is cleared, signifying not empty, only if the queues are enabled and pointers become not equal.

If an empty flag is cleared and the queues are enabled, the empty flag is set only if the tail pointer is incremented and the head and tail pointers become equal.

Full flags are always cleared when the queues are disabled or the head and tail pointers are not equal.

A full flag is set when the queues are enabled, the head pointer is incremented, and the head and tail pointers become equal.

Each circular FIFO has a head pointer and a tail pointer, which are offsets from the Queue Base Address. Writes to a FIFO occur at the head of the FIFO and reads occur from the tail. Head and tail pointers are incremented by either the Local processor or the MU hardware. The unit that writes to the FIFO also maintains the pointer. Pointers are incremented after a FIFO access. Both pointers wrap around to the first address of the circular FIFO when they reach the FIFO size, so that the head and tail pointers continuously “chase” each other around in the circular FIFO. The MU wraps the pointers automatically for the pointers that it maintains. IOP software must wrap the pointers that it maintains. Whenever they are equal, the FIFO is empty. To prevent overflow conditions, I<sub>2</sub>O specifies that the number of message frames allocated should be less than or equal to the number of entries in a FIFO. (Refer to Figure 7-3.)

Each inbound MFA is specified by I<sub>2</sub>O as the offset from the start of shared Local (IOP) memory region 0 to the start of the message frame. Each outbound MFA is specified as the offset from Host memory location 0x00000000h to the start of the message frame in shared Host memory. Because the MFA is an actual address, the message frames need not be contiguous. IOP allocates and initializes inbound message frames in shared IOP memory using any suitable memory allocation technique. Host allocates and initializes outbound message frames in shared Host memory using any suitable memory allocation technique. Message frames are a minimum of 64 bytes in length.

I<sub>2</sub>O uses a “push” (write preferred) memory model. That means the IOP writes messages and data to the shared Host memory, and the Host writes messages and data to shared IOP memory. Software should make use of Burst and DMA transfers whenever possible to ensure efficient use of the PCI Bus for message passing.

Additional information on message passing implementation may be found in *I<sub>2</sub>O Architecture Specification v1.5*.

### 7.1.4 Inbound Free List FIFO

The Local processor allocates inbound message frames in its shared memory and can place the address of a free (available) message frame into the Inbound Free List FIFO by writing its MFA into the

FIFO location pointed to by the Queue Base register + Inbound Free Head Pointer register. The Local processor must then increment the Inbound Free Head Pointer register.

A PCI Master (Host or another IOP) can obtain the MFA of a free message frame by reading the Inbound Queue Port Address (40h of the first PCI Memory Base Address register). If the FIFO is empty (no free inbound message frames are currently available, head and tail pointers are equal), the MU returns -1 (FFFFFFFFh). If the FIFO is not empty (head and tail pointers are not equal), the MU reads the MFA pointed to by the Queue Base register + Inbound Free Tail Pointer register, returns its value and increments the Inbound Free Tail Pointer register. If the Inbound Free Queue is not empty, and the Inbound Free Queue Prefetch Enable bit is set (QSR[3]=1), the next entry in the FIFO is read from the Local Bus into a prefetch register. The prefetch register then provides the data for the next PCI read from this queue, thus reducing the number of PCI wait states. (Refer to Figure 7-3.)

### 7.1.5 Inbound Post Queue FIFO

A PCI Master (Host or another IOP) can write a message into an available message frame in the shared Local (IOP) memory. It can then post that message by writing the Message Frame Address (MFA) to the Inbound Queue Port Address, IQP (40h of the first PCI Memory Base Address register). When the port is written, the MU writes the MFA to the Inbound Post Queue FIFO location pointed to by the Queue Base register + FIFO Size + Inbound Post Head Pointer register. After the MU writes the MFA to the Inbound Post Queue FIFO, it increments the Inbound Post Head Pointer register.

The Inbound Post Tail Pointer register points to the Inbound Post Queue FIFO location, which holds the MFA of the oldest posted message. The Local processor maintains the tail pointer. After a Local processor reads the oldest MFA, it can remove the MFA from the Inbound Post Queue FIFO by incrementing the Inbound Post Tail Pointer register.

The PCI 9054 asserts a Local Interrupt when the Inbound Post Queue FIFO is not empty. The Inbound Post Queue FIFO Interrupt bit in the Queue Status/Control register (QSR[5]) indicates the interrupt status. The interrupt clears when the Inbound Post Queue

FIFO is empty. The Inbound Post Queue FIFO Interrupt Mask bit (QSR[4]) can mask the interrupt.

To prevent racing between the time the PCI Write transaction is received until the data is written in Local memory and the Inbound Post Head Pointer register is incremented, any PCI Target access to the PCI 9054 is issued a Retry.

### 7.1.6 Outbound Post Queue FIFO

A Local Master (IOP) can write a message into an available message frame in shared Host memory. It can then post that message by writing the Message Frame Address (MFA) to the Outbound Post Queue FIFO location pointed to by the Queue Base register + Outbound Post Head Pointer register + (2 \* FIFO Size). The Local processor should then increment the Outbound Post Head Pointer register.

A PCI Master can obtain the MFA of the oldest posted message by reading the Outbound Queue Port Address (44h of the first PCI Memory Base Address register). If the FIFO is empty (no more outbound messages are posted, head and tail pointers are equal), the MU returns -1 (FFFFFFFFh). If the Outbound Post Queue FIFO is not empty (head and tail pointers are not equal), the MU reads the MFA pointed to by the Queue Base register + (2 \* FIFO Size) + outbound Post Tail Pointer register, returns its value and increments the Outbound Post Tail Pointer register.

The PCI 9054 asserts a PCI Interrupt when the Outbound Post Head Pointer register is not equal to the Outbound Post Tail Pointer register. The Outbound Post Queue FIFO Interrupt bit of the Outbound Post Queue Interrupt Status register (OPQIS) indicates the interrupt status. When the pointers become equal, both the interrupt and the Outbound Post Queue FIFO interrupt bit are automatically cleared. Pointers become equal when a PCI Master (Host or another IOP) reads sufficient FIFO entries to empty the FIFO. The Outbound Post Queue FIFO Interrupt Mask register (OPLFIM) can mask the Interrupt.

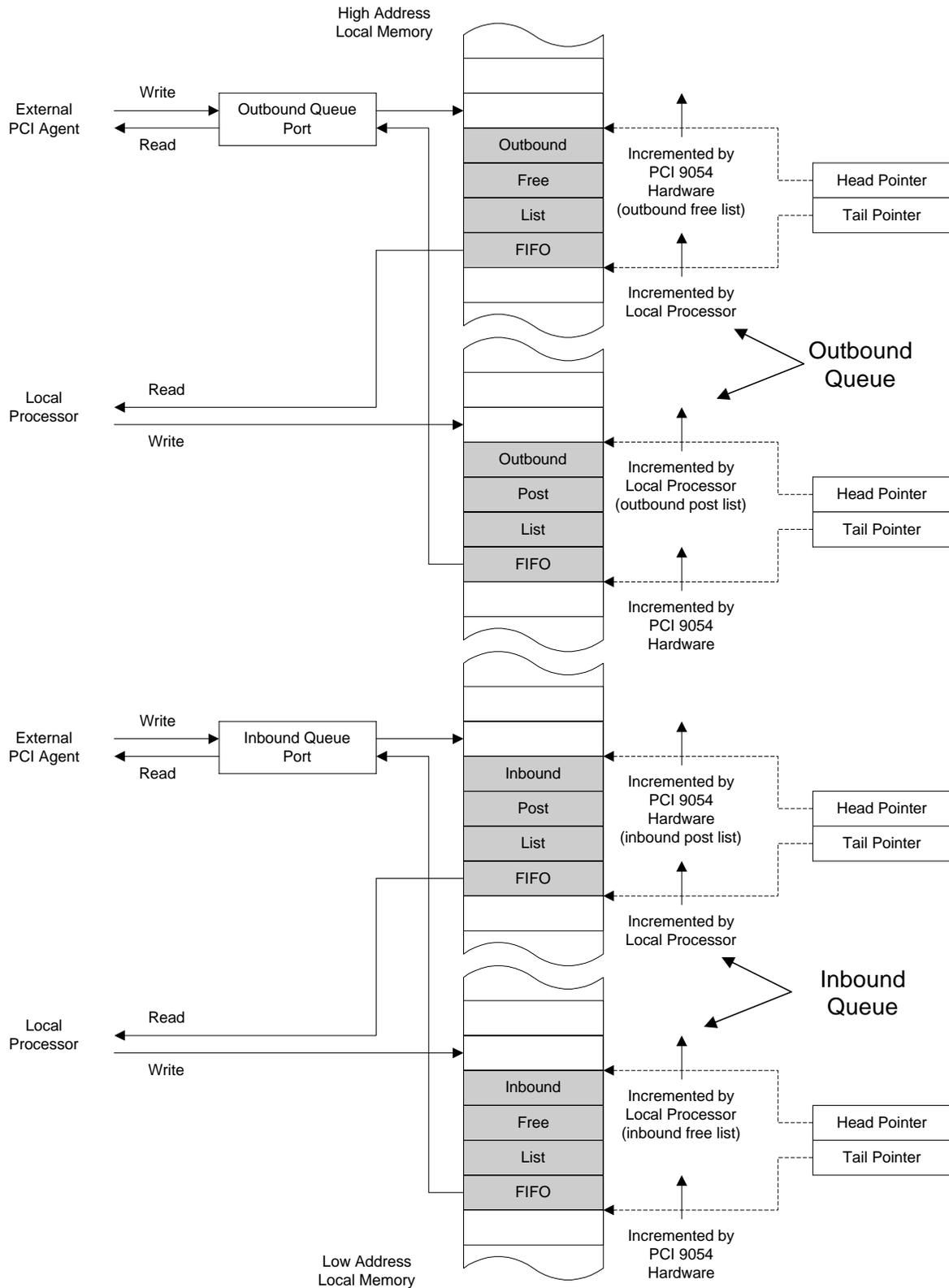


Figure 7-3. Circular FIFO Operation

### 7.1.7 Outbound Post Queue

To reduce read latency, prefetching from the tail of the queue occurs whenever the queue is not empty and the tail pointer is incremented (queue has been read from), or when the queue is empty and the head pointer is incremented (queue has been written to). When the Host CPU reads the Outbound Post Queue, the data is immediately available.

### 7.1.8 Inbound Free Queue

To reduce read latency, prefetching from the tail of the queue occurs whenever the queue is not empty and the tail pointer is incremented (queue has been read from), or when the queue is empty and the head pointer is incremented (queue has been written to). When the Host CPU reads the Inbound Free Queue, the data is immediately available.

### 7.1.9 Outbound Free List FIFO

The PCI Bus Master (Host or other IOP) allocates outbound message frames in its shared memory. The PCI Bus Master can place the address of a free (available) message frame into the Outbound Free List FIFO by writing a Message Frame Address (MFA) to the Outbound Queue Port Address (44h of the first PCI Memory Base Address register). When the port is written, the MU writes the MFA to the Outbound Free List FIFO location pointed to by the Queue Base register + (3 \* FIFO Size) + Outbound Free Head Pointer register. After the MU writes the MFA to the Outbound Free List FIFO, it increments the Outbound Free Head Pointer register.

When the IOP needs a free outbound message frame, it must first check whether any free frames are available. If the Outbound Free List FIFO is empty (outbound free head and tail pointers are equal), the IOP must wait for the Host to place additional outbound free message frames in the Outbound Free List FIFO. If the Outbound Free List FIFO is not empty (head and tail pointers are not equal), the IOP can obtain the MFA of the oldest free outbound message frame by reading the location pointed to by the Queue Base register + (3 \* FIFO Size) + Outbound Free Tail Pointer register. After the IOP reads the MFA, it must increment the Outbound Free Tail Pointer register. To prevent overflow conditions, I<sub>2</sub>O specifies the number of message frames allocated should be less than or

equal to the number of entries in a FIFO. The MU also checks for overflows of the Outbound Free List FIFO. When the head pointer is incremented and becomes equal to the tail pointer, the Outbound Free List FIFO is full, and the MU asserts a Local TEA#/LSERR# (NMI) interrupt. The interrupt is recorded in the Queue Status/Control register (QSR).

From the time the PCI Write transaction is received until the data is written into Local memory and the Outbound Free Head Pointer register is incremented, any PCI Target access to the PCI 9054 is issued a Retry.

### 7.1.10 I<sub>2</sub>O Enable Sequence

To enable I<sub>2</sub>O, the Local processor should perform the following:

- Initialize Space 1 address and range
- Initialize all FIFOs and Message Frame memory
- Set the PCI Base Class Code bits (PCICCR[23:16]) to be an I<sub>2</sub>O device with programming interface 01h
- Set the I<sub>2</sub>O Decode Enable bit (QSR[0])
- Set Local Init Status bit to “done” (LMISC[2]=1)

**Note:** The serial EEPROM must not set the Local Init Status bit so that the PCI 9054 issues retries to all PCI accesses until the Local Init Status bit is set to “done” by the Local processor.

The I<sub>2</sub>O Decode Enable bit (QSR[0]) causes remapping of resources for use in I<sub>2</sub>O mode. When set, all Memory-Mapped Configuration registers (for example, queue ports 40h and 44h) and Space 1 share the PCIBAR0 register. PCI accesses to offset 00h-FFh of PCIBAR0 result in accesses to the PCI 9054 Internal Configuration registers.

Accesses above offset FFh of PCIBAR0 result in Local Space accesses, beginning at offset 100h from the Remap PCI Address to Local Address Space 1 into the Local Address Space bit(s) (LAS1BA[31:4]). Therefore, space located at offset 00h-FFh from LAS1BA is not addressable from the PCI Bus using PCIBAR0.

**Note:** Because PCI accesses to offset 00h-FFh of PCIBAR0 result in internal configuration accesses, the Inbound Free MFAs must be greater than FFh.

Table 7-2. Circular FIFO Summary

FIFO Name	PCI Port	Generate PCI Interrupt	Generate Local Interrupt	Head Pointer Maintained By	Tail Pointer Maintained By
Inbound Free List FIFO	Inbound Queue Port (Host read)	No	No	Local processor	MU hardware
Inbound Pos List FIFO	Inbound Queue Port (Host write)	No	Yes, when Port is written	MU hardware	Local processor
Outbound Post List FIFO	Outbound Queue Port (Host read)	Yes, when FIFO is not empty	No	Local processor	MU hardware
Outbound Free List FIFO	Outbound Queue Port (Host write)	No	Yes, (TEA#/LSERR#) when FIFO is full	MU hardware	Local processor

# 8 PCI POWER MANAGEMENT

## 8.1 OVERVIEW

The PCI Bus Power Management Interface Specification provides a standard mechanism for operating systems to control add-in cards for power management. The Specification defines four PCI functional power states—D<sub>0</sub>, D<sub>1</sub>, D<sub>2</sub>, and D<sub>3</sub>. States D<sub>0</sub> and D<sub>3</sub> are required, while states D<sub>1</sub> and D<sub>2</sub> are optional. State D<sub>0</sub> represents the highest power consumption and state D<sub>3</sub> the least.

- **D<sub>0</sub> (Uninitialized)**—Enters this state from Power-On Reset or from state D<sub>3hot</sub>. Supports PCI Target PCI transactions only.
- **D<sub>0</sub> (Active)**—All functions active.
- **D<sub>1</sub>**—Uses less power than State D<sub>0</sub>, and more than state D<sub>2</sub>. Light sleep state.
- **D<sub>2</sub>**—Uses very little power.

The functional states are defined by the allowed activities of the add-in card with the PCI 9054.

The function supports PCI Configuration cycles to function if clock is running (Memory, I/O, Bus Mastering, and Interrupts are disabled). It also supports the Wakeup Event from function, but not standard PCI interrupts.

- **D<sub>3hot</sub>**—Uses lower power than any other state. Supports PCI Configuration cycles to function if clock is running. Supports Wakeup Event from function, but not standard PCI interrupts. When programmed for state D<sub>0</sub>, an internal soft reset occurs. The PCI Bus drivers must be disabled. PME# context must be retained during this soft reset.
- **D<sub>3cold</sub>**—No power. Supports Bus reset only. All context is lost in this state.

From a power management perspective, the PCI Bus can be characterized at any point in time by one of four power management states—B<sub>0</sub>, B<sub>1</sub>, B<sub>2</sub>, and B<sub>3</sub>:

- **B<sub>0</sub> (Fully On)**—Bus is fully usable with full power and clock frequency, PCI v2.1 compliant. Fully operational bus activity. This is the only Power Management state in which data transactions can occur.

- **B<sub>1</sub>**—Intermediate power management state. Full power with clock frequency, PCI v2.1 compliant. PME Event driven bus activity. Vcc is applied to all devices on the bus, and no transactions are allowed to occur on the bus.
- **B<sub>2</sub>**—Intermediate power management state. Full power clock frequency stopped, PCI v2.1 compliant (in the low state). PME Event-driven bus activity. Vcc is applied to all devices on the bus; however, the clock is stopped and held in the Low state.
- **B<sub>3</sub> (Off)**—Power to the bus is switched off. PME Event-driven bus activity. Vcc is removed from all devices on the PCI Bus.

All system PCI Buses have an originating device, which can support one or more power states. In most cases, this creates a bridge (such as a Host-Bus-to-PCI-Bus or a PCI-to-PCI bridge).

Function States must be at the same or lower energy state than the bus on which they reside.

### 8.1.1 PCI Power Management Functional Description

The PCI 9054 passes power management information and has no inherent power-saving feature.

The PCI Status register (PCISR) and the New Capability Pointer register (CAP\_PTR) indicate whether a new capability (the Power Management function) is available. The New Capability Functions Support bit (PCISR[4]) enables a PCI BIOS to identify a New Capability function support. This bit is executable for writes from the Local Bus, and reads from both the Local and PCI Buses. CAP\_PTR provides an offset into PCI Configuration Space, the start location of the first item in a New Capabilities Linked List.

The Power Management Capability ID register (PMCAPID) specifies the Power Management Capability ID, 01h, assigned by the PCI SIG. The Power Management Next Capability Pointer register (PMNEXT) points to the first location of the next item in the capabilities linked list. If Power Management is the last item in the list, then this register should be set to 0. The default value for the PCI 9054 is 48h (Hot Swap).

For the PCI 9054 to change the power state and assert PME#, the Local Host or PCI Host should set

the PME#\_En bit (PMCSR[8]=1). The Local Host then determines to which power state the backplane should change by reading the Power\_State bits (PMCSR[1:0]).

The Local Host sets up the following:

- D<sub>2</sub>\_Support and D<sub>1</sub>\_Support bits (PMC[10:9]) are used by the Local Host to identify power state support
- PME#\_Support bits (PMC[14:11]) are used by the PCI 9054 to identify the PME# Support correspondent to a specific power state (PMCSR[1:0])

The Local Host then sets the PME#\_Status bit (PMCSR[15]=1) and the PCI 9054 asserts PME#. To clear the PME#\_Status bit, the PCI Host must write a 1 to the PME# Status bit (PMCSR[15]=1). To disable the PME# Interrupt signal, either Host can write a 0 to the PME#\_En bit (PMCSR[8]=0).

LINT# output is asserted every time the power state in the PMCSR register changes. Transmission from state 11 (D<sub>3hot</sub>) to state 00 (D<sub>0</sub>) causes a soft reset. A soft reset should only be initiated from the PCI Bus because the Local Bus interface is reset during a soft reset. The PCI 9054 issues LRESET# and resets all its internal registers to their default values. In state D<sub>3hot</sub>, PCI Memory and I/O accesses are disabled, as well as PCI interrupts, and only configuration is allowed. Before making LINT# output work, set the Power Management Interrupt Enable bit (INTCSR[4]=1), and clear the interrupt by setting the Power Management Interrupt bit (INTCSR[5]=1).

The Data\_Scale bits (PMCSR[14:13]) indicate the scaling factor to use when interpreting the value of the Power Management Data bits (PMDATA[7:0]). The value and meaning of the bits depend upon the data value specified in the Data\_Select bits (PMCSR[12:9]). The Data\_Scale bit value is unique for each Data\_Select bit. For Data\_Select values from 8 to 15, the Data\_Scale bits always return a zero (PMCSR[14:13]=0).

PMDATA provides operating data, such as power consumed or heat dissipation.

## 8.1.2 System Changes Power Mode Example

1. The Host writes to the PCI 9054 PMCSR register to change the power states.
2. The PCI 9054 sends a local interrupt (LINT# output) to a Local CPU (LCPU).
3. The LCPU has 200 μs to read the power management information from the PCI 9054 PMCSR register to implement the power saving function.
4. After the LCPU implements the power saving function, the PCI 9054 disables all PCI Target accesses and PCI Interrupt output (INTA#). In addition, the BIOS disables the PCI 9054 Master Access Enable bit (PCICR[2]).

**Notes:** In Power Saving mode, all PCI and Local Configuration cycles are granted.

The PCI 9054 automatically performs a soft reset to a Local Bus on D<sub>3</sub>-to-D<sub>0</sub> transitions.

## 8.1.3 Wake-Up Request Example

1. The add-in card (with a PCI 9054 chip installed) is in a powered-down state.
2. The Local CPU performs a write to the PCI 9054 PMCSR register to request a wake-up procedure.
3. As soon as the request is detected, the PCI 9054 drives PME# out to the PCI Bus.
4. The PCI Host accesses the PCI 9054 PMCSR register to disable the PME# output signal and restores the PCI 9054 to the D<sub>0</sub> power state.
5. The PCI 9054 completes the power management task by issuing the Local interrupt (LINT# output) to the Local CPU, indicating that the power mode has changed.

# 9 COMPACTPCI HOT SWAP

## 9.1 HOT SWAP

The PCI 9054 is a CompactPCI Hot Swap-*Friendly*-compliant device.

### 9.1.1 Overview

Hot Swap is used for CompactPCI applications. Hot Swap functionality allows the orderly insertion and removal of boards without adversely affecting system operation. This is done for repair of faulty boards or system reconfiguration. Additionally, Hot Swap provides programmatic access to Hot Swap services allowing system reconfiguration and fault recovery to occur with no system down time and minimum operator interaction. Control of adapter insertion/removal logic resides on the individual adapters. The PCI 9054 uses two pins, ENUM# and LEDon/LEDin, to implement the hardware aspects of Hot Swap functionality. The PCI 9054 uses the Hot Swap Capabilities register to implement the software aspects of Hot Swap capabilities.

To avoid confusion in the industry, Hot Swap defines three levels of compatibility:

- Hot Swap-*Capable* devices contain the minimum requirements to operate in Hot Swap environment
- Hot Swap-*Friendly* devices contain additional functions to ease the job of the user
- Hot Swap-*Ready* devices contain all the necessary functions for Hot Swap

Hot Swap-*Capable* requirements are mandatory for a device to be used in Hot Swap environment. These requirements are attributes for which a system user must compensate using external circuitry, as follows:

- PCI Specification v2.1 compliant
- Tolerate Vcc from early power
- Tolerate precharge voltage
- Limited I/O pin leakage at precharge voltage

Hot Swap-*Friendly* silicon includes all the required *Capable* functions and adds some of the functions in the following list. These functions are possible to add externally to the device. The PCI 9054 has integrated them into the PCI silicon, thus reducing the amount of external circuitry that a user must add.

- **Incorporates Hot Swap Control/Status register (HS\_CSR)**—Contained within the configuration space.
- **Incorporates an Extended Capability Pointer (ECP) mechanism**—Software must have a standard method of determining if a specific function is designed in accordance with the specification. The Capabilities Pointer is located within standard CSR space, using a bit in the PCI Status register (offset 04h).

The PCI 9054 has additional resources for software control of the ENUM# ejector switch and the blue LED to indicate insertion/removal.

The PCI 9054 is a Hot Swap-*Friendly* PCI silicon device. The PCI 9054 has incorporated all compliant functions defined by the CompactPCI Hot Swap specification. The PCI 9054 incorporates LEDon/LEDin and ENUM# and Hot Swap Capabilities registers—HS\_CNTL, HS\_NEXT, and HS\_CSR.

### 9.1.2 Controlling Connection Processes

The following sections are excerpts from *CompactPCI Specification*. Refer to the specification for more details.

#### 9.1.2.1 Hardware Connection Control

Hardware Control provides a means for the platform to control the hardware connection process. The signals listed in the following sections must be supported on all Hot Swap boards for interoperability. Implementations on different platforms may vary.

### 9.1.2.1.1 Board Slot Control

BD\_SEL# is one of the shortest pins. It is driven low to enable power-on. For systems not implementing hardware control, it is grounded on the backplane.

Systems implementing hardware control radially connect BD\_SEL# to a Hot Swap Controller (HSC). The controller terminates the signal with a weak pull-down. The controller can detect board present when the board pull-up overrides the pull-down. HSC can then control the power-on process by driving BD\_SEL# low.

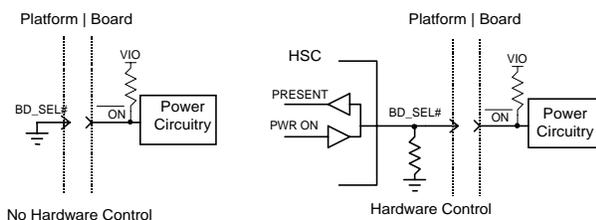


Figure 9-1. Redirection of BD\_SEL#

### 9.1.2.1.2 Board Healthy

A second radial signal is used to acknowledge board health. It signals that a board is suitable to be released from reset and allowed onto the PCI Bus.

Minimally, this signal must be connected to the board's power controller "power good" status line. Use of HEALTHY# can be expanded for applications requiring additional conditions to be met for the board to be considered healthy.

On platforms that do not use Hardware Connection Control, this line is not monitored. Platforms implementing this signaling route these signals radially to a Hot Swap Controller.

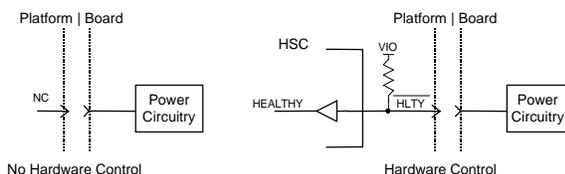


Figure 9-2. Board Healthy

### 9.1.2.1.3 Platform Reset

Reset (PCI\_RST#), as defined by *CompactPCI Specification*, is a bused signal on the backplane, driven by the Host. Platforms may implement this signal as a radial signal from the Hot Swap Controller to further control the electrical connection process. To maintain function of the bused signal, platforms that do this must OR the Host reset signal with the slot-specific signal.

Locally, boards must not come out of reset until the H1 State is reached (healthy), but they must also honor the backplane reset. The Local board reset (Local\_PCI\_RST#) must be the logical OR of these two conditions. Local\_PCI\_RST# is connected to the PCI 9054 RST# input pin.

During a BIOS voltage precharge and platform reset, in insertion and extraction procedures, all PCI I/O buffers must be in a High Impedance state. The PCI 9054 supports this condition any time the Host RST# is asserted (PCI v2.1). To protect the Local board components from early power, the PCI 9054 floats the Local Bus I/Os. The TEST pin can be used to perform the High Impedance condition on the Local Bus. Both the RST# and TEST signals can be simultaneously asserted. The TEST signal is de-asserted some time before the Host RST# is de-asserted to ensure the PCI 9054 asserts the LRESETo# signal to complete a reset task of the Local board.

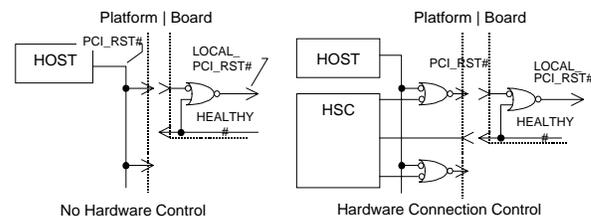


Figure 9-3. PCI Reset

### 9.1.2.2 Software Connection Control

Software Connection Control provides a means to control the Software Connection Process. Resources on the hot swap board facilitate software Connection Control. Access to these resources occurs by way of the bus, using PCI protocol transfers (in-band).

These resources consist of four elements:

- ENUM# driven active indicates the need to change the state of the Hot Swap board
- A switch, tied to the ejector, indicates the intent to remove a board
- LED indicates the status of the software connection process
- Control/Status register allows the software to interact with these resources

#### 9.1.2.2.1 Ejector Switch and Blue LED

A microswitch (switch), located in the board-ejector mechanism of the Hot Swap CompactPCI board, is used to signal the impending removal of a board. This signal asserts ENUM#. The operator normally activates the switch, waits for the LED illumination to indicate it is okay to remove the board, and then removes the board. The PCI 9054 implements control logic for both the microswitch and the Blue LED in one pin (LEDOn/LEDIn).

When the ejector is opened or closed, the switch bounces for a time. The PCI 9054 uses internal debounce circuitry to clean the signal before the remainder of Hot Swap logic acknowledges it. The state of the switch is acknowledged six times, at 1 ms intervals, before it is assumed closed or open.

The Blue “Status” LED, located on the front of the Hot Swap CompactPCI board, is illuminated when it is permissible to remove a board. The hardware connection layer provides protection for the system during all insertions and extractions. This LED indicates the system software is in a state that tolerates board extraction.

Upon insertion, the LED is automatically illuminated by the hardware until the hardware connection process completes. The LED remains *OFF* until the software uses it to indicate extraction is once again permitted.

The PCI 9054 uses a tri-state I/O pin to drive the external LED. This pin is Time-Division Multiplexed (TDM) for input and output functionality. When used as

an output, it drives the external LED. The LED state is driven from the LED Software On/Off Switch bit (HS\_CSR[3]). When used as an input, it acknowledges the state of the ejector handle. With the implementation of TDM, this pin is usually driving the LED. A small portion of time is dedicated to acknowledging ejector status.

#### 9.1.2.2.2 ENUM#

ENUM# is provided to notify the Host CPU that a board has been freshly inserted or is about to be removed. This signal informs the CPU that configuration of the system has changed. The CPU then performs any necessary maintenance such as installing a device driver upon board insertion, or quiescing a device driver prior to board extraction.

ENUM# is an open collector bused signal with a pull-up on the Host. It may drive an interrupt (preferred) or be polled by the system software at regular intervals. The CompactPCI Hot-Plug System Driver on the system Host manages the ENUM# sensing. Full Hot Swap Boards assert ENUM# until serviced by the Hot-Plug system driver.

Regardless of switch status, when a board is inserted into the system and reset is complete, de-asserted, the PCI 9054 defaults the ENUM# Status Indicator for Board Insertion bit (HS\_CSR[7]) to 1, causing the assertion of the ENUM# interrupt on the PCI Bus. Once the Host CPU has installed the proper drivers, it can logically include this board by clearing the interrupt.

When a board is about to be removed, the PCI 9054 acknowledges the ejector switch is closed (ejector open), asserts ENUM# interrupt and sets the ENUM# Status Indicator for Board Removal bit (HS\_CSR[6]). The Host then logically removes the board and turns on the LED. The operator can then remove the board completely.

#### 9.1.2.2.3 Hot Swap Control/Status Register (HS\_CSR)

The PCI 9054 supports Hot Swap directly, as a control/status register is provided in Configuration space. This register is accessed through the PCI Extended Capabilities Pointer (ECP) mechanism.

The Hot Swap Control/Status register (HS\_CSR) provides status read-back for the Hot-Plug System Driver to determine which board is driving ENUM#. This register is also used to control the Hot Swap

Status LED on the front panel of the board, and to de-assert ENUM#.

The PCI 9054 Hot Swap register (HS\_CSR) is accessible for writes from the Local Bus. When a Local CPU is accessing the Hot Swap register (HS\_CSR) from the Local Bus, it must take care to not write into the ENUM# status bits. Otherwise, the value might be overwritten prior to the PCI Hot Swap software acknowledging it.

#### 9.1.2.2.4 Hot Swap Capabilities Register Bit Definition

31	24	23	16	15	8	7	0
Reserved		Control		Next_Cap Pointer		Hot Swap ID	

Figure 9-4. Hot Swap Capabilities Register Bit Definition

**Hot Swap ID.** Bits [7:0]. These bits are set to a default value of 0x06.

**Next\_Cap Pointer.** Bits [15:8]. These bits either point to the next New Capability structure, or are set to 0 if this is the last capability in the structure.

**Control.** Bits [23:16]. This eight-bit control register is defined in the following table.

Table 9-1. Hot Swap Control

Bit	Description
23	ENUM# status—Insertion (1 = board is inserted).
22	ENUM# status—Removal (1 = board is being removed).
21	Not used.
20	Not used.
19	LED state (1 = LED on, 0 = LED off).
18	Not used.
17	ENUM# interrupt enable (1 = de-assert, 0 = enable interrupt).
16	Not used.

# 10 PCI VITAL PRODUCT DATA (VPD)

## 10.1 OVERVIEW

The Vital Product Data (VPD) function in *PCI Specification v2.2* defines a new location and access method. It also defines the Read Only and Read/Write bits. Currently Device ID, Vendor ID, Revision ID, Class Code, Subsystem ID, and Subsystem Vendor ID are required in the Configuration Space Header and are required for basic identification of the device and device configuration. Though this information allows a device to be configured, it is not sufficient to allow a device to be uniquely identified. With the addition of VPD, optional information is provided that allows a device to be uniquely identified and tracked. These additional bits enable current and/or future support tools and reduces the total cost of ownership of PCs and systems.

This provides an alternate access method other than Expansion ROM for VPD. VPD is stored in an external serial EEPROM, which is accessed using the Configuration Space New Capabilities function.

The VPD registers—PVPDCNTL, PVPD\_NEXT, PVPDAD, and PVPDATA—are not accessible for reads from the Local Bus. It is recommended that the VPD function be exercised only from the PCI Bus.

### 10.1.1 VPD Capabilities Register

**VPD ID.** Bits [7:0]. The PCI SIG assigns these bits a value of 03h by. The VPD ID is hardcoded.

**Next\_Cap Pointer.** Bits [15:8]. These bits either point to the next New Capability structure, or are set to 0 if this is the last capability in the structure. The PCI 9054 defaults to 0x00. This value can be overwritten from the Local Bus.

**VPD Address.** Bits [24:16]. These bits specify the byte address of the VPD to be accessed. All accesses are 32-bits wide; bits [17:16] must be 0, with the maximum serial EEPROM size being 4K bits. Bits [30:25] are ignored.

**F.** Bit 31. This bit sets a flag to indicate when a serial EEPROM data operation is complete. For Write cycles, the four bytes of data are first written into the VPD Data bits, after which the VPD Address is written at the same time the F flag is set to 1. The F flag clears when the serial EEPROM Data transfer completes. For Read cycles, the VPD Address is written at the same time the F flag is cleared to 0. The F flag is set when four bytes of data are read from the serial EEPROM.

**VPD Data.** Bits [31:0]. The VPDDATA register is not a pure read/write register. Data read into the register depends upon the last Read operation performed in VPDDAD[15]. VPD data is written or read through this register. The least-significant byte corresponds to the VPD byte at the address specified by the VPD Address register. Four bytes are always transferred between the register and the serial EEPROM.

31	30	16	15	8	7	0
F	VPD Address		Next_Cap Pointer (0X00)		VPD ID (0x03)	
VPD Data						

Figure 10-1. VPD Capabilities Register

### 10.1.2 VPD Serial EEPROM Partitioning

To support VPD, the serial EEPROM is partitioned into read only and read/write sections.

### 10.1.3 Sequential Read Only

The first 1456 bits, 182 bytes of the serial EEPROM contain read-only information. The read-only portion of the serial EEPROM is loaded into the PCI 9054, using a sequential Read command to serial EEPROM and occurs once after power-on.

### 10.1.4 Random Read and Write

The PCI 9054 can read and write the read/write portion of serial EEPROM. The Serial EEPROM Starting at Lword Boundary for VPD Accesses bits (PROT\_AREA[6:0]) designates this portion. This register is loaded upon power-on and can be written with a desired value starting at location 0. This provides the capability of writing the entire serial EEPROM. Writes to serial EEPROM are comprised of the following three commands:

- Write Enable
- Write Enable, followed by Write data
- Write Disable

This is done to ensure against accidental writes to the serial EEPROM. Random cycles allow VPD information to be written and read at any time.

To perform a simple VPD write to the serial EEPROM, the following steps are necessary:

1. Change the write-protected serial EEPROM address in PROT\_AREA[6:0] to the desired address. 0x0000000 makes the serial EEPROM removable from the beginning.
2. Write desired data into the VPDDATA register.
3. Write destination serial EEPROM address and flag of operation, value of 1.
4. Probe a flag of operation until it changes to a 0 to ensure the write completes.

To perform a simple VPD read from serial EEPROM, the following steps are necessary:

1. Write a destination serial EEPROM address and flag of operation, value of 0.
2. Probe a flag of operation until it changes to a 1 to ensure the Read data is available.
3. Read back the VPDDATA register to see the requested data.

# 11 REGISTERS

## 11.1 NEW REGISTER DEFINITIONS SUMMARY

This section summarizes the new registers, as compared to the PCI 9080. Refer to the following sections for a full explanation of each register.

**Table 11-1. New Registers Definitions Summary (As Compared to the PCI 9080)**

PCI Offset	Local Offset	Register	Bits	Description
08h/ACh	88h/12Ch	Mode/DMA Arbitration	31	WAIT# enable. In M mode, the bit lets WAIT# be rerouted to BIGEND#. In C and J modes, the bit has no effect. When set to 1, WAIT# function is selected. When set to 0, BIGEND# input function is selected.
0Dh	8Dh	Local Miscellaneous Control	22:16 0 1 2 3 4 5 6	Write accessible Address Pointer in the serial EEPROM device for VPD support. I/O Base Address Register enable. I/O Base Address Register shift. Local Initialization Done. Serial EEPROM or Local Host sets bit. M-mode related. BDIP# input enabled during PCI Initiator accesses. M-mode related. Deferred PCI Initiator read. M-mode related. TEA# mask interrupt. M-mode related. PCI Initiator Write FIFO full RETRY# enable.
0Eh	8Eh	Serial EEPROM Write-Protected Address Boundary	6:0	Serial EEPROM starting at the Lword boundary for VPD accesses.
34h	34h	New Capability Pointer	7:0	Provides offset into PCI Configuration space for the location of the first item in the New Capability Linked List.
40h	180h	Power Management	30:0	Provides Power Management ID, Power Management Next Capability Pointer, and Power Management Capabilities.
44h	184h	Power Management	31:0	Provides Power Management Status, PMCSR Bridge Support Extensions, and Power Management Data.
48h	188h	CompactPCI Hot Swap	31:0	Hot Swap Control, Hot Swap Next Capability Pointer, and Hot Swap Control/Status Register.
4Ch	18Ch	PCI Vital Product Data	31:0	VPD ID, VPD Next Capability Pointer, and VPD Address Pointer.
50h	190h	PCI Vital Product Data	31:0	VPD Data.
68h	E8h	Interrupt Control/Status	7:6	Local Data Parity Error Check. Bit 7 is the Parity Error Check Status bit. Bit 6 is the Enable bit.
6Ch	ECh	USER I/O Control	19:18	Bit 18 is a select bit between USERi and LLOCKi#. Bit 19 is a select bit between USERo and LLOCKo#.
17Ch	FCh	PCI Initiator PCI Dual Address Cycle	31:0	PCI Initiator PCI Dual Address Cycle. When set to 0, the PCI 9054 performs a 32-bit PCI Address cycle.
94h	114h	DMA Channel 1 Mode	12	The PCI 9054 does not support Demand Mode DMA by way of Channel 1. <b>Reserved.</b>
B4h	134h	DMA Channel 0 PCI Dual Address Cycle	31:0	DMA Channel 0 PCI Dual Address Cycle. When set to 0, the PCI 9054 performs a 32-bit PCI Address cycle.
B8h	138h	DMA Channel 1 PCI Dual Address Cycle	31:0	DMA Channel 1 PCI Dual Address Cycle. When set to 0, the PCI 9054 performs a 32-bit PCI Address cycle.

## 11.2 REGISTER ADDRESS MAPPING

### 11.2.1 PCI Configuration Registers

Table 11-2. PCI Configuration Registers

PCI Configuration Register Address	Local Access (Offset from Chip Select Address)	To ensure software compatibility with other versions of the PCI 9054 family and to ensure compatibility with future enhancements, write 0 to all unused bits.								PCI Writable	Serial EEPROM Writable
		31	30	24	23	16	15	8	7		
00h	00h	Device ID				Vendor ID				N	Y
04h	04h	Status				Command				Y	N
08h	08h	Class Code					Revision ID			N	Y [31:8]
0Ch	0Ch	BIST		Header Type		PCI Bus Latency Timer		Cache Line Size		Y [7:0]	N
10h	10h	PCI Base Address 0; used for Memory-Mapped Configuration Registers (PCIBAR0)								Y	N
14h	14h	PCI Base Address 1; used for I/O-Mapped Configuration Registers (PCIBAR1)								Y	N
18h	18h	PCI Base Address 2; used for Local Address Space 0 (PCIBAR2)								Y	N
1Ch	1Ch	PCI Base Address 3; used for Local Address Space 1 (PCIBAR3)								Y	N
20h	20h	PCI Base Address 4; used for Local Address Space 2 (PCIBAR4)								Y	N
24h	24h	PCI Base Address 5; used for Local Address Space 3 (PCIBAR5)								Y	N
28h	28h	Cardbus CIS Pointer ( <i>Not supported</i> )								N	N
2Ch	2Ch	Subsystem ID				Subsystem Vendor ID				N	Y
30h	30h	PCI Base Address for Local Expansion ROM								Y	N
34h	34h	<b>Reserved</b>					Next_Cap Pointer			Y [7:0]	N
38h	38h	<b>Reserved</b>								N	N
3Ch	3Ch	Max_Lat		Min_Gnt		Interrupt Pin		Interrupt Line		Y [7:0]	Y [15:8]
40h	180h	Power Management Capabilities				Next_Cap Pointer		Capability ID		Y [31:8]	N
44h	184h	Data		PMCSR Bridge Support Extensions		Power Management Control/Status Register			Y [15, 12:8, 1:0]	N	
48h	188h	<b>Reserved</b>		Control/Status Register		Next_Cap Pointer		Capability ID		Y [23:16], Local [15:0]	Y [15:0]
4Ch	18Ch	F	VPD Address			Next_Cap Pointer		Capability ID		Y [31:16], Local [15:8]	N
50h	190h	VPD Data								Y	N

**Note:** Refer to PCI Specification v2.1 for definitions of these registers.

## 11.2.2 Local Configuration Registers

Table 11-3. Local Configuration Registers

PCI (Offset from Base Address)	Local Access (Offset from Chip Select Address)	To ensure software compatibility with other versions of the PCI 9054 family and to ensure compatibility with future enhancements, write 0 to all unused bits.								PCI/Local Writable	Serial EEPROM Writable
		31	24	23	16	15	8	7	0		
00h	80h	Range for PCI-to-Local Address Space 0								Y	Y
04h	84h	Local Base Address (Remap) for PCI-to-Local Address Space 0								Y	Y
08h	88h	Mode/DMA Arbitration								Y	Y
0Ch	8Ch	<b>Reserved</b>	Serial EEPROM Write-Protected Address Boundary	Local Miscellaneous Control	Big/Little Endian Descriptor			Y	Y		
10h	90h	Range for PCI-to-Local Expansion ROM								Y	Y
14h	94h	Local Base Address (Remap) for PCI-to-Local Expansion ROM and BREQo Control								Y	Y
18h	98h	Local Bus Region Descriptors (Space 0 and Expansion ROM) for PCI-to-Local Accesses								Y	Y
1Ch	9Ch	Range for PCI Initiator-to-PCI								Y	Y
20h	A0h	Local Base Address for PCI Initiator-to-PCI Memory								Y	Y
24h	A4h	Local Base Address for PCI Initiator-to-PCI I/O Configuration								Y	Y
28h	A8h	PCI Base Address (Remap) for PCI Initiator-to-PCI								Y	Y
2Ch	ACh	PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration								Y	Y
F0h	170h	Range for PCI-to-Local Address Space 1								Y	Y
F4h	174h	Local Base Address (Remap) for PCI-to-Local Address Space 1								Y	Y
F8h	178h	Local Bus Region Descriptor (Space 1) for PCI-to-Local Accesses								Y	Y
FCh	17Ch	PCI Base Dual Address Cycle (Remap) for PCI Initiator-to-PCI (Upper 32 bits)								Y	N

### 11.2.3 Runtime Registers

Table 11-4. Runtime Registers

PCI (Offset from Base Address)	Local Access (Offset from Chip Select Address)	To ensure software compatibility with other versions of the PCI 9054 family and to ensure compatibility with future enhancements, write 0 to all unused bits.				PCI/Local Writable	Serial EEPROM Writable
		31	16	15	0		
40h	C0h	Mailbox Register 0 (refer to Note)				Y	Y
44h	C4h	Mailbox Register 1 (refer to Note)				Y	Y
48h	C8h	Mailbox Register 2				Y	N
4Ch	CCh	Mailbox Register 3				Y	N
50h	D0h	Mailbox Register 4				Y	N
54h	D4h	Mailbox Register 5				Y	N
58h	D8h	Mailbox Register 6				Y	N
5Ch	DCh	Mailbox Register 7				Y	N
60h	E0h	PCI-to-Local Doorbell Register				Y	N
64h	E4h	Local-to-PCI Doorbell Register				Y	N
68h	E8h	Interrupt Control/Status				Y	N
6Ch	ECh	Serial EEPROM Control, PCI Command Codes, User I/O Control, and Init Control				Y	N
70h	F0h	Device ID		Vendor ID		N	N
74h	F4h	<b>Unused</b>		Revision ID		N	N
78h	C0h	Mailbox Register (refer to Note)				Y	N
7Ch	C4h	Mailbox Register (refer to Note)				Y	N

**Note:** Mailbox registers 0 and 1 are always accessible at addresses 78h/C0h and 7Ch/C4. When the I<sub>2</sub>O function is disabled (QSR[0]=0), Mailbox registers 0 and 1 are also accessible at PCI addresses 40h and 44h for PCI 9060 compatibility. When the I<sub>2</sub>O function is enabled, the Inbound and Outbound Queue pointers are accessed at addresses 40h and 44h, replacing mailbox registers in PCI Address space.

## 11.2.4 DMA Registers

Table 11-5. DMA Registers

PCI (Offset from Base Address)	Local Access (Offset from Chip Select Address)	To ensure software compatibility with other versions of the PCI 9054 family and to ensure compatibility with future enhancements, write 0 to all unused bits.					PCI/Local Writable	Serial EEPROM Writable
		31	16	15	8	7		
80h	100h	DMA Channel 0 Mode					Y	N
84h	104h	DMA Channel 0 PCI Address					Y	N
88h	108h	DMA Channel 0 Local Address					Y	N
8Ch	10Ch	DMA Channel 0 Transfer Byte Count					Y	N
90h	110h	DMA Channel 0 Descriptor Pointer					Y	N
94h	114h	DMA Channel 1 Mode					Y	N
98h	118h	DMA Channel 1 PCI Address					Y	N
9Ch	11Ch	DMA Channel 1 Local Address					Y	N
A0h	120h	DMA Channel 1 Transfer Byte Count					Y	N
A4h	124h	DMA Channel 1 Descriptor Pointer					Y	N
A8h	128h	<i>Reserved</i>		DMA Channel 1 Command/ Status		DMA Channel 0 Command/ Status	Y	N
ACh	12Ch	Mode/DMA Arbitration					Y	N
B0h	130h	DMA Threshold					Y	N
B4h	134h	DMA Channel 0 PCI Dual Address Cycle (Upper 32 bits)					Y	N
B8h	138h	DMA Channel 1 PCI Dual Address Cycle (Upper 32 bits)					Y	N

### 11.2.5 Messaging Queue Registers

Table 11-6. Messaging Queue Registers

PCI (Offset from Base Address)	Local Access (Offset from Chip Select Address)	To ensure software compatibility with other versions of the PCI 9054 family and to ensure compatibility with future enhancements, write 0 to all unused bits.		PCI/Local Writable	Serial EEPROM Writable
		31	0		
30h	B0h	Outbound Post Queue Interrupt Status		N	N
34h	B4h	Outbound Post Queue Interrupt Mask		Y	N
40h	—	Inbound Queue Port		PCI	N
44h	—	Outbound Queue Port		PCI	N
C0h	140h	Messaging Unit Configuration		Y	N
C4h	144h	Queue Base Address		Y	N
C8h	148h	Inbound Free Head Pointer		Y	N
CCh	14Ch	Inbound Free Tail Pointer		Y	N
D0h	150h	Inbound Post Head Pointer		Y	N
D4h	154h	Inbound Post Tail Pointer		Y	N
D8h	158h	Outbound Free Head Pointer		Y	N
DCh	15Ch	Outbound Free Tail Pointer		Y	N
E0h	160h	Outbound Post Head Pointer		Y	N
E4h	164h	Outbound Post Tail Pointer		Y	N
E8h	168h	Queue Status/Control		Y	N

**Notes:** When I<sub>2</sub>O messaging is enabled (QSR[0]= 1), the PCI Master (Host or another IOP) uses the Inbound Queue Port to read Message Frame Addresses (MFAs) from the Inbound Free List FIFO and to write MFAs to the Inbound Post Queue FIFO. The Outbound Queue Port reads MFAs from the Outbound Post Queue FIFO and writes MFAs to the Outbound Free List FIFO.

Each Inbound MFA is specified by I<sub>2</sub>O as an offset from the PCI Base Address 0 (programmed in PCIBAR0) to the start of the message frame. This means that all inbound message frames should reside in PCI Base Address 0 Memory space.

Each Outbound MFA is specified by I<sub>2</sub>O as an offset from system address 0x00000000h. Outbound MFA is a physical 32-bit address of the frame in shared PCI system memory.

The Inbound and Outbound Queues may reside in Local Address Space 0 or 1 by programming QSR. The queues need not be in shared memory.

### 11.3 PCI CONFIGURATION REGISTERS

All registers may be written to or read from in Byte, Word, or Lword accesses.

**Register 11-1. (PCIIDR; PCI:00h, LOC:00h) PCI Configuration ID**

Bit	Description	Read	Write	Value after Reset
15:0	<b>Vendor ID.</b> Identifies manufacturer of device. Defaults to the PCI SIG-issued Vendor ID of PLX (10B5h) if blank or if no serial EEPROM is present.	Yes	Local/ Serial EEPROM	10B5h or 0
31:16	<b>Device ID.</b> Identifies particular device. Defaults to PLX part number for PCI interface chip (9054h) if blank or no serial EEPROM is present.	Yes	Local/ Serial EEPROM	9054h or 0

**Register 11-2. (PCICR; PCI:04h, LOC:04h) PCI Command**

Bit	Description	Read	Write	Value after Reset
0	<b>I/O Space.</b> Writing a 1 allows the device to respond to I/O space accesses. Writing a 0 disables the device from responding to I/O space accesses.	Yes	Yes	0
1	<b>Memory Space.</b> Writing a 1 allows the device to respond to Memory Space accesses. Writing a 0 disables the device from responding to Memory Space accesses.	Yes	Yes	0
2	<b>Master Enable.</b> Writing a 1 allows device to behave as a Bus Master. Writing a 0 disables device from generating Bus Master accesses.	Yes	Yes	0
3	<b>Special Cycle. Not supported.</b>	Yes	No	0
4	<b>Memory Write and Invalidate Enable.</b> Writing a 1 enables the Memory Write and Invalidate mode for PCI Initiator and DMA. (Refer to the DMA Mode register(s), DMAMODE0[13] and/or DMAMODE1[13].)	Yes	Yes	0
5	<b>VGA Palette Snoop. Not supported.</b>	Yes	No	0
6	<b>Parity Error Response.</b> Writing a 0 indicates parity error is ignored and the operation continues. Writing a 1 indicates parity checking is enabled.	Yes	Yes	0
7	<b>Wait Cycle Control.</b> Controls whether a device does address/data stepping. Writing a 0 indicates the device never does stepping. Writing a 1 indicates the device always does stepping. <i>Note: Hardcoded to 0.</i>	Yes	No	0
8	<b>SERR# Enable.</b> Writing a 1 enables SERR# driver. Writing a 0 disables SERR# driver.	Yes	Yes	0
9	<b>Fast Back-to-Back Enable.</b> Indicates what type of fast back-to-back transfers a Master can perform on the bus. Writing a 1 indicates fast back-to-back transfers can occur to any agent on the bus. Writing a 0 indicates fast back-to-back transfers can only occur to the same agent as in the previous cycle. <i>Note: Hardcoded to 0.</i>	Yes	No	0
15:10	<b>Reserved.</b>	Yes	No	0h

Register 11-3. (PCISR; PCI:06h, LOC:06h) PCI Status

Bit	Description	Read	Write	Value after Reset
3:0	<i>Reserved.</i>	Yes	No	0h
4	<b>New Capability Functions Support.</b> Writing a 1 supports New Capabilities Functions. If enabled, the first New Capability Function ID is located at PCI Configuration offset [40h]. Can be written only from the Local Bus. Read-only from the PCI Bus.	Yes	Local	1
5	<i>Reserved.</i>	Yes	No	0
6	<b>User Definable Functions.</b> If set to 1, this device supports user definable functions. Can be written only from the Local Bus. Read-only from the PCI Bus.	Yes	Local	0
7	<b>Fast Back-to-Back Capable.</b> Writing a 1 indicates an adapter can accept fast back-to-back transactions. <i>Note: Hardcoded to 1.</i>	Yes	No	1
8	<b>Master Data Parity Error Detected.</b> Set to 1 when three conditions are met: 1) PCI 9054 asserted PERR# or acknowledged PERR# asserted; 2) PCI 9054 was Bus Master for operation in which error occurred; 3) Parity Error Response bit is set (PCICR[6]=1). Writing a 1 clears this bit to 0.	Yes	Yes/Clr	0
10:9	<b>DEVSEL# Timing.</b> Indicates timing for DEVSEL# assertion. Writing a 01 sets this bit to medium. <i>Note: Hardcoded to 01.</i>	Yes	No	01
11	<b>Target Abort.</b> When set to 1, indicates the PCI 9054 has signaled a Target abort. Writing a 1 clears this bit to 0.	Yes	Yes/Clr	0
12	<b>Received Target Abort.</b> When set to 1, indicates the PCI 9054 has received a Target Abort signal. Writing a 1 clears this bit to 0.	Yes	Yes/Clr	0
13	<b>Received Master Abort.</b> When set to 1, indicates the PCI 9054 has received a Master Abort signal. Writing a 1 clears this bit to 0.	Yes	Yes/Clr	0
14	<b>Signaled System Error.</b> When set to 1, indicates the PCI 9054 has reported a system error on SERR#. Writing a 1 clears this bit to 0.	Yes	Yes/Clr	0
15	<b>Detected Parity Error.</b> When set to 1, indicates the PCI 9054 has detected a PCI Bus parity error, even if parity error handling is disabled (the Parity Error Response bit in the Command register is clear). One of three conditions can cause this bit to be set: 1) PCI 9054 detected parity error during PCI Address phase; 2) PCI 9054 detected data parity error when it was the Target of a write; 3) PCI 9054 detected data parity error when performing Master Read operation. Writing a 1 clears this bit to 0.	Yes	Yes/Clr	0

Register 11-4. (PCIREV; PCI:08h, LOC:08h) PCI Revision ID

Bit	Description	Read	Write	Value after Reset
7:0	<b>Revision ID.</b> Silicon revision of the PCI 9054.	Yes	Local/ Serial EEPROM	Current Rev #

**Register 11-5. (PCICCR; PCI:09-0Bh, LOC:09-0Bh) PCI Class Code**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Register Level Programming Interface.</b> None defined.	Yes	Local/ Serial EEPROM	0h
15:8	<b>Subclass Code (Other Bridge Device).</b>	Yes	Local/ Serial EEPROM	80h
23:16	<b>Base Class Code (Bridge Device).</b>	Yes	Local/ Serial EEPROM	06h

**Register 11-6. (PCICLSR; PCI:0Ch, LOC:0Ch) PCI Cache Line Size**

Bit	Description	Read	Write	Value after Reset
7:0	<b>System Cache Line Size.</b> Specified in units of 32-bit words (8 or 16 Lwords). If a size other than 8 or 16 is specified, the PCI 9054 performs Write transfers rather than Memory Write and Invalidate transfers.	Yes	Yes	0h

**Register 11-7. (PCILTR; PCI:0Dh, LOC:0Dh) PCI Bus Latency Timer**

Bit	Description	Read	Write	Value after Reset
7:0	<b>PCI Bus Latency Timer.</b> Specifies amount of time (in units of PCI Bus clocks) the PCI 9054, as a Bus Master, can burst data on the PCI Bus.	Yes	Yes	0h

**Register 11-8. (PCIHTR; PCI:0Eh, LOC:0Eh) PCI Header Type**

Bit	Description	Read	Write	Value after Reset
6:0	<b>Configuration Layout Type.</b> Specifies layout of bits 10h through 3Fh in configuration space. Only one encoding, 0h, is defined. All other encodings are reserved.	Yes	Local	0h
7	<b>Header Type.</b> Writing a 1 indicates multiple functions. Writing a 0 indicates single function.	Yes	Local	0

**Register 11-9. (PCIBISTR; PCI:0Fh, LOC:0Fh) PCI Built-In Self Test (BIST)**

Bit	Description	Read	Write	Value after Reset
3:0	<b>BIST Pass/Fail.</b> Writing 0h indicates a device passed its test. Non-0h values indicate a device failed its test. Device-specific failure codes can be encoded in a non-0h value.	Yes	Local	0h
5:4	<b>Reserved.</b>	Yes	No	00
6	<b>PCI BIST Interrupt Enable.</b> The PCI Bus writes 1 to enable BIST. Generates an interrupt to the Local Bus. The Local Bus resets this bit when BIST is complete. The software should fail device if BIST is not complete after two seconds. Refer to the Runtime registers for Interrupt Control/Status.	Yes	Yes	0
7	<b>BIST Support.</b> Returns 1 if device supports BIST. Returns 0 if device is not BIST-compatible.	Yes	Local	0

**Register 11-10. (PCIBAR0; PCI:10h, LOC:10h) PCI Base Address Register for Memory Accesses to Local, Runtime, and DMA**

Bit	Description	Read	Write	Value after Reset
0	<b>Memory Space Indicator.</b> Writing a 0 indicates the register maps into Memory space. Writing a 1 indicates the register maps into I/O space. <b>Note:</b> <i>Hardcoded to 0.</i>	Yes	No	0
2:1	<b>Register Location.</b> Values: 00—Locate anywhere in 32-bit Memory Address space 01—Locate below 1-MB Memory Address space 10—Locate anywhere in 64-bit Memory Address space 11—Reserved <b>Note:</b> <i>Hardcoded to 00.</i>	Yes	No	00
3	<b>Prefetchable.</b> Writing a 1 indicates there are no side effects on reads. Does not affect operation of the PCI 9054. <b>Note:</b> <i>Hardcoded to 0.</i>	Yes	No	0
7:4	<b>Memory Base Address.</b> Memory base address for access to Local, Runtime, and DMA registers (requires 256 bytes). <b>Note:</b> <i>Hardcoded to 0h.</i>	Yes	No	0h
31:8	<b>Memory Base Address.</b> Memory base address for access to Local, Runtime, and DMA registers.	Yes	Yes	0h

**Note:** For I<sub>2</sub>O, Inbound message frame pool must reside in address space pointed to by PCIBAR0. Message Frame Address (MFA) is defined by I<sub>2</sub>O as offset from this base address to start of message frame.

**Register 11-11. (PCIBAR1; PCI:14h, LOC:14h) PCI Base Address Register for I/O Accesses to Local, Runtime, and DMA**

Bit	Description	Read	Write	Value after Reset
0	<b>Memory Space Indicator.</b> Writing a 0 indicates the register maps into Memory space. Writing a 1 indicates the register maps into I/O space. <i>Note: Hardcoded to 1.</i>	Yes	No	1
1	<b>Reserved.</b>	Yes	No	0
7:2	<b>I/O Base Address.</b> Base Address for I/O access to Local, Runtime, and DMA registers (requires 256 bytes). <i>Note: Hardcoded to 0h.</i>	Yes	No	0h
31:8	<b>I/O Base Address.</b> Base Address for I/O access to Local, Runtime, and DMA registers. PCIBAR1 can be enabled or disabled by setting or clearing the Base Address Register 1 Enable bit (LMISC[0]).	Yes	Yes	0h

**Register 11-12. (PCIBAR2; PCI:18h, LOC:18h) PCI Base Address Register for Memory Accesses to Local Address Space 0**

Bit	Description	Read	Write	Value after Reset
0	<b>Memory Space Indicator.</b> Writing a 0 indicates the register maps into Memory space. Writing a 1 indicates the register maps into I/O space. (Specified in LASORR register.)	Yes	No	0
2:1	<b>Register Location (If Memory Space).</b> Values: 00—Locate anywhere in 32-bit Memory Address space 01—Locate below 1-MB Memory Address space 10—Locate anywhere in 64-bit Memory Address space 11— <b>Reserved</b> (Specified in LASORR register.) If I/O Space, bit 1 is always 0 and bit 2 is included in the base address.	Yes	Mem: No I/O: bit 1 no, bit 2 yes	00
3	<b>Prefetchable (If Memory Space).</b> Writing a 1 indicates there are no side effects on reads. Reflects value of LASORR[3] and provides only status to the system. Does not affect operation of the PCI 9054. The associated Bus Region Descriptor register controls prefetching functions of this address space. (Specified in LASORR register.) If I/O Space, bit 3 is included in the base address.	Yes	Mem: No I/O: Yes	0
31:4	<b>Memory Base Address.</b> Memory base address for access to Local Address Space 0. PCIBAR2 can be enabled or disabled by setting or clearing the Space 0 Enable bit (LAS0BA[0]).	Yes	Yes	0h

**Register 11-13. (PCIBAR3; PCI:1Ch, LOC:1Ch) PCI Base Address Register for Memory Accesses to Local Address Space 1**

Bit	Description	Read	Write	Value after Reset
0	<b>Memory Space Indicator.</b> Writing a 0 indicates the register maps into Memory space. Writing a 1 indicates the register maps into I/O space. (Specified in LAS1RR register.)	Yes	No	0
2:1	<b>Register Location.</b> Values: 00—Locate anywhere in 32-bit Memory Address space 01—Locate below 1-MB Memory Address space 10—Locate anywhere in 64-bit Memory Address space 11— <b>Reserved</b> (Specified in LAS1RR register.) If I/O Space, bit 1 is always 0 and bit 2 is included in the base address.	Yes	Mem: No I/O: Bit 1 No, Bit 2 Yes	00
3	<b>Prefetchable (If Memory Space).</b> Writing a 1 indicates there are no side effects on reads. Reflects value of LAS1RR[3] and only provides status to the system. Does not affect operation of the PCI 9054. The associated Bus Region Descriptor register controls prefetching functions of this address space. (Specified in LAS1RR register.) If I/O Space, bit 3 is included in base address.	Yes	Mem: No I/O: Yes	0
31:4	<b>Memory Base Address.</b> Memory base address for access to Local Address Space 1. PCIBAR3 can be enabled or disabled by setting or clearing the Space 1 Enable bit (LAS1BA[0]). If QSR[0]=1, PCIBAR3 returns 0h.	Yes	Yes	0h

**Register 11-14. (PCIBAR4; PCI:20h, LOC:20h) PCI Base Address**

Bit	Description	Read	Write	Value after Reset
31:0	<i>Reserved.</i>	Yes	No	0h

**Register 11-15. (PCIBAR5; PCI:24h, LOC:24h) PCI Base Address**

Bit	Description	Read	Write	Value after Reset
31:0	<i>Reserved.</i>	Yes	No	0h

**Register 11-16. (PCICIS; PCI:28h, LOC:28h) PCI Cardbus CIS Pointer**

Bit	Description	Read	Write	Value after Reset
31:0	<b>Cardbus Information Structure Pointer for PCMCIA. <i>Not supported.</i></b>	Yes	No	0h

**Register 11-17. (PCISVID; PCI:2Ch, LOC:2Ch) PCI Subsystem Vendor ID**

Bit	Description	Read	Write	Value after Reset
15:0	<b>Subsystem Vendor ID (unique add-in board Vendor ID).</b>	Yes	Local/ Serial EEPROM	10B5h

**Register 11-18. (PCISID; PCI:2Eh, LOC:2Eh) PCI Subsystem ID**

Bit	Description	Read	Write	Value after Reset
15:0	Subsystem ID (unique add-in board Device ID).	Yes	Local/ Serial EEPROM	9054h

**Register 11-19. (PCIERBAR; PCI:30h, LOC:30h) PCI Expansion ROM Base**

Bit	Description	Read	Write	Value after Reset
0	<b>Address Decode Enable.</b> Writing a 1 indicates a device accepts accesses to the Expansion ROM address. Writing a 0 indicates a device does not accept accesses to Expansion ROM space. Should be set to 0 if there is no Expansion ROM. Works in conjunction with EROMRR[0].	Yes	Yes	0
10:1	<b>Reserved.</b>	Yes	No	0h
31:11	<b>Expansion ROM Base Address (upper 21 bits).</b>	Yes	Yes	0h

**Register 11-20. (CAP\_PTR; PCI:34h, LOC:34h) New Capability Pointer**

Bit	Description	Read	Write	Value after Reset
7:0	<b>New Capability Pointer.</b> Offset into PCI Configuration Space for the location of the first item in the New Capabilities Linked List.	Yes	Local	40h
31:8	<b>Reserved.</b>	Yes	No	0h

**Register 11-21. (PCIILR; PCI:3Ch, LOC:3Ch) PCI Interrupt Line**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Interrupt Line Routing Value.</b> Value indicates which input of the system interrupt controller(s) is connected to each interrupt line of the device.	Yes	Yes	0h

**Register 11-22. (PCIIPR; PCI:3Dh, LOC:3Dh) PCI Interrupt Pin**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Interrupt Pin Register.</b> Indicates which interrupt pin the device uses. The following values are decoded (the PCI 9054 supports only INTA#): 0 = No Interrupt pin 1 = INTA# 2 = INTB# 3 = INTC# 4 = INTD#	Yes	Local/ Serial/ EEPROM	1h

**Register 11-23. (PCIMGR; PCI:3Eh, LOC:3Eh) PCI Min\_Gnt**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Min_Gnt.</b> Specifies how long a burst period device needs, assuming a clock rate of 33 MHz. Value is a multiple of 1/4 $\mu$ s increments.	Yes	Local/ Serial/ EEPROM	0h

**Register 11-24. (PCIMLR; PCI:3Fh, LOC:3Fh) PCI Max\_Lat**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Max_Lat.</b> Specifies how often the device must gain access to the PCI Bus. Value is a multiple of 1/4 $\mu$ s increments.	Yes	Local/ Serial/ EEPROM	0h

**Register 11-25. (PMCAPID; PCI:40h, LOC:180h) Power Management Capability ID**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Power Management Capability ID.</b>	Yes	No	1h

**Register 11-26. (PMNEXT; PCI:41h, LOC:181h) Power Management Next Capability Pointer**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Next_Cap Pointer.</b> Points to the first location of the next item in the capabilities linked list. If power management is the last item in the list, then this register should be set to 0.	Yes	Local	48h

**Register 11-27. (PMC; PCI:42h, LOC:182h) Power Management Capabilities**

Bit	Description	Read	Write	Value after Reset										
2:0	<b>Version.</b> Writing a 1 indicates this function complies with <i>PCI Power Management Interface Specification v1.0</i> .	Yes	Local	001										
3	<b>PCI Clock Required for PME# Signal.</b> When set to 1, indicates a function relies on the presence of the PCI clock for PME# operation. The PCI 9054 does not require the PCI clock for PME#, so this bit should be set to 0.	Yes	Local	0										
4	<b>Auxiliary Power Source.</b> Because the PCI 9054 does not support PME# while in a D3 <sub>COLD</sub> state, this bit is always set to 0.	Yes	No	0										
5	<b>DSI.</b> When set to 1, the PCI 9054 requires special initialization following a transition to a D <sub>0</sub> uninitialized state before a generic class device driver is able to use it.	Yes	Local	0										
8:6	<b>Reserved.</b>	Yes	No	000										
9	<b>D<sub>1</sub> Support.</b> When set to 1, the PCI 9054 supports the D <sub>1</sub> power state.	Yes	Local	0										
10	<b>D<sub>2</sub> Support.</b> When set to 1, the PCI 9054 supports the D <sub>2</sub> power state.	Yes	Local	0										
14:11	<b>PME# Support.</b> Indicates power states in which the PCI 9054 may assert PME#. <table border="1"> <thead> <tr> <th>Value</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>XXX1</td> <td>PME# can be asserted from D<sub>0</sub></td> </tr> <tr> <td>XX1X</td> <td>PME# can be asserted from D<sub>1</sub></td> </tr> <tr> <td>X1XX</td> <td>PME# can be asserted from D<sub>2</sub></td> </tr> <tr> <td>1XXX</td> <td>PME# can be asserted from D<sub>3hot</sub></td> </tr> </tbody> </table>	Value	Description	XXX1	PME# can be asserted from D <sub>0</sub>	XX1X	PME# can be asserted from D <sub>1</sub>	X1XX	PME# can be asserted from D <sub>2</sub>	1XXX	PME# can be asserted from D <sub>3hot</sub>	Yes	Local	0h
Value	Description													
XXX1	PME# can be asserted from D <sub>0</sub>													
XX1X	PME# can be asserted from D <sub>1</sub>													
X1XX	PME# can be asserted from D <sub>2</sub>													
1XXX	PME# can be asserted from D <sub>3hot</sub>													
15	<b>Reserved.</b>	Yes	No	0										

Register 11-28. (PMCSR; PCI:44h, LOC:184h) Power Management Control/Status

Bit	Description	Read	Write	Value after Reset										
1:0	<p><b>Power State.</b> Determines or changes the current power state.</p> <table border="0"> <tr> <td><u>Value</u></td> <td><u>State</u></td> </tr> <tr> <td>00</td> <td>D<sub>0</sub></td> </tr> <tr> <td>01</td> <td>D<sub>1</sub></td> </tr> <tr> <td>10</td> <td>D<sub>2</sub></td> </tr> <tr> <td>11</td> <td>D<sub>3hot</sub></td> </tr> </table> <p>Transition from a D<sub>3hot</sub> state to a D<sub>0</sub> state causes a soft reset. Should only be initiated from the PCI Bus because the Local Bus interface is reset during a soft reset. In a D<sub>3hot</sub> state, PCI Memory and I/O accesses are disabled, as well as PCI interrupts, and only configuration is allowed. The same is true for the D<sub>2</sub> state if the corresponding D<sub>2_Support</sub> pin is set.</p>	<u>Value</u>	<u>State</u>	00	D <sub>0</sub>	01	D <sub>1</sub>	10	D <sub>2</sub>	11	D <sub>3hot</sub>	Yes	Yes	00
<u>Value</u>	<u>State</u>													
00	D <sub>0</sub>													
01	D <sub>1</sub>													
10	D <sub>2</sub>													
11	D <sub>3hot</sub>													
7:2	<b>Reserved.</b>	Yes	No	0h										
8	<b>PME#_En.</b> Writing a 1 enables PME# to be asserted.	Yes	Yes	0										
12:9	<b>Data_Select.</b> Selects which data to report through the Data register and Data_Scale bits.	Yes	Yes	0h										
14:13	<p><b>Data_Scale.</b> Indicates the scaling factor to use when interpreting the value of the Data register. Value and meaning of this bit depends on the data value selected by the Data_Select bit. When the Local CPU initializes the Data_Scale values, must use the Data_Select bit to determine which Data_Scale value it is writing.</p> <p>For Power Consumed and Power Dissipated data, the following scale factors are used. Unit values are in watts.</p> <table border="0"> <tr> <td><u>Value</u></td> <td><u>Scale</u></td> </tr> <tr> <td>0</td> <td>Unknown</td> </tr> <tr> <td>1</td> <td>0.1x</td> </tr> <tr> <td>2</td> <td>0.01x</td> </tr> <tr> <td>3</td> <td>0.001x</td> </tr> </table>	<u>Value</u>	<u>Scale</u>	0	Unknown	1	0.1x	2	0.01x	3	0.001x	Yes	Local	00
<u>Value</u>	<u>Scale</u>													
0	Unknown													
1	0.1x													
2	0.01x													
3	0.001x													
15	<b>PME#_Status.</b> Indicates PME# is being driven if the PME#_En bit is set (PMCSR[8]=1). Writing a 1 from the Local Bus sets this bit; writing a 1 from the PCI Bus clears this bit to 0. Depending on the current power state, set only if the appropriate PME#_Support bit(s) is set (PMC[15:11]=1).	Yes	Local/ Set, PCI/Clr	0										

Register 11-29. (PMCSR\_BSE; PCI:46h, LOC:186h) PMCSR Bridge Support Extensions

Bit	Description	Read	Write	Value after Reset
7:0	<b>Reserved.</b>	Yes	No	0h

**Register 11-30. (PMDATA; PCI:47h, LOC:187h) Power Management Data**

Bit	Description	Read	Write	Value after Reset																		
7:0	<p><b>Power Management Data.</b> Provides operating data, such as power consumed or heat dissipation. Data returned is selected by the Data_Select bit(s) (PMCSR[12:9]) and scaled by the Data_Scale bit(s) (PMCSR[14:13]).</p> <table border="1"> <thead> <tr> <th>Data_Select</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>D<sub>0</sub> Power Consumed</td> </tr> <tr> <td>1</td> <td>D<sub>1</sub> Power Consumed</td> </tr> <tr> <td>2</td> <td>D<sub>2</sub> Power Consumed</td> </tr> <tr> <td>3</td> <td>D<sub>3</sub> Power Consumed</td> </tr> <tr> <td>4</td> <td>D<sub>0</sub> Power Dissipated</td> </tr> <tr> <td>5</td> <td>D<sub>1</sub> Power Dissipated</td> </tr> <tr> <td>6</td> <td>D<sub>2</sub> Power Dissipated</td> </tr> <tr> <td>7</td> <td>D<sub>3hot</sub> Power Dissipated</td> </tr> </tbody> </table>	Data_Select	Description	0	D <sub>0</sub> Power Consumed	1	D <sub>1</sub> Power Consumed	2	D <sub>2</sub> Power Consumed	3	D <sub>3</sub> Power Consumed	4	D <sub>0</sub> Power Dissipated	5	D <sub>1</sub> Power Dissipated	6	D <sub>2</sub> Power Dissipated	7	D <sub>3hot</sub> Power Dissipated	Yes	Local	0h
Data_Select	Description																					
0	D <sub>0</sub> Power Consumed																					
1	D <sub>1</sub> Power Consumed																					
2	D <sub>2</sub> Power Consumed																					
3	D <sub>3</sub> Power Consumed																					
4	D <sub>0</sub> Power Dissipated																					
5	D <sub>1</sub> Power Dissipated																					
6	D <sub>2</sub> Power Dissipated																					
7	D <sub>3hot</sub> Power Dissipated																					

**Register 11-31. (HS\_CNTL; PCI:48h, LOC:188h) Hot Swap Control**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Hot Swap ID.</b>	Yes	Local/ Serial EEPROM	06h

**Register 11-32. (HS\_NEXT; PCI:49h, LOC:189h) Hot Swap Next Capability Pointer**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Next_Cap Pointer.</b> Points to the first location of the next item in the capabilities linked list. If Hot Swap is the last item in the list, then this register should be set to 0.	Yes	Local/ Serial EEPROM	4Ch

**Register 11-33. (HS\_CSR; PCI:4Ah, LOC:18Ah) Hot Swap Control/Status**

Bit	Description	Read	Write	Value after Reset
0	<i>Reserved.</i>	Yes	No	0
1	<b>ENUM# Interrupt Clear.</b> Writing a 0 enables the interrupt. Writing a 1 clears the interrupt.	Yes	Yes/Clr	0
2	<i>Reserved.</i>	Yes	No	0
3	<b>LED Software On/Off Switch.</b> Writing a 1 turns on the LED. Writing a 0 turns off the LED.	Yes	PCI	0
4	<i>Reserved.</i>	Yes	No	0
5	<i>Reserved.</i>	Yes	No	0
6	<b>Board Removal ENUM# Status Indicator.</b> Writing a 1 reports the ENUM# assertion for removal process.	Yes	Yes	0
7	<b>Board Insertion ENUM# Status Indicator.</b> Writing a 1 reports the ENUM# assertion for insertion process.	Yes	Yes	1
15:8	<i>Reserved.</i>	Yes	No	0h

**Register 11-34. (PVPDCNTL; PCI:4Ch, LOC:18Ch) PCI Vital Product Data Control**

Bit	Description	Read	Write	Value after Reset
7:0	<b>VPD ID.</b> Capability ID = 03h for VPD.	PCI	No	03h

**Register 11-35. (PVPD\_NEXT; PCI:4Dh, LOC:18Dh) PCI Vital Product Data Next Capability Pointer**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Next_Cap Pointer.</b> Points to first location of next item in the capabilities linked list. VPD is the last item in the capabilities linked list. This register is set to 0h.	PCI	Local	0h

**Register 11-36. (PVPDAD; PCI:4Eh, LOC:18Eh) PCI Vital Product Data Address**

Bit	Description	Read	Write	Value after Reset
14:0	<b>VPD Address.</b> Byte address of the VPD address to be accessed. Supports a 2K or 4K bit serial EEPROM.	PCI	Yes	0h
15	<b>F.</b> Flag used to indicate when the transfer of data between PVPDATA and the storage component is complete. Writing a 0 along with the VPD address causes a read of VPD information into PVPDATA. The hardware sets this bit to 1 when the VPD Data transfer is complete. Writing a 1 along with the VPD address causes a write of VPD information from PVPDATA into a storage component. The hardware sets this bit to 0 after the Write operation is complete.	PCI	Yes	0

**Register 11-37. (PVPDATA; PCI:50h, LOC:190h) PCI VPD Data**

Bit	Description	Read	Write	Value after Reset
31:0	<b>VPD Data Register.</b>	PCI	Yes	0h

## 11.4 LOCAL CONFIGURATION REGISTERS

Register 11-38. (LAS0RR; PCI:00h, LOC:80h) Local Address Space 0 Range Register for PCI-to-Local Bus

Bit	Description	Read	Write	Value after Reset										
0	<b>Memory Space Indicator.</b> Writing a 0 indicates Local Address Space 0 maps into PCI Memory space. Writing a 1 indicates Local Address Space 0 maps into PCI I/O space.	Yes	Yes	0										
2:1	When mapped into Memory space, encoding is as follows: <table border="0"> <tr> <td>2/1</td> <td>Meaning</td> </tr> <tr> <td>0 0</td> <td>Locate anywhere in 32-bit PCI Address space</td> </tr> <tr> <td>0 1</td> <td>Locate below 1 MB in PCI Address space</td> </tr> <tr> <td>1 0</td> <td>Locate anywhere in 64-bit PCI Address space</td> </tr> <tr> <td>1 1</td> <td><b>Reserved</b></td> </tr> </table> When mapped into I/O space, bit 1 must be set to 0. Bit 2 is included with bits [31:3] to indicate the decoding range.	2/1	Meaning	0 0	Locate anywhere in 32-bit PCI Address space	0 1	Locate below 1 MB in PCI Address space	1 0	Locate anywhere in 64-bit PCI Address space	1 1	<b>Reserved</b>	Yes	Yes	00
2/1	Meaning													
0 0	Locate anywhere in 32-bit PCI Address space													
0 1	Locate below 1 MB in PCI Address space													
1 0	Locate anywhere in 64-bit PCI Address space													
1 1	<b>Reserved</b>													
3	When mapped into Memory space, writing a 1 indicates reads are prefetchable (does not affect operation of the PCI 9054, but is used for system status). When mapped into I/O space, it is included with bits [31:2] to indicate the decoding range.	Yes	Yes	0										
31:4	Specifies which PCI Address bits to use for decoding a PCI access to Local Bus Space 0. Each bit corresponds to a PCI Address bit. Bit 31 corresponds to address bit 31. Write 1 to all bits that must be included in decode and 0 to all others (used in conjunction with PCIBAR2). Default is 1 MB.  <b>Notes:</b> Range ( <i>not</i> Range register) must be power of 2. "Range register value" is inverse of range.  User should limit all I/O spaces to 256 bytes per PCI v2.1 spec.	Yes	Yes	FFF0000h										

Register 11-39. (LAS0BA; PCI:04h, LOC:84h) Local Address Space 0 Local Base Address (Remap)

Bit	Description	Read	Write	Value after Reset
0	<b>Space 0 Enable.</b> Writing a 1 enables decoding of PCI addresses for PCI Target access to Local Bus Space 0. Writing a 0 disables decoding.	Yes	Yes	0
1	<b>Reserved.</b>	Yes	No	0
3:2	If Local Bus Space 0 is mapped into Memory space, bits are not used. When mapped into I/O space, included with bits [31:4] for remapping.	Yes	Yes	00
31:4	<b>Remap PCI Address to Local Address Space 0 into Local Address Space.</b> Bits in this register remap (replace) PCI Address bits used in decode as Local Address bits.  <b>Note:</b> Remap Address value must be a multiple of the Range ( <i>not</i> the Range register).	Yes	Yes	0h

Register 11-40. (MARBR; PCI:08h or ACh, LOC:88h or 12Ch) Mode/DMA Arbitration

Bit	Description	Read	Write	Value after Reset
7:0	<b>Local Bus Latency Timer.</b> Number of Local Bus clock cycles to occur before de-asserting HOLD and releasing the Local Bus.	Yes	Yes	0h
15:8	<b>Local Bus Pause Timer.</b> Number of Local Bus Clock cycles to occur before reasserting HOLD after releasing the Local Bus. The pause timer is valid only during DMA.	Yes	Yes	0h
16	<b>Local Bus Latency Timer Enable.</b> Writing a 1 enables the latency timer. Writing a 0 disables the latency timer.	Yes	Yes	0
17	<b>Local Bus Pause Timer Enable.</b> Writing a 1 enables the pause timer. Writing a 0 disables the pause timer.	Yes	Yes	0
18	<b>Local Bus BREQ Enable.</b> Writing a 1 enables the Local Bus BR#/BREQi. When BR#/BREQi is active, the PCI 9054 de-asserts HOLD and releases the Local Bus.	Yes	Yes	0
20:19	<b>DMA Channel Priority.</b> Writing a 00 indicates a rotational priority scheme. Writing a 01 indicates Channel 0 has priority. Writing a 10 indicates Channel 1 has priority. Writing an 11 indicates <i>reserved</i> .	Yes	Yes	00
21	<b>Local Bus PCI Target Release Bus Mode.</b> When set to 1, the PCI 9054 de-asserts HOLD and releases the Local Bus when the PCI Target Write FIFO becomes empty during a PCI Target Write or when the PCI Target Read FIFO becomes full during a PCI Target Read.	Yes	Yes	1
22	<b>PCI Target LOCK# Enable.</b> Writing a 1 enables PCI Target locked sequences. Writing a 0 disables PCI Target locked sequences.	Yes	Yes	0
23	<b>PCI Request Mode.</b> Writing a 1 causes the PCI 9054 to de-assert REQ# when it asserts FRAME during a Master cycle. Writing a 0 causes the PCI 9054 to leave REQ# asserted for the entire Bus Master cycle.	Yes	Yes	0
24	<b>Delayed Read Mode.</b> When set to 1, the PCI 9054 operates in Delayed Transaction mode for PCI Target reads. The PCI 9054 issues a Retry to the PCI Host and prefetches Read data.	Yes	Yes	0
25	<b>PCI Read No Write Mode.</b> Writing a 1 forces a Retry on writes if a read is pending. Writing a 0 allows writes to occur while a read is pending.	Yes	Yes	0
26	<b>PCI Read with Write Flush Mode.</b> Writing a 1 submits a request to flush a pending Read cycle if a Write cycle is detected. Writing a 0 submits a request to not effect pending reads when a Write cycle occurs ( <i>PCI Specification v2.1 compatible</i> ).	Yes	Yes	0
27	<b>Gate Local Bus Latency Timer with BREQi. (C and J modes only.)</b>	Yes	Yes	0
28	<b>PCI Read No Flush Mode.</b> Writing a 1 submits a request to not flush the Read FIFO if the PCI Read cycle completes (Read Ahead mode). Writing a 0 submits a request to flush the Read FIFO if a PCI Read cycle completes.	Yes	Yes	0
29	When set to 0, reads from the PCI Configuration Register address 00h returns Device ID and Vendor ID. When set to 1, reads from the PCI Configuration register address 00h returns Subsystem ID and Subsystem Vendor ID.	Yes	Yes	0
30	<b>FIFO Full Status Flag.</b> When set to 1, the PCI Initiator Write FIFO is almost full. Reflects the value of the DMPAF pin.	Yes	No	0
31	<b>BIGEND#/WAIT# Input/Output Select (M mode only).</b> Writing a 1 selects the wait functionality of the signal. Writing a 0 selects Big Endian input functionality.	Yes	Yes	0

## Register 11-41. (BIGEND; PCI:0Ch, LOC:8Ch) Big/Little Endian Descriptor

Bit	Description	Read	Write	Value after Reset
0	<b>Configuration Register Big Endian Mode (Address Invariance).</b> Writing a 1 specifies use of Big Endian data ordering for Local accesses to the Configuration registers. Writing a 0 specifies Little Endian ordering. Big Endian mode can be specified for Configuration register accesses by asserting BIGEND# during the Address phase of the access.	Yes	Yes	0
1	<b>PCI Initiator Big Endian Mode (Address Invariance).</b> Writing a 1 specifies use of Big Endian data ordering for PCI Initiator accesses. Writing a 0 specifies Little Endian ordering. Big Endian mode can be specified for PCI Initiator accesses by asserting BIGEND# input pin during the Address phase of the access.	Yes	Yes	0
2	<b>PCI Target Address Space 0 Big Endian Mode (Address Invariance).</b> Writing a 1 specifies use of Big Endian data ordering for PCI Target accesses to Local Address Space 0. Writing a 0 specifies Little Endian ordering.	Yes	Yes	0
3	<b>PCI Target Address Expansion ROM 0 Big Endian Mode (Address Invariance).</b> Writing a 1 specifies use of Big Endian data ordering for PCI Target accesses to Expansion ROM. Writing a 0 specifies Little Endian ordering.	Yes	Yes	0
4	<b>Big Endian Byte Lane Mode.</b> Writing a 1 specifies that in any Endian mode, use the following byte lanes for the modes listed: <b>M Mode</b> [0:15] for a 16-bit Local Bus [0:7] for an 8-bit Local Bus <b>C and J Modes</b> [31:16] for a 16-bit Local Bus [31:24] for an 8-bit Local Bus Writing a 0 specifies that in any Endian mode, use the following byte lanes for the modes listed: <b>M Mode</b> [16:31] for a 16-bit Local Bus [24:31] for an 8-bit Local Bus <b>C and J Modes</b> [15:0] for a 16-bit Local Bus [7:0] for an 8-bit Local Bus	Yes	Yes	0
5	<b>PCI Target Address Space 1 Big Endian Mode (Address Invariance).</b> Writing a 1 specifies use of Big Endian data ordering for PCI Target accesses to Local Address Space 1. Writing a 0 specifies Little Endian ordering.	Yes	Yes	0
6	<b>DMA Channel 1 Big Endian Mode (Address Invariance).</b> Writing a 1 specifies use of Big Endian data ordering for DMA Channel 1 accesses to the Local Address space. Writing a 0 specifies Little Endian ordering.	Yes	Yes	0
7	<b>DMA Channel 0 Big Endian Mode (Address Invariance).</b> Writing a 1 specifies use of Big Endian data ordering for DMA Channel 0 accesses to the Local Address space. Writing a 0 specifies Little Endian ordering.	Yes	Yes	0

**Register 11-42. (LMISC; PCI:0Dh, LOC:8Dh) Local Miscellaneous Control**

Bit	Description	Read	Write	Value after Reset
0	<b>Base Address Register 1 Enable.</b> If set to 1, the Base Address 1 Register for I/O accesses to Configuration registers is enabled. If set to 0, the Base Address 1 Register for I/O accesses to Configuration registers is disabled.	Yes	Yes	1
1	<b>Base Address Register 1 Shift.</b> If Base Address Register 1 Enable is low, and this bit is set to 0, then PCIBAR2 and PCIBAR3 remain at PCI Configuration addresses 18h and 1Ch. If Base Address Register 1 Enable is low, and this bit is set to 1, then PCIBAR2 (Local Address Space 0) and PCIBAR3 (Local Address Space 1) are shifted to become PCIBAR1 and PCIBAR2 at PCI Configuration addresses 14h and 18h. Set if a blank region in Base Address Register Space could not be accepted by system BIOS.	Yes	Yes	0
2	<b>Local Init Status.</b> Writing a 1 indicates Local Init done. Responses to PCI accesses are Retrys until this bit is set. If the PCI 9054 has a blank serial EEPROM attached, the Local processor must set the Local Init Status bit to 1.	Yes	Local/ Serial EEPROM	0
3	<b>Reserved.</b>	Yes	No	0
4	<b>M Mode PCI Initiator Deferred Read Enable.</b> Writing a 1 enables the PCI 9054 to operate in Delayed Transaction mode for PCI Initiator reads. The PCI 9054 issues a RETRY# to the M mode Master and prefetches Read data from the PCI Bus.	Yes	Yes	0
5	<b>M Mode TEA# Input Interrupt Mask.</b> When set to 1, TEA# input causes SERR# output on the PCI Bus if enabled (PCICR[8]=1) and the Signaled System Error bit is set (PCISR[14]=1). Writing 0 masks the TEA# input to create SERR#. The SERR# Status bit is set in both cases.	Yes	Yes	0
6	<b>PCI Initiator Write FIFO Almost Full RETRY# Output Enable.</b> When set to 1, the PCI 9054 issues a RETRY# to the MPC850 or MPC860.	Yes	Yes	0
7	<b>Reserved.</b>	Yes	No	0

**Register 11-43. (PROT\_AREA; PCI:0Eh, LOC:8Eh) Serial EEPROM Write-Protected Address Boundary**

Bit	Description	Read	Write	Value after Reset
6:0	<b>Serial EEPROM Starting at Lword Boundary (48 Lwords = 192 bytes) for VPD Accesses.</b> Any serial EEPROM address below this boundary is read-only. <i>Note: Anything below the programmed address may contain the PCI 9054 Configuration data.</i>	Yes	Yes	0110000
15:7	<b>Reserved.</b>	Yes	No	0h

Register 11-44. (EROMRR; PCI:10h, LOC:90h) Expansion ROM Range

Bit	Description	Read	Write	Value after Reset
0	<b>Address Decode Enable.</b> Bit 0 can only be enabled from the serial EEPROM. To disable, set the PCI Expansion ROM Address Decode Enable bit to 0 (PCIERBAR[0]=0).	Yes	Serial EEPROM Only	0
10:1	<b>Reserved.</b>	Yes	No	0h
31:11	Specifies which PCI Address bits to use for decoding a PCI-to-Local Bus Expansion ROM. Each bit corresponds to a PCI Address bit. Bit 31 corresponds to address bit 31. Write 1 to all bits that must be included in decode and 0 to all others (used in conjunction with PCIERBAR). Default is 64 KB. <b>Note:</b> Range ( <i>not</i> Range register) must be power of 2. "Range register value" is inverse of range.	Yes	Yes	FFFF00h

Register 11-45. (EROMBA; PCI:14h, LOC:94h) Expansion ROM Local Base Address (Remap) and BREQo Control

Bit	Description	Read	Write	Value after Reset
3:0	<b>M Mode: RETRY# Signal Assertion Delay Clocks.</b> Number of Local Bus clocks in which a PCI Target BR# request is pending and a Local PCI Initiator access is in progress and not being granted the bus BG# before asserting RETRY#. Once asserted, RETRY# remains asserted until PCI 9054 samples de-assertion of BB# by the Local Arbiter (Least Significant Bit is 8 or 64 clocks). <b>C and J Modes: Backoff Request Delay Clocks.</b> Number of Local Bus clocks in which a PCI Target HOLD request is pending and a Local PCI Initiator access is in progress and not being granted the bus (LHOLDA) before asserting BREQo (Backoff Request Out). BREQo remains asserted until the PCI 9054 receives LHOLDA (Least Significant Bit is 8 or 64 clocks).	Yes	Yes	0h
4	<b>Local Bus Backoff Enable (C, J, and M Modes).</b> Writing a 1 enables the PCI 9054 to assert BREQo/RETRY#.	Yes	Yes	0
5	<b>Backoff Timer Resolution.</b> Writing a 1 changes the Least Significant Bit of the Backoff Timer from 8 to 64 clocks.	Yes	Yes	0
10:6	<b>Reserved.</b>	Yes	No	0h
31:11	<b>Remap PCI Expansion ROM Space into Local Address Space.</b> Bits in this register remap (replace) the PCI Address bits used in decode as Local Address bits. <b>Note:</b> Remap Address value must be a multiple of the Range ( <i>not</i> the Range register).	Yes	Yes	0h

Register 11-46. (LBRD0; PCI:18h, LOC:98h) Local Address Space 0/Expansion ROM Bus Region Descriptor

Bit	Description	Read	Write	Value after Reset
1:0	<b>Memory Space 0 Local Bus Width.</b> Writing a 00 indicates an 8-bit bus width. Writing a 01 indicates a 16-bit bus width. Writing a 10 or 11 indicates a 32-bit bus width.	Yes	Yes	M = 11 J = 11 C = 11
5:2	<b>Memory Space 0 Internal Wait States (data-to-data; 0-15 wait states).</b>	Yes	Yes	0h
6	<b>Memory Space 0 TA#/READY# Input Enable.</b> Writing a 1 enables TA#/READY# input. Writing a 0 disables TA#/READY# input.	Yes	Yes	1
7	<b>Memory Space 0 BTERM# Input Enable.</b> Writing a 1 enables BTERM# input. Writing a 0 disables BTERM# input. For more information, refer to Section 2.2.5 for M mode or Section 4.2.5 for C and J modes.	Yes	Yes	0
8	<b>Memory Space 0 Prefetch Disable.</b> When mapped into Memory space, writing a 0 enables Read prefetching. Writing a 1 disables prefetching. If prefetching is disabled, the PCI 9054 disconnects after each Memory read.	Yes	Yes	0
9	<b>Expansion ROM Space Prefetch Disable.</b> Writing a 0 enables Read prefetching. Writing a 1 disables prefetching. If prefetching is disabled, the PCI 9054 disconnects after each Memory read.	Yes	Yes	0
10	<b>Prefetch Counter Enable.</b> When set to 1 and Memory prefetching is enabled, the PCI 9054 prefetches up to the number of Lwords specified in prefetch count. When set to 0, the PCI 9054 ignores the count and continues prefetching until it is terminated by the PCI Bus.	Yes	Yes	0
14:11	<b>Prefetch Counter.</b> Number of Lwords to prefetch during Memory Read cycles (0-15). A count of zero selects a prefetch of 16 Lwords.	Yes	Yes	0h
15	<b>Reserved.</b>	Yes	No	0
17:16	<b>Expansion ROM Space Local Bus Width.</b> Writing a 00 indicates an 8-bit bus width. Writing a 01 indicates a 16-bit bus width. Writing a 10 or 11 indicates a 32-bit bus width.	Yes	Yes	M = 11 J = 11 C = 11
21:18	<b>Expansion ROM Space Internal Wait States (data-to-data; 0-15 wait states).</b>	Yes	Yes	0h
22	<b>Expansion ROM Space TA#/READY# Input Enable.</b> Writing a 1 enables TA#/READY# input. Writing a 0 disables TA#/READY# input.	Yes	Yes	1
23	<b>Expansion ROM Space BTERM# Input Enable.</b> Writing a 1 enables BTERM# input. Writing a 0 disables BTERM# input. For more information, refer to Section 2.2.5 for M mode or to Section 4.2.5 for C and J modes.	Yes	Yes	0
24	<b>Memory Space 0 Burst Enable.</b> Writing a 1 enables bursting. Writing a 0 disables bursting.	Yes	Yes	0
25	<b>Extra Long Load from Serial EEPROM.</b> Writing a 1 loads the Subsystem ID and Local Address Space 1 registers. Writing a 0 indicates not to load them.	Yes	Serial EEPROM Only	0
26	<b>Expansion ROM Space Burst Enable.</b> Writing a 1 enables bursting. Writing a 0 disables bursting.	Yes	Yes	0
27	<b>PCI Target PCI Write Mode.</b> Writing a 0 indicates the PCI 9054 should disconnect when the PCI Target Write FIFO is full. Writing a 1 indicates the PCI 9054 should de-assert TRDY# when the PCI Target Write FIFO is full.	Yes	Yes	0
31:28	<b>PCI Target Retry Delay Clocks.</b> Contains the value (multiplied by 8) of the number of PCI Bus clocks after receiving a PCI-to-Local Read or Write access and not successfully completing a transfer. Pertains only to PCI Target writes when the PCI Target PCI Write Mode bit is set (LBRD0[27]=1).	Yes	Yes	4h (32 clocks)

**Register 11-47. (DMRR; PCI:1Ch, LOC:9Ch) Local Range Register for PCI Initiator-to-PCI**

Bit	Description	Read	Write	Value after Reset
15:0	<b>Reserved</b> (64-KB increments).	Yes	No	0h
31:16	Specifies which Local Address bits to use for decoding a Local-to-PCI Bus access. Each bit corresponds to a PCI Address bit. Bit 31 corresponds to address bit 31. Write 1 to all bits that must be included in decode and 0h to all others. <b>Note:</b> Range ( <b>not</b> Range register) must be power of 2. "Range register value" is inverse of range.	Yes	Yes	0h

**Register 11-48. (DMLBAM; PCI:20h, LOC:A0h) Local Bus Base Address Register for PCI Initiator-to-PCI Memory**

Bit	Description	Read	Write	Value after Reset
15:0	<b>Reserved.</b>	Yes	No	0h
31:16	Assigns a value to bits to use for decoding Local-to-PCI Memory accesses. <b>Note:</b> Local Base Address value must be a multiple of the Range ( <b>not</b> the Range register).	Yes	Yes	0h

**Register 11-49. (DMLBAI; PCI:24h, LOC:A4h) Local Base Address Register for PCI Initiator-to-PCI I/O Configuration**

Bit	Description	Read	Write	Value after Reset
15:0	<b>Reserved.</b>	Yes	No	0h
31:16	Assigns a value to bits to use for decoding Local-to-PCI I/O or Configuration accesses. <b>Notes:</b> Local Base Address value must be a multiple of the Range ( <b>not</b> the Range register). Refer to DMPBAM[13] for the I/O Remap Address option.	Yes	Yes	0h

**Register 11-50. (DMPBAM; PCI:28h, LOC:A8h) PCI Base Address (Remap) Register for PCI Initiator-to-PCI Memory**

Bit	Description	Read	Write	Value after Reset
0	<b>PCI Initiator Memory Access Enable.</b> Writing a 1 enables decode of PCI Initiator Memory accesses. Writing a 0 disables decode of PCI Initiator Memory accesses.	Yes	Yes	0
1	<b>PCI Initiator I/O Access Enable.</b> Writing a 1 enables decode of PCI Initiator I/O accesses. Writing a 0 disables decode of PCI Initiator I/O accesses.	Yes	Yes	0
2	<b>PCI Initiator Cache Enable.</b> Writing a 1 causes prefetch to occur infinitely.	Yes	Yes	0
12, 3	<b>PCI Initiator Read Prefetch Size Control.</b> Values: 00 = PCI 9054 continues to prefetch Read data from the PCI Bus until the PCI Initiator access is finished. This may result in an additional four unneeded Lwords being prefetched from the PCI Bus. 01 = Prefetch up to four Lwords from the PCI Bus. 10 = Prefetch up to eight Lwords from the PCI Bus. 11 = Prefetch up to 16 Lwords from the PCI Bus. PCI Initiator Burst reads should not exceed programmed limit.	Yes	Yes	00
4	<b>PCI Initiator PCI Read Mode.</b> Writing a 0 indicates the PCI 9054 should release the PCI Bus when the Read FIFO becomes full. Writing a 1 indicates the PCI 9054 should keep the PCI Bus and de-assert IRDY# when the Read FIFO becomes full.	Yes	Yes	0
10, 8:5	<b>Programmable Almost Full Flag.</b> When the number of entries in the 32-word PCI Initiator Write FIFO exceeds this value, the MDREQ#/DMPAF signal is asserted high.	Yes	Yes	00000
9	<b>Memory Write and Invalidate Mode.</b> When set to 1, the PCI 9054 waits for 8 or 16 Lwords to be written from the Local Bus before starting a PCI access. In addition, all Memory Write and Invalidate cycles to the PCI Bus must be 8 or 16 Lword bursts.	Yes	Yes	0
11	<b>PCI Initiator Prefetch Limit.</b> Writing a 1 causes the PCI 9054 to not prefetch past 4-KB boundaries.	Yes	Yes	0
13	<b>I/O Remap Select.</b> Writing a 1 forces PCI Address bits [31:16] to all zeros. Writing a 0 uses bits [31:16] of this register as PCI Address bits [31:16].	Yes	Yes	0
15:14	<b>PCI Initiator Write Delay.</b> Delays PCI Bus request after PCI Initiator Burst Write cycle has started. Values: 00 = No delay; start cycle immediately 01 = Delay 4 PCI clocks 10 = Delay 8 PCI clocks 11 = Delay 16 PCI clocks	Yes	Yes	00
31:16	<b>Remap Local-to-PCI Space into PCI Address Space.</b> Bits in this register remap (replace) Local Address bits used in decode as the PCI Address bits. <b>Note:</b> Remap Address value must be a multiple of the Range ( <b>not</b> the Range register).	Yes	Yes	0h

**Register 11-51. (DMCFG\_A; PCI:2Ch, LOC:ACh) PCI Configuration Address Register for PCI Initiator-to-PCI I/O Configuration**

Bit	Description	Read	Write	Value after Reset
1:0	<b>Configuration Type (00=Type 0, 01=Type 1).</b>	Yes	Yes	00
7:2	<b>Register Number.</b>	Yes	Yes	0
10:8	<b>Function Number.</b>	Yes	Yes	0
15:11	<b>Device Number.</b>	Yes	Yes	0
23:16	<b>Bus Number.</b>	Yes	Yes	0h
30:24	<b>Reserved.</b>	Yes	No	0h
31	<p><b>Configuration Enable.</b> Writing a 1 allows Local-to-PCI I/O accesses to be converted to a PCI Configuration cycle. Parameters in this table are used to assert the PCI Configuration address.</p> <p><b>Note:</b> For more information, refer to the PCI Initiator Configuration Cycle example in Section 3.4.1.7 for M mode or Section 5.4.1.6.1 for C and J modes.</p>	Yes	Yes	0

**Register 11-52. (LAS1RR; PCI:F0h, LOC:170h) Local Address Space 1 Range Register for PCI-to-Local Bus**

Bit	Description	Read	Write	Value after Reset										
0	<b>Memory Space Indicator.</b> Writing a 0 indicates Local Address Space 1 maps into PCI Memory space. Writing a 1 indicates Address Space 1 maps into PCI I/O space.	Yes	Yes	0										
2:1	<p>When mapped into Memory space, encoding is as follows:</p> <table border="0"> <tr> <td>2/1</td> <td>Meaning</td> </tr> <tr> <td>0 0</td> <td>Locate anywhere in 32-bit PCI Address space</td> </tr> <tr> <td>0 1</td> <td>Locate below 1 MB in PCI Address space</td> </tr> <tr> <td>1 0</td> <td>Locate anywhere in 64-bit PCI Address space</td> </tr> <tr> <td>1 1</td> <td><b>Reserved</b></td> </tr> </table> <p>When mapped into I/O space, bit 1 must be set to 0. Bit 2 is included with bits [31:3] to indicate the decoding range.</p>	2/1	Meaning	0 0	Locate anywhere in 32-bit PCI Address space	0 1	Locate below 1 MB in PCI Address space	1 0	Locate anywhere in 64-bit PCI Address space	1 1	<b>Reserved</b>	Yes	Yes	00
2/1	Meaning													
0 0	Locate anywhere in 32-bit PCI Address space													
0 1	Locate below 1 MB in PCI Address space													
1 0	Locate anywhere in 64-bit PCI Address space													
1 1	<b>Reserved</b>													
3	When mapped into Memory space, writing a 1 indicates reads are prefetchable (does not affect operation of the PCI 9054, but is used for system status). When mapped into I/O space, included with bits [31:2] to indicate the decoding range.	Yes	Yes	0										
31:4	<p>Specifies which PCI Address bits to use for decoding a PCI access to Local Bus Space 1. Each bit corresponds to a PCI Address bit. Bit 31 corresponds to address bit 31. Write 1 to all bits that must be included in decode and 0 to all others. (Used in conjunction with PCIBAR3.) Default is 1 MB.</p> <p>If QSR[0]=1, defines PCI Base Address 0.</p> <p><b>Notes:</b> Range (<b>not</b> Range register) must be power of 2. "Range register value" is inverse of range.</p> <p>User should limit all I/O spaces to 256 bytes per PCI Specification v2.1.</p>	Yes	Yes	FFF000h										

Register 11-53. (LAS1BA; PCI:F4h, LOC:174h) Local Address Space 1 Local Base Address (Remap)

Bit	Description	Read	Write	Value after Reset
0	<b>Space 1 Enable.</b> Writing a 1 enables decoding of PCI addresses for PCI Target access to Local Bus Space 1. Writing a 0 disables decoding.	Yes	Yes	0
1	<b>Reserved.</b>	Yes	No	0
3:2	Not used if Local Bus Space 1 is mapped into Memory space. Included with bits [31:4] for remapping when mapped into I/O space.	Yes	Yes	00
31:4	<b>Remap PCI Address to Local Address Space 1 into Local Address Space.</b> Bits in this register remap (replace) the PCI Address bits used in decode as Local Address bits. <b>Note:</b> Remap Address value must be a multiple of the Range ( <b>not</b> the Range register).	Yes	Yes	0h

Register 11-54. (LBRD1; PCI:F8h, LOC:178h) Local Address Space 1 Bus Region Descriptor

Bit	Description	Read	Write	Value after Reset
1:0	<b>Memory Space 1 Local Bus Width.</b> Writing a 00 indicates an 8-bit bus width. Writing a 01 indicates a 16-bit bus width. Writing a 10 or 11 indicates a 32-bit bus width.	Yes	Yes	M = 11 J = 11 C = 11
5:2	<b>Memory Space 1 Internal Wait States (data-to-data; 0-15 wait states).</b>	Yes	Yes	0h
6	<b>Memory Space 1 TA#/READY# Input Enable.</b> Writing a 1 enables TA#/READY# input. Writing a 0 disables TA#/READY# input.	Yes	Yes	1
7	<b>Memory Space 1 BTERM# Input Enable.</b> Writing a 1 enables BTERM# input. Writing a 0 disables BTERM# input. For more information, refer to Section 2.2.5 for M mode or Section 4.2.5 for C and J modes.	Yes	Yes	0
8	<b>Memory Space 1 Burst Enable.</b> Writing a 1 enables bursting. Writing a 0 disables bursting.	Yes	Yes	0
9	<b>Memory Space 1 Prefetch Disable.</b> When mapped into Memory space, writing a 0 enables Read prefetching. Writing a 1 disables prefetching. If prefetching is disabled, the PCI 9054 disconnects after each Memory read.	Yes	Yes	0
10	<b>Read Prefetch Count Enable.</b> When set to 1 and Memory prefetching is enabled, the PCI 9054 prefetches up to the number of Lwords specified in prefetch count. When set to 0, the PCI 9054 ignores the count and continues prefetching until it is terminated by the PCI Bus.	Yes	Yes	0
14:11	<b>Read Prefetch Count.</b> Number of Lwords to prefetch during Memory Read cycles (0-15).	Yes	Yes	0h
31:15	<b>Reserved.</b>	Yes	No	0h

Register 11-55. (DMDAC; PCI:FCh, LOC:17Ch) PCI Initiator PCI Dual Address Cycle

Bit	Description	Read	Write	Value after Reset
31:0	<b>Upper 32 Bits of PCI Dual Address Cycle PCI Address during PCI Initiator Cycles.</b> If set to 0, the PCI 9054 performs 32-bit PCI Initiator Address access.	Yes	Yes	0h

## 11.5 RUNTIME REGISTERS

Register 11-56. (MBOX0; PCI:40h or 78h, LOC:C0h) Mailbox Register 0

Bit	Description	Read	Write	Value after Reset
31:0	<b>32-Bit Mailbox Register.</b> <i>Note:</i> Inbound Queue Port replaces Mailbox Register 0 when the I <sub>2</sub> O function is enabled (QSR[0]=1). Mailbox Register 0 is always accessible at PCI address 78h and Local address C0h.	Yes	Yes	0h

Register 11-57. (MBOX1; PCI:44h or 7Ch, LOC:C4h) Mailbox Register 1

Bit	Description	Read	Write	Value after Reset
31:0	<b>32-Bit Mailbox Register.</b> <i>Note:</i> Mailbox Register 1 is replaced by Outbound Queue Port when the I <sub>2</sub> O function is enabled (QSR[0]=1). Mailbox Register 1 is always accessible at PCI address 7Ch and Local address C4h.	Yes	Yes	0h

Register 11-58. (MBOX2; PCI:48h, LOC:C8h) Mailbox Register 2

Bit	Description	Read	Write	Value after Reset
31:0	<b>32-Bit Mailbox Register.</b>	Yes	Yes	0h

Register 11-59. (MBOX3; PCI:4Ch, LOC:CCh) Mailbox Register 3

Bit	Description	Read	Write	Value after Reset
31:0	<b>32-Bit Mailbox Register.</b>	Yes	Yes	0h

Register 11-60. (MBOX4; PCI:50h, LOC:D0h) Mailbox Register 4

Bit	Description	Read	Write	Value after Reset
31:0	<b>32-Bit Mailbox Register.</b>	Yes	Yes	0h

Register 11-61. (MBOX5; PCI:54h, LOC:D4h) Mailbox Register 5

Bit	Description	Read	Write	Value after Reset
31:0	<b>32-Bit Mailbox Register.</b>	Yes	Yes	0h

Register 11-62. (MBOX6; PCI:58h, LOC:D8h) Mailbox Register 6

Bit	Description	Read	Write	Value after Reset
31:0	<b>32-Bit Mailbox Register.</b>	Yes	Yes	0h

**Register 11-63. (MBOX7; PCI:5Ch, LOC:DCh) Mailbox Register 7**

Bit	Description	Read	Write	Value after Reset
31:0	<b>32-Bit Mailbox Register.</b>	Yes	Yes	0h

**Register 11-64. (P2LDBELL; PCI:60h, LOC:E0h) PCI-to-Local Doorbell**

Bit	Description	Read	Write	Value after Reset
31:0	<b>Doorbell Register.</b> The PCI Bus Master can write to this register and assert a Local interrupt to the Local processor. The Local processor can then read this register to determine which doorbell bit was set. The PCI Bus Master sets the doorbell by writing a 1 to a particular bit. The Local processor can clear a doorbell bit by writing a 1 to that bit position.	Yes	Yes/Clr	0h

**Register 11-65. (L2PDBELL; PCI:64h, LOC:E4h) Local-to-PCI Doorbell**

Bit	Description	Read	Write	Value after Reset
31:0	<b>Doorbell Register.</b> The Local processor can write to this register and assert a PCI interrupt. The PCI Bus Master can then read this register to determine which doorbell bit was set. The Local processor sets the doorbell by writing a 1 to a particular bit. The PCI Bus Master can clear a doorbell bit by writing a 1 to that bit position.	Yes	Yes/Clr	0h

Register 11-66. (INTCSR; PCI:68h, LOC:E8h) Interrupt Control/Status

Bit	Description	Read	Write	Value after Reset
0	<b>Enable Local Bus TEA#/LSERR#.</b> Writing a 1 enables PCI 9054 to assert TEA#/LSERR# interrupt when the PCI Bus Target Abort bit is set (PCISR[11]=1) or the Received Master Abort bit is set (PCISR[13]=1 or INTCSR[6]=1).	Yes	Yes	0
1	Enable Local Bus TEA#/LSERR# when a PCI parity error occurs during a PCI 9054 Master Transfer or a PCI 9054 Slave access.	Yes	Yes	0
2	<b>Generate PCI Bus SERR# Interrupt.</b> When set to 0, writing 1 asserts the PCI Bus SERR# interrupt.	Yes	Yes	0
3	<b>Mailbox Interrupt Enable.</b> Writing a 1 enables a Local Interrupt to be asserted when the PCI Bus writes to MBOX0 through MBOX3. To clear a Local Interrupt, the Local Bus Master must read the Mailbox. Used in conjunction with the Local Interrupt Output Enable bit (INTCSR[16]).	Yes	Yes	0
4	<b>Power Management Interrupt Enable.</b> Writing a 1 enables a Local Interrupt to be asserted when the Power Management Power State changes.	Yes	Yes	0
5	<b>Power Management Interrupt.</b> When set to 1, indicates a Power Management interrupt is pending. A Power Management interrupt is caused by a change in the Power State register (PMCSR). Writing a 1 clears the interrupt.	Yes	Yes/Clr	0
6	<b>PCI Initiator Write/PCI Target Read Local Data Parity Check Error Enable.</b> Writing a 1 enables a Local Data Parity error signal to be asserted through the LSERR#/TEA# pin. INTCSR[0] must be enabled for this to have an effect.	Yes	Yes	0
7	<b>PCI Initiator Write/PCI Target Read Local Data Parity Check Error Status.</b> When set to 1, indicates the PCI 9054 has detected a Local Data Parity check error, even if the Check Parity Error bit is disabled. Writing 1 clears this bit to 0.	Yes	Yes/Clr	0
8	<b>PCI Interrupt Enable.</b> Writing a 1 enables PCI interrupts.	Yes	Yes	1
9	<b>PCI Doorbell Interrupt Enable.</b> Writing a 1 enables Doorbell interrupts. Used in conjunction with the PCI Interrupt Enable bit (INTCSR[8]). Clearing the doorbell interrupt bits that caused the interrupt also clears the interrupt.	Yes	Yes	0
10	<b>PCI Abort Interrupt Enable.</b> Values of 1 enables Master abort or Master detect of a Target abort to assert a PCI interrupt. Used in conjunction with the PCI Interrupt Enable bit (INTCSR[8]). Clearing the abort status bits also clears the PCI interrupt.	Yes	Yes	0
11	<b>Local Interrupt Input Enable.</b> Writing a 1 enables a Local interrupt input to assert a PCI interrupt. Used in conjunction with the PCI Interrupt Enable bit (INTCSR[8]). Clearing the Local Bus cause of the interrupt also clears the interrupt.	Yes	Yes	0
12	<b>Retry Abort Enable.</b> Writing a 1 enables the PCI 9054 to treat 256 Master consecutive Retrys to a Target as a Target abort. Writing a 0 enables the PCI 9054 to attempt Master Retrys indefinitely.	Yes	Yes	0
13	<b>PCI Doorbell Interrupt Active.</b> When set to 1, indicates the PCI Doorbell interrupt is active.	Yes	No	0
14	<b>PCI Abort Interrupt Active.</b> When set to 1, indicates the PCI Abort interrupt is active.	Yes	No	0
15	<b>Local Input Interrupt Active.</b> When set to 1, indicates the Local Input interrupt is active.	Yes	No	0
16	<b>Local Interrupt Output Enable.</b> Writing a 1 enables Local interrupt output. Used in conjunction with the Mailbox Interrupt Enable bit (INTCSR[3]).	Yes	Yes	1
17	<b>Local Doorbell Interrupt Enable.</b> Writing a 1 enables Doorbell interrupts. Used in conjunction with the Local Interrupt Enable bit. Clearing the Local Doorbell interrupt bits that caused the interrupt also clears the interrupt.	Yes	Yes	0
18	<b>Local DMA Channel 0 Interrupt Enable.</b> Writing a 1 enables DMA Channel 0 interrupts. Used in conjunction with the Local Interrupt Enable bit. Clearing the DMA status bits also clears the interrupt.	Yes	Yes	0

Register 11-66. (INTCSR; PCI:68h, LOC:E8h) Interrupt Control/Status (Continued)

Bit	Description	Read	Write	Value after Reset
19	<b>Local DMA Channel 1 Interrupt Enable.</b> Writing a 1 enables DMA Channel 1 interrupts. Used in conjunction with the Local Interrupt Enable bit. Clearing the DMA status bits also clears the interrupt.	Yes	Yes	0
20	<b>Local Doorbell Interrupt Active.</b> Reading a 1 indicates the Local Doorbell interrupt is active.	Yes	No	0
21	<b>DMA Channel 0 Interrupt Active.</b> Reading a 1 indicates the DMA Channel 0 interrupt is active.	Yes	No	0
22	<b>DMA Channel 1 Interrupt Active.</b> Reading a 1 indicates the DMA Channel 1 interrupt is active.	Yes	No	0
23	<b>BIST Interrupt Active.</b> Reading a 1 indicates the BIST interrupt is active. The BIST (built-in self test) interrupt is asserted by writing a 1 to bit 6 of the PCI Configuration BIST register. Clearing bit 6 clears the interrupt. Refer to the PCIBISTR register for a description of the self test.	Yes	No	0
24	Reading a 0 indicates the PCI Initiator was the Bus Master during a Master or Target abort.	Yes	No	1
25	Reading a 0 indicates DMA CH 0 was the Bus Master during a Master or Target abort.	Yes	No	1
26	Reading a 0 indicates DMA CH 1 was the Bus Master during a Master or Target abort.	Yes	No	1
27	Reading a 0 indicates a Target abort was asserted by the PCI 9054 after 256 consecutive Master retries to a Target.	Yes	No	1
28	Reading a 1 indicates the PCI Bus wrote data to MBOX0. Enabled only if the Mailbox Interrupt Enable bit is set (INTCSR[3]=1).	Yes	No	0
29	Reading a 1 indicates the PCI Bus wrote data to MBOX1. Enabled only if the Mailbox Interrupt Enable bit is set (INTCSR[3]=1).	Yes	No	0
30	Reading a 1 indicates the PCI Bus wrote data to MBOX2. Enabled only if the Mailbox Interrupt Enable bit is set (INTCSR[3]=1).	Yes	No	0
31	Reading a 1 indicates the PCI Bus wrote data to MBOX3. Enabled only if the Mailbox Interrupt Enable bit is set (INTCSR[3]=1).	Yes	No	0

**Register 11-67. (CNTRL; PCI:6Ch, LOC:ECh) Serial EEPROM Control, PCI Command Codes, User I/O Control, and Init Control**

Bit	Description	Read	Write	Value after Reset
3:0	<b>PCI Read Command Code for DMA.</b>	Yes	Yes	1110
7:4	<b>PCI Write Command Code for DMA.</b>	Yes	Yes	0111
11:8	<b>PCI Memory Read Command Code for PCI Initiator.</b>	Yes	Yes	0110
15:12	<b>PCI Memory Write Command Code for PCI Initiator.</b>	Yes	Yes	0111
16	<b>General Purpose Output.</b> Writing a 1 causes USERo output to go high. Writing a 0 causes USERo output to go low.	Yes	Yes	1
17	<b>General Purpose Input.</b> Reading a 1 indicates the USERi input pin is high. Reading a 0 indicates the USERi pin is low.	Yes	No	—
18	Writing a 1 selects USERi to be an input to the chip. Writing a 0 selects LLOCKi# as an input. Enables the user to select between the USERi and LLOCKi# functions when USERi is chosen to be an input. DMAMODE0[12] and/or DMAMODE1[12] are select bit(s) for the pin.	Yes	Yes	1
19	Writing a 1 selects USERo to be an output from the chip. Writing a 0 selects LLOCKo# as an output. Enables the user to select between the USERo and LLOCKo# functions when USERo is chosen to be an output. DMAMODE0[12] and/or DMAMODE1[12] are select bit(s) for the pin.	Yes	Yes	1
23:20	<b>Reserved.</b>	Yes	No	0h
24	<b>Serial EEPROM Clock for Local or PCI Bus Reads or Writes to Serial EEPROM.</b> Toggling this bit asserts the serial EEPROM clock. (Refer to manufacturer's data sheet for particular serial EEPROM being used.)	Yes	Yes	0
25	<b>Serial EEPROM Chip Select.</b> For Local or PCI Bus reads or writes to the serial EEPROM, setting this bit to 1 provides the serial EEPROM chip select.	Yes	Yes	0
26	<b>Write Bit to Serial EEPROM.</b> For writes, this output bit is input to the serial EEPROM. Clocked into the serial EEPROM by the serial EEPROM clock.	Yes	Yes	0
27	<b>Read Bit from Serial EEPROM.</b> (Refer to Sections 2.4.2 and 2.4.2.1 for M mode or Sections 4.4.2 and 4.4.2.1 for C and J modes.)	Yes	No	—
28	<b>Programmed Serial EEPROM Present.</b> When set to 1, indicates that a programmed serial EEPROM is present.	Yes	No	0
29	<b>Reload Configuration Registers.</b> When set to 0, writing a 1 causes the PCI 9054 to reload the Local Configuration registers from the serial EEPROM.	Yes	Yes	0
30	<b>PCI Adapter Software Reset.</b> Writing a 1 holds the PCI 9054 Local Bus logic in a reset state, and asserts LRESETo# output. Contents of the PCI Configuration registers and the Shared Runtime registers are not reset. A software reset can only be cleared from the PCI Bus.	Yes	Yes	0
31	<b>Reserved.</b>	Yes	No	0

**Register 11-68. (PCIHIDR; PCI:70h, LOC:F0h) PCI Hardcoded Configuration ID**

Bit	Description	Read	Write	Value after Reset
15:0	<b>Vendor ID.</b> Identifies manufacturer of device. Hardcoded to the PCI SIG-issued Vendor ID of PLX (10B5h).	Yes	No	10B5h
31:16	<b>Device ID.</b> Identifies particular device. Hardcoded to the PLX part number for PCI interface chip 9054h.	Yes	No	9054h

**Register 11-69. (PCIHREV; PCI:74h, LOC:F4h) PCI Hardcoded Revision ID**

Bit	Description	Read	Write	Value after Reset
7:0	<b>Revision ID.</b> Hardcoded silicon revision of the PCI 9054.	Yes	No	Current Rev #

## 11.6 DMA REGISTERS

Register 11-70. (DMAMODE0; PCI:80h, LOC:100h) DMA Channel 0 Mode

Bit	Description	Read	Write	Value after Reset
1:0	<b>Local Bus Width.</b> Writing a 00 indicates an 8-bit bus width. Writing a 01 indicates a 16-bit bus width. Writing a 10 or 11 indicates a 32-bit bus width.	Yes	Yes	M = 11 J = 11 C = 11
5:2	<b>Internal Wait States (data-to-data).</b>	Yes	Yes	0h
6	<b>TA#/READY# Input Enable.</b> Writing a 1 enables TA#/READY# input. Writing a 0 disables TA#/READY# input.	Yes	Yes	1
7	<b>BTERM# Input Enable.</b> Writing a 1 enables BTERM# input. Writing a 0 disables BTERM# input. For more information, refer to Section 2.2.5 for M mode or Section 4.2.5 for C and J modes.	Yes	Yes	0
8	<b>Local Burst Enable.</b> Writing a 1 enables Local bursting. Writing a 0 disables Local bursting.	Yes	Yes	0
9	<b>Scatter/Gather Mode.</b> Writing a 1 indicates Scatter/Gather mode is enabled. For Scatter/Gather mode, DMA source address, destination address, and byte count are loaded from memory in PCI or Local Address spaces. Writing a 0 indicates Block mode is enabled.	Yes	Yes	0
10	<b>Done Interrupt Enable.</b> Writing a 1 enables an interrupt when done. Writing a 0 disables an interrupt when done. If DMA Clear Count mode is enabled, the interrupt does not occur until the byte count is cleared.	Yes	Yes	0
11	<b>Local Addressing Mode.</b> Writing a 1 holds the Local Address bus constant. Writing a 0 indicates the Local Address is incremented.	Yes	Yes	0
12	<b>Demand Mode.</b> Writing a 1 causes the DMA controller to operate in Demand mode, as well as to make USERo/DREQ0#/LLOCKo# an input (DREQ0#) and USERi/DACK0#/LLOCKi# an output (DACK0#). In Demand mode, the DMA controller transfers data when its DREQ0# input is asserted. Asserts DACK0# to indicate the current Local Bus transfer is in response to DREQ0# input. DMA controller transfers Lwords (32 bits) of data. This may result in multiple transfers for an 8- or 16-bit bus.	Yes	Yes	0
13	<b>Memory Write and Invalidate Mode for DMA Transfers.</b> When set to 1, the PCI 9054 performs Memory Write and Invalidate cycles to the PCI Bus. The PCI 9054 supports Memory Write and Invalidate sizes of 8 or 16 Lwords. Size is specified in the System Cache Line Size bits (PCICLSR[7:0]). If a size other than 8 or 16 is specified, the PCI 9054 performs Write transfers rather than Memory Write and Invalidate transfers. Transfers must start and end at cache line boundaries.	Yes	Yes	0
14	<b>DMA EOT# Enable.</b> Writing a 1 enables the EOT# input pin. Writing a 0 disables the EOT# input pin.	Yes	Yes	0
15	<b>Fast/Slow Terminate Mode Select.</b> Writing a 0 sets PCI 9054 into the Slow Terminate mode. As a result in C or J modes, BLAST# is asserted on the last Data transfer to terminate DMA transfer. As a result in M mode, BDIP# is de-asserted at the nearest 16-byte boundary and stops the DMA transfer. Writing a 1 indicates that if EOT# is asserted or DREQ0# is de-asserted in Demand mode during DMA will immediately terminate the DMA transfer. In M mode, writing a 1 indicates BDIP# output is disabled. As a result, the PCI 9054 DMA transfer terminates immediately when EOT# is asserted or when DREQ0# is de-asserted in Demand mode.	Yes	Yes	0
16	<b>DMA Clear Count Mode.</b> Writing a 1 clears the byte count in each Scatter/Gather descriptor when the corresponding DMA transfer is complete.	Yes	Yes	0
17	<b>DMA Channel 0 Interrupt Select.</b> Writing a 1 routes the DMA Channel 0 interrupt to the PCI Bus interrupt. Writing a 0 routes the DMA Channel 0 interrupt to the Local Bus interrupt.	Yes	Yes	0
18	<b>DAC Chain Load.</b> When set to 1, enables the descriptor to load the PCI Dual Address Cycle value. Otherwise, it uses the contents of the register.	Yes	Yes	0
31:19	<b>Reserved.</b>	Yes	No	0h

## DMA Registers

## Register 11-71. (DMAPADR0; PCI:84h, LOC:104h) DMA Channel 0 PCI Address

Bit	Description	Read	Write	Value after Reset
31:0	<b>PCI Address Register.</b> Indicates from where in PCI Memory space DMA transfers (reads or writes) start.	Yes	Yes	0h

## Register 11-72. (DMALADR0; PCI:88h, LOC:108h) DMA Channel 0 Local Address

Bit	Description	Read	Write	Value after Reset
31:0	<b>Local Address Register.</b> Indicates from where in Local Memory space DMA transfers (reads or writes) start.	Yes	Yes	0h

## Register 11-73. (DMASIZ0; PCI:8Ch, LOC:10Ch) DMA Channel 0 Transfer Size (Bytes)

Bit	Description	Read	Write	Value after Reset
22:0	<b>DMA Transfer Size (Bytes).</b> Indicates the number of bytes to transfer during a DMA operation.	Yes	Yes	0h
31:23	<b>Reserved.</b>	Yes	No	0h

## Register 11-74. (DMADPR0; PCI:90h, LOC:110h) DMA Channel 0 Descriptor Pointer

Bit	Description	Read	Write	Value after Reset
0	<b>Descriptor Location.</b> Writing a 1 indicates PCI Address space. Writing a 0 indicates Local Address space.	Yes	Yes	0
1	<b>End of Chain.</b> Writing a 1 indicates end of chain. Writing a 0 indicates not end of chain descriptor. (Same as Block mode.)	Yes	Yes	0
2	<b>Interrupt after Terminal Count.</b> Writing a 1 causes an interrupt to be asserted after the terminal count for this descriptor is reached. Writing a 0 disables interrupts from being asserted.	Yes	Yes	0
3	<b>Direction of Transfer.</b> Writing a 1 indicates transfers from the Local Bus to the PCI Bus. Writing a 0 indicates transfers from the PCI Bus to the Local Bus.	Yes	Yes	0
31:4	<b>Next Descriptor Address.</b> Qword aligned (bits [3:0]=0000).	Yes	Yes	0h

Register 11-75. (DMAMODE1; PCI:94h, LOC:114h) DMA Channel 1 Mode

Bit	Description	Read	Write	Value after Reset
1:0	<b>Local Bus Width.</b> Writing a 00 indicates an 8-bit bus width. Writing a 01 indicates a 16-bit bus width. Writing a 10 or 11 indicates a 32-bit bus width.	Yes	Yes	M = 11 J = 11 C = 11
5:2	<b>Internal Wait States (data-to-data).</b>	Yes	Yes	0h
6	<b>TA#/READY# Input Enable.</b> Writing a 1 enables TA#/READY# input. Writing a 0 disables TA#/READY# input.	Yes	Yes	1
7	<b>BTERM# Input Enable.</b> Writing a 1 enables BTERM# input. Writing a 0 disables BTERM# input. For more information, refer to Section 2.2.5 for M mode or Section 4.2.5 for C and J modes.	Yes	Yes	0
8	<b>Local Burst Enable.</b> Writing a 1 enables Local bursting. Writing a 0 disables Local bursting.	Yes	Yes	0
9	<b>Scatter/Gather Mode.</b> Writing a 1 indicates Scatter/Gather mode is enabled. For Scatter/Gather mode, the DMA source address, destination address, and byte count are loaded from memory in PCI or Local Address spaces. Writing a 0 indicates Block mode is enabled.	Yes	Yes	0
10	<b>Done Interrupt Enable.</b> Writing a 1 enables interrupt when done. Writing a 0 disables the interrupt when done. If DMA Clear Count mode is enabled, the interrupt does not occur until the byte count is cleared.	Yes	Yes	0
11	<b>Local Addressing Mode.</b> Writing a 1 holds the Local address bus constant. Writing a 0 indicates the Local address is incremented.	Yes	Yes	0
12	<b>Reserved.</b>	Yes	No	0
13	<b>Memory Write and Invalidate Mode for DMA Transfers.</b> When set to 1, the PCI 9054 performs Memory Write and Invalidate cycles to the PCI Bus. The PCI 9054 supports Memory Write and Invalidate sizes of 8 or 16 Lwords. Size is specified in the System Cache Line Size bits (PCICLSR[7:0]). If a size other than 8 or 16 is specified, the PCI 9054 performs Write transfers rather than Memory Write and Invalidate transfers. Transfers must start and end at cache line boundaries.	Yes	Yes	0
14	<b>DMA EOT# Enable.</b> Writing a 1 enables the EOT# input pin. Writing a 0 disables the EOT# output pin.	Yes	Yes	0
15	<b>Fast/Slow Terminate Mode Select.</b> Writing a 0 sets the PCI 9054 into Slow Terminate mode. As a result in C or J modes, BLAST# is asserted to terminate the DMA transfer. As a result in M mode, BDIP# is de-asserted at the nearest 16-byte boundary and stops the DMA transfer. Writing a 1 indicates that asserting EOT# during DMA will terminate the DMA transfer. In M mode, writing a 1 indicates BDIP# output is disabled. As a result, the PCI 9054 DMA transfer terminates immediately when EOT# is asserted.	Yes	Yes	0
16	<b>DMA Clear Count Mode.</b> When set to 1, the byte count in each Scatter/Gather descriptor is cleared when the corresponding DMA transfer is complete.	Yes	Yes	0
17	<b>DMA Channel 1 Interrupt Select.</b> Writing a 1 routes the DMA Channel 1 interrupt to the PCI Bus interrupt. Writing a 0 routes the DMA Channel 1 interrupt to the Local Bus interrupt.	Yes	Yes	0
18	<b>DAC Chain Load.</b> When set to 1, enables the descriptor to load the PCI Dual Address Cycle value. Otherwise, it uses the contents of the register.	Yes	Yes	0
31:19	<b>Reserved.</b>	Yes	No	0h

**Register 11-76. (DMAPADR1; PCI:98h, LOC:118h) DMA Channel 1 PCI Address**

Bit	Description	Read	Write	Value after Reset
31:0	<b>PCI Address Register.</b> Indicates from where in PCI Memory space DMA transfers (reads or writes) start.	Yes	Yes	0h

**Register 11-77. (DMALADR1; PCI:9Ch, LOC:11Ch) DMA Channel 1 Local Address**

Bit	Description	Read	Write	Value after Reset
31:0	<b>Local Address Register.</b> Indicates from where in Local Memory space DMA transfers (reads or writes) start.	Yes	Yes	0h

**Register 11-78. (DMASIZ1; PCI:A0h, LOC:120h) DMA Channel 1 Transfer Size (Bytes)**

Bit	Description	Read	Write	Value after Reset
22:0	<b>DMA Transfer Size (Bytes).</b> Indicates the number of bytes to transfer during a DMA operation.	Yes	Yes	0h
31:23	<b>Reserved.</b>	Yes	No	0h

**Register 11-79. (DMADPR1; PCI:A4h, LOC:124h) DMA Channel 1 Descriptor Pointer**

Bit	Description	Read	Write	Value after Reset
0	<b>Descriptor Location.</b> Writing a 1 indicates PCI Address space. Writing a 0 indicates Local Address space.	Yes	Yes	0
1	<b>End of Chain.</b> Writing a 1 indicates end of chain. Writing a 0 indicates not end of chain descriptor. (Same as Block mode.)	Yes	Yes	0
2	<b>Interrupt after Terminal Count.</b> Writing a 1 causes an interrupt to be asserted after the terminal count for this descriptor is reached. Writing a 0 disables interrupts from being asserted.	Yes	Yes	0
3	<b>Direction of Transfer.</b> Writing a 1 indicates transfers from the Local Bus to the PCI Bus. Writing a 0 indicates transfers from the PCI Bus to the Local Bus.	Yes	Yes	0
31:4	<b>Next Descriptor Address.</b> Qword aligned (bits [3:0]=0000).	Yes	Yes	0h

**Register 11-80. (DMACSR0; PCI:A8h, LOC:128h) DMA Channel 0 Command/Status**

Bit	Description	Read	Write	Value after Reset
0	<b>Channel 0 Enable.</b> Writing a 1 enables channel to transfer data. Writing a 0 disables the channel from starting a DMA transfer, and if in the process of transferring data, suspends the transfer (pause).	Yes	Yes	0
1	<b>Channel 0 Start.</b> Writing a 1 causes the channel to start transferring data if the channel is enabled.	No	Yes/Set	0
2	<b>Channel 0 Abort.</b> Writing a 1 causes the channel to abort current transfer. Channel 0 Enable bit must be cleared (bit [0]=0). Sets Channel 0 Done (bit [4] = 1) when abort is complete.	No	Yes/Set	0
3	<b>Channel 0 Clear Interrupt.</b> Writing a 1 clears Channel 0 interrupts.	No	Yes/Clr	0
4	<b>Channel 0 Done.</b> Reading a 1 indicates a channel transfer is complete. Reading a 0 indicates a channel transfer is not complete.	Yes	No	1
7:5	<b>Reserved.</b>	Yes	No	000

**Register 11-81. (DMACSR1; PCI:A9h, LOC:129h) DMA Channel 1 Command/Status**

Bit	Description	Read	Write	Value after Reset
0	<b>Channel 1 Enable.</b> Writing a 1 enables channel to transfer data. Writing a 0 disables the channel from starting a DMA transfer, and if in the process of transferring data, suspends the transfer (pause).	Yes	Yes	0
1	<b>Channel 1 Start.</b> Writing a 1 causes channel to start transferring data if the channel is enabled.	No	Yes/Set	0
2	<b>Channel 1 Abort.</b> Writing a 1 causes channel to abort current transfer. Channel 1 Enable bit must be cleared (bit [0]=0). Sets Channel 1 Done (bit [4] = 1) when abort is complete.	No	Yes/Set	0
3	<b>Channel 1 Clear Interrupt.</b> Writing a 1 clears Channel 1 interrupts.	No	Yes/Clr	0
4	<b>Channel 1 Done.</b> Reading a 1 indicates a channel transfer is complete. Reading a 0 indicates a channel transfer is not complete.	Yes	No	1
7:5	<b>Reserved.</b>	Yes	No	000

**DMA Registers****Register 11-82. (DMAARB; PCI:ACh, LOC:12Ch) DMA Arbitration**

Same as “(MARBR; PCI:08h or ACh, LOC:88h or 12Ch) Mode/DMA Arbitration,” on page 11-20.

**Register 11-83. (DMATHR; PCI:B0h, LOC:130h) DMA Threshold**

Bit	Description	Read	Write	Value after Reset
3:0	<b>DMA Channel 0 PCI-to-Local Almost Full (C0PLAF).</b> Number of full entries (divided by two, minus one) in the FIFO before requesting the Local Bus for writes. (C0PLAF+1) + (C0PLAE+1) should be $\leq$ a FIFO Depth of 32.	Yes	Yes	0h
7:4	<b>DMA Channel 0 Local-to-PCI Almost Empty (C0LPAE).</b> Number of empty entries (divided by two, minus one) in the FIFO before requesting the Local Bus for reads. (C0LPAF+1) + (C0LPAE+1) should be $\leq$ a FIFO depth of 32.	Yes	Yes	0h
11:8	<b>DMA Channel 0 Local-to-PCI Almost Full (C0LPAF).</b> Number of full entries (divided by two, minus one) in the FIFO before requesting the PCI Bus for writes.	Yes	Yes	0h
15:12	<b>DMA Channel 0 PCI-to-Local Almost Empty (C0PLAE).</b> Number of empty entries (divided by two, minus one) in the FIFO before requesting the PCI Bus for reads.	Yes	Yes	0h
19:16	<b>DMA Channel 1 PCI-to-Local Almost Full (C1PLAF).</b> Number of full entries, minus one, in the FIFO before requesting the Local Bus for writes. (C1PLAF+1) + (C1PLAE+1) should be $\leq$ a FIFO depth of 16.	Yes	Yes	0h
23:20	<b>DMA Channel 1 Local-to-PCI Almost Empty (C1LPAE).</b> Number of empty entries, minus one, in the FIFO before requesting the Local Bus for reads. (C1PLAF) + (C1PLAE) should be $\leq$ a FIFO depth of 16.	Yes	Yes	0h
27:24	<b>DMA Channel 1 Local-to-PCI Almost Full (C1LPAF).</b> Number of full entries, minus one, in the FIFO before requesting the PCI Bus for writes.	Yes	Yes	0h
31:28	<b>DMA Channel 1 PCI-to-Local Almost Empty (C1PLAE).</b> Number of empty entries, minus one, in the FIFO before requesting the PCI Bus for reads.	Yes	Yes	0h

**Note:** For DMA Channel 0 only, if number of entries needed is  $x$ , then the value is one less than half the number of entries (that is,  $x/2 - 1$ ).

**Register 11-84. (DMADAC0; PCI:B4h, LOC:134h) DMA Channel 0 PCI Dual Address Cycle Address**

Bit	Description	Read	Write	Value after Reset
31:0	<b>Upper 32 Bits of the PCI Dual Address Cycle PCI Address during DMA Channel 0 Cycles.</b> If set to 0h, the PCI 9054 performs a 32-bit DMA Channel 0 Address access.	Yes	Yes	0h

**Register 11-85. (DMADAC1; PCI:B8h, LOC:138h) DMA Channel 1 PCI Dual Address Cycle Address**

Bit	Description	Read	Write	Value after Reset
31:0	<b>Upper 32 Bits of the PCI Dual Address Cycle PCI Address during DMA Channel 1 Cycles.</b> If set to 0h, the PCI 9054 performs a 32-bit DMA Channel 1 Address access.	Yes	Yes	0h

## 11.7 MESSAGING QUEUE REGISTERS

**Register 11-86. (OPQIS; PCI:30h, LOC:B0h) Outbound Post Queue Interrupt Status**

Bit	Description	Read	Write	Value after Reset
2:0	<i>Reserved.</i>	Yes	No	000
3	<b>Outbound Post Queue Interrupt.</b> Set when the Outbound Post Queue is not empty. Not affected by the Interrupt Mask bit.	Yes	No	0
31:4	<i>Reserved.</i>	Yes	No	0h

**Register 11-87. (OPQIM; PCI:34h, LOC:B4h) Outbound Post Queue Interrupt Mask**

Bit	Description	Read	Write	Value after Reset
2:0	<i>Reserved.</i>	Yes	No	000
3	<b>Outbound Post Queue Interrupt Mask.</b> Writing a 1 masks the interrupt.	Yes	Yes	1
31:4	<i>Reserved.</i>	Yes	No	0h

**Register 11-88. (IQP; PCI:40h) Inbound Queue Port**

Bit	Description	Read	Write	Value after Reset
31:0	Value written by the PCI Master is stored into the Inbound Post Queue, which is located in Local memory at the address pointed to by the Queue Base Address + Queue Size + Inbound Post Head Pointer. From the time of the PCI write until the Local Memory write and update of the Inbound Post Queue Head Pointer, further accesses to this register result in a Retry. A Local interrupt is asserted when the Inbound Post Queue is not empty. When the port is read by the PCI Master, the value is read from the Inbound Free Queue, which is located in Local memory at the address pointed to by the Queue Base Address + Inbound Free Tail Pointer. If the queue is empty, FFFFFFFh is returned.	PCI	PCI	0h

**Register 11-89. (OQP; PCI:44h) Outbound Queue Port**

Bit	Description	Read	Write	Value after Reset
31:0	Value written by the PCI Master is stored into the Outbound Free Queue, which is located in Local memory at the address pointed to by the Queue Base Address + 3*Queue Size + Outbound Free Head Pointer. From the time of the PCI write until the Local Memory write and update of the Outbound Free Queue Head Pointer, further accesses to this register result in a Retry. If the queue fills up, a Local NMI interrupt is asserted. When the port is read by the PCI Master, the value is read from the Outbound Post Queue, which is located in Local memory at the address pointed to by the Queue Base Address + 2*Queue Size + Outbound Post Tail Pointer. If the queue is empty, FFFFFFFh is returned. A PCI interrupt is asserted if the Outbound Post Queue is not empty.	PCI	PCI	0h

**Register 11-90. (MQCR; PCI:C0h, LOC:140h) Messaging Queue Configuration**

Bit	Description	Read	Write	Value after Reset																		
0	<b>Queue Enable.</b> Writing a 1 allows accesses to the Inbound and Outbound Queue ports. If cleared to 0, writes are accepted but ignored and reads return FFFFFFFh.	Yes	Yes	0																		
5:1	<b>Circular Queue Size.</b> Contains the size of one of the circular FIFO queues. Each of the four queues are the same size. Queue Size Encoding values: <table border="1"> <thead> <tr> <th>Bits [5:1]</th> <th>Number of entries</th> <th>Total size</th> </tr> </thead> <tbody> <tr> <td>00001</td> <td>4-KB entries</td> <td>64 KB</td> </tr> <tr> <td>00010</td> <td>8-KB entries</td> <td>128 KB</td> </tr> <tr> <td>00100</td> <td>16-KB entries</td> <td>256 KB</td> </tr> <tr> <td>01000</td> <td>32-KB entries</td> <td>512 KB</td> </tr> <tr> <td>10000</td> <td>64-KB entries</td> <td>1 MB</td> </tr> </tbody> </table>	Bits [5:1]	Number of entries	Total size	00001	4-KB entries	64 KB	00010	8-KB entries	128 KB	00100	16-KB entries	256 KB	01000	32-KB entries	512 KB	10000	64-KB entries	1 MB	Yes	Yes	00001
Bits [5:1]	Number of entries	Total size																				
00001	4-KB entries	64 KB																				
00010	8-KB entries	128 KB																				
00100	16-KB entries	256 KB																				
01000	32-KB entries	512 KB																				
10000	64-KB entries	1 MB																				
31:6	<b>Reserved.</b>	Yes	No	0h																		

**Register 11-91. (QBAR; PCI:C4h, LOC:144h) Queue Base Address**

Bit	Description	Read	Write	Value after Reset
19:0	<b>Reserved.</b>	Yes	No	0h
31:20	<b>Queue Base Address.</b> Local Memory base address of circular queues. Queues must be aligned on a 1-MB boundary.	Yes	Yes	0h

**Register 11-92. (IFHPR; PCI:C8h, LOC:148h) Inbound Free Head Pointer**

Bit	Description	Read	Write	Value after Reset
1:0	<b>Reserved.</b>	Yes	No	00
19:2	<b>Inbound Free Head Pointer.</b> Local Memory Offset for the Inbound Free Queue. Maintained by the Local CPU software.	Yes	Yes	0h
31:20	<b>Queue Base Address.</b>	Yes	No	0h

**Register 11-93. (IFTPR; PCI:CCh, LOC:14Ch) Inbound Free Tail Pointer**

Bit	Description	Read	Write	Value after Reset
1:0	<b>Reserved.</b>	Yes	No	00
19:2	<b>Inbound Free Tail Pointer.</b> Local Memory offset for the Inbound Free Queue. Maintained by the hardware and incremented modulo the queue size.	Yes	Yes	0h
31:20	<b>Queue Base Address.</b>	Yes	No	0h

**Register 11-94. (IPHPR; PCI:D0h, LOC:150h) Inbound Post Head Pointer**

Bit	Description	Read	Write	Value after Reset
1:0	<i>Reserved.</i>	Yes	No	00
19:2	<b>Inbound Post Head Pointer.</b> Local Memory offset for the Inbound Post Queue. Maintained by the hardware and incremented modulo the queue size.	Yes	Yes	0h
31:20	<b>Queue Base Address.</b>	Yes	No	0h

**Register 11-95. (IPTPR; PCI:D4h, LOC:154h) Inbound Post Tail Pointer**

Bit	Description	Read	Write	Value after Reset
1:0	<i>Reserved.</i>	Yes	No	00
19:2	<b>Inbound Post Tail Pointer.</b> Local Memory offset for the Inbound Post Queue. Maintained by the Local CPU software.	Yes	Yes	0h
31:20	<b>Queue Base Address.</b>	Yes	No	0h

**Register 11-96. (OFHPR; PCI:D8h, LOC:158h) Outbound Free Head Pointer**

Bit	Description	Read	Write	Value after Reset
1:0	<i>Reserved.</i>	Yes	No	00
19:2	<b>Outbound Free Head Pointer.</b> Local Memory offset for the Outbound Free Queue. Maintained by the hardware and incremented modulo the queue size.	Yes	Yes	0h
31:20	<b>Queue Base Address.</b>	Yes	No	0h

**Register 11-97. (OFTPR; PCI:DCh, LOC:15Ch) Outbound Free Tail Pointer**

Bit	Description	Read	Write	Value after Reset
1:0	<i>Reserved.</i>	Yes	No	00
19:2	<b>Outbound Free Tail Pointer.</b> Local Memory offset for the Outbound Free Queue. Maintained by the Local CPU software.	Yes	Yes	0h
31:20	<b>Queue Base Address.</b>	Yes	No	0h

**Register 11-98. (OPHPR; PCI:E0h, LOC:160h) Outbound Post Head Pointer**

Bit	Description	Read	Write	Value after Reset
1:0	<i>Reserved.</i>	Yes	No	00
19:2	<b>Outbound Post Head Pointer.</b> Local Memory offset for the Outbound Post Queue. Maintained by the Local CPU software.	Yes	Yes	0h
31:20	<b>Queue Base Address.</b>	Yes	No	0h

**Register 11-99. (OPTPR; PCI:E4h, LOC:164h) Outbound Post Tail Pointer**

Bit	Description	Read	Write	Value after Reset
1:0	<i>Reserved.</i>	Yes	No	00
19:2	<b>Outbound Post Tail Pointer.</b> Local Memory offset for the Outbound Post Queue. Maintained by the hardware and incremented modulo the queue size.	Yes	Yes	0h
31:20	<b>Queue Base Address.</b>	Yes	No	0h

**Register 11-100. (QSR; PCI:E8h, LOC:168h) Queue Status/Control**

Bit	Description	Read	Write	Value after Reset
0	<b>I<sub>2</sub>O Decode Enable.</b> When set, replaces the MBOX0 and MBOX1 registers with the Inbound and Outbound Queue Port registers and redefines Space 1 as PCI Base Address 0 to be accessed by PCIBAR0. Former Space 1 registers F0, F4, and F8 should be programmed to configure their shared I <sub>2</sub> O Memory space, defined as PCI Base Address 0.	Yes	Yes	0
1	<b>Queue Local Space Select.</b> When set to 0, use the Local Address Space 0 Bus Region descriptor for Queue accesses. When set to 1, use the Local Address Space 1 Bus Region descriptor for Queue accesses.	Yes	Yes	0
2	<b>Outbound Post Queue Prefetch Enable.</b> Writing a 1 causes prefetching to occur from the Outbound Post Queue if it is not empty.	Yes	Yes	0
3	<b>Inbound Free Queue Prefetch Enable.</b> Writing a 1 causes prefetching to occur from the Inbound Free Queue if it is not empty.	Yes	Yes	0
4	<b>Inbound Post Queue Interrupt Mask.</b> Writing a 1 masks the interrupt.	Yes	Yes	1
5	<b>Inbound Post Queue Interrupt Not Empty.</b> Set when the Inbound Post Queue is not empty. Not affected by the Interrupt Mask bit.	Yes	No	0
6	<b>Outbound Free Queue Overflow Interrupt Mask.</b> When set to 0, masks the interrupt. Default is 1.	Yes	Yes	1
7	<b>Outbound Free Queue Overflow Interrupt Full.</b> Set when the Outbound Free Queue becomes full. A Local TEA#/LSERR# (NMI) interrupt is asserted. Writing a 1 clears the interrupt.	Yes	Yes/Clr	0
31:8	<i>Unused.</i>	Yes	No	0h



## 12 PIN DESCRIPTION

### 12.1 PIN SUMMARY

Tables in this section describe each PCI 9054 pin. Table 12-2 through Table 12-6 provide pin information common to all Local Bus modes of operation:

- Power and Ground
- Serial EEPROM Interface
- PCI System Bus Interface
- Local Bus Mode and Processor Independent Interface

Pins in Table 12-7 through Table 12-9 correspond to the PCI 9054 Local Bus modes—M, C, and J:

- M Bus Mode Interface Pin Description (32-bit address/32-bit data, non-multiplexed)
- C Bus Mode Interface Pin Description (32-bit address/32-bit data, non-multiplexed)
- J Bus Mode Interface Pin Description (32-bit address/32-bit data, multiplexed)

For a visual view of the chip pinout, refer to Section 14, “Package, Signal, and Pinout Specs.”

The following pins have internal pull-ups:

ADS#, BDIP#, BI#, BIGEND#/WAIT#, BLAST#, BTERM#, BURST#, CCS#, DMPAF/EOT#, DP[3:0]#, EEDI/EEDO, LA[31:30], LA[28:0], LAD[31:0], LBE[3:0]#, LINT#, LRESETo#, LSERR#, LW/R#, RD/WR#, MDREQ#/DMPAF/EOT#, MODE[1:0], READY#, TA#, TEA#, TS#, TSIZ[0:1], WAIT#.

The TEST pin has an internal pull-down.

**Notes:** Due to the complexity of pin multiplexing, LA[29] (C mode) or LA[2] (M mode) requires an external pull-up. In J mode, ALE requires a pull-down.

**M mode only:** The BB# pin requires a 510 ohm external pull-up resistor for PCI 9054\_AB revision. The BB# pin requires a weaker pull-up (4.7K or above) for PCI 9054\_AB revision.

Table 12-1 lists abbreviations used in this section to represent various pin types.

**Table 12-1. Pin Type Abbreviations**

Abbreviation	Pin Type
I/O	Input and output pin
I	Input pin only
O	Output pin only
TS	Tri-state pin
OC	Open collector pin
TP	Totem pole pin
STS	Sustained tri-state pin, driven high for one CLK before float
DTS	Driven tri-state pin, driven high for one-half CLK before float

All Local Bus internal pull-ups go through a 100k-ohm resistor. All Local Bus internal pull-downs go through a 50k-ohm resistor.

All Local I/O pins should have external pull-ups or pull-downs, which depend upon the application and pin polarity. (Use approximately 3k to 10k ohms.) This is recommended due to the weak value of the internal pull-ups and pull-downs.

Unspecified pins are not connected (NC).

**Note for PCI Pins:** **DO NOT** pull any pins up or down unless the PCI 9054 is being used in an embedded design. Refer to PCI Local Bus Specification, v2.1, page 123.

## 12.2 PINOUT COMMON TO ALL BUS MODES

Table 12-2. Power and Ground Pins (176-Pin PQFP)

Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	Function
TEST	Test Pin	1	I	155	Pulled high for test and low for normal operation. When pulled high: All outputs except USERo/DREQ0#/LLOCKo# and LEDon/LEDin are placed in tri-state. USERo/DREQ0#/LLOCKo# provide NANDTREE output.
VDD	Power (+3.3V)	15	I	1, 20, 28, 35, 45, 62, 70, 89, 99, 109, 116, 133, 141, 147, 162	Three-volt power supply pins for core and I/O buffers. Liberal .01 to .1 $\mu$ F decoupling capacitors should be placed near the PCI 9054.
VSS	Ground	12	I	19, 27, 44, 61, 69, 88, 108, 115, 132, 140, 161, 176	Ground pins.

**Note:** The die contains 224 pads. Power and Grounds are double bounded in the PQFP package to meet proper drive strength of the buffers.

Table 12-3. Power and Ground Pins (225-Pin PBGA)

Symbol	Signal Name	Total Die Pads	Total Pins	Pin Type	PBGA Pin Number	Function
TEST	Test Pin	1	1	I	A8	Pulled high for test, low for normal operation. When pulled high: All outputs except USERo/DREQ0#/LLOCKo# and LEDon/LEDin are placed in tri-state. USERo/DREQ0#/LLOCKo# provides NANDTREE output.
VDD (Core)	Power (+3.3V)	34	34	I	A1, E2, G2, G5, H5, J4, J5, L2, K5, R1, P5, R6, P7, L8, R9, P10, N11, R15, L13, K11, J13, J11, H12, G12, E15, A15, B12, C11, C10, D9, D8, A6, B6, D6	Three-volt power supply pins for core and I/O buffers. Liberal .01 to .1 $\mu$ F decoupling capacitors should be placed near the PCI 9054.
VSS	Ground	39	13	I	F8, G7, G8, G9, H6, H7, H8, H9, H10, J7, J8, J9, K8	Ground pins.

Table 12-4. Serial EEPROM Interface Pins

Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
EECS	Serial EEPROM Chip Select	1	O TP 6 mA	164	F7	Serial EEPROM Chip Select.
EEDI/EEDO	Serial EEPROM Data IN/ Serial EEPROM Data OUT	1	I/O TP 6 mA	166	C5	Multiplexed Write/Read data to a serial EEPROM pin.
EESK	Serial Data Clock	1	O TP 6 mA	165	B5	Serial EEPROM clock pin.

**Note:** The serial EEPROM interface operates at core voltage (+3.3V). The PCI 9054 requires use of a serial EEPROM that can operate up to 1 MHz.

Table 12-5. PCI System Bus Interface Pins

Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
AD[31:0]	Address and Data	32	I/O TS PCI	173-175, 2-5, 8-15, 31-34, 36-40, 42-43, 46-51	F6, A2, C3, D4, B1, C2, E5, D2, G6, D1, E3, F5, F4, F3, F2, J6, L1, K4, M1, M2, L4, N1, M3, N2, P1, N3, M4, R2, P3, L5, N4, R3	All multiplexed on the same PCI pins. The Bus transaction consists of an Address phase, followed by one or more Data phases. The PCI 9054 supports both Read and Write bursts.
C/BE[3:0]#	Bus Command and Byte Enables	4	I/O TS PCI	6, 16, 30, 41	D3, G4, K3, K6	All multiplexed on the same PCI pins. During the Address phase of a transaction, defines the bus command. During the Data phase, used as byte enables. Refer to the PCI spec for further details.
DEVSEL#	Device Select	1	I/O STS PCI	22	H1	When actively driven, indicates the driving device has decoded its address as Target of current access. As an input, indicates whether any device on the bus is selected.
ENUM#	Enumeration	1	O OC PCI	52	P4	Interrupt output set when an adapter using the PCI 9054 is freshly inserted or ready to be removed from a PCI slot.
FRAME#	Cycle Frame	1	I/O STS PCI	17	G3	Driven by the current Master to indicate the beginning and duration of an access. FRAME# is asserted to indicate the bus transaction is beginning. While FRAME# is asserted, Data transfers continue. When FRAME# is de-asserted, the transaction is in the final Data phase.
GNT#	Grant	1	I	171	C4	Indicates to the agent that access to the bus is granted. Every Master has its own REQ# and GNT#.
IDSEL	Initialization Device Select	1	I	7	C1	Used as a chip select during Configuration Read and Write transactions.

Table 12-5. PCI System Bus Interface Pins (Continued)

Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
INTA#	Interrupt A	1	O OC PCI	168	B4	PCI Interrupt request.
IRDY#	Initiator Ready	1	I/O STS PCI	18	G1	Indicates ability of the initiating agent (Bus Master) to complete the current Data phase of the transaction.
LOCK#	Lock	1	I/O STS PCI	24	J2	Indicates an atomic operation that may require multiple transactions to complete.
PAR	Parity	1	I/O TS PCI	29	K1	Even parity across AD[31:0] and C/BE[3:0]#. All PCI agents require parity generation. PAR is stable and valid one clock after the Address phase. For Data phases, PAR is stable and valid one clock after either IRDY# is asserted on a Write transaction or TRDY# is asserted on a Read transaction. Once PAR is valid, it remains valid until one clock after current Data phase completes.
PCLK	Clock	1	I	170	A3	Provides timing for all transactions on PCI and is an input to every PCI device. The PCI 9054 operates up to 33 MHz.
PERR#	Parity Error	1	I/O STS PCI	25	J1	Reports data parity errors during all PCI transactions, except during a special cycle.
PME#	Power Management Event	1	O OC PCI	167	A4	Wake-up event interrupt.
REQ#	Request	1	O STS PCI	172	B3	Indicates to arbiter that this agent must use the bus. Every Master has its own GNT# and REQ#.
RST#	Reset	1	I	169	D5	Used to bring PCI-specific registers, sequencers, and signals to a consistent state.
SERR#	Systems Error	1	O OC PCI	26	J3	Reports address parity errors, data parity errors on the Special Cycle command, or any other system error where the result is catastrophic.
STOP#	Stop	1	I/O STS PCI	23	H4	Indicates the current Target is requesting that the Master stop the current transaction.
TRDY#	Target Ready	1	I/O STS PCI	21	H3	Indicates ability of the Target agent (selected device) to complete the current Data phase of the transaction.

Table 12-6. Local Bus Mode and Processor Independent Interface Pins

Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function															
CCS#	Configuration Register Select	1	I	160	E7	Internal PCI 9054 registers are selected when CCS# is asserted low.															
LCLK	Local Processor Clock	1	I	142	D11	Local clock input.															
LEDOn/ LEDIn	LEDOn/LEDIn	1	I/O TS 24 mA	53	K7	As an output, acts as the Hot Swap board indicator LED. As an input, monitors the CompactPCI board latch status.															
LFRAME#	PCI Buffered FRAME# Signal	1	O TP 6 mA	—	G13	Could be used to monitor PCI Bus activity. <i>Available only on the PBGA package.</i>															
LINT#	Local Interrupt	1	I/O OC 12 mA	154	B8	As an input, when asserted low, causes PCI interrupt. As an output, a synchronous level output that remains asserted as long as an interrupt condition exists. If edge level interrupt is required, disabling and then enabling Local interrupts through INTCSR creates an edge if an interrupt condition still exists or a new interrupt condition occurs.															
LRESETo#	Local Bus Reset Out	1	O TP 12 mA	152	A9	Asserted when the PCI 9054 chip is reset. Can be used to drive RESET# input of a Local processor.															
MODE[1:0]	Bus Mode	2	I	157-156	B7, E8	Selects the PCI 9054 bus operation mode: <table border="1"> <thead> <tr> <th>Mode 0</th> <th>Mode 1</th> <th>Bus Mode</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>1</td> <td>M</td> </tr> <tr> <td>1</td> <td>0</td> <td>J</td> </tr> <tr> <td>0</td> <td>1</td> <td>Reserved</td> </tr> <tr> <td>0</td> <td>0</td> <td>C</td> </tr> </tbody> </table>	Mode 0	Mode 1	Bus Mode	1	1	M	1	0	J	0	1	Reserved	0	0	C
Mode 0	Mode 1	Bus Mode																			
1	1	M																			
1	0	J																			
0	1	Reserved																			
0	0	C																			
USERi/  DACK0#/  LLOCKi#	User Input  Demand Mode DMA Acknowledge  Local Lock Input	1	I  O TS 12 mA  I	159	C7	Multiplexed input/output pin. USERi: General-purpose input that can be read by way of the PCI 9054 Configuration registers. DACK0#: When a channel is programmed through the Configuration registers to operate in Demand mode, this output indicates a DMA transfer is being executed. DACK0# corresponds to PCI 9054 DMA Ch 0. LLOCKi#: Indicates an atomic operation that may require multiple transactions to complete. Used by the PCI 9054 for direct Local access to the PCI Bus.															
USERo/  DREQ0#/  LLOCKo#	User Output  Demand DMA Request  Local Lock Output	1	O TS 12 mA  I  O	158	A7	Multiplexed input/output pin. USERo: General-purpose output controlled from the PCI 9054 Configuration registers. DREQ0#: When a channel is programmed through the Configuration registers to operate in Demand mode, this input serves as a DMA request. DREQ0# corresponds to the PCI 9054 DMA Ch 0. LLOCKo#: Indicates an atomic operation for a PCI Target PCI-to-Local Bus access may require multiple transactions to complete.															

### 12.3 M BUS MODE PINOUT

Table 12-7. M Bus Mode Interface Pins

M Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
BB#	Bus Busy	1	I/O OC 12 mA	150	C9	As an input, monitors this signal to determine whether the external Master has ended a Bus cycle. As an output, asserts this signal after an external arbiter has granted ownership of the Local Bus and BB# is inactive from another Master. For PCI 9054_AA revision, Signal requires an external pull-up resistor value of 510 ohms be applied to guarantee a fast transition to the inactive state when the PCI 9054_AA revision relinquishes ownership of the Local Bus. For the PCI 9054_AB revision, the pull-up resistor value must be 4.7K ohms or higher.
BDIP#	Burst Data in Progress	1	I/O TS 12 mA	151	B9	As an input, driven by the Bus Master during a Burst transaction. The Master de-asserts before the last Data phase on the bus. As an output, driven by the PCI 9054 during the Data phase of a Burst transaction. The PCI 9054 de-asserts before the last Burst Data phase on the bus.
BG#	Bus Grant	1	I	144	B11	Asserted by the Local Bus arbiter in response to BR#. Indicates the requesting Master is next.
BI#	Burst Inhibit	1	I	134	D12	Whenever BR# is asserted, indicates that the Target device does not support Burst transactions.
BIGEND#/  WAIT#	Big Endian Select  WAIT Input/Output Select ( <i>WAIT# is available at this location only in M mode</i> )	1	I  I/O TS 12 mA	163	C6	Multiplexed input/output pin. Can be asserted during the Local Bus Address phase of a PCI Initiator transfer or Configuration register access to specify use of Big Endian Byte ordering. Big Endian Byte order for PCI Initiator transfers or Configuration register accesses is also programmable through the Configuration registers. If wait is selected, then PCI 9054 issues WAIT# when it is a Master on the Local Bus and has internal wait states setup. As a Slave, the PCI 9054 accepts WAIT# as an input from the Bus Master.
BR#	Bus Request	1	O TP 12 mA	143	A12	Asserted by the Master to request use of the Local Bus. The Local Bus arbiter asserts BG# when the Master is next in line for bus ownership.
BURST#	Burst	1	I/O TS 12 mA	148	B10	As an input, driven by the Master along with address and data indicating a Burst transfer is in progress. As an output, driven by the PCI 9054 along with address and data indicating a Burst transfer is in progress.
DP[0:3]	Data Parity	4	I/O TS 12 mA	136-139	B13, E11, C12, A13	Parity is even for each of up to four byte lanes on the Local Bus. Parity is checked for writes or reads to the PCI 9054. Parity is asserted for reads from or writes by the PCI 9054. DP0 is the most significant bit of the Bus address.

Table 12-7. M Bus Mode Interface Pins (Continued)

M Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
LA[0:31]	Address Bus	32	I/O TS 12 mA	54-60, 63-68, 71-87, 93-94	R4, N5, L6, M6, N6, P6, M7, R7, L7, N8, R8, M8, P9, N9, L9, R10, N10, K9, R11, M10, R12, L10, M11, R13, N12, P13, K10, R14, N13, P14, L11, M13	Carries the 32 bits of the physical Address Bus. LA0 is most significant bit of bus address.
LD[0:31]	Data Bus	32	I/O TS 12 mA	95-98, 100-107, 110-114, 117-131	N15, M14, J10, M15, L14, L15, K12, K13, K15, J12, J14, J15, H14, H15, H11, G14, G15, G11, F14, F13, G10, E14, E13, D15, D14, E12, C15, D13, C14, F10, B15, C13	Carries 8-, 16-, or 32-bit data quantities, depending upon the bus-width configuration. All Master accesses to the PCI 9054 are 32 bits only. LD0 is most significant bit of bus address.
MDREQ#/  DMPAF/  EOT#	IDMA Data Transfer Request ( <i>MDREQ# is available at this location in M mode only</i> )  PCI Initiator Programmable Almost Full  End of Transfer for Current DMA Channel	1	O TS 12 mA  O TS 12 mA  I	153	E9	Multiplexed input or output pin.  MDREQ#: IDMA M mode Data transfer request start. Always asserted, indicating Data transfer should start. De-asserted only when the PCI Initiator FIFO becomes full. Programmable through a Configuration register.  DMPAF: PCI Initiator Write FIFO Almost Full status output. Programmable through a Configuration register.  EOT#: Terminates the current DMA transfer.  <b>Note:</b> EOT# serves as a general purpose EOT. Before asserting EOT#, user should be aware of DMA channel activity.
RD/WR#	Read/Write	1	I/O TS 12 mA	90	M12	Asserted high for reads and low for writes.
RETRY#	Retry	1	O TP 12 mA	149	A10	Driven by the PCI 9054 when it is a Slave to indicate a Local Master must back off and restart the cycle. In Deferred Read mode, indicates a Local Master should return for the requested data.

Table 12-7. M Bus Mode Interface Pins (Continued)

M Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
TA#	Transfer Acknowledge	1	I/O DTS 12 mA	135	A14	As an input, when a Local Bus access is made to the PCI 9054, indicates a Write Data transfer can complete or that Read data on the bus is valid. As an output, when the PCI 9054 is a Bus Master, indicates a Write Data transfer is complete or that Read data on the bus is valid.
TEA#	Transfer Error Acknowledge	1	I/O OC 12 mA	146	D10	Driven by the Target device, indicating an error condition occurred during a Bus cycle.
TS#	Address Strobe	1	I/O TS 12 mA	145	A11	Indicates the valid address and start of a new Bus access. Asserted for the first clock of a Bus access.
TSIZ[0:1]	Transfer Size	2	I/O TS 12 mA	92-91	N14, P15	Driven by the current Master along with the address, indicating the data-transfer size. TSIZ0 is most significant bit of bus address. Refer to Section 3.4.3.5 for further information.

## 12.4 C BUS MODE PINOUT

Table 12-8. C Bus Mode Interface Pins

C Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
ADS#	Address Strobe	1	I/O TS 12 mA	145	A11	Indicates the valid address and start of a new Bus access. Asserted for the first clock of a Bus access.
BIGEND#	Big Endian Select	1	I	163	C6	Can be asserted during the Local Bus Address phase of a PCI Initiator transfer or Configuration register access to specify use of Big Endian Byte ordering. Big Endian Byte order for PCI Initiator transfers or Configuration register accesses is also programmable through Configuration registers.
BLAST#	Burst Last	1	I/O TS 12 mA	148	B10	Driven by the current Local Bus Master to indicate the last transfer in a Bus access.
BREQi	Bus Request	1	I	150	C9	Asserted to indicate the Local Bus Master requires the bus. If enabled through the PCI 9054 Configuration registers, the PCI 9054 releases the bus during a DMA transfer if this signal is asserted.
BREQo	Bus Request Out	1	O TP 12 mA	149	A10	Asserted to indicate the PCI 9054 requires the bus to perform a PCI Target PCI-to-Local Bus access while a PCI Initiator access is pending on the Local Bus. Can be used with external logic to assert backoff to a Local Bus Master. Operational parameters are set up by way of the PCI 9054 Configuration registers.
BTERM#	Burst Terminate	1	I/O DTS 12 mA	134	D12	As input to the PCI 9054: For processors that burst up to four Lwords. If the Bterm Mode bit is disabled through the PCI 9054 Configuration registers, the PCI 9054 also bursts up to four Lwords. If enabled, the PCI 9054 continues to burst until BTERM# input is asserted. BTERM# is a Ready input that breaks up a Burst cycle and causes another Address cycle to occur. Used in conjunction with the PCI 9054 programmable wait state generator. As output from the PCI 9054: Asserted, along with READY#, to request break up of a burst and start of a new Address cycle (PCI Aborts only).
DMPAF/ EOT#	PCI Initiator Programmable Almost Full End of Transfer for Current DMA Channel	1	O TS 12 mA I	153	E9	Multiplexed input or output pin. DMPAF: PCI Initiator Write FIFO Almost Full status output. Programmable through a Configuration register. EOT#: Terminates the current DMA transfer. <b>Note:</b> EOT# serves as a general purpose EOT. Before asserting EOT, user should be aware of DMA channel activity.
DP[3:0]	Data Parity	4	I/O TS 12 mA	136-139	B13, E11, C12, A13	Parity is even for each of up to four byte lanes on the Local Bus. Parity is checked for writes or reads to the PCI 9054. Parity is asserted for reads from or writes by the PCI 9054.

Table 12-8. C Bus Mode Interface Pins (Continued)

C Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
LA[31:2]	Address Bus	30	I/O TS 12 mA	54-60, 63-68, 71-87	R4, N5, L6, M6, N6, P6, M7, R7, L7, N8, R8, M8, P9, N9, L9, R10, N10, K9, R11, M10, R12, L10, M11, R13, N12, P13, K10, R14, N13, P14	Carries the upper 30 bits of the physical Address Bus. During bursts, LA[31:2] increment to indicate successive Data cycles.
LBE[3:0]#	Byte Enables	4	I/O TS 12 mA	92, 91, 93, 94	N14, P15, L11, M13	Encoded, based on the bus-width configuration, as follows: <b>32-Bit Bus</b> The four byte enables indicate which of the four bytes are active during a Data cycle: BE3# Byte Enable 3—LD[31:24] BE2# Byte Enable 2—LD[23:16] BE1# Byte Enable 1—LD[15:8] BE0# Byte Enable 0—LD[7:0] <b>16-Bit Bus</b> BE3#, BE1# and BE0# are encoded to provide BHE#, LA1, and BLE#, respectively: BE3# Byte High Enable (BHE#)—LD[15:8] BE2# not used BE1# Address bit 1 (LA1) BE0# Byte Low Enable (BLE#)—LD[7:0] <b>8-Bit Bus</b> BE1# and BE0# are encoded to provide LA1 and LA0, respectively: BE3# not used BE2# not used BE1# Address bit 1 (LA1) BE0# Address bit 0 (LA0)
LD[31:0]	Data Bus	32	I/O TS 12 mA	95-98, 100-107, 110-114, 117-131	N15, M14, J10, M15, L14, L15, K12, K13, K15, J12, J14, J15, H14, H15, H11, G14, G15, G11, F14, F13, G10, E14, E13, D15, D14, E12, C15, D13, C14, F10, B15, C13	Carries 8-, 16-, or 32-bit data quantities, depending upon a Target bus-width configuration. All Master accesses to the PCI 9054 are 32 bits only.
LHOLD	Hold Request	1	O TP 12 mA	143	A12	Asserted to request use of the Local Bus. The Local Bus arbiter asserts LHOLDA when control is granted.

Table 12-8. C Bus Mode Interface Pins (Continued)

C Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
LHOLDA	Hold Acknowledge	1	I	144	B11	Asserted by the Local Bus arbiter when control is granted in response to LHOLD. Bus should not be granted to the PCI 9054 unless requested by LHOLD.
LSERR#	System Error Interrupt Output	1	O OC 12 mA	146	D10	Synchronous level output asserted PCI Bus Target Abort bit is set (PCISR[11]=1) or Received Master Abort bit is set (PCISR[13]=1). If edge level interrupt is required, disabling and then enabling LSERR# interrupts through the interrupt/control status creates an edge if an interrupt condition still exists or a new interrupt condition occurs.
LW/R#	Write/Read	1	I/O TS 12 mA	90	M12	Asserted low for reads and high for writes.
READY#	Ready Input/Output	1	I/O DTS 12 mA	135	A14	When the PCI 9054 is a Bus Master, indicates Read data on the bus is valid or a Write Data transfer is complete. Used in conjunction with the PCI 9054 programmable wait state generator. When a Local Bus access is made to the PCI 9054, indicates the Bus Read data is valid or a Write Data transfer is complete.
WAIT#	Wait Input/Output	1	I/O TS 12 mA	151	B9	As an input, asserted to cause the PCI 9054 to insert wait states for Local PCI Initiator accesses to the PCI Bus. Can be thought of as a Ready input from an external Master for PCI Initiator accesses. As an output, asserted by the PCI 9054 when the internal wait state generator causes wait states. Can be thought of as an output providing PCI 9054 Ready status.

## 12.5 J BUS MODE PINOUT

Table 12-9. J Bus Mode Interface Pins

J Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
ADS#	Address Strobe	1	I/O TS 12 mA	145	A11	Indicates the valid address and start of a new Bus access. Asserted for the first clock of a Bus access.
ALE	Address Latch Enable	1	I/O TS 12 mA	56	L6	Asserted during the Address phase and de-asserted before the Data phase and before the next LCLK rising edge.
BIGEND#	Big Endian Select	1	I	163	C6	Can be asserted during the Local Bus Address phase of a PCI Initiator transfer or Configuration register access to specify use of Big Endian Byte ordering. Big Endian Byte order for PCI Initiator transfers or Configuration register accesses is also programmable through the Configuration registers.
BLAST#	Burst Last	1	I/O TS 12 mA	148	B10	Driven by the current Local Bus Master to indicate the last transfer in a Bus access.
BREQi	Bus Request	1	I	150	C9	Asserted to indicate the Local Bus Master requires the bus. If enabled through the PCI 9054 Configuration registers, the PCI 9054 releases the bus during a DMA transfer if this signal is asserted.
BREQo	Bus Request Out	1	O TP 12 mA	149	A10	Asserted to indicate the PCI 9054 requires the bus to perform a PCI Target PCI-to-Local Bus access while a PCI Initiator access is pending on the Local Bus. Can be used with external logic to assert backoff to a Local Bus Master. Operational parameters are set up through the PCI 9054 Configuration registers.
BTERM#	Burst Terminate	1	I/O DTS 12 mA	134	D12	As input for processors that burst up to four Lwords. If the Bterm Mode bit is disabled through the PCI 9054 Configuration registers, the PCI 9054 also bursts up to four Lwords. If enabled, the PCI 9054 continues to burst until a BTERM# input is asserted. BTERM# is a Ready input that breaks up a Burst cycle and causes another Address cycle to occur. Used in conjunction with the PCI 9054 programmable wait state generator. As output, asserted along with READY# to request breakup of a burst and start of a new Address cycle (PCI Aborts only).
DEN#	Data Enable	1	O TS 12 mA	55	N5	Used in conjunction with DT/R# to provide control for data transceivers attached to the Local Bus.
DMPAF/ EOT#	PCI Initiator Programmable Almost Full End of Transfer for Current DMA Channel	1	O TS 12 mA I	153	E9	Multiplexed input or output pin. DMPAF: PCI Initiator Write FIFO Almost Full status output. Programmable through a Configuration register. EOT#: Terminates the current DMA transfer. <b>Note:</b> EOT# serves as a general purpose EOT. Before asserting EOT#, user should be aware of DMA channel activity.

Table 12-9. J Bus Mode Interface Pins (Continued)

J Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
DP[3:0]	Data Parity	4	I/O TS 12 mA	136-139	B13, E11, C12, A13	Parity is even for each of up to four byte lanes on the Local Bus. Parity is checked for writes or reads to the PCI 9054. Parity is asserted for reads from or writes by the PCI 9054.
DT/R#	Data Transmit/Receive	1	O TS 12 mA	54	R4	Used in conjunction with DEN# to provide control for data transceivers attached to the Local Bus. When asserted, indicates the PCI 9054 receives data.
LA[28:2]	Local Address Bus	27	I/O TS 12 mA	57-60, 63-68, 71-87	M6, N6, P6, M7, R7, L7, N8, R8, M8, P9, N9, L9, R10, N10, K9, R11, M10, R12, L10, M11, R13, N12, P13, K10, R14, N13, P14	Carries the upper 27 bits of the physical Address Bus. During bursts, it is incremented to indicate successive Data cycles. The lowest two bits, LA[3:2], carry the word address of the 32-bit Memory Address. The bits are incremented during a Burst access.
LAD[31:0]	Address/Data Bus	32	I/O TS 12 mA	95-98, 100-107, 110-114, 117-131	N15, M14, J10, M15, L14, L15, K12, K13, K15, J12, J14, J15, H14, H15, H11, G14, G15, G11, F14, F13, G10, E14, E13, D15, D14, E12, C15, D13, C14, F10, B15, C13	During an Address phase, the bus carries the upper 30 bits of the physical Address Bus. During a Data phase, the bus carries 32 bits of data.

Table 12-9. J Bus Mode Interface Pins (Continued)

J Bus Mode Symbol	Signal Name	Total Pins	Pin Type	PQFP Pin Number	PBGA Pin Number	Function
LBE[3:0]#	Byte Enables	4	I/O TS 12 mA	92, 91, 93, 94	N14, P15, L11, M13	Encoded, based on the bus-width configuration, as follows: <b>32-Bit Bus</b> The four byte enables indicate which of the four bytes are active during a Data cycle: BE3# Byte Enable 3—LAD[31:24] BE2# Byte Enable 2—LAD[23:16] BE1# Byte Enable 1—LAD[15:8] BE0# Byte Enable 0—LAD[7:0] <b>16-Bit Bus</b> BE3#, BE1# and BE0# are encoded to provide BHE#, LAD1, and BLE#, respectively: BE3# Byte High Enable (BHE#)—LAD[15:8] BE2# not used BE1# Address bit 1 (LAD1) BE0# Byte Low Enable (BLE#)—LAD[7:0] <b>8-Bit Bus</b> BE1# and BE0# are encoded to provide LAD1 and LAD0, respectively: BE3# not used BE2# not used BE1# Address bit 1 (LAD1) BE0# Address bit 0 (LAD0)
LHOLD	Hold Request	1	O TP 12 mA	143	A12	Asserted to request use of the Local Bus. The Local Bus arbiter asserts LHOLDA when control is granted.
LHOLDA	Hold Acknowledge	1	I	144	B11	Asserted by the Local Bus arbiter when control is granted in response to LHOLD. The bus should not be granted to the PCI 9054 unless requested by LHOLD.
LW/R#	Write/Read	1	I/O TS 12 mA	90	M12	Asserted low for reads and high for writes.
LSERR#	System Error Interrupt Output	1	O OC 12 mA	146	D10	Synchronous level output asserted the PCI Bus Target Abort bit is set (PCISR[11]=1) or the Received Master Abort bit is set (PCISR[13]=1). If the edge level interrupt is required, disabling and then enabling LSERR# interrupts through the interrupt/control status creates an edge if an interrupt condition still exists or a new interrupt condition occurs.
READY#	Ready Input/Output	1	I/O DTS 12 mA	135	A14	When the PCI 9054 is Bus Master, indicates the Read data on the bus is valid or a Write Data transfer is complete. Used in conjunction with the PCI 9054 programmable wait state generator. When a Local Bus access is made to the PCI 9054, indicates the bus Read data is valid or a Write Data transfer is complete.
WAIT#	Wait Input/Output	1	I/O TS 12 mA	151	B9	As an input, asserted to cause the PCI 9054 to insert wait states for Local PCI Initiator accesses to the PCI Bus. Can be thought of as a Ready input from an external Master for PCI Initiator accesses. As an output, asserted by the PCI 9054 when the internal wait state generator causes wait states. Can be thought of as an output providing PCI 9054 Ready status.

## 12.6 NANDTREE TEST ACCESS METHOD

Manufacturing Test (the proper soldering of components to a board) is the first step toward a successful product release. Some silicon vendors provide Joint Test Action Group (JTAG), while other silicon vendors provide an alternative method, NANDTREE, to achieve the same goal. NANDTREE, as well JTAG, provide users with a methodology to verify connectivity between the board and the components.

NANDTREE, the method used by PLX, is a serial interconnection of NAND gates that are used only for testing shorts, opens, and bridging faults of installed devices on a board.

JTAG, by comparison, is a boundary-scan test, or Design for Test (DFT) technique, that simplifies printed circuit board testing, using a standard chip-board test interface. This standard is known as the *IEEE Standard Test Access Port and Boundary Scan Architecture*. JTAG is a group that initiated the standardization of this test interface.

The PCI 9054 supports the NANDTREE method for this verification methodology. The PCI 9054 NANDTREE is an internal, serial interconnection of all functional pins to provide a convenient method for board manufacturers to check connectivity of package pins to the board. To perform a NANDTREE check, all NANDTREE-connected pins must be forced high "1", as well as a TEST pin, to enable NANDTREE functionality.

Table 12-10. Sequential Interconnection of the PCI 9054 NANDTREE (Pin Definition)

Component Name	Input #1	Input #2	Output
IN2_INV1	(.A(LA[31]),	—	.X(NAN0));
NA2_NAND1	(.A1(LA[30]),	.A2(NAN0),	.X(NAN1));
NA2_NAND2	(.A1(LA[29]),	.A2(NAN1),	.X(NAN2));
NA2_NAND3	(.A1(LA[28]),	.A2(NAN2),	.X(NAN3));
NA2_NAND4	(.A1(LA[27]),	.A2(NAN3),	.X(NAN4));
NA2_NAND5	(.A1(LA[26]),	.A2(NAN4),	.X(NAN5));
NA2_NAND6	(.A1(LA[25]),	.A2(NAN5),	.X(NAN6));
NA2_NAND7	(.A1(LA[24]),	.A2(NAN6),	.X(NAN7));
NA2_NAND8	(.A1(LA[23]),	.A2(NAN7),	.X(NAN8));
NA2_NAND9	(.A1(LA[22]),	.A2(NAN8),	.X(NAN9));
NA2_NAND10	(.A1(LA[21]),	.A2(NAN9),	.X(NAN10));
NA2_NAND11	(.A1(LA[20]),	.A2(NAN10),	.X(NAN11));
NA2_NAND12	(.A1(LA[19]),	.A2(NAN11),	.X(NAN12));
NA2_NAND13	(.A1(LA[18]),	.A2(NAN12),	.X(NAN13));
NA2_NAND14	(.A1(LA[17]),	.A2(NAN13),	.X(NAN14));
NA2_NAND15	(.A1(LA[16]),	.A2(NAN14),	.X(NAN15));
NA2_NAND16	(.A1(LA[15]),	.A2(NAN15),	.X(NAN16));
NA2_NAND17	(.A1(LA[14]),	.A2(NAN16),	.X(NAN17));
NA2_NAND18	(.A1(LA[13]),	.A2(NAN17),	.X(NAN18));
NA2_NAND19	(.A1(LA[12]),	.A2(NAN18),	.X(NAN19));
NA2_NAND20	(.A1(LA[11]),	.A2(NAN19),	.X(NAN20));
NA2_NAND21	(.A1(LA[10]),	.A2(NAN20),	.X(NAN21));
NA2_NAND22	(.A1(LA[9]),	.A2(NAN21),	.X(NAN22));
NA2_NAND23	(.A1(LA[8]),	.A2(NAN22),	.X(NAN23));
NA2_NAND24	(.A1(LA[7]),	.A2(NAN23),	.X(NAN24));
NA2_NAND25	(.A1(LA[6]),	.A2(NAN24),	.X(NAN25));
NA2_NAND26	(.A1(LA[5]),	.A2(NAN25),	.X(NAN26));
NA2_NAND27	(.A1(LA[4]),	.A2(NAN26),	.X(NAN27));
NA2_NAND28	(.A1(LA[3]),	.A2(NAN27),	.X(NAN28));
NA2_NAND29	(.A1(LA[2]),	.A2(NAN28),	.X(NAN29));
NA2_NAND30	(.A1(LBE[1]#/LA[1]),	.A2(NAN29),	.X(NAN30));
NA2_NAND31	(.A1(LBE[0]#/LA[0]),	.A2(NAN30),	.X(NAN31));
NA2_NAND32	(.A1(LAD[31]/LD[31]),	.A2(NAN31),	.X(NAN32));
NA2_NAND33	(.A1(LAD[30]/LD[30]),	.A2(NAN32),	.X(NAN33));
NA2_NAND34	(.A1(LAD[29]/LD[29]),	.A2(NAN33),	.X(NAN34));
NA2_NAND35	(.A1(LAD[28]/LD[28]),	.A2(NAN34),	.X(NAN35));
NA2_NAND36	(.A1(LAD[27]/LD[27]),	.A2(NAN35),	.X(NAN36));
NA2_NAND37	(.A1(LAD[26]/LD[26]),	.A2(NAN36),	.X(NAN37));
NA2_NAND38	(.A1(LAD[25]/LD[25]),	.A2(NAN37),	.X(NAN38));
NA2_NAND39	(.A1(LAD[24]/LD[24]),	.A2(NAN38),	.X(NAN39));
NA2_NAND40	(.A1(LAD[23]/LD[23]),	.A2(NAN39),	.X(NAN40));
NA2_NAND41	(.A1(LAD[22]/LD[22]),	.A2(NAN40),	.X(NAN41));
NA2_NAND42	(.A1(LAD[21]/LD[21]),	.A2(NAN41),	.X(NAN42));

Table 12-10. Sequential Interconnection of the PCI 9054 NANDTREE (Pin Definition) (Continued)

Component Name	Input #1	Input #2	Output
NA2_NAND43	(.A1(LAD[20]/LD[20]),	.A2(NAN42),	.X(NAN43));
NA2_NAND44	(.A1(LAD[19]/LD[19]),	.A2(NAN43),	.X(NAN44));
NA2_NAND45	(.A1(LAD[18]/LD[18]),	.A2(NAN44),	.X(NAN45));
NA2_NAND46	(.A1(LAD[17]/LD[17]),	.A2(NAN45),	.X(NAN46));
NA2_NAND47	(.A1(LAD[16]/LD[16]),	.A2(NAN46),	.X(NAN47));
NA2_NAND48	(.A1(LAD[15]/LD[15]),	.A2(NAN47),	.X(NAN48));
NA2_NAND49	(.A1(LAD[14]/LD[14]),	.A2(NAN48),	.X(NAN49));
NA2_NAND50	(.A1(LAD[13]/LD[13]),	.A2(NAN49),	.X(NAN50));
NA2_NAND51	(.A1(LAD[12]/LD[12]),	.A2(NAN50),	.X(NAN51));
NA2_NAND52	(.A1(LAD[11]/LD[11]),	.A2(NAN51),	.X(NAN52));
NA2_NAND53	(.A1(LAD[10]/LD[10]),	.A2(NAN52),	.X(NAN53));
NA2_NAND54	(.A1(LAD[9]/LD[9]),	.A2(NAN53),	.X(NAN54));
NA2_NAND55	(.A1(LAD[8]/LD[8]),	.A2(NAN54),	.X(NAN55));
NA2_NAND56	(.A1(LAD[7]/LD[7]),	.A2(NAN55),	.X(NAN56));
NA2_NAND57	(.A1(LAD[6]/LD[6]),	.A2(NAN56),	.X(NAN57));
NA2_NAND58	(.A1(LAD[5]/LD[5]),	.A2(NAN57),	.X(NAN58));
NA2_NAND59	(.A1(LAD[4]/LD[4]),	.A2(NAN58),	.X(NAN59));
NA2_NAND60	(.A1(LAD[3]/LD[3]),	.A2(NAN59),	.X(NAN60));
NA2_NAND61	(.A1(LAD[2]/LD[2]),	.A2(NAN60),	.X(NAN61));
NA2_NAND62	(.A1(LAD[1]/LD[1]),	.A2(NAN61),	.X(NAN62));
NA2_NAND63	(.A1(LAD[0]/LD[0]),	.A2(NAN62),	.X(NAN63));
NA2_NAND64	(.A1(AD[31]),	.A2(NAN63),	.X(NAN64));
NA2_NAND65	(.A1(AD[30]),	.A2(NAN64),	.X(NAN65));
NA2_NAND66	(.A1(AD[29]),	.A2(NAN65),	.X(NAN66));
NA2_NAND67	(.A1(AD[28]),	.A2(NAN66),	.X(NAN67));
NA2_NAND68	(.A1(AD[27]),	.A2(NAN67),	.X(NAN68));
NA2_NAND69	(.A1(AD[26]),	.A2(NAN68),	.X(NAN69));
NA2_NAND70	(.A1(AD[25]),	.A2(NAN69),	.X(NAN70));
NA2_NAND71	(.A1(AD[24]),	.A2(NAN70),	.X(NAN71));
NA2_NAND72	(.A1(AD[23]),	.A2(NAN71),	.X(NAN72));
NA2_NAND73	(.A1(AD[22]),	.A2(NAN72),	.X(NAN73));
NA2_NAND74	(.A1(AD[21]),	.A2(NAN73),	.X(NAN74));
NA2_NAND75	(.A1(AD[20]),	.A2(NAN74),	.X(NAN75));
NA2_NAND76	(.A1(AD[19]),	.A2(NAN75),	.X(NAN76));
NA2_NAND77	(.A1(AD[18]),	.A2(NAN76),	.X(NAN77));
NA2_NAND78	(.A1(AD[17]),	.A2(NAN77),	.X(NAN78));
NA2_NAND79	(.A1(AD[16]),	.A2(NAN78),	.X(NAN79));
NA2_NAND80	(.A1(AD[15]),	.A2(NAN79),	.X(NAN80));
NA2_NAND81	(.A1(AD[14]),	.A2(NAN80),	.X(NAN81));
NA2_NAND82	(.A1(AD[13]),	.A2(NAN81),	.X(NAN82));
NA2_NAND83	(.A1(AD[12]),	.A2(NAN82),	.X(NAN83));
NA2_NAND84	(.A1(AD[11]),	.A2(NAN83),	.X(NAN84));
NA2_NAND85	(.A1(AD[10]),	.A2(NAN84),	.X(NAN85));
NA2_NAND86	(.A1(AD[9]),	.A2(NAN85),	.X(NAN86));

Table 12-10. Sequential Interconnection of the PCI 9054 NANDTREE (Pin Definition) (Continued)

Component Name	Input #1	Input #2	Output
NA2_NAND87	(.A1(AD[8]),	.A2(NAN86),	.X(NAN87));
NA2_NAND88	(.A1(AD[7]),	.A2(NAN87),	.X(NAN88));
NA2_NAND89	(.A1(AD[6]),	.A2(NAN88),	.X(NAN89));
NA2_NAND90	(.A1(AD[5]),	.A2(NAN89),	.X(NAN90));
NA2_NAND91	(.A1(AD[4]),	.A2(NAN90),	.X(NAN91));
NA2_NAND92	(.A1(AD[3]),	.A2(NAN91),	.X(NAN92));
NA2_NAND93	(.A1(AD[2]),	.A2(NAN92),	.X(NAN93));
NA2_NAND94	(.A1(AD[1]),	.A2(NAN93),	.X(NAN94));
NA2_NAND95	(.A1(AD[0]),	.A2(NAN94),	.X(NAN95));
NA2_NAND96	(.A1(RETRY#/BREQo),	.A2(NAN95),	.X(NAN96));
NA2_NAND97	(.A1(BB#/BREQi),	.A2(NAN96),	.X(NAN97));
NA2_NAND98	(.A1(BI#/BTERM#),	.A2(NAN97),	.X(NAN98));
NA2_NAND99	(.A1(TA#/READY#),	.A2(NAN98),	.X(NAN99));
NA2_NAND100	(.A1(LINT#),	.A2(NAN99),	.X(NAN100));
NA2_NAND101	(.A1(BG#/LHOLDA),	.A2(NAN100),	.X(NAN101));
NA2_NAND102	(.A1(RDWR#/LW/R#),	.A2(NAN101),	.X(NAN102));
NA2_NAND103	(.A1(TS#/ADS#),	.A2(NAN102),	.X(NAN103));
NA2_NAND104	(.A1(BURST#/BLAST#),	.A2(NAN103),	.X(NAN104));
NA2_NAND105	(.A1(USERi/DACK0#/LLOCKi#),	.A2(NAN104),	.X(NAN105));
NA2_NAND106	(.A1(CCS#),	.A2(NAN105),	.X(NAN106));
NA2_NAND107	(.A1(MODE0),	.A2(NAN106),	.X(NAN107));
NA2_NAND108	(.A1(MODE1),	.A2(NAN107),	.X(NAN108));
NA2_NAND109	(.A1(TSIZ[1]/LBE[2]#),	.A2(NAN108),	.X(NAN109));
NA2_NAND110	(.A1(TSIZ[0]/LBE[3]#),	.A2(NAN109),	.X(NAN110));
NA2_NAND111	(.A1(WAIT#/BDIP#),	.A2(NAN110),	.X(NAN111));
NA2_NAND112	(.A1(DP[0]),	.A2(NAN111),	.X(NAN112));
NA2_NAND113	(.A1(DP[1]),	.A2(NAN112),	.X(NAN113));
NA2_NAND114	(.A1(DP[2]),	.A2(NAN113),	.X(NAN114));
NA2_NAND115	(.A1(DP[3]),	.A2(NAN114),	.X(NAN115));
NA2_NAND116	(.A1(MDREQ#/DMPAF/EOT#),	.A2(NAN115),	.X(NAN116));
NA2_NAND117	(.A1(BR#/LHOLD),	.A2(NAN116),	.X(NAN117));
NA2_NAND118	(.A1(LRESETo#),	.A2(NAN117),	.X(NAN118));
NA2_NAND119	(.A1(TEA#/LSERR#),	.A2(NAN118),	.X(NAN119));
NA2_NAND120	(.A1(FRAME#),	.A2(NAN119),	.X(NAN120));
NA2_NAND121	(.A1(IRDY#),	.A2(NAN120),	.X(NAN121));
NA2_NAND122	(.A1(DEVSEL#),	.A2(NAN121),	.X(NAN122));
NA2_NAND123	(.A1(TRDY#),	.A2(NAN122),	.X(NAN123));
NA2_NAND124	(.A1(STOP#),	.A2(NAN123),	.X(NAN124));
NA2_NAND125	(.A1(PAR),	.A2(NAN124),	.X(NAN125));
NA2_NAND126	(.A1(SERR#),	.A2(NAN125),	.X(NAN126));
NA2_NAND127	(.A1(LOCK#),	.A2(NAN126),	.X(NAN127));
NA2_NAND128	(.A1(GNT#),	.A2(NAN127),	.X(NAN128));
NA2_NAND129	(.A1(IDSEL),	.A2(NAN128),	.X(NAN129));
NA2_NAND130	(.A1(CBE[0]#),	.A2(NAN129),	.X(NAN130));

Table 12-10. Sequential Interconnection of the PCI 9054 NANDTREE (Pin Definition) (Continued)

Component Name	Input #1	Input #2	Output
NA2_NAND131	(.A1(CBE[1]#),	.A2(NAN130),	.X(NAN131));
NA2_NAND132	(.A1(CBE[2]#),	.A2(NAN131),	.X(NAN132));
NA2_NAND133	(.A1(CBE[3]#),	.A2(NAN132),	.X(NAN133));
NA2_NAND134	(.A1(REQ#),	.A2(NAN133),	.X(NAN134));
NA2_NAND135	(.A1(RST#),	.A2(NAN134),	.X(NAN135));
NA2_NAND136	(.A1(INTA#),	.A2(NAN135),	.X(NAN136));
NA2_NAND137	(.A1(PME#),	.A2(NAN136),	.X(NAN137));
NA2_NAND138	(.A1(BIGEND#/WAIT#),	.A2(NAN137),	.X(NAN138));
NA2_NAND139	(.A1(PERR#),	.A2(NAN138),	.X(NAN139));
NA2_NAND140	(.A1(E ECS),	.A2(NAN139),	.X(NAN140));
NA2_NAND141	(.A1(EESK),	.A2(NAN140),	.X(NAN141));
NA2_NAND142	(.A1(EEDI/EEDO),	.A2(NAN141),	.X(NAN142));
NA2_NAND143	(.A1(ENUM#),	.A2(NAN142),	.X(USERo/DREQ0#/LLOCKo#));



# 13 ELECTRICAL SPECIFICATIONS

## 13.1 GENERAL ELECTRICAL SPECIFICATIONS

**Table 13-1. Absolute Maximum Ratings**

Specification	Maximum Rating
Storage Temperature	-55 to +125 °C
Ambient Temperature with Power Applied	-40 to +85 °C
Supply Voltage to Ground	-0.5 to +4.6V
Input Voltage (VIN)	VSS -0.5 to 5.5V
Output Voltage (VOUT)	VSS -0.5V to VDD +0.5
Maximum Package Power Dissipation (176-Pin PQFP)	1.0W
Maximum Package Power Dissipation (225-Pin PBGA)	1.0W

**Table 13-2. Operating Ranges**

Ambient Temperature	Supply Voltage (VDD)	Input Voltage (VIN)	
		Min	Max
-40 to +85 °C	3.0 to 3.6V	VSS	VDD

**Table 13-3. Capacitance (Sample Tested Only)**

Parameter	Test Conditions	Pin Type	Value		Units
			Typical	Maximum	
CIN	VIN = 0V	Input	4	6	pF
COUT	VOUT = 0V	Output	6	10	pF

The following table lists the package thermal resistance ( $\Theta_{j-a}$ ).

**Table 13-4. Thermal Resistance of Packages**

Package Type	Air Flow			
	0m	1m	2m	3m
176-Pin PQFP	65 (°C/W)	45	35	30
225-Pin PBGA	72 (°C/W)	46	37	32

Table 13-5. Electrical Characteristics over Operating Range

Parameter	Description	Test Conditions		Min	Max	Units
VOH	Output High Voltage	VDD = Min VIN = VIH or VIL	IOH = -12.0 mA	2.4	—	V
VOL	Output Low Voltage		IOL = 12 mA	—	0.4	V
VIH	Input High Level	—	—	2.0	5.5	V
VIL	Input Low Level	—	—	-0.5	0.8	V
VOH3	PCI 3.3V Output High Voltage	VDD = Min VIN = VIH or VIL	IOH = -500 $\mu$ A	0.9 VDD	—	V
VOL3	PCI 3.3V Output Low Voltage		IOL = 1500 $\mu$ A	—	0.1 VDD	V
VIH3	PCI 3.3V Input High Level	—	—	0.5 VDD	VDD +0.5	V
VIL3	PCI 3.3V Input Low Level	—	—	-0.5	0.3 VDD	V
IIL	Input Leakage Current	VSS $\leq$ VIN $\leq$ VDD, VDD = Max		-10	+10	$\mu$ A
ILPC	DC Current Per Pin During Pre-Charge	VP = 0.8 to 1.2V		—	1.0	mA
IOZ	Tri-State Output Leakage Current	VDD = Max		-10	+10	$\mu$ A
ICC	Power Supply Current <sup>1</sup>	VDD=3.6V, PCLK = 33 MHz, LCLK = 50 MHz 120 outputs switching simultaneously		—	200	mA
ICCL ICCH ICCZ	Quiescent Power Supply Current	VCC = Max VIN = GND or VCC		—	50	$\mu$ A

<sup>1</sup> ILPC is the DC current flowing from VDD to Ground during precharge, as both PMOS and NMOS devices remain on during precharge. It is not the leakage current flowing into or out of the pin under precharge.  
Maximum value based upon 120 simultaneously switching outputs.

### 13.2 LOCAL INPUTS

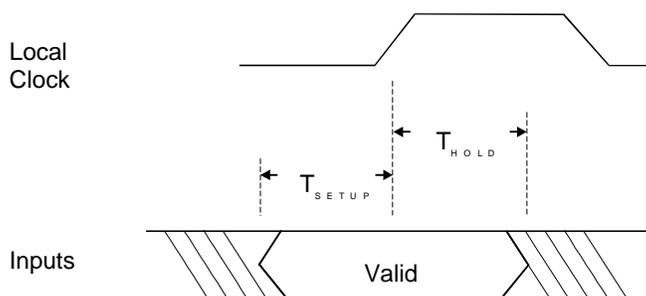


Figure 13-1. PCI 9054 Local Input Setup and Hold Waveform

Table 13-6. AC Electrical Characteristics (Local Inputs) over Operating Range (M Mode)

Signals (Synchronous Inputs) $C_L = 50 \text{ pF}$ , $V_{CC} = 3.0V$ , $T_a = 85^\circ\text{C}$	$T_{SETUP}$ (ns) (WORST CASE)	$T_{HOLD}$ (ns) (WORST CASE)
BB#	5.0	1
BDIP#/WAIT#	<b>6.0</b>	<b>1</b>
BG#	7.0	1
BI#	9.0	1
BIGEND#/WAIT#	7.0	1
BURST#	6.0	1
CCS#	1.5	1
DP[0:3]	3.0	1
LA[0:31]	6.0	1
LD[0:31]	3.5	1
MDREQ#/DMPAF/EOT#	8.5 (EOT#)	1
RD/WR#	6.0	1
TA#	8.0	1
TS#	5.0	1
TSIZ[0:1]	<b>6.0</b>	1
USERi/DACK0#/LLOCKi#	7.0 (LLOCKi#)	1
USERo/DREQ0#/LLOCKo#	7.0 (DREQ0#)	1
Input Clocks	Min	Max
Local Clock Input Frequency	0	50 MHz
PCI Clock Input Frequency	0	33 MHz

Section 13—Electrical Specs

Table 13-7. AC Electrical Characteristics (Local Inputs) over Operating Range (C and J Modes)

Signals (Synchronous Inputs) C <sub>L</sub> = 50 pF, V <sub>cc</sub> = 3.0V, Ta = 85 °C	Bus Mode	T <sub>SETUP</sub> (ns) (WORST CASE)	T <sub>HOLD</sub> (ns) (WORST CASE)
ADS#	C, J	5.0	1
ALE	J	3.5	1
BIGEND#	C, J	7.0	1
BLAST#	C, J	6.5	1
BREQi	C, J	5.0	1
BTERM#	C, J	9.5	1
CCS#	C, J	1.5	1
DMPAF/EOT#	C, J	8.5 (EOT#)	1
DP[3:0]	C, J	3.0	1
LAD[31:0]	J	6.5	1
LBE[3:0]#	C, J	9.0	1
LD[31:0]	C	3.5	1
LHOLDA	C, J	7.0	1
LW/R#	C, J	8.5	1
READY#	C, J	9.5	1
USERi/DACK0#/LLOCKi#	C, J	7.0 (LLOCKi#)	1
USERo/DREQ0#/LLOCKo#	C, J	7.0 (DREQ0#)	1
WAIT#	C, J	7.0	1
Input Clocks	Bus Mode	Min	Max
Local Clock Input Frequency	C, J	0	50 MHz
PCI Clock Input Frequency	C, J	0	33 MHz

## 13.3 LOCAL OUTPUTS

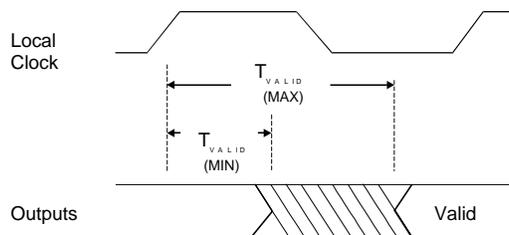


Figure 13-2. PCI 9054 Local Output Delay

Table 13-8. AC Electrical Characteristics (Local Outputs) over Operating Range (M Mode)

Signals (Synchronous Outputs) $C_L = 50 \text{ pF}$ , $V_{CC} = 3.0\text{V}$ , $T_a = 85 \text{ }^\circ\text{C}$	Clock to Out Worst Case (ns) $T_{\text{VALID}} (\text{Max})$
BB#	9.5
BDIP#	10.5
BI#	9.0
BIGEND#/WAIT#	10.5 (WAIT#)
BR#	10.0
BURST#	10.5
DP[0:3]	10.0
LA[0:31]	10.0
LD[0:31]	11.0
MDREQ#/DMPAF/EOT#	13.0 (MDREQ#/DMPAF)
RD/WR#	12.0
RETRY#	9.0
TA#	9.0
TEA#	8.5
TS#	10.0
TSIZ[0:1]	10.0
USERi/DACK0#/LLOCKi#	10.5 (DACK0#)
USERo/DREQ0#/LLOCKo#	9.5 (USERo/LLOCKo#)

**Notes:** All  $T_{\text{VALID}} (\text{Min})$  values are greater than 5 ns.

Timing derating for loading is  $\pm 35 \text{ PS/PF}$ .

Table 13-9. AC Electrical Characteristics (Local Outputs) over Operating Range (C and J Modes)

Signals (Synchronous Outputs) $C_L = 50 \text{ pF}$ , $V_{CC} = 3.0\text{V}$ , $T_a = 85 \text{ }^\circ\text{C}$	Bus Mode	Output $T_{VALID} \text{ (Max)}$
ADS#	C, J	10.0
BLAST#	C, J	12.5
BREQo	C, J	8.5
BTERM#	C, J	10.0
DEN#	J	10.0
DMPAF/EOT#	C, J	13.0 (DMPAF)
DP[3:0]	C, J	10.0
DT/R#	J	12.5
LA[31:2]	C	10.0
LA[28:2]	J	9.5
LAD[31:0]	J	11.0
LBE[3:0]#	C, J	10.0
LD[31:0]	C	11.0
LHOLD	C, J	10.0
LSERR#	C, J	8.5
LW/R#	C, J	12.0
READY#	C, J	9.5
USERi/DACK0#/LLOCKi#	C, J	10.5 (DACK0#)
USERo/DREQ0#/LLOCKo#	C, J	9.5 (USERo/LLOCKo#)
WAIT#	C, J	10.5

**Notes:** All  $T_{VALID} \text{ (Min)}$  values are greater than 5 ns.  
Timing derating for loading is  $\pm 35 \text{ PS/PF}$ .

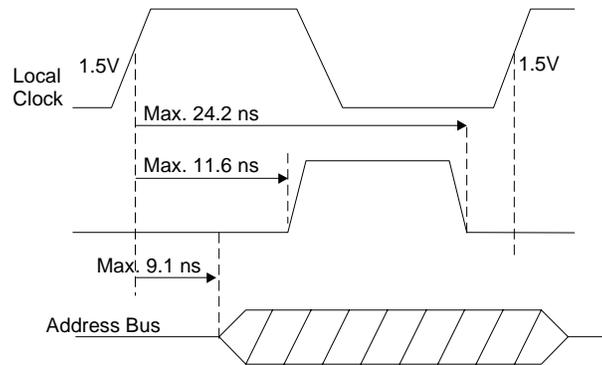


Figure 13-3. PCI 9054 ALE Output Delay to the Local Clock

# 14 PACKAGE, SIGNAL, AND PINOUT SPECS

## 14.1 176-PIN PQFP

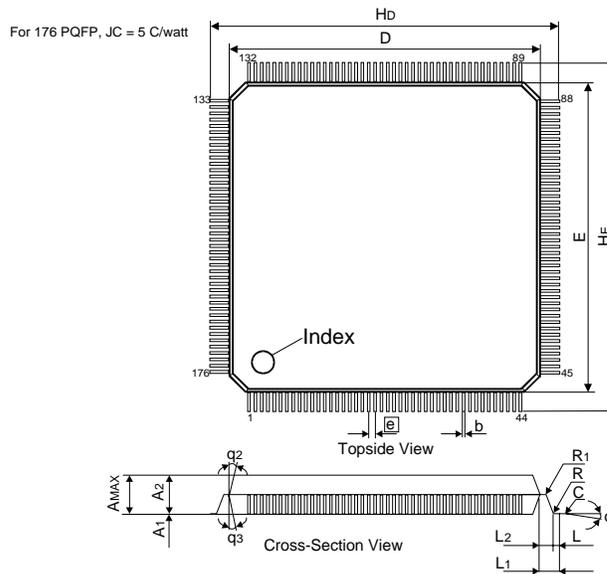
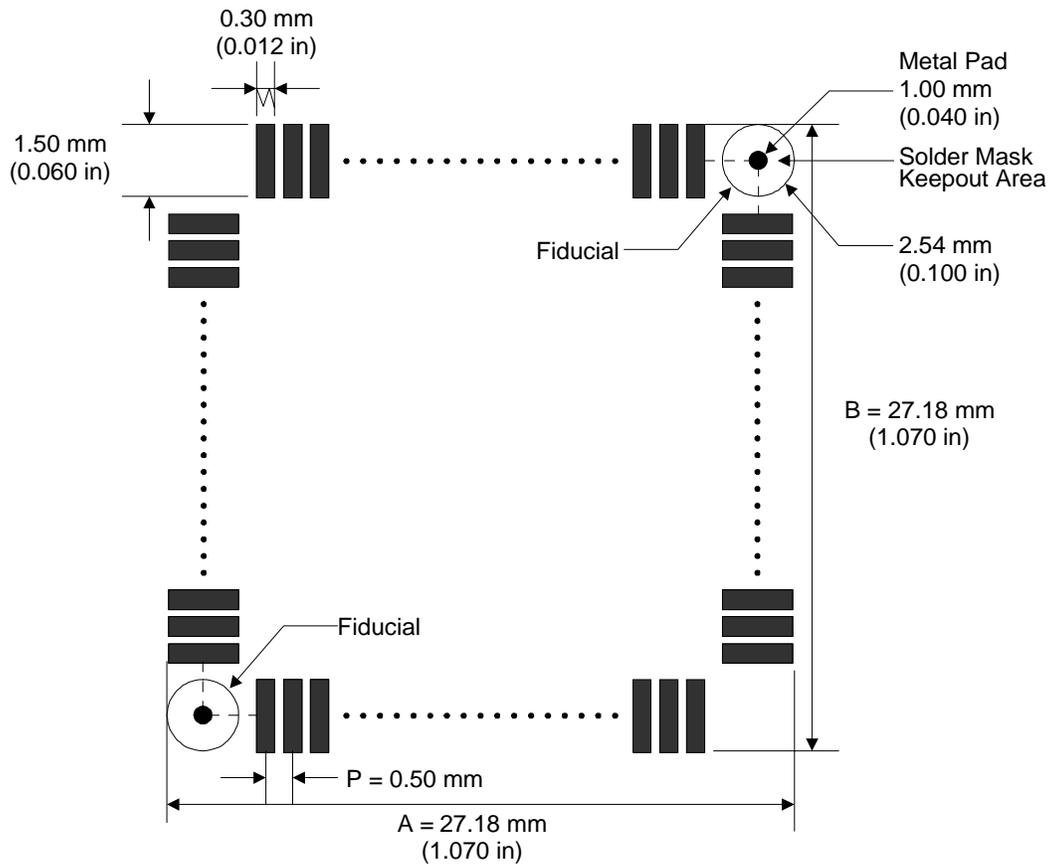


Figure 14-1. 176-Pin PQFP Package Mechanical Dimensions—Topside and Cross-Section Views

Table 14-1. 176-Pin PQFP Package Mechanical Dimensions

Lead Type STD (QfP18-176 Pin STD)			
Symbol	Dimensions (in Millimeters)		
	Min.	Nom.	Max.
E	23.9	24	24.1
D	23.9	24	24.1
A	—	—	3
A1	—	0.1	—
A2	2.6	2.7	2.8
e	—	0.5	—
b	0.15	0.2	0.3
C	0.1	0.15	0.2
q	0°	—	10°
L	0.3	0.5	0.7
L1	—	1	—
L2	—	0.5	—
He	25.6	26	26.4
Hd	25.6	26	26.4
θ2	—	15°	—
θ3	—	15°	—
R	—	0.2	—
R1	—	0.2	—



**Note:** Leave 0.002 inch solder mask clearance around pads.

Figure 14-2. 176-Pin PQFP PCB Layout Suggested Land Pattern



14.2 225-PIN PBGA

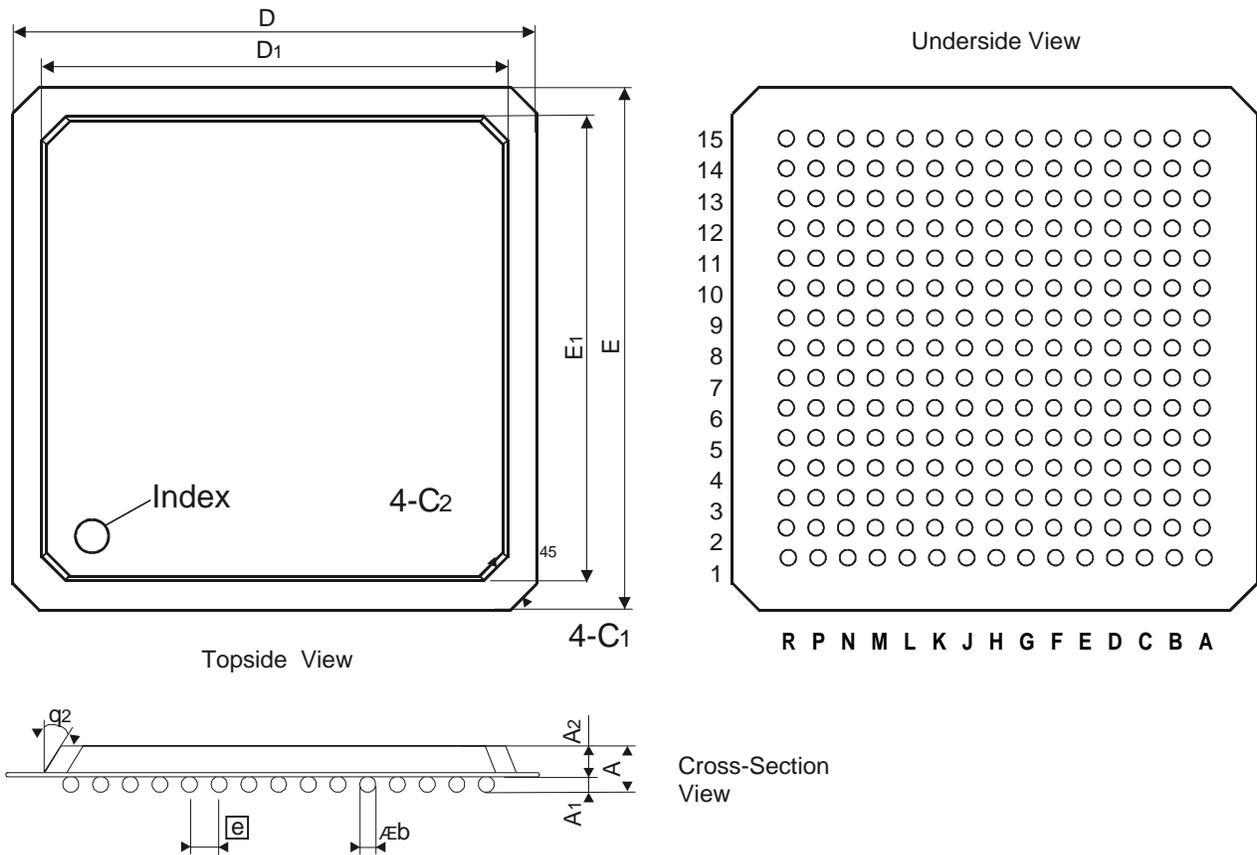


Figure 14-4. 225-Pin PBGA Package Mechanical Dimensions—Topside, Underside, and Cross-Section Views

Table 14-2. 225-Pin PBGA Package Mechanical Dimensions

Symbol	Dimensions (in Millimeters)		
	Min.	Nom.	Max.
$\phi b$	0.6	0.76	0.90
A	2.13	—	—
A1	0.5	0.6	0.7
A2	1.48	1.53	1.85
$\theta 2$	—	30°	—
C1	—	1.0	—
C2	—	4.0	—
e	—	1.5	—
D1	23.5	24	24.7
E1	23.5	24	24.7
D	—	27	—
E	—	27	—

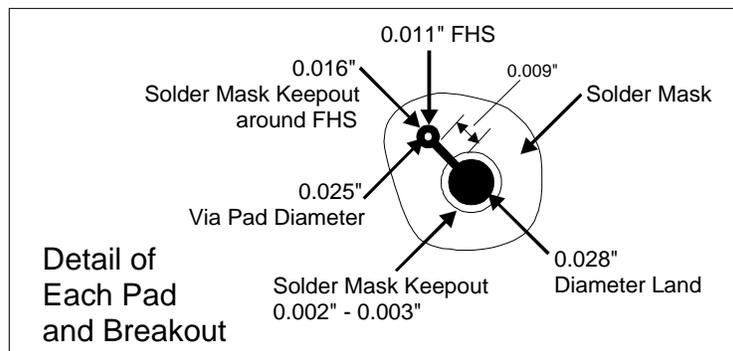
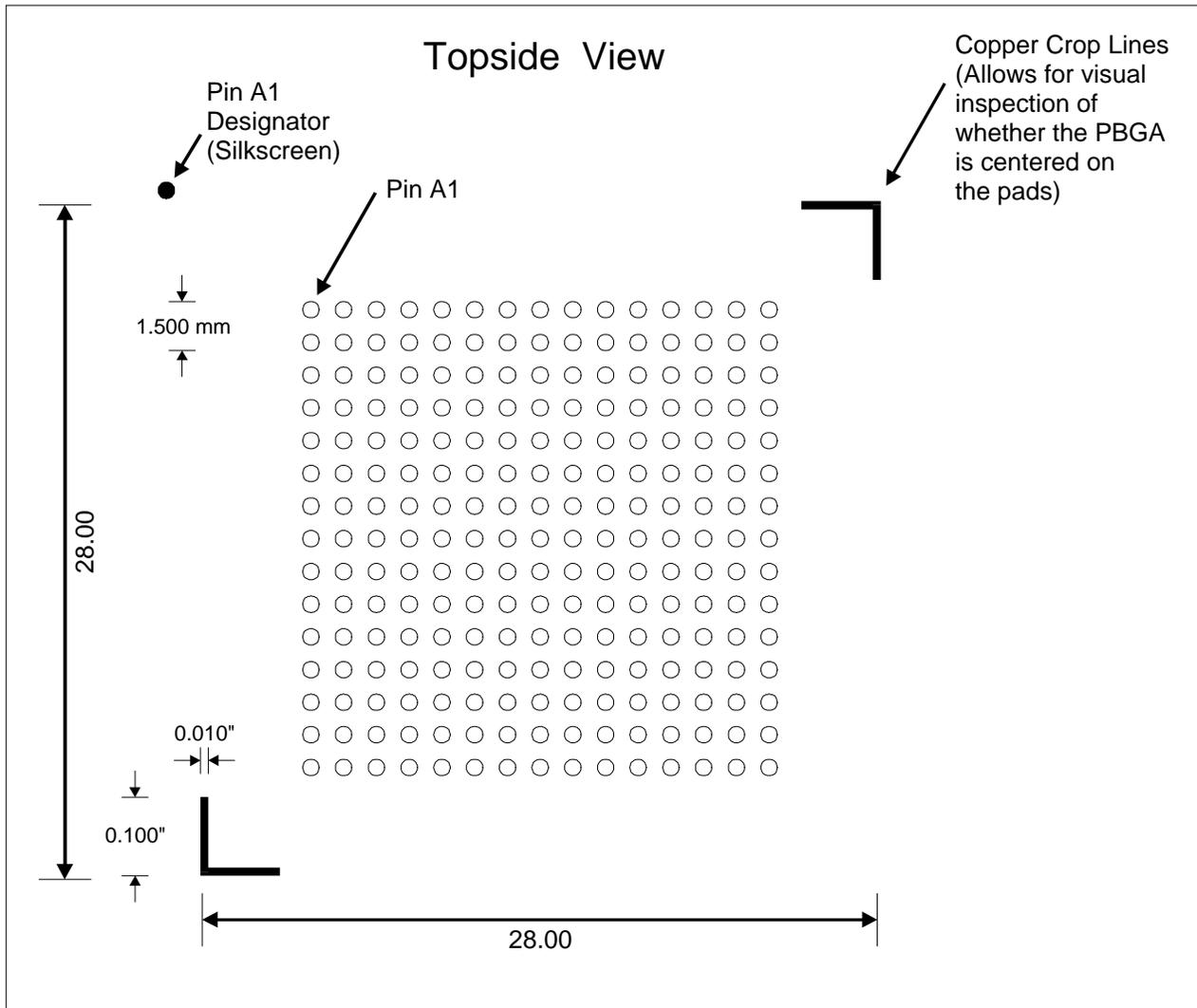


Figure 14-5. 225-Pin PBGA PCB Layout Suggested Land Pattern

G=GND

15	R15	P15	N15	M15	L15	K15	J15	H15	G15	F15	E15	D15	C15	B15	A15
14	R14	P14	N14	M14	L14	K14	J14	H14	G14	F14	E14	D14	C14	B14	A14
13	R13	P13	N13	M13	L13	K13	J13	H13	G13	F13	E13	D13	C13	B13	A13
12	R12	P12	N12	M12	L12	K12	J12	H12	G12	F12	E12	D12	C12	B12	A12
11	R11	P11	N11	M11	L11	K11	J11	H11	G11	F11	E11	D11	C11	B11	A11
10	R10	P10	N10	M10	L10	K10	J10	H10	G10	F10	E10	D10	C10	B10	A10
9	R9	P9	N9	M9	L9	K9	J9	H9	G9	F9	E9	D9	C9	B9	A9
8	R8	P8	N8	M8	L8	K8	J8	H8	G8	F8	E8	D8	C8	B8	A8
7	R7	P7	N7	M7	L7	K7	J7	H7	F7	F7	E7	D7	C7	B7	A7
6	R6	P6	N6	M6	L6	K6	J6	H6	G6	F6	E6	D6	C6	B6	A6
5	R5	P5	N5	M5	L5	K5	J5	H5	G5	F5	E5	D5	C5	B5	A5
4	R4	P4	N4	M4	L4	K4	J4	H4	G4	F4	E4	D4	C4	B4	A4
3	R3	P3	N3	M3	L3	K3	J3	H3	G3	F3	E3	D3	C3	B3	A3
2	R2	P2	N2	M2	L2	K2	J2	H2	G2	F2	E2	D2	C2	B2	A2
1	R1	P1	N1	M1	L1	K1	J1	H1	G1	F1	E1	D1	C1	B1	A1

R	P	N	M	L	K	J	H	G	F	E	D	C	B	A
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

13 Connected Grounds (40 Total Grounds)

28 No Connects (NC)

Figure 14-6. 225-Pin PBGA Package Layout (Underside View)

Table 14-3. 225-Pin PBGA PCI 9054 Pinout

Pin #	Symbol	Bus Mode	Pin #	Symbol	Bus Mode	Pin #	Symbol	Bus Mode	Pin #	Symbol	Bus Mode
A1	VDD (Core)	All	C1	IDSEL	All	E1	NC	All	G1	IRDY#	All
A2	AD30	All	C2	AD26	All	E2	VDD (Core)	All	G2	VDD (Core)	All
A3	PCLK	All	C3	AD29	All	E3	AD21	All	G3	FRAME#	All
A4	PME#	All	C4	GNT#	All	E4	NC	All	G4	C/BE2#	All
A5	NC	All	C5	EEDI/EEDO	All	E5	AD25	All	G5	VDD (Core)	All
A6	VDD (Core)	All	C6	BIGEND#/WAIT# BIGEND#	M C, J	E6	NC	All	G6	AD23	All
A7	USERo/DREQ0#/ LLOCKo#	All	C7	USERi/DACK0#/ LLOCKi#	All	E7	CCS#	All	G7	VSS	All
A8	TEST	All	C8	NC	All	E8	MODE0	All	G8	VSS	All
A9	LRESETo#	All	C9	BB# BREQi	M C, J	E9	MDREQ#/DMPAF/ EOT# DMPAF/EOT#	M C, J	G9	VSS	All
A10	RETRY# BREQo	M C, J	C10	VDD (Core)	All	E10	NC	All	G10	LD20 LD11 LAD11	M C J
A11	TS# ADS#	M C, J	C11	VDD (Core)	All	E11	DP1 DP2	M C, J	G11	LD17 LD14 LAD14	M C J
A12	BR# LHOLD	M C, J	C12	DP2 DP1	M C, J	E12	LD25 LD6 LAD6	M C J	G12	VDD (Core)	All
A13	DP3 DP0	M C, J	C13	LD31 LD0 LAD0	M C J	E13	LD22 LD9 LAD9	M C J	G13	LFRAME#	All
A14	TA# READY#	M C, J	C14	LD28 LD3 LAD3	M C J	E14	LD21 LD10 LAD10	M C J	G14	LD15 LD16 LAD16	M C J
A15	VDD (Core)	All	C15	LD26 LD5 LAD5	M C J	E15	VDD (Core)	All	G15	LD16 LD15 LAD15	M C J
B1	AD27	All	D1	AD22	All	F1	NC	All	H1	DEVSEL#	All
B2	NC	All	D2	AD24	All	F2	AD17	All	H2	NC	All
B3	REQ#	All	D3	C/BE3#	All	F3	AD18	All	H3	TRDY#	All
B4	INTA#	All	D4	AD28	All	F4	AD19	All	H4	STOP#	All
B5	EESK	All	D5	RST#	All	F5	AD20	All	H5	VDD (Core)	All
B6	VDD (Core)	All	D6	VDD (Core)	All	F6	AD31	All	H6	VSS	All
B7	MODE1	All	D7	NC	All	F7	EECS	All	H7	VSS	All
B8	LINT#	All	D8	VDD (Core)	All	F8	VSS	All	H8	VSS	All
B9	BDIP# WAIT#	M C	D9	VDD (Core)	All	F9	NC	All	H9	VSS	All
B10	BURST# BLAST#	M C, J	D10	TEA# LSERR#	M C, J	F10	LD29 LD2 LAD2	M C J	H10	VSS	All
B11	BG# LHOLDA	M C, J	D11	LCLK	All	F11	NC	All	H11	LD14 LD17 LAD17	M C J
B12	VDD (Core)	All	D12	Bi# BTERM#	M C, J	F12	NC	All	H12	VDD (Core)	All
B13	DP0 DP3	M C, J	D13	LD27 LD4 LAD4	M C J	F13	LD19 LD12 LAD12	M C J	H13	NC	All
B14	NC	All	D14	LD24 LD7 LAD7	M C J	F14	LD18 LD13 LAD13	M C J	H14	LD12 LD19 LAD19	M C J
B15	LD30 LD1 LAD1	M C J	D15	LD23 LD8 LAD8	M C J	F15	NC	All	H15	LD13 LD18 LAD18	M C J
J1	PERR#	All	L1	AD15	All	N1	AD10	All	R1	VDD (Core)	All
J2	LOCK#	All	L2	VDD (Core)	All	N2	AD8	All	R2	AD4	All
J3	SERR#	All	L3	NC	All	N3	AD6	All	R3	AD0	All

Table 14-3. 225-Pin PBGA PCI 9054 Pinout (Continued)

Pin #	Symbol	Bus Mode	Pin #	Symbol	Bus Mode	Pin #	Symbol	Bus Mode	Pin #	Symbol	Bus Mode
J4	VDD (Core)	All	L4	AD11	All	N4	AD1	All	R4	LA0 LA31 DT/R#	M C J
J5	VDD (Core)	All	L5	AD2	All	N5	LA1 LA30 DEN#	M C J	R5	NC	All
J6	AD16	All	L6	LA2 LA29 ALE	M C J	N6	LA4 LA27	M C, J	R6	VDD (Core)	All
J7	VSS	All	L7	LA8 LA23	M C, J	N7	NC	All	R7	LA7 LA24	M C, J
J8	VSS	All	L8	VDD (Core)	All	N8	LA9 LA22	M C, J	R8	LA10 LA21	M C, J
J9	VSS	All	L9	LA14 LA17	M C, J	N9	LA13 LA18	M C, J	R9	VDD (Core)	All
J10	LD2 LD29 LAD29	M C J	L10	LA21 LA10	M C, J	N10	LA16 LA15	M C, J	R10	LA15 LA16	M C, J
J11	VDD (Core)	All	L11	LA30 LBE1#	M C, J	N11	VDD (Core)	All	R11	LA18 LA13	M C, J
J12	LD9 LD22 LAD22	M C J	L12	NC	All	N12	LA24 LA7	M C, J	R12	LA20 LA11	M C, J
J13	VDD (Core)	All	L13	VDD (Core)	All	N13	LA28 LA3	M C, J	R13	LA23 LA8	M C, J
J14	LD10 LD21 LAD21	M C J	L14	LD4 LD27 LAD27	M C J	N14	TSIZ0 LBE3#	M C, J	R14	LA27 LA4	M C, J
J15	LD11 LD20 LAD20	M C J	L15	LD5 LD26 LAD26	M C J	N15	LD0 LD31 LAD31	M C J	R15	VDD (Core)	All
K1	PAR	All	M1	AD13	All	P1	AD7	All			
K2	NC	All	M2	AD12	All	P2	NC	All			
K3	C/BE1#	All	M3	AD9	All	P3	AD3	All			
K4	AD14	All	M4	AD5	All	P4	ENUM#	All			
K5	VDD (Core)	All	M5	NC	All	P5	VDD (Core)	All			
K6	C/BE0#	All	M6	LA3 LA28	M C, J	P6	LA5 LA26	M C, J			
K7	LEDOn/LEDin	All	M7	LA6 LA25	M C, J	P7	VDD (Core)	All			
K8	VSS	All	M8	LA11 LA20	M C, J	P8	NC	All			
K9	LA17 LA14	M C, J	M9	NC	All	P9	LA12 LA19	M C, J			
K10	LA26 LA5	M C, J	M10	LA19 LA12	M C, J	P10	VDD (Core)	All			
K11	VDD (Core)	All	M11	LA22 LA9	M C, J	P11	NC	All			
K12	LD6 LD25 LAD25	M C J	M12	RD/WR# LW/R#	M C, J	P12	NC	All			
K13	LD7 LD24 LAD24	M C J	M13	LA31 LBE0#	M C, J	P13	LA25 LA6	M C, J			
K14	NC	All	M14	LD1 LD30 LAD30	M C J	P14	LA29 LA2	M C, J			
K15	LD8 LD23 LAD23	M C J	M15	LD3 LD28 LAD28	M C J	P15	TSIZ1 LBE2#	M C, J			

## A GENERAL INFORMATION

### A.1 ORDERING INSTRUCTIONS

The PCI 9054 is a 32-bit, 33-MHz PCI Bus Master I/O Accelerator featuring advanced Data Pipe Architecture technology, which includes two DMA engines, programmable Target and Initiator Data Transfer modes, and PCI messaging functions. The PCI 9054 offers 3.3V, 5V tolerant PCI and Local signaling, and supports Universal PCI Adapter designs, 3.3V core, low-power CMOS offered in two package options, a 176-pin PQFP and 225-pin (ball) PBGA. The device is designed to operate at Industrial Temperature range.

**Table A-1. Available Packages**

Package	Ordering Part Number
176-pin PQFP	PCI 9054-AB50PI
225-pin PBGA	PCI 9054-AB50BI

### A.2 UNITED STATES AND INTERNATIONAL REPRESENTATIVES, AND DISTRIBUTORS

A list of PLX Technology, Inc., representatives and distributors can be found at <http://www.plxtech.com>.

### A.3 TECHNICAL SUPPORT

PLX Technology, Inc., technical support information is listed at <http://www.plxtech.com>; or call 408 774-9060 or 800 759-3735.



# Index

## A

### abort

- DMA 3-20, 5-20
- PCI Target 3-7, 5-6-5-7

**absolute maximum rating** 13-1

### accesses

- address
  - decode enable 11-13
  - pointer 11-1
- burst memory-mapped 1-7
- bus 1-10, 11-7
- byte 2-1, 4-1, 11-7
- current Target 1-9
- disabled 8-2, 11-16
- duration 1-9
- during software reset 3-1, 5-1
- Hot Swap 9-1, 9-3
- I/O 2-1, 4-1
  - configuration 3-5, 5-5
  - space 11-7
- internal registers 2-11, 2-12, 4-12
- Local Bus 1-13
  - Big/Little Endian 2-7, 4-7
  - read 2-4, 4-4
  - write 2-4, 4-4
- locked atomic operations 3-10, 5-9
- Lword 2-1, 4-1, 11-7
- Lword, partial 4-4
- Max\_Lat 11-14
- memory base address 11-10
- memory space 11-7
- NANDTREE 12-15-12-19
- other than VPD Expansion ROM 1-5
- partial Lword 2-4, 4-4
- PCI 2-7, 4-7, 11-19
  - Master 3-1, 5-1
- PCI Bus Master 1-6
- PCI Initiator 3-3, 3-5, 5-3, 5-5, 11-1
- PCI Initiator memory 3-4, 5-4
- PCI Target 2-1, 2-4, 2-7, 2-11, 3-1, 3-3, 3-13, 4-1, 4-4, 4-7, 4-11, 5-1, 5-4, 5-11, 11-19, 11-28
- PCIBAR0 11-10
- PCIBAR1 11-11
- PCIBAR2 11-11
- PCIBAR3 11-12
- PCIBAR4 11-12
- PCIBAR5 11-12
- PCI-to-Local 11-3

- remote 1-4
- RST# timing 1-7
- single 1-7
- slave 1-1, 1-6
- VPD 10-1, 11-1, 11-18, 11-22
- Word 2-1, 4-1, 11-7

### AD[31:0]

- 1-9, 12-3

### adapter block diagram 1-1

### address

- bits for decoding 3-12, 5-10
- boundary 2-3, 4-4
- burst start 2-4, 4-4
- cycle 2-3, 2-4, 4-4, 11-1
- data 1-1, 1-6
- decode enable 11-13, 11-23
- EROMBA 11-23
- invariance 2-5, 4-5, 4-6
- LAS0BA 11-19
- LAS0RR 11-19
- LAS1BA 11-28
- LAS1RR 11-27
- LBRD0 11-24
- local bits 2-4, 4-4
- local bus initialization 3-12, 5-10
- local space
  - 0 2-7, 4-7
  - 1 2-7, 4-7
- local spaces 1-1, 1-8
- mapping 3-12, 5-10
- multiple independent spaces 1-4
- Multiplexed Bus mode 12-1
- Non-multiplexed Bus mode 12-1
- PCI
  - command 11-7
  - status 11-8
- PCI Target
  - byte enables 5-12
  - initialization 3-12, 5-10
- PCIBAR0 11-10
- PCIBAR1 11-11
- PCIBAR2 11-11
- PCIBAR3 11-12
- PCIBAR4 11-12
- PCIBAR5 11-12
- PCI-to-Local spaces 1-6
- pointer 11-1
- PROT\_AREA 11-22

## ADS# to C and J modes

---

PVPDAD 11-18  
random read and write 10-2  
Read Ahead mode 1-6  
register 1-8  
register mapping 11-2-11-6  
signal listing 1-9, 1-10  
VPD 10-1

**ADS#**  
1-10, 12-9, 12-12

**ALE**  
1-10, 12-12  
output delay to the local clock 13-6

**arbitration**  
DMA 3-25, 5-23  
local bus 2-2, 4-2  
MARBR 11-20  
PCI 2-2, 4-2

**arbitration, PCI or Local Bus** 4-2

**atomic operations, locked** 3-10, 5-9

**B**

**backoff**  
during deadlock 3-16, 5-13  
preempt deadlock solution 3-16, 5-14  
software solution for deadlock 3-16, 5-14  
solution for deadlock 3-16, 5-14

**BB#**  
1-10, 12-6

**BDIP#**  
1-10, 12-6

**BG#**  
1-10, 12-6

**BI#**  
1-10, 12-6

**Big Endian**  
See Big/Little Endian or Endian

**Big/Little Endian**  
Descriptor register (BIGEND) 11-21

**BIGEND**  
11-21

**BIGEND#**  
1-10, 12-6, 12-9, 12-12

**BIST**  
6-3

**BLAST#**  
1-10, 12-9, 12-12

**block diagram**  
internal 1-2  
typical adapter 1-1

**Block Dual Address cycle** 3-19, 5-16

**BR#**  
1-10, 12-6

**BREQi**  
1-10, 12-9, 12-12

**BREQo**  
1-10, 12-9, 12-12

**BTERM#**  
1-10, 2-3, 4-3, 5-7, 12-9, 12-12

**burst operation, zero wait state** 1-1, 1-5

**BURST#**  
1-11, 12-6

**Bus**  
See Local Bus or PCI Bus

**bus operation** 2-1-2-15, 4-1-4-16

**bus protocol devices** 1-1, 1-6

**byte enables** 5-5, 5-12, 5-22  
See Also LBE[3:0]#

## C

### C and J modes

AC electrical characteristics 13-4, 13-6

accesses  
byte 4-1  
I/O 4-1  
internal registers 4-12  
Local Bus  
Big/Little Endian 4-7  
read 4-4  
write 4-4  
Lword 4-1  
Lword, partial 4-4  
PCI 4-7  
PCI Initiator memory 5-4  
PCI Master 5-1  
PCI Target 4-1, 4-4, 4-7, 5-1  
software reset, during 5-1

address  
boundary 4-4  
cycle 4-4  
local bits 4-4  
PCI Target byte enables 5-12

arbitration, PCI or Local Bus 4-2

backoff  
during deadlock 5-13  
preempt deadlock solution 5-14  
software solution for deadlock 5-14  
solution for deadlock 5-14

base address 5-12

Big Endian, 8-bit Local Bus  
lower byte lane transfer 4-6  
upper byte lane transfer 4-6

- Block DMA mode 5-15-5-16
- Block DMA, Dual Address cycle 5-16
- Bus
  - interface pin description 12-1
  - operation 4-1-4-16
- byte enables 5-12
- command codes 4-1
- configuration
  - cycles 5-5
  - cycles, example 5-6
  - local initialization 4-7
  - local registers 4-7
  - PCI, registers 4-11, 4-12
  - registers 5-1
  - serial EEPROM 5-1
- cycles 5-5
  - dual address 5-6
  - Local Bus 4-2
- deadlock conditions 5-13
- Delayed Read mode 5-9
- Direct Local-to-PCI Master Command 4-1
- DMA
  - abort 5-20
  - arbitration 5-23
  - channel interrupts 5-20
  - Clear Count mode 5-18
  - configuration 5-1
  - data transfers 5-20-5-22
    - Local-to-PCI 5-21
    - PCI-to-Local 5-21
  - Demand mode 5-22
  - Dual Address cycle 5-15
  - Master Command codes 4-1
  - memory write and invalidate 5-20
  - operation 5-14-5-23
  - priority 5-20
  - timing diagrams, C mode only 5-63-5-72
  - timing diagrams, J mode only 5-81-5-82
  - transfer 4-4
  - unaligned transfers 5-22
- dual address 4-1, 5-7
- end of transfer (EOT#) input 5-22
- Endian, Big
  - 16-bit Local Bus 4-6
    - lower word lane transfer 4-6
    - upper word lane transfer 4-6
  - 32-bit Local Bus 4-6
    - upper Lword lane transfer 4-6
  - 8-bit Local Bus 4-6
  - byte number and lane cross-reference 4-5
  - cycle reference table 4-5
- Local Bus 4-5
- Local Bus accesses 4-7
- Program mode 4-5
- Endian, Little
  - byte number and lane cross-reference 4-5
  - cycle reference table 4-5
  - Local Bus 4-5
  - Local Bus accesses 4-7
  - PCI Bus 4-5
  - Program mode 4-5
- FIFOs 5-9
  - Continuous Burst mode 4-4
  - PCI Target 5-1, 5-9
- functional description 5-1-5-82
- I/O
  - decode 5-4
  - PCI Initiator 5-5
- initialization
  - local 4-7
  - PCI Target 5-10
- Local Address
  - Big/Little Endian mode 4-7
  - mapping 5-10
- Local Address increment 4-4, 5-14
- Local Bus 5-1
  - Big/Little Endian mode 4-5-4-7
  - characteristics 5-10
  - cycles 4-2, 4-4, 4-5
  - FIFO, response to 5-2
  - Latency Timer 5-23
  - PCI Target
    - access 5-11
    - Command codes 4-1
    - operation 5-1
- Local NMI 6-4
- Local SERR# 6-4
- Local-to-PCI doorbell interrupt 6-3
- map
  - PCI software 5-10
  - serial EEPROM memory 4-11
- map/remap 5-10, 5-12
- Master Command codes 4-1
- memory 5-9
  - commands
    - aliased 4-1
    - basic 4-1
    - read 4-1
    - serial EEPROM map 4-11
    - write 4-1
    - write and invalidate, PCI Initiator 5-8

- new capabilities structure 4-11
- operation 1-7
- Pause Timer 5-23
- PCI
  - arbitration 4-2
  - Dual Address cycle 4-1
  - Master Command codes 4-1
- PCI Bus 5-1, 5-3
  - cycles 4-1
  - FIFO, response to 5-2
  - internal registers, access to 4-12
  - Little Endian mode 4-5
  - PCI Target operation 5-1
  - software reset 5-1
  - wait states 4-3
- PCI Initiator
  - access 5-3
  - Dual Address cycle 5-6
  - I/O 5-5
  - I/O configuration access 5-5
  - I/O decode 5-4
  - memory 5-4
  - memory access 5-4
  - memory write and invalidate 5-8
  - operation 5-1
  - PCI Target abort 5-6, 5-6-5-7
- PCI Initiator FIFOs 5-4
- PCI Initiator timing diagrams
  - C mode only 5-25-5-43
  - J mode only 5-73-5-76
- PCI Target
  - accesses to 8- or 16-bit Local Bus 4-4
  - Big Endian/Little Endian cycle reference table 4-5
  - Command code, memory write and invalidate 4-1
  - Command codes 4-1
  - Direct Data Transfer mode 5-1-5-14
  - during serial EEPROM initialization 4-11
  - example 5-12-5-13
  - FIFO full or empty, response to 5-2
  - initialization 5-10
  - Local Bus Big Endian/Little Endian mode
    - accesses 4-7
  - lock 5-9
  - operation 5-8-5-13
  - priority 5-13
  - serial EEPROM, during initialization 4-7
  - timing diagrams
    - C mode only 5-44-5-62
    - J mode only 5-77-5-80
  - transfer 5-9
    - wait states Local Bus 4-3
- pinout 12-9-12-14
- range for decoding 5-10, 5-12
- read 5-2
  - accesses 4-1, 4-12
  - Command codes, PCI Target 4-1
  - FIFOs 4-4
  - I/O command 4-1
  - Local Bus accesses 4-4
  - memory command 4-1
  - memory line 4-1
  - memory multiple 4-1
  - Read Ahead mode 4-4
  - serial EEPROM 4-11
- Read Ahead mode 5-9
  - in burst read cycles 4-4
- READY#, serial EEPROM initialization 4-7, 4-11
- response to FIFO 5-1
- Scatter/Gather DMA mode 5-16-5-19
- serial EEPROM 4-7-4-16
  - device ID registers 4-7
  - extra long load 4-10
  - extra long load registers 4-10
  - initialization 4-11
  - internal register access 4-11
  - long load 4-8-4-9
  - long load registers 4-9
  - memory map 4-11
  - new capabilities function support 4-11
  - operation 4-7
  - PCI Bus, access to internal registers 4-12
  - recommended 4-11
  - software reset 5-1
  - timing diagrams 4-13-4-16
  - vendor ID registers 4-7
- signal listing 1-9-1-14
- single address block DMA initialization 5-7
- timing diagrams 4-13-4-16, 5-24-5-82
- Vital Product Data (VPD)
  - internal register access 4-11
  - new capabilities function feature 4-11
  - read or write serial EEPROM 4-8
- wait state
  - control 4-3
  - generation 4-4
  - Local Bus 4-3
  - PCI Bus 4-3
- write 5-2
  - accesses 4-1, 4-12
  - Command codes, PCI Target 4-1

- I/O command 4-1
- Local Bus accesses 4-4
- memory command 4-1
- memory write and invalidate 4-1
- serial EEPROM operation 4-7
- C/BE[3:0]#**
  - 1-9, 12-3
- CAP\_PTR**
  - 11-14
- capacitance**
  - 13-1
- CCS#**
  - 12-5
  - chip select 1-11
- channels, dual DMA** 1-3, 1-4, 1-5, 1-6, 1-8, 1-13
- chip select**
  - EECS 1-11
  - IDSEL 1-9
  - initialization device 1-9
  - serial EEPROM 1-11
- circular FIFOs summary**
  - 7-6
- Clear Count mode**
  - 3-20, 5-18
- clocks**
  - 1-9, 1-10, 1-12, 8-1
  - buffered PCI 1-8
  - EESK 1-11, 2-7, 4-7
  - input 13-3
  - local 13-6
  - local input 1-12
  - PCI 2-7, 4-7, 11-15
  - PCIMGR 11-14
  - PCLK 1-9
  - RST# 1-7
  - serial EEPROM 2-7, 4-7
  - serial EEPROM clock pin 1-11
- CNTRL**
  - 3-1, 5-1, 11-33
- Command codes** 2-1, 4-1
  - Local-to-PCI 2-1
- CompactPCI** 1-3, 1-12
  - Hot Swap 9-1-9-4
    - adapters 1-4
    - applications 9-1
    - blue LED 9-3
    - board healthy 9-2
    - board slot control 9-2
    - capabilities register bit definition 9-4
    - capable 9-1
    - compliant 1-1, 1-5, 1-8
    - control bits 9-4
    - Control/Status register (HS\_CSR) 9-3
    - controlling connection processes 9-1
    - ejector switch 9-3
    - ENUM# 9-3
    - friendly 9-1
    - hardware connection control 9-1
    - ID 9-4
    - levels of compatibility 9-1
    - Next\_Cap pointer 9-4
    - platform reset 9-2
    - ready 9-1
    - register 11-1
    - software connection control 9-3
- compatibility**
  - with PCI 9050 1-7
  - with PCI 9080 1-7
- compatible message unit** 7-1
- configuration**
  - accesses 1-7
  - Big/Little Endian 3-11, 5-10
  - BTERM# 1-10
  - bus-width 1-12
  - command type 2-1, 4-1
  - control/status register 9-3
  - cycles 3-5, 5-5
    - example 3-6, 5-6
  - Hot Swap 9-1, 9-3
  - IDSEL 1-9
  - initialization 5-1
  - LBE[3:0]# 1-12
  - load information 1-6
  - local
    - cycles 8-2
    - initialization 2-7, 4-7
    - registers 2-7, 2-11, 3-1, 4-7, 4-11
  - new capabilities 10-1
  - new capability linked list 11-1
  - PCI
    - cycles 8-2
    - registers 2-11, 4-11, 4-12
  - PCI 9054 initialization 3-1
  - Power Management 8-2
  - read and write transactions 1-9
  - register space 1-7
  - registers 1-10, 3-1, 5-1, 11-7-11-28
    - address mapping 11-2-11-6
  - serial EEPROM 3-1, 5-1
  - software reset 3-1

## Configuration Address register (DMCFGA) to DMA

---

space 9-3, 10-1, 11-1  
subsystem ID 1-7  
subsystem vendor ID 1-7  
system reconfiguration 9-1, 9-3  
type 0 cycle 2-11, 4-12  
VPD 1-1, 10-1

**Configuration Address register (DMCFGA)**  
11-27

### conversion

Big/Little Endian 1-1, 1-4, 1-8, 3-11, 5-10  
on-the-fly 3-11, 5-10

### counter

prefetch 1-1, 1-6  
programmable 1-8

### cycles

configuration 3-5, 5-5  
configuration example 3-6, 5-6  
dual address 3-6, 5-6  
Local Bus 2-2-2-5, 4-2-4-5  
memory write and invalidate 3-7, 5-8

## D

### DAC

3-6, 5-6

### DACK0#

1-14, 12-5

**data assignment convention** 1-7

### data bus

contents for read cycles 3-14  
contents for write cycles 3-14

### Data Pipe Architecture

I/O Accelerator 1-3  
See *Also* dual DMA channels  
technology 1-3

### data transfers, DMA

Local-to-PCI 3-23, 5-21  
PCI-to-Local 3-23, 5-21

### deadlock

conditions 3-15, 5-13  
during access 3-15, 5-13  
full 3-15, 5-13  
partial 3-15, 5-13

### Deferred Read mode

Local-to-PCI 1-1

**Delayed Read mode** 1-1, 3-10, 5-9

**Demand mode, DMA support** 1-8

### DEN#

1-11

### Descriptor register (LBRD1)

11-28

### device ID

11-7

### DEVSEL#

1-9, 12-3

### direct

interface Local Bus 1-6  
Local-to-PCI Command codes 2-1  
Local-to-PCI Master Command codes 4-1

### Direct Slave

See PCI Target

### DMA

abort 3-20, 5-20  
acknowledge 1-14  
arbitration 3-25, 5-23  
Arbitration register (DMAARB) 11-1, 11-39  
Block Dual Address cycle 3-19, 5-16  
Channel 0  
Command/Status register (DMACSR0) 11-38  
Descriptor Pointer register (DMADPR0) 11-35  
Local Address register (DMALADR0) 11-35  
Mode register (DMAMODE0) 11-34  
PCI Address register (DMAPADR0) 11-35  
Transfer Size register (DMASIZ0) 11-35

#### Channel 1

Command/Status register (DMACSR1) 11-38  
Descriptor Pointer register (DMADPR1) 11-37  
Local Address register (DMALADR1) 11-37  
Mode register (DMAMODE1) 11-36  
PCI Address register (DMAPADR1) 11-37  
Transfer Size register (DMASIZ1) 11-37

channel interrupts 3-22, 5-20

channels 1-1, 1-3

channels, number of 1-8

Clear Count mode 3-20, 5-18

configuration 3-1, 5-1

controller 2-4, 4-4

cycle 11-1

data transfers

Local-to-PCI 3-23, 5-21

PCI-to-Local 3-23, 5-21

Demand mode 1-8, 1-14, 3-24, 5-22

direct hardware control 1-3

dual address 3-6, 3-19

Dual Address cycle 3-17, 5-6, 5-15, 5-16

engines 1-1, 1-3

interrupts 6-1, 6-3-6-4

list management 1-3

local bus error 3-24

Master Command codes 2-1, 4-1

priority 3-20, 5-20

- read 1-5
- registers 11-10, 11-34-11-39
- registers, address mapping 11-5
- request 1-14
- ring management 1-3
- scatter/gather 3-20, 3-21
- scatter/gather list management 1-3
- Shuttle mode 1-3
- Threshold register (DMAATH) 11-39
- transfer 1-10, 1-13, 1-14, 2-4, 4-4
- unaligned
  - transfer support 1-7
  - transfers 3-24, 5-22
- wait states 2-3
- write 1-5
- DMA 0 PCI Dual Address Cycle register (DMADAC0)**  
11-39
- DMA 1 PCI Dual Address Cycle register (DMADAC1)**  
11-39
- DMAARB**  
11-39
- DMACSR0**  
11-38
- DMACSR1**  
11-38
- DMADAC0**  
3-6, 5-6, 11-39
- DMADAC1**  
3-6, 5-6, 11-39
- DMADPR0**  
11-35
- DMADPR1**  
11-37
- DMALADR0**  
11-35
- DMALADR1**  
11-37
- DMAMODE**  
3-6, 5-6
- DMAMODE0**  
11-34
- DMAMODE1**  
11-36
- DMAPADR0**  
11-35
- DMAPADR1**  
11-37
- DMASIZ0**  
11-35
- DMASIZ1**  
11-37
- DMATHR**  
11-39
- DMCBFGA**  
11-27
- DMCFGA**  
3-1, 5-1
- DMDAC**  
11-28
- DMDAD**  
3-1, 5-1
- DMLBAI**  
3-1, 5-1, 11-25
- DMLBAM**  
3-1, 5-1, 11-25
- DMPAF**  
1-11, 1-13, 12-7, 12-9, 12-12
- DMPBAM**  
3-1, 5-1, 11-26
- DMRR**  
3-1, 5-1, 11-25
- Doorbell registers** 1-8, 6-2
  - L2PDBELL 11-30
  - P2LDBELL 11-30
- DP[0:3]**  
1-11, 12-6, 12-9
- DP[3:0]**  
1-11, 12-13
- DREQ0#**  
1-11, 1-14, 12-5
- DT/R#**  
1-11, 12-13
- Dual Address**
  - block DMA initialization 5-7
  - cycle 2-1, 3-17, 4-1, 5-15
  - Cycle register (DMDAC) 11-28
  - timing diagram 3-8, 3-18, 5-7
- dual DMA channels** 1-3, 1-4, 1-5, 1-6, 1-8, 1-13
- E**
- EECS**  
1-11, 12-3
- EEDI**  
1-11, 12-3
- EEDO**  
1-11, 12-3
- EESK**  
1-11, 12-3

**electrical characteristics over operating range**

13-2

**electrical specifications**

13-1-13-6

AC characteristics 13-3, 13-4, 13-5, 13-6

ALE output delay 13-6

capacitance 13-1

local inputs 13-3

local outputs 13-5

delay 13-5

**embedded**

design 12-1

host design 1-5

systems 1-1, 1-3

**enable sequence 7-5**

**end of transfer (EOT#) input 3-24, 5-22**

**Endian**

Big 2-5-2-7

16-bit Local Bus 2-6, 4-6

lower word lane transfer 2-6, 4-6

upper word lane transfer 2-6, 4-6

32-bit Local Bus 2-5, 4-6

upper Lword lane transfer 2-5, 4-6

8-bit Local Bus 2-6, 4-6

lower byte lane transfer 2-6, 4-6

upper byte lane transfer 2-6, 4-6

byte number and lane cross-reference 2-5, 4-5

conversion 1-1, 1-4, 1-8

cycle reference table 2-5, 4-5

Local Bus 2-5, 4-5

Local Bus accesses 2-7, 4-7

on-the-fly conversion 3-11, 5-10

Program mode 2-5, 4-5

Little 2-5-2-7

byte number and lane cross-reference 2-5, 4-5

conversion 1-1, 1-4, 1-8

cycle reference table 2-5, 4-5

Local Bus 2-5, 4-5

Local Bus accesses 2-7, 4-7

on-the-fly conversion 3-11, 5-10

PCI Bus 2-5, 4-5

Program mode 2-5, 4-5

**Endian, Big/Little**

2-5-2-7, 4-5-4-7

See Also Endian

**ENUM#**

9-3, 11-17, 12-3

**EOT#**

1-13, 12-7, 12-9, 12-12

**EROMBA**

3-12, 5-10, 11-23

**EROMRR**

11-23

**Expansion ROM**

Range register (EROMRR) 11-23

register (EROMBA) 11-23

**F**

**F. Bit, as a flag 10-1**

**FIFOs**

circular operation 7-4

circular summary 7-6

Continuous Burst mode 2-4, 4-4

depth 1-5

enable high-performance bursting 1-7

inbound free list 7-2

inbound post queue 7-3

number of 1-8

outbound free list 7-5

outbound post queue 7-3

PCI Initiator 3-3-3-4, 5-4

PCI Target

operation 3-1, 5-1

Read Ahead mode 3-10, 5-9

transfer 3-11, 5-9

programmable 1-1, 1-5

Read Ahead mode 1-6

response to 3-1, 5-1

**flush 3-10, 5-9**

**FRAME#**

1-9, 12-3, 12-5

**free list FIFO, inbound 7-2**

**free list FIFO, outbound 7-5**

**free queue, inbound 7-5**

**functional description**

C and J modes 5-1-5-82

M mode 3-1-3-65

**G**

**general electrical specifications 13-1-13-2**

absolute maximum ratings 13-1

operating ranges 13-1

over operating range 13-2

thermal resistance of packages 13-1

**GNT#**

12-3

**H****hold waveform**

13-3

**Hot Plug** 1-1, 1-5, 1-8**Hot Swap**

capable 1-4

Control register (HS\_CNTL)

ID

Hot Swap 11-17

Control/Status register (HS\_CSR) 11-17

friendly 1-4

ID 11-17

Next Capability Pointer register (HS\_NEXT) 11-17

resources 9-3

See Also CompactPCI Hot Swap

**HS\_CNTL**

11-17

**HS\_CSR**

9-3, 11-17

**HS\_NEXT**

11-17

**I****I/O**

accelerator 1-3

configuration 5-5

configuration access 3-5, 5-5

decode 3-3, 5-4

PCI Initiator 3-5, 5-5

See Also user input and output

**I<sub>2</sub>O**

messaging unit 1-8

See Also Intelligent I/O 7-1

**I<sub>2</sub>O SIG** 1-3**IBM, PPC401** 1-1, 1-6**ID**

Configuration ID register (PCIIDR) 11-7

device 11-7

Hardcoded Configuration ID register (PCIHIDR) 11-33

Hardcoded Revision ID register (PCIHREV) 11-33

Power Management

Capability ID register (PMCAPIID) 11-14

Revision ID register (PCIREV) 11-8

serial EEPROM, revision 11-8

serial EEPROM, subsystem 11-13

serial EEPROM, vendor 11-7

Subsystem register 11-13

Subsystem Vendor register 11-12

vendor 11-7

VPD 11-18

**IDMA**

channel 1-4

operation 3-9

PowerQUICC operation 1-4

**IDSEL**

1-9, 12-3

**IFHPR**

11-41

**IFTPR**

11-41

**Inbound**

Free Head Pointer register (IFHPR) 11-41

free list FIFO 7-2

Free Tail Pointer register (IFTPR) 11-41

IPHPR 7-2

IPTPR 7-2

Post Head Pointer register (IPHPR) 11-42

post queue FIFOs 7-3

Post Tail Pointer register (IPTPR) 11-42

Queue Port register (IQP) 11-40

**industrial**

PCI implementations 1-3

temp range operation 1-1

**initialization**

3-1, 5-1, 11-1

C and J modes 5-1

IDSEL 1-9

local 2-7, 4-7

M mode 3-1

PCI Target 3-12

example 3-14, 5-12

local bus 3-12, 5-10

PCI 5-10

PMC 11-15

reset 3-12, 5-10

**Initialization Control register (CNTRL)** 11-33**Initiator**

See PCI Initiator

**input**

PCI Bus 3-1, 5-1

user 6-5

**INTA#**

1-9, 6-1, 12-4

**INTCSR**

11-31-11-32

**Intel i960**

1-1, 1-6

**intelligent I/O**

7-1-7-6

circular FIFOs

## internal block diagram to LBRD1

---

- operation 7-4
- summary 7-6
- compatible message unit 7-1
- enable sequence 7-5
- inbound
  - free list FIFOs 7-2
  - free queue 7-5
  - messages 7-1
  - post queue FIFOs 7-3
- outbound
  - free list FIFO 7-5
  - messages 7-1
  - post queue 7-5
  - post queue FIFOs 7-3
- pointer management 7-2
- internal block diagram**
  - 1-2
- Interrupt Control/Status register (INTCSR)** 11-31-11-32
- interrupts**
  - built-in self test (BIST) 6-3
  - control/status 11-1
  - disabled 8-1, 8-2, 11-16
  - DMA 3-22, 5-20
  - DMA Channel 0/1 6-3-6-4
  - doorbell register 6-2
  - ENUM#
    - driven 9-3
    - interrupt 9-3, 9-4
    - interrupt clear 11-17
  - error sources 6-1
  - generator 1-1, 1-7
  - INTA# 6-1
  - interrupt A 1-9
  - line 11-2, 11-14
  - local
    - input (LINT#) 6-1
    - output (LINT#) 6-2, 8-2
  - Local-to-PCI 1-6
  - Local-to-PCI doorbell 6-3
  - mailbox register 6-2
  - Master/Target abort 6-2
  - PCI local 6-1-6-5
  - PCI request 1-9
  - PCI-to-Local 1-6
  - pin 11-2, 11-14
  - power management event (PME#) 1-5, 1-9
  - TEA# mask 11-1
  - User I/O 6-1-6-5
  - wake-up event 1-9

- IPHPR**
  - 11-42
- IPTPR**
  - 11-42
- IQP**
  - 11-40
- IRDY#**
  - 1-9, 12-4
- J**
- J mode**
  - See C and J modes
- K**
- Keep Bus mode**
  - C and J modes 1-7
  - M mode 1-7
- L**
- L2PDBELL**
  - 11-30
- LA[0:31]**
  - 1-11, 12-7
- LA[1:0]**
  - 4-4
- LA[28:2]**
  - 1-11, 12-13
- LA[31:2]**
  - 1-11, 3-14, 12-10
- LAD[31:0]**
  - 1-11, 12-13
- LAS0BA**
  - 3-12, 5-10, 11-19
- LAS0RR**
  - 3-12, 5-10, 11-19
- LAS1BA**
  - 3-12, 5-10, 11-28
- LAS1RR**
  - 3-12, 5-10, 11-27
- Latency Timer**
  - Local Bus 1-7, 2-2, 2-3, 3-11, 3-12, 3-17, 3-24, 3-25, 4-2, 5-9, 5-15, 5-23, 5-70, 11-20
  - timing diagram 3-62
  - PCI Bus 3-17, 11-9
  - See Also PCI Latency Timer, PCI Bus Latency Timer or Local Bus Latency Timer
- latency, reduced data** 1-6
- LBE[3:0]#**
  - 1-12, 12-10, 12-14
- LBRD0**
  - 11-24
- LBRD1**

- 11-28
- LCLK**
  - 1-12, 12-5
- LD[0:31]**
  - 1-12, 12-7
- LD[0:7]**
  - 3-14
- LD[16:23]**
  - 3-14
- LD[24:31]**
  - 3-14
- LD[31:0]**
  - 1-12, 12-10
- LD[8:15]**
  - 3-14
- LEDOn/LEDIn**
  - 1-12, 12-5
- LFRAME#**
  - 1-12
- LHOLD**
  - 1-12, 12-14
- LHOLDA**
  - 1-13, 12-11, 12-14
- LINT#**
  - 1-13, 6-1, 6-2, 8-2, 12-5
- list management** 1-3
- Little Endian**
  - See Big/Little Endian or Endian
- LLOCKi#**
  - 1-13, 1-14, 12-5
- LLOCKo#**
  - 1-13, 1-14, 12-5
- LMISC**
  - 11-22
- Local Address**
  - Big/Little Endian mode 2-7, 4-7
  - EROMBA 11-23
  - increment 2-3, 3-17, 4-4, 5-14
  - LAS0BA 11-19
  - LAS0RR 11-19
  - LAS1RR 11-27
  - mapping 3-12, 5-10
  - PCIBAR2 11-11
  - PCIBAR3 11-12
  - PCIBAR4 11-12
  - PCIBAR5 11-12
  - PCI-to-Local spaces 1-6
  - Space 0 Local Base Address register (LAS0BA) 11-19
  - Space 0 Range register (LAS0RR) 11-19
  - Space 0/Expansion ROM register (LBRD0) 11-24
  - Space 1 Bus Region Descriptor register (LBRD1) 11-28
  - Space 1 Local Base Address register (LAS1BA) 11-28
  - Space 1 Range register (LAS1RR) 11-27
  - spaces 1-1, 1-8, 3-9, 5-8, 11-2, 11-3
- Local Base**
  - Address register for PCI Memory (DMPBAM) 11-26
- Local Base Address register for PCI Configuration (DMLBAI)** 11-25
- Local Bus**
  - 3-1, 3-12, 5-9
  - access internal registers 2-12, 4-12
  - accesses 5-1
  - arbitration 2-2, 4-2
  - atomic operation 1-14
  - Base Address register (DMLBAM) 11-25
  - Big/Little Endian mode 2-5-2-7, 4-5-4-7
  - characteristics defined for target regions 5-10
  - configuration registers 11-3
  - current 1-10
  - cycles 2-2-2-4, 4-2-4-4
  - direct interface 1-6
  - error condition 3-24
  - FIFO, response to 5-2
  - hold request 1-12
  - independent interface pins 12-5
  - interface 2-11
  - LAS0RR 11-19
  - Latency Timer 1-7, 2-2, 2-3, 3-11, 3-12, 3-17, 3-24, 3-25, 4-2, 5-9, 5-15, 5-23, 5-70, 11-20
    - timing diagram 3-62
  - PCI Target
    - access 3-13, 5-11
    - Command codes 2-1, 4-1
    - initialization 3-12, 5-10
    - operation 3-1, 5-1
  - PCISR 11-8
  - PCI-to-Local 1-3
  - pin information 12-1
  - PMCSR 11-16
  - power management 8-1
  - priority control 1-4
  - programmable 1-6
  - Read Ahead mode 1-6
  - Read Ahead mode, in relationship to 3-10
  - registers 4-11
  - response to 3-1
  - response to FIFO 3-2
  - serial EEPROM 2-7, 4-7
  - signaling 1-5, 1-8
  - timing diagrams 3-26, 5-24

## local input setup to M mode

---

- types
  - 176-Pin PQFP 2-2, 4-2
  - 225-Pin PBGA 2-2, 4-2
- VPD 10-1
- wait states 2-3
- width 5-12
- width control 1-4
- local input setup**
  - 13-3
- Local Miscellaneous Control register (LMISC)**
  - 11-22
- Local NMI**
  - 6-4
- Local Range register (DMRR)**
  - 11-25
- Local SERR#**
  - 6-4
- local signal output delay**
  - 13-5
- Local-to-PCI**
  - Deferred Read mode 1-1
  - doorbell interrupt 6-3
- lock**
  - atomic operations 3-10, 5-9
  - cycles 1-8
  - local output 1-14
  - LOCK# 1-9, 3-10, 5-9, 12-4
  - PCI Target 3-10, 5-9
- LOCK#**
  - 1-9, 3-10, 5-9, 12-4
- LRESETo#**
  - 1-13, 12-5
- LSERR#**
  - 1-13, 12-11, 12-14
- LW/R#**
  - 1-13, 12-14
- M**
- M mode**
  - 1-6, 2-1
  - AC electrical characteristics 13-3, 13-5
  - accesses
    - byte 2-1
    - during software reset 3-1
    - I/O 2-1
    - internal registers 2-11, 2-12
    - Local Bus Big/Little Endian 2-7
    - Local Bus read 2-4
    - Local Bus write 2-4
    - Lword 2-1
    - PCI 2-7
    - PCI Master 3-1
    - PCI Initiator memory 3-4
    - PCI Target 2-1, 2-4, 2-7, 3-1
  - address
    - boundary 2-3
    - burst start 2-4
    - cycle 2-3, 2-4
    - local bits 2-4
  - arbitration
    - local bus 2-2
    - PCI 2-2
  - backoff
    - during deadlock 3-16
    - preempt deadlock solution 3-16
    - software solution for deadlock 3-16
    - solution for deadlock 3-16
  - base address 3-14
  - Block DMA Dual Address cycle 3-19
  - Block DMA mode 3-17-3-19
  - Bus mode interface pin description 12-1
  - Bus operation 2-1-2-15
  - Command codes 2-1
  - Command codes Local-to-PCI 2-1
  - configuration
    - cycles 3-5
    - cycles, example 3-6
    - local initialization 2-7
    - local registers 2-7, 3-1
    - PCI registers 2-11
    - registers 3-1
    - serial EEPROM 3-1
  - cycles 3-5
    - dual address 3-6
    - Local Bus 2-2
  - deadlock conditions 3-15
  - Delayed Read mode 3-10
  - direct Local-to-PCI Command codes 2-1
  - DMA
    - abort 3-20
    - arbitration 3-25
    - channel interrupts 3-22
    - Clear Count mode 3-20
    - configuration 3-1
    - data transfers 3-22-3-24
      - Local-to-PCI 3-23
      - PCI-to-Local 3-23
    - Demand mode 3-24
    - Dual Address cycle 3-17
    - local bus error 3-24

- Master Command codes 2-1
- memory write and invalidate 3-20
- operation 3-16-3-25
- priority 3-20
- timing diagrams 3-51-3-65
- transfer 2-4
- unaligned transfers 3-24
- Dual Address cycle 2-1
- dual address timing diagram 3-18
- end of transfer (EOT#) input 3-24
- Endian, Big 2-5-2-7
  - 16-bit Local Bus 2-6
    - lower word lane transfer 2-6
    - upper word lane transfer 2-6
  - 32-bit Local Bus 2-5
    - upper Lword lane transfer 2-5
  - 8-bit Local Bus 2-6
    - lower byte lane transfer 2-6
    - upper byte lane transfer 2-6
  - byte number and lane cross-reference 2-5
  - cycle reference table 2-5
  - Local Bus 2-5
  - Local Bus accesses 2-7
  - Program mode 2-5
- Endian, Little 2-5-2-7
  - byte number and lane cross-reference 2-5
  - cycle reference table 2-5
  - Local Bus 2-5
  - Local Bus accesses 2-7
  - Program mode 2-5
- FIFOs 3-11
  - Continuous Burst mode 2-4
  - PCI Target 3-10
  - PCI Target operation 3-1
- functional description 3-1-3-65
- I/O decode 3-3
- I/O PCI Initiator 3-5
- IDMA operation 3-9
- initialization 3-1
  - local 2-7
  - PCI Target 3-12
- Keep Bus mode 1-7
- Local Address
  - Big/Little Endian mode 2-7
  - Mapping 3-12
- Local Address increment 2-3, 3-17
- Local Bus 3-14
  - access internal registers 2-12
  - Big/Little Endian mode 2-5-2-7
  - characteristics 3-12
  - cycles 2-2, 2-4, 2-5
  - Latency Timer 3-25
  - PCI Target
    - access 3-13
    - Command codes 2-1
    - PCI Target operation 3-1
    - response to 3-1
    - response to FIFO 3-2
- Local NMI 6-4
- Local TEA# 6-4
- Local-to-PCI doorbell interrupt 6-3
- map
  - PCI software 3-12
  - serial EEPROM memory 2-11
- map/remap 3-12, 3-14
- Master Command codes 2-1
- memory 3-11
  - commands
    - aliased 2-1
    - basic 2-1
    - read 2-1
    - serial EEPROM map 2-11
    - write 2-1
    - write and invalidate, PCI Initiator 3-7
- new capabilities structure 2-10
- Pause Timer 3-25
- PCI
  - arbitration 2-2
  - Dual Address cycle 2-1
  - Master Command codes 2-1
- PCI Bus 3-3
  - access to internal registers 2-11
  - cycles 2-1
  - input 3-1
  - Little Endian mode 2-5
  - PCI Target operation 3-1
  - response to FIFO 3-2
  - software reset 3-1
  - wait states 2-3
- PCI Initiator
  - access 3-3
  - Dual Address cycle 3-6
  - I/O 3-5
    - configuration access 3-5
    - decode 3-3

## Mailbox to mailbox registers

---

- memory 3-3
  - access 3-4
  - write and invalidate 3-7
- operation 3-1
- PCI Target abort 3-7, 3-7
- PCI Initiator FIFOs 3-3-3-4
- PCI Initiator timing diagrams 3-27-3-35
- PCI SERR# 6-4
- PCI Target
  - accesses to 8- or 16-bit Local Bus 2-4
  - Big Endian/Little Endian cycle reference table 2-5
  - Command codes 2-1
  - during serial EEPROM initialization 2-7, 2-11
  - example 3-14-3-15
  - FIFO full or empty, response to 3-2
  - initialization 3-12
  - Local Bus Big Endian/Little Endian mode
    - accesses 2-7
  - lock 3-10
  - operation 3-9-3-15
  - partial Lword accesses 2-4
  - priority 3-15
  - timing diagrams 3-36-3-50
  - transfer 3-11-3-12
  - transfer size 3-14-3-15
  - wait states 2-3
- pinout 12-6-12-8
- range for decoding 3-12, 3-14
- read
  - accesses 2-1, 2-12
  - FIFOs 2-4
  - I/O command 2-1
  - Local Bus accesses 2-4
  - memory command 2-1
  - memory line 2-1
  - memory multiple 2-1
  - PCI Target 3-2
  - PCI Target Command codes 2-1
  - Read Ahead mode 2-4
  - serial EEPROM 2-10
  - serial EEPROM operation 2-7
- Read Ahead mode 3-10
  - in burst read cycles 2-4
- read cycles 3-14
- response to FIFO 3-1
- Scatter/Gather DMA mode 3-19-3-20
- Scatter/Gather DMA PCI Dual Address cycle 3-20, 3-21
- SDMA operation 3-9

- serial EEPROM 2-7-2-15
  - device ID registers 2-7
  - extra long load 2-10
  - initialization 2-11
  - internal register access 2-11
  - long load 2-8-2-9
  - memory map 2-11
  - new capabilities function 2-10
  - operation 2-7
  - PCI Bus access to internal registers 2-11
  - recommended 2-10
  - software reset 3-1
  - timing diagrams 2-13-2-16
  - vendor ID registers 2-7
- signal listing 1-9-1-14
- TA# serial EEPROM initialization 2-7, 2-11
- timing diagrams 3-26-3-65
  - serial EEPROM 2-13-2-16
- transfer size bits (TSIZ[0:1]) 2-4
- Vital Product Data (VPD)
  - internal register access 2-11
  - new capabilities function 2-10
  - read or write serial EEPROM 2-8
- wait state
  - control 2-2
  - generation 2-4
  - Local Bus 2-3
  - PCI Bus 2-3
- write
  - accesses 2-1, 2-12
  - cycles 3-14
  - I/O command 2-1
  - Local Bus accesses 2-4
  - memory command 2-1
  - memory write and invalidate 2-1
  - PCI Target 3-2
  - PCI Target Command codes 2-1
  - serial EEPROM operation 2-7

### Mailbox

- register 0 (MBOX0) 11-29
- register 1 (MBOX1) 11-29
- register 2 (MBOX2) 11-29
- register 3 (MBOX3) 11-29
- register 4 (MBOX4) 11-29
- register 5 (MBOX5) 11-29
- register 6 (MBOX6) 11-29
- register 7 (MBOX7) 11-30

### mailbox registers

- 1-8, 6-2, 11-29-11-30

**management, pointer**

7-2

**map**

PCI software 3-12, 5-10

PCI Target 1-7

remap PCI-to-Local addresses 3-12, 5-10

See *Also* mapping and remap

serial EEPROM memory 2-11, 4-11

**mapping**

EROMBA 11-23

LAS0BA 11-19

LAS1BA 11-28

PCI-to-Local 3-12, 5-10

register address 11-2-11-6

**MARBR**

11-20

**Master Command codes** 2-1, 4-1**Master/Target abort interrupt** 6-2**maximum rating** 13-1**MBOX**

0 11-29

1 11-29

2 11-29

3 11-29

4 11-29

5 11-29

6 11-29

7 11-30

**MDREQ#**

1-13, 12-7

**memory**

accesses 11-10, 11-11, 11-12

address spaces 11-11, 11-12, 11-19, 11-27

base address 11-10, 11-11, 11-12

commands

aliased 2-1, 4-1

basic 2-1, 4-1

disabled 8-1

local spaces 1-6

location of registers 11-12

mapped configuration registers 11-2

mapping 11-19, 11-27

PCI 8-2, 11-16

PCI Target transfer 3-11, 5-9

PCIBAR0 11-10

PCIBAR2 11-11

PCIBAR3 11-12

PCIBAR4 11-12

PCIBAR5 11-12

posted writes (PMW) 1-7

prefetchable 11-11, 11-12

read 2-1, 4-1

registers, location 11-10, 11-11

remap 11-19

serial EEPROM map 2-11, 4-11

space indicator 11-10, 11-11, 11-12, 11-19, 11-27

spaces 11-7, 11-19, 11-27

write 2-1, 4-1

write and invalidate 1-8, 11-7, 11-9

write and invalidate, PCI Initiator 3-7, 5-8

write transfers 11-9

**messages**

inbound 7-1

outbound 7-1

**messaging** 1-4**Messaging Queue Configuration register (MQCR)** 11-41**Messaging Queue registers**

11-40-11-43

address mapping 11-6

**Mode/DMA Arbitration register (MARBR)**

11-20

**MODE[1:0]**

1-13, 12-5

**Motorola**

adapters for telecom and networking applications 1-4

high performance design 1-4

MPC850 1-1, 1-6, 2-4, 3-9, 3-18

direct connect interface 1-6

PowerQUICC adapter design 1-3, 1-4

RISC processors 1-6

MPC860 1-1, 1-6, 2-3, 2-4, 3-9, 3-18, 3-34, 3-35

direct connect interface 1-6

PowerQUICC adapter design 1-3, 1-4

RISC processors 1-6

**MQCR**

11-41

**Multiplexed mode**

featured in PCI 9054 1-1

input and output, user 6-5

interface pin 12-1

LED 9-3

**N****NANDTREE**

compared to JTAG 12-15

test access method 12-15-12-19

**new capabilities**

functions support 11-8

linked list 8-1

Next\_Cap pointer 9-4

## New Capabilities Pointer register (CAP\_PTR) to PCI

---

pointer 11-1  
register 8-1  
structure 1-7, 2-10, 4-11, 9-4, 10-1  
support bit 8-1  
VPD 10-1

### New Capabilities Pointer register (CAP\_PTR)

new capabilities  
register 11-14

### Non-multiplexed mode

featured in PCI 9054 1-1  
interface pin 12-1

## O

### OFHPR

11-42

### OFTPR

11-42

### 176-pin PQFP

See PQFP

### on-the-fly Big/Little Endian conversion

3-11, 5-10

### operating ranges

13-1

### OPHPR

11-42

### OPQIM

11-40

### OPQIS

11-40

### OPTPR

11-43

### OQP

11-40

### ordering instructions A-1

### Outbound

Free Head Pointer register (OFHPR) 11-42  
free list FIFO 7-5  
free queue 7-5  
Free Tail Pointer register (OFTPR) 11-42  
OFHPR 7-2  
OFTPR 7-2  
OPHPR 11-42  
Post Head Pointer register (OPHPR) 11-42  
post queue 7-5  
post queue FIFOs 7-3  
Post Queue Interrupt Mask register (OPQIM) 11-40  
Post Queue Interrupt Status register (OPQIS) 11-40  
Post Tail Pointer register (OPTPR) 11-43  
Queue Port register (OQP) 11-40

### output, user

6-5

## P

### P2LDBELL

11-30

### package specs

14-1-14-8

### PAR

1-9, 12-4

### partial Lword accesses 4-4

### Pause Timer

3-25, 5-23

### PBGA

Local Bus types 2-2, 4-2  
ordering instructions A-1  
package layout 14-6  
package mechanical dimensions 14-4  
PCB layout suggested land pattern 14-5  
pinout 14-7-14-8

## PCI

applications 1-4  
arbitration 2-2, 4-2  
Built-In Self Test register (PCIBISTR) 11-10  
Cache Line Size register (PCICLSR) 11-9  
Cardbus CIS Pointer register (PCICIS) 11-12  
Class Code register (PCICCR) 11-9  
Command Codes register (CNTRL) 11-33  
Command register (PCICR) 11-7  
Configuration ID register (PCIIDR) 11-7  
device 1-9  
Dual Address cycle 2-1, 4-1  
Expansion ROM Base register (PCIERBAR) 11-13  
Hardcoded Configuration ID register (PCIHIDR) 11-33  
Hardcoded Revision ID register (PCIHREV) 11-33  
Header Type register (PCIHTR) 11-9  
Interrupt Line register (PCIILR) 11-14  
Interrupt Pin register (PCIIPR) 11-14  
Master Command codes 2-1, 4-1  
Max\_Lat register (PCIMLR) 11-14  
Min\_Gnt register (PCIMGR) 11-14  
Revision ID register (PCIREV) 11-8  
Status register (PCISR) 11-8  
Subsystem ID register (PCISID) 11-13  
Subsystem Vendor ID register (PCISVID) 11-12  
Vital Product Data Address register (PVPDAD) 11-18  
Vital Product Data Control register (PVPDCNTL) 11-18  
Vital Product Data Next Capability Pointer register  
(PVPD\_NEXT) 11-18

## PCI 9054

compared with PCI 9050 and 9080 1-8  
major features 1-5-1-7

## PCI 9054\_AA revision 12-6

## PCI 9054\_AB revision 2-7, 4-7, 12-1, 12-6

## PCI Base Address register

PCIBAR0 11-10  
PCIBAR1 11-11  
PCIBAR2 11-11  
PCIBAR3 11-12  
PCIBAR4 11-12  
PCIBAR5 11-12

## PCI Bus

accesses 1-6, 2-4  
accessibility of spaces 5-10  
accessible spaces 3-12  
board healthy 9-2  
cycles 2-1, 4-1  
drivers 8-1  
embedded host design 1-5  
features 1-1  
FIFO, response to 5-2  
input RST# 3-1, 5-1  
internal registers, access to 2-11, 4-12  
Latency Time register (PCILTR) 11-9  
Latency Timer 3-17, 11-2, 11-9  
Little Endian mode 2-5, 2-7, 4-5  
Master 1-3  
Masters 13-3, 13-4, 13-5, 13-6  
PCI Target accesses 4-4  
PCI Target example 3-14, 5-12  
PCI Target lock 3-10, 5-9  
PCI Target operation 3-1, 5-1  
PCI Target transfer 3-11, 5-9  
PCIMLR 11-14  
PMCSR 11-16  
power management interface specification 8-1  
response to FIFO 3-2  
software reset 3-1, 5-1  
system bus interface pins 12-3-12-4  
Vcc 1-8  
VPD 10-1  
wait states 2-3, 4-3

## PCI Initiator

3-11, 5-9  
access 3-3, 5-3  
access to PCI Bus 12-11, 12-14  
configuration 3-5, 5-5  
Dual Address cycle 3-6, 5-6  
FIFO 12-7, 12-9, 12-12

FIFOs 3-3-3-4, 5-4

I/O 3-5, 5-5

configuration access 3-5, 5-5

decode 3-3, 5-4

interrupts 6-4

Local Bus Address phase 12-6, 12-9, 12-12

Local Bus cycles 2-2, 4-2

Local Bus data parity 2-5, 4-5

Local Bus Endian mode 2-5, 2-7, 4-5, 4-7

Local Bus Read accesses 2-4, 4-4

Master Command codes 2-1, 4-1

memory 3-3, 5-4

access 3-4, 5-4

write and invalidate 3-7, 5-8

operation 3-1, 5-1

PCI Arbitration 2-2, 4-2

PCI Target abort 3-7, 5-6-5-7

reads 3-4-3-5, 5-5

wait states, Local Bus 2-3, 4-3

writes 3-4, 5-4

## PCI Initiator register (DMDAC) 11-28

## PCI Latency Timer

PCI Target 1-4

## PCI NMI, all modes

6-4

## PCI SERR#, all modes

6-4

## PCI Target

1-1, 1-6

abort 3-7, 5-6-5-7

accesses to 8- or 16-bit Local Bus 2-4, 4-4

atomic operation 1-14

Big Endian/Little Endian cycle reference table 2-5, 4-5

BTERM# input 2-3, 4-3

bursting 1-7

Command code 2-1, 4-1

Data Pipe Architecture 1-1

description 1-4

Direct Data Transfer mode 3-1-3-16, 5-1-5-14

during serial EEPROM initialization 2-7, 2-11, 4-7, 4-11

example 5-12-5-13

FIFO depth 1-8

FIFO full or empty, response to 3-2, 5-2

FIFOs depth 1-5

initialization, local bus 3-12, 5-10

Keep Bus mode 1-7

Local Bus Big Endian/Little Endian mode accesses 2-7, 4-7

Local Bus cycles 2-2, 4-2

lock 3-10, 5-9

## PCI Target Command code to pins

---

operation 3-9-3-15, 5-8-5-13  
partial Lword accesses 2-4, 4-4  
PCI Latency Timer 1-4  
power management 8-1  
Power mode example 8-2  
priority 3-15, 5-13  
Read Ahead mode 1-6  
response to FIFO full or empty 3-2  
space 0 enable in LAS0BA 11-19  
space 1 enable in LAS1BA 11-28  
transactions 1-6  
transfer 3-11  
transfer size 3-14  
wait states 2-3, 4-3

### PCI Target Command code

memory write and invalidate 4-1

#### PCIBAR0

11-10

#### PCIBAR1

11-11

#### PCIBAR2

3-12, 5-10, 11-11

#### PCIBAR3

3-12, 5-10, 11-12

#### PCIBAR4

11-12

#### PCIBAR5

11-12

#### PCIBISTR

11-10

#### PCICCR

11-9

#### PCICIS

11-12

#### PCICLSR

11-9

#### PCICR

3-1, 5-1, 11-7

#### PCIERBAR

3-12, 5-10, 11-13

#### PCIHIDR

2-7, 4-7, 11-33

#### PCIHREV

11-33

#### PCIHTR

11-9

#### PCIIDR

2-7, 4-7, 11-7

#### PCIILR

11-14

#### PCIIPR

11-14

#### PCILTR

11-9

#### PCIMGR

11-14

#### PCIMLR

11-14

#### PCIREV

11-8

#### PCISID

11-13

#### PCISR

11-8

#### PCISVID

2-7, 4-7, 11-12

#### PCI-to-Local doorbell interrupt 6-3

#### PCLK

1-9, 12-4

#### PERR#

1-9, 12-4

#### PICMG 1-3

#### pinout

PBGA 14-6, 14-7-14-8

PQFP 14-3

specs 14-1-14-8

#### pins

AD[31:0] 1-9, 12-3

ADS# 1-10, 12-9, 12-12

ALE 1-10, 12-12

BB# 1-10, 12-6

BDIP# 1-10, 12-6

BG# 1-10, 12-6

BI# 1-10, 12-6

BIGEND# 1-10, 12-6, 12-9, 12-12

BLAST# 1-10, 12-9, 12-12

BR# 1-10, 12-6

BREQi 1-10, 12-9, 12-12

BREQo 1-10, 12-9, 12-12

BTERM# 1-10, 12-9, 12-12

BURST# 1-11, 12-6

C and J Bus modes 12-9-12-14

C/BE[3:0]# 1-9, 12-3

CCS# 1-11, 12-5

common to all bus modes 12-2-12-5

DACK0# 1-14, 12-5

DEN# 1-11

DEVSEL# 1-9, 12-3

DMPAF 1-11, 1-13, 12-7, 12-9, 12-12

DP[0:3] 1-11, 12-6, 12-9

- DP[3:0] 1-11, 12-13
- DREQ0# 1-11, 1-14, 12-5
- DT/R# 1-11, 12-13
- EECS 1-11, 12-3
- EEDI 1-11, 12-3
- EEDO 1-11, 12-3
- EESK 1-11, 12-3
- ENUM# 12-3
- EOT# 1-13, 12-7, 12-9, 12-12
- FRAME# 1-9, 12-3, 12-5
- GNT# 12-3
- IDSEL 1-9, 12-3
- INTA# 1-9, 12-4
- IRDY# 1-9, 12-4
- LA[0:31] 1-11, 12-7
- LA[28:2] 1-11, 12-13
- LA[31:2] 1-11, 3-14, 12-10
- LAD[31:0] 1-11, 12-13
- LBE[3:0]# 1-12, 12-10, 12-14
- LCLK 1-12, 12-5
- LD[0:31] 1-12, 12-7
- LD[0:7] 3-14
- LD[16:23] 3-14
- LD[24:31] 3-14
- LD[31:0] 1-12, 12-10
- LD[8:15] 3-14
- LEDOn/LEDin 1-12, 12-5
- LFRAME# 1-12
- LHOLD 1-12, 12-14
- LHOLDA 1-13, 12-11, 12-14
- LINT# 1-13, 12-5
- LLOCKi# 1-13, 1-14, 12-5
- LLOCKo# 1-13, 1-14, 12-5
- Local Bus independent interface 12-5
- LOCK# 1-9, 12-4
- LRESETo# 1-13, 12-5
- LSERR# 1-13, 12-11, 12-14
- LW/R# 1-13, 12-14
- M Bus mode 12-6-12-8
- MDREQ# 1-13, 12-7
- MODE[1:0] 1-13, 12-5
- NANDTREE sequential interconnection 12-16-12-19
- PAR 1-9, 12-4
- PCI system bus interface 12-3-12-4
- PCLK 1-9, 12-4
- PERR# 1-9, 12-4
- PME# 1-9, 12-4
- power and ground 12-2
- processor-independent interface 12-5
- RD/WR# 1-13, 12-7
- READY# 1-13, 4-3, 4-4, 4-7, 4-11, 5-2, 5-4, 5-5, 5-7, 5-8, 5-14, 5-22, 12-11, 12-14, 13-4, 13-6
- REQ# 12-4
- RETRY# 1-13, 12-7
- RST# 1-9, 12-4
- serial EEPROM 12-3
- SERR# 1-9, 12-4
- STOP# 1-9, 12-4
- TA# 1-13, 3-4, 3-5, 3-6, 3-7, 3-9, 3-10, 3-12, 12-8 2-3, 3-2
- TEA# 1-14, 12-8
- TEST 1-14, 12-2
- TRDY# 1-9, 12-4
- TS# 1-14, 12-8
- TSIZ[0:1] 1-14, 3-14, 12-8
- USERi 1-14, 12-5
- USERo 1-14, 12-5
- VDD 1-14, 12-2
- VSS 1-14, 12-2
- WAIT# 1-10, 1-14, 12-6, 12-11, 12-14
- PLX Technology, Inc.**
  - company background 1-3
  - product ordering instructions A-1
  - representatives and distributors A-1
  - technical support A-1
- PMC**
  - 11-15
- PMCAPID**
  - 11-14
- PMCSR**
  - 11-16
- PMCSR\_BSE**
  - 11-16
- PMDATA**
  - 11-17
- PME#**
  - 1-9, 12-4
- PME#\_Status**
  - 8-2
- PMNEXT**
  - 11-15
- pointer management** 7-2
- post queue** 7-3, 7-5
- Power Management**
  - 8-1-8-2
  - Capabilities register (PMC) 11-15
  - Capability ID register (PMCAPID) 11-14
  - Control/Status register (PMCSR) 11-16
  - Data register (PMDATA) 11-17
  - functional description 8-1-8-2

Next Capability Pointer register (PMNEXT) 11-15  
PMCSR Bridge Support Extension register  
(PMCSR\_BSE) 11-16  
Power mode example 8-2  
system changes 8-2  
wake-up request example 8-2

**PowerQUICC**

1-1, 1-4, 1-6  
adapter design 1-3  
IDMA 1-4

**PQFP**

Local Bus types 2-2, 4-2  
ordering instructions A-1  
package mechanical dimensions 14-1  
PCB layout suggested land pattern 14-2  
pinout 14-3

**preempt**

deadlock solution 3-16, 5-14

**prefetch**

counter 1-1, 1-6  
LAS0RR 11-19  
LAS1RR 11-27  
PCI Target read 1-4  
PCI Target Read Ahead mode 3-10, 5-9  
PCIBAR0 11-10  
PCIBAR2 11-11  
PCIBAR3 11-12  
programmable counter 1-6, 1-8  
Read Ahead mode 1-6

**priority**

control Local Bus 1-4  
DMA 3-20, 5-20

**Processor-independent interface pins** 12-5

**programmable**

Local Bus 1-6  
prefetch counter  
Master burst length 1-6  
Read Ahead mode 1-6

**PROT\_AREA**

11-22

**VVPD\_NEXT**

11-18

**VVPDAD**

11-18

**VVPDATA**

PCI VPD Data register 11-18

**VVPDCNTL**

11-18

**Q**

**QBAR**

11-41

**QSR**

11-43

**Queue**

Base Address register (QBAR) 11-41  
Status/Control register (QSR) 11-43

**R**

**Range registers**

Local Address Space 0 for PCI-to-Local Bus  
(LAS0RR) 11-19  
Local Address Space 1 for PCI-to-Local Bus  
(LAS1RR) 11-27  
Local Range Register for PCI Initiator-to-PCI (DMRR) 11-  
25

**Range, Expansion ROM register (EROMRR)** 11-23

**ranges**

operating 13-1, 13-2

**RD/WR#**

1-13, 12-7

**read**

accesses 2-1, 2-12, 4-1, 4-12  
bursts 1-9  
configuration 1-9  
configuration command 2-1, 4-1  
deferred 1-8  
deferred PCI Initiator 11-1  
DMA 1-5  
FIFOs 1-7, 1-8, 2-4, 3-4, 3-9, 4-4, 5-4, 5-8  
I/O command 2-1, 4-1  
in registers 11-7-11-43  
Local Bus accesses 2-4, 4-4  
LW/R# 1-13  
memory  
command 2-1, 4-1  
line 2-1, 4-1  
multiple 2-1, 4-1  
PAR 1-9  
PCI  
initialization 3-12, 5-10  
Power mode example 8-2  
PCI Initiator 1-5  
PCI Target 1-5, 1-8, 3-2, 3-10, 5-2, 5-9  
PCI Target Command codes 2-1, 4-1  
PCI Target prefetching 1-4  
PCI Target transfer 3-11, 5-9  
PCI-to-Local Delayed mode 1-1  
random read and write 10-2  
Read Ahead mode 2-4, 3-10, 4-4, 5-9

- READY# 1-13
- sequential read only 10-1
- serial EEPROM 1-11, 2-10
  - accidental write 10-2
  - control 1-8, 2-9, 4-9
  - operation 2-7, 4-7
- transactions 1-9
- VPD 10-1
  - data 10-1
  - random read and write 10-2
  - registers 10-1
  - sequential read only 10-1
  - serial EEPROM partitioning 10-1
- Read Ahead mode**
  - in burst read cycles 2-4, 4-4
  - PCI Target 3-10, 5-9
  - supported 1-6
- Read Delayed mode** 3-10, 5-9
- READY#** 1-13, 4-3, 5-2, 5-4, 5-5, 5-7, 5-8, 5-14, 5-22
  - 12-11, 12-14
  - Bus mode 13-4, 13-6
  - input 4-4
  - output 13-6
  - serial EEPROM initialization 4-7, 4-11
- reconfiguration, system** 9-1
  - See *Also* configuration
- recovery states, J mode only** 4-4
- registers**
  - addresses 1-8
  - BIGEND 11-21
  - CAP\_PTR 11-14
  - CNTRL 11-33
  - DMAARB 11-39
  - DMACSR0 11-38
  - DMACSR1 11-38
  - DMADAC0 11-39
  - DMADAC1 11-39
  - DMADPR0 11-35
  - DMADPR1 11-37
  - DMALADR0 11-35
  - DMALADR1 11-37
  - DMAMODE0 11-34
  - DMAMODE1 11-36
  - DMAPADR0 11-35
  - DMAPADR1 11-37
  - DMASIZ0 11-35
  - DMASIZ1 11-37
  - DMATHR 11-39
  - DMCBFGA 11-27
  - DMDAC 11-28
  - DMLBAI 11-25
  - DMLBAM 11-25
  - DMPBAM 11-26
  - DMRR 11-25
  - doorbell 6-2, 11-30
  - EROMBA 11-23
  - EROMRR 11-23
  - HS\_CNTL 11-17
  - HS\_CSR 11-17
  - HS\_NEXT 11-17
  - IFHPR 11-41
  - IFTPR 11-41
  - INTCSR 11-31
  - IPHPR 11-42
  - IPTPR 11-42
  - IQP 11-40
  - L2PDBELL 11-30
  - LAS0BA 11-19
  - LAS0RR 11-19
  - LAS1BA 11-28
  - LAS1RR 11-27
  - LBRD0 11-24
  - LBRD1 11-28
  - LMISC 11-22
  - mailbox 6-2, 11-29-11-30
  - MARBR 11-20
  - MBOX0 11-29
  - MBOX1 11-29
  - MBOX2 11-29
  - MBOX3 11-29
  - MBOX4 11-29
  - MBOX5 11-29
  - MBOX6 11-29
  - MBOX7 11-30
  - MQCR 11-41
  - OFHPR 11-42
  - OFTPR 11-42
  - OPHPR 11-42
  - OPQIM 11-40
  - OPQIS 11-40
  - OPTPR 11-43
  - OQP 11-40
  - P2LDBELL 11-30
  - PCIBAR0 11-10
  - PCIBAR1 11-11
  - PCIBAR2 11-11
  - PCIBAR3 11-12
  - PCIBAR4 11-12
  - PCIBAR5 11-12
  - PCIBISTR 11-10

PCICCR 11-9  
PCICIS 11-12  
PCICLSR 11-9  
PCICR 11-7  
PCIERBAR 11-13  
PCIHIDR 2-7, 4-7, 11-33  
PCIHREV 11-33  
PCIHTR 11-9  
PCIIDR 2-7, 4-7, 11-7  
PCIILR 11-14  
PCIIPR 11-14  
PCILTR 11-9  
PCIMGR 11-14  
PCIMLR 11-14  
PCIREV 11-8  
PCISID 11-13  
PCISR 11-8  
PCISVID 2-7, 4-7, 11-12  
PMC 11-15  
PMCAPID 11-14  
PMCSR 11-16  
PMCSR\_BSE 11-16  
PMDATA 11-17  
PMNEXT 11-15  
PROT\_AREA 11-22  
VVPD\_NEXT 11-18  
VVPDAD 11-18  
VVPDATA 11-18  
VVPDCNTL 11-18  
QBAR 11-41  
QSR 11-43

**remap**  
PCI-to-Local addresses 3-12, 5-10  
See *Also* map and mapping

**REQ#**  
12-4

**reset, software** 3-1, 5-1

**RETRY#**  
1-13, 3-5, 12-7  
capability 3-5

**ring management** 1-3

**RST#**  
1-9, 3-1, 5-1, 12-4

**runtime registers** 11-29-11-33  
address mapping 11-4

## S

**scatter/gather** 3-20, 3-21, 5-16  
list management 1-3  
ring management 1-3

**SDMA operation** 3-9

**Serial EEPROM**  
Control register (CNTRL) 11-33  
Write-Protected Address Boundary register (PROT\_AREA) 11-22

**serial EEPROM**  
2-7-2-15, 3-1, 4-7-4-16, 5-1  
accidental write to 10-2  
address decode enable 11-23  
address pointer 11-1  
base class code 11-9  
chip select 1-11  
clock pin 1-11  
device ID 11-7  
device ID registers 2-7, 4-7  
extra long load 2-10, 4-10  
extra long load registers 2-10, 4-10  
Hot Swap ID 11-17  
initialization 2-7, 2-11, 4-7, 4-11  
interface 1-1, 1-6  
interface pins 12-1, 12-3  
internal register access 2-11, 4-11  
interrupt pin register 11-14  
long load 2-8-2-9, 4-8-4-9  
long load registers 2-9, 4-9  
memory map 2-11, 4-11  
new capabilities function 2-10, 4-11  
Next\_Cap pointer 11-17  
operation 2-7, 4-7  
PCI Bus access to internal registers 2-11  
PCI Bus, access to internal registers 4-12  
random read and write 10-2  
read control 1-8  
read data 1-11  
read-only portion 10-1  
recommended 2-10, 4-11  
register level programming interface 11-9  
revision ID 11-8  
simple VPD read to 10-2  
simple VPD write to 10-2  
software reset 3-1, 5-1  
subclass code 11-9  
subsystem ID 11-13  
support 1-8  
timing diagrams 2-13-2-16, 4-13-4-16  
vendor ID 11-7

vendor ID registers 2-7, 4-7  
 VPD address 11-18  
 VPD stored in 10-1  
 write data 1-11  
 Write-Protected Address Boundary register (PROT\_AREA) 11-1  
 writes to 10-2

**SERR#**  
 1-9, 6-4, 12-4

**Shuttle mode, DMA** 1-3

**signal listing** 1-9

**signal name**  
 C and J Bus modes 12-9-12-14  
 Local Bus mode independent interface pins 12-5  
 M Bus mode 12-6-12-8  
 PCI system bus interface pins 12-3-12-4  
 power and ground pins 12-2  
 Processor-independent interface pins 12-5  
 serial EEPROM interface pins 12-3

**signal specs** 14-1-14-8

**signaling** 1-5  
 Local Bus 1-8  
 support 1-1

**signals**  
 synchronous inputs 13-3, 13-4  
 synchronous outputs 13-5, 13-6

**single address**  
 block DMA initialization 5-7

**software reset** 3-1, 5-1

**specifications**  
 See electrical specifications or general electrical specifications

**states, recovery J mode only** 4-4

**STOP#**  
 1-9, 12-4

**synchronous**  
 inputs 13-3, 13-4  
 outputs 13-5, 13-6

**system reconfiguration** 9-1  
 See *Also* configuration

**T**

**TA#**  
 1-13, 2-3, 3-2, 3-4, 3-5, 3-6, 3-7, 3-9, 3-10, 3-16, 3-24, 12-8  
 Bus mode 13-3, 13-5  
 input 2-4  
 output 13-5  
 serial EEPROM initialization 2-7, 2-11

**Target**  
 See PCI Target

**TEA#**  
 1-14, 3-12, 12-8

technical support A-1

**TEST**  
 1-14, 12-2

**thermal resistance, package** 13-1

**Timer, pause** 3-25, 5-23

**timing diagrams**  
 C and J modes 5-24-5-82  
 DMA 3-51-3-65, 5-63-5-72, 5-81-5-82  
 dual address 3-18  
 Local Bus 3-26, 5-24  
 M mode 3-26-3-65  
 PCI Initiator 3-27-3-35, 5-25-5-43, 5-73-5-76  
 PCI Target 3-36-3-50, 5-44-5-62, 5-77-5-80  
 serial EEPROM 2-13-2-16, 4-13-4-16

**transfer size** 3-14

**TRDY#**  
 1-9, 12-4

**TS#**  
 1-14, 12-8

**TSIZ[0:1]**  
 1-14, 2-4, 3-14, 12-8  
 data bus 3-14

**225-pin PBGA**  
 See PBGA

**Type 0, configuration cycle**  
 3-5, 5-5

**Type 1, configuration cycle**  
 3-5, 5-5

**U**

**unaligned DMA transfer support**  
 1-7

**User I/O** 6-1-6-5

**user input and output**  
 See *Also* I/O

**USERi**  
 1-14, 12-5

**USERo**  
 1-14, 12-5

**V**

**VDD** 1-14, 12-2

**vendor ID** 11-7

**Vital Product Data (VPD)**  
 1-1, 10-1-10-2  
 address 11-2, 11-18  
 address pointer 11-1  
 capabilities register 10-1  
 data 11-2

## VSS

### to zero wait state burst operation

---

- data register 11-18
- ID register 11-18
- internal register access 2-11, 4-8, 4-11
- last item in capabilities linked list 11-18
- new capabilities function 2-10
- Next\_Cap pointer 11-18
- PCI register 11-1
- random read and write 10-2
- read or write serial EEPROM 2-8
- sequential read only 10-1
- serial EEPROM
  - accesses 11-22
  - values programmed with 2-9, 4-9
  - VPD partitioning 10-1
- support in 9054 1-5, 1-8
- Write-Protected Address Boundary register (PROT\_AREA) 11-1

## VSS

1-14, 12-2

## W

### wait state

- control 2-2, 4-3
- cycle control 11-7
- generation 1-8, 1-10, 1-13, 2-4, 3-11, 4-4
- Local Bus 1-1, 2-3, 4-3
- PCI Bus 2-3, 4-3
- zero 1-1, 1-5

### WAIT#

1-10, 1-14, 12-6, 12-11, 12-14

### width control 1-4

### write

- accesses 2-1, 2-12, 4-1, 4-12
- and invalidate, memory 1-8
- bursts 1-9
- configuration 1-9
- configuration command 2-1, 4-1
- cycles 10-1
- DMA 1-5
- FIFOs 1-7, 1-8, 2-4, 3-4, 3-9, 4-4, 5-4, 5-8
- flush pending 3-10, 5-9
- I/O command 2-1, 4-1
- in registers 11-7-11-43
- Local Bus accesses 2-4, 4-4
- LW/R# 1-13
- memory
  - command 2-1, 4-1
  - write and invalidate 2-1, 4-1
- PCI Initiator 1-5
- PCI power management 8-1
- PCI Power mode example 8-2

- PCI Target 1-5, 1-8, 3-2, 5-2
- PCI Target Command codes 2-1, 4-1
- PCI Target transfer 3-11, 5-9
- posted memory (PMW) 1-7
- random read and write 10-2
- READY# 1-13
- serial EEPROM 1-11, 2-10, 4-11, 10-2
  - accidental 10-2
  - control 2-9, 4-9
  - operation 2-7, 4-7
- transactions 1-9
- VPD 10-1
  - data 10-1
  - random read and write 10-2
  - serial EEPROM partitioning 10-1
  - simple 10-2
- wake-up request example 8-2

## Z

### zero wait state burst operation

1-1, 1-5