TEXAS INSTRUMENTS

PCI2250

PCI-to-PCI Bridge

Data Manual

TEXAS INSTRUMENTS

Printed in U.S.A., 12/99 SCPS051

PCI2250
PCI-to-PCI Bridge

Data Manual

Literature Number: SCPS051
December 1999

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

Contents

List of Illustrations

List of Tables

1 Introduction

1.1 Description

The Texas Instruments PCI2250 PCI-to-PCI bridge provides a high performance connection path between two peripheral component interconnect (PCI) buses. Transactions occur between masters on one PCI bus and targets on another PCI bus, and the PCI2250 allows bridged transactions to occur concurrently on both buses. The bridge supports burst-mode transfers to maximize data throughput, and the two bus traffic paths through the bridge act independently.

The PCI2250 bridge is compliant with the *PCI Local Bus Specification*, and can be used to overcome the electrical loading limits of 10 devices per PCI bus and one PCI device per expansion slot by creating hierarchical buses. The PCI2250 provides two-tier internal arbitration for up to four secondary bus masters and may be implemented with an external secondary PCI bus arbiter.

The PCI2250 provides compact-PCI (CPCI) hot-swap extended capability, which makes it an ideal solution for multifunction compact-PCI cards and adapting single function cards to hot-swap compliance.

The PCI2250 bridge is compliant with the *PCI-to-PCI Bridge Specification*. It can be configured for positive decoding or subtractive decoding on the primary interface, and provides several additional decode options that make it an ideal bridge to custom PCI applications. Two extension windows are included, and the PCI2250 provides decoding of serial and parallel port addresses.

The PCI2250 is compliant with *PCI Power Management Interface Specification Revisions 1.0 and 1.1*. Also, the PCI2250 offers PCI $\overline{\text{CLKRUN}}$ bridging support for low-power mobile and docking applications. The PCI2250 has been designed to lead the industry in power conservation. An advanced CMOS process is utilized to achieve low system power consumption while operating at PCI clock rates up to 33 MHz.

1.2 Features

The PCI2250 supports the following features:

- Configurable for *PCI Power Management Interface Specification Revision 1.0 or 1.1* support
- Compact-PCI friendly silicon as defined in the *Compact-PCI Hot Swap Specification*
- 3.3-V core logic with universal PCI interface compatible with 3.3-V and 5-V PCI signaling environments
- Two 32-bit, 33-MHz PCI buses
- Provides internal two-tier arbitration for up to four secondary bus masters and supports an external secondary bus arbiter
- Burst data transfers with pipeline architecture to maximize data throughput in both directions
- Provides programmable extension windows and port decode options
- Independent read and write buffers for each direction
- Provides five secondary PCI clock outputs
- Predictable latency per *PCI Local Bus Specification*
- Propagates bus locking
- Secondary bus is driven low during reset
- Provides VGA palette memory and I/O, and subtractive decoding options
- Advanced submicron, low-power CMOS technology

- Fully compliant with *PCI-to-PCI Bridge Architecture Specification*
- Packaged in 160-pin QFP (PCM) and 176-pin thin QFP (PGF)

1.3 Related Documents

- *Advanced Configuration and Power Interface (ACPI) Revision 1.0*
- *PCI Local Bus Specification Revision 2.2*
- *PCI Mobile Design Guide, Revision 1.0*
- *PCI-to-PCI Bridge Architecture Specification Revision 1.1*
- *PCI Power Management Interface Specification Revision 1.1*
- *PICMG Compact-PCI Hot Swap Specification Revision 1.0*

1.4 Ordering Information

ORDERING NUMBER	NAME	VOLTAGE	PACKAGE
PCI2250	PCI–PCI Bridge	3.3 V, 5-V tolerant I/Os	160-pin QFP 176-pin LQFP

2 Terminal Descriptions

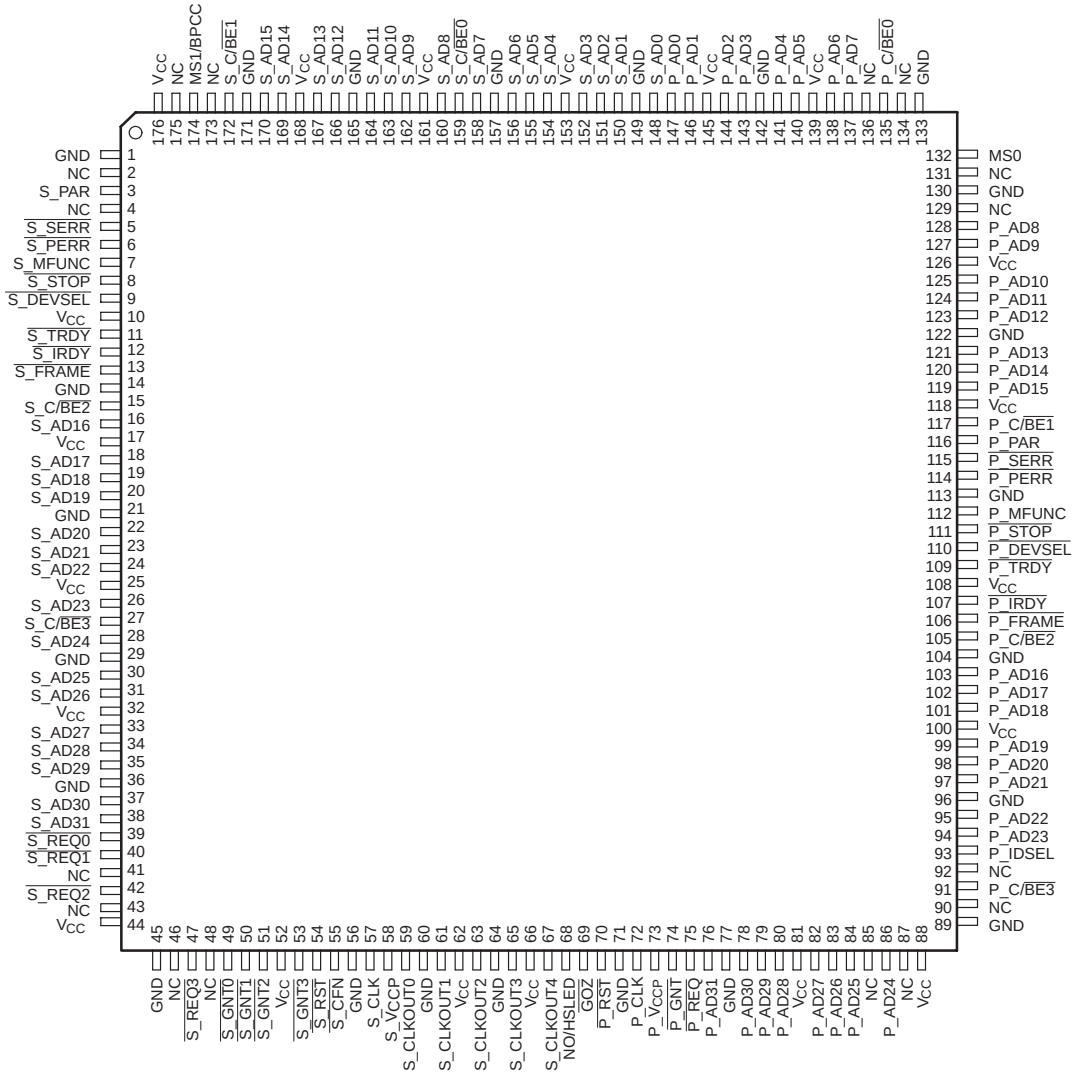


Figure 2–1. PCI2250 PGF LQFP Terminal Diagram

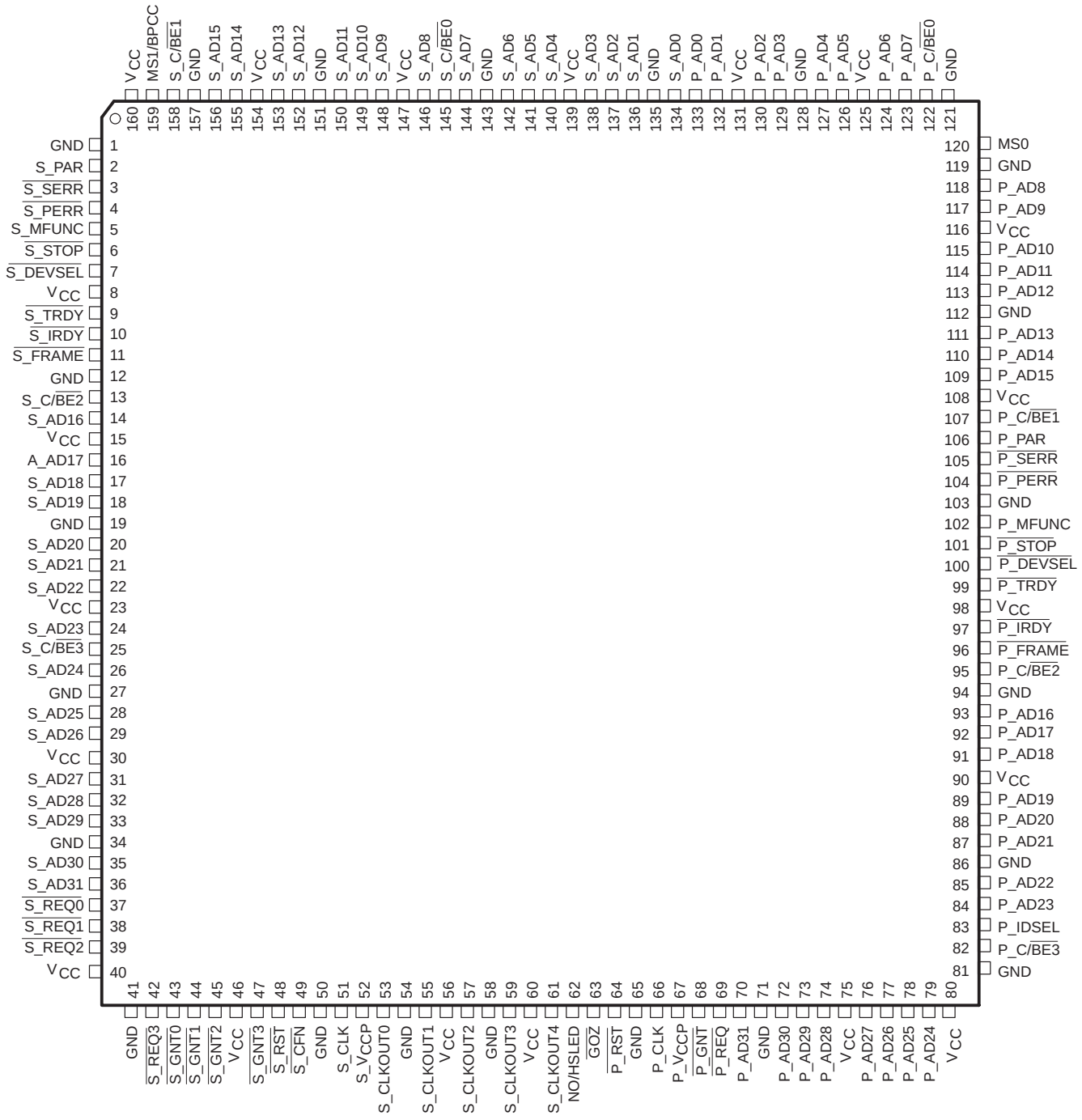


Figure 2–2. PCI2250 PCM PQFP Terminal Diagram

Table 2–1. PGF LQFP Signal Names Sorted by Terminal Number

TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME
1	GND	45	GND	89	GND	133	GND
2	NC	46	NC	90	NC	134	NC
3	S_PAR	47	S_REQ3	91	P_C/BE3	135	P_C/BE0
4	NC	48	NC	92	NC	136	NC
5	S_SERR	49	S_GNT0	93	P_IDSEL	137	P_AD7
6	S_PERR	50	S_GNT1	94	P_AD23	138	P_AD6
7	S_MFUNC	51	S_GNT2	95	P_AD22	139	V_{CC}
8	S_STOP	52	V_{CC}	96	GND	140	P_AD5
9	S_DEVSEL	53	S_GNT3	97	P_AD21	141	P_AD4
10	V_{CC}	54	S_RST	98	P_AD20	142	GND
11	S_TRDY	55	S_CFN	99	P_AD19	143	P_AD3
12	S_IRDY	56	GND	100	V_{CC}	144	P_AD2
13	S_FRAME	57	S_CLK	101	P_AD18	145	V_{CC}
14	GND	58	S_V_{CCP}	102	P_AD17	146	P_AD1
15	S_C/BE2	59	S_CLKOUT0	103	P_AD16	147	P_AD0
16	S_AD16	60	GND	104	GND	148	S_AD0
17	V_{CC}	61	S_CLKOUT1	105	P_C/BE2	149	GND
18	S_AD17	62	V_{CC}	106	P_FRAME	150	S_AD1
19	S_AD18	63	S_CLKOUT2	107	P_IRDY	151	S_AD2
20	S_AD19	64	GND	108	V_{CC}	152	S_AD3
21	GND	65	S_CLKOUT3	109	P_TRDY	153	V_{CC}
22	S_AD20	66	V_{CC}	110	P_DEVSEL	154	S_AD4
23	S_AD21	67	S_CLKOUT4	111	P_STOP	155	S_AD5
24	S_AD22	68	NO/HSLED	112	P_MFUNC	156	S_AD6
25	V_{CC}	69	GOZ	113	GND	157	GND
26	S_AD23	70	P_RST	114	P_PERR	158	S_AD7
27	S_C/BE3	71	GND	115	P_SERR	159	S_C/BE0
28	S_AD24	72	P_CLK	116	P_PAR	160	S_AD8
29	GND	73	P_V_{CCP}	117	P_C/BE1	161	V_{CC}
30	S_AD25	74	P_GNT	118	V_{CC}	162	S_AD9
31	S_AD26	75	P_REQ	119	P_AD15	163	S_AD10
32	V_{CC}	76	P_AD31	120	P_AD14	164	S_AD11
33	S_AD27	77	GND	121	P_AD13	165	GND
34	S_AD28	78	P_AD30	122	GND	166	S_AD12
35	S_AD29	79	P_AD29	123	P_AD12	167	S_AD13
36	GND	80	P_AD28	124	P_AD11	168	V_{CC}
37	S_AD30	81	V_{CC}	125	P_AD10	169	S_AD14
38	S_AD31	82	P_AD27	126	V_{CC}	170	S_AD15
39	S_REQ0	83	P_AD26	127	P_AD9	171	GND
40	S_REQ1	84	P_AD25	128	P_AD8	172	S_C/BE1
41	NC	85	NC	129	NC	173	NC
42	S_REQ2	86	P_AD24	130	GND	174	MS1/BPCC
43	NC	87	NC	131	NC	175	NC
44	V_{CC}	88	V_{CC}	132	MS0	176	V_{CC}

Table 2–2. PCM LQFP Signals Sorted by Terminal Number

TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME
1	GND	41	GND	81	GND	121	GND
2	S_PAR	42	S_REQ3	82	P_C/BE3	122	P_C/BE0
3	S_SERR	43	S_GNT0	83	P_IDSEL	123	P_AD7
4	S_PERR	44	S_GNT1	84	P_AD23	124	P_AD6
5	S_MFUNC	45	S_GNT2	85	P_AD22	125	V_{CC}
6	S_STOP	46	V_{CC}	86	GND	126	P_AD5
7	S_DEVSEL	47	S_GNT3	87	P_AD21	127	P_AD4
8	V_{CC}	48	S_RST	88	P_AD20	128	GND
9	S_TRDY	49	S_CFN	89	P_AD19	129	P_AD3
10	S_IRDY	50	GND	90	V_{CC}	130	P_AD2
11	S_FRAME	51	S_CLK	91	P_AD18	131	V_{CC}
12	GND	52	S_V_{CCP}	92	P_AD17	132	P_AD1
13	S_C/BE2	53	S_CLKOUT0	93	P_AD16	133	P_AD0
14	S_AD16	54	GND	94	GND	134	S_AD0
15	V_{CC}	55	S_CLKOUT1	95	P_C/BE2	135	GND
16	S_AD17	56	V_{CC}	96	P_FRAME	136	S_AD1
17	S_AD18	57	S_CLKOUT2	97	P_IRDY	137	S_AD2
18	S_AD19	58	GND	98	V_{CC}	138	S_AD3
19	GND	59	S_CLKOUT3	99	P_TRDY	139	V_{CC}
20	S_AD20	60	V_{CC}	100	P_DEVSEL	140	S_AD4
21	S_AD21	61	S_CLKOUT4	101	P_STOP	141	S_AD5
22	S_AD22	62	NO/HSLED	102	P_MFUNC	142	S_AD6
23	V_{CC}	63	GOZ	103	GND	143	GND
24	S_AD23	64	P_RST	104	P_PERR	144	S_AD7
25	S_C/BE3	65	GND	105	P_SERR	145	S_C/BE0
26	S_AD24	66	P_CLK	106	P_PAR	146	S_AD8
27	GND	67	P_V_{CCP}	107	P_C/BE1	147	V_{CC}
28	S_AD25	68	P_GNT	108	V_{CC}	148	S_AD9
29	S_AD26	69	P_REQ	109	P_AD15	149	S_AD10
30	V_{CC}	70	P_AD31	110	P_AD14	150	S_AD11
31	S_AD27	71	GND	111	P_AD13	151	GND
32	S_AD28	72	P_AD30	112	GND	152	S_AD12
33	S_AD29	73	P_AD29	113	P_AD12	153	S_AD13
34	GND	74	P_AD28	114	P_AD11	154	V_{CC}
35	S_AD30	75	V_{CC}	115	P_AD10	155	S_AD14
36	S_AD31	76	P_AD27	116	V_{CC}	156	S_AD15
37	S_REQ0	77	P_AD26	117	P_AD9	157	GND
38	S_REQ1	78	P_AD25	118	P_AD8	158	S_C/BE1
39	S_REQ2	79	P_AD24	119	GND	159	MS1/BPCC
40	V_{CC}	80	V_{CC}	120	MS0	160	V_{CC}

Table 2–3. Signal Names Sorted Alphabetically to PGF Terminal Number

SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.
GND	1	P_AD1	146	$\overline{\text{P\_REQ}}$	75	S_CLKOUT0	59
GND	14	P_AD2	144	$\overline{\text{P\_RST}}$	70	S_CLKOUT1	61
GND	21	P_AD3	143	$\overline{\text{P\_SERR}}$	115	S_CLKOUT2	63
GND	29	P_AD4	141	$\overline{\text{P\_STOP}}$	111	S_CLKOUT3	65
GND	36	P_AD5	140	$\overline{\text{P\_TRDY}}$	109	S_CLKOUT4	67
GND	45	P_AD6	138	P_V_{CCP}	73	$\overline{\text{S\_DEVSEL}}$	9
GND	56	P_AD7	137	S_AD0	148	$\overline{\text{S\_FRAME}}$	13
GND	60	P_AD8	128	S_AD1	150	$\overline{\text{S\_GNT0}}$	49
GND	64	P_AD9	127	S_AD2	151	$\overline{\text{S\_GNT1}}$	50
GND	71	P_AD10	125	S_AD3	152	$\overline{\text{S\_GNT2}}$	51
GND	77	P_AD11	124	S_AD4	154	$\overline{\text{S\_GNT3}}$	53
GND	89	P_AD12	123	S_AD5	155	$\overline{\text{S\_IRDY}}$	12
GND	96	P_AD13	121	S_AD6	156	S_MFUNC	7
GND	104	P_AD14	120	S_AD7	158	S_PAR	3
GND	113	P_AD15	119	S_AD8	160	$\overline{\text{S\_PERR}}$	6
GND	122	P_AD16	103	S_AD9	162	$\overline{\text{S\_REQ0}}$	39
GND	130	P_AD17	102	S_AD10	163	$\overline{\text{S\_REQ1}}$	40
GND	133	P_AD18	101	S_AD11	164	$\overline{\text{S\_REQ2}}$	42
GND	142	P_AD19	99	S_AD12	166	$\overline{\text{S\_REQ3}}$	47
GND	149	P_AD20	98	S_AD13	167	$\overline{\text{S\_RST}}$	54
GND	157	P_AD21	97	S_AD14	169	$\overline{\text{S\_SERR}}$	5
GND	165	P_AD22	95	S_AD15	170	$\overline{\text{S\_STOP}}$	8
GND	171	P_AD23	94	S_AD16	16	$\overline{\text{S\_TRDY}}$	11
$\overline{\text{GOZ}}$	69	P_AD24	86	S_AD17	18	S_V_{CCP}	58
MS0	132	P_AD25	84	S_AD18	19	V_{CC}	10
MS1/BPCC	174	P_AD26	83	S_AD19	20	V_{CC}	17
NC	2	P_AD27	82	S_AD20	22	V_{CC}	25
NC	4	P_AD28	80	S_AD21	23	V_{CC}	32
NC	41	P_AD29	79	S_AD22	24	V_{CC}	44
NC	43	P_AD30	78	S_AD23	26	V_{CC}	52
NC	46	P_AD31	76	S_AD24	28	V_{CC}	62
NC	48	P_C/$\overline{\text{BE0}}$	135	S_AD25	30	V_{CC}	66
NC	85	P_C/$\overline{\text{BE1}}$	117	S_AD26	31	V_{CC}	81
NC	87	P_C/$\overline{\text{BE2}}$	105	S_AD27	33	V_{CC}	88
NC	90	P_C/$\overline{\text{BE3}}$	91	S_AD28	34	V_{CC}	100
NC	92	P_CLK	72	S_AD29	35	V_{CC}	108
NC	129	$\overline{\text{P\_DEVSEL}}$	110	S_AD30	37	V_{CC}	118
NC	131	$\overline{\text{P\_FRAME}}$	106	S_AD31	38	V_{CC}	126
NC	134	$\overline{\text{P\_GNT}}$	74	S_C/$\overline{\text{BE0}}$	159	V_{CC}	139
NC	136	P_IDSEL	93	S_C/$\overline{\text{BE1}}$	172	V_{CC}	145
NC	173	$\overline{\text{P\_IRDY}}$	107	S_C/$\overline{\text{BE2}}$	15	V_{CC}	153
NC	175	P_MFUNC	112	S_C/$\overline{\text{BE3}}$	27	V_{CC}	161
NO/HSLED	68	P_PAR	116	$\overline{\text{S\_CFN}}$	55	V_{CC}	168
P_AD0	147	$\overline{\text{P\_PERR}}$	114	S_CLK	57	V_{CC}	176

Table 2–4. Signal Names Sorted Alphabetically to PCM Terminal Number

SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.	SIGNAL NAME	TERM. NO.
GND	1	P_AD13	111	S_AD2	137	S_CLKOUT4	61
GND	12	P_AD14	110	S_AD3	138	$\overline{\text{S\_DEVSEL}}$	7
GND	19	P_AD15	109	S_AD4	140	$\overline{\text{S\_FRAME}}$	11
GND	27	P_AD16	93	S_AD5	141	$\overline{\text{S\_GNT0}}$	43
GND	34	P_AD17	92	S_AD6	142	$\overline{\text{S\_GNT1}}$	44
GND	41	P_AD18	91	S_AD7	144	$\overline{\text{S\_GNT2}}$	45
GND	50	P_AD19	89	S_AD8	146	$\overline{\text{S\_GNT3}}$	47
GND	54	P_AD20	88	S_AD9	148	$\overline{\text{S\_IRDY}}$	10
GND	58	P_AD21	87	S_AD10	149	S_MFUNC	5
GND	65	P_AD22	85	S_AD11	150	S_PAR	2
GND	71	P_AD23	84	S_AD12	152	$\overline{\text{S\_PERR}}$	4
GND	81	P_AD24	79	S_AD13	153	$\overline{\text{S\_REQ0}}$	37
GND	86	P_AD25	78	S_AD14	155	$\overline{\text{S\_REQ1}}$	38
GND	94	P_AD26	77	S_AD15	156	$\overline{\text{S\_REQ2}}$	39
GND	103	P_AD27	76	S_AD16	14	$\overline{\text{S\_REQ3}}$	42
GND	112	P_AD28	74	S_AD17	16	$\overline{\text{S\_RST}}$	48
GND	119	P_AD29	73	S_AD18	17	$\overline{\text{S\_SERR}}$	3
GND	121	P_AD30	72	S_AD19	18	$\overline{\text{S\_STOP}}$	6
GND	128	P_AD31	70	S_AD20	20	$\overline{\text{S\_TRDY}}$	9
GND	135	P_C/$\overline{\text{BE0}}$	122	S_AD21	21	S_V_{CCP}	52
GND	143	P_C/$\overline{\text{BE1}}$	107	S_AD22	22	V_{CC}	8
GND	151	P_C/$\overline{\text{BE2}}$	95	S_AD23	24	V_{CC}	15
GND	157	P_C/$\overline{\text{BE3}}$	82	S_AD24	26	V_{CC}	23
$\overline{\text{GOZ}}$	63	P_CLK	66	S_AD25	28	V_{CC}	30
MS0	120	$\overline{\text{P\_DEVSEL}}$	100	S_AD26	29	V_{CC}	40
MS1/BPCC	159	$\overline{\text{P\_FRAME}}$	96	S_AD27	31	V_{CC}	46
NO/HSLED	62	$\overline{\text{P\_GNT}}$	68	S_AD28	32	V_{CC}	56
P_AD0	133	P_IDSEL	83	S_AD29	33	V_{CC}	60
P_AD1	132	$\overline{\text{P\_IRDY}}$	97	S_AD30	35	V_{CC}	75
P_AD2	130	P_MFUNC	102	S_AD31	36	V_{CC}	80
P_AD3	129	P_PAR	106	S_C/$\overline{\text{BE0}}$	145	V_{CC}	90
P_AD4	127	$\overline{\text{P\_PERR}}$	104	S_C/$\overline{\text{BE1}}$	158	V_{CC}	98
P_AD5	126	$\overline{\text{P\_REQ}}$	69	S_C/$\overline{\text{BE2}}$	13	V_{CC}	108
P_AD6	124	$\overline{\text{P\_RST}}$	64	S_C/$\overline{\text{BE3}}$	25	V_{CC}	116
P_AD7	123	$\overline{\text{P\_SERR}}$	105	$\overline{\text{S\_CFN}}$	49	V_{CC}	125
P_AD8	118	$\overline{\text{P\_STOP}}$	101	S_CLK	51	V_{CC}	131
P_AD9	117	$\overline{\text{P\_TRDY}}$	99	S_CLKOUT0	53	V_{CC}	139
P_AD10	115	P_V_{CCP}	67	S_CLKOUT1	55	V_{CC}	147
P_AD11	114	S_AD0	134	S_CLKOUT2	57	V_{CC}	154
P_AD12	113	S_AD1	136	S_CLKOUT3	59	V_{CC}	160

Table 2–5. Primary PCI System

TERMINAL			I/O	DESCRIPTION
NAME	PCM NUMBER	PGF NUMBER		
P_CLK	66	72	I	Primary PCI bus clock. P_CLK provides timing for all transactions on the primary PCI bus. All primary PCI signals are sampled at rising edge of P_CLK.
$\overline{\text{P\_RST}}$	64	70	I	PCI reset. When the primary PCI bus reset is asserted, $\overline{\text{P\_RST}}$ causes the bridge to put all output buffers in a high-impedance state and reset all internal registers. When asserted, the device is completely nonfunctional. During $\overline{\text{P\_RST}}$, the secondary interface is driven low and NO/HSLED is driven high if hot-swap is enabled. After $\overline{\text{P\_RST}}$ is deasserted, the bridge is in its default state.

Table 2–6. Primary PCI Address and Data

TERMINAL			I/O	DESCRIPTION
NAME	PCM NUMBER	PGF NUMBER		
P_AD31	70	76	I/O	Primary address/data bus. These signals make up the multiplexed PCI address and data bus on the primary interface. During the address phase of a primary bus PCI cycle, P_AD31–P_AD0 contain a 32-bit address or other destination information. During the data phase, P_AD31–P_AD0 contain data.
P_AD30	72	78		
P_AD29	73	79		
P_AD28	74	80		
P_AD27	76	82		
P_AD26	77	83		
P_AD25	78	84		
P_AD24	79	86		
P_AD23	84	94		
P_AD22	85	95		
P_AD21	87	97		
P_AD20	88	98		
P_AD19	89	99		
P_AD18	91	101		
P_AD17	92	102		
P_AD16	93	103		
P_AD15	109	119		
P_AD14	110	120		
P_AD13	111	121		
P_AD12	113	123		
P_AD11	114	124		
P_AD10	115	125		
P_AD9	117	127		
P_AD8	118	128		
P_AD7	123	137		
P_AD6	124	138		
P_AD5	126	140		
P_AD4	127	141		
P_AD3	129	143		
P_AD2	130	144		
P_AD1	132	146		
P_AD0	133	147		
P_C/$\overline{\text{BE3}}$	82	91	I/O	Primary bus commands and byte enables. These signals are multiplexed on the same PCI terminals. During the address phase of a primary bus cycle, P_C/$\overline{\text{BE3}}$–P_C/$\overline{\text{BE0}}$ define the bus command. During the data phase, this 4-bit bus is used as byte enables. The byte enables determine which byte paths of the full 32-bit data bus carry meaningful data. P_C/$\overline{\text{BE0}}$ applies to byte 0 (P_AD7–P_AD0), P_C/$\overline{\text{BE1}}$ applies to byte 1 (P_AD15–P_AD8), P_C/$\overline{\text{BE2}}$ applies to byte 2 (P_AD23–P_AD16), and P_C/$\overline{\text{BE3}}$ applies to byte 3 (P_AD31–P_AD24).
P_C/$\overline{\text{BE2}}$	95	105		
P_C/$\overline{\text{BE1}}$	107	117		
P_C/$\overline{\text{BE0}}$	122	135		

Table 2–7. Primary PCI Interface Control

TERMINAL			I/O	DESCRIPTION
NAME	PCM NUMBER	PGF NUMBER		
P_DEVSEL	100	110	I/O	Primary device select. The bridge asserts P_DEVSEL to claim a PCI cycle as the target device. As a PCI initiator on the primary bus, the bridge monitors P_DEVSEL until a target responds. If no target responds before a time-out occurs, then the bridge terminates the cycle with a master abort.
P_FRAME	96	106	I/O	Primary cycle frame. P_FRAME is driven by the initiator of a primary bus cycle. P_FRAME is asserted to indicate that a bus transaction is beginning, and data transactions continue while this signal is asserted. When P_FRAME is deasserted, the primary bus transaction is in the final data phase.
P_GNT	68	74	I	Primary bus grant to bridge. P_GNT is driven by the primary PCI bus arbiter to grant the bridge access to the primary PCI bus after the current data transaction has completed. P_GNT may or may not follow a primary bus request, depending on the primary bus parking algorithm.
P_IDSEL	83	93	I	Primary initialization device select. P_IDSEL selects the bridge during configuration space accesses. P_IDSEL can be connected to one of the upper 16 PCI address lines on the primary PCI bus. Note: There is no IDSEL signal interfacing the secondary PCI bus; thus, the entire configuration space of the bridge can only be accessed from the primary bus.
P_IRDY	97	107	I/O	Primary initiator ready. P_IRDY indicates the ability of the primary bus initiator to complete the current data phase of the transaction. A data phase is completed on a rising edge of P_CLK where both P_IRDY and P_TRDY are asserted. Until P_IRDY and P_TRDY are both sampled asserted, wait states are inserted.
P_PAR	106	116	I/O	Primary parity. In all primary bus read and write cycles, the bridge calculates even parity across the P_AD and P_C/BE buses. As an initiator during PCI write cycles, the bridge outputs this parity indicator with a one-P_CLK delay. As a target during PCI read cycles, the calculated parity is compared to the initiator parity indicator; a miscompare can result in a parity error assertion (P_PERR).
P_PERR	104	114	I/O	Primary parity error indicator. P_PERR is driven by a primary bus PCI device to indicate that calculated parity does not match P_PAR when P_PERR is enabled through bit 6 of the command register (offset 04h, see Section 4.3).
P_REQ	69	75	O	Primary PCI bus request. P_REQ is asserted by the bridge to request access to the primary PCI bus as an initiator.
P_SERR	105	115	O	Primary system error. Output pulsed from the bridge when enabled through the command register (offset 04h, see Section 4.3) indicating a system error has occurred. The bridge need not be the target of the primary PCI cycle to assert this signal. When bit 1 is enabled in the bridge control register (offset 3Eh, see Section 4.32), this signal will also pulse indicating that a system error has occurred on one of the subordinate buses downstream from the bridge.
P_STOP	101	111	I/O	Primary cycle stop signal. This signal is driven by a PCI target to request the initiator to stop the current primary bus transaction. This signal is used for target disconnects and is commonly asserted by target devices which do not support burst data transfers.
P_TRDY	99	109	I/O	Primary target ready. P_TRDY indicates the ability of the primary bus target to complete the current data phase of the transaction. A data phase is completed on the rising edge of P_CLK where both P_IRDY and P_TRDY are asserted. Until P_IRDY and P_TRDY are both sample asserted, wait states are inserted.

Table 2–8. Secondary PCI System

TERMINAL			I/O	DESCRIPTION
NAME	PCM NUMBER	PGF NUMBER		
S_CLKOUT4 S_CLKOUT3 S_CLKOUT2 S_CLKOUT1 S_CLKOUT0	61 59 57 55 53	67 65 63 61 59	O	Secondary PCI bus clocks. Provide timing for all transactions on the secondary PCI bus. Each secondary bus device samples all secondary PCI signals at the rising edge of its corresponding S_CLKOUT input.
S_CLK	51	57	I	Secondary PCI bus clock input. This input syncronizes the PCI2250 to the secondary bus clocks.
$\overline{\text{S\_CFN}}$	49	55	I	Secondary external arbiter enable. When this signal is high, the secondary external arbiter is enabled. When the external arbiter is enabled, the $\overline{\text{S\_REQ0}}$ pin is reconfigured as a secondary bus grant input to the bridge and $\overline{\text{S\_GNT0}}$ is reconfigured as a secondary bus master request to the external arbiter on the secondary bus.
$\overline{\text{S\_RST}}$	48	54	O	Secondary PCI reset. $\overline{\text{S\_RST}}$ is a logical OR of $\overline{\text{P\_RST}}$ and the state of the secondary bus reset bit of the bridge control register (offset 3Eh, see Section 4.32). $\overline{\text{S\_RST}}$ is asynchronous with respect to the state of the secondary interface CLK signal.

Table 2–9. Secondary PCI Address and Data

TERMINAL			I/O	DESCRIPTION
NAME	PCM NUMBER	PGF NUMBER		
S_AD31	36	38	I/O	Secondary address/data bus. These signals make up the multiplexed PCI address and data bus on the secondary interface. During the address phase of a secondary bus PCI cycle, S_AD31–S_AD0 contain a 32-bit address or other destination information. During the data phase, S_AD31–S_AD0 contain data.
S_AD30	35	37		
S_AD29	33	35		
S_AD28	32	34		
S_AD27	31	33		
S_AD26	29	31		
S_AD25	28	30		
S_AD24	26	28		
S_AD23	24	26		
S_AD22	22	24		
S_AD21	21	23		
S_AD20	20	22		
S_AD19	18	20		
S_AD18	17	19		
S_AD17	16	18		
S_AD16	14	16		
S_AD15	156	170		
S_AD14	155	169		
S_AD13	153	167		
S_AD12	152	166		
S_AD11	150	164		
S_AD10	149	163		
S_AD9	148	162		
S_AD8	146	160		
S_AD7	144	158		
S_AD6	142	156		
S_AD5	141	155		
S_AD4	140	154		
S_AD3	138	152		
S_AD2	137	151		
S_AD1	136	150		
S_AD0	134	148		
S_C/$\overline{BE3}$	25	27	I/O	Secondary bus commands and byte enables. These signals are multiplexed on the same PCI terminals. During the address phase of a secondary bus cycle, S_C/$\overline{BE3}$–S_C/$\overline{BE0}$ define the bus command. During the data phase, this 4-bit bus is used as byte enables. The byte enables determine which byte paths of the full 32-bit data bus carry meaningful data. S_C/$\overline{BE0}$ applies to byte 0 (S_AD7–S_AD0), S_C/$\overline{BE1}$ applies to byte 1 (S_AD15–S_AD8), S_C/$\overline{BE2}$ applies to byte 2 (S_AD23–S_AD16), and S_C/$\overline{BE3}$ applies to byte 3 (S_AD31–S_AD24).
S_C/$\overline{BE2}$	13	15		
S_C/$\overline{BE1}$	158	172		
S_C/$\overline{BE0}$	145	159		

Table 2–10. Secondary PCI Interface Control

TERMINAL			I/O	DESCRIPTION
NAME	PCM NUMBER	PGF NUMBER		
S_DEVSEL	7	9	I/O	Secondary device select. The bridge asserts S_DEVSEL to claim a PCI cycle as the target device. As a PCI initiator on the secondary bus, the bridge monitors S_DEVSEL until a target responds. If no target responds before a timeout occurs, then the bridge terminates the cycle with a master abort.
S_FRAME	11	13	I/O	Secondary cycle frame. S_FRAME is driven by the initiator of a secondary bus cycle. S_FRAME is asserted to indicate that a bus transaction is beginning and data transfers continue while S_FRAME is asserted. When S_FRAME is deasserted, the secondary bus transaction is in the final data phase.
S_GNT3 S_GNT2 S_GNT1 S_GNT0	47 45 44 43	53 51 50 49	O	Secondary bus grant to the bridge. The bridge provides internal arbitration and these signals are used to grant potential secondary PCI masters access to the bus. Five potential initiators (including the bridge) can be located on the secondary PCI bus. When the internal arbiter is disabled, S_GNT0 is reconfigured as an external secondary bus request signal for the bridge.
S_IRDY	10	12	I/O	Secondary initiator ready. S_IRDY indicates the ability of the secondary bus initiator to complete the current data phase of the transaction. A data phase is completed on a rising edge of S_CLK where both S_IRDY and S_TRDY are asserted. Until S_IRDY and S_TRDY are both sample asserted, wait states are inserted.
S_PAR	2	3	I/O	Secondary parity. In all secondary bus read and write cycles, the bridge calculates even parity across the S_AD and S_C/BE buses. As an initiator during PCI write cycles, the bridge outputs this parity indicator with a one-S_CLK delay. As a target during PCI read cycles, the calculated parity is compared to the initiator parity indicator. A miscompare can result in a parity error assertion (S_PERR).
S_PERR	4	6	I/O	Secondary parity error indicator. S_PERR is driven by a secondary bus PCI device to indicate that calculated parity does not match S_PAR when S_PERR is enabled through bit 6 of the command register (offset 04h, see Section 4.3).
S_REQ3 S_REQ2 S_REQ1 S_REQ0	42 39 38 37	47 42 40 39	I	Secondary PCI bus request signals. The bridge provides internal arbitration, and these signals are used as inputs from secondary PCI bus initiators requesting the bus. Five potential initiators (including the bridge) can be located on the secondary PCI bus. When the internal arbiter is disabled, the S_REQ0 signal is reconfigures as an external secondary bus grant for the bridge.
S_SERR	3	5	I	Secondary system error. S_SERR is passed through the primary interface by the bridge if enabled through the bridge control register (offset 3Eh, see Section 4.32). S_SERR is never asserted by the bridge.
S_STOP	6	8	I/O	Secondary cycle stop signal. S_STOP is driven by a PCI target to request the initiator to stop the current secondary bus transaction. S_STOP is used for target disconnects and is commonly asserted by target devices that do not support burst data transfers.
S_TRDY	9	11	I/O	Secondary target ready. S_TRDY indicates the ability of the secondary bus target to complete the current data phase of the transaction. A data phase is completed on a rising edge of S_CLK where both S_IRDY and S_TRDY are asserted. Until S_IRDY and S_TRDY are both sample asserted, wait states are inserted.

Table 2–11. Miscellaneous Terminals

TERMINAL			I/O	DESCRIPTION
NAME	PCM NUMBER	PGF NUMBER		
$\overline{GOZ}$	63	69	I	NAND tree enable pin.
NO/HSLED	62	68	I/O	NAND tree out when $\overline{GOZ}$ is asserted. Hot-swap LED when $\overline{GOZ}$ is deasserted.
MS0	120	132	I	Mode select 0
MS1/BPCC	159	174	I	Mode select 1 when mode select 0 is low, bus power clock control when mode select 0 is high.
P_MFUNC	102	112	I/O	Primary multifunction terminal. This terminal can be configured as P_CLKRUN, $\overline{P\_LOCK}$, or $\overline{HS\_ENUM}$ depending on the values of MS0 and MS1.
S_MFUNC	5	7	I/O	Secondary multifunction terminal. This terminal can be configured as S_CLKRUN, $\overline{S\_LOCK}$, or $\overline{HS\_SWITCH}$ depending on the values of MS0 and MS1.

Table 2–12. Power Supply

TERMINAL			DESCRIPTION
NAME	PCM NUMBER	PGF NUMBER	
GND	1, 12, 19, 27, 34, 41, 50, 54, 58, 65, 71, 81, 86, 94, 103, 112, 119, 121, 128, 135, 143, 151, 157	1, 14, 21, 29, 36, 45, 56, 60, 64, 71, 77, 89, 96, 104, 113, 122, 130, 133, 142, 149, 157, 165, 171	Device ground terminals
V_{CC}	8, 15, 23, 30, 40, 46, 56, 60, 75, 80, 90, 98, 108, 116, 125, 131, 139, 147, 154, 160	10, 17, 25, 32, 44, 52, 62, 66, 81, 88, 100, 108, 118, 126, 139, 145, 153, 161, 168, 176	Power-supply terminal for core logic (3.3 V)
P_V_{CCP}	67	73	Primary bus-signaling environment supply. P_V_{CCP} is used in protection circuitry on primary bus I/O signals.
S_V_{CCP}	52	58	Secondary bus-signaling environment supply. S_V_{CCP} is used in protection circuitry on secondary bus I/O signals.

3 Feature/Protocol Descriptions

The following sections give an overview of the PCI2250 PCI-to-PCI bridge features and functionality. Figure 3–1 shows a simplified block diagram of a typical system implementation using the PCI2250.

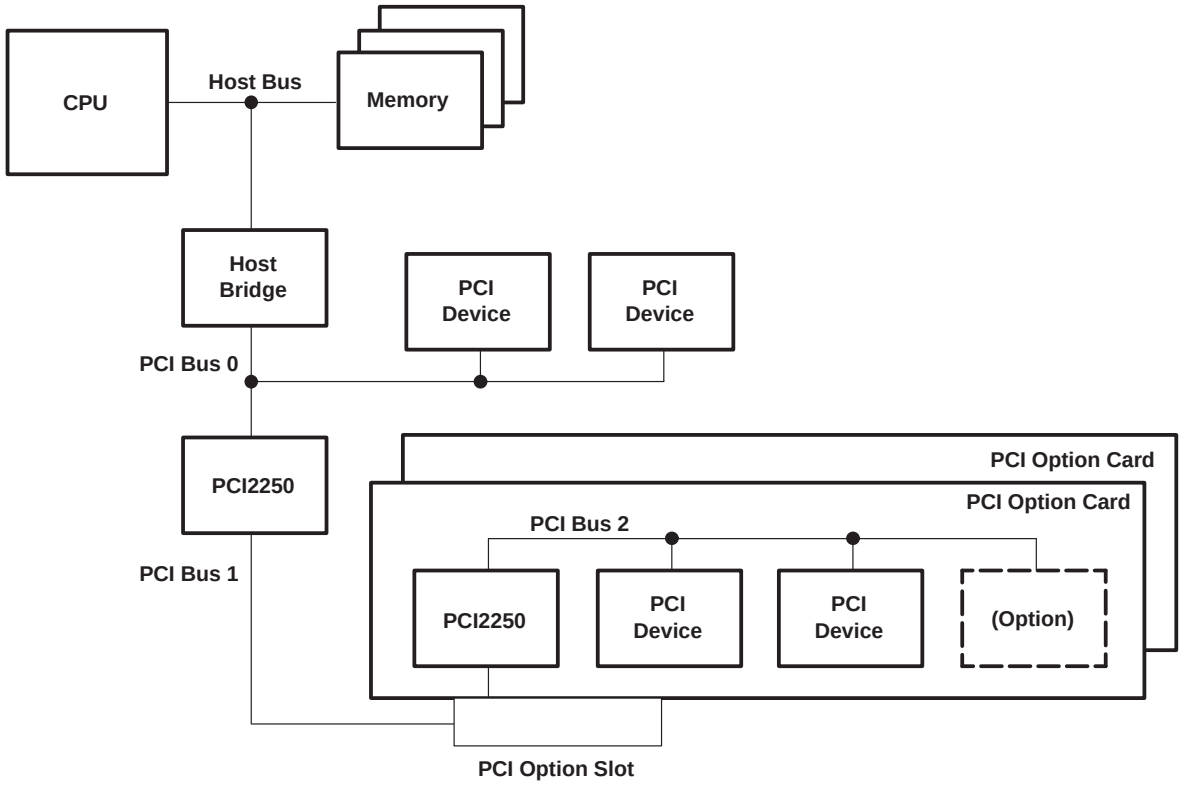


Figure 3–1. System Block Diagram

3.1 Introduction to the PCI2250

The PCI2250 is a bridge between two PCI buses and is compliant with both the *PCI Local Bus Specification* and the *PCI-to-PCI Bridge Specification*. The bridge supports two 32-bit PCI buses operating at a maximum of 33 MHz. The primary and secondary buses operate independently in either a 3.3-V or 5-V signaling environment. The core logic of the bridge, however, is powered at 3.3 V to reduce power consumption.

Host software interacts with the bridge through internal registers. These internal registers provide the standard PCI status and control for both the primary and secondary buses. Many vendor-specific features that exist in the TI extension register set are included in the bridge. The PCI configuration header of the bridge is only accessible from the primary PCI interface.

The bridge provides internal arbitration for the four possible secondary bus masters, and provides each with a dedicated active low request/grant pair ($\overline{\text{REQ}}$/$\overline{\text{GNT}}$). The arbiter features a two-tier rotational scheme with the PCI2250 bridge defaulting to the highest priority tier. The bus parking scheme is also configurable and can be set to either park grant ($\overline{\text{GNT}}$) on the bridge or on the last mastering device.

Upon system power up, power-on self-test (POST) software configures the bridge according to the devices that exist on subordinate buses, and enables the performance-enhancing features of the PCI2250. In a typical system, this is the only communication with the bridge internal register set.

3.2 PCI Commands

The bridge responds to PCI bus cycles as a PCI target device based on the decoding of each address phase and internal register settings. Table 3–1 lists the valid PCI bus cycles and their encoding on the command/byte enables (C/$\overline{BE}$) bus during the address phase of a bus cycle.

Table 3–1. PCI Command Definition

C/$\overline{BE3}$–C/$\overline{BE0}$	COMMAND
0000	Interrupt acknowledge
0001	Special cycle
0010	I/O read
0011	I/O write
0100	Reserved
0101	Reserved
0110	Memory read
0111	Memory write
1000	Reserved
1001	Reserved
1010	Configuration read
1011	Configuration write
1100	Memory read multiple
1101	Dual address cycle
1110	Memory read line
1111	Memory write and invalidate

The bridge never responds as a PCI target to the interrupt acknowledge, special cycle, dual address cycle, or reserved commands. The bridge does, however, initiate special cycles on both interfaces when a type 1 configuration cycle issues the special cycle request. The remaining PCI commands address either memory, I/O, or configuration space. The bridge accepts PCI cycles by asserting $\overline{DEVSEL}$ as a medium-speed device, i.e., $\overline{DEVSEL}$ is asserted two clock cycles after the address phase.

The PCI2250 converts memory write and invalidate commands to memory write commands when forwarding transactions from either the primary or secondary side of the bridge.

3.3 Configuration Cycles

The *PCI Local Bus Specification* defines two types of PCI configuration read and write cycles: type 0 and type 1. The bridge decodes each type differently. Type 0 configuration cycles are intended for devices on the primary bus, while type 1 configuration cycles are intended for devices on some hierarchically subordinate bus. The difference between these two types of cycles is the encoding of the primary PCI (P_AD) bus during the address phase of the cycle. Figure 3–2 shows the P_AD bus encoding during the address phase of a type 0 configuration cycle. The 6-bit register number field represents an 8-bit address with the two lower bits masked to 0, indicating a doubleword boundary. This results in a 256-byte configuration address space per function per device. Individual byte accesses may be selected within a doubleword by using the P_C/$\overline{BE}$ signals during the data phase of the cycle.

31 11	10 8	7 2	1	0
Reserved	Function Number	Register Number	0	0

Figure 3–2. PCI AD31–AD0 During Address Phase of a Type 0 Configuration Cycle

The bridge claims only type 0 configuration cycles when its P_IDSEL terminal is asserted during the address phase of the cycle and the PCI function number encoded in the cycle is 0. If the function number is 1 or greater, the bridge does not recognize the configuration command. In this case, the bridge does not assert $\overline{\text{DEVSEL}}$ and the configuration transaction results in a master abort. The bridge services valid type 0 configuration read or write cycles by accessing internal registers from the configuration header.

Because type 1 configuration cycles are issued to devices on subordinate buses, the bridge claims type 1 cycles based on the bus number of the destination bus. Figure 3–3 shows the P_AD bus encoding during the address phase of a type 1 cycle. The device number and bus number fields define the destination bus and device for the cycle.

31 24	23 16	15 11	10 8	7 2	1	0
Reserved	Bus Number	Device Number	Function Number	Register Number	0	0

Figure 3–3. PCI AD31–AD0 During Address Phase of a Type 1 Configuration Cycle

Several bridge configuration registers shown in Table 4–1 are significant when decoding and claiming type 1 configuration cycles. The destination bus number encoded on the P_AD bus is compared to the values programmed in the bridge configuration registers 18h, 19h, and 1Ah, which are the primary bus number, secondary bus number, and subordinate bus number registers, respectively. These registers default to 00h and are programmed by host software to reflect the bus hierarchy in the system (see Figure 3–4 for an example of a system bus hierarchy and how the PCI2250 bus number registers would be programmed in this case).

When the PCI2250 claims a type 1 configuration cycle that has a bus number equal to its secondary bus number, the PCI2250 converts the type 1 configuration cycle to a type 0 configuration cycle and asserts the proper S_AD line as the IDSEL (see Table 3–2). All other type 1 transactions that access a bus number greater than the bridge secondary bus number but less than or equal to its subordinate bus number are forwarded as type 1 configuration cycles.

Table 3–2. PCI S_AD31–S_AD16 During Address Phase of a Type 0 Configuration Cycle

DEVICE NUMBER	SECONDARY IDSEL S_AD31–S_AD16	S_AD ASSERTED
0h	0000 0000 0000 0001	16
1h	0000 0000 0000 0010	17
2h	0000 0000 0000 0100	18
3h	0000 0000 0000 1000	19
4h	0000 0000 0001 0000	20
5h	0000 0000 0010 0000	21
6h	0000 0000 0100 0000	22
7h	0000 0000 1000 0000	23
8h	0000 0001 0000 0000	24
9h	0000 0010 0000 0000	25
Ah	0000 0100 0000 0000	26
Bh	0000 1000 0000 0000	27
Ch	0001 0000 0000 0000	28
Dh	0010 0000 0000 0000	29
Eh	0100 0000 0000 0000	30
Fh	1000 0000 0000 0000	31
10h–1Eh	0000 0000 0000 0000	–

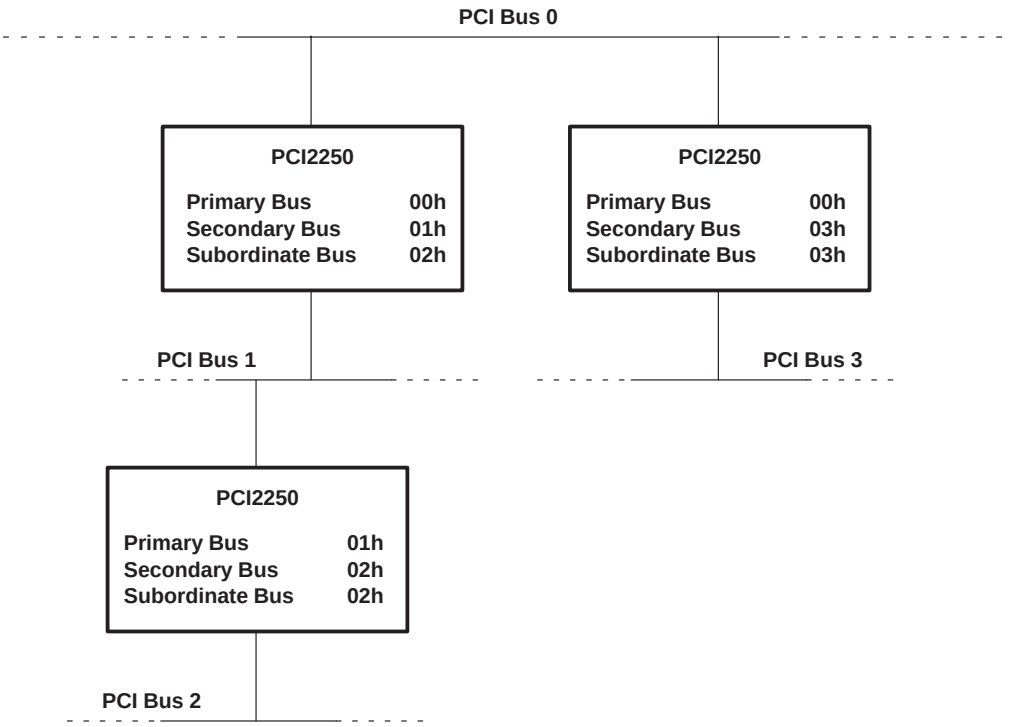


Figure 3–4. Bus Hierarchy and Numbering

3.4 Special Cycle Generation

The bridge is designed to generate special cycles on both buses through a type 1 cycle conversion. During a type 1 configuration cycle, if the bus number field matches the bridge secondary bus number, then the device number field is 1Fh, the function number field is 07h, and the bridge generates a special cycle on the secondary bus with a message that matches the type 1 configuration cycle data. If the bus number is a subordinate bus and not the secondary bus, then the bridge passes the type 1 special cycle request through to the secondary interface along with the proper message.

Special cycles are never passed through the bridge. Type 1 configuration cycles with a special cycle request can propagate in both directions.

3.5 Secondary Clocks

The PCI2250 provides five secondary clock outputs (S_CLKOUT[0:4]). Four are provided for clocking secondary devices. The fifth clock should be routed back into the PCI2250 S_CLK input to ensure all secondary bus devices see the same clock.

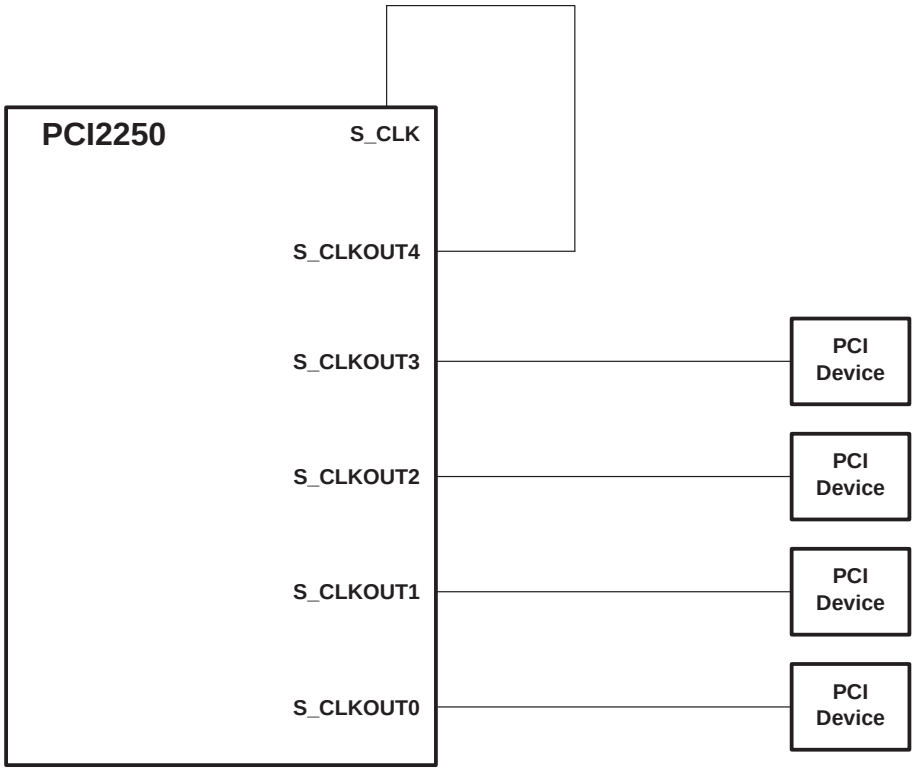


Figure 3–5. Secondary Clock Block Diagram

3.6 Bus Arbitration

The PCI2250 implements bus request ($\overline{\text{P\_REQ}}$) and bus grant ($\overline{\text{P\_GNT}}$) terminals for primary bus arbitration. Four secondary bus requests and four secondary bus grants are provided on the secondary of the PCI2250. Five potential initiators, including the bridge, can be located on the secondary bus. The PCI2250 provides a two-tier arbitration scheme on the secondary bus for priority bus-master handling.

The two-tier arbitration scheme improves performance in systems in which master devices do not all require the same bandwidth. Any master that requires frequent use of the bus can be programmed to be in the higher priority tier.

3.6.1 Primary Bus Arbitration

The PCI2250, acting as an initiator on the primary bus, asserts $\overline{\text{P\_REQ}}$ when forwarding transactions upstream to the primary bus. In the upstream direction, as long as a posted write data or a delayed transaction request is in the queue, the PCI2250 keeps $\overline{\text{P\_REQ}}$ asserted. If a target disconnect, a target retry, or a target abort is received in response to a transaction initiated on the primary bus by the PCI2250, $\overline{\text{P\_REQ}}$ is deasserted for two PCI clock cycles.

When the primary bus arbiter asserts $\overline{\text{P\_GNT}}$ in response to a $\overline{\text{P\_REQ}}$ from the PCI2250, the device initiates a transaction on the primary bus during the next PCI clock cycle after the primary bus is sampled idle.

When $\overline{\text{P\_REQ}}$ is not asserted and the primary bus arbiter asserts $\overline{\text{P\_GNT}}$ to the PCI2250, the device responds by parking the P_AD31–P_AD0 bus, the C/$\overline{\text{BE}}$3–C/$\overline{\text{BE}}$0 bus, and primary parity (P_PAR) by driving them to valid logic levels. If the PCI2250 is parking the primary bus and wants to initiate a transaction on the bus, then it can start the transaction on the next PCI clock by asserting the primary cycle frame ($\overline{\text{P_FRAME}}$) while $\overline{\text{P\_GNT}}$ is still asserted. If $\overline{\text{P\_GNT}}$ is deasserted, then the bridge must rearbitrate for the bus to initiate a transaction.

3.6.2 Internal Secondary Bus Arbitration

$\overline{\text{S\_CFN}}$ controls the state of the secondary internal arbiter. The internal arbiter can be enabled by pulling $\overline{\text{S\_CFN}}$ low or disabled by pulling $\overline{\text{S\_CFN}}$ high. The PCI2250 provides four secondary bus request terminals and four secondary

bus grant terminals. Including the bridge, there are a total of five potential secondary bus masters. These request and grant signals are connected to the internal arbiter. When an external arbiter is implemented, S_REQ3–S_REQ0 and S_GNT3–S_GNT0 are placed in a high impedance mode.

3.6.3 External Secondary Bus Arbitration

An external secondary bus arbiter can be used instead of the PCI2250 internal arbiter. When using an external arbiter, the PCI2250's internal arbiter should be disabled by pulling S_CFN high.

When an external secondary bus arbiter is used, the PCI2250 internally reconfigures the S_REQ0 and S_GNT0 signals so that S_REQ0 becomes the secondary bus grant for the bridge and S_GNT0 becomes the secondary bus request for the bridge. This is done because S_REQ0 is an input and can thus be used to provide the grant input to the bridge, and S_GNT0 is an output and can thus provide the request output from the bridge.

When an external arbiter is used, all unused secondary bus grant outputs (S_GNT3–S_GNT1) are placed in a high impedance mode. Any unused secondary bus request inputs (S_REQ3–S_REQ1) should be pulled high to prevent the inputs from oscillating.

3.7 Decode Options

The PCI2250 supports positive, subtractive, and negative decoding but defaults to positive decoding on the primary interface and negative decoding on the secondary bus. Positive decoding is a method of address decoding in which a device responds only to accesses within an assigned address range. Negative decoding is a method of address decoding in which a device responds only to accesses outside an assigned address range. Subtractive decoding is a method of address decoding in which a device responds to accesses not claimed by any other devices on the bus. Subtractive decoding can be enabled on the primary bus or the secondary bus.

3.8 Extension Windows With Programmable Decoding

The PCI2250 provides two programmable 32-bit extension windows. Each window can be programmed to be a prefetchable memory window, a nonprefetchable memory window, or an I/O window. The TI extension memory windows have a 4K-byte granularity, and the I/O windows have a doubleword granularity. These extension windows can be positively decoded on either the primary bus or secondary bus.

The standard PCI-to-PCI bridge memory and I/O windows specified by the *PCI-to-PCI Bridge Specification* have a 1M-byte and 4K-byte granularity, respectively (see Section 4.20, *Memory Base Register* and Section 4.26, *I/O Base Upper 16 Bits Register*). The TI extension windows provide smaller granularity for memory and I/O windows. The extension windows' granularity matches the requirements of CardBus card windows, which also have 4K-byte granularity for memory windows and doubleword granularity for I/O windows. When a CardBus I/O card is sitting behind the bridge, the smaller doubleword I/O window granularity with the extension windows allows a smaller I/O window than the 4K-byte window with the standard I/O base and limit registers.

A common I/O base address for popular sound cards is 300h–303h. Using the TI extension windows and configuring the base I/O address for 300h establishes a 4-byte I/O address window from 300h–303h for communicating with the sound card. Using the bridge's standard I/O base register requires a minimum 4K-byte window of memory.

The extension windows can be excluded from the primary bus decoding, thus creating a hole in a primary window address range.

3.9 System Error Handling

The PCI2250 can be configured to signal a system error (SERR) under a variety of conditions. The P_SERR event disable register (offset 64h, see Section 5.18) and the P_SERR status register (offset 6Ah, see Section 5.20) provide control and status bits for each condition for which the bridge can signal SERR. These individual bits enable SERR reporting for both downstream and upstream transactions.

By default, the PCI2250 will not signal SERR. If the PCI2250 is configured to signal SERR by setting bit 8 of the command register (offset 04h, see Section 4.3), then the bridge signals SERR if any of the error conditions in the

P_SERR event disable register occur and that condition is enabled. By default, all error conditions are enabled in the P_SERR event disable register. When the bridge signals $\overline{\text{SERR}}$, bit 14 of the secondary status register (offset 1Eh, see Section 4.19) is set.

3.9.1 Posted Write Parity Error

If bit 1 in the P_SERR event disable register (offset 64h, see Section 5.18) is 0, then parity errors on the target bus during a posted write are passed to the initiating bus as an $\overline{\text{SERR}}$. When this occurs, bit 1 of the P_SERR status register (offset 6Ah, see Section 5.20) is set. The status bit is cleared by writing a 1.

3.9.2 Posted Write Timeout

If bit 2 in the P_SERR event disable register (offset 64h, see Section 5.18) is 0 and the retry timer expires while attempting to complete a posted write, then the PCI2250 signals $\overline{\text{SERR}}$ on the initiating bus. When this occurs, bit 2 of the P_SERR status register (offset 6Ah, see Section 5.20) is set. The status bit is cleared by writing a 1.

3.9.3 Target Abort on Posted Writes

If bit 3 in the P_SERR event disable register (offset 64h, see Section 5.18) is 0 and the bridge gets a target abort during a posted write transaction, then the PCI2250 signals $\overline{\text{SERR}}$ on the initiating bus. When this occurs, bit 3 of the P_SERR status register (offset 6Ah, see Section 5.20) is set. The status bit is cleared by writing a 1.

3.9.4 Master Abort on Posted Writes

If bit 4 in the P_SERR event disable register (offset 64h, see Section 5.18) is 0 and a posted write transaction results in a master abort, then the PCI2250 signals $\overline{\text{SERR}}$ on the initiating bus. When this occurs, bit 4 of the P_SERR status register (offset 6Ah, see Section 5.20) is set. The status bit is cleared by writing a 1.

3.9.5 Master Delayed Write Timeout

If bit 5 in the P_SERR event disable register (offset 64h, see Section 5.18) is 0 and the retry timer expires while attempting to complete a delayed write, then the PCI2250 signals $\overline{\text{SERR}}$ on the initiating bus. When this occurs, bit 5 of the P_SERR status register (offset 6Ah, see Section 5.20) is set. The status bit is cleared by writing a 1.

3.9.6 Master Delayed Read Timeout

If bit 6 in the P_SERR event disable register (offset 64h, see Section 5.18) is 0 and the retry timer expires while attempting to complete a delayed read, then the PCI2250 signals $\overline{\text{SERR}}$ on the initiating bus. When this occurs, bit 6 of the P_SERR status register (offset 6Ah, see Section 5.20) is set. The status bit is cleared by writing a 1.

3.9.7 Secondary $\overline{\text{SERR}}$

The PCI2250 passes $\overline{\text{SERR}}$ from the secondary bus to the primary bus if it is enabled for $\overline{\text{SERR}}$ response (bit 8 in the command register is 1) and bit 1 in the bridge control register (offset 3Eh, see Section 4.32) is set.

3.10 Parity Handling and Parity Error Reporting

The PCI2250 can be configured to pass parity or provide parity via bit 14 of the diagnostic control register (offset 5Ch, see Section 5.14). When this bit is cleared to 0, the bridge is enabled for passing parity errors. Parity error passing is the default mode in the bridge. The following parity conditions result in the bridge signaling an error.

3.10.1 Address Parity Error

If the parity error response bit (bit 6) in the command register (offset 04h, see Section 4.3) is set, then the PCI2250 signals $\overline{\text{SERR}}$ on address parity errors and target abort transactions.

3.10.2 Data Parity Error

If the parity error response bit (bit 6) in the command register (offset 04h, see Section 4.3) is set, then the PCI2250 signals $\overline{\text{PERR}}$ when it receives bad data. When the bridge detects bad parity, bit 15 (detected parity error) in the status register (offset 06h, see Section 4.4) is set.

If the bridge is configured to respond to parity errors via bit 6 in the command register, then the data parity error detected bit (bit 8 in the status register) is set when the bridge detects bad parity. The data parity error detected bit is also set when the bridge, as a bus master, asserts $\overline{\text{PERR}}$ or detects $\overline{\text{PERR}}$.

3.11 Master and Target Abort Handling

If the PCI2250 receives a target abort during a write burst, then it signals target abort back on the initiator bus. If it receives a target abort during a read burst, then it provides all of the valid data on the initiator bus and disconnects. Target aborts for posted and nonposted transactions are reported as specified in the *PCI-to-PCI Bridge Specification*.

Master aborts for posted and nonposted transactions are reported as specified in the *PCI-to-PCI Bridge Specification*. If a transaction is attempted on the primary bus after a secondary reset is asserted, then the PCI2250 follows bit 5 (master abort mode bit setting) in the bridge control register (offset 3Eh, see Section 4.32) for reporting errors.

3.12 Discard Timer

The PCI2250 is free to discard the data or status of a delayed transaction that was completed with a delayed transaction termination when a bus master has not repeated the request within 2^{10} or 2^{15} PCI clocks (approximately 30 μs and 993 μs, respectively). The *PCI Local Bus Specification* recommends that a bridge wait 2^{15} PCI clocks before discarding the transaction data or status.

The PCI2250 implements a discard timer for use in delayed transactions. After a delayed transaction is completed on the destination bus, the bridge may discard it under two conditions. The first condition occurs when a read transaction is made to a region of memory that that is inside a defined prefetchable memory region, or when the command is a memory read line or a memory read multiple, implying that the memory region is prefetchable. The other condition occurs when the master originating the transaction (either a read or a write, prefetchable or nonprefetchable) has not retried the transaction within 2^{10} or 2^{15} clocks. The number of clocks is tracked by a timer referred to as the discard timer. When the discard timer expires, the bridge is required to discard the data. The PCI2250 default value for the discard timer is 2^{15} clocks; however, this value can be set to 2^{10} clocks by setting bit 9 in the bridge control register (offset 3Eh, see Section 4.32). For more information on the discard timer, see *error conditions* in *PCI Local Bus Specification*.

3.13 Delayed Transactions

The bridge supports delayed transactions as defined in the *PCI Local Bus Specification*. A target must be able to complete the initial data phase in 16 PCI clocks or less from the assertion of the cycle frame ($\overline{\text{FRAME}}$), and subsequent data phases must complete in 8 PCI clocks or less. A delayed transaction consists of three phases:

- An initiator device issues a request.
- The target completes the request on the destination bus and signals the completion to the initiator.
- The initiator completes the request on the originating bus.

If the bridge is the target of a PCI transaction and it must access a slow device to write or read the requested data, and the transaction takes longer than 16 clocks, then the bridge must latch the address, the command, and the byte enables, and then issue a retry to the initiator. The initiator must end the transaction without any transfer of data and is required to retry the transaction later using the same address, command, and byte enables. This is the first phase of the delayed transaction.

During the second phase, if the transaction is a read cycle, then the bridge fetches the requested data on the destination bus, stores it internally, and obtains the completion status, thus completing the transaction on the

destination bus. If it is a write transaction, then the bridge writes the data and obtains the completion status, thus completing the transaction on the destination bus. The bridge stores the completion status until the master on the initiating bus retries the initial request.

During the third phase, the initiator rearbitrates for the bus. When the bridge sees the initiator retry the transaction, it compares the second request to the first request. If the address, command, and byte enables match the values latched in the first request, then the completion status (and data if the request was a read) is transferred to the initiator. At this point, the delayed transaction is complete. If the second request from the initiator does not match the first request exactly, then the bridge issues another retry to the initiator.

When bit 2 of the diagnostic control register (offset 5Ch, see Section 5.14) is 0, the PCI2250 is configured for immediate retry mode. In immediate retry mode, the bridge issues a retry immediately, instead of after 16 clocks, on delayed transactions.

The PCI2250 supports one delayed transaction in each direction at any given time.

3.14 Multifunction Pins

The PCI2250 has two multifunction pins that can be configured as $\overline{\text{LOCK}}$, $\overline{\text{CLKRUN}}$ or compact-PCI hot-swap $\overline{\text{ENUM}}$ and $\overline{\text{SWITCH}}$. The configuration of P_MFUNC and S_MFUNC is controlled by MS0 and MS1 and is shown in Table 3–3. The PCI2250 has two modes of operation: Intel-compatible mode and TI mode. In the Intel mode, the PCI2250 is pin compatible with the Intel 21152 bridge.

Table 3–3. Multifunction Pin Definitions Based on Mode Select Pins

MS0	MS1	P_MFUNC	S_MFUNC	MODE
0	0	$\overline{\text{HS\_ENUM}}$	$\overline{\text{HS\_SWITCH}}$	TI hot-swap
0	1	$\overline{\text{P\_CLKRUN}}$	$\overline{\text{S\_CLKRUN}}$	TI clock run
1	BPCC	$\overline{\text{P\_LOCK}}$	$\overline{\text{S\_LOCK}}$	Intel

3.14.1 Compact-PCI Hot-Swap Support

The PCI2250 is hot-swap friendly silicon that supports all the CPCI hot-swap capable features, contains support for software control, and integrates circuitry required by the *CPCI Hot-Swap Specification*. To be hot-swap capable, the PCI2250 supports the following:

- Compliance with *PCI Local Bus Specification*
- Tolerance of V_{CC} from early power
- Asynchronous reset
- Tolerance of precharge voltage
- I/O buffers must meet modified V/I requirements
- Limited I/O pin voltage at precharge voltage
- Hot-swap control and status programming via extended PCI capabilities linked list
- Hot-swap terminals: $\overline{\text{HS\_ENUM}}$, $\overline{\text{HS\_SWITCH}}$, and HS_LED.

CPCI hot-swap defines a process for installing and removing PCI boards without adversely affecting a running system. The PCI2250 provides this functionality such that it can be implemented on a board that can be removed and inserted in a hot-swap system.

The PCI2250 provides three terminals to support hot-swap when configured to be in hot-swap mode: $\overline{\text{HS\_ENUM}}$ (output), $\overline{\text{HS\_SWITCH}}$ (input), and HS_LED (output). The $\overline{\text{HS\_ENUM}}$ output indicates to the system that an insertion event occurred or that a removal event is about to occur. The $\overline{\text{HS\_SWITCH}}$ input indicates the state of a board ejector handle, and the HS_LED output lights a blue LED to signal insertion and removal ready status.

3.14.2 PCI Clock Run Feature

The PCI2250 supports the PCI clock run protocol when in clock run mode, as defined in the *PCI Mobile Design Guide*. When the system's central resource signals to the system that it wants to stop the PCI clock (P_CLK) by driving the primary clock run ($\overline{\text{P\_CLKRUN}}$) signal high, the bridge either signals that it is OK to stop the PCI clock by leaving $\overline{\text{P\_CLKRUN}}$ deasserted (high) or signals to the system to keep the clock running by driving $\overline{\text{P\_CLKRUN}}$ low.

The PCI2250 clock run control register provides a clock run enable bit for the primary bus and a separate clock run enable bit for the secondary bus. The bridge's $\overline{\text{P\_CLKRUN}}$ and secondary clock run ($\overline{\text{S\_CLKRUN}}$) features are enabled by setting bits 3 and 1, respectively, in the clock run control register (offset 5Bh, see Section 5.13). Bit 2 of the clock run control register allows software to enable the bridge's keep clock running mode to prevent the system from stopping the primary PCI clock. There are two conditions for restarting the secondary clock: a downstream transaction restarts the secondary clock or $\overline{\text{S\_CLKRUN}}$ is asserted.

Two clock run modes are supported on the secondary bus. The bridge can be configured to stop the secondary PCI clock only in response to a request from the primary bus to stop the clock, or it can be configured to stop the secondary clock whenever the secondary bus is idle and there are no transaction requests from the primary bus, regardless of the primary clock (see Section 5.13, *Clock Run Control Register*).

3.15 PCI Power Management

The *PCI Power Management Interface Specification* establishes the infrastructure required to let the operating system control the power of PCI functions. This is done by defining a standard PCI interface and operations to manage the power of PCI functions on the bus. The PCI bus and the PCI functions can be assigned one of four software visible power management states, which result in varying levels of power savings.

The four power management states of PCI functions are D0—fully on state, D1 and D2—intermediate states, and D3—off state. Similarly, bus power states are B0–B3. The bus power states B0–B3 are derived from the device power state of the originating device. The power state of the secondary bus is derived from the power state of the PCI2250.

For the operating system to manage the device power states on the PCI bus, the PCI function supports four power management operations:

- Capabilities reporting
- Power status reporting
- Setting the power state
- System wake–up

The operating system identifies the capabilities of the PCI function by traversing the new capabilities list. The presence of the new capabilities list is indicated by a bit in the status register (offset 06h, see Section 4.4) which provides access to the capabilities list.

3.15.1 Behavior in Low Power States

The PCI2250 supports D0, D1, D2, and $D3_{hot}$ power states when in TI mode. The PCI2250 only supports D0 and $D3_{hot}$ power states when in Intel mode. The PCI2250 is fully functional only in the D0 state. In the lower power states, the bridge does not accept any I/O or memory transactions. These transactions are aborted by the master. The bridge accepts type 0 configuration cycles in all power states. The bridge also accepts type 1 configuration cycles but does not pass these cycles to the secondary bus in any of the low power states. Type 1 configuration writes are discarded and reads return all 1s. All error reporting is done in the low power states. When in D2 and $D3_{hot}$ states, the bridge turns off all secondary clocks for further power savings when in TI mode or if BPCC is pulled high in the Intel mode.

When going from $D3_{hot}$ to D0, an internal reset is generated. This reset initializes all PCI configuration registers to their default values. All TI extension registers (40h–FFh) are not reset. The power management registers (offset E0h) are also not reset.

4 Bridge Configuration Header

The PCI2250 bridge is a single-function PCI device. The configuration header is in compliance with the *PCI-to-PCI Bridge Architecture Specification*. Table 4–1 shows the PCI configuration header, which includes the predefined portion of the bridge's configuration space. The PCI configuration offset is shown in the right column under the OFFSET heading.

Table 4–1. Bridge Configuration Header

REGISTER NAME				OFFSET
Device ID		Vendor ID		00h
Status		Command		04h
Class code			Revision ID	08h
BIST	Header type	Primary latency timer	Cache line size	0Ch
Base address register 0				10h
Base address register 1				14h
Secondary bus latency timer	Subordinate bus number	Secondary bus number	Primary bus number	18h
Secondary status		I/O limit	I/O base	1Ch
Memory limit		Memory base		20h
Prefetchable memory limit		Prefetchable memory base		24h
Prefetchable base upper 32 bits				28h
Prefetchable limit upper 32 bits				2Ch
I/O limit upper 16 bits		I/O base upper 16 bits		30h
Reserved			Capability pointer	34h
Expansion ROM base address				38h
Bridge control		Interrupt pin	Interrupt line	3Ch
Arbiter control		Extended diagnostic	Chip control	40h
Extension window base 0				44h
Extension window limit 0				48h
Extension window base 1				4Ch
Extension window limit 1				50h
Primary decode control	Secondary decode control	Extension window map	Extension window enable	54h
Clock run control	Port decode map	Buffer control	Port decode enable	58h
Diagnostic status		Diagnostic control		5Ch
Arbiter timeout status	Arbiter mask control	Reserved		60h
Reserved			P_SERR event disable	64h
Reserved	P_SERR status	Secondary clock control		68h
Reserved				6Ch–D8h
Power management capabilities		PM next item pointer	PM capability ID	DCh
Data	PMCSR bridge support	Power management control/status		E0h
Reserved	Hot-swap control status	HS next item pointer	HS capability ID	E4h
Reserved				E8h–FFh

A bit description table is typically included that indicates bit field names, a detailed field description, and field access tags. Table 4–2 describes the field access tags.

Table 4–2. Bit Field Access Tag Descriptions

ACCESS TAG	NAME	MEANING
R	Read	Field may be read by software.
W	Write	Field may be written by software to any value.
S	Set	Field may be set by a write of 1. Writes of 0 have no effect.
C	Clear	Field may be cleared by a write of one. Writes of 0 have no effect.
U	Update	Field may be autonomously updated by PCI2040.

4.1 Vendor ID Register

This 16-bit value is allocated by the PCI Special Interest Group (SIG) and identifies TI as the manufacturer of this device. The vendor ID assigned to TI is 104Ch.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Vendor ID															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	1	0	0	0	0	0	1	0	0	1	1	0	0

Register: **Vendor ID**
Type: Read-only
Offset: 00h
Default: 104Ch

4.2 Device ID Register

This 16-bit value is allocated by the vendor and identifies the PCI device. The device ID for the PCI2250 is AC23h.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Device ID															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	1	0	1	0	1	1	0	0	0	0	1	0	0	0	1	1

Register: **Device ID**
Type: Read-only
Offset: 02h
Default: AC23h

4.3 Command Register

The command register provides control over the bridge interface to the primary PCI bus. VGA palette snooping is enabled through this register, and all other bits adhere to the definitions in the *PCI Local Bus Specification*. Table 4–3 describes the bit functions in the command register.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Command															
Type	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Command**
Type: Read-only, read/write (see individual bit descriptions)
Offset: 04h
Default: 0000h

Table 4–3. Command Register

BIT	TYPE	FUNCTION
15–10	R	Reserved. Bits 15–10 return 0s when read.
9	R/W	Fast back-to-back enable. The bridge does not generate fast back-to-back transactions on the primary PCI bus. Bit 9 is read/write, but does not affect the bridge when set. This bit defaults to 0.
8	R/W	System error ($\overline{\text{SERR}}$) enable. Bit 8 controls the enable for the $\overline{\text{SERR}}$ driver on the primary interface. 0 = Disable $\overline{\text{SERR}}$ driver on primary interface (default) 1 = Enable the $\overline{\text{SERR}}$ driver on primary interface
7	R	Wait cycle control. Bit 7 controls address/data stepping by the bridge on both interfaces. The bridge does not support address/data stepping and this bit is hardwired to 0.
6	R/W	Parity error response enable. Bit 6 controls the bridge response to parity errors. 0 = Parity error response disabled (default) 1 = Parity error response enabled
5	R/W	VGA palette snoop enable. When set, the bridge passes I/O writes on the primary PCI bus with addresses 3C6h, 3C8h, and 3C9h inclusive of ISA aliases (i.e., only bits AD9–AD0 are included in the decode).
4	R	Memory write and invalidate enable. In a PCI-to-PCI bridge, bit 4 must be read-only and return 0 when read.
3	R	Special cycle enable. A PCI-to-PCI bridge cannot respond as a target to special cycle transactions, so bit 3 is defined as read-only and must return 0 when read.
2	R/W	Bus master enable. Bit 2 controls the ability of the bridge to initiate a cycle on the primary PCI bus. When bit 2 is 0, the bridge does not respond to any memory or I/O transactions on the secondary interface since they cannot be forwarded to the primary PCI bus. 0 = Bus master capability disabled (default) 1 = Bus master capability enabled
1	R/W	Memory space enable. Bit 1 controls the bridge response to memory accesses for both prefetchable and nonprefetchable memory spaces on the primary PCI bus. Only when bit 1 is set will the bridge forward memory accesses to the secondary bus from a primary bus initiator. 0 = Memory space disabled (default) 1 = Memory space enabled
0	R/W	I/O space enable. Bit 0 controls the bridge response to I/O accesses on the primary interface. Only when bit 0 is set will the bridge forward I/O accesses to the secondary bus from a primary bus initiator. 0 = I/O space disabled (default) 1 = I/O space enabled

4.4 Status Register

The status register provides device information to the host system. This register is read-only. Bits in this register are cleared by writing a 1 to the respective bit; writing a 0 to a bit location has no effect. Table 4–4 describes the status register.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Status															
Type	R/C/U	R/C/U	R/C/U	R/C/U	R/C/U	R	R	R/C/U	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0

Register: **Status**
Type: Read-only, Read/Clear/Update
Offset: 06h
Default: 0210h

Table 4–4. Status Register

BIT	TYPE	FUNCTION
15	R/C/U	Detected parity error. Bit 15 is set when a parity error is detected.
14	R/C/U	Signaled system error ($\overline{\text{SERR}}$). Bit 14 is set if $\overline{\text{SERR}}$ is enabled in the command register (offset 04h, see Section 4.3) and the bridge signals a system error (SERR). See Section 3.9, *System Error Handling*. 0 = No SERR signaled (default) 1 = Signals SERR
13	R/C/U	Received master abort. Bit 13 is set when a cycle initiated by the bridge on the primary bus has been terminated by a master abort. 0 = No master abort received (default) 1 = Master abort received
12	R/C/U	Received target abort. Bit 12 is set when a cycle initiated by the bridge on the primary bus has been terminated by a target abort. 0 = No target abort received (default) 1 = Target abort received
11	R/C/U	Signaled target abort. Bit 11 is set by the bridge when it terminates a transaction on the primary bus with a target abort. 0 = No target abort signaled by the bridge (default) 1 = Target abort signaled by the bridge
10–9	R	DEVSEL timing. These read-only bits encode the timing of $\overline{\text{P\_DEVSEL}}$ and are hardwired 01b, indicating that the bridge asserts this signal at a medium speed. 01 = Hardwired (default)
8	R/C/U	Data parity error detected. Bit 8 is encoded as: 0 = The conditions for setting this bit have not been met. No parity error detected. (default) 1 = A data parity error occurred and the following conditions were met: a. $\overline{\text{P\_PERR}}$ was asserted by any PCI device including the bridge. b. The bridge was the bus master during the data parity error. c. Bit 6 (parity error response enable) is set in the command register (offset 04h, see Section 4.3).
7	R	Fast back-to-back capable. The bridge does not support fast back-to-back transactions as a target; therefore, bit 7 is hardwired to 0.
6	R	User-definable feature (UDF) support. The PCI2250 does not support the user-definable features; therefore, bit 6 is hardwired to 0.
5	R	66-MHz capable. The PCI2250 operates at a maximum P_CLK frequency of 33 MHz; therefore, bit 5 is hardwired to 0.
4	R	Capabilities list. Bit 4 is read-only and is hardwired to 1, indicating that capabilities additional to standard PCI are implemented. The linked list of PCI power management capabilities is implemented by this function.
3–0	R	Reserved. Bits 3–0 return 0s when read.

4.5 Revision ID Register

The revision ID register indicates the silicon revision of the PCI2250.

Bit	7	6	5	4	3	2	1	0
Name	Revision ID							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	1

Register: **Revision ID**
Type: Read-only
Offset: 08h
Default: 01h (reflects the current revision of the silicon)

4.6 Class Code Register

This register categorizes the PCI2250 as a PCI-to-PCI bridge device (0604h) with a 01h or 00h programming interface. Bit 0 is read-only but its value is aliased with bit 0 of the primary decode control register (offset 57h, see Section 5.9). Bit 0 of the primary decode control register defaults to 1b which means the primary interface is set for subtractive decode. If software writes a 0 to bit 0 of the primary decode control register, then this value is aliased to bit 0 of the class code register and the bridge will positively decode the primary interface.

Bit	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Class code																							
	Base class								Sub class								Programming interface							
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1

Register: **Class code**
Type: Read-only
Offset: 09h
Default: 060401h

4.7 Cache Line Size Register

The cache line size register is programmed by host software to indicate the system cache line size needed by the bridge on memory read line and memory read multiple transactions.

Bit	7	6	5	4	3	2	1	0
Name	Cache line size							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Cache line size**
Type: Read/write
Offset: 0Ch
Default: 00h

4.8 Primary Latency Timer Register

The latency timer register specifies the latency timer for the bridge in units of PCI clock cycles. When the bridge is a primary PCI bus initiator and asserts $\overline{\text{P\_FRAME}}$, the latency timer begins counting from 0. If the latency timer expires before the bridge transaction has terminated, then the bridge terminates the transaction when its $\overline{\text{P\_GNT}}$ is deasserted.

Bit	7	6	5	4	3	2	1	0
Name	Latency timer							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Latency timer**
Type: Read/write
Offset: 0Dh
Default: 00h

4.9 Header Type Register

The header type register is read-only and returns 01h when read, indicating that the PCI2250 configuration space adheres to the PCI-to-PCI bridge configuration. Only the layout for bytes 10h–3Fh of configuration space is considered.

Bit	7	6	5	4	3	2	1	0
Name	Header type							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	1

Register: **Header type**
Type: Read-only
Offset: 0Eh
Default: 01h

4.10 BIST Register

The PCI2250 does not support built-in self test (BIST). The BIST register is read-only and returns the value 00h when read.

Bit	7	6	5	4	3	2	1	0
Name	BIST							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

Register: **BIST**
Type: Read-only
Offset: 0Fh
Default: 00h

4.11 Base Address Register 0

The bridge requires no additional resources. Base address register 0 is read-only and returns 0s when read.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Base address register 0															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Base address register 0															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Base address register 0**
Type: Read-only
Offset: 10h
Default: 0000 0000h

4.12 Base Address Register 1

The bridge requires no additional resources. Base address register 1 is read-only and returns 0s when read.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Base address register 1															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Base address register 1															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Base address register 1**
Type: Read-only
Offset: 14h
Default: 0000 0000h

4.13 Primary Bus Number Register

The primary bus number register indicates the primary bus number to which the bridge is connected. The bridge uses this register, in conjunction with the secondary bus number and subordinate bus number registers, to determine when to forward PCI configuration cycles to the secondary buses.

Bit	7	6	5	4	3	2	1	0
Name	Primary bus number							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Primary bus number**
Type: Read/write
Offset: 18h
Default: 00h

4.14 Secondary Bus Number Register

The secondary bus number register indicates the secondary bus number to which the bridge is connected. The PCI2250 uses this register, in conjunction with the primary bus number and subordinate bus number registers, to determine when to forward PCI configuration cycles to the secondary buses. Configuration cycles directed to the secondary bus are converted to type 0 configuration cycles.

Bit	7	6	5	4	3	2	1	0
Name	Secondary bus number							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Secondary bus number**
Type: Read/write
Offset: 19h
Default: 00h

4.15 Subordinate Bus Number Register

The subordinate bus number register indicates the bus number of the highest numbered bus beyond the primary bus existing behind the bridge. The PCI2250 uses this register, in conjunction with the primary bus number and secondary bus number registers, to determine when to forward PCI configuration cycles to the subordinate buses. Configuration cycles directed to a subordinate bus (not the secondary bus) remain type 1 cycles as the cycle crosses the bridge.

Bit	7	6	5	4	3	2	1	0
Name	Subordinate bus number							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Subordinate bus number**
Type: Read/write
Offset: 1Ah
Default: 00h

4.16 Secondary Bus Latency Timer Register

The secondary bus latency timer specifies the latency timer for the bridge in units of PCI clock cycles. When the bridge is a secondary PCI bus initiator and asserts $\overline{\text{S\_FRAME}}$, the latency timer begins counting from 0. If the latency timer expires before the bridge transaction has terminated, then the bridge terminates the transaction when its $\overline{\text{S\_GNT}}$ is deasserted. The PCI-to-PCI bridge $\overline{\text{S\_GNT}}$ is an internal signal and is removed when another secondary bus master arbitrates for the bus.

Bit	7	6	5	4	3	2	1	0
Name	Secondary bus latency timer							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Secondary bus latency timer**
Type: Read/write
Offset: 1Bh
Default: 00h

4.17 I/O Base Register

The I/O base register is used in decoding I/O addresses to pass through the bridge. The bridge supports 32-bit I/O addressing; thus, bits 3–0 are read-only and default to 0001b. The upper four bits are writable and correspond to address bits AD15–AD12. The lower 12 address bits of the I/O base address are considered 0. Thus, the bottom of the defined I/O address range is aligned on a 4K-byte boundary. The upper 16 address bits of the 32-bit I/O base address corresponds to the contents of the I/O base upper 16 bits register (offset 30h, see Section 4.26).

Bit	7	6	5	4	3	2	1	0
Name	I/O base							
Type	R/W	R/W	R/W	R/W	R	R	R	R
Default	0	0	0	0	0	0	0	1

Register: **I/O base**
Type: Read-only, read/write
Offset: 1Ch
Default: 01h

4.18 I/O Limit Register

The I/O limit register is used in decoding I/O addresses to pass through the bridge. The bridge supports 32-bit I/O addressing; thus, bits 3–0 are read-only and default to 0001b. The upper four bits are writable and correspond to address bits AD15–AD12. The lower 12 address bits of the I/O limit address are considered FFFh. Thus, the top of the defined I/O address range is aligned on a 4K-byte boundary. The upper 16 address bits of the 32-bit I/O limit address corresponds to the contents of the I/O limit upper 16 bits register (offset 32h, see Section 4.27).

Bit	7	6	5	4	3	2	1	0
Name	I/O limit							
Type	R/W	R/W	R/W	R/W	R	R	R	R
Default	0	0	0	0	0	0	0	1

Register: **I/O limit**
Type: Read-only, read/write
Offset: 1Dh
Default: 01h

4.19 Secondary Status Register

The secondary status register is similar in function to the status register (offset 06h, see Section 4.4); however, its bits reflect status conditions of the secondary interface. Bits in this register are cleared by writing a 1 to the respective bit.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Secondary status															
Type	R/C/U	R/C/U	R/C/U	R/C/U	R/C/U	R	R	R/C/U	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

Register: **Secondary status**
Type: Read-only, Read/Clear/Update
Offset: 1Eh
Default: 0200h

Table 4–5. Secondary Status Register

BIT	TYPE	FUNCTION
15	R/C/U	Detected parity error. Bit 15 is set when a parity error is detected on the secondary interface. 0 = No parity error detected on the secondary bus (default) 1 = Parity error detected on the secondary bus
14	R/C/U	Received system error. Bit 14 is set when the secondary interface detects $\overline{\text{S\_SERR}}$ asserted. Note that the bridge never asserts $\overline{\text{S\_SERR}}$. 0 = No $\overline{\text{S\_SERR}}$ detected on the secondary bus (default) 1 = $\overline{\text{S\_SERR}}$ detected on the secondary bus
13	R/C/U	Received master abort. Bit 13 is set when a cycle initiated by the bridge on the secondary bus has been terminated by a master abort. 0 = No master abort received (default) 1 = Bridge master aborted the cycle
12	R/C/U	Received target abort. Bit 12 is set when a cycle initiated by the bridge on the secondary bus has been terminated by a target abort. 0 = No target abort received (default) 1 = Bridge received a target abort
11	R/C/U	Signaled target abort. Bit 11 is set by the bridge when it terminates a transaction on the secondary bus with a target abort. 0 = No target abort signaled (default) 1 = Bridge signaled a target abort
10–9	R	DEVSEL timing. Bits 10 and 9 encode the timing of $\overline{\text{S\_DEVSEL}}$ and are hardwired to 01b, indicating that the bridge asserts this signal at a medium speed.
8	R/C/U	Data parity error detected. 0 = The conditions for setting this bit have not been met 1 = A data parity error occurred and the following conditions were met: a. $\overline{\text{S\_PERR}}$ was asserted by any PCI device including the bridge. b. The bridge was the bus master during the data parity error. c. The parity error response bit (bit 0) is set in the bridge control register (offset 3Eh, se Section 4.32).
7	R	Fast back-to-back capable. Bit 7 is hardwired to 0.
6	R	User-definable feature (UDF) support. Bit 6 is hardwired to 0.
5	R	66-MHz capable. Bit 5 is hardwired to 0.
4–0	R	Reserved. Bits 4–0 return 0s when read.

4.20 Memory Base Register

The memory base register defines the base address of a memory-mapped I/O address range used by the bridge to determine when to forward memory transactions from one interface to the other. The upper 12 bits of this register are read/write and correspond to the address bits AD31–AD20. The lower 20 address bits are considered 0s; thus, the address range is aligned to a 1M-byte boundary. The bottom four bits are read-only and return 0s when read.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Memory base															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Memory base**
Type: Read-only, read/write
Offset: 20h
Default: 0000h

4.21 Memory Limit Register

The memory limit register defines the upper-limit address of a memory-mapped I/O address range used to determine when to forward memory transactions from one interface to the other. The upper 12 bits of this register are read/write and correspond to the address bits AD31–AD20. The lower 20 address bits are considered 1s; thus, the address range is aligned to a 1M-byte boundary. The bottom four bits are read-only and return 0s when read.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Memory limit															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Memory limit**
Type: Read-only, read/write
Offset: 22h
Default: 0000h

4.22 Prefetchable Memory Base Register

The prefetchable memory base register defines the base address of a prefetchable memory address range used by the bridge to determine when to forward memory transactions from one interface to the other. The upper 12 bits of this register are read/write and correspond to the address bits AD31–AD20. The lower 20 address bits are considered 0; thus, the address range is aligned to a 1M-byte boundary. The bottom four bits are read-only and return 0s when read.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Prefetchable memory base															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Prefetchable memory base**
Type: Read-only, read/write
Offset: 24h
Default: 0000h

4.23 Prefetchable Memory Limit Register

The prefetchable memory limit register defines the upper-limit address of a prefetchable memory address range used to determine when to forward memory transactions from one interface to the other. The upper 12 bits of this register are read/write and correspond to the address bits AD31–AD20. The lower 20 address bits are considered 1s; thus, the address range is aligned to a 1M-byte boundary. The bottom four bits are read-only and return 0s when read.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Prefetchable memory limit															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Prefetchable memory limit**
Type: Read-only, read/write
Offset: 26h
Default: 0000h

4.24 Prefetchable Base Upper 32 Bits Register

The PCI2250 does not support 64-bit addressing; thus, the prefetchable base upper 32-bit register is read-only and returns 0s when read.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Prefetchable base upper 32 bits															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Prefetchable base upper 32 bits															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Prefetchable base upper 32 bits**
Type: Read-only
Offset: 28h
Default: 0000 0000h

4.25 Prefetchable Limit Upper 32 Bits Register

The PCI2250 does not support 64-bit addressing; thus the prefetchable limit upper 32-bit register is read-only and returns 0s when read.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Prefetchable limit upper 32 bits															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Prefetchable limit upper 32 bits															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Prefetchable limit upper 32 bits**
Type: Read-only
Offset: 2Ch
Default: 0000 0000h

4.26 I/O Base Upper 16 Bits Register

The I/O base upper 16 bits register specifies the upper 16 bits corresponding to AD31–AD16 of the 32-bit address that specifies the base of the I/O range to forward from the primary PCI bus to the secondary PCI bus.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	I/O base upper 16 bits															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **I/O base upper 16 bits**
Type: Read/Write
Offset: 30h
Default: 0000h

4.27 I/O Limit Upper 16 Bits Register

The I/O limit upper 16-bits register specifies the upper 16 bits corresponding to AD31–AD16 of the 32-bit address that specifies the upper limit of the I/O range to forward from the primary PCI bus to the secondary PCI bus.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	I/O limit upper 16 bits															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **I/O limit upper 16 bits**
Type: Read/Write
Offset: 32h
Default: 0000h

4.28 Capability Pointer Register

The capability pointer register provides the pointer to the PCI configuration header where the PCI power management register block resides. The capability pointer provides access to the first item in the linked list of capabilities. The capability pointer register is read-only and returns DCh when read, indicating the power management registers are located at PCI header offset DCh.

Bit	7	6	5	4	3	2	1	0
Name	Capability pointer register							
Type	R	R	R	R	R	R	R	R
Default	1	1	0	1	1	1	0	0

Register: **capability pointer**
Type: Read-only
Offset: 34h
Default: DCh

4.29 Expansion ROM Base Address Register

The PCI2250 does not implement the expansion ROM remapping feature. The expansion ROM base address register returns all 0s when read.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Expansion ROM base address															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Name	Expansion ROM base address															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Expansion ROM base address**
Type: Read-only
Offset: 38h
Default: 0000 0000h

4.30 Interrupt Line Register

The interrupt line register is read/write and is used to communicate interrupt line routing information. Since the bridge does not implement an interrupt signal terminal, this register defaults to FFh.

Bit	7	6	5	4	3	2	1	0
Name	Interrupt line							
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	1	1	1	1	1	1	1	1

Register: **Interrupt line**
Type: Read/write
Offset: 3Ch
Default: FFh

4.31 Interrupt Pin Register

The bridge default state does not implement any interrupt terminals. Reads from bits 7–0 of this register return 0s.

Bit	7	6	5	4	3	2	1	0
Name	Interrupt pin							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

Register: **Interrupt pin**
Type: Read-only
Offset: 3Dh
Default: 00h

4.32 Bridge Control Register

The bridge control register provides many of the same controls for the secondary interface that are provided by the command register (offset 04h, see Section 4.3) for the primary interface. Some bits affect the operation of both interfaces.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Bridge control															
Type	R	R	R	R	R/W	RCU	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Bridge control**
Type: Read-only, read/write (see individual bit descriptions)
Offset: 3Eh
Default: 0000h

Table 4–6. Bridge Control Register

BIT	TYPE	FUNCTION
15–12	R	Reserved. Bits 15–12 return 0s when read.
11	R/W	Discard timer $\overline{\text{SERR}}$ enable. 0 = $\overline{\text{SERR}}$ signaling disabled for primary discard timeouts (default) 1 = $\overline{\text{SERR}}$ signaling enabled for primary discard timeouts
10	RCU	Discard timer status. Once set, this bit must be cleared by writing 1 to this bit. 0 = No discard timer error (default) 1 = Discard timer error. Either primary or secondary discard timer expired and a delayed transaction was discarded from the queue in the bridge.
9	R/W	Secondary discard timer. Selects the number of PCI clocks that the bridge will wait for a master on the secondary interface to repeat a delayed transaction request. 0 = Secondary discard timer counts 2^{15} PCI clock cycles (default) 1 = Secondary discard timer counts 2^{10} PCI clock cycles
8	R/W	Primary discard timer. Selects the number of PCI clocks that the bridge will wait for a master on the primary interface to repeat a delayed transaction request. 0 = The primary discard timer counts 2^{15} PCI clock cycles (default) 1 = The primary discard timer counts 2^{10} PCI clock cycles
7	R	Fast back-to-back capable. The bridge never generates fast back-to-back transactions to different secondary devices. Bit 7 returns 0 when read.
6	R/W	Secondary bus reset. When bit 6 is set, the secondary reset signal ($\overline{\text{S\_RST}}$) is asserted. $\overline{\text{S\_RST}}$ is deasserted by resetting this bit. Bit 6 is encoded as: 0 = Do not force the assertion of $\overline{\text{S\_RST}}$ (default). 1 = Force the assertion of $\overline{\text{S\_RST}}$.
5	R/W	Master abort mode. Bit 5 controls how the bridge responds to a master abort that occurs on either interface when the bridge is the master. If this bit is set and the posted write transaction has completed on the requesting interface, and $\overline{\text{SERR}}$ enable (bit 8) of the command register (offset 04h, see Section 4.3) is 1, then P_$\overline{\text{SERR}}$ is asserted when a master abort occurs. If the transaction has not completed, then a target abort is signaled. If the bit is cleared, then all 1s are returned on reads and write data is accepted and discarded when a transaction that crosses the bridge is terminated with master abort. The default state of bit 5 after a reset is 0. 0 = Do not report master aborts (return FFFF FFFFh on reads and discard data on writes) (default). 1 = Report master aborts by signaling target abort if possible, or if $\overline{\text{SERR}}$ is enabled via bit 1 of this register, by asserting $\overline{\text{SERR}}$.
4	R	Reserved. Bit 4 returns 0 when read.
3	R/W	VGA enable. When bit 3 is set, the bridge positively decodes and forwards VGA-compatible memory addresses in the video frame buffer range 000A 0000h–000B FFFFh, I/O addresses in the range 03B0h–03BBh, and 03C0–03DFh from the primary to the secondary interface, independent of the I/O and memory address ranges. When this bit is set, the bridge blocks forwarding of these addresses from the secondary to the primary. Reset clears this bit. Bit 3 is encoded as: 0 = Do not forward VGA-compatible memory and I/O addresses from the primary to the secondary interface (default). 1 = Forward VGA-compatible memory and I/O addresses from the primary to the secondary, independent of the I/O and memory address ranges and independent of the ISA enable bit.

Table 4–6. Bridge Control Register (Continued0)

BIT	TYPE	FUNCTION
2	R/W	ISA enable. When bit 2 is set, the bridge blocks the forwarding of ISA I/O transactions from the primary to the secondary, addressing the last 768 bytes in each 1K-byte block. This applies only to the addresses (defined by the I/O window registers) that are located in the first 64K bytes of PCI I/O address space. From the secondary to the primary, I/O transactions are forwarded if they address the last 768 bytes in each 1K-byte block in the address range specified in the I/O window registers. Bit 2 is encoded as: 0 = Forward all I/O addresses in the address range defined by the I/O base and I/O limit registers (default). 1 = Block forwarding of ISA I/O addresses in the address range defined by the I/O base and I/O limit registers when these I/O addresses are in the first 64K bytes of PCI I/O address space and address the top 768 bytes of each 1K-byte block.
1	R/W	$\overline{\text{SERR}}$ enable. Bit 1 controls the forwarding of secondary interface $\overline{\text{SERR}}$ assertions to the primary interface. Only when this bit is set will the bridge forward S_$\overline{\text{SERR}}$ to the primary bus signal P_$\overline{\text{SERR}}$. For the primary interface to assert $\overline{\text{SERR}}$, bit 8 of the command register (offset 04h, see Section 4.3) must be set. 0 = $\overline{\text{SERR}}$ disabled (default) 1 = $\overline{\text{SERR}}$ enabled
0	R/W	Parity error response enable. Bit 0 controls the bridge response to parity errors on the secondary interface. When this bit is set, the bridge asserts S_PERR to report parity errors on the secondary interface. 0 = Ignore address and parity errors on the secondary interface (default). 1 = Enable parity error reporting and detection on the secondary interface.

5 Extension Registers

The TI extension registers are those registers that lie outside the standard PCI-to-PCI bridge device configuration space (i.e., registers 40h–FFh in PCI configuration space in the PCI2250). These registers can be accessed through configuration reads and writes. The TI extension registers add flexibility and performance benefits to the standard PCI-to-PCI bridge. The TI extension registers are not reset on the transition from D3 to D0.

5.1 Chip Control Register

The chip control register is read/write and has a default value of 00h. This register is used to control the functionality of certain PCI transactions. See Table 5–1 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Chip control							
Type	R	R	R	R/W	R	R	R/W	R
Default	0	0	0	0	0	0	0	0

Register: **Chip control**
Type: Read/Write, Read–only
Offset: 40h
Default: 00h

Table 5–1. Chip Control Register

BIT	TYPE	FUNCTION
7–5	R	Reserved. Bits 7–5 return 0s when read.
4	R/W	Memory read prefetch. When cleared, bit 4 enables the memory read prefetch. 0 = Upstream memory reads are enabled (default) 1 = Upstream memory reads are disabled
3–2	R	Reserved. Bits 3 and 2 return 0s when read.
1	R/W	Reserved
0	R	Reserved. Bit 0 returns 0 when read.

5.2 Extended Diagnostic Register

The extended diagnostic register is read or write and has a default value of 00h. Bit 0 of this register is used to reset both the PCI2250 and the secondary bus.

Bit	7	6	5	4	3	2	1	0
Name	Extended diagnostic							
Type	R	R	R	R	R	R	R	W
Default	0	0	0	0	0	0	0	0

Register: **Extended diagnostic**
Type: Read-only, Write-only
Offset: 41h
Default: 00h

Table 5–2. Extended Diagnostic Register

BIT	TYPE	FUNCTION
7–1	R	Reserved. Bits 7–1 return 0s when read.
0	W	Writing a 1 to this bit causes the PCI2250 to set bit 6 of the bridge control register (offset 3Eh, see Section 4.32) and then internally reset the PCI2250. Bit 6 of the bridge control register will not be reset by the internal reset. Bit 0 is self-clearing.

5.3 Arbiter Control Register

The arbiter control register is used for the bridge's internal arbiter. The arbitration scheme used is a two-tier rotational arbitration. The PCI2250 bridge is the only secondary bus initiator that defaults to the higher priority arbitration tier.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Arbiter control															
Type	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

Register: **Arbiter control**
Type: Read-only, Read/Write
Offset: 42h
Default: 0200h

Table 5–3. Arbiter Control Register

BIT	TYPE	FUNCTION
15–10	R	Reserved. Bits 15–10 return 0s when read.
9	R/W	Bridge tier select. This bit determines in which tier the bridge is placed in the two-tier arbitration scheme. 0 = Lowest priority tier 1 = Highest priority tier (default)
8–4	R	Reserved. Bits 8–4 return 0s when read.
3	R/W	GNT3 tier select. This bit determines in which tier the S_GNT3 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier
2	R/W	GNT2 tier select. This bit determines in which tier the S_GNT2 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier
1	R/W	GNT1 tier select. This bit determines in which tier the S_GNT1 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier
0	R/W	GNT0 tier select. This bit determines in which tier the S_GNT0 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier

5.4 Extension Window Base 0, 1 Registers

The bridge supports two extension windows that define an address range decoded as described in the window enable register and window map register. The extension window base registers define the 32-bit base address of the window.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Extension window base 0, 1															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Extension window base 0, 1															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Extension window base 0, 1**
Type: Read-only, Read/Write
Offset: 44h, 4Ch
Default: 0000 0000h

5.5 Extension Window Limit 0, 1 Registers

The bridge supports two extension windows. Each window defines an address range that is decoded as described in the window enable register and window map register. The extension window limit registers define the 32-bit limit address of the window.

Bits 0 and 1 of this register determine whether the extension window is a prefetchable memory window, a nonprefetchable window, or an I/O window. These bits are encoded as:

00 = Nonprefetchable memory
01 = Prefetchable memory
1x = I/O

Memory windows have a 4–Kbyte granularity and I/O windows have a doubleword (4-byte) granularity. When a memory window is selected, bits 11–2 have no effect and are assumed to be 1s for the limit register and 0s for the base register. This is consistent with the 4K-byte granularity of the memory windows.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Name	Extension window limit 0, 1															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Extension window limit 0, 1															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Extension window limit 0, 1**
Type: Read/Write
Offset: 48h, 50h
Default: 0000 0000h

5.6 Extension Window Enable Register

The decode of the extension windows is enabled through bits 0 and 1 of this register. See Table 5–4 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Extension window enable							
Type	R	R	R	R	R	R	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Extension window enable**
Type: Read-only, Read/Write
Offset: 54h
Default: 00h

Table 5–4. Extension Window Enable Register

BIT	TYPE	FUNCTION
7–2	R	Reserved. Bits 7–2 return 0s when read.
1	R/W	Extension window 1 interface enable 0 = Disable window 1 (default) 1 = Enable window 1
0	R/W	Extension window 0 interface enable 0 = Disable window 0 (default) 1 = Enable window 0

5.7 Extension Window Map Register

The inclusion or exclusion of the extension windows on the primary interface is selected through bits 0 and 1 of this register. The bit descriptions discuss the decode in reference to the primary interface. The secondary interface is the negative decode of the primary interface. Regions excluded on the primary interface can be positively decoded on the secondary interface if negative decoding is disabled on the secondary interface. See Table 5–5 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Extension window map							
Type	R	R	R	R	R	R	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Extension window map**
Type: Read-only, Read/Write
Offset: 55h
Default: 00h

Table 5–5. Extension Window Map Register

BIT	TYPE	FUNCTION
7–2	R	Reserved. Bits 7–2 return 0s when read.
1	R/W	Extension window 1 interface include/exclude 0 = Extension window 1 included in primary interface decode (default) 1 = Extension window 1 excluded in primary interface decode
0	R/W	Extension window 0 interface include/exclude 0 = Extension window 0 included in primary interface decode (default) 1 = Extension window 0 excluded in primary interface decode

5.8 Secondary Decode Control Register

The secondary decode control register is used to enable/disable the secondary-bus negative decoding. Only through this register can an extension window be defined for positive decoding or excluded from negative decoding from the secondary bus to the primary bus. The window interface bits in the window control registers must be set for the extension window definitions in this register to have meaning.

Bit	7	6	5	4	3	2	1	0
Name	Secondary decode control							
Type	R	R	R	R	R	R/W	R/W	R/W
Default	0	0	0	0	0	1	1	0

Register: **Secondary decode control**
Type: Read-only, Read/Write
Offset: 56h
Default: 06h

Table 5–6. Secondary Decode Control Register

BIT	TYPE	FUNCTION
7–3	R	Reserved. Bits 7–3 return 0s when read.
2	R/W	Secondary-bus subtractive decode speed. The bridge defaults to subtractive decoding after slow decode speed (four clocks after $\overline{\text{FRAME}}$ is asserted). Bit 0 must be set to enable subtractive decoding. When bit 0 and this bit are set, subtractive decoding is enabled at slow decode speed. This bit is encoded as: 0 = Selects normal subtractive decode speed. 1 = Selects subtractive decode in the slow decode time slot (default).
1	R/W	Secondary bus negative decode enable. The bridge defaults to negative decoding on the secondary PCI bus. All transactions that do not fall into windows positively decoded from the primary to the secondary are passed through to the primary bus. This bit is encoded as: 0 = Disable secondary-bus negative decoding. 1 = Enable secondary-bus negative decoding (default).
0	R/W	Secondary-bus subtractive decode enable. The bridge defaults to negative decoding on the secondary PCI bus. When bit 0 is set, the bridge uses subtractive decoding on the secondary bus. When the bridge is using negative decoding on the secondary, all transactions not claimed by a slow device on the secondary bus are passed through the bridge to the primary bus. This bit is encoded as: 0 = Disable secondary bus subtractive decoding (default). 1 = Enable secondary bus subtractive decoding.

5.9 Primary Decode Control Register

This register is used to enable and disable the primary bus subtractive decoding and to select the primary bus subtractive decode speed. The bridge defaults to primary bus subtractive decoding enabled (bit 0 is set to 1b). Bit 0 of this register is aliased to bit 0 of the class code register (offset 09h, see Section 4.6) so that the class code register reflects whether or not subtractive decoding is enabled on the primary interface. See Table 5–7 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Primary decode control							
Type	R	R	R	R	R	R	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Primary decode control**
Type: Read-only, Read/Write
Offset: 57h
Default: 00h

Table 5–7. Primary Decode Control Register

BIT	TYPE	FUNCTION
7–2	R	Reserved. Bits 7–2 return 0s when read.
1	R/W	Primary-bus subtractive decode speed. The bridge defaults to subtractive decoding after slow decode speed (four clocks after $\overline{\text{FRAME}}$ is asserted). Bit 0 must be set to enable subtractive decoding. When bit 0 and this bit are set, subtractive decoding is enabled at slow decode speed. This bit is encoded as: 0 = Selects normal subtractive decode speed on primary bus (default) 1 = Selects subtractive decode in the slow decode time slot on the primary bus
0	R/W	Primary-bus subtractive decode enable. The bridge defaults to subtractive decoding disabled from the primary to secondary PCI bus. Each PCI bus may only have one subtractive decode device. 0 = Disable primary bus subtractive decoding 1 = Enable primary bus subtractive decoding (default)

5.10 Port Decode Enable Register

The port decode enable register is used to select which serial and parallel port addresses are positively decoded from the bridge primary bus to the secondary bus. See Table 5–8 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Port decode enable							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Port decode enable**
Type: Read-only, Read/Write
Offset: 58h
Default: 00h

Table 5–8. Port Decode Enable Register

BIT	TYPE	FUNCTION
7	R	Reserved. Bit 7 returns 0 when read.
6	R/W	LPT3 enable. When bit 6 is set, the address ranges 278h–27Fh and 678h–67Bh are positively decoded and the cycles passed to the secondary bus based on the setting of bit 6 of the port decode map register (offset 5Ah, see Section 5.12).
5	R/W	LPT2 enable. When bit 5 is set, the address ranges 378h–37Fh and 778h–77Bh are positively decoded and the cycles passed to the secondary bus based on the setting of bit 5 of the port decode map register (offset 5Ah, see Section 5.12).
4	R/W	LPT1 enable. When bit 4 is set, the address ranges 3BCh–3BFh and 7BCh–7BFh are positively decoded and the cycles passed to the secondary bus based on the setting of bit 4 of the port decode map register (offset 5Ah, see Section 5.12).
3	R/W	COM4 enable. When bit 3 is set, the address range 2E8h–2EFh is positively decoded and the cycles passed to the secondary bus based on the setting of bit 3 of the port decode map register (offset 5Ah, see Section 5.12).
2	R/W	COM3 enable. When bit 2 is set, the address range 3E8h–3EFh is positively decoded and the cycles passed to the secondary bus based on the setting of bit 2 of the port decode map register (offset 5Ah, see Section 5.12).
1	R/W	COM2 enable. When bit 1 is set, the address range 2F8h–2FFh is positively decoded and the cycles passed to the secondary bus based on the setting of bit 1 of the port decode map register (offset 5Ah, see Section 5.12).
0	R/W	COM1 enable. When bit 0 is set, the address range 3F8h–3FFh is positively decoded and the cycles passed to the secondary bus based on the setting of bit 0 of the port decode map register (offset 5Ah, see Section 5.12).

5.11 Buffer Control Register

The buffer control register allows software to enable/disable write posting and control memory read burst prefetching. The buffer control register also enables/disables the posted memory write reconnect feature. See Table 5–9 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Buffer control							
Type	R	R	R	R/W	R	R/W	R/W	R/W
Default	0	0	0	0	0	1	1	1

Register: **Buffer control**
Type: Read-only, Read/Write
Offset: 59h
Default: 07h

Table 5–9. Buffer Control Register

BIT	TYPE	FUNCTION
7–5	R	Reserved. Bits 7 through 5 return 0s when read.
4	R/W	Upstream MRM/MRL read burst enable. By default, the PCI2250 is set to memory read burst a single cache line. By setting this bit to 1, the PCI2250 will memory read burst multiple cache lines or until the FIFO is full. To utilize this feature, bit 4 of the chip control register (offset 40h, see Section 5.1) must be set to 0. 0 = Disabled (default) 1 = Enabled
3	R	Reserved. Bit 3 returns 0 when read.
2	R/W	Downstream memory read burst enable. The bridge defaults to downstream memory read bursting enabled. Bit 2 enables downstream memory read bursting in prefetchable windows. This bit is encoded as: 0 = Disabled 1 = Enabled (default)
1	R/W	Secondary-to-primary write posting enable. Enables posting of write data to and from the primary interface. If bit 1 is not set, the bridge must drain any data in its buffers before accepting data to or from the primary interface. Each data word must then be accepted by the target before the bridge can accept the next word from the source master. The bridge must not release the source master until the last word is accepted by the target. Operating with the write posting enabled enhances system performance. 0 = Write posting disabled 1 = Write posting enabled (default)
0	R/W	Primary-to-secondary write posting enable. Enables posting of write data to and from the secondary interface. If bit 0 is not set, then the bridge must drain any data in its buffers before accepting data to or from the secondary interface. Each data word must then be accepted by the target before the bridge can accept the next word from the source master. The bridge must not release the source master until the last word is accepted by the target. Operating with the write posting enabled enhances system performance. 0 = Write posting disabled 1 = Write posting enabled (default)

5.12 Port Decode Map Register

The port decode map register is used to select whether the serial- and parallel-port address ranges positively decoded from the primary bridge interface to the secondary interface are included or excluded from the primary interface. For example, if bit 0 is set, then addresses in the range of 3F8h–3FFh are positively decoded on the primary bus. If bit 0 is cleared and an I/O window is enabled that covers the range from 3F8h–3FFh, then these addresses are not claimed by the bridge. See Table 5–10 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Port decode map							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Port decode map**
Type: Read-only, Read/Write
Offset: 5Ah
Default: 00h

Table 5–10. Port Decode Map Register

BIT	TYPE	FUNCTION
7	R	Reserved. Bit 7 returns 0 when read.
6	R/W	LPT3 include/exclude. Bit 6 is encoded as: 0 = 278h–27Fh and 678h–67Bh excluded from the primary bus (default) 1 = 278h–27Fh and 678h–67Bh positively decoded on the primary bus
5	R/W	LPT2 include/exclude. Bit 5 is encoded as: 0 = 378h–37Fh and 778h–77Bh excluded from the primary bus (default) 1 = 378h–37Fh and 778h–77Bh positively decoded on the primary bus
4	R/W	LPT1 include/exclude. Bit 4 is encoded as: 0 = 3BCh–3BFh and 7BCh–7BFh excluded from the primary bus (default) 1 = 3BCh–3BFh and 7BCh–7BFh positively decoded on the primary bus
3	R/W	COM4 include/exclude. Bit 3 is encoded as: 0 = 2E8h–2EFh excluded from the primary bus (default) 1 = 2E8h–2EFh positively decoded on the primary bus
2	R/W	COM3 include/exclude. Bit 2 is encoded as: 0 = 3E8h–3EFh excluded from the primary bus (default) 1 = 3E8h–3EFh positively decoded on the primary bus
1	R/W	COM2 include/exclude. Bit 1 is encoded as: 0 = 2F8h–2FFh excluded from the primary bus (default) 1 = 2F8h–2FFh positively decoded on the primary bus
0	R/W	COM1 include/exclude. Bit 0 is encoded as: 0 = 3F8h–3FFh excluded from the primary bus (default) 1 = 3F8h–3FFh positively decoded on the primary bus

5.13 Clock Run Control Register

The clock run control register controls the PCI clock-run mode enable/disable. It is also used to enable the keep-clock-running feature. Bit 0 reflects the status of the secondary clock. There are two clock run modes supported on the secondary bus. The bridge can be configured to stop the secondary PCI clock only in response to a request from the primary bus to stop the clock or it can be configured to stop the secondary clock whenever the secondary bus is idle and there are no transaction requests from the primary bus.

There are two conditions for restarting the secondary clock. A downstream transaction restarts the secondary clock, or if the $\overline{\text{S\_CLKRUN}}$ signal is asserted, the secondary clock is restarted. See Table 5–11 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Clock run control							
Type	R	R	R	R/W	R/W	R/W	R/W	R
Default	0	0	0	0	0	0	0	0

Register: **Clock run control**
Type: Read-only, Read/Write
Offset: 5Bh
Default: 00h

Table 5–11. Clock Run Control Register

BIT	TYPE	FUNCTION
7–5	R	Reserved. Bits 7–5 return 0s when read.
4	R/W	Clock run mode. Bit 4 is encoded as: 0 = Stop the secondary clock only on request from the primary bus (default). 1 = Stop the secondary clock whenever the secondary bus is idle and there are no requests from the primary bus.
3	R/W	Primary clock run enable. Bit 3 must be enabled for the bridge to respond to requests by the central resource on the primary bus to stop the clock. 0 = Disable clock run (default) 1 = Enable clock run
2	R/W	Primary keep clock. When bit 2 is set, it causes the bridge to request that the central resource keep the PCI clock running. 0 = Allow primary clock to stop if secondary clock stopped (default) 1 = Always keep primary clock running
1	R/W	Secondary clock run enable 0 = Disable clock run for secondary (default) 1 = Enable clock run for secondary
0	R	Secondary clock status bit. If the clock is stopped, this bit is 1. If the clock is running, this bit is 0. 0 = Secondary clock not stopped (default) 1 = Secondary clock stopped

5.14 Diagnostic Control Register

The diagnostic control register is used for bridge diagnostics. See Table 5–12 for a complete description of the register contents.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Diagnostic control															
Type	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0

Register: **Diagnostic control**
Type: Read/Write, Read-only
Offset: 5Ch–5Dh
Default: 1040h

Table 5–12. Diagnostic Control Register

BIT	TYPE	FUNCTION
15	R/W	Arbiter performance enhancement feature. When enabled, this feature provides automatic tier operation for bus masters that have been retried or that have pending delayed transactions. In this case, the bus master gets promoted to the highest priority tier. 0 = Disabled (default) 1 = Enabled
14	R/W	Parity mode. Bit 14 is encoded as: 0 = Parity error passing enabled (default) 1 = Parity error passing disabled
13	R/W	Upstream lock enable. The bridge default is to disable upstream lock. When set, bit 13 enables upstream resource locking. This bit is encoded as: 0 = Selects upstream lock disabled (default) 1 = Selects upstream lock enabled
12	R/W	Downstream lock enable. The bridge default is to enable downstream lock. When set, bit 12 enables downstream resource locking. This bit is encoded as: 0 = Selects downstream lock disabled 1 = Selects downstream lock enabled (default)
11	R/W	Secondary-bus decode speed. The bridge defaults to medium decode speed on the secondary bus. Bit 11 selects between medium and slow decode speed. This bit is encoded as: 0 = Secondary bus decodes at medium decode speed (default) 1 = Secondary bus decodes at slow decode speed
10	R/W	Primary-bus decode speed. The bridge defaults to medium decode speed on the primary bus. Bit 10 selects between medium and slow decode speed. This bit is encoded as: 0 = Primary bus decodes at medium decode speed (default) 1 = Primary bus decodes at slow decode speed
9–8	R	Reserved. Bits 9 and 8 return 0s when read.
7	R/W	Arbiter timeout. When set, bit 0 enables SERR reporting when the arbiter timer expires (times out). 0 = SERR on arbiter timeout disabled (default) 1 = SERR on arbiter timeout enabled
6	R/W	Transaction ordering enable 0 = Disabled 1 = Enabled (default)
5	R/W	Secondary initial data phase counter extension 0 = Normal 16 clock to initial data phase (default) 1 = Extends initial data phase to 64 clocks
4	R/W	Primary initial data phase counter disable 0 = Enable 16 clocks initial data phase counter (default) 1 = Disable 16 clock initial data phase counter Note: The secondary initial data phase counter is always enabled.
3	R/W	Primary initial data phase counter extension 0 = Normal 16 clocks to initial data phase (default) 1 = Extends initial data phase to 64 clocks
2	R/W	Immediate retry mode 0 = Immediate retry mode enabled (default) 1 = Immediate retry mode disabled
1	R/W	Bus parking bit. This bit determines where the PCI2250 internal arbiter parks the secondary bus. When this bit is set, the arbiter parks the secondary bus on the bridge. When this bit is cleared, the arbiter parks the bus on the last device mastering the secondary bus. This bit is encoded as: 0 = Park the secondary bus on the last secondary bus master (default) 1 = Park the secondary bus on the bridge
0	R/W	TI internal test mode bit.

5.15 Diagnostic Status Register

The diagnostic status register is used to reflect the bridge diagnostic status. See Table 5–13 for a complete description of the register contents.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Diagnostic status															
Type	R	R	R	R	R/C/U	R/C/U	R	R	R/C/U	R	R	R	R	R	R	R/C/U
Default	0	0	0	0	X	X	0	0	0	0	0	0	0	X	X	X

Register: **Diagnostic status**
Type: Read-only, Read/Write
Offset: 5Eh
Default: 0X0Xh

Table 5–13. Diagnostic Status Register

BIT	TYPE	FUNCTION
15–12	R	Reserved. Bits 15–12 return 0s when read.
11	R/C/U	Bridge detected a parity error while mastering on the secondary bus. When set, bit 11 indicates that the secondary bus master detected a parity error. Writing a 1 to this bit clears it. 0 = No parity error detected 1 = Parity error detected
10	R/C/U	Bridge detected a parity error while mastering on the primary bus. When set, bit 10 indicates that the primary bus master detected a parity error. Writing a 1 to this bit clears it. 0 = No parity error detected 1 = Parity error detected
9	R	MS1 status. Returns the logical value of the MS1/BPCC input.
8	R	MS0 status. Returns the logical value of the MS0 input.
7	R/C/U	Arbiter timeout SERR status. When set, bit 0 indicates that SERR has occurred due to the expiration of the arbiter timer. Writing a 1 to this bit clears it. 0 = No SERR (default) 1 = SERR occurred due to an arbiter timeout
6	R	Reserved. Bit 6 returns 0 when read.
5	R	HS_SWITCH status. This registers returns the logical value of the S_MFUNC input regardless of the value of MS0/MS1.
4–3	R	Reserved
2	R	External arbiter enable pin status. Bit 2 contains the current state of the external pin external arbiter enable. 0 = Signal low 1 = Signal high
1	R	Serial EEPROM block status. Bit 1 indicates the status of the serial EEPROM block. When set, bit 1 indicates that the serial EEPROM block is busy. 0 = Serial EEPROM block not busy 1 = Serial EEPROM block busy
0	R/C/U	Arbiter timeout status. Bit 0 indicates the status of the arbiter timer. When set, bit 0 indicates that a bus master did not begin the cycle within 16 clocks. Writing a 1 to this bit clears it. This bit is encoded as: 0 = No timeout (default). 1 = Master requesting the bus did not start cycle within 16 clocks.

5.16 Arbiter Request Mask Register

The arbiter request mask register contains the SERR enable on arbiter timeouts and the request mask controls. See Table 5–14 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Arbiter request mask							
Type	R	R/W	R	R	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0

Register: **Arbiter request mask**
Type: Read-only, Read/Write
Offset: 62h
Default: 00h

Table 5–14. Arbiter Request Mask Register

BIT	TYPE	FUNCTION
7	R	Reserved. Bit 7 returns 0 when read.
6	R/W	Timeout automatic masking enable 0 = Masking not automatic (default) 1 = Allow masking after 16-clock timeout
5–4	R	Reserved. Bits 5 and 4 return 0s when read.
3	R/W	Request 3 ($\overline{\text{REQ3}}$) mask bit 0 = Use request 3 (default) 1 = Ignore request 3
2	R/W	Request 2 ($\overline{\text{REQ2}}$) mask bit 0 = Use request 2 (default) 1 = Ignore request 2
1	R/W	Request 1 ($\overline{\text{REQ1}}$) mask bit 0 = Use request 1 (default) 1 = Ignore request 1
0	R/W	Request 0 ($\overline{\text{REQ0}}$) mask bit 0 = Use request 0 (default) 1 = Ignore request 0

5.17 Arbiter Timeout Status Register

The arbiter timeout status register contains the status of each request (request 5–0) timeout. The timeout status bit for the respective request is set if the device did not assert $\overline{\text{FRAME}}$ after 16 clocks. See Table 5–15 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Arbiter timeout status							
Type	R	R	R	R	R/C/U	R/C/U	R/C/U	R/C/U
Default	0	0	0	0	0	0	0	0

Register: **Arbiter timeout status**
Type: Read-only
Offset: 63h
Default: 00h

Table 5–15. Arbiter Timeout Status Register

BIT	TYPE	FUNCTION
7–4	R	Reserved. Bits 7–4 return 0s when read.
3	R/C/U	Request 3 timeout status. Cleared by writing a 1. 0 = No timeout (default) 1 = Timeout has occurred
2	R/C/U	Request 2 timeout status. Cleared by writing a 1. 0 = No timeout (default) 1 = Timeout has occurred
1	R/C/U	Request 1 timeout status. Cleared by writing a 1. 0 = No timeout (default) 1 = Timeout has occurred
0	R/C/U	Request 0 timeout status. Cleared by writing a 1. 0 = No timeout (default) 1 = Timeout has occurred

5.18 P_SERR Event Disable Register

The P_SERR event disable register is used to enable/disable SERR event on the primary interface. All events are enabled by default. See Table 5–16 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	P_SERR event disable							
Type	R	R/W	R/W	R/W	R/W	R/W	R/W	R
Default	0	0	0	0	0	0	0	0

Register: **P_SERR event disable**
Type: Read-only, Read/Write
Offset: 64h
Default: 00h

Table 5–16. P_SERR Event Disable Register

BIT	TYPE	FUNCTION
7	R	Reserved. Bit 7 returns 0 when read.
6	R/W	Master delayed read time-out 0 = P_SERR signaled on a master time-out after 2^{24} retries on a delayed read (default). 1 = P_SERR is not signaled on a master time-out.
5	R/W	Master delayed write time-out. 0 = P_SERR signaled on a master time-out after 2^{24} retries on a delayed write (default). 1 = P_SERR is not signaled on a master time-out.
4	R/W	Master abort on posted write transactions. When set, bit 4 enables P_SERR reporting on master aborts on posted write transactions. 0 = Master aborts on posted writes enabled (default) 1 = Master aborts on posted writes disabled
3	R/W	Target abort on posted writes. When set, bit 3 enables P_SERR reporting on target aborts on posted write transactions. 0 = Target aborts on posted writes enabled (default). 1 = Target aborts on posted writes disabled.
2	R/W	Master posted write time-out 0 = P_SERR signaled on a master time-out after 2^{24} retries on a posted write (default). 1 = P_SERR is not signaled on a master time-out.
1	R/W	Posted write parity error 0 = P_SERR signaled on a posted write parity error (default). 1 = P_SERR is not signaled on a posted write parity error.
0	R	Reserved. Bit 0 returns 0 when read.

5.19 Secondary Clock Control Register

The secondary clock control register is used to control the secondary clock outputs. See Table 5–17 for a complete description of the register contents.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Secondary clock control															
Type	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Secondary clock control**
Type: Read-only, Read/Write
Offset: 68h
Default: 0000h

Table 5–17. Secondary Clock Control Register

BIT	TYPE	FUNCTION
15–9	R	Reserved. Bits 15–9 return 0s when read.
8	R/W	Clockout4 disable. 0 = Clockout4 enabled (default) 1 = Clockout4 disabled and driven high
7–6	R/W	Clockout3 disable. 00, 01, 10 = Clockout3 enabled (00 default) 11 = Clockout3 disabled and driven high
5–4	R/W	Clockout2 disable. 00, 01, 10 = Clockout2 enabled (00 default) 11 = Clockout2 disabled and driven high
3–2	R/W	Clockout1 disable. 00, 01, 10 = Clockout1 enabled (00 default) 11 = Clockout1 disabled and driven high
1–0	R/W	Clockout0 disable. 00, 01, 10 = Clockout0 enabled (00 default) 11 = Clockout0 disabled and driven high

5.20 P_SERR Status Register

The P_SERR status register indicates what caused a SERR event on the primary interface. See Table 5–18 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	P_SERR status							
Type	R	R/C/U	R/C/U	R/C/U	R/C/U	R/C/U	R/C/U	R
Default	0	0	0	0	0	0	0	0

Register: **P_SERR status**
Type: Read-only, Read/Clear/Update
Offset: 6Ah
Default: 00h

Table 5–18. P_SERR Status Register

BIT	TYPE	FUNCTION
7	R	Reserved. Bit 7 returns 0 when read.
6	R/C/U	Master delayed read time-out. A 1 indicates that P_SERR was signaled because of a master time-out after 2^{24} retries on a delayed read.
5	R/C/U	Master delayed write time-out. A 1 indicates that P_SERR was signaled because of a master time-out after 2^{24} retries on a delayed write.
4	R/C/U	Master abort on posted write transactions. A 1 indicates that P_SERR was signaled because of a master abort on a posted write.
3	R/C/U	Target abort on posted writes. A 1 indicates that P_SERR was signaled because of a target abort on a posted write.
2	R/C/U	Master posted write time-out. A 1 indicates that P_SERR was signaled because of a master time-out after 2^{24} retries on a posted write.
1	R/C/U	Posted write parity error. A 1 indicates that P_SERR was signaled because of parity error on a posted write.
0	R	Reserved. Bit 0 returns 0 when read.

5.21 PM Capability ID Register

The capability ID register identifies the linked list item as the register for PCI power management. The capability ID register returns 01h when read, which is the unique ID assigned by the PCI SIG for the PCI location of the capabilities pointer and the value.

Bit	7	6	5	4	3	2	1	0
Name	Capability ID							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	1

Register: **Capability ID**
Type: Read-only
Offset: DCh
Default: 01h

5.22 PM Next Item Pointer Register

The next item pointer register is used to indicate the next item in the linked list of PCI power management capabilities. The next item pointer returns E4h in compact PCI mode, indicating that the PCI2250 supports more than one extended capability, but in all other modes returns 00h, indicating that only one extended capability is supported.

Bit	7	6	5	4	3	2	1	0
Name	Next item pointer							
Type	R	R	R	R	R	R	R	R
Default	1	1	1	0	0	1	0	0

Register: **Next item pointer**
Type: Read-only
Offset: DDh
Default: E4h Compact PCI mode
00h All other modes

5.23 Power Management Capabilities Register

The power management capabilities register contains information on the capabilities of the PCI2250 functions related to power management. The PCI2250 function supports D0, D1, D2, and D3 power states when MS1 is low. The PCI2250 does not support any power states when MS1 is high. See Table 5–19 for a complete description of the register contents.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Power management capabilities															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0

Register: **Power management capabilities**
Type: Read-only
Offset: DEh
Default: 0602h or 0001h

Table 5–19. Power Management Capabilities Register

BIT	TYPE	FUNCTION
15–11	R	PME support. This five-bit field indicates the power states that the device supports asserting $\overline{\text{PME}}$. A 0 for any of these bits indicates that the PCI2250 cannot assert $\overline{\text{PME}}$ signal from that power state. For the PCI2250, these five bits return 00000b when read, indicating that $\overline{\text{PME}}$ is not supported.
10	R	D2 support. This bit returns 1 when MS0 is 0, indicating that the bridge function supports the D2 device power state. This bit returns 0 when MS0 is 1, indicating that the bridge function does not support the D2 device power state.
9	R	D1 support. This bit returns 1 when MS0 is 0, indicating that the bridge function supports the D1 device power state. This bit returns 0 when MS0 is 1, indicating that the bridge function does not support the D1 device power state.
8–6	R	Reserved. Bits 8–6 return 0s when read.
5	R	Device specific initialization. This bit returns 0 when read, indicating that the bridge function does not require special initialization (beyond the standard PCI configuration header) before the generic class device driver is able to use it.
4	R	Auxiliary power source. This bit returns a 0 because the PCI2250 does not support $\overline{\text{PME}}$ signaling.
3	R	PMECLK. This bit returns a 0 because the $\overline{\text{PME}}$ signaling is not supported.
2–0	R	Version. This three-bit register returns the *PCI Bus Power Management Interface Specification* revision. 001 = Revision 1.0, MS0 = 1 010 = Revision 1.1, MS0 = 0

5.24 Power Management Control/Status Register

The power management control/status register determines and changes the current power state of the PCI2250. The contents of this register are not affected by the internally generated reset caused by the transition from $D3_{hot}$ to D0 state. See Table 5–20 for a complete description of the register contents.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Power management control/status															
Type	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Default	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Register: **Power management control/status**
Type: Read-only, Read/Write
Offset: E0h
Default: 0000h

Table 5–20. Power Management Capabilities Register

BIT	TYPE	FUNCTION
15	R	$\overline{\text{PME}}$ status. This bit returns a 0 when read because the PCI2250 does not support $\overline{\text{PME}}$.
14–13	R	Data scale. This two-bit read-only field indicates the scaling factor to be used when interpreting the value of the data register. These bits return only 00b, because the data register is not implemented.
12–9	R	Data select. This four-bit field is used to select which data is to be reported through the data register and data-scale field. These bits return only 0000b, because the data register is not implemented.
8	R	$\overline{\text{PME}}$ enable. This bit returns a 0 when read because the PCI2250 does not support $\overline{\text{PME}}$ signaling.
7–2	R	Reserved. Bits 7–2 return 0s when read.
1–0	R/W	Power state. This two-bit field is used both to determine the current power state of a function and to set the function into a new power state. The definition of the two-bit field is given below: 00 – D0 01 – D1 10 – D2 11 – $D3_{hot}$

5.25 PMCSR Bridge Support Register

The PMCSR bridge support register is required for all PCI bridges and supports PCI bridge specific functionality. See Table 5–21 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	PMCSR bridge support							
Type	R	R	R	R	R	R	R	R
Default	X	X	0	0	0	0	0	0

Register: **PMCSR bridge support**
Type: Read-only
Offset: E2h
Default: X0h

Table 5–21. PMCSR Bridge Support Register

BIT	TYPE	FUNCTION
7	R	Bus power control enable. This bit returns the value of the MS1/BCC input. 0 = Bus power/ clock control disabled 1 = Bus power/clock control enabled
6	R	B2/B3 support for $D3_{hot}$. This bit returns the value of MS1/BCC input. When this bit is 1, the secondary clocks are stopped when the device is placed in $D3_{hot}$. When this bit is 0, the secondary clocks remain on in all device states. Note: If the primary clock is stopped, then the secondary clocks will stop because the primary clock is used to generate the secondary clocks.
5–0	R	Reserved. Bits 5–0 return 0s when read.

5.26 Data Register

The data register is an optional, 8-bit read–only register that provides a mechanism for the function to report state-dependent operating data such as power consumed or heat dissipatin. The PCI2050 does not implement the data register.

Bit	7	6	5	4	3	2	1	0
Name	Data							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

Register: **Data**
Type: Read-only
Offset: E3h
Default: 00h

5.27 HS Capability ID Register

The HS capability ID register identifies the linked list item as the register for CPCI hot swap capabilities. The register returns 06h when read, which is the unique ID assigned by the PICMG for PCI location of the capabilities pointer and the value.

Bit	7	6	5	4	3	2	1	0
Name	HS capability ID							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	1	1	0

Register: **HS capability ID**
Type: Read-only
Offset: E4h
Default: 06h

5.28 HS Next Item Pointer Register

The HS next item pointer register is used to indicate the next item in the linked list of CPCI hot swap capabilities. Since the PCI2250 functions only include two capabilities list item, this register returns 0s when read.

Bit	7	6	5	4	3	2	1	0
Name	HS next item pointer							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

Register: **HS next item pointer**
Type: Read-only
Offset: E5h
Default: 00h

5.29 Hot Swap Control Status Register

The hot swap control status register contains control and status information for CPCI hot swap resources. See Table 5–22 for a complete description of the register contents.

Bit	7	6	5	4	3	2	1	0
Name	Hot swap control status							
Type	R/C/U	R/C/U	R	R	R/W	R	R/W	R
Default	0	0	0	0	0	0	0	0

Register: **Hot swap control status**
Type: Read-only, Read/Write
Offset: E6h
Default: 00h

Table 5–22. Hot Swap Control Status Register

BIT	TYPE	FUNCTION
7	R/C/U	$\overline{\text{ENUM}}$ insertion status. When set, the $\overline{\text{ENUM}}$ output is driven by the PCI2250. This bit defaults to 0, and will be set after a PCI reset occurs, the ejector handle is closed, and bit 6 is 0. Thus, this bit is set following an insertion when the board implementing the PCI2250 is ready for configuration. This bit cannot be set under software control.
6	R/C/U	$\overline{\text{ENUM}}$ extraction status. When set, the $\overline{\text{ENUM}}$ output is driven by the PCI2250. This bit defaults to 0, and is set when the ejector handle is opened and bit 7 is 0. Thus, this bit is set when the board implementing the PCI2250 is about to be removed. This bit cannot be set under software control.
5–4	R	Reserved. Bits 5 and 4 return 0s when read.
3	R/W	LED ON/OFF. This bit defaults to 0, and controls the external LED indicator (HSLED) under normal conditions. However, for a duration following a $\overline{\text{PCI\_RST}}$, the HSLED output is driven high by the PCI2250 and this bit is ignored. When this bit is interpreted, a 1 will cause HSLED high and a 0 will cause HSLED low. Following $\overline{\text{PCI\_RST}}$, the HSLED output is driven high by the PCI2250 until the ejector handle is closed. When these conditions are met, the HSLED is under software control via this bit.
2	R	Reserved. Bit 2 returns 0 when read.
1	R/W	$\overline{\text{ENUM}}$ interrupt mask. This bit allows the $\overline{\text{HSENUM}}$ output to be masked by software. Bits 6 and 7 are set independently from this bit. 0 = Enable $\overline{\text{HSENUM}}$ output 1 = Mask $\overline{\text{HSENUM}}$ output
0	R	Reserved. Bit 0 returns 0 when read.

6 Electrical Characteristics

6.1 Absolute Maximum Ratings Over Operating Temperature Ranges †

Supply voltage range: V_{CC} . . . . . –0.5 V to 3.6 V
: SV_{CCP} . . . . . –0.5 V to 6 V
: PV_{CCP} . . . . . –0.5 V to 6 V
Input voltage range, V_I: PCI . . . . . –0.5 V to 6.5 V
: TTL . . . . . –0.5 V to V_{CC} + 0.5 V
Output voltage range, V_O: PCI . . . . . –0.5 V to V_{CC} + 0.5 V
Input clamp current, I_{IK} (V_I < 0 or V_I > V_{CC}) (see Note 1) . . . . . ±20 mA
Output clamp current, I_{OK} (V_O < 0 or V_O > V_{CC}) (see Note 2) . . . . . ±20 mA
Storage temperature range, T_{stg} . . . . . –65°C to 150°C
Virtual junction temperature, T_J . . . . . 150°C

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. Applies for external input and bidirectional buffers. $V_I > V_{CC}$ does not apply to fail-safe terminals.
2. Applies to external output and bidirectional buffers. $V_O > V_{CC}$ does not apply to fail-safe terminals.

6.2 Recommended Operating Conditions (see Note 3)

			OPERATION	MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage (core)	Commercial	3.3 V	3	3.3	3.6	V
PV_{CCP}	PCI primary bus I/O clamping rail voltage	Commercial	3.3 V	3	3.3	3.6	V
			5 V	4.75	5	5.25	
SV_{CCP}	PCI secondary bus I/O clamping rail voltage	Commercial	3.3 V	3	3.3	3.6	V
			5 V	4.75	5	5.25	
V_{IH}†	High-level input voltage	PCI	3.3 V	0.5 V_{CCP}		V_{CCP}	V
			5 V	2		V_{CCP}	
		TTL‡	3.3 V	2.25		V_{CC}	
V_{IL}†	High-level input voltage	PCI	3.3 V	0		0.3 V_{CCP}	V
			5 V	0		0.8	
		TTL‡		0		0.75	
V_I	Input voltage	PCI		0		V_{CCP}	V
		TTL‡		0		V_{CC}	
V_O§	Output voltage		3.3 V	0		V_{CC}	V
			5 V	0		V_{CC}	
t_t	Input transition time (t_r and t_f)	PCI		1		4	nS
		TTL‡		0		6	
T_A	Operating ambient temperature range		3.3 V	0	25	70	°C
T_J¶	Virtual junction temperature		5 V	0	25	115	

NOTES: 3. Unused or floating pins (input or I/O) must be held high or low.
† Applies for external input and bidirectional buffers without hysteresis
‡ TTL terminals are Schmitt-trigger input-only terminals: 55, 69, 132, 174 for PGF-packaged device; and 49, 63, 120, 159 for PCM-packaged device.
§ Applies for external output buffers
¶ These junction temperatures reflect simulation conditions. The customer is responsible for verifying junction temperature.

6.3 Recommended Operating Conditions for PCI Interface

			OPERATION	MIN	NOM	MAX	UNIT
V_{CC}	Core voltage	Commercial	3.3 V	3	3.3	3.6	V
V_{CCP}	PCI supply voltage	Commercial	3.3 V	3	3.3	3.6	V
			5 V	4.75	5	5.25	
V_I	Input voltage		3.3 V	0		V_{CCP}	V
			5 V	0		V_{CCP}	
V_O§	Output voltage		3.3 V	0		V_{CCP}	V
			5 V	0		V_{CCP}	
V_{IH}¶	High-level input voltage	CMOS compatible	3.3 V	0.5 V_{CCP}			V
			5 V	2			
V_{IL}¶	Low-level input voltage	CMOS compatible	3.3 V			0.3 V_{CCP}	V
			5 V			0.8	

§ Applies to external output buffers
¶ Applies to external input and bidirectional buffers without hysteresis

6.4 Electrical Characteristics Over Recommended Operating Conditions

PARAMETER		TERMINALS		OPERATION	TEST CONDITIONS	MIN	MAX	UNIT
V_{OH}†	High-level output voltage			3.3 V	I_{OH} = –0.5 mA	0.9 V_{CC}		V
				5 V	I_{OH} = –2 mA	2.4		
V_{OL}	Low-level output voltage			3.3 V	I_{OL} = 1.5 mA		0.1 V_{CC}	V
				5 V	I_{OL} = 6 mA		0.55	
I_{IH}‡	High-level input current	Input terminals	TTL§		$V_I = V_{CC}$		1	µA
			PCI		$V_I = V_{CCP}$		10	
		I/O terminals¶			$V_I = V_{CCP}$		10	
I_{IL}‡	Low-level input current	Input terminals	TTL§		V_I = GND		–1	µA
			PCI				–1	
		I/O terminals¶					–10	
I_{OZ}	High-impedance output current				$V_O = V_{CCP}$ or GND		±10	µA

† V_{OH} is not tested on $\overline{\text{PSERR}}$ due to open-drain configuration.
‡ I_{IH} and I_{IL} are not tested on NO_HSLED dur to its active ourput-only configuration.
§ TTL terminals are 55, 69, 132, 174 for PGF-packaged device; and 49, 63, 120, 159 for PCM-packaged device.
¶ For I/O terminals, the input leakage current includes the off-state output current I_{OZ}.

6.5 PCI Clock/Reset Timing Requirements Over Recommended Ranges of Supply Voltage and Operating Free-Air Temperature (see Figure 6–2 and Figure 6–3)

		ALTERNATE SYMBOL	MIN	MAX	UNIT
t_c	Cycle time, PCLK	t_{cyc}	30	∞	ns
t_{wH}	Pulse duration, PCLK high	t_{high}	11		ns
t_{wL}	Pulse duration, PCLK low	t_{low}	11		ns
$\Delta v/\Delta t$	Slew rate, PCLK	t_r, t_f	1	4	V/ns
t_w	Pulse duration, RSTIN	t_{rst}	1		ms
t_{su}	Setup time, PCLK active at end of $\overline{\text{RSTIN}}$ (see Note 4)	$t_{rst\text{-}clk}$	100		µs

NOTE 4: The setup and hold times for the secondary are identical to those for the primary; however, the times are relative to the secondary PCI close.

6.6 PCI Timing Requirements Over Recommended Ranges of Supply Voltage and Operating Free-Air Temperature (see Note 5 and Figure 6–1 and Figure 6–4)

			ALTERNATE SYMBOL	TEST CONDITIONS	MIN	MAX	UNIT
t_{pd}	Propagation delay time	PCLK to shared signal valid delay time	t_{val}	C_L = 50 pF, See Note 6		11	ns
		PCLK to shared signal invalid delay time	t_{inv}		2		
t_{en}	Enable time, high-impedance-to-active delay time from PCLK		t_{on}		2		ns
t_{dis}	Disable time, active-to-high-impedance delay time from PCLK		t_{off}			28	ns
t_{su}	Setup time before PCLK valid		t_{su}, See Note 4		7		ns
t_h	Hold time after PCLK high		t_h, See Note 4		0		ns

5. This data sheet uses the following conventions to describe time (t) intervals. The format is: t_A, where *subscript A* indicates the type of dynamic parameter being represented. One of the following is used: t_{pd} = propagation delay time, t_d = delay time, t_{su} = setup time, and t_h = hold time.
6. PCI shared signals are AD31–AD0, C/$\overline{\text{BE3}}$–C/$\overline{\text{BE0}}$, $\overline{\text{FRAME}}$, $\overline{\text{TRDY}}$, $\overline{\text{IRDY}}$, $\overline{\text{STOP}}$, IDSEL, $\overline{\text{DEVSEL}}$, and PAR.

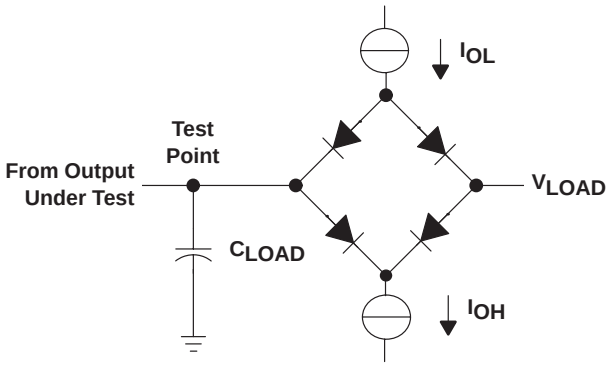
6.7 Parameter Measurement Information

LOAD CIRCUIT PARAMETERS

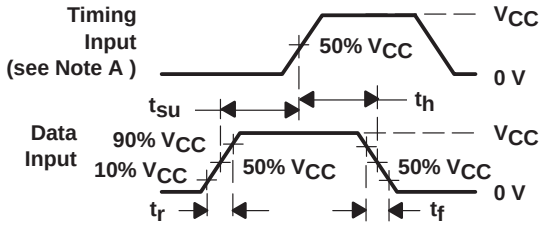
TIMING PARAMETER		C_{LOAD}† (pF)	I_{OL} (mA)	I_{OH} (mA)	V_{LOAD} (V)
t_{en}	t_{PZH}	50	8	–8	0
	t_{PZL}				3
t_{dis}	t_{PHZ}	50	8	–8	1.5
	t_{PLZ}				
t_{pd}		50	8	–8	‡

† C_{LOAD} includes the typical load-circuit distributed capacitance.

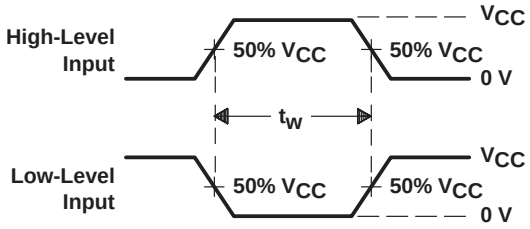
‡ $\frac{V_{LOAD} - V_{OL}}{I_{OL}}$ = 50 Ω, where V_{OL} = 0.6 V, I_{OL} = 8 mA



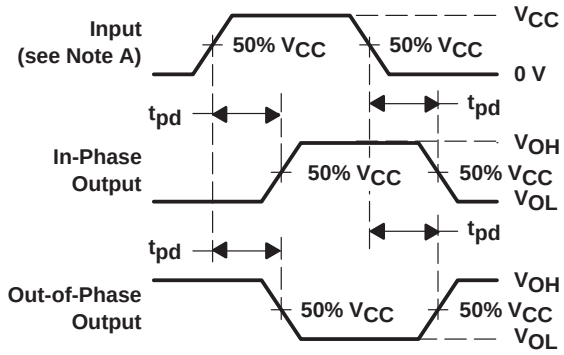
LOAD CIRCUIT



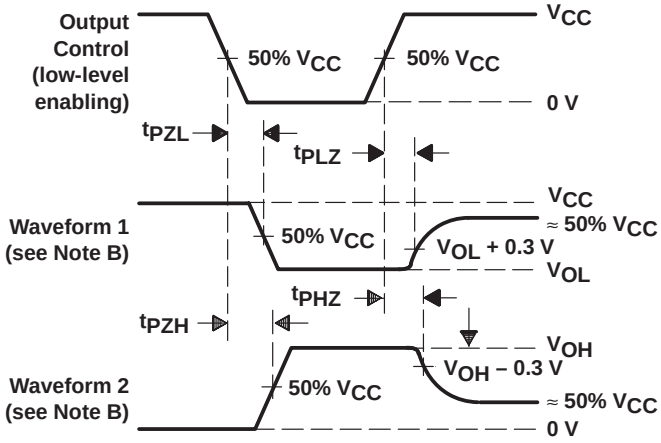
VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES
INPUT RISE AND FALL TIMES



VOLTAGE WAVEFORMS
PULSE DURATION



VOLTAGE WAVEFORMS
PROPAGATION DELAY TIMES



VOLTAGE WAVEFORMS
ENABLE AND DISABLE TIMES, 3-STATE OUTPUTS

NOTES: A. Phase relationships between waveforms were chosen arbitrarily. All input pulses are supplied by pulse generators having the following characteristics: PRR = 1 MHz, Z_O = 50 Ω, $t_r \le 6$ ns, $t_f \le 6$ ns.

B. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.

C. For t_{PLZ} and t_{PHZ}, V_{OL} and V_{OH} are measured values.

Figure 6–1. Load Circuit and Voltage Waveforms

6.8 PCI Bus Parameter Measurement Information

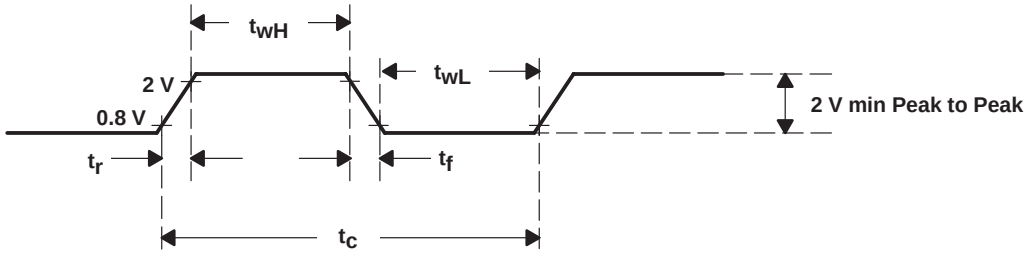


Figure 6–2. PCLK Timing Waveform

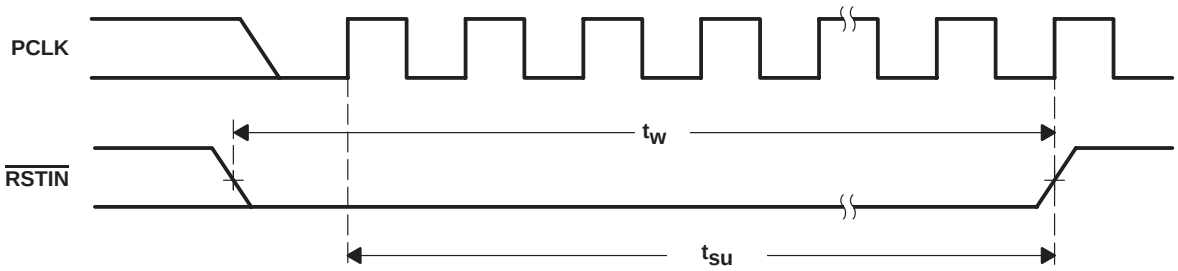


Figure 6–3. $\overline{\text{RSTIN}}$ Timing Waveforms

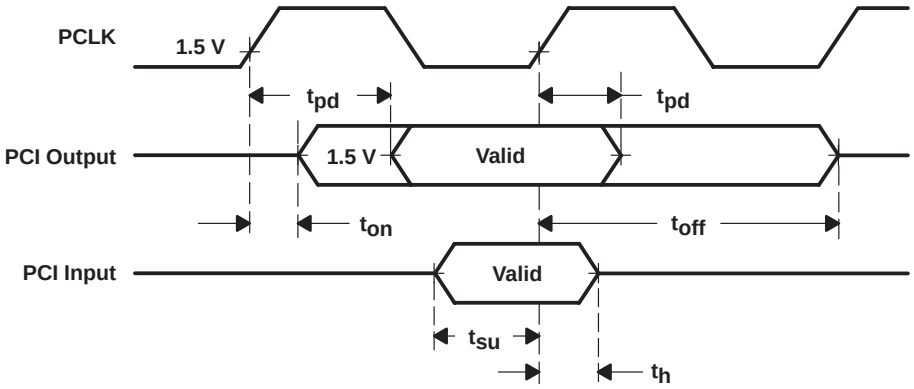
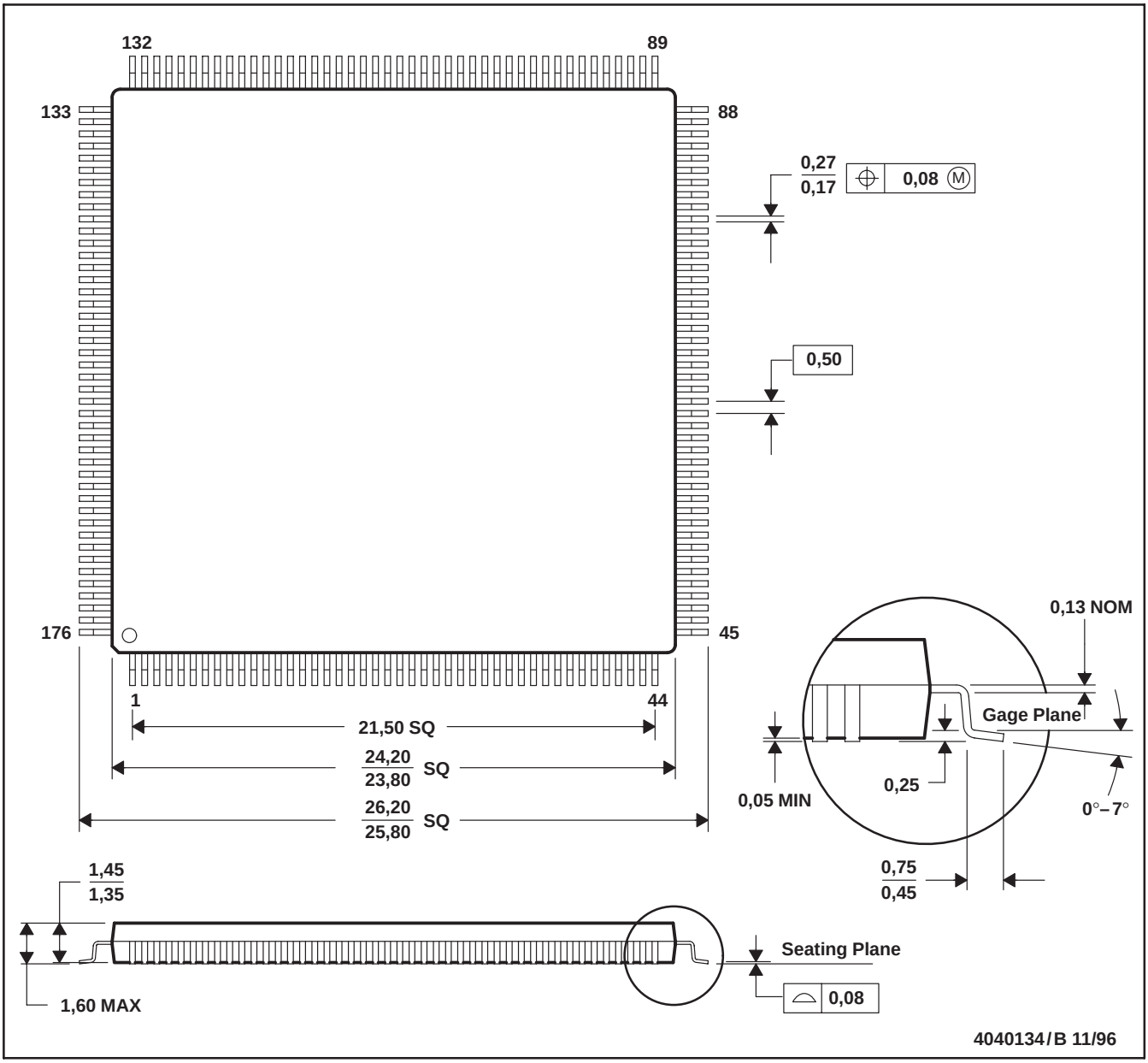


Figure 6–4. Shared-Signals Timing Waveforms

7 Mechanical Data

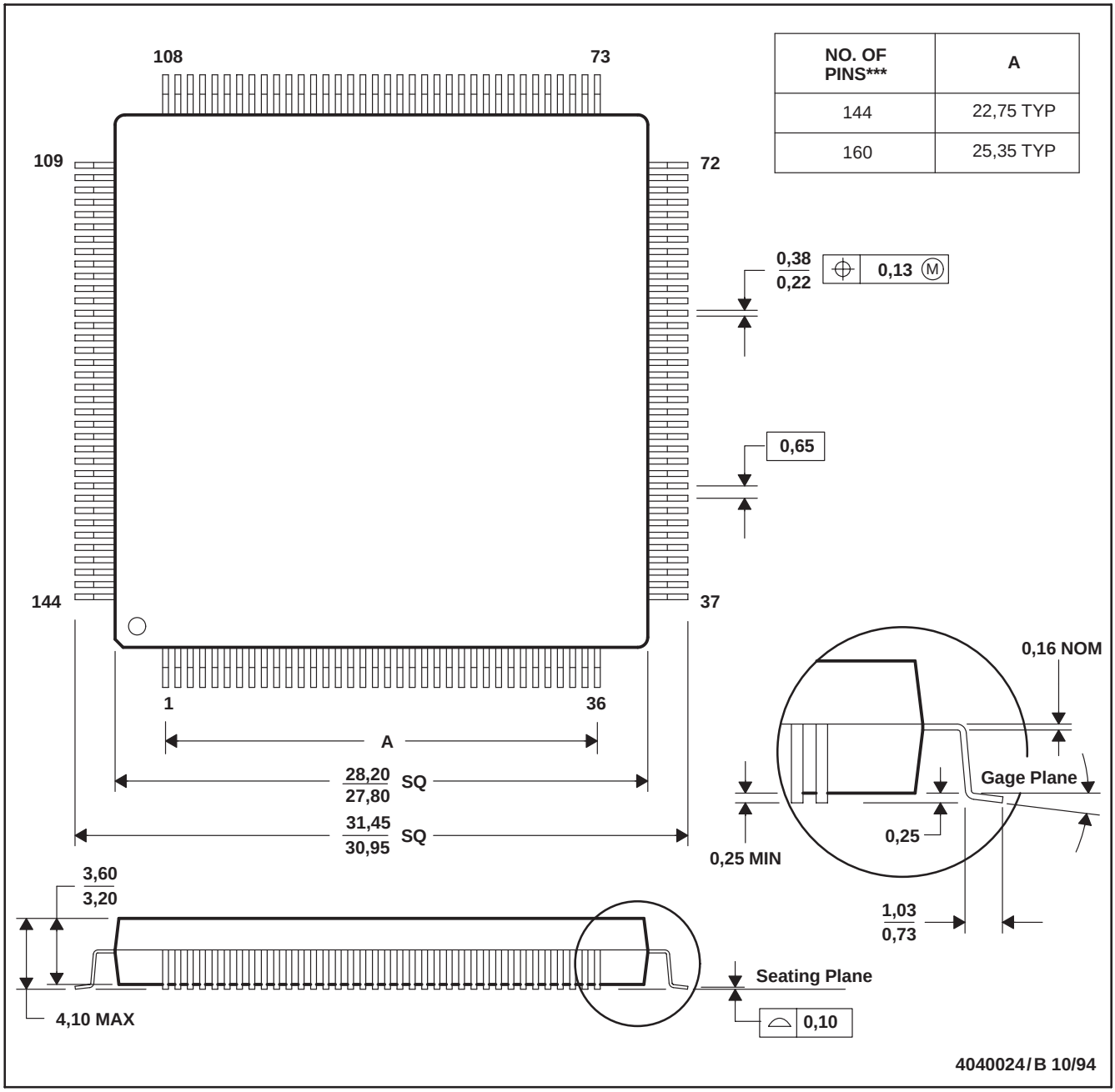
PGF (S-PQFP-G176)
PLASTIC QUAD FLATPACK



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-026

PCM (S-PQFP-G*)** **PLASTIC QUAD FLATPACK**

144 PINS SHOWN



NO. OF PINS***	A
144	22,75 TYP
160	25,35 TYP

NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-022
D. The 144 PCM is identical to the 160 PCM except that four leads per corner are removed.

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.