TEXAS INSTRUMENTS

PCI2050

PCI-to-PCI Bridge

Data Manual

1999

PCIBus Solutions

TEXAS INSTRUMENTS

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

SCPS053

PCI2050
PCI-to-PCI Bridge

Data Manual

Literature Number: SCPS053
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 PCI2050 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 PCI2050 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 PCI2050 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 PCI2050 provides two-tier internal arbitration for up to nine secondary bus masters and may be implemented with an external secondary PCI bus arbiter.

The compact-PCI hot-swap extended PCI capability is provided which makes the PCI2050 an ideal solution for multifunction compact PCI cards and adapting single function cards to hot-swap compliance.

The PCI2050 bridge is compliant with the *PCI-to-PCI Bridge Specification 1.1*. The PCI 2050 provides compliance for *PCI Power Management 1.0 and 1.1*. The PCI2050 has been designed to lead the industry in power conservation. An advanced CMOS process is used to achieve low system power consumption while operating at PCI clock rates up to 33 MHz.

1.2 Features

The PCI2050 supports the following features:

- Configurable for *PCI Bus Power Management Interface Specification*
- Provides compact PCI hot-swap functionality
- 3.3-V core logic with universal PCI interfaces 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 nine secondary bus masters and supports an external secondary bus arbiter
- Burst data transfers with pipeline architecture to maximize data throughput in both directions
- Independent read and write buffers for each direction
- Up to three delayed transactions in both directions
- Provides 10 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
- Packaged in 208-terminal QFP or 209-terminal MicroStar BGA

1.3 Related Documents

- *Advanced Configuration and Power Interface (ACPI) Specification (Revision 1.0)*
- *PCI Local Bus Specification (Revision 2.2)*
- *PCI-to-PCI Bridge Architecture Specification (Revision 1.1)*
- *PCI Bus 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
PCI2050	PCI–PCI Bridge	3.3 V, 5-V Tolerant I/Os	208-terminal QFP 209-terminal MicroStar BGA

2 Terminal Descriptions

PDV LOW-PROFILE QUAD FLAT PACKAGE
TOP VIEW

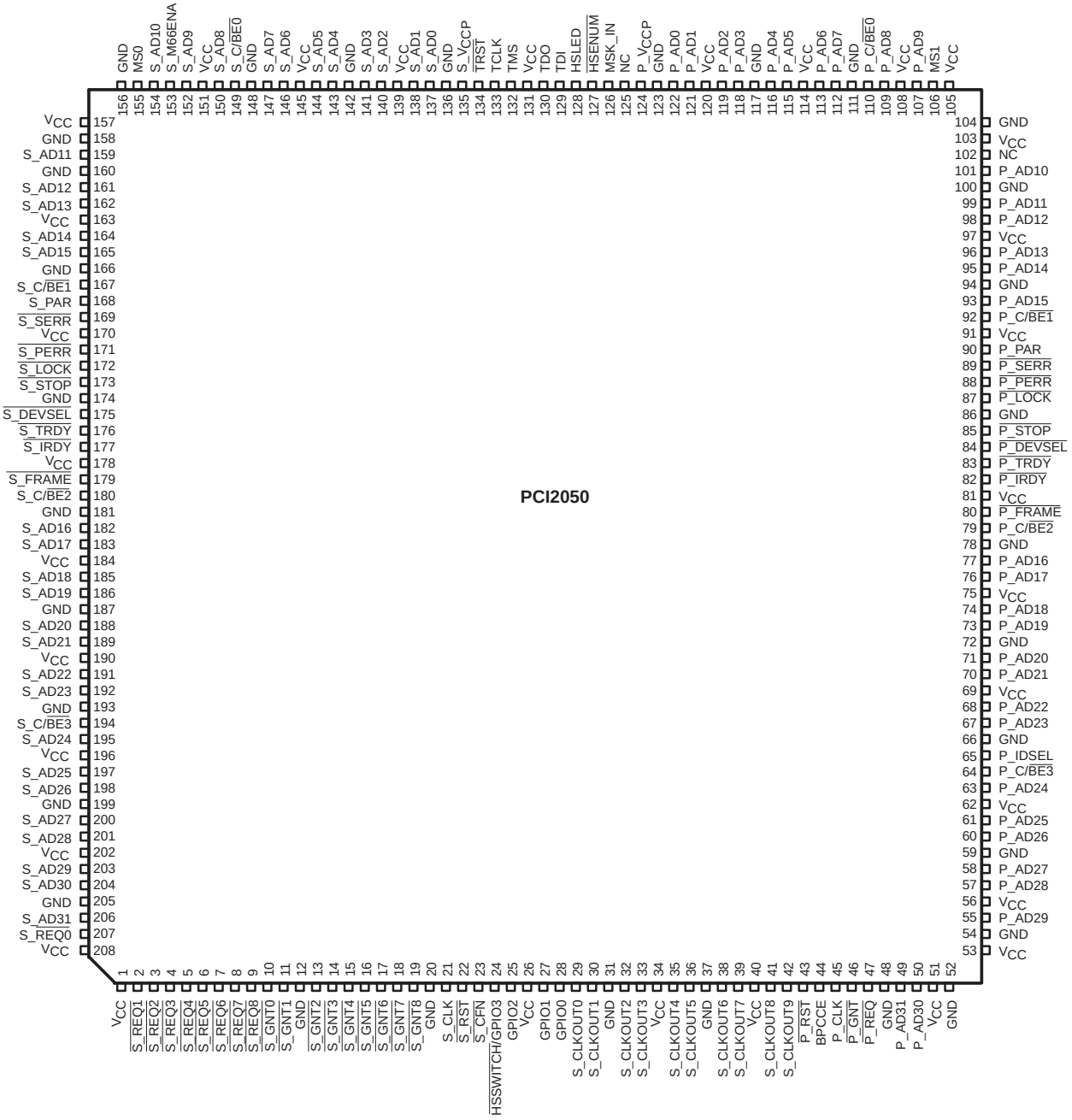


Figure 2–1. PCI2050 Terminal Diagram

Table 2–1. 208-Terminal PDV Signal Names Sorted by Terminal Number

PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME
1	D1	V_{CC}	44	P2	BPCCE	87	V12	$\overline{\text{P\_LOCK}}$	130	K17	TDO
2	E3	$\overline{\text{S\_REQ1}}$	45	N5	P_CLK	88	U12	$\overline{\text{P\_PERR}}$	131	K15	V_{CC}
3	F5	$\overline{\text{S\_REQ2}}$	46	P3	$\overline{\text{P\_GNT}}$	89	P12	$\overline{\text{P\_SERR}}$	132	K14	TMS
4	G6	$\overline{\text{S\_REQ3}}$	47	R1	$\overline{\text{P\_REQ}}$	90	R12	P_PAR	133	J19	TCLK
5	E2	$\overline{\text{S\_REQ4}}$	48	P6	GND	91	W13	V_{CC}	134	J18	$\overline{\text{TRST}}$
6	E1	$\overline{\text{S\_REQ5}}$	49	R2	P_AD31	92	V13	P_C/$\overline{\text{BE1}}$	135	J17	S_V_{CCP}
7	F3	$\overline{\text{S\_REQ6}}$	50	P5	P_AD30	93	U13	P_AD15	136	J14	GND
8	F2	$\overline{\text{S\_REQ7}}$	51	R3	V_{CC}	94	P13	GND	137	J15	S_AD0
9	G5	$\overline{\text{S\_REQ8}}$	52	T1	GND	95	W14	P_AD14	138	H19	S_AD1
10	F1	$\overline{\text{S\_GNT0}}$	53	W4	V_{CC}	96	V14	P_AD13	139	H18	V_{CC}
11	H6	$\overline{\text{S\_GNT1}}$	54	U5	GND	97	R13	V_{CC}	140	H17	S_AD2
12	G3	GND	55	R6	P_AD29	98	U14	P_AD12	141	H14	S_AD3
13	G2	$\overline{\text{S\_GNT2}}$	56	P7	V_{CC}	99	W15	P_AD11	142	H15	GND
14	G1	$\overline{\text{S\_GNT3}}$	57	V5	P_AD28	100	P14	GND	143	G19	S_AD4
15	H5	$\overline{\text{S\_GNT4}}$	58	W5	P_AD27	101	V15	P_AD10	144	G18	S_AD5
16	H3	$\overline{\text{S\_GNT5}}$	59	U6	GND	102	R14	NC	145	G17	V_{CC}
17	H2	$\overline{\text{S\_GNT6}}$	60	V6	P_AD26	103	U15	V_{CC}	146	G14	S_AD6
18	H1	$\overline{\text{S\_GNT7}}$	61	R7	P_AD25	104	W16	GND	147	F19	S_AD7
19	J1	$\overline{\text{S\_GNT8}}$	62	W6	V_{CC}	105	T19	V_{CC}	148	F18	GND
20	J2	GND	63	P8	P_AD24	106	R17	MS1	149	G15	S_C/$\overline{\text{BE0}}$
21	J3	S_CLK	64	U7	P_C/$\overline{\text{BE3}}$	107	P15	P_AD9	150	F17	S_AD8
22	J5	$\overline{\text{S\_RST}}$	65	V7	P_IDSEL	108	N14	V_{CC}	151	E19	V_{CC}
23	J6	$\overline{\text{S\_CFN}}$	66	W7	GND	109	R18	P_AD8	152	F14	S_AD9
24	K1	$\overline{\text{HSSWITCH}}$/GPIO3	67	R8	P_AD23	110	R19	P_C/$\overline{\text{BE0}}$	153	E18	S_M66ENA
25	K2	GPIO2	68	U8	P_AD22	111	P17	GND	154	F15	S_AD10
26	K3	V_{CC}	69	V8	V_{CC}	112	P18	P_AD7	155	E17	MS0
27	K5	GPIO1	70	W8	P_AD21	113	N15	P_AD6	156	D19	GND
28	K6	GPIO0	71	W9	P_AD20	114	P19	V_{CC}	157	A16	V_{CC}
29	L1	S_CLKOUT0	72	V9	GND	115	M14	P_AD5	158	C15	GND
30	L2	S_CLKOUT1	73	U9	P_AD19	116	N17	P_AD4	159	E14	S_AD11
31	L3	GND	74	R9	P_AD18	117	N18	GND	160	F13	GND
32	L6	S_CLKOUT2	75	P9	V_{CC}	118	N19	P_AD3	161	B15	S_AD12
33	L5	S_CLKOUT3	76	W10	P_AD17	119	M15	P_AD2	162	A15	S_AD13
34	M1	V_{CC}	77	V10	P_AD16	120	M17	V_{CC}	163	C14	V_{CC}
35	M2	S_CLKOUT4	78	U10	GND	121	M18	P_AD1	164	B14	S_AD14
36	M3	S_CLKOUT5	79	R10	P_C/$\overline{\text{BE2}}$	122	M19	P_AD0	165	E13	S_AD15
37	M6	GND	80	P10	$\overline{\text{P\_FRAME}}$	123	L19	GND	166	A14	GND
38	M5	S_CLKOUT6	81	W11	V_{CC}	124	L18	P_V_{CCP}	167	F12	S_C/$\overline{\text{BE1}}$
39	N1	S_CLKOUT7	82	V11	$\overline{\text{P\_IRDY}}$	125	L17	NC	168	C13	S_PAR
40	N2	V_{CC}	83	U11	$\overline{\text{P\_TRDY}}$	126	L15	MSK_IN	169	B13	$\overline{\text{S\_SERR}}$
41	N3	S_CLKOUT8	84	P11	$\overline{\text{P\_DEVSEL}}$	127	L14	$\overline{\text{HSENUM}}$	170	A13	V_{CC}
42	N6	S_CLKOUT9	85	R11	$\overline{\text{P\_STOP}}$	128	K19	HSLED	171	E12	$\overline{\text{S\_PERR}}$
43	P1	$\overline{\text{P\_RST}}$	86	W12	GND	129	K18	TDI	172	C12	$\overline{\text{S\_LOCK}}$

Table 2–1. 208-Terminal PDV Signal Names Sorted by Terminal Number (continued)

PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME
173	B12	$\overline{\text{S\_STOP}}$	182	C10	S_AD16	191	B8	S_AD22	200	B6	S_AD27
174	A12	GND	183	E10	S_AD17	192	C8	S_AD23	201	E7	S_AD28
175	A11	$\overline{\text{S\_DEVSEL}}$	184	F10	V_{CC}	193	F8	GND	202	C6	V_{CC}
176	B11	$\overline{\text{S\_TRDY}}$	185	A9	S_AD18	194	E8	S_C/$\overline{\text{BE3}}$	203	A5	S_AD29
177	C11	$\overline{\text{S\_IRDY}}$	186	B9	S_AD19	195	A7	S_AD24	204	F6	S_AD30
178	E11	V_{CC}	187	C9	GND	196	B7	V_{CC}	205	B5	GND
179	F11	$\overline{\text{S\_FRAME}}$	188	F9	S_AD20	197	C7	S_AD25	206	E6	S_AD31
180	A10	S_C/$\overline{\text{BE2}}$	189	E9	S_AD21	198	F7	S_AD26	207	C5	$\overline{\text{S\_REQ0}}$
181	B10	GND	190	A8	V_{CC}	199	A6	GND	208	A4	V_{CC}

Table 2–2. Signal Names Sorted Alphabetically

SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.
BPCCE	44	P2	P_AD0	122	M19	P_PAR	90	R12	S_C/$\overline{BE3}$	194	E8
GND	12	G3	P_AD1	121	M18	$\overline{P\_PERR}$	88	U12	$\overline{S\_CFN}$	23	J6
GND	20	J2	P_AD2	119	M15	$\overline{P\_REQ}$	47	R1	S_CLK	21	J3
GND	31	L3	P_AD3	118	N19	$\overline{P\_RST}$	43	P1	S_CLKOUT0	29	L1
GND	37	M6	P_AD4	116	N17	$\overline{P\_SERR}$	89	P12	S_CLKOUT1	30	L2
GND	48	P6	P_AD5	115	M14	$\overline{P\_STOP}$	85	R11	S_CLKOUT2	32	L6
GND	52	T1	P_AD6	113	N15	$\overline{P\_TRDY}$	83	U11	S_CLKOUT3	33	L5
GND	54	U5	P_AD7	112	P18	P_V_{CCP}	124	L18	S_CLKOUT4	35	M2
GND	59	U6	P_AD8	109	R18	S_AD0	137	J15	S_CLKOUT5	36	M3
GND	66	W7	P_AD9	107	P15	S_AD1	138	H19	S_CLKOUT6	38	M5
GND	72	V9	P_AD10	101	V15	S_AD2	140	H17	S_CLKOUT7	39	N1
GND	78	U10	P_AD11	99	W15	S_AD3	141	H14	S_CLKOUT8	41	N3
GND	86	W12	P_AD12	98	U14	S_AD4	143	G19	S_CLKOUT9	42	N6
GND	94	P13	P_AD13	96	V14	S_AD5	144	G18	$\overline{S\_DEVSEL}$	175	A11
GND	100	P14	P_AD14	95	W14	S_AD6	146	G14	$\overline{S\_FRAME}$	179	F11
GND	104	W16	P_AD15	93	U13	S_AD7	147	F19	$\overline{S\_GNT0}$	10	F1
GND	111	P17	P_AD16	77	V10	S_AD8	150	F17	$\overline{S\_GNT1}$	11	H6
GND	117	N18	P_AD17	76	W10	S_AD9	152	F14	$\overline{S\_GNT2}$	13	G2
GND	123	L19	P_AD18	74	R9	S_AD10	154	F15	$\overline{S\_GNT3}$	14	G1
GND	136	J14	P_AD19	73	U9	S_AD11	159	E14	$\overline{S\_GNT4}$	15	H5
GND	142	H15	P_AD20	71	W9	S_AD12	161	B15	$\overline{S\_GNT5}$	16	H3
GND	148	F18	P_AD21	70	W8	S_AD13	162	A15	$\overline{S\_GNT6}$	17	H2
GND	156	D19	P_AD22	68	U8	S_AD14	164	B14	$\overline{S\_GNT7}$	18	H1
GND	158	C15	P_AD23	67	R8	S_AD15	165	E13	$\overline{S\_GNT8}$	19	J1
GND	160	F13	P_AD24	63	P8	S_AD16	182	C10	$\overline{S\_IRDY}$	177	C11
GND	166	A14	P_AD25	61	R7	S_AD17	183	E10	$\overline{S\_LOCK}$	172	C12
GND	174	A12	P_AD26	60	V6	S_AD18	185	A9	S_M66ENA	153	E18
GND	181	B10	P_AD27	58	W5	S_AD19	186	B9	S_PAR	168	C13
GND	187	C9	P_AD28	57	V5	S_AD20	188	F9	$\overline{S\_PERR}$	171	E12
GND	193	F8	P_AD29	55	R6	S_AD21	189	E9	$\overline{S\_REQ0}$	207	C5
GND	199	A6	P_AD30	50	P5	S_AD22	191	B8	$\overline{S\_REQ1}$	2	E3
GND	205	B5	P_AD31	49	R2	S_AD23	192	C8	$\overline{S\_REQ2}$	3	F5
GPIO0	28	K6	P_C/$\overline{BE0}$	110	R19	S_AD24	195	A7	$\overline{S\_REQ3}$	4	G6
GPIO1	27	K5	P_C/$\overline{BE1}$	92	V13	S_AD25	197	C7	$\overline{S\_REQ4}$	5	E2
GPIO2	25	K2	P_C/$\overline{BE2}$	79	R10	S_AD26	198	F7	$\overline{S\_REQ5}$	6	E1
$\overline{HSENUM}$	127	L14	P_C/$\overline{BE3}$	64	U7	S_AD27	200	B6	$\overline{S\_REQ6}$	7	F3
HSLED	128	K19	P_CLK	45	N5	S_AD28	201	E7	$\overline{S\_REQ7}$	8	F2
$\overline{HSSWITCH}$/GPIO3	24	K1	$\overline{P\_DEVSEL}$	84	P11	S_AD29	203	A5	$\overline{S\_REQ8}$	9	G5
MS0	155	E17	$\overline{P\_FRAME}$	80	P10	S_AD30	204	F6	$\overline{S\_RST}$	22	J5
MS1	106	R17	$\overline{P\_GNT}$	46	P3	S_AD31	206	E6	$\overline{S\_SERR}$	169	B13
MSK_IN	126	L15	P_IDSEL	65	V7	S_C/$\overline{BE0}$	149	G15	$\overline{S\_STOP}$	173	B12
NC	102	R14	$\overline{P\_IRDY}$	82	V11	S_C/$\overline{BE1}$	167	F12	$\overline{S\_TRDY}$	176	B11
NC	125	L17	$\overline{P\_LOCK}$	87	V12	S_C/$\overline{BE2}$	180	A10	S_V_{CCP}	135	J17

Table 2–2. Signal Names Sorted Alphabetically (continued)

SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.	SIGNAL NAME	PDV NO.	GHK NO.
TCLK	133	J19	V_{CC}	51	R3	V_{CC}	103	U15	V_{CC}	157	A16
TDI	129	K18	V_{CC}	53	W4	V_{CC}	105	T19	V_{CC}	163	C14
TDO	130	K17	V_{CC}	56	P7	V_{CC}	108	N14	V_{CC}	170	A13
TMS	132	K14	V_{CC}	62	W6	V_{CC}	114	P19	V_{CC}	178	E11
$\overline{TRST}$	134	J18	V_{CC}	69	V8	V_{CC}	120	M17	V_{CC}	184	F10
V_{CC}	1	D1	V_{CC}	75	P9	V_{CC}	131	K15	V_{CC}	190	A8
V_{CC}	26	K3	V_{CC}	81	W11	V_{CC}	139	H18	V_{CC}	196	B7
V_{CC}	34	M1	V_{CC}	91	W13	V_{CC}	145	G17	V_{CC}	202	C6
V_{CC}	40	N2	V_{CC}	97	R13	V_{CC}	151	E19	V_{CC}	208	A4

Table 2–3. 209-Terminal GHK Signal Names Sorted by Terminal Number

GHK NO.	SIGNAL NAME	GHK NO.	SIGNAL NAME	GHK NO.	SIGNAL NAME	GHK NO.	SIGNAL NAME	GHK NO.	SIGNAL NAME
A4	V_{CC}	E9	S_AD21	H17	S_AD2	N1	S_CLKOUT7	T19	V_{CC}
A5	S_AD29	E10	S_AD17	H18	V_{CC}	N2	V_{CC}	U5	GND
A6	GND	E11	V_{CC}	H19	S_AD1	N3	S_CLKOUT8	U6	GND
A7	S_AD24	E12	$\overline{\text{S\_PERR}}$	J1	$\overline{\text{S\_GNT8}}$	N5	P_CLK	U7	P_C/$\overline{\text{BE3}}$
A8	V_{CC}	E13	S_AD15	J2	GND	N6	S_CLKOUT9	U8	P_AD22
A9	S_AD18	E14	S_AD11	J3	S_CLK	N14	V_{CC}	U9	P_AD19
A10	S_C/$\overline{\text{BE2}}$	E17	MS0	J5	$\overline{\text{S\_RST}}$	N15	P_AD6	U10	GND
A11	$\overline{\text{S\_DEVSEL}}$	E18	S_M66ENA	J6	$\overline{\text{S\_CFN}}$	N17	P_AD4	U11	$\overline{\text{P\_TRDY}}$
A12	GND	E19	V_{CC}	J14	GND	N18	GND	U12	$\overline{\text{P\_PERR}}$
A13	V_{CC}	F1	$\overline{\text{S\_GNT0}}$	J15	S_AD0	N19	P_AD3	U13	P_AD15
A14	GND	F2	$\overline{\text{S\_REQ7}}$	J17	S_V_{CCP}	P1	$\overline{\text{P\_RST}}$	U14	P_AD12
A15	S_AD13	F3	$\overline{\text{S\_REQ6}}$	J18	$\overline{\text{TRST}}$	P2	BPCCE	U15	V_{CC}
A16	V_{CC}	F5	$\overline{\text{S\_REQ2}}$	J19	TCLK	P3	$\overline{\text{P\_GNT}}$	V5	P_AD28
B5	GND	F6	S_AD30	K1	$\overline{\text{HSSWITCH}}$/GPIO3	P5	P_AD30	V6	P_AD26
B6	S_AD27	F7	S_AD26	K2	GPIO2	P6	GND	V7	P_IDSEL
B7	V_{CC}	F8	GND	K3	V_{CC}	P7	V_{CC}	V8	V_{CC}
B8	S_AD22	F9	S_AD20	K5	GPIO1	P8	P_AD24	V9	GND
B9	S_AD19	F10	V_{CC}	K6	GPIO0	P9	V_{CC}	V10	P_AD16
B10	GND	F11	$\overline{\text{S\_FRAME}}$	K14	TMS	P10	$\overline{\text{P\_FRAME}}$	V11	$\overline{\text{P\_IRDY}}$
B11	$\overline{\text{S\_TRDY}}$	F12	S_C/$\overline{\text{BE1}}$	K15	V_{CC}	P11	$\overline{\text{P\_DEVSEL}}$	V12	$\overline{\text{P\_LOCK}}$
B12	$\overline{\text{S\_STOP}}$	F13	GND	K17	TDO	P12	$\overline{\text{P\_SERR}}$	V13	P_C/$\overline{\text{BE1}}$
B13	$\overline{\text{S\_SERR}}$	F14	S_AD9	K18	TDI	P13	GND	V14	P_AD13
B14	S_AD14	F15	S_AD10	K19	HSLED	P14	GND	V15	P_AD10
B15	S_AD12	F17	S_AD8	L1	S_CLKOUT0	P15	P_AD9	W4	V_{CC}
C5	$\overline{\text{S\_REQ0}}$	F18	GND	L2	S_CLKOUT1	P17	GND	W5	P_AD27
C6	V_{CC}	F19	S_AD7	L3	GND	P18	P_AD7	W6	V_{CC}
C7	S_AD25	G1	$\overline{\text{S\_GNT3}}$	L5	S_CLKOUT3	P19	V_{CC}	W7	GND
C8	S_AD23	G2	$\overline{\text{S\_GNT2}}$	L6	S_CLKOUT2	R1	$\overline{\text{P\_REQ}}$	W8	P_AD21
C9	GND	G3	GND	L14	$\overline{\text{HSENUM}}$	R2	P_AD31	W9	P_AD20
C10	S_AD16	G5	$\overline{\text{S\_REQ8}}$	L15	MSK_IN	R3	V_{CC}	W10	P_AD17
C11	$\overline{\text{S\_IRDY}}$	G6	$\overline{\text{S\_REQ3}}$	L17	NC	R6	P_AD29	W11	V_{CC}
C12	$\overline{\text{S\_LOCK}}$	G14	S_AD6	L18	P_V_{CCP}	R7	P_AD25	W12	GND
C13	S_PAR	G15	S_C/$\overline{\text{BE0}}$	L19	GND	R8	P_AD23	W13	V_{CC}
C14	V_{CC}	G17	V_{CC}	M1	V_{CC}	R9	P_AD18	W14	P_AD14
C15	GND	G18	S_AD5	M2	S_CLKOUT4	R10	V_{CC}	W15	P_AD11
D1	V_{CC}	G19	S_AD4	M3	S_CLKOUT5	R11	$\overline{\text{P\_STOP}}$	W16	GND
D19	GND	H1	$\overline{\text{S\_GNT7}}$	M5	S_CLKOUT6	R12	P_PAR		
E1	$\overline{\text{S\_REQ5}}$	H2	$\overline{\text{S\_GNT6}}$	M6	GND	R13	V_{CC}		
E2	$\overline{\text{S\_REQ4}}$	H3	$\overline{\text{S\_GNT5}}$	M14	P_AD5	R14	NC		
E3	$\overline{\text{S\_REQ1}}$	H5	$\overline{\text{S\_GNT4}}$	M15	P_AD2	R17	MS1		
E6	S_AD31	H6	$\overline{\text{S\_GNT1}}$	M17	V_{CC}	R18	P_AD8		
E7	S_AD28	H14	S_AD3	M18	P_AD1	R19	P_C/$\overline{\text{BE0}}$		
E8	S_C/$\overline{\text{BE3}}$	H15	GND	M19	P_AD0	T1	GND		

Table 2–4. Primary PCI System

TERMINAL			I/O	DESCRIPTION
NAME	PDV NO.	GHK NO.		
P_CLK	45	N5	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.
$\overline{\text{P\_RST}}$	43	P1	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. After $\overline{\text{P\_RST}}$ is deasserted, the bridge is in its default state.

Table 2–5. Primary PCI Address and Data

TERMINAL			I/O	DESCRIPTION
NAME	PDV NO.	GHK NO.		
P_AD31	49	R2	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	50	P5		
P_AD29	55	R6		
P_AD28	57	V5		
P_AD27	58	W5		
P_AD26	60	V6		
P_AD25	61	R7		
P_AD24	63	P8		
P_AD23	67	R8		
P_AD22	68	U8		
P_AD21	70	W8		
P_AD20	71	W9		
P_AD19	73	U9		
P_AD18	74	R9		
P_AD17	76	W10		
P_AD16	77	V10		
P_AD15	93	U13		
P_AD14	95	W14		
P_AD13	96	V14		
P_AD12	98	U14		
P_AD11	99	W15		
P_AD10	101	V15		
P_AD9	107	P15		
P_AD8	109	R18		
P_AD7	112	P18		
P_AD6	113	N15		
P_AD5	115	M14		
P_AD4	116	N17		
P_AD3	118	N19		
P_AD2	119	M15		
P_AD1	121	M18		
P_AD0	122	M19		
P_C/$\overline{\text{BE3}}$	64	U7	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 PCI 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/$\overline{\text{BE2}}$	79	R10		
P_C/$\overline{\text{BE1}}$	92	V13		
P_C/$\overline{\text{BE0}}$	110	R19		

Table 2–6. Primary PCI Interface Control

TERMINAL			I/O	DESCRIPTION
NAME	PDV NO.	GHK NO.		
P_DEVSEL	84	P11	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 time-out occurs, then the bridge terminates the cycle with an initiator abort.
P_FRAME	80	P10	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 transfers continue while this signal is asserted. When P_FRAME is deasserted, the primary bus transaction is in the final data phase.
P_GNT	46	P3	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	65	V7	I	Primary initialization device select. P_IDSEL selects the bridge during configuration space accesses. P_IDSEL can be connected to one of the upper 24 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	82	V11	I/O	Primary initiator ready. P_IRDY indicates 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_LOCK	87	V12	I/O	Primary PCI bus lock. P_LOCK is used to lock the primary bus and gain exclusive access as an initiator.
P_PAR	90	R12	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 parity indicator of the initiator; a miscompare can result in a parity error assertion (P_PERR).
P_PERR	88	U12	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.
P_REQ	47	R1	O	Primary PCI bus request. Asserted by the bridge to request access to the primary PCI bus as an initiator.
P_SERR	89	P12	O	Primary system error. Output pulsed from the bridge when enabled through the command register indicating a system error has occurred. The bridge needs not be the target of the primary PCI cycle to assert this signal. When bit 6 is enabled in the bridge control register (offset 3Eh, see Section 4.32), this signal also pulses indicating that a system error has occurred on one of the subordinate buses downstream from the bridge.
P_STOP	85	R11	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	83	U11	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 upon a rising edge of P_CLK where both P_IRDY and P_TRDY are asserted. Until both P_IRDY and P_TRDY are asserted, wait states are inserted.

Table 2–7. Secondary PCI System

TERMINAL			I/O	DESCRIPTION
NAME	PDV NO.	GHK NO.		
S_CLKOUT9 S_CLKOUT8 S_CLKOUT7 S_CLKOUT6 S_CLKOUT5 S_CLKOUT4 S_CLKOUT3 S_CLKOUT2 S_CLKOUT1 S_CLKOUT0	42 41 39 38 36 35 33 32 30 29	N6 N3 N1 M5 M3 M2 L5 L6 L2 L1	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	21	J3	I	Secondary PCI bus clock input. This input syncronizes the PCI2050 to the secondary bus clocks.
$\overline{\text{S\_CFN}}$	23	J6	I	Secondary external arbiter enable. When this signal is high, the secondary external arbiter is enabled. When the external arbiter is enabled, the PCI2050 $\overline{\text{S\_REQ0}}$ 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.
$\overline{\text{S\_RST}}$	22	J5	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 (bit 6) 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 2–8. Secondary PCI Address and Data

TERMINAL			I/O	DESCRIPTION
NAME	PDV NO.	GHK NO.		
S_AD31	206	E6	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	204	F6		
S_AD29	203	A5		
S_AD28	201	E7		
S_AD27	200	B6		
S_AD26	198	F7		
S_AD25	197	C7		
S_AD24	195	A7		
S_AD23	192	C8		
S_AD22	191	B8		
S_AD21	189	E9		
S_AD20	188	F9		
S_AD19	186	B9		
S_AD18	185	A9		
S_AD17	183	E10		
S_AD16	182	C10		
S_AD15	165	E13		
S_AD14	164	B14		
S_AD13	162	A15		
S_AD12	161	B15		
S_AD11	159	E14		
S_AD10	154	F15		
S_AD9	152	F14		
S_AD8	150	F17		
S_AD7	147	F19		
S_AD6	146	G14		
S_AD5	144	G18		
S_AD4	143	G19		
S_AD3	141	H14		
S_AD2	140	H17		
S_AD1	138	H19		
S_AD0	137	J15		
S_C/$\overline{BE3}$	194	E8	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 PCI cycle, S_C/BE3–S_C/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/$\overline{BE2}$	180	A10		
S_C/$\overline{BE1}$	167	F12		
S_C/$\overline{BE0}$	149	G15		
$\overline{S\_DEVSEL}$	175	A11	I/O	Secondary device select. The bridge asserts $\overline{S\_DEVSEL}$ to claim a PCI cycle as the target device. As a PCI initiator on the secondary bus, the bridge monitors $\overline{S\_DEVSEL}$ until a target responds. If no target responds before timeout occurs, then the bridge terminates the cycle with an initiator abort.
$\overline{S\_FRAME}$	179	F11	I/O	Secondary cycle frame. $\overline{S\_FRAME}$ is driven by the initiator of a secondary bus cycle. $\overline{S\_FRAME}$ is asserted to indicate that a bus transaction is beginning and data transfers continue while $\overline{S\_FRAME}$ is asserted. When $\overline{S\_FRAME}$ is deasserted, the secondary bus transaction is in the final data phase.
$\overline{S\_GNT8}$	19	J1	O	Secondary bus grant to the bridge. The bridge provides internal arbitration and these signals are used to grant potential secondary PCI bus masters access to the bus. Ten potential initiators (including the bridge) can be located on the secondary PCI bus. When the internal arbiter is disabled, $\overline{S\_GNT0}$ is reconfigured as an external secondary bus request signal for the bridge.
$\overline{S\_GNT7}$	18	H1		
$\overline{S\_GNT6}$	17	H2		
$\overline{S\_GNT5}$	16	H3		
$\overline{S\_GNT4}$	15	H5		
$\overline{S\_GNT3}$	14	G1		
$\overline{S\_GNT2}$	13	G2		
$\overline{S\_GNT1}$	11	H6		
$\overline{S\_GNT0}$	10	F1		

Table 2–9. Secondary PCI Interface Control

TERMINAL			I/O	DESCRIPTION
NAME	PDV NO.	GHK NO.		
S_IRDY	177	C11	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_PCLKn where both S_IRDY and S_TRDY are asserted; until S_IRDY and S_TRDY are asserted, wait states are inserted.
S_LOCK	172	C12	I/O	Secondary PCI bus lock. S_LOCK is used to lock the secondary bus and gain exclusive access as an initiator.
S_PAR	168	C13	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	171	E12	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 enabled through the command register.
S_REQ8 S_REQ7 S_REQ6 S_REQ5 S_REQ4 S_REQ3 S_REQ2 S_REQ1 S_REQ0	9 8 7 6 5 4 3 2 207	G5 F2 F3 E1 E2 G6 F5 E3 C5	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. Ten 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	169	B13	I	Secondary system error. S_SERR is passed through the primary interface by the bridge if enabled through the bridge control register. S_SERR is never asserted by the bridge.
S_STOP	173	B12	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	176	B11	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 asserted, wait states are inserted.

Table 2–10. Miscellaneous Terminals

TERMINAL			I/O	DESCRIPTION
NAME	PDV NO.	GHK NO.		
BPCCE	44	P2	I	Bus/power clock control management terminal. When signal BPCCE is tied high, and when the PCI2050 is placed in the D3 power state, it enables the PCI2050 to place the secondary bus in the B2 power state. The PCI2050 disables the secondary clocks and drives them to 0. When tied low, placing the PCI2050 in the D3 power state has no effect on the secondary bus clocks.
GPIO3/HSSWITCH GPIO2 GPIO1 GPIO0	24 25 27 28	K1 K2 K5 K6	I	General-purpose I/O pins GPIO3 is HSSWITCH in CPCI mode. HSSWITCH provides the status of the ejector handle switch to the CPCI logic.
HSENUM	127	L14	O	Hot swap ENUM
HSLED	128	K19	O	Hot swap LED output
MS0	155	E17	I	Mode select 0
MS1	106	R17	I	Mode select 1
NC	102 125	R14 L17	NC	These terminals have no function on the PCI2050.
S_M66ENA	153	E18	O	Secondary bus 66-MHz enable pin. This pin is always driven low to incicate that the secondary bus speed is 33 MHz.

Table 2–11. JTAG Interface Terminals

TERMINAL			I/O	DESCRIPTION
NAME	PDV NO.	GHK NO.		
TCLK	133	J19	I	JTAG boundary-scan clock. TCLK is the clock controlling the JTAG logic.
TDI	129	K18	I	JTAG serial data in. TDI is the serial input through which JTAG instructions and test data enter the JTAG interface. The new data on TDI is sampled on the rising edge of TCLK.
TDO	130	K17	O	JTAG serial data out. TDO is the serial output through which test instructions and data from the test logic leave the PCI2050.
TMS	132	K14	I	JTAG test mode select. TMS causes state transitions in the test access port controller.
$\overline{\text{TRST}}$	134	J18	I	JTAG TAP reset. When $\overline{\text{TRST}}$ is asserted low, the TAP controller is asynchronously forced to enter a reset state and initialize the test logic.

Table 2–12. Power Supply

TERMINAL			DESCRIPTION
NAME	PDV NO.	GHK NO.	
GND	12, 20, 31, 37, 48, 52, 54, 59, 66, 72, 78, 86, 94, 100, 104, 111, 117, 123, 136, 142, 148, 156, 158, 160, 166, 174, 181, 187, 193, 199, 205	A6, A12, A14, B5, B10, C9, C15, D19, F8, F13, F18, G3, H15, J2, J14, L3, L19, M6, N18, P6, P13, P14, P17, T1, U5, U6, U10, V9, W7, W12, W16	Device ground terminals
V_{CC}	1, 26, 34, 40, 51, 53, 56, 62, 69, 75, 81, 91, 97, 103, 105, 108, 114, 120, 131, 139, 145, 151, 157, 163, 170, 178, 184, 190, 196, 202, 208	A4, A8, A13, A16, B7, C6, C14, D1, E11, E19, F10, G17, H18, K3, K15, M1, M17, N2, N14, P7, P9, P19, R3, R13, T19, U15, V8, W4, W6, W11, W13	Power-supply terminal for core logic (3.3 V)
P_V_{CCP}	124	L18	Primary bus-signaling environment supply. P_V_{CCP} is used in protection circuitry on primary bus I/O signals.
S_V_{CCP}	135	J17	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 PCI2050 PCI-to-PCI bridge features and functionality. Figure 3–1 shows a simplified block diagram of a typical system implementation using the PCI2050.

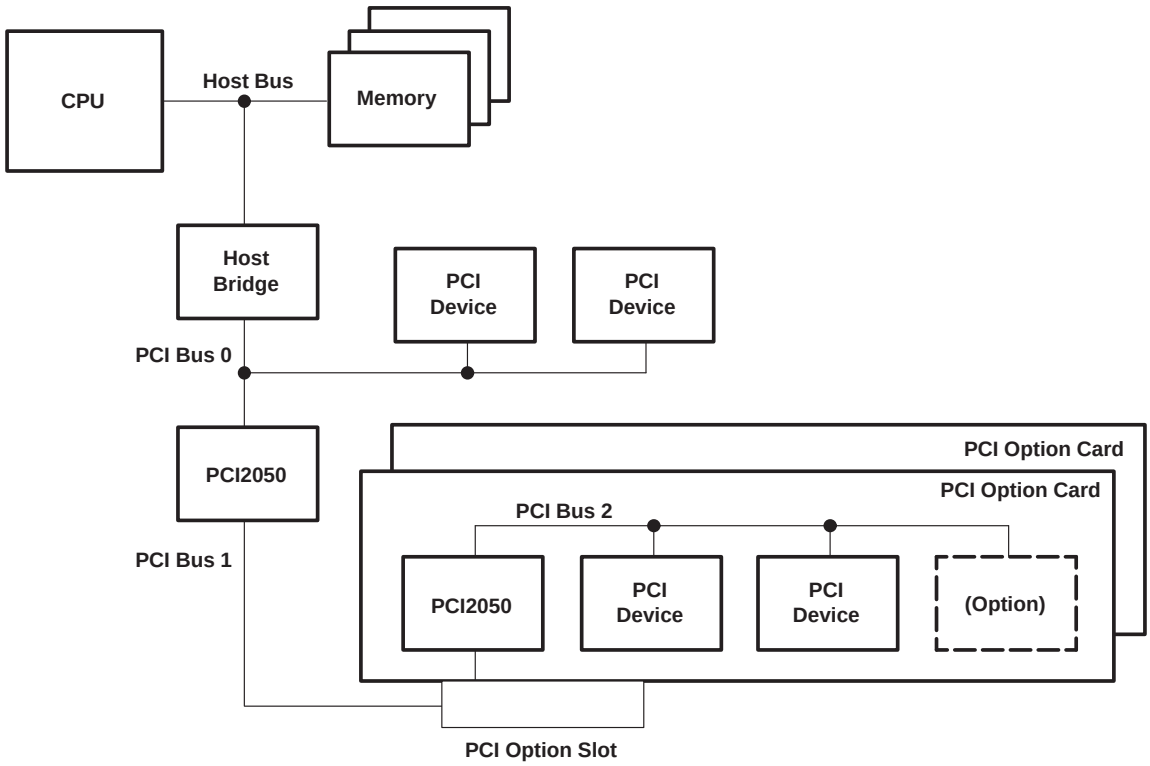


Figure 3–1. System Block Diagram

3.1 Introduction to the PCI2050

The PCI2050 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 nine 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 PCI2050 bridge defaulting to the highest priority tier.

Upon system power up, power-on self-test (POST) software configures the bridge according to the devices that exist on subordinate buses, and enables performance-enhancing features of the PCI2050. 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, 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 PCI2050 converts memory write and invalidate commands to memory write commands when forwarding transactions from either the primary or secondary side of the bridge if the bridge cannot guarantee that an entire cache line will be delivered.

3.3 Configuration Cycles

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 double-word 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, then 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 (see Table 4–1).

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. The P_AD bus encoding during the address phase of a type 1 cycle is shown in Figure 3–3. 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	1

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 PCI2050 bus number registers would be programmed in this case).

When the PCI2050 claims a type 1 configuration cycle that has a bus number equal to its secondary bus number, the PCI2050 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 the 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	–

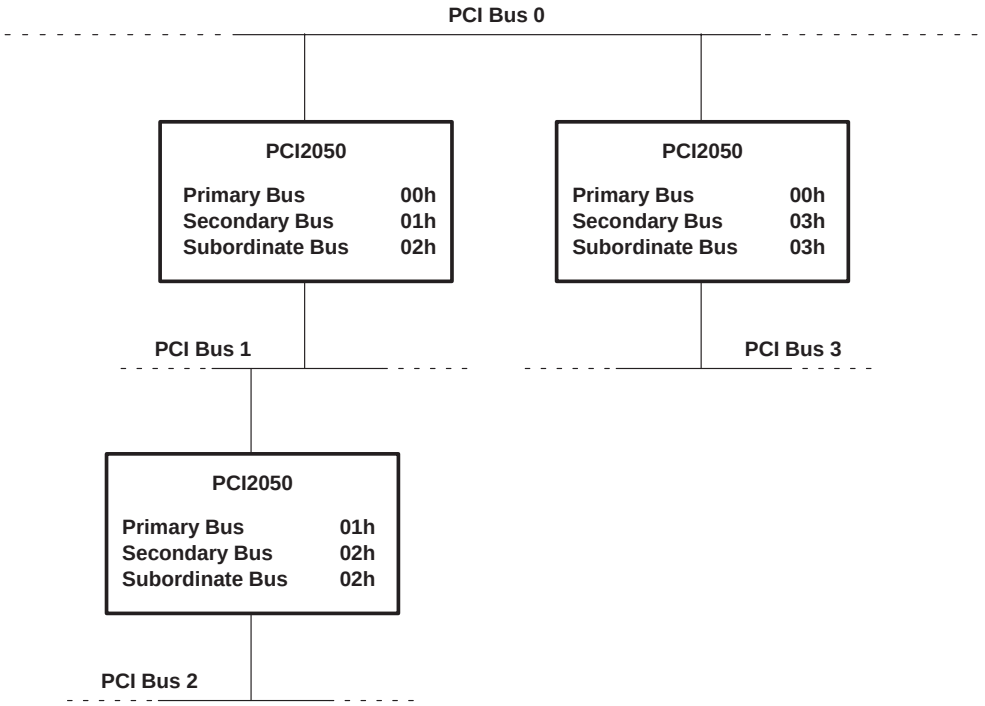


Figure 3–4. Bus Hierarchy and Numbering

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, the device number field is 1Fh, and the function number field is 07h, then 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, 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 PCI2050 provides 10 secondary clock outputs (S_CLKOUT[0:9]). Nine are provided for clocking secondary devices. The tenth clock should be routed back into the PCI2050 S_CLK input to ensure all secondary bus devices see the same clock.

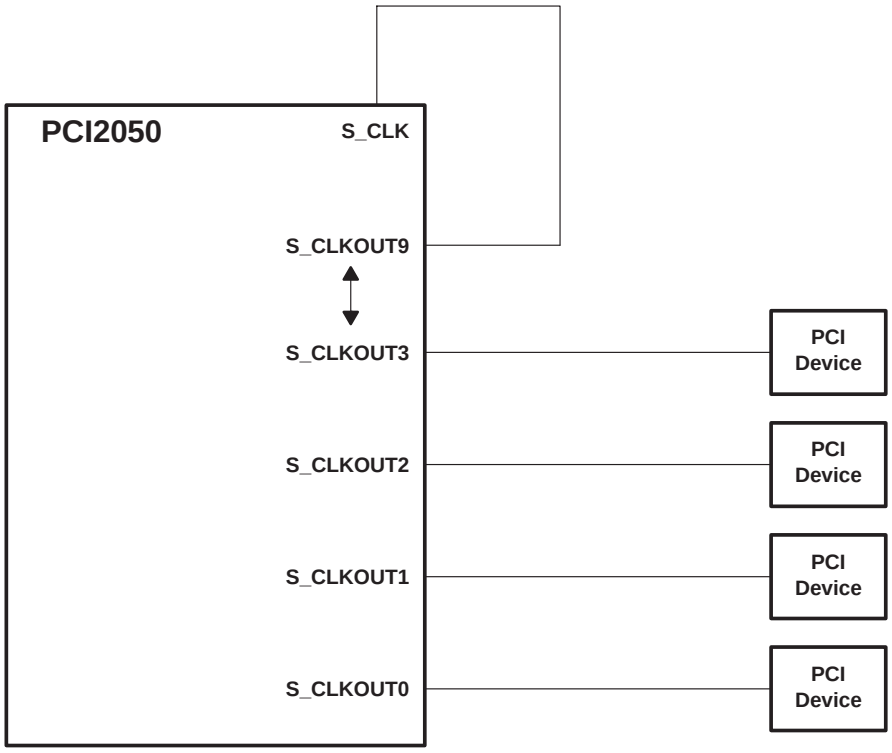


Figure 3–5. Secondary Clock Block Diagram

3.6 Bus Arbitration

The PCI2050 implements bus request ($\overline{\text{P\_REQ}}$) and bus grant ($\overline{\text{P\_GNT}}$) terminals for primary PCI bus arbitration. Nine secondary bus requests and nine secondary bus grants are provided on the secondary of the PCI2050. Ten potential initiators, including the bridge, can be located on the secondary bus. The PCI2050 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 PCI2050, acting as an initiator on the primary bus, asserts $\overline{\text{P\_REQ}}$ when forwarding transactions upstream to the primary bus. 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 PCI2050, then the device deasserts $\overline{\text{P\_REQ}}$ 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 PCI2050, 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 PCI2050, the device responds by parking the P_AD31–P_AD0 bus, the C/$\overline{\text{BE3}}$–C/$\overline{\text{BE0}}$ bus, and primary parity (P_PAR) by driving them to valid logic levels. If the PCI2050 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 PCI2050 provides nine secondary bus request terminals and nine secondary

bus grant terminals. Including the bridge, there are a total of ten potential secondary bus masters. These request and grant signals are connected to the internal arbiter. When an external arbiter is implemented, S_REQ8–S_REQ1 and S_GNT8–S_GNT1 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 PCI2050 internal bus arbiter. When using an external arbiter, the PCI2050 internal arbiter should be disabled by pulling S_CFN high.

When an external secondary bus arbiter is used, the PCI2050 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_GNT8–S_GNT1) are placed in a high impedance mode. Any unused secondary bus request inputs (S_REQ8–S_REQ1) should be pulled high to prevent the inputs from oscillating.

3.7 Decode Options

The PCI2050 supports positive decoding on the primary interface and negative decoding on the secondary interface. 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 of an assigned address range.

3.8 System Error Handling

The PCI2050 can be configured to signal a system error (SERR) for a variety of conditions. The P_SERR event disable register (offset 64h, see Section 5.4) and the P_SERR status register (offset 6Ah, see Section 5.9) 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 PCI2050 will not signal SERR. If the PCI2050 is configured to signal SERR by setting bit 8 in 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 SERR, bit 14 in the secondary status register (offset 1Eh, see Section 4.19) is set.

3.8.1 Posted Write Parity Error

If bit 1 in the P_SERR event disable register (offset 64h, see Section 5.4) is 0, then parity errors on the target bus during a posted write are passed to the initiating bus as a SERR. When this occurs, bit 1 of the P_SERR status register (offset 6Ah, see Section 5.9) is set. The status bit is cleared by writing a 1.

3.8.2 Posted Write Timeout

If bit 2 in the P_SERR event disable register (offset 64h, see Section 5.4) is 0 and the retry timer expires while attempting to complete a posted write, then the PCI2050 signals SERR on the initiating bus. When this occurs, bit 2 of the P_SERR status register (offset 6Ah, see Section 5.9) is set. The status bit is cleared by writing a 1.

3.8.3 Target Abort on Posted Writes

If bit 3 in the P_SERR event disable register (offset 64h, see Section 5.4) is 0 and the bridge receives a target abort during a posted write transaction, then the PCI2050 signals SERR on the initiating bus. When this occurs, bit 3 of the P_SERR status register (offset 6Ah, see Section 5.9) is set. The status bit is cleared by writing a 1.

3.8.4 Master Abort on Posted Writes

If bit 4 in the P_SERR event disable register (offset 64h, see Section 5.4) is 0 and a posted write transaction results in a master abort, then the PCI2050 signals $\overline{\text{SERR}}$ on the initiating bus. When this occurs, bit 4 of the P_SERR status register (offset 6Ah, see Section 5.9) is set. The status bit is cleared by writing a 1.

3.8.5 Master Delayed Write Timeout

If bit 5 in the P_SERR event disable register (offset 64h, see Section 5.4) is 0 and the retry timer expires while attempting to complete a delayed write, then the PCI2050 signals $\overline{\text{SERR}}$ on the initiating bus. When this occurs, bit 5 of the P_SERR status register (offset 6Ah, see Section 5.9) is set. The status bit is cleared by writing a 1.

3.8.6 Master Delayed Read Timeout

If bit 6 in the P_SERR event disable register (offset 64h, see Section 5.4) is 0 and the retry timer expires while attempting to complete a delayed read, then the PCI2050 signals $\overline{\text{SERR}}$ on the initiating bus. When this occurs, bit 6 of the P_SERR status register (offset 6Ah, see Section 5.9) is set. The status bit is cleared by writing a 1.

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

The PCI2050 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 (offset 04h, see Section 4.3) is 1) and bit 1 in the bridge control register (offset 3Eh, see Section 4.32) is set.

3.9 Parity Handling and Parity Error Reporting

When forwarding transactions, the PCI2050 attempts to pass the data parity condition from one interface to the other unchanged, whenever possible, to allow the master and target devices to handle the error condition.

3.9.1 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 PCI2050 signals $\overline{\text{SERR}}$ on address parity errors and target abort transactions.

3.9.2 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 PCI2050 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 (offset 04h, see Section 4.3), then bit 8 (data parity error detected) in the status register (offset 06h, see Section 4.4) 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 $\overline{\text{PERR}}$.

3.10 Master and Target Abort Handling

If the PCI2050 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 PCI2050 follows bit 5 (master abort mode) in the bridge control register (offset 3Eh, see Section 4.32) for reporting errors.

3.11 Discard Timer

The PCI2050 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). *PCI Local Bus Specification* recommends that a bridge wait 2^{15} PCI clocks before discarding the transaction data or status.

The PCI2050 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 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 PCI2050 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*.

3.12 Delayed Transactions

The bridge supports delayed transactions as defined in *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 eight 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, 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.

The PCI supports up to three delayed transactions in each direction at any given time.

3.13 Mode Selection

Table 3–3 shows the mode selection via MS0 (PDV terminal 155, GHK terminal E17) and MS1 (PDV terminal 106, GHK terminal R17).

Table 3–3. Configuration Via MS0 and MS1

MS0	MS1	MODE
0	0	Compact PCI hot-swap friendly *PCI Bus Power Management Interface Specification Revision 1.1* HSSWITCH/GPIO(3) functions as HSSWITCH
0	1	Compact PCI hot-swap disabled *PCI Bus Power Management Interface Specification Revision 1.1* HSSWITCH/GPIO(3) functions as GPIO(3)
1	X	Intel compatible No CPCI hot swap, *PCI Bus Power Management Interface Specification Revision 1.0*

3.14 Compact PCI Hot-Swap Support

The PCI2050 is hot-swap friendly silicon that supports all of the 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 PCI2050 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 PCI2050 provides this functionality such that it can be implemented on a board that can be removed and inserted in a hot-swap system.

The PCI2050 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.

3.15 JTAG Support

The PCI2050 implements a JTAG test port based on the IEEE Standard 1149.1, *IEEE Standard Test Access Port and Boundary-Scan Architecture*. The JTAG test port consists of the following:

- A 5-wire test access port
- A test access port controller
- An instruction register
- A bypass register
- A boundary-scan register

3.15.1 Test Port Instructions

The PCI 2050 supports the following JTAG instructions:

- EXTEST, BYPASS, and SAMPLE
- HIGHZ and CLAMP
- Private (various private instructions used by TI for test purposes)

Table 3–4. JTAG Instructions and Op Codes

INSTRUCTION	OP CODE	DESCRIPTION
EXTEST	00000	External test: drives terminals from the boundary scan register
SAMPLE	00001	Sample I/O terminals
CLAMP	00100	Drives terminals from the boundary scan register and selects the bypass register for shifts
HIGHZ	00101	Puts all outputs and I/O terminals except for the TDO terminal in a high–impedance state
BYPASS	11111	Selects the bypass register for shifts

Table 3–5. Boundary Scan Terminal Order

BOUNDARY SCAN REGISTER NO.	PDV TERMINAL NUMBER	GHK TERMINAL NUMBER	TERMINAL NAME	GROUP DISABLE REGISTER	BOUNDARY-SCAN CELL TYPE
0	137	J15	S_AD0	19	Bidirectional
1	138	H19	S_AD1	19	Bidirectional
2	140	H17	S_AD2	19	Bidirectional
3	141	H14	S_AD3	19	Bidirectional
4	143	G19	S_AD4	19	Bidirectional
5	144	G18	S_AD5	19	Bidirectional
6	146	G14	S_AD6	19	Bidirectional
7	147	F19	S_AD7	19	Bidirectional
8	149	G15	$\overline{\text{S\_C/BE0}}$	19	Bidirectional
9	150	F17	S_AD8	19	Bidirectional
10	152	F14	S_AD9	19	Bidirectional
11	153	E18	S_M66ENA	19	Bidirectional
12	154	F15	S_AD10	19	Bidirectional
13	155	E17	MS0	–	Input
14	158	C15	S_AD11	19	Bidirectional
15	161	B15	S_AD12	19	Bidirectional
16	162	A15	S_AD13	19	Bidirectional
17	164	B14	S_AD14	19	Bidirectional
18	165	E13	S_AD15	19	Bidirectional
19	–	–	–	–	Control

Table 3–5. Boundary Scan Terminal Order (continued)

BOUNDARY SCAN REGISTER NO.	PDV TERMINAL NUMBER	GHK TERMINAL NUMBER	TERMINAL NAME	GROUP DISABLE REGISTER	BOUNDARY-SCAN CELL TYPE
20	167	F12	S_C/BE1	19	Bidirectional
21	168	C13	S_PAR	19	Bidirectional
22	169	B13	S_SERR	–	Input
23	171	E13	S_PERR	26	Bidirectional
24	172	C12	S_LOCK	26	Bidirectional
25	173	B12	S_STOP	26	Bidirectional
26	–	–	–	–	Control
27	175	A11	S_DEVSEL	26	Bidirectional
28	176	B11	S_TRDY	26	Bidirectional
29	177	C11	S_IRDY	26	Bidirectional
30	179	F11	S_FRAME	26	Bidirectional
31	180	A10	S_C/BE2	48	Bidirectional
32	182	C10	S_AD16	48	Bidirectional
33	183	E10	S_AD17	48	Bidirectional
34	185	A9	S_AD18	48	Bidirectional
35	186	B9	S_AD19	48	Bidirectional
36	188	F9	S_AD20	48	Bidirectional
37	189	E9	S_AD21	48	Bidirectional
38	191	B8	S_AD22	48	Bidirectional
39	192	C8	S_AD23	48	Bidirectional
40	194	E8	S_C/BE3	48	Bidirectional
41	195	A7	S_AD24	48	Bidirectional
42	197	C7	S_AD25	48	Bidirectional
43	198	F7	S_AD26	48	Bidirectional
44	200	B6	S_AD27	48	Bidirectional
45	201	E7	S_AD28	48	Bidirectional
46	203	A5	S_AD29	48	Bidirectional
47	204	F6	S_AD30	48	Bidirectional
48	–	–	–	–	Control
49	206	E6	S_AD31	48	Bidirectional
50	207	C5	S_REQ0	–	Input
51	2	E3	S_REQ1	–	Input
52	3	F5	S_REQ2	–	Input
53	4	G6	S_REQ3	–	Input
54	5	E2	S_REQ4	–	Input
55	6	E1	S_REQ5	–	Input
56	7	F3	S_REQ6	–	Input
57	8	F2	S_REQ7	–	Input
58	9	G5	S_REQ8	–	Input

Table 3–5. Boundary Scan Terminal Order (continued)

BOUNDARY SCAN REGISTER NO.	PDV TERMINAL NUMBER	GHK TERMINAL NUMBER	TERMINAL NAME	GROUP DISABLE REGISTER	BOUNDARY-SCAN CELL TYPE
59	10	F1	$\overline{\text{S\_GNT0}}$	61	Output
60	11	H6	$\overline{\text{S\_GNT1}}$	61	Output
61	–	–	–	–	Control
62	13	G2	$\overline{\text{S\_GNT2}}$	61	Output
63	14	G1	$\overline{\text{S\_GNT3}}$	61	Output
64	15	H5	$\overline{\text{S\_GNT4}}$	61	Output
65	16	H3	$\overline{\text{S\_GNT5}}$	61	Output
66	17	H2	$\overline{\text{S\_GNT6}}$	61	Output
67	18	H1	$\overline{\text{S\_GNT7}}$	61	Output
68	19	J1	$\overline{\text{S\_GNT8}}$	61	Output
69	21	J3	S_CLK	–	Input
70	22	J5	$\overline{\text{S\_RST}}$	78	Output
71	23	J6	$\overline{\text{S\_CFN}}$	–	Input
72	24	K1	GPIO3	78	Bidirectional
73	25	K2	GPIO2	78	Bidirectional
74	27	K5	GPIO1	78	Bidirectional
75	28	K6	GPIO0	78	Bidirectional
76	29	L1	S_CLKOUT0	–	Output
77	30	L2	S_CLKOUT1	–	Output
78	–	–	–	–	Output
79	32	L6	S_CLKOUT2	–	Output
80	33	L5	S_CLKOUT3	–	Output
81	35	M2	S_CLKOUT4	–	Output
82	36	M3	S_CLKOUT5	–	Output
83	38	M5	S_CLKOUT6	–	Output
84	39	N1	S_CLKOUT7	–	Output
85	41	N3	S_CLKOUT8	–	Output
86	42	N6	S_CLKOUT9	–	Output
87	43	P1	$\overline{\text{P\_RST}}$	–	Input
88	44	P2	BPCCE	–	Input
89	45	N5	P_CLK	–	Input
90	46	P3	$\overline{\text{P\_GNT}}$	–	Input
91	47	R1	$\overline{\text{P\_REQ}}$	92	Output
92	–	–	–	–	Control
93	49	R2	P_AD31	111	Bidirectional
94	50	P5	P_AD30	111	Bidirectional
95	55	R6	P_AD29	111	Bidirectional
96	57	V5	P_AD28	111	Bidirectional
97	58	W5	P_AD27	111	Bidirectional

Table 3–5. Boundary Scan Terminal Order (continued)

BOUNDARY SCAN REGISTER NO.	PDV TERMINAL NUMBER	GHK TERMINAL NUMBER	TERMINAL NAME	GROUP DISABLE REGISTER	BOUNDARY-SCAN CELL TYPE
98	60	V6	P_AD26	111	Bidirectional
99	61	R7	P_AD25	111	Bidirectional
100	63	W6	P_AD24	111	Bidirectional
101	64	U7	$\overline{\text{P\_C/BE3}}$	111	Bidirectional
102	65	V7	P_IDSEL	–	Input
103	67	R8	P_AD23	111	Bidirectional
104	68	U8	P_AD22	111	Bidirectional
105	70	W8	P_AD21	111	Bidirectional
106	71	W9	P_AD20	111	Bidirectional
107	73	U9	P_AD19	111	Bidirectional
108	74	R9	P_AD18	111	Bidirectional
109	76	W10	P_AD17	111	Bidirectional
110	77	V10	P_AD16	111	Bidirectional
111	–	–	–	–	Control
112	79	R10	$\overline{\text{P\_C/BE2}}$	111	Bidirectional
113	80	P10	$\overline{\text{P\_FRAME}}$	118	Bidirectional
114	82	V11	$\overline{\text{P\_IRDY}}$	118	Bidirectional
115	83	U11	$\overline{\text{P\_TRDY}}$	118	Bidirectional
116	84	P11	$\overline{\text{P\_DEVSEL}}$	118	Bidirectional
117	85	R11	$\overline{\text{P\_STOP}}$	118	Bidirectional
118	–	–	–	–	Control
119	87	V12	$\overline{\text{P\_LOCK}}$	118	Input
120	88	U12	$\overline{\text{P\_PERR}}$	118	Bidirectional
121	89	P12	$\overline{\text{P\_SERR}}$	142	Output
122	90	R12	P_PAR	142	Bidirectional
123	92	V13	$\overline{\text{P\_C/BE1}}$	142	Bidirectional
124	93	U13	P_AD15	142	Bidirectional
125	95	W14	P_AD14	142	Bidirectional
126	96	V14	P_AD13	142	Bidirectional
127	98	U14	P_AD12	142	Bidirectional
128	99	W15	P_AD11	142	Bidirectional
129	101	V15	P_AD10	142	Bidirectional
130	106	R17	MS1	–	Input
131	107	P15	P_AD9	142	Bidirectional
132	109	R18	P_AD8	142	Bidirectional
133	110	R19	$\overline{\text{P\_C/BE0}}$	142	Bidirectional
134	112	P18	P_AD7	142	Bidirectional
135	113	N15	P_AD6	142	Bidirectional
136	115	M14	P_AD5	142	Bidirectional
137	116	N17	P_AD4	142	Bidirectional
138	118	N19	P_AD3	142	Bidirectional

Table 3–5. Boundary Scan Terminal Order (continued)

BOUNDARY SCAN REGISTER NO.	PDV TERMINAL NUMBER	GHK TERMINAL NUMBER	TERMINAL NAME	GROUP DISABLE REGISTER	BOUNDARY-SCAN CELL TYPE
139	119	M15	P_AD2	142	Bidirectional
140	121	M18	P_AD1	142	Bidirectional
141	122	M19	P_AD0	142	Bidirectional
142	–	–	–	–	–
143	126	L15	MSK_IN	–	Input
144	–	–	–	–	Control
145	127	L14	$\overline{\text{HS\_ENUM}}$	144	Output
146	128	K19	HS_LED	144	Output

3.16 GPIO Interface

The PCI2050 implements a four-terminal general-purpose I/O interface. Besides functioning as a general-purpose I/O interface, the GPIO terminals can be used to read in the secondary clock mask and to stop the bridge from accepting I/O and memory transactions.

3.16.1 Secondary Clock Mask

The PCI2050 uses GPIO0, GPIO2, and MSK_IN to shift in the secondary clock mask from an external shift register.

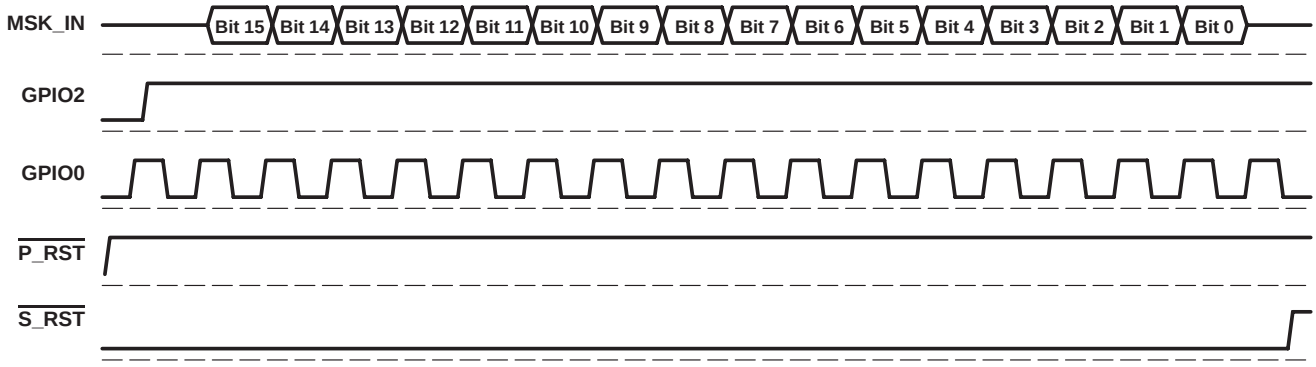


Figure 3–6. Clock Mask Read Timing After Reset

Table 3–6. Clock Mask Data Format

BIT	CLOCK
[0:1]	S_CLKOUT0
[2:3]	S_CLKOUT1
[4:5]	S_CLKOUT2
[6:7]	S_CLKOUT3
8	S_CLKOUT4
9	S_CLKOUT5
10	S_CLKOUT6
11	S_CLKOUT7
12	S_CLKOUT8
13	S_CLKOUT9 (PCI2050 S_CLK input)
[14:15]	Reserved

3.16.2 Transaction Forwarding Control

The PCI 2050 will stop forwarding I/O and memory transactions if bit 5 of the chip control register (offset 40h, see Section 5.1) is set to 1 and GPIO3 is driven high. The bridge will complete all queued posted writes and delayed requests, but delayed completions will not be returned until GPIO3 is driven low and transaction forwarding is resumed. The bridge will continue to accept configuration cycles in this mode. This feature is not available when in compact PCI hot-swap mode because GPIO3 iis used as the $\overline{\text{HS\_SWITCH}}$ input in this mode.

3.17 PCI Power Management

The *PCI Power Management 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 of the PCI bus are B0–B3. The bus power states B0–B3 are derived from the device power state of the originating PCI2050 device.

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.

3.17.1 Behavior in Low Power States

The PCI2050 supports D0, D1, D2, and $D3_{hot}$ power states when in TI mode. The PCI2050 only supports D0 and D3 power states when in Intel mode. The PCI2050 is fully functional only in D0 state. In the lower power states, the bridge does not accept any memory or I/O transactions. These transactions are aborted by the master. The bridge accepts type 0 configuration cycles in all power states except $D3_{cold}$. The bridge also accepts type 1 configuration cycles but does not pass these cycles to the secondary bus in any of the lower 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 going from $D3_{hot}$ to D0, an internal reset is generated. This reset initializes all PCI configuration registers to their default values. All TI specific registers (40h – FFh) are not reset. Power management registers are also not reset.

4 Bridge Configuration Header

The PCI2050 bridge is a single-function PCI device. The configuration header is in compliance with the *PCI-to-PCI Bridge Architecture Specification 1.0*. 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 0				10h
Base address 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
Reserved				44h
Reserved				48h
Reserved				4Ch
Reserved				50h
Reserved				54h
Reserved				58h
Reserved		Reserved		5Ch
Reserved	Reserved	Reserved	Reserved	60h
GPIO input data	GPIO output enable	GPIO output data	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

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 PCI2050 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 4–2 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 4–2. Command Register

BIT	TYPE	FUNCTION
15–10	R	Reserved
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. 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 4–3 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/W	R/W	R/W	R/W	R/W	R	R	R/W	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/write (see individual bit descriptions)
Offset: 06h
Default: 0210h

Table 4–3. Status Register

BIT	TYPE	FUNCTION
15	R/W	Detected parity error. Bit 15 is set when a parity error is detected.
14	R/W	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 3.8, System Error Handling. 0 = No SERR signaled (default) 1 = Signals SERR
13	R/W	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/W	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/W	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.
8	R/W	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. The parity error response bit (bit 6) 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 PCI2050 does not support the user-definable features; therefore, bit 6 is hardwired to 0.
5	R	66-MHz capable. The PCI2050 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 PCI2050.

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	0

Register: **Revision ID**
Type: Read-only
Offset: 08h
Default: 00h (reflects the current revision of the silicon)

4.6 Class Code Register

This register categorizes the PCI2050 as a PCI-to-PCI bridge device (0604h) with a 00h programming 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	0

Register: **Class code**
Type: Read-only
Offset: 09h
Default: 060400h

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 PCI2050 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 PCI2050 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 PCI2050 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 PCI2050 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/W	R/W	R/W	R/W	R/W	R	R	R/W	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/write (see individual bit descriptions)
Offset: 1Eh
Default: 0200h

Table 4–4. Secondary Status Register

BIT	TYPE	FUNCTION
15	R/W	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/W	Received system error. Bit 14 is set when the secondary interface detects S_SERR asserted. Note that the bridge never asserts S_SERR. 0 = No S_SERR detected on the secondary bus (default) 1 = S_SERR detected on the secondary bus
13	R/W	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/W	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/W	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. These read-only bits encode the timing of S_DEVSEL and are hardwired to 01b, indicating that the bridge asserts this signal at a medium speed.
8	R/W	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. 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 1) is set in the bridge control register (offset 3Eh, see 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 prefetchable base upper 32 bits register plus the prefetchable memory base register defines the base address of the 64-bit prefetchable memory address range used by the bridge to determine when to forward memory transactions from one interface to the other. The prefetchable base upper 32 bits register should be programmed to all zeros when 32-bit addressing is being used.

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/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	Prefetchable base upper 32 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: **Prefetchable base upper 32 bits**
Type: Read/Write
Offset: 28h
Default: 0000 0000h

4.25 Prefetchable Limit Upper 32 Bits Register

The prefetchable LIMIT upper 32 bits register plus the prefetchable memory LIMIT register defines the base address of the 64-bit prefetchable memory address range used by the bridge to determine when to forward memory transactions from one interface to the other. The prefetchable LIMIT upper 32 bits register should be programmed to all zeros when 32-bit addressing is being used.

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/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	Prefetchable limit upper 32 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: **Prefetchable limit upper 32 bits**
Type: Read/Write
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 PCI2050 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 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	R/W	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 4–5. 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	R/W	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 $\overline{\text{S\_RST}}$.

Table 4–5. Bridge Control Register (continued)

BIT	TYPE	FUNCTION
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, the posted write transaction has completed on the requesting interface, and SERR enable (bit 8) of the command register (offset 04h, see Section 4.3) is 1, then P_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 SERR is enabled via bit 1 of this register, by asserting SERR.
4	R	Reserved. Returns 0 when read. Writes have no effect.
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.
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	SERR enable. Bit 1 controls the forwarding of secondary interface SERR assertions to the primary interface. Only when this bit is set will the bridge forward S_SERR to the primary bus signal P_SERR. For the primary interface to assert SERR, bit 8 of the command register (offset 04h, see Section 4.3) must be set. 0 = SERR disabled (default) 1 = 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 PCI2050). 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.

5.1 Chip Control Register

The chip control register contains read/write and read-only bits has a default value of 00h. This register is used to control the functionality of certain PCI transactions.

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

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

Table 5–1. Chip Control Register

BIT	TYPE	FUNCTION
7–6	R	Reserved. Bits 7–5 return 0s when read.
5	R/W	Transaction forwarding control for I/O and memory cycles. 0 = Transaction forwarding controlled by bits 0 and 1 of the command register (offset 04h, see Section 4.3) (default). 1 = Transaction forwarding will be disabled if GPIO3 is driven high.
4	R/W	Memory read prefetch. When set, bit 4 enables the memory read prefetch. 0 = Upstream memory reads are disabled (default). 1 = Upstream memory reads are enabled
3–2	R	Reserved. Bits 3 and 2 return 0s when read.
1	R/W	Memory write and memory write and invalidate disconnect control. 0 = Disconnects on queue full or 4-KB boundaries (default) 1 = Disconnects on queue full, 4-KB boundaries and cacheline boundries.
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 PCI2050 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 PCI2050 to set bit 6 of the bridge control register (offset 3Eh, see Section 4.32) and then internally reset the PCI2050. 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 PCI2050 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/W	R/W	R/W	R/W	R/W	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 PCI2250 bridge is placed in the two-tier arbitration scheme. 0 = Lowest priority tier 1 = Highest priority tier (default)
8	R/W	GNT8 tier select. This bit determines in which tier the S_GNT8 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier
7	R/W	GNT7 tier select. This bit determines in which tier the S_GNT7 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier
6	R/W	GNT6 tier select. This bit determines in which tier the S_GNT6 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier
5	R/W	GNT5 tier select. This bit determines in which tier the S_GNT5 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier
4	R/W	GNT4 tier select. This bit determines in which tier the S_GNT4 is placed in the arbitration scheme. This bit is encoded as: 0 = Lowest priority tier (default) 1 = Highest priority tier
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 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.

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 5–4. 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.

5.5 GPIO Output Data Register

The GPIO output data register controls the data driven on the GPIO terminals configured as outputs.

Bit	7	6	5	4	3	2	1	0
Name	GPIO output data							
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: **GPIO output data**
Type: Read/Write
Offset: 65h
Default: 00h

Table 5–5. GPIO Output Data Register

BIT	TYPE	FUNCTION
7–4	R/W	GPIO3–GPIO0 output high. Writing a 1 to any of these bits causes the corresponding GPIO signal to be driven high. Writing a 0 has no effect. GPIO terminals programmed as inputs are not affected by these bits.
3–0	R/W	GPIO3–GPIO0 output low. Writing a 1 to any of these bits causes the corresponding GPIO signal to be driven low. Writing a 0 has no effect. GPIO terminals programmed as inputs are not affected by these bits.

5.6 GPIO Output Enable Register

The GPIO output enable register controls the direction of the GPIO signal. By default all GPIO terminals are inputs.

Bit	7	6	5	4	3	2	1	0
Name	GPIO output enable							
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: **GPIO output enable**
Type: Read/Write
Offset: 66h
Default: 00h

Table 5–6. GPIO Output Enable Register

BIT	TYPE	FUNCTION
7–4	R/W	GPIO3–GPIO0 output enable. Writing a 1 to any of these bits causes the corresponding GPIO signal to be configured as an output. Writing a 0 has no effect.
3–0	R/W	GPIO3–GPIO0 input enable. Writing a 1 to any of these bits causes the corresponding GPIO signal to be configured as an input. Writing a 0 has no effect.

5.7 GPIO Input Data Register

The GPIO input data register returns the current state of the GPIO terminals when read.

Bit	7	6	5	4	3	2	1	0
Name	GPIO input data							
Type	R	R	R	R	R	R	R	R
Default	0	0	0	0	0	0	0	0

Register: **GPIO input data**
Type: Read-only
Offset: 67h
Default: 00h

Table 5–7. GPIO Input Data Register

BIT	TYPE	FUNCTION
7–4	R	GPIO3–GPIO0 input data. These four bits return the current state of the GPIO terminals.
3–0	R	Reserved. Bits 3–0 return 0s when read.

5.8 Secondary Clock Control Register

The secondary clock control register is used to control the secondary clock outputs.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Name	Secondary clock control															
Type	R	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: **Secondary clock control**
Type: Read-only, Read/Write
Offset: 68h
Default: 0000h

Table 5–8. Secondary Clock Control Register

BIT	TYPE	FUNCTION
15	R	Reserved. Bits 15 returns 0 when read.
14	R/W	Clockout10 disable. 0 = Clockout10 enabled (default). 1 = Clockout10 disabled and driven high.
13	R/W	Clockout9 disable. 0 = Clockout9 enabled (default). 1 = Clockout9 disabled and driven high.
12	R/W	Clockout8 disable. 0 = Clockout8 enabled (Default). 1 = Clockout8 disabled and driven high.
11	R/W	Clockout7 disable. 0 = Clockout7 enabled (default). 1 = Clockout7 disabled and driven high.
10	R/W	Clockout6 disable. 0 = Clockout6 enabled (default). 1 = Clockout6 disabled and driven high.
9	R/W	Clockout5 disable. 0 = Clockout5 enabled (default). 1 = Clockout5 disabled and driven high.
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 is the default). 11 = Clockout3 disabled and driven high.
5–4	R/W	Clockout2 disable. 00, 01, 10 = Clockout2 enabled (00 is the default). 11 = Clockout2 disabled and driven high.
3–2	R/W	Clockout1 disable. 00, 01, 10 = Clockout1 enabled (00 is the default). 11 = Clockout1 disabled and driven high.
1–0	R/W	Clockout0 disable. 00, 01, 10 = Clockout0 enabled (00 is the default). 11 = Clockout0 disabled and driven high.

5.9 P_SERR Status Register

The P_SERR status register indicates what caused a SERR event on the primary interface.

Bit	7	6	5	4	3	2	1	0
Name	P_SERR status							
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 status**
Type: Read-only
Offset: 6Ah
Default: 00h

Table 5–9. P_SERR Status Register

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

5.10 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

5.11 PM Next Item Pointer Register

The PM 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 PCI2050 supports more than one extended capability, but in all other modes returns 00h, indicating that only one extended capability is provided.

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 CPCI mode
00h All other modes

5.12 Power Management Capabilities Register

The power management capabilities register contains information on the capabilities of the PCI2050 functions related to power management. The PCI2050 function supports D0, D1, D2, and D3 power states when MS1 is low. The PCI2050 does not support any power states when MS1 is high.

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 5–10. Power Management Capabilities Register

BIT	TYPE	FUNCTION
15–11	R	$\overline{\text{PME}}$ 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 PCI2050 cannot assert $\overline{\text{PME}}$ signal from that power state. For the PCI2050, 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 when read because the PCI2050 does not support $\overline{\text{PME}}$ signaling.
3	R	PMECLK. This bit returns a 0 when read 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

5.13 Power Management Control/Status Register

The power management control/status register determines and changes the current power state of the PCI2050. The contents of this register are not affected by the internally generated reset caused by the transition from $D3_{hot}$ to D0 state.

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 5–11. Power Management Control/Status Register

BIT	TYPE	FUNCTION
15	R	$\overline{\text{PME}}$ status. This bit returns a 0 when read because the PCI2050 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 PCI2050 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 this is given below: 00 – D0 01 – D1 10 – D2 11 – $D3_{hot}$

5.14 PMCSR Bridge Support Register

The PMCSR bridge support register is required for all PCI bridges and supports PCI-bridge-specific functionality.

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 5–12. 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.

5.15 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

5.16 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

5.17 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 PCI2050 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

5.18 Hot Swap Control Status Register

The hot swap control status register contains control and status information for CPCI hot swap resources.

Bit	7	6	5	4	3	2	1	0
Name	Hot swap control status							
Type	R	R	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 5–13. Hot Swap Control Status Register

BIT	TYPE	FUNCTION
7	R	$\overline{\text{ENUM}}$ insertion status. When set, the $\overline{\text{ENUM}}$ output is driven by the PCI2050. This bit defaults to 0, and will be set after a PCI reset occurs, the pre-load of serial ROM is complete, the ejector handle is closed, and bit 6 is 0. Thus, this bit is set following an insertion when the board implementing the PCI2050 is ready for configuration. This bit cannot be set under software control.
6	R	$\overline{\text{ENUM}}$ extraction status. When set, the $\overline{\text{ENUM}}$ output is driven by the PCI2050. 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 PCI2050 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 PCI2050 and this bit will be 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 PCI2050 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
: S_V_{CCP} . . . . . –0.5 V to 6 V
: P_V_{CCP} . . . . . –0.5 V to 6 V
Input voltage range, V_I: PCI . . . . . –0.5 V to 6.5 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
P_V_{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	
S_V_{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}	
V_{IL}†	High-level input voltage	PCI	3.3 V	0		0.3 V_{CCP}	V
			5 V	0		0.8	
V_I	Input voltage	PCI		0		V_{CCP}	V
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
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
§ 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
P_V_{CCP}	PCI supply voltage	Commercial	3.3 V	3	3.3	3.6	V
			5 V	4.75	5	5.25	
S_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, PCI		V_I = V_{CCP}		10	µA
		I/O terminals¶		V_I = V_{CCP}		10	
I_{IL}‡	Low-level input current	Input terminals, PCI		V_I = GND		–1	µA
		I/O terminals¶				–10	
I_{OZ}	High-impedance output current			V_O = V_{CCP} or GND		±10	µA

† V_{OH} is not tested on $\overline{\text{PSERR}}$ or $\overline{\text{HSENUM}}$ due to open-drain configuration.
‡ I_{IH} and I_{IL} are not tested on NO_HSLED dur to its active ourput-only configuration.
¶ 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/$\overline{BE3}$–C/$\overline{BE0}$, $\overline{FRAME}$, $\overline{TRDY}$, $\overline{IRDY}$, $\overline{STOP}$, IDSEL, $\overline{DEVSEL}$, and PAR.

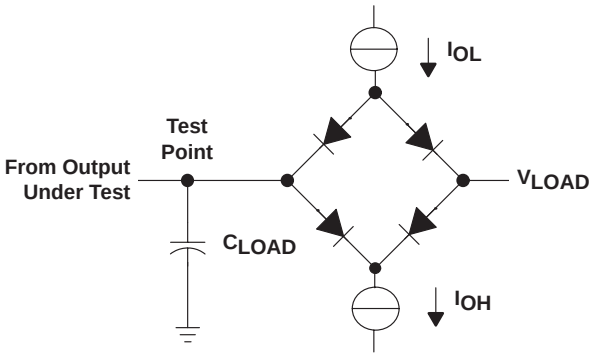
6.7 Parameter Measurement Information

LOAD CIRCUIT PARAMETERS

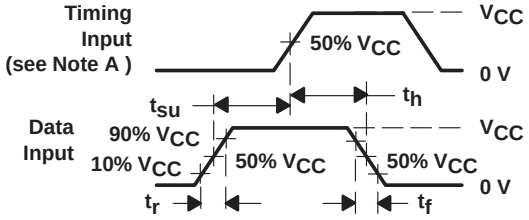
TIMING PARAMETER		C_{LOAD}† (pF)	I_{OL} (mA)	I_{OH} (mA)	V_{LOAD} (V)
t_{en}	t_{PZH}	50	8	–8	0
	t_{PZL}				3
t_{dis}	t_{PHZ}	50	8	–8	1.5
	t_{PLZ}				
t_{pd}		50	8	–8	‡

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

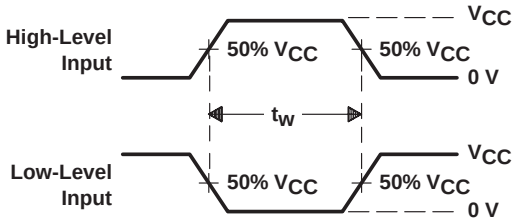
‡ $\frac{V_{LOAD} - V_{OL}}{I_{OL}}$ = 50 Ω, where V_{OL} = 0.6 V, I_{OL} = 8 mA



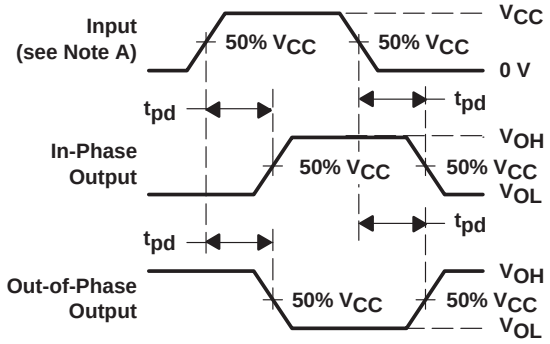
LOAD CIRCUIT



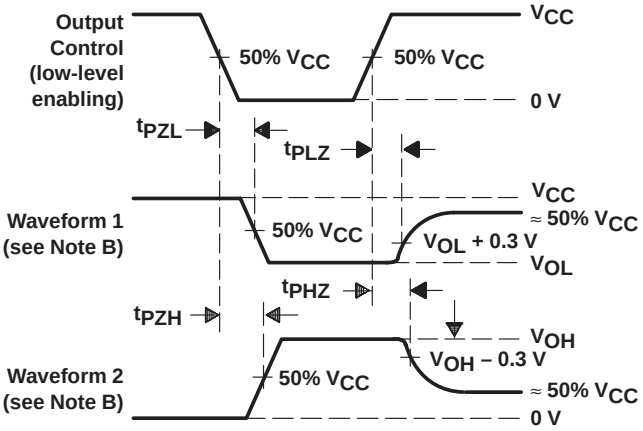
VOLTAGE WAVEFORMS
SETUP AND HOLD TIMES
INPUT RISE AND FALL TIMES



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

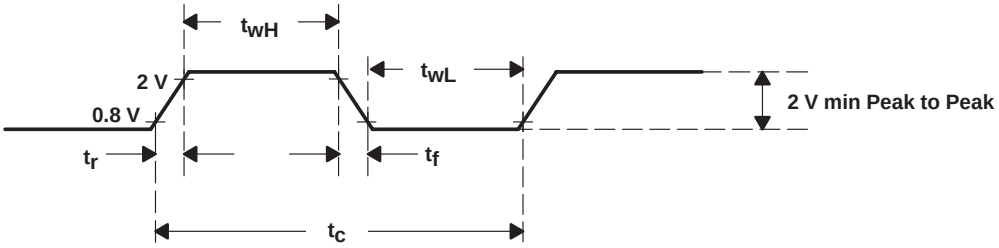


Figure 6–2. PCLK Timing Waveform

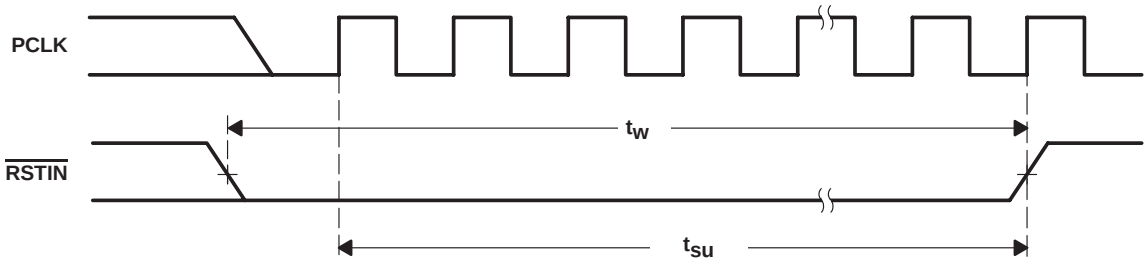


Figure 6–3. $\overline{\text{RSTIN}}$ Timing Waveforms

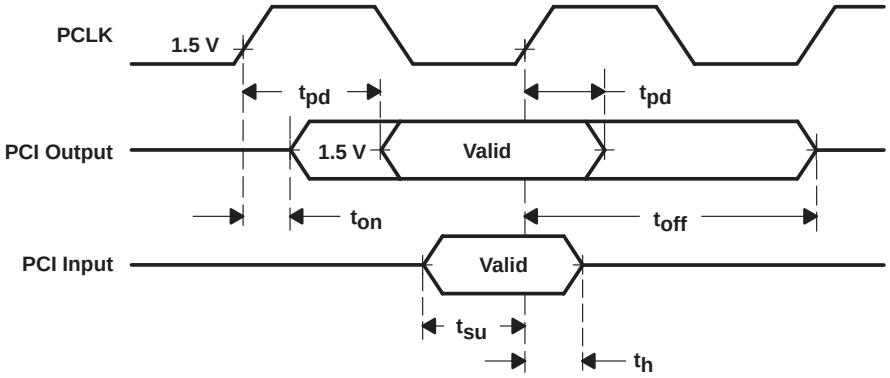
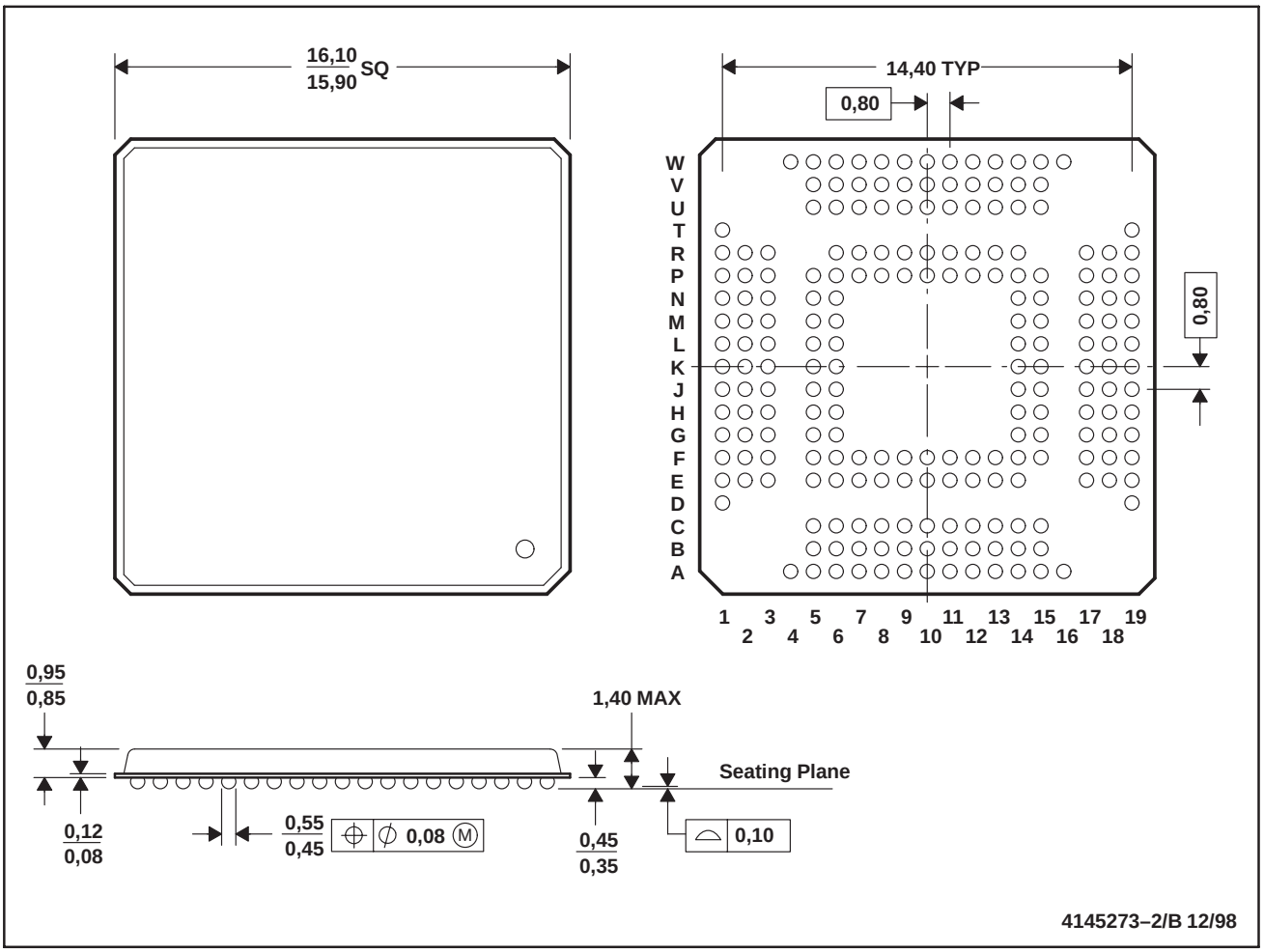


Figure 6–4. Shared-Signals Timing Waveforms

7 Mechanical Data

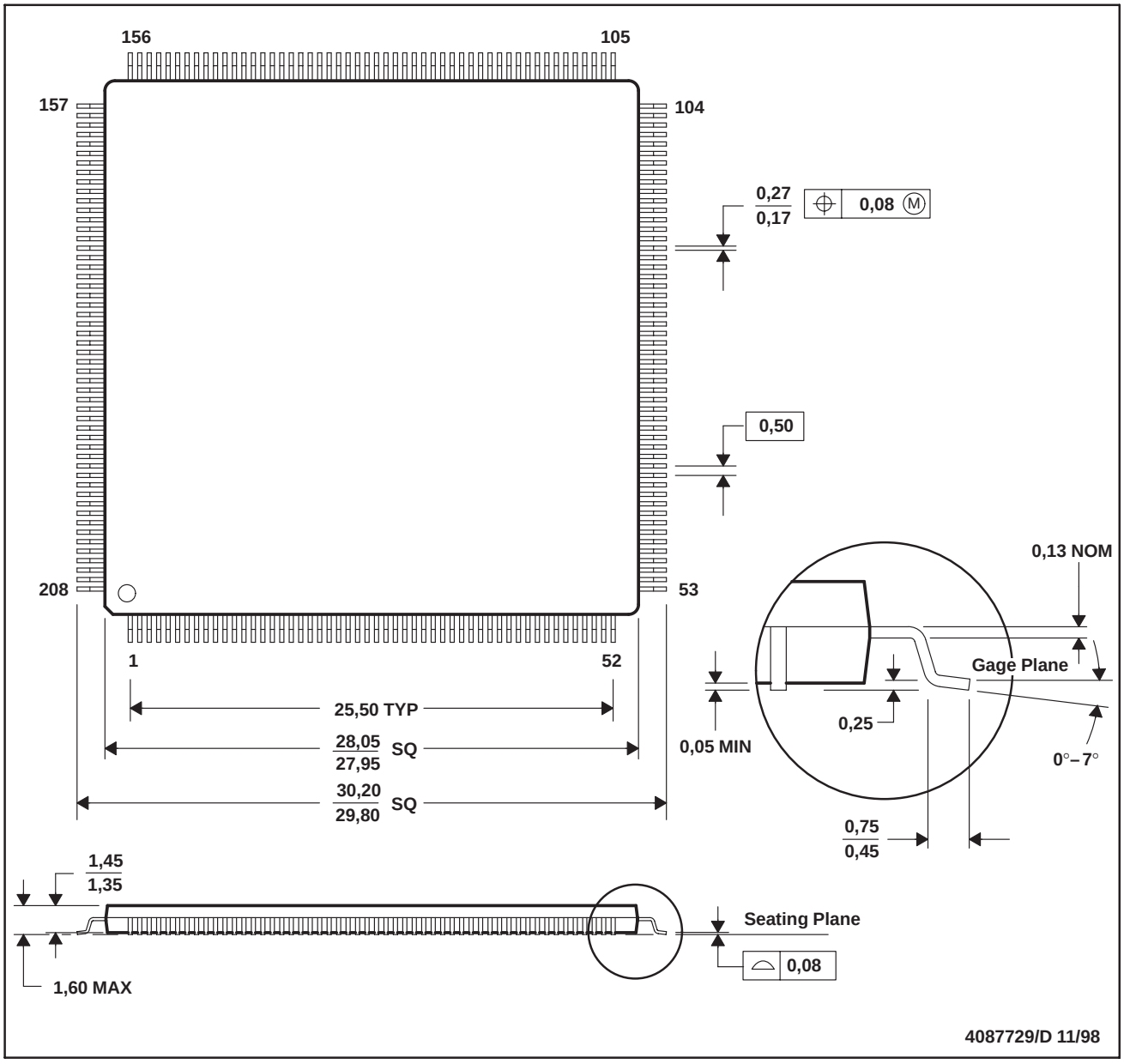
GHK (S-PBGA-N209) **PLASTIC BALL GRID ARRAY**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. MicroStar BGA™ configuration

MicroStar BGA is a trademark of Texas Instruments Incorporated.

PDV (S-PQFP-G208) **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

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.