PCI 6140 (HB1)
PCI-to-PCI Bridge
Data Book

PCI 6140 (HB1)
PCI-to-PCI Bridge
Data Book

Version 2.0

May 2003

Website: http://www.plxtech.com
Technical Support: http://www.plxtech.com/support
Phone: 408 774-9060
800 759-3735
Fax: 408 774-2169

PLX Part Number: PCI 6140-AA33PC; Former HiNT Part Number: HB1

Order Number: 6140-SIL-DB-P1-2.0

Printed in the USA, May 2003

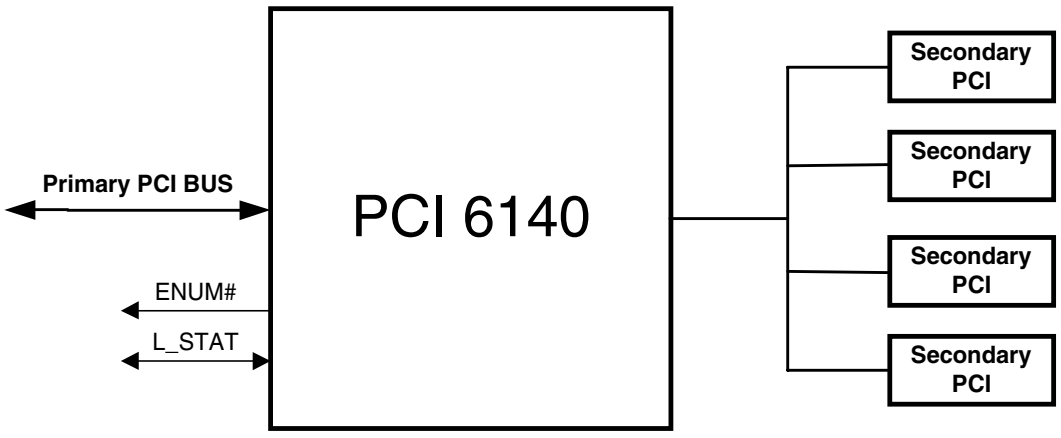
32-Bit, 33 MHz PCI-to-PCI Bridge

The world's first, Patent Pending PCI 6140 provides very low cost, PCI-to-PCI bridging functions. It is optimized to provide PCI masters to achieve optional **Zero Clock latency** in bursting data through the PCI 6140 PCI-to-PCI bridge. It supports up to 4 Secondary PCI master devices.

Optimized for the following applications:

- *PCI Latency Sensitive Systems*
- *PCI-Retry Penalty Sensitive Systems*
- *PCI Slave Access Intensive bridging functions*
- *High Performance and Low cost PCI-to-PCI bridging functions*
- *Low Power and PCI ClockRun support*
- *Compact PCI with Hot Swap*

- 3.3V, 33 MHz, PCI 2.1 and Compact PCI Hot Swap friendly features
- PC99 Power Management D3 Cold Wakeup Capable*
- Very efficient, low power, low cost and easy to use
- PCI ClockRun support.
- Optional Zero clock latency when bursting data across PCI 6140 to preserve maximum data rate
- PCI compatible cycle completion without PCI Retry penalty of a traditional PCI bridge
- Legacy VGA and Audio IO address support
- Provides arbitration support for 4 bus masters on secondary interface
- Supports PCI Type 1 to Type 0 and Type 1 configuration command conversion
- Supports 1-Clock Latency Mode
- Synchronous Primary and Secondary Ports
- 128-pin PQFP package



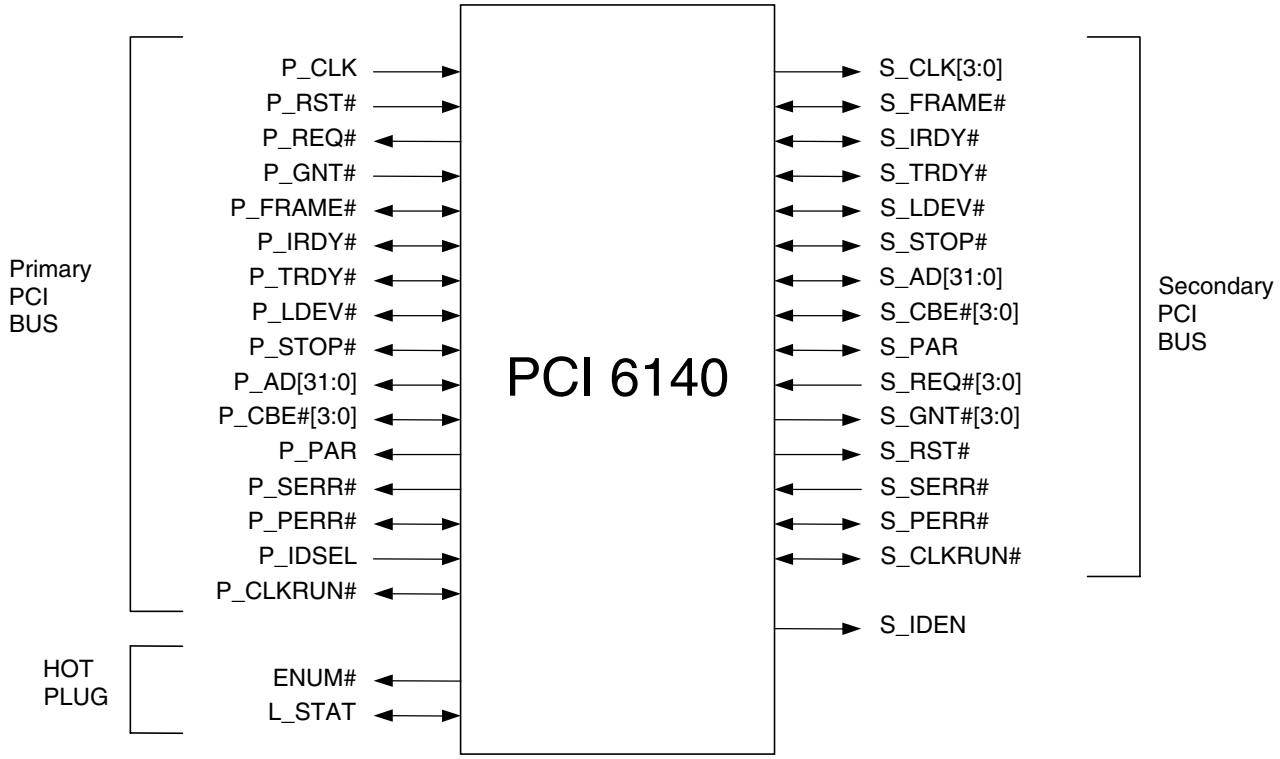
* Supported only in latest revision of PCI 6140, PCI register 82h bits 15-11 is set to 11110b, indicating PME# assertion possible during D1 state.

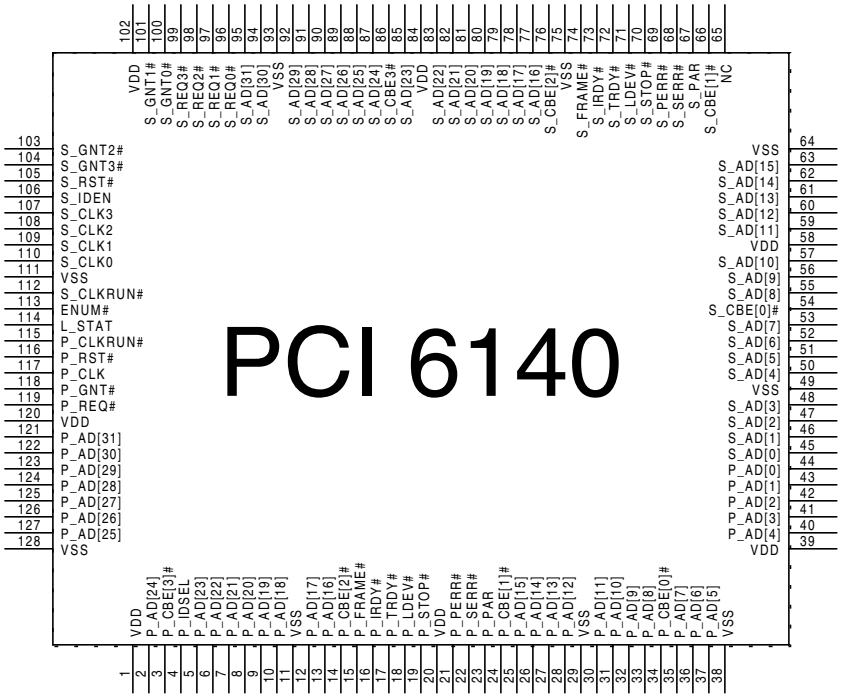
History

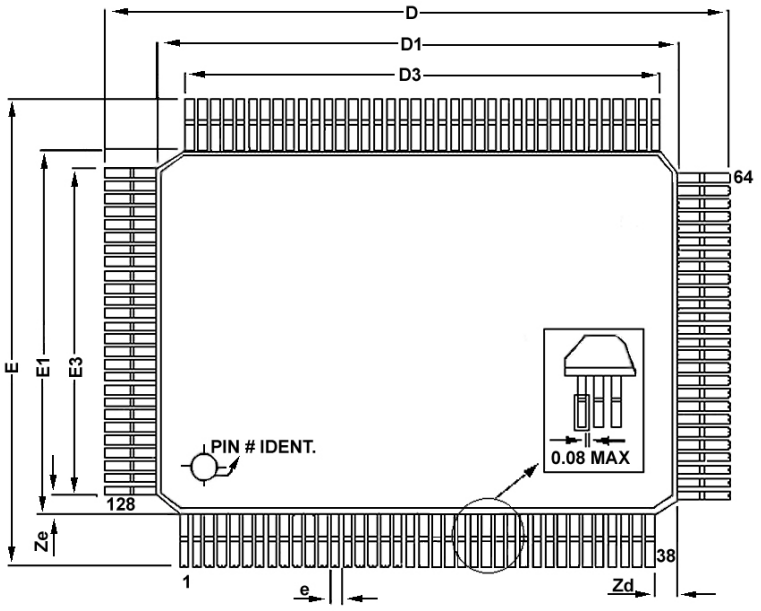
Rev	Date	Description	Eng Chk	Mkt Chk
Rev 2.0	5/23/03	This release reflects PLX part numbering. • Changed S_IDEN signal to reserved • Added operating ambient temperature information		

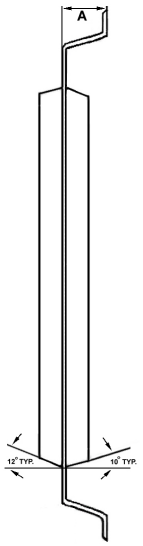
Contents

1 Pin Diagram and Package Outline



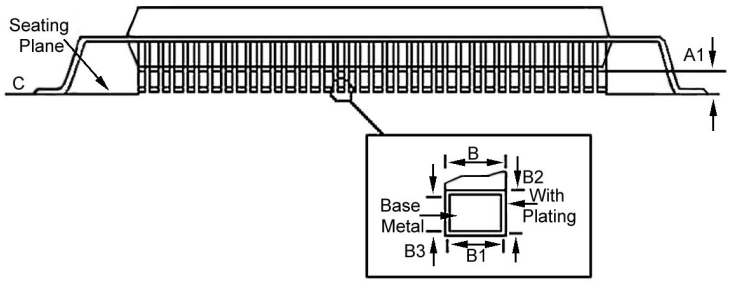


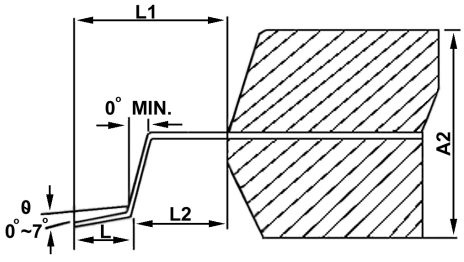




Notes:

1. Controlling dimensions are in millimeters (mm)
2. Dimension D1/E1 do not include mold protrusion





Symbol	Min	Max
A	-	3.4
A1	0.25	-
A2	2.60	3.00
B	0.17	0.27
B1	0.17	0.23
B2	0.13	0.23
B3	0.13	0.17
C	0.076	-
D	23.00	23.40
D1	19.90	20.10
D3	18.50 REF.	-
E	17.00	17.40
E1	13.90	14.10
E3	12.50 REF.	-
e	0.50 BSC	-
L	0.65	0.95
L1	1.60 REF	-
L2	0.4	-
Zd	0.75 REF	-
Ze	0.75 REF	-

2 Pin Description

SIGNAL	PIN	I/O	DESCRIPTION
P_CLK	117	I	PCI system clock
P_RST#	116	I	PCI system reset
P_REQ#	119	O	PCI bus request
P_GNT#	118	I	PCI bus grant
P_FRAME#	15	I/O	PCI FRAME, input during slave, output during master.
P_IRDY#	16	I/O	PCI IRDY, input during slave, output during master.
P_TRDY#	17	I/O	PCI TRDY, output during slave, input during master.
P_LDEV#	18	I/O	PCI LDEV, output during slave, input during master.
P_STOP#	19	I/O	PCI STOP, output during slave, input during master.
P_IDSEL	4	I	PCI IDSEL signal
P_AD[31:0]	121,122, 123,124, 125,126, 127,2,5, 6,7,8,9, 10,12,13, 25,26,27, 28,30,31, 32,33,35, 36,37,40, 41,42,43, 44	I/O	PCI address/data. Slave mode: output only during data read phase. Master mode: output during address phase and data write phase.
P_CBE#[3:0]	3,14,24, 34	I/O	PCI command/byte-enable, input during slave, output during master.
P_PAR	23	O	PCI parity
P_PERR#	21	I/O	PCI parity error
P_SERR#	22	O	PCI system error
P_CLKRUN#	115	I/O	Primary PCI bus clock run. Used by the central resource to stop the PCI clock or to slow it down
S_CLK[3:0]	110,109, 108,107	O	Secondary PCI clock
S_IDEN	106	O	Reserved. Must be pulled high.
S_RST#	105	O	Secondary PCI reset
S_FRAME#	74	I/O	Secondary PCI FRAME
S_IRDY#	73	I/O	Secondary PCI IRDY
S_TRDY#	72	I/O	Secondary PCI TRDY
S_LDEV#	71	I/O	Secondary PCI LDEV
S_STOP#	70	I/O	Secondary PCI STOP

S_AD[31:0]	95,94,92, 91,90,89, 88,87,85, 83,82,81, 80,79,78, 77,63,62, 61,60,59, 57,56,55, 53,52,51, 50,48,47, 46,45	I/O	Secondary PCI address/data
S_CBE#[3:0]	86,76,66, 54	I/O	Secondary PCI command/byte-enable
S_PAR	67	O	Secondary PCI parity
S_SERR#	68	I	Secondary PCI system error
S_PERR#	69	I/O	Secondary PCI parity error
S_REQ[3:0]#	99,98,97, 96	I	Secondary PCI bus request
S_GNT[3:0]#	104,103, 101,100	O	Secondary PCI bus grant
S_CLKRUN#	112	I/O	Secondary PCI bus clock run. Drive high to stop or slow down secondary PCI clock, driven by secondary PCI device to keep clock running
ENUM#	113	O	Hot Swap Interrupt
L_STAT	114	I/O	Hot Swap LED
NC	65		No connect pin
VSS	11,29,38, 49,64,75, 93,111, 128	GND	Ground Pins
VDD	1,20,39, 58, 84, 102,120	PWR	Power Pins

3 PCI 6140 Configuration Registers

The following PCI registers are supported by PCI 6140. All registers, except for register C0h follow the standard PCI register definition.

3.1 Configuration Space Address Map

31-24	23-16	15-8	7-0	Address
Device ID		**Vendor ID**		00h
Status		**Command**		04h
Class Code			**Revision ID**	08h
Reserved	**Header Type**	**Latency Timer**	**Cache Line Size**	0Ch
Reserved				10h
Reserved				14h
Secondary Latency Timer	**Subordinate Bus Number**	**Secondary Bus Number**	Reserved	18h
Secondary Status		**I/O Limit**	**I/O Base**	1Ch
Memory Limit		**Memory Base**		20h
Prefetchable Memory Limit		**Prefetchable Memory Base**		24h
Reserved				28h-33h
Reserved			**Capability Pointer = 80**	34h
Expansion ROM Base Address				38h
Bridge Control		**Interrupt Pin = 00**	**Interrupt Line = 00**	3ch
Subsystem ID = 0000		**Subsystem Vendor ID = 0000**		40h
Reserved				44h-6Bh
CLKRUN control	**Reserved**			6Ch
Reserved				70h-7Fh
PMC = F601		**Next Item Ptr = 90**	**Capability ID = 01**	80h
Data = 00	**PMCSR Bridge Support**	**PMCSR = 0000**		84h

Reserved				88h-8Fh
Reserved	**HSCSR = 00**	**Next Item Ptr = 00**	**Capability ID = 06**	90h
Reserved				94h-BFh
Reserved			**PCI 6140 control**	C0h
Reserved				C4h-FFh

3.2 Configuration Register Description

The following subsection describes the configuration registers of PCI 6140.

Table 3-1 Register Access

Abbreviation	Definition
RO	Read only. Writes have no effect
R/W	Read/Write
R/WC	Read. Write 1 to clear

3.2.1 Vendor ID Register (RO) (offset 00h)

Hardwired to 3388(h).

3.2.2 Device ID Register (RO) (offset 02h)

Hardwired to 0021(h)

3.2.3 Command Register (offset 04h)

Bit	Function	Type	Description
0	I/O Space Enable	R/W	Controls the bridge's response to I/O accesses on the primary interface. **0=ignore I/O transaction** 1=enable response to I/O transaction Reset to 0.
1	Memory Space Enable	R/W	Controls the bridge's response to memory accesses on the primary interface. **0=ignore all memory transaction** 1=enable response to memory transaction Reset to 0.
2	Bus Master Enable	R/W	Controls the bridge's ability to operate as a master on the primary interface. **0=do not initiate transaction on the primary interface and disable response to memory or I/O transactions on secondary interface** 1=enable the bridge to operate as a master on the primary interface Reset to 0.
3	Special Cycle Enable	R/O	No special cycle implementation (**set to '0'**).
4	Memory Write and Invalidate Enable	R/O	Memory write and invalidate not supported (**set to '0'**).
5	Reserved	R/O	Reserved. Reset to 0
6	Parity Error Enable	R/W	Controls the bridge's response to parity errors. **0=ignore any parity errors** 1=normal parity checking performed Reset to 0.
7	Wait Cycle Control	R/O	No data stepping supported (**set to '0'**).

8	Primary SERR# Enable	R/W	SERR# enable 0 = Disable SERR# driver on primary interface 1 = Enable the SERR# driver
9-15	reserved	R/O	Reserved. Reset to 0.

3.2.4 Primary Status Register (offset 06h)

Bit	Function	Type	Description
0-3	reserved	R/O	Reserved (**set to '0's**).
4	Capabilities List	R/O	Set to '1' indicating the presence of additional PCI Capabilities such as the power management and hot swap functions
5	33 MHz	R/O	33 MHz maximum frequency (**set to '0'**).
6	UDF	R/O	No User-Definable Features (**set to '0'**).
7	Fast Back to Back Capable	R/O	Fast back-to-back write capable on primary side (**set to '1'**). *NOTE: Fast back-to-back mode is always enabled, even though this bit actually returns '0' on reads.*
8	Data Parity Error Detected	R/WC	Reset to 0.
9-10	LDEV timing	R/O	LDEV timing (**default to '10'**).
11	Signaled Target Abort	R/WC	Should be set (by a target device) whenever a Target Abort cycle occurs. Reset to 0.
12	Received Target Abort	R/WC	Set to '1' (by a master device) when transactions are terminated with Target Abort. Reset to 0.
13	Received Master Abort	R/WC	Set to '1' (by a master) when transactions are terminated with Master Abort. Reset to 0.
14	Signaled System Error	R/WC	Set high to indicate SERR_L assertion. Reset to 0.
15	Detected Parity Error	R/WC	Set high to indicate data parity error. Reset to 0.

3.2.5 Revision ID Register (RO) (offset 08h)

Hardwired to 10h.

3.2.6 Class Code Register (RO) (offset 09h)

Hardwired to 060400h.

3.2.7 Cache Line Size Register (R/W) (offset 0Ch)

Bit 5-1: Used to set the memory read line or memory read multiple size in dword unit. Must be a multiple of 2.

3.2.8 Latency Timer (R/W) (offset 0Dh)

Bit 7-3: Used to satisfy PCI specifications.

3.2.9 Header Type Register (RO) (offset 0Eh)

Hardwired to 01h.

3.2.10 Secondary Bus Number Register (R/W) (offset 19h)

Programmed with the number of the PCI bus to which the secondary bridge interface is connected. This value is set with configuration software. Reset to 0.

3.2.11 Subordinate Bus Number Register (R/W) (offset 1Ah)

Programmed with the number of the PCI bus with the highest number that is subordinate to the bridge. This value is set with configuration software. Reset to 0.

3.2.12 Secondary Latency Timer Register (R/W) (offset 1Bh)

Bit 7-3: Used to satisfy PCI specifications.

3.2.13 I/O Base Register (R/W) (offset 1Ch)

This register defines the bottom address of the I/O address range for the bridge. The upper four bits define the bottom address range used by the chip to determine when to forward I/O transactions from one interface to the other. These 4 bits correspond to address bits <15:12> and are writeable. The upper 16 bits corresponding to address bits <31:16> are defined in the I/O base address upper 16 bits register. The address bits <11:0> are assumed to be 000h. The lower four bits (3:0) of this register set to ‘0001’ (read-only) to indicate 32-bit I/O addressing. Reset to 0.

3.2.14 I/O Limit Register (R/W) (offset 1Dh)

This register defines the top address of the I/O address range for the bridge. The upper four bits define the top address range used by the chip to determine when to forward I/O transactions from one interface to the other. These 4 bits correspond to address bits <15:12> and are writeable. The upper 16 bits corresponding to address bits <31:16> are defined in the I/O limit address upper 16 bits register. The address bits <11:0> are assumed to be FFFh. The lower four bits (3:0) of this register set to '0001' (read-only) to indicate 32-bit I/O addressing. Reset to 0.

3.2.15 Secondary Status Register (offset 1Eh)

Bit	Function	Type	Description
0-4	reserved	R/O	Reserved (**set to '0's**).
5	33 MHz	R/O	33 MHz maximum frequency (**set to '0'**).
6	UDF	R/O	No User-Definable Features (**set to '0'**).
7	Fast Back to Back Capable	R/O	Fast back-to-back write capable on secondary port (**set to '1'**). *NOTE: Fast back-to-back mode is always enabled, even though this bit actually returns '0' on reads.*
8	Data Parity Error Detected	R/WC	Reset to 0.
9-10	LDEV timing	R/O	Medium LDEV timing (**set to '01'**)
11	Signaled Target Abort	R/WC	Should be set (by a target device) whenever a Target Abort cycle occurs. Should be '0' after reset. Reset to 0.
12	Received Target Abort	R/WC	Set to '1' (by a master device) when transactions are terminated with Target Abort. Reset to 0.
13	Received Master Abort	R/WC	Set to '1' (by a master) when transactions are terminated with Master Abort. Reset to 0.
14	Received System Error	R/WC	Set high to indicate SERR_L assertion. Reset to 0.
15	Detected Parity Error	R/WC	Set high to indicate data parity error. Reset to 0.

3.2.16 Memory Base Register (R/W) (offset 20h)

This register defines the base address of the memory-mapped address range for forwarding the cycle through the bridge. The upper twelve bits corresponding to address bits <31:20> are writeable. The lower 20 address bits (19:0) are assumed to be 00000h. The 12 bits are reset to 0. The lower 4 bits are read only and set to 0.

3.2.17 Memory Limit Register (R/W) (offset 22h)

This register defines the upper limit address of the memory-mapped address range for forwarding the cycle through the bridge. The upper twelve bits corresponding to address bits <31:20> are writeable. The 12 bits are reset to 0. The lower 4 bits are read only and is set to 0. The lower 20 address bits (19:0) are assumed to be FFFFFh. Reset to 0.

3.2.18 Prefetchable Memory Base Register (R/W) (offset 24h)

This register defines the base address of the prefetchable memory-mapped address range for forwarding the cycle through the bridge. The upper twelve bits corresponding to address bits <31:20> are writeable. The 12 bits are reset to 0. The lower 4 bits are read only and is set to 0. The lower 20 address bits (19:0) are assumed to be 00000h. Reset to 0.

3.2.19 Prefetchable Memory Limit Register (R/W) (offset 26h)

This register defines the upper limit address of the memory-mapped address range for forwarding the cycle through the bridge. The upper twelve bits correspond to address bits <31:20> are writeable. The 12 bits are reset to 0. The lower 4 bits are read only and is set to 0. The lower 20 address bits (19:0) are assumed to be FFFFFh. Reset to 0.

3.2.20 Capability Pointer(R) (offset 34h)

This pointer points to the PCI power management registers

3.2.21 Expansion ROM Base Address(R) (offset 38h)

PCI 6140 does not implement the expansion ROM remapping feature. Register returns all 0s when read.

3.2.22 Interrupt Line Register(R/W) (offset 3Ch)

PCI 6140 does not implement an interrupt signal pin, thus register defaults to FFh

3.2.23 Interrupt Pin(R) (offset 3Dh)

PCI 6140 does not implement any interrupt pins, so this register returns 0s.

3.2.24 Bridge Control (offset 3Eh)

Bit	Function	Type	Description
0	Parity Error Response Enable	R/W	Specify bridge's response to parity errors on the secondary interface. 0 = ignore address and parity errors 1 = enable parity error reporting
1	SERR Enable	R/W	Specify forwarding of secondary interface SERR assertions to the primary interface. 0 = SERR disabled 1 = SERR enabled
2	ISA Enable	R/W	Controls forwarding of ISA IO transactions from the primary to the secondary. 0 = forward all IO addresses in the address range defined by IO base and IO limit registers. 1 = Block forwarding of ISA IO addresses
3	VGA Enable	R/W	Controls positive decoding and forwarding of VGA-compatible memory addresses. 0 = do not forward VGA-compatible memory and IO addresses from the primary to the secondary interface. 1 = forward VGA addresses
4	Reserved	R	Reserved
5	Master Abort Mode	R/W	Specifies how the bridge responds to a master abort that occurs on either interface when the bridge is the master 0 = do not report master aborts 1 = report master aborts by signaling target abort if possible or SERR if enabled in bit 1 of this register
6	Secondary Bus Reset	R/W	Controls assertion of S_RST# 0 = do not force the assertion of S_RST# 1 = force the assertion of S_RST#
7	Fast Back to Back	R	The bridge always generates fast back to back transactions to secondary devices.
8-15	Reserved	R	Reserved

3.2.25 Subsystem Vendor ID(R/W) (offset 40h)

An add-in card manufacturer may write to this register via a device driver for identification purposes.

3.2.26 Subsystem ID (R/W) (offset 42h)

An add-in card manufacturer may write to this register via a device driver for identification purposes.

3.2.27 Secondary Clock Disable Register (R/W) (offset 6Ch)

Bit	Function	Type	Description
0	Disable S_CLK0	R/W	Disable secondary clock 0 0 = enable clock 1 = disable clock
1	Disable S_CLK0	R/W	Disable secondary clock 1 0 = enable clock 1 = disable clock
2	Disable S_CLK0	R/W	Disable secondary clock 2 0 = enable clock 1 = disable clock
3	Disable S_CLK0	R/W	Disable secondary clock 3 0 = enable clock 1 = disable clock
4-7	Reserved	R	Reserved

3.2.28 Clock run control Register (R/W) (offset 6Fh)

Bit	Function	Type	Description
0	Secondary clock status	R	Secondary clock status 0 = secondary clock is running 1 = secondary clock is stopped
1	Secondary CLKRUN# enable	R/W	Secondary clock run enable 0 = disable secondary CLKRUN# 1 = enable secondary CLKRUN# (default)
2	Primary keep clock running	R/W	Primary keep clock run 0 = Allow primary clock to stop when secondary clock is stopped 1 = Keep primary clock running regardless of secondary clock
3	Primary CLKRUN# enable	R/W	Primary clock run enable 0 = disable primary CLKRUN# 1 = enable primary CLKRUN# (default)
4	CLKRUN# mode	R/W	Secondary clock run mode 0 = Stop secondary clock when requested by primary clock source 1 = Stop secondary clock when secondary bus is idle and no cycle from primary
5-7	Reserved	R	Reserved

3.2.29 Capability Identifier (R) (offset 80h)

This register is set to 01h to indicate power management interface registers.

3.2.30 Next Item Pointer (R) (offset 81h)

Set to 90h. This field provides an offset into the function's PCI Configuration Space pointing to the location of next item in the function's capability list. In PCI 6140, this points to the hot swap registers.

3.2.31 Power Management Capabilities(R) (offset 82h)

Bit	Function	Type	Description
0-2	Version	R	This register is set to 001b, indicating that this function complies with Rev 1.0 of the PCI Power Management Interface Specification
3	PME Clock	R	This bit is a '0', indicating that the PCI 6140 does not support PME# signaling.
4	Auxiliary Power Source	R	This bit is set to '0' since PCI 6140 does not support PME# signaling
5	DSI	R	Device Specific Initialization . Returns '0' indicating that PCI 6140 does not need special initialization
6-8	Reserved	R	Reserved
9	D1 Support	R	Returns '1' indicating that PCI 6140 supports the D1 device power state
10	D2 Support	R	Returns '1' indicating that PCI 6140 supports the D2 device power state
11-15	PME Support	R	Set to '11110b' indicating that PME# can be asserted from D1,D2,D3hot and D3cold states. *This is true for the latest revision 10. Earlier revisions have this set to 000000b.*

3.2.32 Power Management Control/ Status(R/W) (offset 84h)

Bit	Function	Type	Description
0-1	Power State	R/W	This 2-bit field is used both to determine the current power state of a function and to set the function into a new power state. The definition of the field values is given below. 00b - D0 01b - D1 10b - D2 11b - D3hot
2-7	Reserved	R	Reserved
8	PME Enable	R	This bit is set to '0' since PCI 6140 does not support PME# signaling
9-12	Data Select	R	This field returns '0000b' indicating PCI 6140 does not return any dynamic data
13-14	Data Scale	R	Returns '00b' when read. PCI 6140 does not return any dynamic data.
15	PME Status	R	This bit is set to '0' since PCI 6140 does not support PME# signaling

3.2.33 PMCSR Bridge Support(R) (offset 86h)

Bit	Function	Type	Description
0-5	Reserved	R	Reserved
6	B2/B3 Support for D3hot	R	This bit returns a '1' when read indicating that when the PCI 6140 is programmed to D3hot state the secondary bus's clock is stopped.
7	Bus Power Control Enable	R	Returns '1' indicating that the power management state of the secondary bus follows that of the PCI 6140 with one exception , D3hot state.

3.2.34 Capability Identifier (R) (offset 90h)

This register is set to 06h to indicate power management interface registers.

3.2.35 Next Item Pointer (R) (offset 91h)

Set to 00h. Indicates the end of the capabilities list.

3.2.36 Hot Swap Register(R/W) (offset 92h)

Bit	Function	Type	Description
0	Reserved	R	Reserved
1	ENUM# Mask Status	R/W	Enables or disables ENUM# assertion 0 = enable ENUM# signal 1 = mask off ENUM# signal
2	Reserved	R	Reserved
3	LED status	R	Indicates if LED is on or off 0 = LED is off 1 = LED is on
4-5	Reserved	R	Reserved
6	Extraction State	R	Indicates assertion of ENUM# due to the device being extracted 0 = ENUM# asserted 1 = ENUM# not asserted
7	Insertion State	R	Indicates assertion of ENUM# due to the device being inserted 0 = ENUM# asserted 1 = ENUM# not asserted

3.2.37 PCI 6140 mode register (offset C0h)

Bit	Function	Type	Description
0	Primary to Secondary transaction delay	R/W	Specify delay for transactions going from primary to secondary PCI interface 0 = delay Primary bus to Secondary bus transfer by 1 P_CLK 1 = no delay
1	P_CBE[3:0] active	R/W	Specify decoding timing 0 = use the correct P_CBE[3:0] 1 = force P_CBE[3:0] active only all remaining burst read cycle.
2	Transparent P_GNT	R/W	0 = transparent P_GNT 1 = use 1-clock delayed P_GNT
3-4	Reserved	R/W	Reserved
5	IRDY Mode	R/W	0 = assert IRDY only after IRDY has been active on the master side 1 = assert IRDY immediately during memory-read-line, or memory-read-multiple after IRDY has been active
6	Memory Read Termination	R/W	Always terminate memory read line or memory read multiple on the line boundary
7	Fast Read Enable	R/W	Allowing fast read during burst-read.
8	Legacy Audio IO Address Enable	R/W	0 = disable, 1 = enable When enabled, IO address 200-207 (Game), 388-38B (FM), 220-233 (Audio), 330-331 (MIDI) will be claimed by PCI 6140 and such cycles will be passed onto the secondary PCI bus.

4 PCI Bus Operation (Transparent Mode)

This chapter presents detailed information about PCI transactions, transaction forwarding across PCI 6140, and transaction termination.

PCI 6140 provides a simple, but complete PCI-to-PCI bridge capability, allowing PCI master and slave on its either side. It passes control and data between primary and secondary bus to guarantee complete visibility from either side. PCI 6140 is designed to behave like an intelligent buffer.

PCI 6140 achieves its zero wait state bridging function by controlling the direction of control and data. It divides control and data into 3 signal groups; the FRAME#/IRDY#/CBE#, LDEV#/TRDY#/STOP#, and AD signal groups.

Direction of FRAME#/IRDY#/CBE# is determined by P_GNT#. If P_GNT# is asserted at the time FRAME# is active, direction of LDEV#/TRDY#/STOP# is determined by address decode, as described in the address decode section. Direction of AD[31:0] is determined by the combination of address decode and location of slave.

4.1 Types of Transactions

This section provides a summary of PCI transactions performed by PCI 6140. Table 4–1 lists the command code and name of each PCI transaction. The Master and Target columns indicate support for each transaction when PCI 6140 initiates transactions as a master, on the primary bus and on the secondary bus, and when PCI 6140 responds to transactions as a target, on the primary bus and on the secondary bus.

Table 4–1 PCI Transactions

Type of transaction		Initiates as Master		Responds as Target	
		Primary	Secondary	Primary	Secondary
0000	Interrupt acknowledge	N	N	N	N
0001	Special cycle	Y	Y	N	N
0010	I/O read	Y	Y	Y	Y
0011	I/O write	Y	Y	Y	Y
0100	Reserved	N	N	N	N
0101	Reserved	N	N	N	N
0110	Memory read	Y	Y	Y	Y
0111	Memory write	Y	Y	Y	Y
1000	Reserved	N	N	N	N
1001	Reserved	N	N	N	N
1010	Configuration read	N	Y	Y	N
1011	Configuration write	Type 1	Y	Y	Type 1
1100	Memory read multiple	Y	Y	Y	Y
1101	Dual address cycle	N	N	N	N
1110	Memory read line	Y	Y	Y	Y
1111	Memory write and invalidate	Y	Y	Y	Y

As indicated in Table 4–1, the following PCI commands are not supported by PCI 6140 :

- PCI 6140 never initiates a PCI transaction with a reserved command code and, as a target, PCI 6140 ignores reserved command codes.
- PCI 6140 never initiates an interrupt acknowledge transaction and, as a target, PCI 6140 ignores interrupt acknowledge transactions. Interrupt acknowledge transactions are expected to reside entirely on the primary PCI bus closest to the host bridge.
- PCI 6140 does not respond to special cycle transactions. To generate special cycle transactions on other PCI buses, either upstream or downstream, a Type 1 configuration command must be used.

- PCI 6140 does not generate Type 0 configuration transactions on the primary interface, nor does it respond to Type 0 configuration transactions on the secondary PCI interface. The PCI-to-PCI Bridge Architecture Specification does not support configuration from the secondary bus.
- PCI 6140 does not respond to nor initiate DAC cycle transactions.

4.2 Address Phase

A 32-bit address uses a single address phase. This address is driven on AD<31:0>, and the bus command is driven on P_CBE[3:0]

4.3 Device Select (LDEV#) Generation

PCI 6140 always performs positive address decoding when accepting transactions on either the primary or secondary buses. PCI 6140 never subtractively decodes. Medium LDEV# timing is used on both interfaces.

4.4 Data Phase

The address phase or phases of a PCI transaction are followed by one or more data phases. A data phase is completed when IRDY# and either TRDY# or STOP# are asserted. A transfer of data occurs only when both IRDY# and TRDY# are asserted during the same PCI clock cycle. The last data phase of a transaction is indicated when FRAME# is de-asserted and both TRDY# and IRDY# are asserted, or when IRDY# and STOP# are asserted.

4.5 Write Transactions

Acting as PCI bus extender, PCI 6140 responds differently according to the address and initiator.

Case 1: Primary master access device on primary bus
PCI 6140 will forward all PCI signals from primary to secondary so that any device there can track the PCI bus.

Case 2: Primary master access device on secondary bus
PCI 6140 will forward address, command, data, byte enable, P_IRDY# to secondary while forwarding S_LDEV#, S_TRDY# and S_STOP# to primary bus.

Case 3: Secondary master access device on secondary bus
PCI 6140 will forward all PCI signals from secondary to primary so that any device there can track the PCI bus.

Case 4: Secondary master access device on primary bus
PCI 6140 will forward address, command, data, byte enable, S_IRDY# to primary while forwarding P_LDEV#, P_TRDY# and P_STOP# to secondary.

There is no buffer inside PCI 6140 for write.

4.6 Read Transactions

PCI 6140 responds according to the address and initiator of the read command.

Case 1: Primary master access device on primary bus
PCI 6140 will forward all PCI signals from primary to secondary so that any device there can track the PCI bus.

Case 2: Primary master access device on secondary bus
PCI 6140 will forward address, command, byte enable, P_IRDY# to secondary while forwarding data, S_LDEV#, S_TRDY# and S_STOP# to primary bus.

Case 3: Secondary master access device on secondary bus
PCI 6140 will forward all PCI signals from secondary to primary so that any device there can track the PCI bus.

Case 4: Secondary master access device on primary bus
PCI 6140 will forward address, command, byte enable, S_IRDY# to primary while forwarding data, P_LDEV#, P_TRDY# and P_STOP# to secondary.

There is no buffer inside PCI 6140 for read.

4.7 Configuration Transactions

Configuration transactions are used to initialize a PCI system. Every PCI device has a configuration space that is accessed by configuration commands. All registers are accessible in configuration space only.

In addition to accepting configuration transactions for initialization of its own configuration space, PCI 6140 also forwards configuration transactions for device initialization in hierarchical PCI systems, as well as for special cycle generation.

To support hierarchical PCI bus systems, two types of configuration transactions are specified: Type 0 and Type 1.

Type 0 configuration transactions are issued when the intended target resides on the same PCI bus as the initiator. A Type 0 configuration transaction is identified by the configuration command and the Lowest 2 bits of the address set to 00b.

Type 1 configuration transactions are issued when the intended target resides on another PCI bus, or when a special cycle is to be generated on another PCI bus. A Type 1 configuration command is identified by the configuration command and the Lowest 2 address bits set to 01b.

The register number is found in both Type 0 and Type 1 formats and gives the Dword address of the configuration register to be accessed. The function number is also included in both Type 0 and Type 1 formats and indicates which function of a multifunction device is to be accessed. For single-function devices, this value is not decoded. Type 1 configuration transaction addresses also include a 5-bit field designating the device number that identifies the device on the target PCI bus that is to be accessed. In addition, the bus number in Type 1 transactions specifies the PCI bus to which the transaction is targeted.

4.7.1 Type 0 Access to PCI 6140

The configuration space is accessed by a Type 0 configuration transaction on the primary interface. The configuration space cannot be accessed from the secondary bus. PCI 6140 responds to a Type 0 configuration transaction by asserting P_LDEV# when the following conditions are met during the address phase:

- The bus command is a configuration read or configuration write transaction.
- low 2 address bits P_AD<1:0> must be 00b.
- Signal P_IDSEL must be asserted.
- The function code is 0.

PCI 6140 limits all configuration accesses to a single Dword data transfer and returns a target disconnect with the first data transfer if additional data phases are requested. Because read transactions to configuration space do not have side effects, all bytes in the requested Dword are returned, regardless of the value of the byte enable bits.

Type 0 configuration write and read transactions do not use data buffers; that is, these transactions are completed immediately.

PCI 6140 ignores all Type 0 transactions initiated on the secondary interface.

4.7.2 Type 1 to Type 0 Translation

Type 1 configuration transactions are used specifically for device configuration in a hierarchical PCI bus system. A PCI-to-PCI bridge is the only type of device that should respond to a Type 1 configuration command. Type 1 configuration commands are used when the configuration access is intended for a PCI device that resides on a PCI bus other than the one where the Type 1 transaction is generated.

PCI 6140 performs a Type 1 to Type 0 translation when the Type 1 transaction is generated on the primary bus and is intended for a device attached directly to the secondary bus. PCI 6140 must convert the configuration command to a Type 0 format so that the secondary bus device can respond to it. Type 1 to Type 0 translations are performed only in the downstream direction; that is, PCI 6140 generates a Type 0 transaction only on the secondary bus, and never on the primary bus.

PCI 6140 responds to a Type 1 configuration transaction and translates it into a Type 0 transaction on the secondary bus when the following conditions are met during the address phase:

- The low 2 address bits on P_AD<1:0> are 01b.
- The bus number in address field P_AD<23:16> is equal to the value in the secondary bus number register in configuration space.
- The bus command on P_CBE<3:0> is a configuration read or configuration write transaction.
-

When PCI 6140 translates the Type 1 transaction to a Type 0 transaction on the secondary interface, it performs the following translations to the address:

- Sets the low 2 address bits on S_AD<1:0> to 00b.
- Decodes the device number and drives the bit pattern specified in Table 4–6 on S_AD<31:16> for the purpose of asserting the device's IDSEL signal.
- Sets S_AD<15:11> to 0.
- Leaves unchanged the function number and register number fields.

PCI 6140 asserts a unique address line based on the device number. These address lines may be used as secondary bus IDSEL signals. The mapping of the address lines depends on the device number in the Type 1 address bits P_AD<15:11>. Table 4–6 presents the mapping that PCI 6140 uses.

Table 4–6 Device Number to IDSEL S_AD Pin Mapping

Device Number	P_AD<15:11>	Secondary IDSEL S_AD<31:16>	S_AD Bit
0h	00000	0000 0000 0000 0001	16
1h	00001	0000 0000 0000 0010	17
2h	00010	0000 0000 0000 0100	18
3h	00011	0000 0000 0000 1000	19
4h	00100	0000 0000 0001 0000	20
5h	00101	0000 0000 0010 0000	21
6h	0110	0000 0000 0100 0000	22
7h	00111	0000 0000 1000 0000	23
8h	01000	0000 0001 0000 0000	24
9h	01001	0000 0010 0000 0000	25
Ah	01010	0000 0100 0000 0000	26
Bh	01011	0000 1000 0000 0000	27
Ch	01100	0001 0000 0000 0000	28
Dh	01101	0010 0000 0000 0000	29
Eh	01110	0100 0000 0000 0000	30
Fh	01111	1000 0000 0000 0000	31
1Fh	11111	Generate special cycle (P_AD<7:2> = 00h) 0000 0000 0000 0000 (P_AD<7:2> != 00h)	-

PCI 6140 can assert up to 16 unique address lines to be used as IDSEL signals for up to 16 devices on the secondary bus, for device numbers ranging from 0 through 15. Because of electrical loading constraints of the PCI bus, more than 16 IDSEL signals should not be necessary. However, if device numbers greater than 15 are desired, some external method of generating IDSEL lines must be used, and no upper address bits are then asserted. The configuration transaction is still translated and passed from the primary bus to the secondary bus. If no IDSEL pin is asserted to a secondary device, the transaction ends in a master abort.

PCI 6140 forwards Type 1 to Type 0 configuration read or write transactions as delayed transactions. Type 1 to Type 0 configuration read or write transactions are limited to a single 32-bit data transfer.

4.7.3 Type 1 to Type 1 Forwarding

Type 1 to Type 1 transaction forwarding provides a hierarchical configuration mechanism when two or more levels of PCI-to-PCI bridges are used.

When PCI 6140 detects a Type 1 configuration transaction intended for a PCI bus downstream from the secondary bus, PCI 6140 forwards the transaction unchanged to the secondary bus. Ultimately, this transaction is translated to a Type 0 configuration command or to a special cycle transaction by a downstream PCI-to-PCI bridge. Downstream Type 1 to Type 1 forwarding occurs when the following conditions are met during the address phase:

- The low 2 address bits are equal to 01b.
- The bus number falls in the range defined by the lower limit (exclusive) in the secondary bus number register and the upper limit (inclusive) in the subordinate bus number register.
- The bus command is a configuration read or write transaction.

PCI 6140 also supports Type 1 to Type 1 forwarding of configuration write transactions upstream to support upstream special cycle generation. A Type 1 configuration command is forwarded upstream when the following conditions are met:

- The low 2 address bits are equal to 01b.
- The bus number falls outside the range defined by the lower limit (inclusive) in the secondary bus number register and the upper limit (inclusive) in the subordinate bus number register.
- The device number in address bits AD<15:11> is equal to 11111b.
- The function number in address bits AD<10:8> is equal to 111b.
- The bus command is a configuration write transaction.

PCI 6140 forwards Type 1 to Type 1 configuration write transactions as delayed transactions. Type 1 to Type 1 configuration write transactions are limited to a single data transfer.

4.7.4 Special Cycles

The Type 1 configuration mechanism is used to generate special cycle transactions in hierarchical PCI systems. Special cycle transactions are ignored by acting as a target and are not forwarded across the bridge. Special cycle transactions can be generated from Type 1 configuration write transactions in either the upstream or the downstream direction.

PCI 6140 initiates a special cycle on the target bus when a Type 1 configuration write transaction is detected on the initiating bus and the following conditions are met during the address phase:

- The low 2 address bits on AD<1:0> are equal to 01b.
- The device number in address bits AD<15:11> is equal to 11111b.
- The function number in address bits AD<10:8> is equal to 111b.
- The register number in address bits AD<7:2> is equal to 000000b.
- The bus number is equal to the value in the secondary bus number register in configuration space for downstream forwarding or equal to the value in the primary bus number register in configuration space for upstream forwarding.
- The bus command on P_CBE# is a configuration write command.

When PCI 6140 initiates the transaction on the target interface, the bus command is changed from configuration write to special cycle. The address and data are forwarded unchanged. Devices that use special cycles ignore the address and decode only the bus command. The data phase contains the special cycle message. The transaction is forwarded as a delayed transaction, but in this case the target response is not forwarded back (because special cycles result in a master abort). Once the transaction is completed on the target bus, through detection of the master abort condition, PCI 6140 responds with TRDY# to the next attempt of the configuration transaction from the initiator.

4.8 Transaction Termination

This section describes how PCI 6140 returns transaction termination conditions back to the initiator.

The initiator can terminate transactions with one of the following types of termination:

- Normal termination
  Normal termination occurs when the initiator de-asserts FRAME# at the beginning of the last data phase, and de-asserts IRDY# at the end of the last data phase in conjunction with either TRDY# or STOP# assertion from the target.

- Master abort
  A master abort occurs when no target response is detected. When the initiator does not detect a LDEV# from the target within five clock cycles after asserting FRAME#, the initiator terminates the transaction with a master abort. If FRAME# is still asserted, the initiator de-asserts FRAME# on the next cycle, and then de-asserts IRDY# on the following cycle. IRDY# must be asserted in the same cycle in which FRAME# de-asserts. If FRAME# is already de-asserted, IRDY# can be de-asserted on the next clock cycle following detection of the master abort condition.

The target can terminate transactions with one of the following types of termination:

- Normal termination—TRDY# and LDEV# asserted in conjunction with FRAME# de-asserted and IRDY# asserted.
- Target retry—STOP# and LDEV# asserted without TRDY# during the first data phase. No data transfers occur during the transaction. This transaction must be repeated.
- Target disconnect with data transfer—STOP# and LDEV# asserted with TRDY#. Signals that this is the last data transfer of the transaction.
- Target disconnect without data transfer—STOP# and LDEV# asserted without TRDY# after previous data transfers have been made. Indicates that no more data transfers will be made during this transaction.
- Target abort—STOP# asserted without LDEV# and without TRDY#.

Indicates that the target will never be able to complete this transaction. LDEV# must be asserted for at least one cycle during the transaction before the target abort is signaled.

4.8.1 Master Termination Initiated by PCI 6140

PCI 6140, as an initiator, uses normal termination if LDEV# is returned by the target within five clock cycles of PCI 6140's assertion of FRAME# on the target bus. PCI 6140 terminates a transaction when the target terminates the transaction with last data transfer, retry, disconnect, or target abort.

4.8.2 Master Abort Received by PCI 6140

If the initiator initiates a transaction on the target bus and does not detect LDEV# returned by the target within five clock cycles of PCI 6140's assertion of FRAME#, PCI 6140 terminates the transaction with a master abort. This sets the received master abort bit in the status register corresponding to the target bus.

For delayed read and write transactions, PCI 6140 is able to reflect the master abort condition back to the initiator. When PCI 6140 detects a master abort in response to a delayed transaction, and when the initiator repeats the transaction, PCI 6140 does not respond to the transaction with LDEV#. This passes the master abort condition back to the initiator.

Note
When PCI 6140 performs a Type 1 to special cycle translation, a master abort is the expected termination for the special cycle on the target bus. In this case, the master abort received bit is not set, and the Type 1 configuration transaction is disconnected after the first data phase.

4.8.3 Target Termination Received by PCI 6140

When PCI 6140 initiates a transaction on the target bus and the target responds with LDEV#, the target can end the transaction with one of the following types of termination:

- Normal termination (upon de-assertion of FRAME#)
- Target retry
- Target disconnect
- Target abort

PCI 6140 handles these terminations in different ways, depending on the type of transaction being performed.

4.8.3.1 Delayed Write Target Termination Response

When PCI 6140 initiates a delayed write transaction, the type of target termination received from the target can be passed back to the initiator. Table 4–7 shows the response to each type of target termination that occurs during a delayed write transaction.

PCI 6140 repeats a delayed write transaction until one of the following conditions is met:

- PCI 6140 completes at least one data transfer.
- PCI 6140 receives a master abort.
- PCI 6140 receives a target abort.

PCI 6140 makes 2^{24} write attempts resulting in a response of target retry.

Table 4–7 Response to Delayed Write Target Termination

Target Termination	Response
Normal	Return disconnect to initiator with first data transfer only if multiple data phases requested.
Target retry	Return target retry to initiator. Continue write attempts to target.
Target disconnect	Return disconnect to initiator with first data transfer only if multiple data phases requested.
Target abort	Return target abort to initiator. Set received target abort bit in target interface status register. Set signaled target abort bit in initiator interface status register.

4.8.3.2 Delayed Read Target Termination Response

When PCI 6140 initiates a delayed read transaction, the abnormal target responses can be passed back to the initiator. Other target responses depend on how much data the initiator requests. Table 4–9 shows the response to each type of target termination that occurs during a delayed read transaction.

Table 4–9 Response to Delayed Read Target Termination

Target termination	Response
Normal	If prefetchable, target disconnect only if initiator requests more data than read from target. If nonprefetchable, target disconnect on first data phase.
Target retry	Reinitiate read transaction to target
Target disconnect	If initiator requests more data than read from target, return target disconnect to initiator
Target abort	Return target abort to initiator. Set received target abort bit in the target interface status register. Set signaled target abort bit in the initiator interface status register.

PCI 6140 repeats a delayed read transaction until one of the following conditions is met:

- PCI 6140 completes at least one data transfer.
- PCI 6140 receives a master abort.
- PCI 6140 receives a target abort.

4.8.4 Target Termination Initiated by PCI 6140

PCI 6140 can return a target retry, target disconnect, or target abort to an initiator for reasons other than detection of that condition at the target interface.

4.8.4.1 Target Retry

PCI 6140 returns a target retry to the initiator when it cannot accept write data or return read data as a result of internal conditions. PCI 6140 returns a target retry to an initiator when any of the following conditions is met:

- For delayed write transactions:
  - The transaction is being entered into the delayed transaction queue.
  - The transaction has already been entered into the delayed transaction queue, but target response has not yet been received.
  - The delayed transaction queue is full, and the transaction cannot be queued.
  - A transaction with the same address and command has been queued.
  - Uses more than 16 clocks to accept this transaction.

- For delayed read transactions:
  - The transaction is being entered into the delayed transaction queue.
  - The read request has already been queued, but read data is not yet available.
  - The delayed transaction queue is full, and the transaction cannot be queued.
  - A delayed read request with the same address and bus command has already been queued.
  - Uses more than 16 clocks to accept this transaction.

When a target retry is returned to the initiator of a delayed transaction, the initiator must repeat the transaction with the same address and bus command as well as the data if this is a write transaction, within the time frame specified by the master timeout value; otherwise, the transaction is discarded from the buffer.

4.8.4.2 Target Disconnect

PCI 6140 returns a target disconnect to an initiator when the target returns target disconnect.

4.8.4.3 Target Abort

PCI 6140 returns a target abort to an initiator when one of the following conditions is met:

- PCI 6140 is returning a target abort from the intended target.
- PCI 6140 is unable to obtain delayed read data from the target or to deliver delayed write data to the target after 2• attempts.

When PCI 6140 returns a target abort to the initiator, it sets the signaled target abort bit in the status register corresponding to the initiator interface.

5 Address Decoding

PCI 6140 uses three address ranges that control I/O and memory transaction forwarding. These address ranges are defined by base and limit address registers in the configuration space. This chapter describes these address ranges, as well as ISA-mode and VGA-addressing support.

5.1 Address Ranges

PCI 6140 uses the following address ranges to determine which I/O and memory transactions are forwarded from the primary PCI bus to the secondary PCI bus, and from the secondary bus to the primary bus:

- Two 32-bit I/O address ranges

Transactions falling within these ranges are forwarded downstream from the primary PCI bus to the two secondary PCI buses. Transactions falling outside these ranges are forwarded upstream from the two secondary PCI buses to the primary PCI bus.

PCI 6140 uses a flat address space; that is, it does not perform any address translations. The address space has no ‘‘gaps’’—addresses that are not marked for downstream forwarding are always forwarded upstream.

5.2 I/O Address Decoding

PCI 6140 uses the following mechanisms that are defined in the configuration space to specify the I/O address space for downstream and upstream forwarding:

- I/O base and limit address registers
- The ISA enable bit
- The VGA mode bit
- The VGA snoop bit

This section provides information on the I/O address registers and ISA mode.

To enable downstream forwarding of I/O transactions, the I/O enable bit must be set in the command register in configuration space. If the I/O enable bit is not set, all I/O transactions initiated on the primary bus are ignored. To enable upstream forwarding of I/O transactions, the master enable bit must be set in the command register. If the master enable bit is not set, PCI 6140 ignores all I/O and memory transactions initiated on the secondary bus. Setting the master enable bit also Allows upstream forwarding of memory transactions.

Caution
If any configuration state affecting I/O transaction forwarding is changed by a configuration write operation on the primary bus at the same time that I/O transactions are ongoing on the secondary bus, the PCI 6140 response to the secondary bus I/O transactions is not predictable. Configure the I/O base and limit address registers, ISA enable bit, VGA mode bit, and VGA snoop bit before setting the I/O enable and master enable bits, and change them subsequently only when the primary and secondary PCI buses are idle.

5.2.1 I/O Base and Limit Address Registers

PCI 6140 implements one set of I/O base and limit address registers in configuration space that define an I/O address range per port downstream forwarding. PCI 6140 supports 32-bit I/O addressing, which Allows I/O addresses downstream of PCI 6140 to be mapped anywhere in a 4GB I/O address space.

I/O transactions with addresses that fall inside the range defined by the I/O base and limit registers are forwarded downstream from the primary PCI bus to the secondary PCI bus. I/O transactions with addresses that fall outside this range are forwarded upstream from the secondary PCI bus to the primary PCI bus.

The I/O range can be turned off by setting the I/O base address to a value greater than that of the I/O limit address. When the I/O range is turned off, all I/O transactions are forwarded upstream, and no I/O transactions are forwarded downstream.

The I/O range has a minimum granularity of 4KB and is aligned on a 4KB boundary. The maximum I/O range is 4GB in size.

The I/O base register consists of an 8-bit field at configuration address 1Ch, and a 16-bit field at address 30h. The top 4 bits of the 8-bit field define bits <15:12> of the I/O base address. The bottom 4 bits read only as 1h to indicate that PCI 6140 supports 32-bit I/O addressing. Bits <11:0> of the base address are assumed to be 0, which naturally aligns the base address to a 4KB boundary. The 16 bits contained in the I/O base upper 16 bits register at configuration offset 30h define AD<31:16> of the I/O base address. All 16 bits are read/write. After primary bus reset or chip reset, the value of the I/O base address is initialized to 0000 0000h.

The I/O limit register consists of an 8-bit field at configuration offset 1Dh and a 16-bit field at offset 32h. The top 4 bits of the 8-bit field define bits <15:12> of the I/O limit address. The bottom 4 bits read only as 1h to indicate that 32-bit I/O addressing is supported. Bits <11:0> of the limit address are assumed to be FFFh, which naturally aligns the limit address to the top of a 4KB I/O address block. The 16 bits contained in the I/O limit upper 16 bits register at configuration offset 32h define AD<31:16> of the I/O limit address. All 16 bits are read/write. After primary bus reset or chip reset, the value of the I/O limit address is reset to 0000 0FFFh.

Note
The initial states of the I/O base and I/O limit address registers define an I/O range of 0000 0000h to 0000 0FFFh, which is the bottom 4KB of I/O space. Write these registers with their appropriate values before setting either the I/O enable bit or the master enable bit in the command register in configuration space.

5.2.2 ISA Mode

PCI 6140 supports ISA mode by providing an ISA enable bit in the bridge control register in configuration space. ISA mode modifies the response of PCI 6140 inside the I/O address range in order to support mapping of I/O space in the presence of an ISA bus in the system. This bit only affects the response of PCI 6140 when the transaction falls inside the address range defined by the I/O base and limit address registers, and only when this address also falls inside the first 64KB of I/O space (address bits <31:16> are 0000h).

When the ISA enable bit is set, PCI 6140 does not forward downstream any I/O transactions addressing the top 768 bytes of each aligned 1KB block. Only those transactions addressing the bottom 256 bytes of an aligned 1KB block inside the base and limit I/O address range are forwarded downstream. Transactions above the 64KB I/O address boundary are forwarded as defined by the address range defined by the I/O base and limit registers.

Accordingly, if the ISA enable bit is set, PCI 6140 forwards upstream those I/O transactions addressing the top 768 bytes of each aligned 1KB block within the first 64KB of I/O space. The master enable bit in the command configuration register must also be set to enable upstream forwarding. All other I/O transactions initiated on the secondary bus are forwarded upstream only if they fall outside the I/O address range.

When the ISA enable bit is set, devices downstream of PCI 6140 can have I/O space mapped into the first 256 bytes of each 1KB chunk below the 64KB boundary, or anywhere in I/O space above the 64KB boundary.

5.3 Memory Address Decoding

PCI 6140 has three mechanisms for defining memory address ranges for forwarding of memory transactions:

- Memory-mapped I/O base and limit address registers
- Prefetchable memory base and limit address registers
- VGA mode

This section describes the first two mechanisms.

To enable downstream forwarding of memory transactions, the memory enable bit must be set in the command register in configuration space. To enable upstream forwarding of memory transactions, the master enable bit must be set in the command register. Setting the master enable bit also Allows upstream forwarding of I/O transactions.

Caution
If any configuration state affecting memory transaction forwarding is changed by a configuration write operation on the primary bus at the same time that memory transactions are ongoing on the secondary bus, response to the secondary bus memory transactions is not predictable. Configure the memory-mapped I/O base and limit address registers, prefetchable memory base and limit address registers, and VGA mode bit before setting the memory enable and master enable bits, and change them subsequently only when the primary and secondary PCI buses are idle.

5.3.1 Memory-Mapped I/O Base and Limit Address Registers

Memory-mapped I/O is also referred to as nonprefetchable memory. Memory addresses that cannot automatically be prefetched but that can conditionally prefetch based on command type should be mapped into this space. Read transactions to nonprefetchable space may exhibit side effects; this space may have non-memory-like behavior. PCI 6140 prefetches in this space only if the memory read line or memory read multiple commands are used; transactions using the memory read command are limited to a single data transfer.

The memory-mapped I/O base address and memory-mapped I/O limit address registers define an address range that PCI 6140 uses to determine when to forward memory commands. PCI 6140 forwards a memory transaction from the primary to the secondary interface if the transaction address falls within the memory-mapped I/O address range. PCI 6140 ignores memory transactions initiated on the secondary interface that fall into this address range. Any transactions that fall outside this address range are ignored on the primary interface and are forwarded upstream from the secondary interface (provided that they do not fall into the prefetchable memory range or are not forwarded downstream by the VGA mechanism).

The memory-mapped I/O range supports 32-bit addressing only. The PCI-to-PCI Bridge Architecture Specification does not provide for 64-bit addressing in the memory-mapped I/O space. The memory-mapped I/O address range has a granularity and alignment of 1MB. The maximum memory-mapped I/O address range is 4GB.

The memory-mapped I/O address range is defined by a 16-bit memory-mapped I/O base address register at configuration offset 20h and by a 16-bit memory-mapped I/O limit address register at offset 22h. The top 12 bits of each of these registers correspond to bits <31:20> of the memory address. The low 4 bits are hardwired to 0. The low 20 bits of the memory-mapped I/O base address are assumed to be 0 0000h, which results in a natural alignment to a 1MB boundary. The low 20 bits of the memory-mapped I/O limit address are assumed to be F FFFFh, which results in an alignment to the top of a 1MB block.

Note
The initial state of the memory-mapped I/O base address register is 0000 0000h. The initial state of the memory-mapped I/O limit address register is 000F FFFFh. Note that the initial states of these registers define a memory-mapped I/O range at the bottom 1MB block of memory. Write these registers with their appropriate values before setting either the memory enable bit or the master enable bit in the command register in configuration space.

To turn off the memory-mapped I/O address range, write the memory-mapped I/O base address register with a value greater than that of the memory-mapped I/O limit address register.

6 PCI Bus Arbitration

PCI 6140 must arbitrate for use of the primary bus when forwarding upstream transactions, and for use of the secondary bus when forwarding downstream transactions. The arbiter for the primary bus resides external to the typically on the motherboard. For the secondary PCI bus, PCI 6140 implements an internal arbiter.

6.1 Primary PCI Bus Arbitration

PCI 6140 implements a request output pin, P_REQ#, and a grant input pin, P_GNT#, for primary PCI bus arbitration. PCI 6140 asserts P_REQ# when forwarding transactions upstream; that is, it acts as initiator on the primary PCI bus. However, if a target retry, target disconnect, or a target abort is received in response to a transaction initiated by PCI 6140 on the primary PCI bus, PCI 6140 de-asserts P_REQ# for two PCI clock cycles.

When P_GNT# is asserted LOW by the primary bus arbiter after PCI 6140 has asserted P_REQ#, PCI 6140 initiates a transaction on the primary bus on behalf of master on secondary. When P_GNT# is asserted to PCI 6140 when P_REQ# is not asserted, PCI 6140 parks P_AD, P_CBE, and P_PAR by driving them to valid logic levels. When the primary bus is parked at PCI 6140 and PCI 6140 then has a transaction to initiate on the primary bus, PCI 6140 starts the transaction if P_GNT# was asserted during the previous cycle.

6.2 Secondary PCI Bus Arbitration

PCI 6140 implements an internal secondary PCI bus arbiter. This arbiter supports 4 external masters in addition to PCI 6140. There are no user programmable registers for arbitration. PCI 6140 implements a 1-level rotating priority algorithm on the secondary bus, and will assert SGNT# to a master on the secondary bus, provided that the primary has granted the bus already.

7 One Clock Latency Mode

This mode is for application that has to meet PCI timing requirement and is also the default mode after power up. All PCI signals are clocked output. LDEV# is generated based on internal base and limit address registers comparison. During burst mode, it controls the IRDY# and TRDY# signals to achieve maximum one clock latency. See Section 11.3 and 11.4 for timing diagrams in this mode of operation.

8 Error Handling

PCI 6140 checks, forwards, and generates parity on both the primary and secondary interfaces. To maintain transparency, PCI 6140 always tries to forward the existing parity condition on one bus to the other bus, along with address and data.

To support error reporting on the PCI bus, PCI 6140 implements the following:

- S_SERR# signal on the secondary interface
- Primary status and secondary status registers

This chapter provides detailed information about how PCI 6140 handles errors. It also describes error status reporting and error operation disabling.

8.1 Address Parity Errors

PCI 6140 checks address parity for all transactions on both buses, for all address and all bus commands.

When PCI 6140 detects an address parity error on the primary interface, the following events occur:

- If the parity error response bit is set in the command register, PCI 6140 does not claim the transaction with P_LDEV#; this may allow the transaction to terminate in a master abort.

  If the parity error response bit is not set, PCI 6140 proceeds normally and accepts the transaction if it is directed to or across the PCI 6140.
- PCI 6140 sets the detected parity error bit in the status register.
- PCI 6140 asserts P_SERR# and sets the signaled system error bit in the status register, if both of the following conditions are met:
  - The SERR# enable bit is set in the command register.
  - The parity error response bit is set in the command register.

When PCI 6140 detects an address parity error on the secondary interface, the following events occur:

- If the parity error response bit is set in the bridge control register, PCI 6140 does not claim the transaction with S_LDEV#; this may allow the transaction to terminate in a master abort.

If the parity error response bit is not set, PCI 6140 proceeds normally and accepts the transaction if it is directed to or across the PCI 6140.

- PCI 6140 sets the detected parity error bit in the secondary status register.
- PCI 6140 asserts S_SERR# and sets the signaled system error bit in the status register, if both of the following conditions are met:
  - The SERR# enable bit is set in the command register.
  - The parity error response bit is set in the bridge control register.

8.2 Data Parity Errors

When forwarding transactions, PCI 6140 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.
The following sections describe, for each type of transaction, the sequence of events that occurs when a parity error is detected and the way in which the parity condition is forwarded across PCI 6140.

8.2.1 Configuration Write Transactions to Configuration Space

When PCI 6140 detects a data parity error during a Type 0 configuration write transaction to configuration space, the following events occur:

- If the parity error response bit is set in the command register, PCI 6140 asserts P_TRDY# and writes the data to the configuration register. PCI 6140 also asserts S_SERR#.
  If the parity error response bit is not set, PCI 6140 does not assert P_SERR#.
- PCI 6140 sets the detected parity error bit in the status register, regardless of the state of the parity error response bit.

8.2.2 Read Transactions

When PCI 6140 detects a parity error during a read transaction, the target drives data and data parity, and the initiator checks parity and conditionally asserts SERR#.

For downstream transactions, when PCI 6140 detects a read data parity error on the secondary bus, the following events occur:

- PCI 6140 asserts P_SERR# two cycles following the data transfer, if the secondary interface parity error response bit is set in the bridge control register.
- PCI 6140 sets the detected parity error bit in the secondary status register.
- PCI 6140 sets the data parity detected bit in the secondary status register, if the secondary interface parity error response bit is set in the bridge control register.
- PCI 6140 forwards the bad parity with the data back to the initiator on the primary bus.
- PCI 6140 completes the transaction normally.

For upstream transactions, when PCI 6140 detects a read data parity error on the primary bus, the following events occur:

- PCI 6140 asserts P_SERR# two cycles following the data transfer, if the primary interface parity error response bit is set in the command register.
- PCI 6140 sets the detected parity error bit in the primary status register.
- PCI 6140 sets the data parity detected bit in the primary status register, if the primary interface parity error response bit is set in the command register.
- PCI 6140 forwards the bad parity with the data back to the initiator on the secondary bus.
- PCI 6140 completes the transaction normally.

PCI 6140 returns to the initiator the data and parity that was received from the target. When the initiator detects a parity error on this read data and is enabled to report it, the initiator asserts PERR# two cycles after the data transfer occurs. It is assumed that the initiator takes responsibility for handling a parity error condition.

8.3 Data Parity Error Reporting Summary

In the previous sections, the PCI 6140's responses to data parity errors are presented according to the type of transaction in progress. This section organizes the PCI 6140's responses to data parity errors according to the status bits that the PCI 6140 sets and the signals that it asserts.

Table 8–1 shows setting the detected parity error bit in the status register, corresponding to the primary interface. This bit is set when PCI 6140 detects a parity error on the primary interface.

Table 8–1 Setting the Primary Interface Detected Parity Error Bit

Primary detected parity error bit	Transaction type	Direction	Bus where error was detected	Primary/secondary parity error response bits
0	Read	Downstream	Primary	x/x[1]
0	Read	Downstream	Secondary	x/x
1	Read	Upstream	Primary	x/x
0	Read	Upstream	Secondary	x/x
1	Delayed write	Downstream	Primary	x/x
0	Delayed write	Downstream	Secondary	x/x
0	Delayed write	Upstream	Primary	x/x
0	Delayed write	Upstream	Secondary	x/x

[1]x =don't care

Table 8–2 shows setting the detected parity error bit in the secondary status register, corresponding to the secondary interface. This bit is set when PCI 6140 detects a parity error on the secondary interface.

Table 8–2 Setting the Secondary Interface Detected Parity Error Bit

Secondary detected parity error bit	Transaction type	Direction	Bus where error was detected	Primary/secondary parity error response bits
0	Read	Downstream	Primary	x/x[1]
1	Read	Downstream	Secondary	x/x
0	Read	Upstream	Primary	x/x
0	Read	Upstream	Secondary	x/x
0	Delayed write	Downstream	Primary	x/x
0	Delayed write	Downstream	Secondary	x/x
0	Delayed write	Upstream	Primary	x/x
1	Delayed write	Upstream	Secondary	x/x

[1]x =don't care

Table 8–3 shows setting the data parity detected bit in the status register, corresponding to the primary interface. This bit is set under the following conditions:

- PCI 6140 must be a master on the primary bus.
- The parity error response bit in the command register, corresponding to the primary interface, must be set.

Table 8–3 Setting the Primary Interface Data Parity Detected Bit

Primary data parity detected bit	Transaction type	Direction	Bus where error was detected	Primary/secondary parity error response bits
0	Read	Downstream	Primary	x/x[1]
0	Read	Downstream	Secondary	x/x
1	Read	Upstream	Primary	1/x
0	Read	Upstream	Secondary	x/x
0	Delayed write	Downstream	Primary	x/x
0	Delayed write	Downstream	Secondary	x/x
1	Delayed write	Upstream	Primary	1/x
0	Delayed write	Upstream	Secondary	x/x

[1]x =don't care

Table 8–4 shows setting the data parity detected bit in the secondary status register, corresponding to the secondary interface. This bit is set under the following conditions:

- The PCI 6140 must be a master on the secondary bus.
- The parity error response bit in the bridge control register, corresponding to the secondary interface, must be set.

Table 8–4 Setting the Secondary Interface Data Parity Detected Bit

Secondary data parity detected bit	Transaction type	Direction	Bus where error was detected	Primary/secondary parity error response bits
0	Read	Downstream	Primary	x/x[1]
1	Read	Downstream	Secondary	x/1
0	Read	Upstream	Primary	x/x
0	Read	Upstream	Secondary	x/x
0	Delayed write	Downstream	Primary	x/x
1	Delayed write	Downstream	Secondary	x/1
0	Delayed write	Upstream	Primary	x/x
0	Delayed write	Upstream	Secondary	x/x

[1]x =dont care

Table 8-5 shows assertion of S_SERR#. This signal is set under the following conditions:

- The parity error response bit on the command register and the parity error response bit on the bridge control register must both be set.
- The SERR# enable bit must be set in the command register.

Table 8-5 Assertion of S_SERR# for Data Parity Errors

Transaction type	Direction	Bus where error was detected	Primary/secondary parity error response bits
Read	Downstream	Primary	x/x[1]
Read	Downstream	Secondary	x/x
Read	Upstream	Primary	x/x
Read	Upstream	Secondary	x/x
Delayed write	Downstream	Primary	x/x
Delayed write	Downstream	Secondary	x/x
Delayed write	Upstream	Primary	x/x
Delayed write	Upstream	Secondary	x/x

[1]x =don't care
[2]The parity error was detected on the target (secondary) bus but not on the initiator (primary) bus.
[3]The parity error was detected on the target (primary) bus but not on the initiator (secondary) bus.

8.4 System Error (SERR#) Reporting

PCI 6140 uses the P_SERR# signal to report conditionally a number of system error conditions in addition to the special case parity error conditions described in Section 8.2.3.

Whenever the assertion of P_SERR# is discussed in this document, it is assumed that the following conditions apply:

- For PCI 6140 to assert P_SERR# for any reason, the SERR# enable bit must be set in the command register.
- Whenever PCI 6140 asserts P_SERR#, PCI 6140 must also set the signaled system error bit in the status register.

When S_SERR# is asserted by secondary device, PCI 6140 sets the received system error bit in the secondary status register.

The PCI 6140 also conditionally asserts P_SERR# when parity error reported on target bus during write transaction.

9 Reset

This chapter describes the primary interface, secondary interface, and chip reset mechanisms.

9.1 Primary Interface Reset

PCI 6140 has one reset input, P_RST#. When P_RST# is asserted, the following events occur:

- PCI 6140 immediately three-states all primary and secondary PCI interface signals.
- PCI 6140 performs a chip reset.
- Registers that have default values are reset.

The P_RST# asserting and de-asserting edges can be asynchronous to P_CLK and S_CLK.

9.2 Secondary Interface Reset

PCI 6140 is responsible for driving the secondary bus reset signal, S_RST#.

PCI 6140 asserts S_RST# when any of the following conditions is met:

- Signal P_RST# is asserted.
  Signal S_RST# remains asserted as long as P_RST# is asserted and does not de-assert until P_RST# is de-asserted and the secondary clock serial disable mask has been shifted in (23 clock cycles after P_RST# de-assertion).
- The secondary reset bit in the bridge control register is set.
  Signal S_RST# remains asserted until a configuration write operation clears the secondary reset bit and the secondary clock serial mask has been shifted in.
- The chip reset bit in the diagnostic control register is set.
  Signal S_RST# remains asserted until a configuration write operation clears the secondary reset bit and the secondary clock serial mask has been shifted in.

When S_RST# is asserted, all secondary PCI interface control signals, including the secondary grant outputs, are immediately three-stated. Signals S_AD, S_CBE#, and S_PAR are driven LOW for the duration of S_RST# assertion. All posted write and delayed transaction data buffers are reset; therefore, any transactions residing in buffers at the time of secondary reset are discarded.

When S_RST# is asserted by means of the secondary reset bit, PCI 6140 remains accessible during secondary interface reset and continues to respond to accesses to its configuration space from the primary interface.

10 Bridge Behavior

A PCI cycle is initiated by asserting the FRAME# signal. In a bridge, there are a number of possibilities. These are summarized in the table below.

Bridge Actions for Various Cycle Types

Initiator	Target	Response
Master on primary	Target on Primary	PCI 6140 forward all signals to secondary. It detects this situation by decoding the address as well as monitoring the P_LDEV# for other fast and medium devices on the primary port.
Master on primary	Target on secondary	PCI 6140 asserts P_LDEV#, then passes the cycle to the secondary. When cycle is complete on the target port, it will wait for the initiator to end with normal termination.
Master on primary	Target not on primary nor secondary port	PCI 6140 does not respond and the cycle will terminate as master abort.
Master on secondary	Target on the secondary port	PCI 6140 forward all signals to primary. It detects this situation by decoding the address as well as monitoring the S_LDEV# for other fast and medium devices on the secondary port.
Master on secondary	Target on primary port	PCI 6140 asserts S_LDEV#, then passes the cycle to the appropriate port. When cycle is complete on the target port, it will wait for the initiator to end with normal termination.
Master on secondary	Target not on primary nor the other secondary	PCI 6140 does not respond.

A target then has up to three cycles to respond before subtractive decoding is initiated. If the target detects an address hit, it should assert its LDEV# signal in the cycle corresponding to the values of bits 9 and 10 in the Configuration Status Register.

Termination of a PCI cycle can occur in a number of ways. Normal termination begins by the initiator (master) de-asserting FRAME# with IRDY# being asserted (or remaining asserted) on the same cycle. The cycle completes when TRDY# and IRDY# are both asserted simultaneously. The target should de-assert TRDY# for one cycle following final assertion (sustained three-state signal).

10.1 Abnormal Termination (Initiated by Bridge Master)

10.1.1 Master Abort

Master abort indicates that PCI 6140 acting as a master receives no response (i.e., no target asserts P_LDEV# or S_LDEV#) from a target., the bridge de-asserts FRAME# and then de-asserts IRDY#.

10.1.2 PCI Master on Primary Bus

The table illustrates the direction of the PCI control/data path when a PCI transaction is initiated by a PCI master residing on the primary bus. It guarantees the integrity of the cycle, viewed from the primary and the secondary side.

Slave location	Command	FRAME/CBE IRDY	LDEV/TRDY STOP	AD
Primary bus	read	S->P	P->S	P->S
Primary bus	write	S->P	P->S	P<-S
Secondary bus	read	S->P	P<-S	---
Secondary bus	write	S->P	P<-S	---

PCI 6140 is designed to pass almost all the primary cycles to the secondary side, except configuration cycle in the 1 clock delay case, as described below. PCI 6140 performs configuration type #1 to type #0 conversion, on the cycle with a matched bus-number. It will pass the type #1 configuration cycle that locates on the secondary side of the bridge.

Configuration type #1 to type #0 conversion

When a type #1 configuration cycle appears on the primary side with a bus-number equaling to PCI 6140 bridge bus-number, PCI 6140 performs the type #1 to type #0 conversion cycle, as follows. First, it will retry all the subsequent all the primary cycles, until its completion. And it will de-grant the secondary bus to block the secondary master. The conversion cycle will appear on the secondary bus only, without being reflected to the primary side.

Then, it issues the converted type #0 configuration cycle on the secondary side. The termination of the cycle can be either normal, master abort or slave abort. In the case of read, PCI 6140 will latch the data, and wait for the same type #1 configuration cycle on the primary side.

Configuration type #1 to type #1 by-passing

When a type #1 configuration cycle appears on the primary side with a bus-number greater than PCI 6140 bridge bus-number, but smaller than the secondary sub-ordinate bus-number, the same type #1 configuration cycle will appear on the secondary side. Otherwise, the bypassing process is very similar to the type #1 to type #0 conversion process. PCI 6140's internal state machine will generate the secondary cycle, retry all the primary cycle, and block any secondary master.

Type-0 Configuration cycle filter mode

In this Type-0 configuration cycle filter mode, PCI 6140 will filter out all the primary Type-0 configuration cycle by delaying passing of primary P_FRAME# by one PCI clock. In case of Type-1 configuration cycle through the bridge, it will return retry and relies on the internal state machine to do the conversion cycle to generate Type-0 or Type-1 on the secondary side.

Decoding

PCI 6140 uses decoding circuit to determine the slave device location.

During the memory cycle, PCI 6140 uses Memory Base/Limit and PrefetchBase/Limit. Slave is on the secondary side if:

MemoryBase[31:16] <= Address <= MemoryLimit[31:16]
or PrefetchBase[31:16] <= Address <= PrefetchLimit[31:16]
or when VGA is enabled, 0xa0000 <= Address <= 0xbffff

During the I/O cycle, PCI 6140 uses I/O Base/Limit register. Slave device is on the secondary side if:

IoBase[15:4] <= Address <= IoLimit[15:4], see Note 1.
or when VGA is enabled, 0x3b0 <= Address <= 0x3bb
or when VGA is enabled, 0x3c0 <= Address <= 0x3df
***Note1: when ISA is enabled, the I/O space between address 0-256 are always reserved at every 1K boundary.

During the Type-1 configuration cycle, PCI 6140 uses the device bus-number and the sub-ordinate bus-number. Slave device is on the secondary side if:

BusNumber[7:0] <= Address <= SubordinateBusNumber[7:0]

All the type #0 configuration cycle, interrupt acknowledge cycle and the special cycle appearing on the primary side is considered to have slave on the primary side. Likewise, all the similar cycle appearing on the secondary side is considered to have slave on the secondary side.

Secondary master

The secondary master issues S_REQ#[3:0] to request the bus. PCI 6140 will generate P_REQ# on the primary side. When P_GNT# is active, PCI 6140 will use the round-robin algorithm to grant one secondary master using S_GNT#[3:0].

The control/data path is illustrated below. PCI 6140 will pass all the cycle from the secondary side to the primary.

Slave location	read/ Write	FRAME/CBE IRDY	LDEV/TRDY STOP	AD
primary	read	S->P	P->S	P->S
primary	write	S->P	P->S	P<-S
slave	read	S->P	P<-S	---
slave	write	S->P	P<-S	---

PCI clock run feature

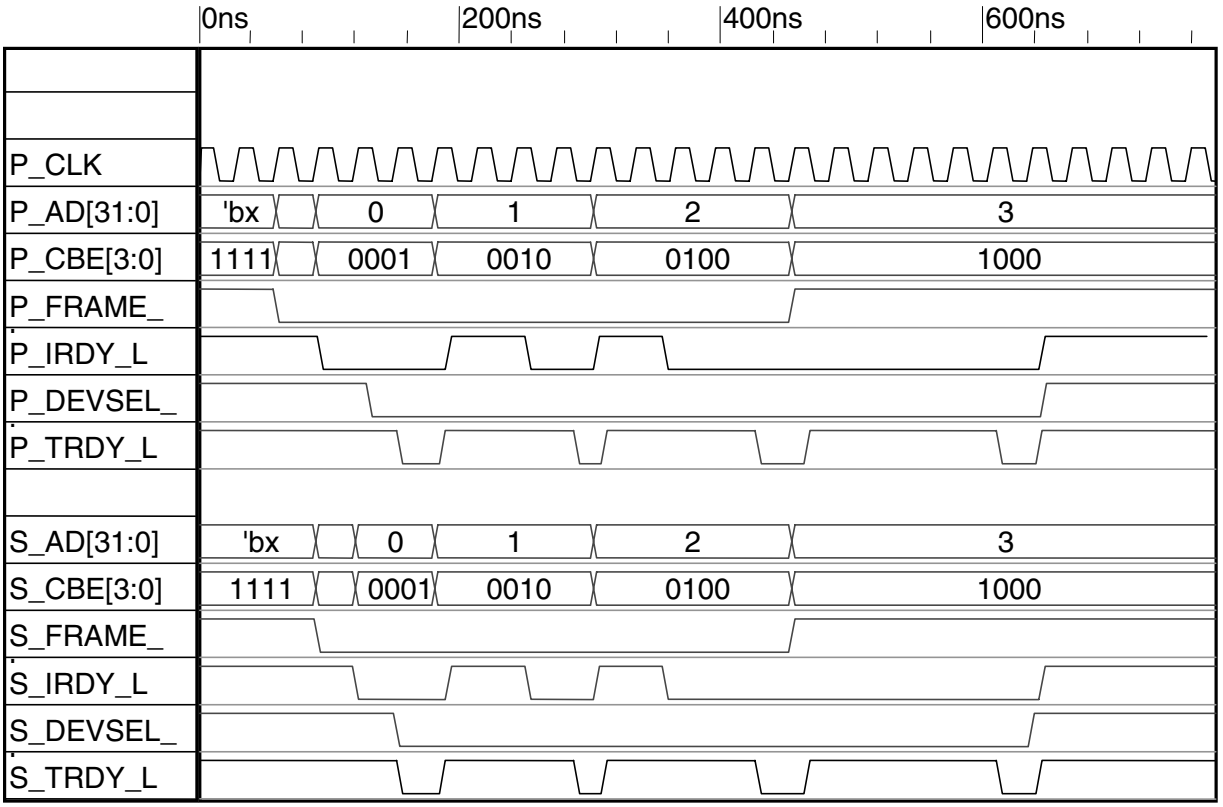
PCI 6140 supports PCI clock run protocol defined in the PCI Mobile Design Guide 1.0. P_CLKRUN# is set high when the system's central resource wants to stop P_CLK, and then PCI 6140 will either signal that it allows PCI clock to be stopped by letting P_CLKRUN# remain high, or it will signal to the system to keep P_CLK running by driving P_CLKRUN# low for 2 clocks then release by then the system's central resource will keep P_CLKRUN# low. There are three situations that PCI 6140 will keep primary clock running. First is bit 2 of clock run control register is set, second there is a pending cycle on going through the chip, third is on behalf of secondary PCI device.

Secondary clock run is enabled by bit 1 of the clock run control register, by default the initiation of stopping/slowing down secondary comes from primary, however if bit 4 of clock run control register is set, secondary clock will be stopped when the bus is idle and there is no cycle from primary bus.

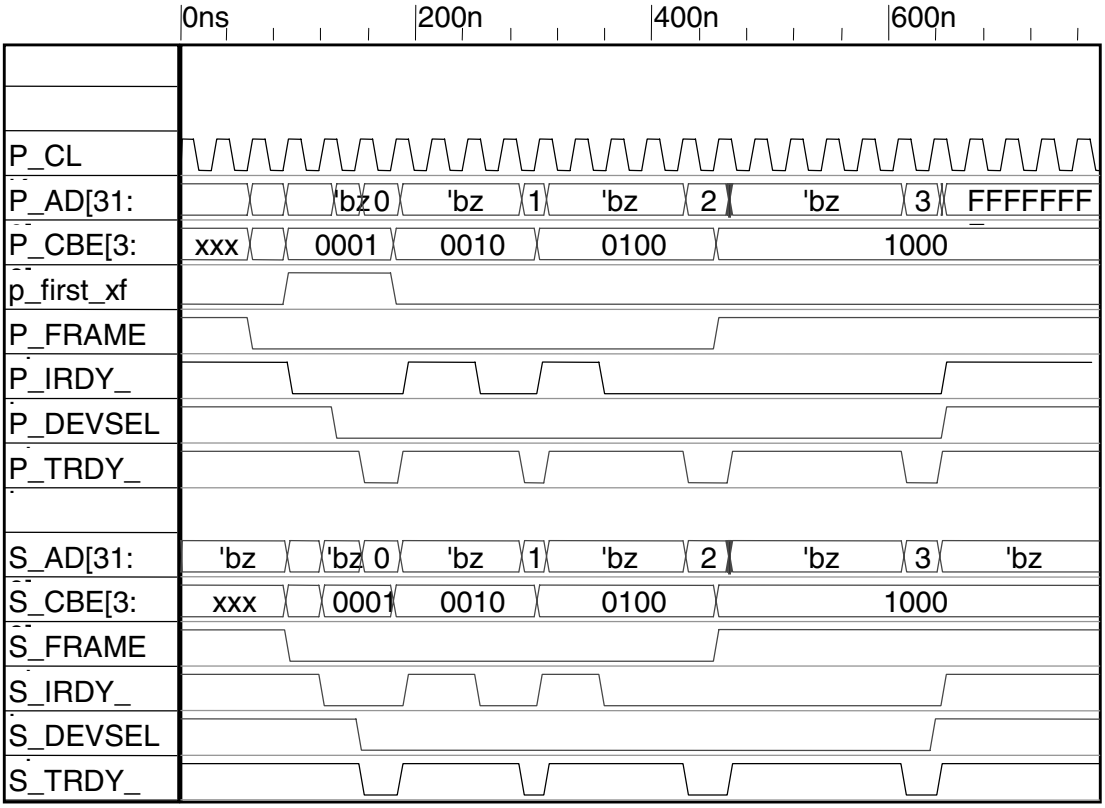
11 Timing Diagrams

11.1 Zero-clock Latency Mode

11.1.1 Write Transaction in Zero-clock Latency Mode



11.1.2 Read Transaction in Zero-clock Latency mode

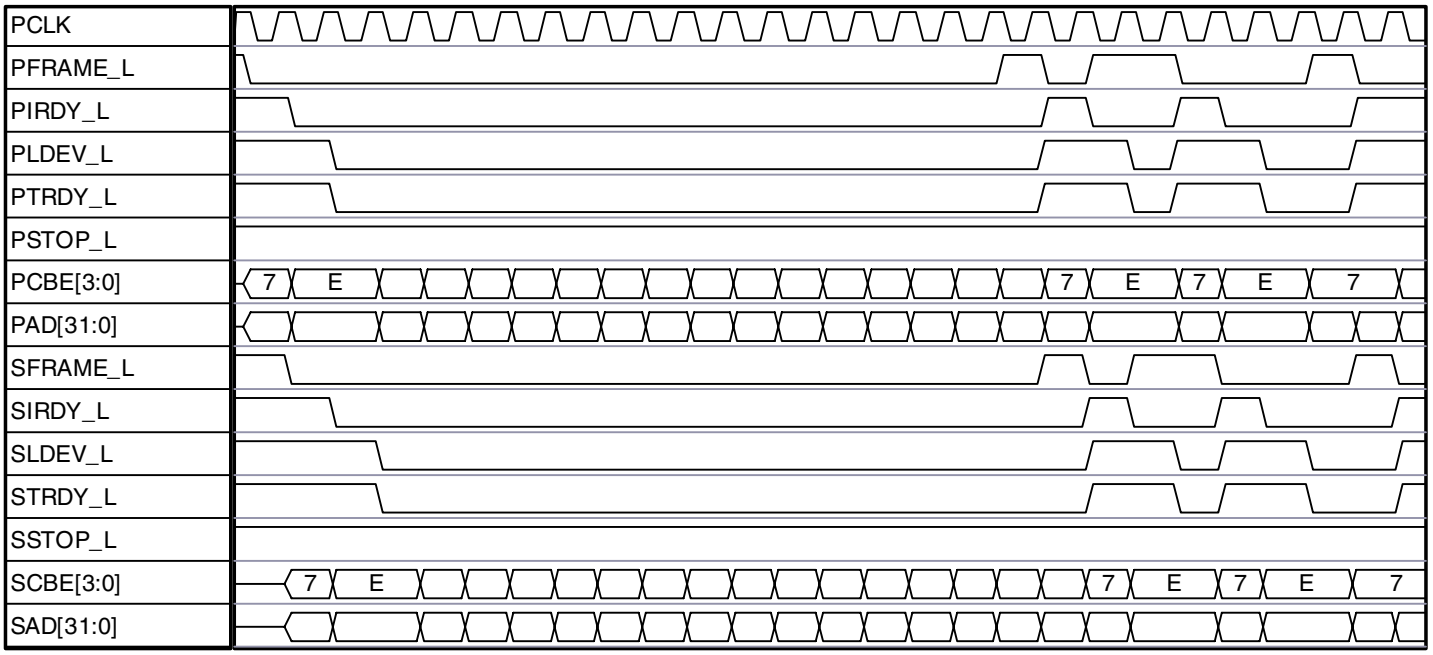


11.2 One-Clock Latency Mode

PCI 6140 passes the start of any memory or I/O cycle from P-bus to S-bus with 1 PCLK delay. On the second PCLK, it uses the decoder to determine whether the cycle belonging to the P or S-bus.

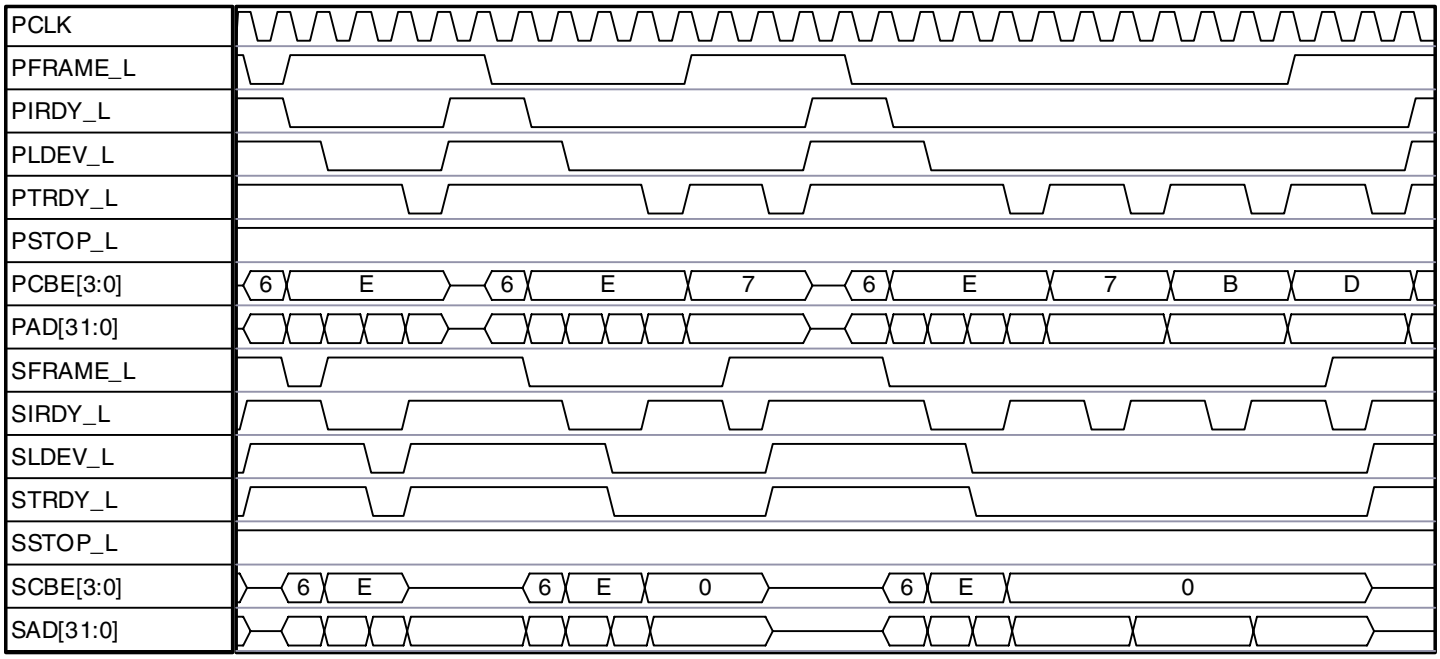
11.2.1 Primary to Secondary Write Transaction

PCI 6140 passes the I/O or memory write cycle to the S-bus with 1 PCLK delay, including both address and data phase. Thus, it can always sustain 0-ws burst.



11.2.2 Primary to Secondary Read Transaction

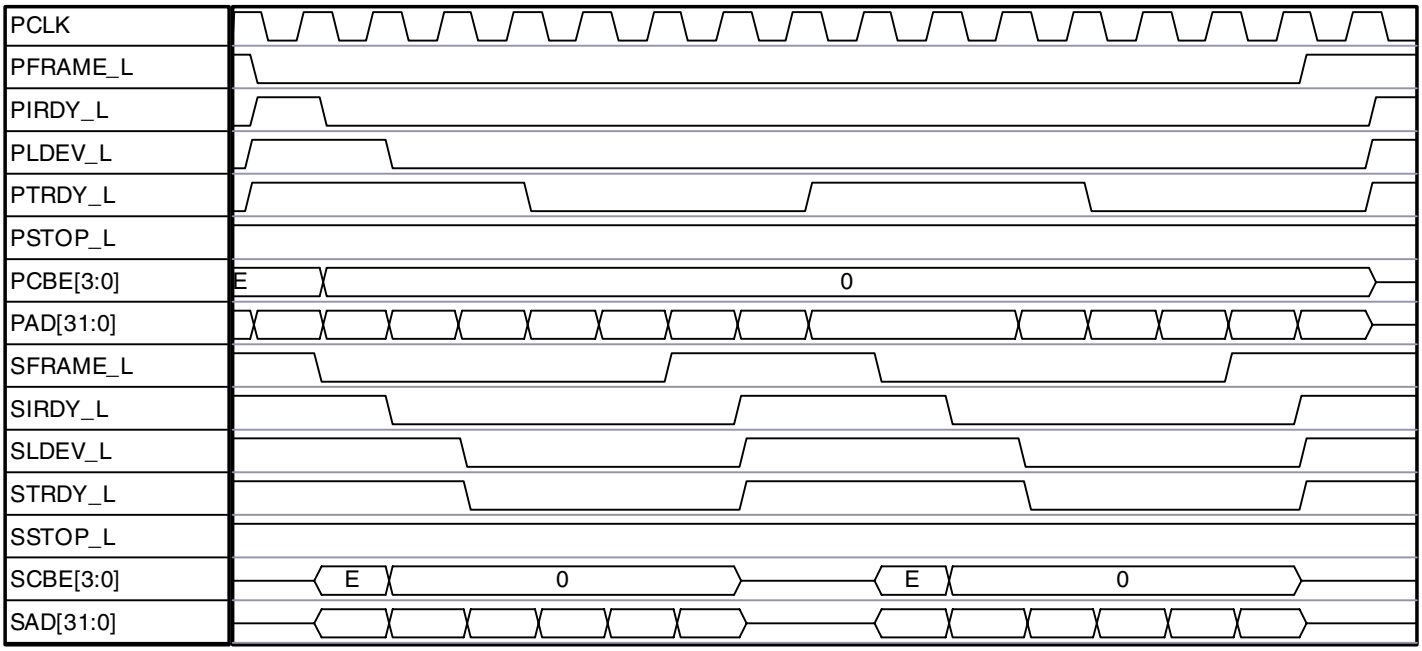
Note, in this mode, user has the option to force the bursting SBE[3:0] to be all active, by setting bit1 of register C0h. Otherwise, PCI 6140 can still guarantee the correct SBE[3:0] when SIRDY_L is active.



11.2.3 Memory Read Line / Memory Read Multiple Transaction (Block Read Mode)

When bit 6 of register C0h is '1', PCI 6140 will read exactly 1 cache-line size of data, when it is memory read line or memory read multiple or memory read in the prefetch area.

In this mode, PCI 6140 will read 1 cache-line data with 0 wait states on the S-bus. At the same time, it delivers the data to the P-bus in with 0 wait states. If the P-bus continues the transfer, PCI 6140 will initiate another memory read line/multiple on the S-bus and deliver to the P-bus as before.



12. Electrical Specifications

Maximum Ratings

(Above which the useful life may be impaired. For user guidelines, not tested.)

Symbol	Parameter	Rating	Unit
V_{cc}	Power Supply	-0.3 to 3.8	V
V_{in}	Input Voltage	-0.3 to 5.75	V
V_{out}	Output Voltage	-0.3 to V_{cc}+0.3	V
T_{STG}	Storage Temperature	-40 to +125	°C
	Operating Ambient Temperature	0 to +70	°C

Note: Stresses greater than those listed under MAXIMUM RATINGS may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

DC Electrical Characteristics

Symbol	Parameter	Condition	Min	Max	Unit	Notes
V_{cc}	Supply Voltage		3.0	3.6	V	
V_{ih}	Input High Voltage		$0.5V_{cc}$	V_{cc} + 0.5	V	
V_{il}	Input Low Voltage		-0.5	$0.3V_{cc}$	V	
I_{ih}	Input High Leakage Current	$0 < V_{in} < V_{cc}$		10	µA	
I_{il}	Input Low Leakage Current	$0 < V_{in} < V_{cc}$		-10	µA	
V_{oh}	Output High Voltage	I_{out} = -0.5mA	2.0		V	
V_{ol}	Output Low Voltage	I_{out} = 1.5 mA		1.0	V	
C_{in}	Input Pin Capacitance			10	pF	
C_{clk}	**CLK** Pin Capacitance		5	12	pF	
C_{IDSEL}	**IDSEL** Pin Capacitance			8	pF	
L_{pin}	Pin Inductance			20	nH	

AC Specifications

Symbol	Parameter	Condition	Min	Max	Unit	Notes
I_{oh}	Switching Current HIGH	V_{out} = 0.89 V	-39.6		mA	
		V_{out} = 2.31 V		-158.4	mA	
		V_{out} = 2.97 V		-42	mA	
I_{ol}	Switching Current LOW	V_{out} = 1.98 V	52.8		mA	
		V_{out} = 0.59 V		211.2	mA	
		V_{out} = 0.33 V	8.9	76	mA	
I_{cl}	Low Clamp Current	-5V < V_{in} < -1V	-25 + (V_{in} + 1)/0.015		mA	
$slew_r$	Output Rise Slew Rate	0.4V to 2.4V load	1	3.3	V/ns	
$slew_f$	Output Fall Slew Rate	2.4V to 0.4V load	1	3.3	V/ns	

5V Signal Tolerant

Input signals can tolerate 5V signals. However, all Vcc must be 3V and output signals should still be at 3V.

Maximum Power

Maximum power consumption of PCI 6140 at 33 MHz is 200mW.

13. Clock and Timing Specification

PCI Clock Timing

Symbol	Parameter	33 MHz		Units
		Min	Max	
T_cyc	PCICLK Cycle Time	30		ns
T_high	PCICLK High Time	11		ns
T_low	PCICLK low Time	11		ns
	PCICLK Slew Rate	1	4	V/ns
T_sclk	Secondary PCI clock output delay from Primary PCI clock input	1.5	5	ns

PCI Timing Parameters

Symbol	Parameter	33 MHz		Units
		Min	Max	
T_val	PCICLK to signal valid delay	2	11	ns
T_on	Float to active delay	2		ns
T_off	Active to float delay		28	ns
T_su	Input setup time to PCICLK	7		ns
T_h	Input hold time from PCICLK	0		ns
T_rst-clk	Reset active time after PCICLK stable	100		µs
T_rst-off	Reset Active to output float delay		40	ns

Appendix A. Interrupt –Device Number Binding

The PCI-to-PCI Bridge Architecture Specification, Rev 1.0 specifies that the system BIOS will assume an association between device location and which INTx line it uses when requesting an interrupt. For PCI 6140, the following table should be followed for IDSEL and INTx mapping:

IDSEL connection

Device Number on Secondary Bus	S_AD line assignment for IDSEL
0	S_AD[16]
1	S_AD[17]
2	S_AD[18]
3	S_AD[19]
4	S_AD[20]
5	S_AD[21]
6	S_AD[22]
7	S_AD[23]
8	S_AD[24]
9	S_AD[25]
10	S_AD[26]
11	S_AD[27]
12	S_AD[28]
13	S_AD[29]
14	S_AD[30]
15	S_AD[31]

Interrupt-Device Number Binding

S_AD line assignment for IDSEL	Interrupt pin on device	Interrupt pin on connector
S_AD[16, 20, 24, 28]	INTA#	INTA#
	INTB#	INTB#
	INTC#	INTC#
	INTD#	INTD#
S_AD[17, 21, 25, 29]	INTA#	INTB#
	INTB#	INTC#
	INTC#	INTD#
	INTD#	INTA#
S_AD[18, 22, 26, 30]	INTA#	INTC#
	INTB#	INTD#
	INTC#	INTA#
	INTD#	INTB#
S_AD[19, 23, 27, 31]	INTA#	INTD#
	INTB#	INTA#
	INTC#	INTB#
	INTD#	INTC#

Appendix B. Sample Application Schematics

Note: The following pages contain the schematics for the PCI 6140 PCI-to-PCI Bridge reference design.

PCI 6140 PCI-to-PCI BRIDGE Reference Schematic

PAGE	FUNCTIONAL DESCRIPTION
1.	COVER
2.	PCI 6140 PCI-to-PCI BRIDGE
3.	PCI BUS GOLD FINGER
4.	PCI SLOT 0 and 1
5.	PCI SLOT 2 and 3

REVISION HISTORY:
Rev 1.0 4/3/2001

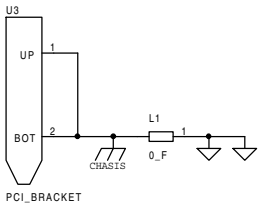
FEATURES:
- 4 PCI SLOTs

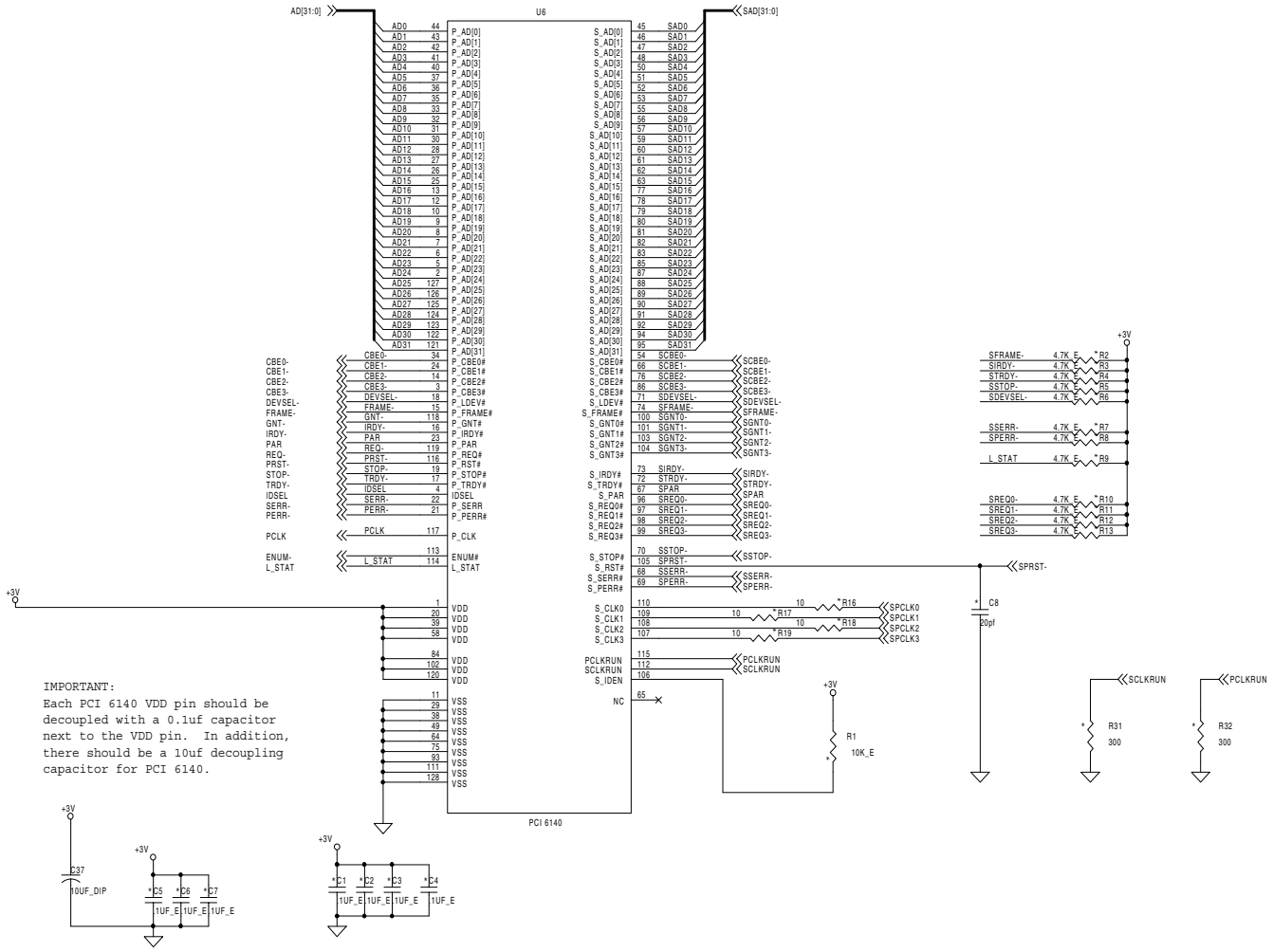
COMPONENT PACKAGE INFORMATION:
CAP, RES:
E: 0603
F: 0805
G: 1206

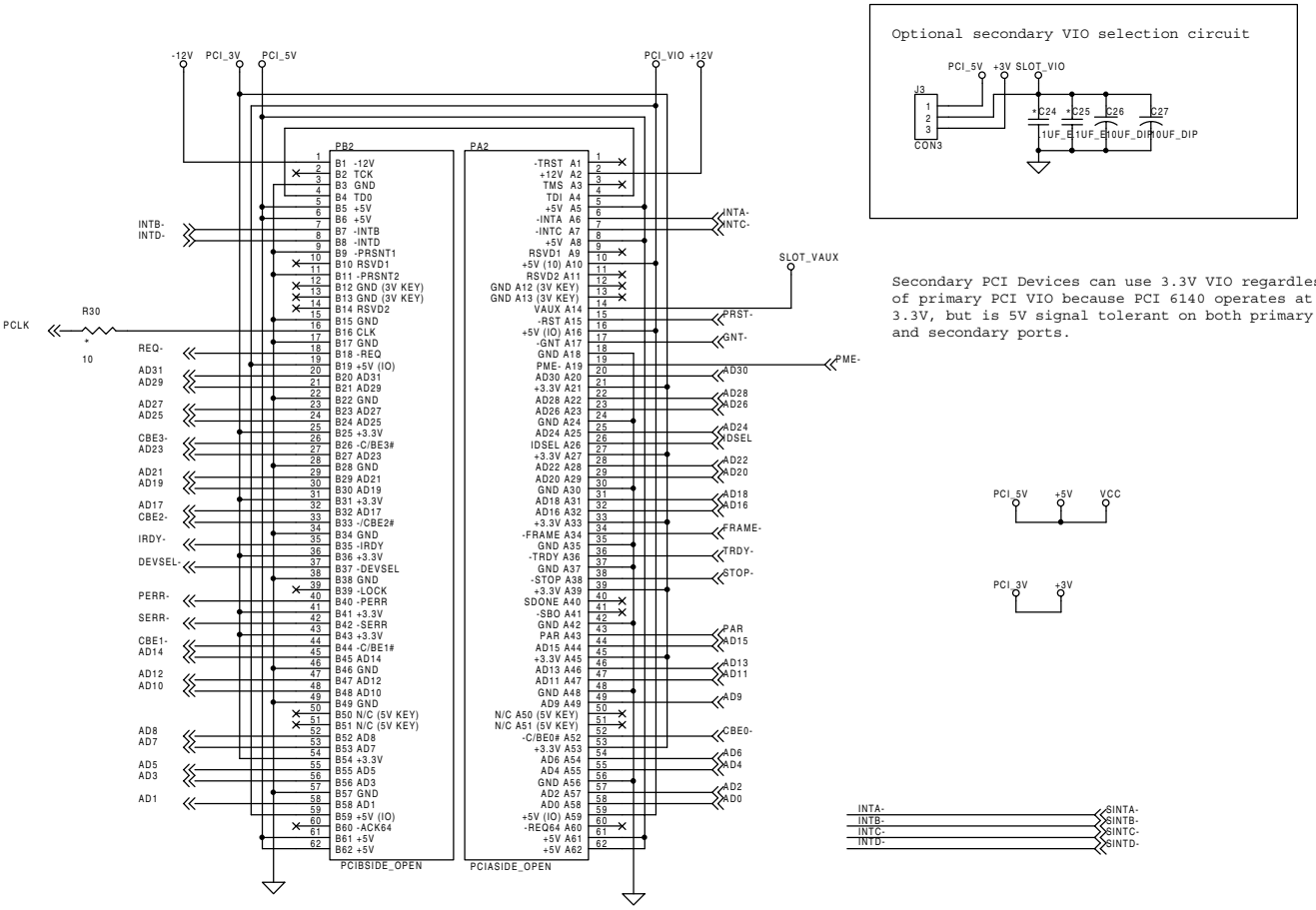
_OPEN: DO NOT INSTALL.
_NL: DO NOT LOAD.

TRANSISTORS ARE SOT-23 UNLESS SPECIFIED.
DIODE, ZENER DIODE ARE MINI-MELF UNLESS SPECIFIED.

ORCAD Capture for Windows - Ver. 7.1

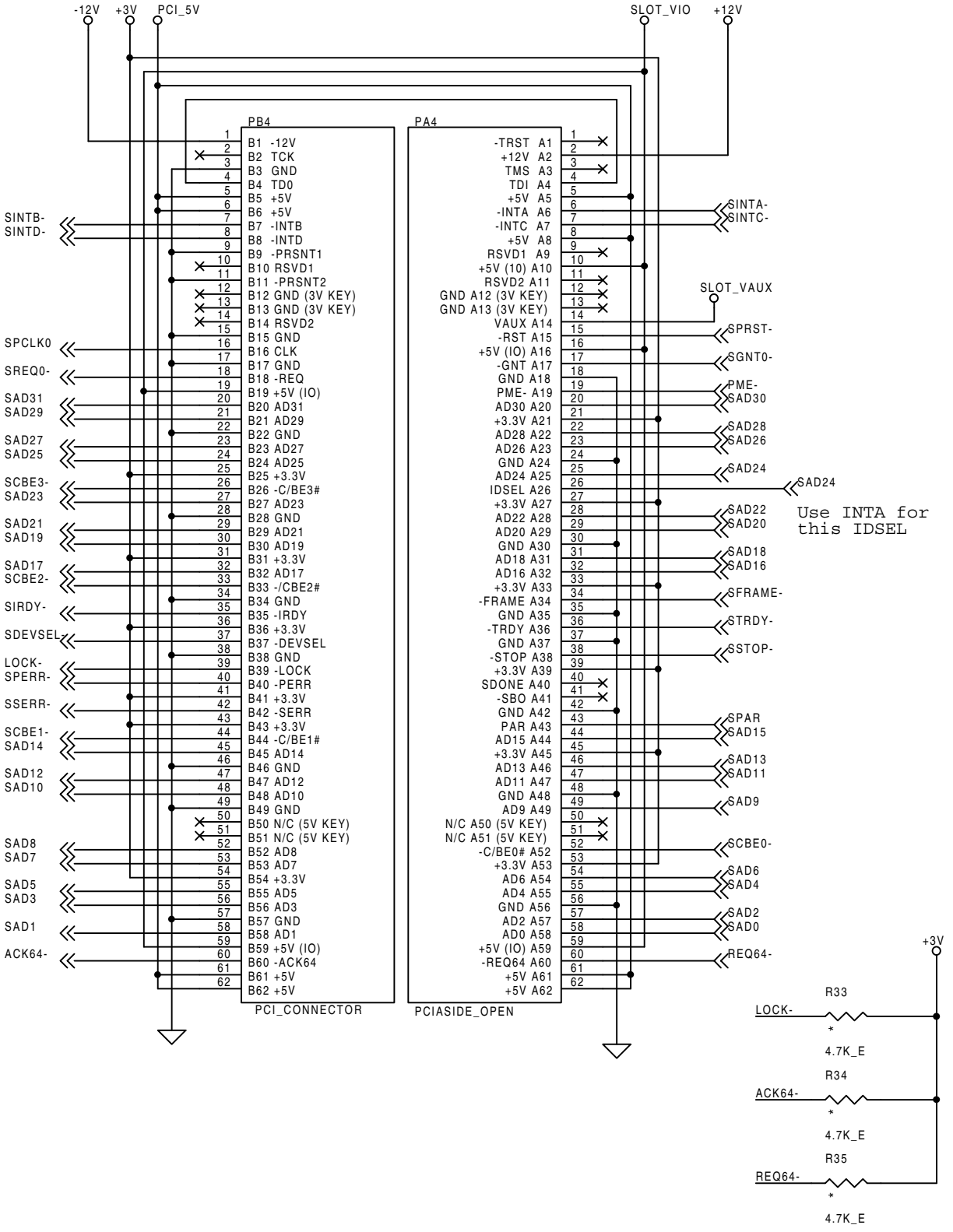




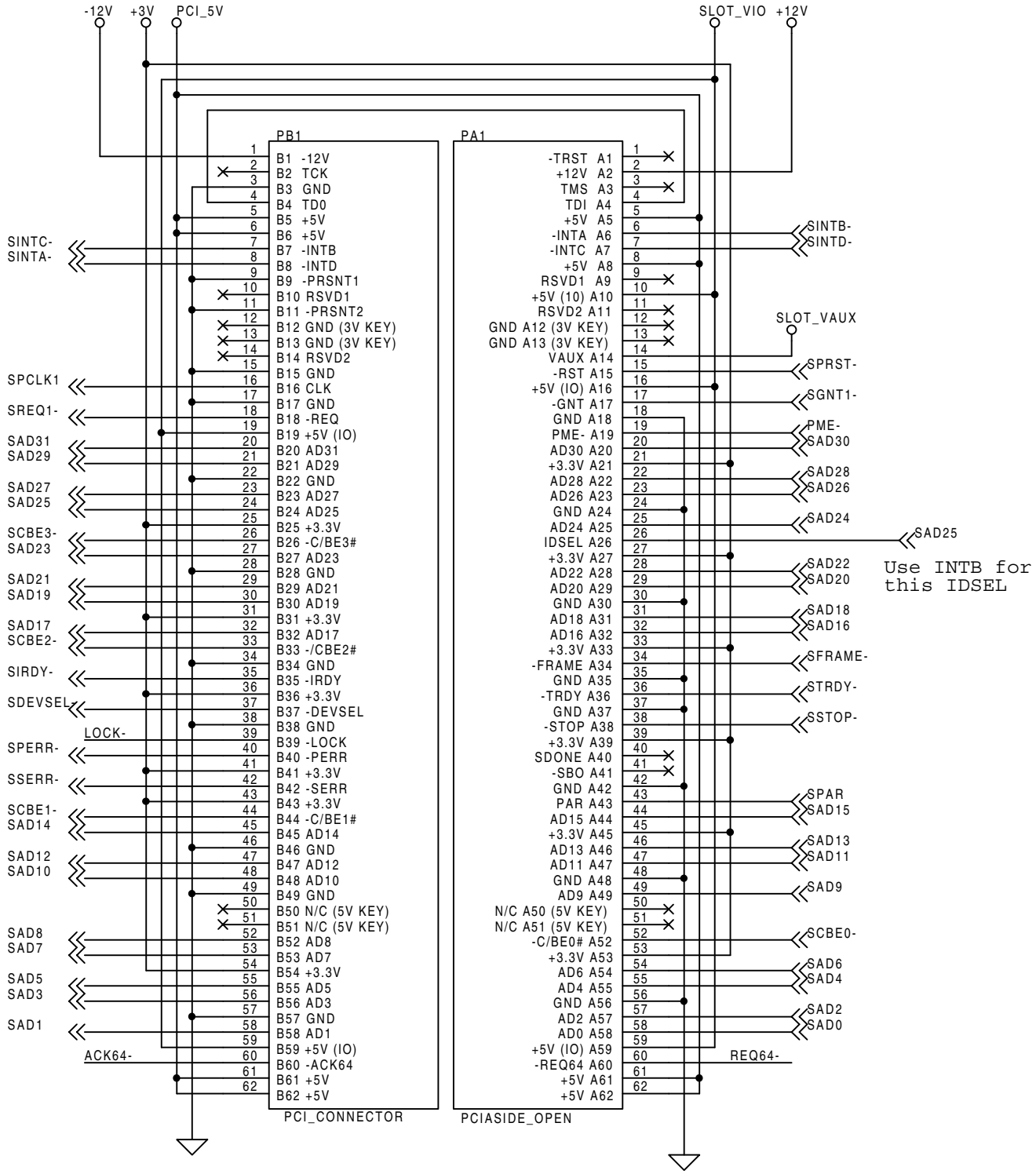


GOLD FINGER

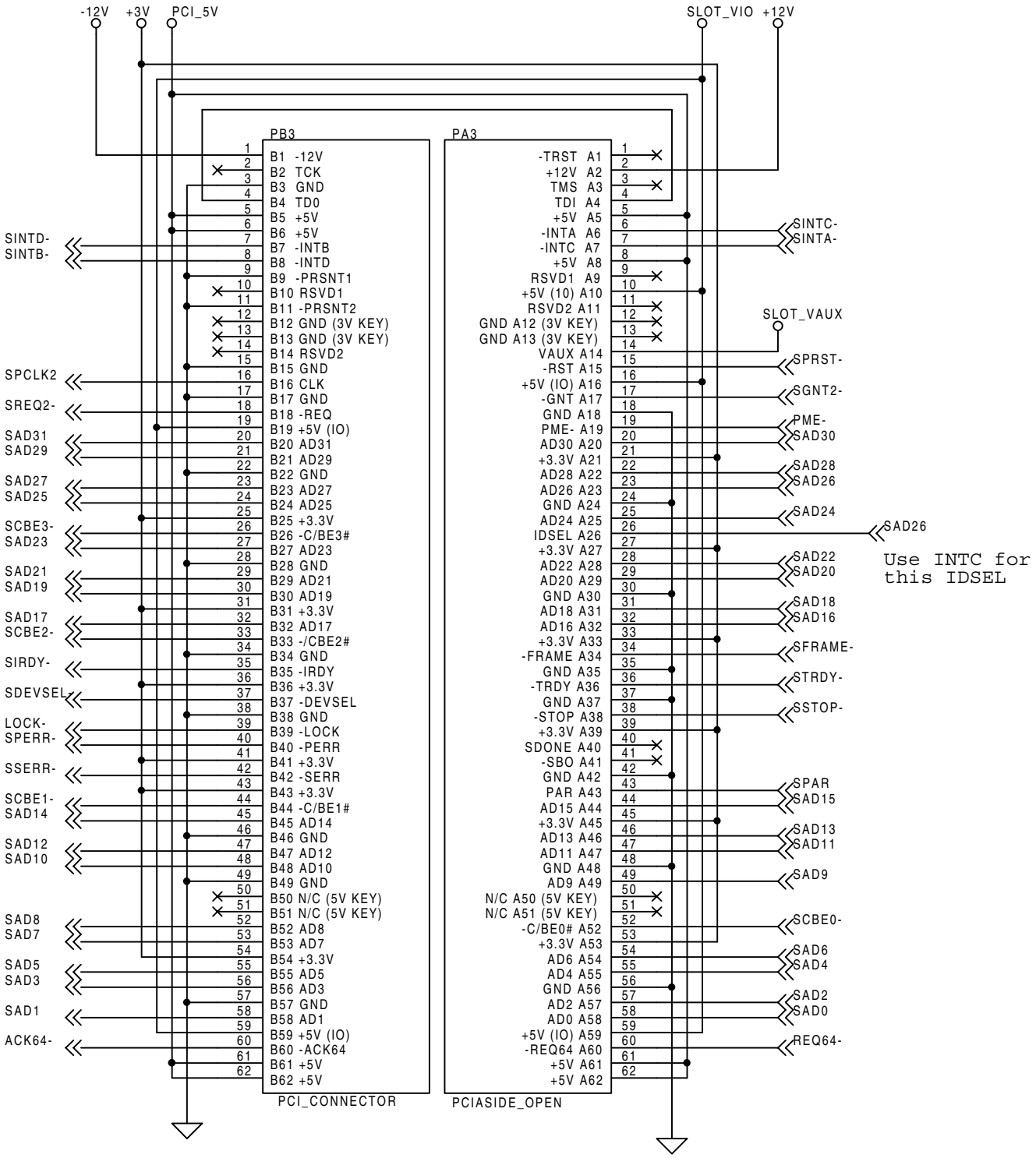
PCI SLOT 0



PCI SLOT 1



PCI SLOT 2



PCI SLOT 3