DECchip 21050 PCI-PCI Bridge Configuration: An Application Note

Order Number: EC–Q9Y4A–TE

June 1994

This application note presents guidelines for configuration of the DECchip 21050 in a system.

Revision/Update Information: This is a new application note.

Digital Equipment Corporation
Maynard, Massachusetts

June 1994

While Digital believes the information in this publication is correct as of the date of publication, it is subject to change without notice.

Digital Equipment Corporation makes no representation that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with this description.

© Digital Equipment Corporation 1994

All Rights Reserved.
Printed in U.S.A.

The following are trademarks of Digital Equipment Corporation: DEC, DECchip, DECwrite , and the DIGITAL logo.

All other trademarks and registered trademarks are the property of their respective holders.

This document was prepared using DECwrite, Version 3.0.

Contents

List of Tables

Introduction

This document presents guidelines for configuration of the DECchip 21050 in a system. This application note is limited to PCI configuration of the 21050 only and does not cover any hardware application topics. Contents of this document include:

- Summary of the 21050 configuration space
- State of 21050 configuration space
- 21050 initialization guidelines
- Sample initializations for the 21050

The following related documents may be useful.

Published by Digital Equipment Corporation:

- *DECchip 21050 PCI-PCI Bridge Data Sheet*, Revision 1.0
- *DECchip 21050 PCI-PCI Bridge Hardware Implementation: An Application Note*
- *DECchip 21050 PCI Evaluation Board User's Guide*

Published by the PCI Special Interest Group:

- *PCI Local Bus Specification*, Revision 2.0
- *PCI-PCI Bridge Architecture Specification*, Revision 1.0

Functional Overview

The DECchip 21050 PCI-PCI Bridge connects two independent PCI buses. You can implement the 21050 either on a motherboard to provide capability for additional slots and devices, or on an option card to allow multiple devices. You can also use the 21050 to isolate bus traffic.

The DECchip 21050 is a transparent device, that is, it requires no special driver software to run in a system. The 21050 does require initialization code to set up its configuration space, allocate memory and I/O space behind the bridge, identify VGA devices behind the 21050, and so on. This initialization code is included in the BIOS code or firmware.

Once configured, the 21050 operates without the need for any additional software control.

21050 Configuration Space

The configuration registers of the 21050 comply with the registers of the type 1 configuration header format for PCI-PCI Bridges, as specified in the *PCI-to-PCI Bridge Architecture Specification*. The first 16 bytes of this header are identical to those of any other PCI device. The remaining 48 bytes of the predefined header are specific to the PCI-PCI Bridge architectural standard. The 21050 also implements 16 bytes of implementation-specific configuration space header. For a detailed description of each field, refer to the *DECchip 21050 PCI-PCI Bridge Data Sheet*.

Table 1 21050 Configuration Space

Address	31 24	23 16	15 8	7 0
00h	Device ID		Vendor ID	
04h	Primary Status		Primary Command	
08h	Class Code			Revision ID
0Ch	Reserved	Header Type	Primary Latency Timer	Cache Line Size
10h	Reserved			
14h	Reserved			
18h	Secondary Latency Timer	Subordinate Bus Number	Secondary Bus Number	Primary Bus Number
1Ch	Secondary Status		I/O Limit	I/O Base
20h	Memory Limit		Memory Base	
24h	Prefetchable Memory Limit		Prefetchable Memory Base	
28h	Reserved			
2Ch	Reserved			
30h	Reserved			
34h	Reserved			
38h	Reserved			
3Ch	Bridge Control		Interrupt Pin	Reserved
40h	SErr Event Disable	Burst Limit Counter	Diagnostic Control	Chip Control
44h	Reserved		Secondary Target Wait Timer	Primary Target Wait Timer
48h	Secondary Write Attempt Counter			
4Ch	Primary Write Attempt Counter			
50h - FFh	Reserved			

The following tables break out the bit definitions of 21050 control registers. For a description of these bits, refer to the *DECchip 21050 PCI-PCI Bridge Data Sheet*.

Primary Command		
Field Bit	**DWord Bit**	**Description**
0	0	IO Enable
1	1	Memory Enable
2	2	Master Enable
3	3	Reserved
4	4	VGA Snoop Enable
5	5	Parity Error Response
6	6	Wait Cycle Control
7	7	**p_serr_l** Driver Enable
8	8	Fast Back-to-Back Control
15:9	15:9	Reserved

Primary Status		
Field Bit	**DWord Bit**	**Description**
6:0	22:16	Reserved
7	23	Fast Back-to-Back Capable
8	24	Data Parity Detected
10:9	26:25	**p_devsel_l** Timing
11	27	Signaled Target Abort
12	28	Received Target Abort
13	29	Received Master Abort
14	30	Signaled System Error
15	31	Detected Parity Error

Secondary Status		
Field Bit	**DWord Bit**	**Description**
6:0	22:16	Reserved
7	23	Fast Back-to-Back Capable
8	24	Data Parity Detected
10:9	26:25	**s_devsel_l** Timing
11	27	Signaled Target Abort
12	28	Received Target Abort
13	29	Received Master Abort
14	30	Detected System Error

Bridge Control		
Field Bit	**DWord Bit**	**Description**
0	16	Parity Error Response
1	17	**s_serr_l** Forward Enable
2	18	ISA Enable
3	19	VGA Enable
4	20	Reserved
5	21	Master Abort Mode
6	22	Secondary Bus Reset
7	23	Fast Back-to-Back Control
15:0	31:24	Reserved

Chip Control		
Field Bit	**DWord Bit**	**Description**
0	0	Arbitration Mode
1	1	Disable Posting
3:2	3:2	Request Mask Timer
4	4	Secondary Prefetch Enable

Diagnostic Control		
Field Bit	**DWord Bit**	**Description**
0	8	Chip Reset
1	9	Test Mode
2	10	**p_perr_l** Diag Assertion
3	11	**s_perr_l** Diag Assertion
4	12	**p_serr_l** Diag Assertion
7:5	15:13	Reserved

p_serr_l Disable		
Field Bit	**DWord Bit**	**Description**
0	24	Reserved
1	25	Bad data parity on target bus for posted write
2	26	Unable to deliver posted write data
3	27	Target abort on target bus during posted write
4	28	Master abort on target bus during posted write
7:5	31:29	Reserved

21050 Initial Conditions

After system reset, the 21050 is left in the following state:

- Memory, I/O, and Master Enables all reset to zero. The 21050 does not respond to any memory or I/O transactions on either the primary or secondary bus.
- Primary, secondary, and subordinate bus numbers are all zero. The 21050 forwards only Type 1 transactions downstream with a bus number of 0. You must initialize these bus numbers before you can configure any devices downstream of the 21050.
- Prefetchable memory, nonprefetchable memory, and I/O base and limit address registers are all zero. Note that this does not mean that these address ranges are turned off. In fact, it does mean that a memory range of addresses 000FFFFF (hex) – 00000000 (hex) and an I/O range of addresses 0FFF (hex) – 0000 (hex) are defined for downstream forwarding. Because the enables are off, no memory or I/O transactions are forwarded, however. Before turning on those enables, it is very important that you initialize these address ranges. To turn off downstream forwarding, set the address base to a value greater than the address limit.
- All master latency and target wait timers are set to zero, which means that no limits are imposed. The implementation-specific burst count is set to zero, so no burst count limits are imposed.
- Cache line size is set to zero. No cache line boundaries are observed, and memory write and invalidate commands are forwarded as memory write commands.
- Parity checking, **p_serr_l** assertion, VGA modes, and ISA modes are all disabled.
- All device-specific features are disabled (reset to 0).

System Initialization

The *PCI-PCI Bridge Architecture Specification* provides guidelines for initialization. Some of this information is included here to aid initialization code development.

When the 21050 or any other PCI-PCI Bridge is present in a system, either on the system board or on a PCI option card, the BIOS is required to provide the following functions during the initialization process:

- Assigning PCI bus numbers
- Allocating address spaces (prefetchable memory, nonprefetchable memory, and I/O)
- Writing the IRQ number into each device
- Initializing the PCI display subsystem

Assigning PCI Bus Numbers

The BIOS must assign PCI bus numbers to each PCI-PCI Bridge in the system. In what order they are assigned and when they are assigned is not specified. The following requirements must be followed:

- All buses located behind a PCI-PCI Bridge must reside between the secondary bus number and the subordinate bus number (inclusive).
- Bus number ranges in parallel branches of PCI bus hierarchy must be mutually exclusive.

Allocating Address Spaces

The BIOS that finds a 21050 must map all devices residing behind the bridge into one of the 21050 address ranges. Address ranges supported by the 21050 are I/O space, prefetchable memory, and nonprefetchable memory (also called memory mapped I/O). The address ranges have the characteristics listed in Table 2.

Table 2 21050 Address Range Characteristics

Address Range	Maximum Range	Minimum Granularity
I/O	64 KByte	4 KByte
Prefetchable memory	4 GByte	1 MByte
Nonprefetchable memory (memory mapped I/O)	4 GByte	1 MByte

Because the 21050 has only one window for each address type, the BIOS must group all devices that use a particular address space into a single window. The address range of a 21050 must include the address range of the same type of all PCI-PCI Bridges and devices behind it. To minimize wasted space, particularly in memory space, it is recommended that you map larger memory requests before smaller memory requests.

Mapping I/O Space Using ISA Mode

When an ISA or EISA bus is in the system, I/O space can become fragmented. An ISA address uses only 10 bits of addressing, giving a maximum 1 KByte of address space. Because ISA devices decode only the lowest 10 bits, these addresses are aliased throughout the entire I/O space. The bottom 256 bytes of a 1 KByte chunk point to system-specific addresses. The top 768 bytes are used for general I/O. EISA systems use the entire 64 KByte space by dividing the space up into 16 4-KByte slots, and use a slot-specific control strobe to point to a particular slot. Slot 0 is generally allocated to the motherboard.

PCI-ISA and PCI-EISA bridges typically perform full 32-bit addressing, using positive decoding for the bottom 256 bytes of the first 1K chunk and subtractive decoding for the upper 768 bytes of each 1K chunk. Although the top 768 bytes of each chunk might be used for general ISA/EISA devices, the bottom 256 bytes in all slots but slot 0 (the bottom 4K slot) are available for use by PCI devices.

The 21050 provides a mode that allows mapping into the low 256 bytes of each 1K chunk. When the ISA Mode on the 21050 is set, it forwards only downstream transactions addressing the low 256 bytes of each 1K chunk inside the defined address range. This allows you to map PCI devices into I/O space when there is an ISA or EISA bus in the system. When mapping a device downstream of the 21050 with the ISA Mode set, bits <9:8> of the device's I/O Base Address should be 00.

Using the 21050 VGA Modes

You can configure the 21050 to support VGA compatible devices.

When a VGA compatible device is located downstream of a PCI-PCI Bridge, set the VGA Mode bit in 21050 configuration space in every PCI-PCI Bridge between the device and the most primary PCI bus. Setting this bit configures the 21050 to pass all VGA frame buffer memory addresses and all VGA I/O addresses downstream.

When a graphics device that does not have VGA compatibility is located behind the 21050, you must configure the bridge to forward only VGA palette writes by setting the VGA Snoop bit in the 21050 configuration space. Again, you must set this bit in every PCI-PCI Bridge between the most primary PCI bus and the graphics device.

If both types of devices are located downstream of the 21050, setting the VGA Mode bit is sufficient for proper operation. However, setting both the VGA Mode and the VGA Snoop bits results in the same behavior as setting only the VGA Mode bit.

For more information on VGA support in PCI systems, refer to the *PCI-PCI Bridge Architecture Specification*.

Sample Initializations

This section provides several sample initializations of the 21050.

Initializing the 21050 for Hierarchical Configuration Accesses

To initialize the 21050 so that configuration transactions can be forwarded, write the following bridge configuration registers (values in hex):

Offset	Register	Value	Notes
04 h	Primary Command/ Status	FFFF0000 h	Clear all status bits Make sure memory, I/O, and master enables are turned off
18 h	Subordinate/ Secondary/ Primary Bus Numbers	00*xxyyzz* h	*xx* = subordinate bus number *yy* = secondary bus number *zz* = primary bus number

The important points are:

- Turn off memory, I/O, and master enables (to prevent memory and I/O transactions from crossing the bridge).
- Initialize PCI bus numbers for forwarding of configuration transactions across the bridge.

Initializing the 21050 for I/O and Configuration Forwarding

To initialize the 21050 so that I/O and configuration transactions can be forwarded, write the following bridge configuration registers (values in hex):

• Offset	Register	Value	Notes
18 h	Subordinate/ Secondary/ Primary Bus Numbers	00*xxyyzz* h	*xx* = subordinate bus number *yy* = secondary bus number *zz* = primary bus number
1C h	Secondary Status/ I/O Limit Addr/ I/O Base Addr	FFFF*x*0*y*0 h	Clear status bits *x* = I/O Limit Address bits <15:12> *y* = I/O Base Address bits <15:12>
20 h	Memory Limit/ Memory Base	0000FFFF h	Turn off downstream memory address range
24 h	PF Mem Limit/ PF Mem Base	0000FFFF h	Turn off downstream PF memory address range
3C h	Bridge Control	000*x*0000 h	*x* = 0: no ISA bus in system *x* = 4: ISA bus in system
04 h	Primary Command/ Status	FFFF0005 h	Clear status bits Turn on I/O enable (for downstream I/O) Turn on master enable (for upstream I/O) Write this register last

The important points are:

- Initialize PCI bus numbers for forwarding of configuration transactions across the bridge.
- Initialize I/O Base and Limit registers to define a downstream window.
- Turn off both memory ranges.
- If an (E)ISA bus is in the system, turn on ISA Mode.
- Turn on I/O and master enables ***(write this register last).***

Note: Setting the master enable bit also enables forwarding of memory transactions upstream.

Initializing the 21050 for Memory and Configuration Forwarding

To initialize the 21050 so that memory and configuration transactions can be forwarded, write the following bridge configuration registers (values in hex):

• Offset	Register	Value	Notes
18 h	Subordinate/ Secondary/ Primary Bus Numbers	00*xxyyzz* h	*xx* = subordinate bus number *yy* = secondary bus number *zz* = primary bus number
1C h	Secondary Status/ I/O Limit Addr/ I/O Base Addr	FFFF00FF h	Clear status bits Turn off I/O address range
20 h	Memory Limit/ Memory Base	*xxx*0*yyy*0 h	*xxx* = nonprefetchable memory limit addr <31:20> *yyy* = nonprefetchable memory base addr <31:20> To disable this range, write 0000FFFF h
24 h	PF Mem Limit/ PF Mem Base	*xxx*0*yyy*0 h	*xxx* = prefetchable memory limit addr <31:20> *yyy* = prefetchable memory base addr <31:20> To disable this range, write 0000FFFF h
04 h	Primary Command/ Status	FFFF0006 h	Clear status bits Turn on memory enable (for downstream memory) Turn on master enable (for upstream memory) Write this register last

The important points are:

- Initialize PCI bus numbers for forwarding of configuration transactions across the bridge.
- Turn off I/O Range.
- Initialize either or both nonprefetchable and prefetchable memory ranges.
- Turn on memory and master enables ***(write this register last)***.

Note: Setting the master enable bit also enables forwarding of the I/O transactions upstream.

Initializing the 21050 for Memory, I/O, and Configuration Forwarding

To initialize the 21050 so that memory, I/O, and configuration transactions can be forwarded, write the following bridge configuration registers (values in hex):

Offset	Register	Value	Notes
18 h	Subordinate/ Secondary/ Primary Bus Numbers	00*xxyyzz* h	*xx* = subordinate bus number *yy* = secondary bus number *zz* = primary bus number
1C h	Secondary Status/ I/O Limit Addr/ I/O Base Addr	FFFF*x*0*y*0 h	Clear status bits *x* = I/O Limit Address bits <15:12> *y* = I/O Base Address bits <15:12>
20 h	Memory Limit/ Memory Base	*xxx*0*yyy*0 h	*xxx* = nonprefetchable memory limit addr <31:20> *yyy* = nonprefetchable memory base addr <31:20> To disable this range, write 0000FFFF h
24 h	PF Mem Limit/ PF Mem Base	*xxx*0*yyy*0 h	*xxx* = prefetchable memory limit addr <31:20> *yyy* = prefetchable memory base addr <31:20> To disable this range, write 0000FFFF h
3C h	Bridge Control	000*x*0000 h	*x* = 0: no ISA bus in system *x* = 4: ISA bus in system
04 h	Primary Command/ Status	FFFF0007 h	Clear status bits Turn on I/O enable (for downstream I/O) Turn on memory enable (for downstream memory) Turn on master enable (for upstream memory and I/O) Write this register last

The important points are:

- Initialize PCI bus numbers for forwarding of configuration transactions across the bridge.
- Initialize the I/O range.
- Initialize either or both nonprefetchable and prefetchable memory ranges.
- Turn on memory and master enables ***(write this register last).***

Initializing VGA Modes in the 21050

The following sections provide examples of initializing VGA mode and VGA Snoop mode.

VGA Mode

If a VGA device resides behind the 21050, enable VGA mode. To enable VGA mode, write bit 18 of configuration address 3C h. Simply modify the contents of the write to this register shown in previous sections. For example:

Offset	Register	Value	Notes
3C h	Bridge Control	000*x*0000 h	*x* = 8: VGA mode, no ISA bus in system *x* = C: VGA mode, ISA bus in system

You must write this register before memory, I/O, and master enables located at Cfg address 04h are set.

VGA Snoop Mode

If a graphics device requiring VGA Snoop mode resides behind the 21050, you must write bit 5 of configuration address 04 h. Simply modify the contents of the write to this register shown in previous sections. For example:

Offset	Register	Value	Notes
04 h	Primary Status/ Command	FFFF002*x* h	*x* = 7: Set memory, I/O and master enable[1] *x* = 5: Set I/O and master enable[2]

[1]If all memory and I/O ranges are used

[2]If only the I/O range is used

Initializing Optional Registers in the 21050

This section discusses initialization of optional 21050 features. The features are not required for basic bridge operation. It is recommended that you write these registers before (or in the same access as) setting the memory, I/O, and master enables in configuration address 04h. Otherwise, order is not important.

Enabling Parity and Error Checking

The following accesses enables parity checking on both interfaces, as well as forwarding of SERR# from the secondary to the primary interface.

Offset	Register	Value	Notes
3C h	Bridge Control	000*x*0000 h	*x* = 3: no ISA bus in system, no VGA behind 21050 *x* = 7: ISA bus in system, no VGA behind 21050 *x* = B; no ISA bus in system, VGA behind 21050 *x* = F; ISA bus in system, VGA behind 21050
04 h	Primary Command/ Status	FFFF01*xy* h	Clear status bits *x* = 4: No VGA Snoop mode *x* = 6: VGA Snoop mode *y* = [0,5,6,7]: [Cfg Only, I/O only, Mem Only, All] enabled for downstream forwarding Write this register last.

Enabling Timers and Set Cache Line Size

The following accesses enable master latency timers, target wait timers, and burst count to desired values. Setting of cache line size is also shown. Writing a value of zero disables the timers and cache line size.

Offset	Register	Value	Notes
0C h	Primary MLT/ Cache Line Size	0000*xxyy* h	*xx* = pri. master latency timer value (low 3 bits = 0) *yy* = Cache line size
18 h	Secondary MLT/ Subordinate/ Secondary/ Primary Bus Numbers	*wwxxyyzz* h	*ww* = sec. master latency timer value (low 3 bits = 0) *xx* = subordinate bus number *yy* = secondary bus number *zz* = primary bus number
40 h	Burst Count Limit/ Chip Control	00*xx*0000 h	*xx* = burst count limit (top 2 bits = 0)
44 h	Sec. Wait Timer/ Pri Wait Timer	0000*xxyy* h	*xx* = Secondary wait timer value *yy* = Primary wait timer value

Selecting Other Options

You can select other implementation-specific options by writing bits in the Chip Control register (Configuration address **40h**).

Field	Feature	Value	Notes
<0>	Secondary arbitration mode	00000001 h	Enable arbiter to perform in strict rotation mode Disable posting
<3:2>	Request mask timer	0000000*x* h	*x<3:2>* : encoded timer value

Technical Support, Ordering, and Associated Literature

This section describes how to order DECchip information and technical support, as well as how to order DECchip products and associated literature.

Calling the DECchip Information Line for Information and Technical Support

Use the following numbers to call the DECchip Information Line:

United States and Canada	**1-800-332-2717**
TTY (United States only)	**1-800-332-2515**
Outside North America	**+1-508-568-6868**

Ordering DECchip Products

To order the DECchip 21050 PCI-to-PCI Bridge, contact your local Digital sales office. Your sales representative may be able to help you take advantage of discounts and volume pricing.

You can order the following DECchip products from Digital.

Product	Order Number
DECchip 21050 PCI-to-PCI Bridge	21050-AA

Ordering Associated DECchip Literature

The following table lists some of the DECchip literature that is available. For a complete list and for information about ordering, contact the DECchip Information Line.

Title	Order Number
DECchip 21050 PCI-to-PCI Bridge Data Sheet	EC–Q9XVA–TE
DECchip 21050 PCI-PCI Bridge Product Brief	EC–Q9Y0A–TE
DECchip 21050 PCI-PCI Bridge Configuration: An Application Note	EC–Q9Y4A–TE
DECchip 21050 PCI-PCI Bridge Hardware Implementation: An Application Note	EC–Q9Y3A–TE

Ordering Third-Party Literature

You can order the following third-party literature directly from the vendor.

Title	Vendor
PCI Local Bus Specification, Revision 2.0 *PCI-to-PCI Bridge Architecture Specification,* Revision 1.0	PCI Special Interest Group M/S HF3-15A 5200 N.E. Elam Young Pkwy Hillsboro, Oregon 97124-6497 (503) 696-2000