DECchip 21052 PCI-to-PCI Bridge Hardware Implementation: An Application Note

Order Number: EC–QLZAA–TE

This application note presents guidelines for hardware implementation of the DECchip 21052.

Revision/Update Information: This is a new document.

Digital Equipment Corporation
Maynard, Massachusetts

July 1995

This document was prepared using DECwrite, Version 3.0.

Contents

Introduction 1
Functional Overview 1
5-V and 3.3-V Signaling 2
21052 Power Supply 2
General Layout Guidelines 3
- Pull-Ups 3
- Expansion Card Routing 4

Clocking 4
- 21052 Clocking 4
- 21052 Output Clocks 4
- Clock Implementation on the Motherboard 5
- Clock Implementation Using Secondary Clocks 5

Secondary IDSEL Mapping 6
Data Synchronization and Interrupts 7
Controlling Write Posting 7
Interrupt Binding on Option Cards 8
Arbitration 9
- Primary Bus Arbitration 9
- Secondary Bus Arbitration 9

Technical Support and Ordering Information 10
- Ordering Digital Semiconductor Products 10
- Ordering Associated Literature 10
- Ordering Third-Party Literature 10

List of Figures

Figure 1 21052 Voltage Regulator Circuit 3
Figure 2 Secondary IDSEL Implementation Example 7

List of Tables

Table 1 p_vio and s_vio Connections 2
Table 2 Voltage Regulator Vendors 3
Table 3 Low-Skew Clock Buffers 5
Table 4 Device Number to Secondary IDSEL Mapping 6
Table 5 Interrupt Binding on Option Cards 8

Introduction

This document presents guidelines for hardware implementation of the DECchip 21052 chip (21052) in a system. This application note is limited to hardware implementation of the 21052 only and does not cover any devices that might be behind the 21052, or initialization code needed to configure the DECchip 21052 chip.

This application note includes implementation notes on layout, clocking, secondary bus IDSEL mapping, interrupt routing, and secondary bus arbitration. Implementation on both a motherboard and an option card is covered. For most situations, hardware implementation issues are the same. In addition, guidelines for interrupt routing on option cards are provided in the interrupt discussion.

The following related Digital Equipment Corporation documents may be useful:

- *DECchip 21052 PCI-to-PCI Bridge Data Sheet*
- *DECchip 21052 PCI-to-PCI Bridge Configuration: An Application Note*
- *DECchip 21052 PCI-to-PCI Bridge Evaluation Board User's Guide*

The following PCI Special Interest Group documents may also be useful:

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

See the Technical Support and Ordering Information section at the end of this document for ordering information.

Functional Overview

The 21052 PCI-to-PCI Bridge provides a connection between two independent PCI buses. The PCI buses operate concurrently, except when read or nonposted write transactions are crossing the bridge. The two PCI buses are referred to as the **primary PCI bus**, which is the PCI bus closest to the CPU, and the **secondary PCI bus**, which is the PCI bus farther from the CPU.

The 21052 has two common applications. System designers can use the 21052 on a motherboard to add more devices or add-in card slots than a single PCI bus can support. Add-in card designers can use the 21052 to enable multiple device option cards. (PCI add-in cards are restricted to a single connection to the signals in the PCI connector.) Designers can also use the 21052 to isolate bus traffic on one PCI bus segment from other PCI bus segments.

The 21052 PCI-PCI Bridge has the following interfaces:

- **Primary and secondary PCI bus interfaces**

  Control and data to two PCI buses, including LOCK#, PERR#, and SERR#. To implement these interfaces, follow the guidelines in the *PCI Local Bus Specification, Revision 2.0*. The 21052 primary interface implements Revision 2.0 compliant 3.3-V drivers.

- **Secondary bus arbiter**

  Four request/grant pairs for secondary bus master control, plus one pin, **s_cfn_l**, to enable the arbiter.

- **Clocks**

  One primary interface clock input (**p_clk**), one secondary interface clock input (**s_clk**), and five secondary interface clock outputs (**s_clk_o<4:0>**).

- **Diagnostic pins**

  One test input **(goz_l**) that tristates all outputs, and one test output **(nand_out**) that is the output of the diagnostic nand tree.

5-V and 3.3-V Signaling

Both the primary and secondary PCI buses can be independently interfaced to either the 3.3-V or the 5-V signaling environment by connecting the **p_vio** and **s_vio** pins to the appropriate voltages, as shown in Table 1.

Table 1 p_vio and s_vio Connections

Signaling Environment			
Primary Bus	Secondary Bus	p_vio (pin 67)	s_vio (pin 52)
5 V	5 V	5 V	5 V
5 V	3.3 V	5 V	3.3 V
3.3 V	5 V	3.3 V	5 V
3.3 V	3.3 V	3.3 V	3.3 V

The 21052 I/O pads are 5-V tolerant and will operate under both 3.3-V and 5-V signaling environments. A clamp circuit has been implemented in the drivers that turns on when the output voltage matches the voltage on the **p_vio** and **s_vio** pins. These pins should be connected to the appropriate voltages, depending on the signaling environment, as described in Table 1.

It is guaranteed by design that the 21052 is reliable in both 3.3-V and 5-V environments if **p_vio** and **s_vio** are connected in accordance with Table 1.

21052 Power Supply

The 21052 must be powered by a 3.3-V power supply. If 3.3 V is not available in the system, it can be generated from a 5-V power supply by using a voltage regulator. Figure 1 shows a recommended regulator circuit. A tantalum electrolytic capacitor of at least 10 µF is required at the output of the regulator.

Figure 1 21052 Voltage Regulator Circuit

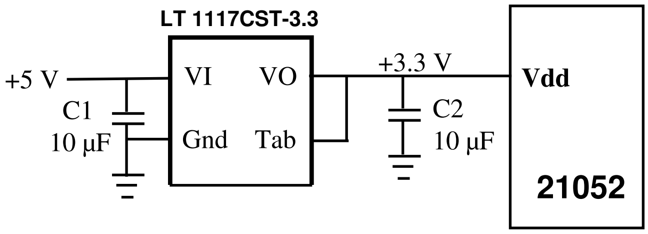


Table 2 lists the voltage regulators that can be used. To implement any of these regulators, refer to the vendor's data sheet.

Table 2 Voltage Regulator Vendors

Vendor	Part Number
Linear Technology	LT1117CST-3.3
Texas Instruments	TLV2217-33
Motorola	MC33269D
National Semiconductor	LM3940

General Layout Guidelines

When using the 21052, you need to consult the general layout guidelines provided in the *PCI Local Bus Specification Revision 2.0*. Clock routing has some special requirements. Guidelines and requirements for clock routing are discussed in the Clocking section of this document. Guidelines for routing of secondary IDSEL signals are given in the Secondary IDSEL Mapping section. The following sections discuss pull-ups and expansion card routing.

Pull-Ups

Pull-ups are required on the following shared PCI control signals: FRAME#, IRDY#, TRDY#, STOP#, DEVSEL#, PERR#, SERR#, and LOCK#. These signals must have pull-ups on both the primary and secondary PCI buses, and they should be pulled up to **p_vio** or **s_vio** voltages appropriately. When optional shared signals such as LOCK# are not used, they *must* be pulled up for proper 21052 operation. For resistor values, use the formula specified in Section 4.3.3 of the *PCI Local Bus Specification Revision 2.0*. In this formula, plug in a value of 1.5 mA for *Iol* (per specifications in the *DECchip 21052 PCI-to-PCI Bridge Data Sheet*) and the appropriate value for *Vcc (*which should be the same as the **s_vio** voltage).

If the 21052 is implemented on a motherboard, you must place both primary and secondary bus pull-ups on the motherboard. If the 21052 is implemented on an option card with the primary bus interfacing to the card edge, place primary bus pull-ups on the motherboard and place secondary bus pull-ups on the option card. In addition, **s_req_l<3:0>** must be pulled up to **s_vio**. The recommended value for the pull-up resistors is 5 kΩ.

Point-to-point signals, except **s_req_l<3:0>**, and shared 32-bit signals do not require pull-ups.

Expansion Card Routing

Follow the guidelines and requirements for routing on expansion cards that are provided in the *PCI Local Bus Specification, Revision 2.0*. This section highlights the most important requirements.

PCI signals coming from the motherboard on the expansion card must be limited to only one load. This includes the primary CLK. These are the signals on the primary interface of the 21052. These signals also have trace length limitations, which are 1.5 inches for PCI signals and 2.5 inches for the primary CLK.

PCI signals on the secondary side of the 21052, however, do not need to adhere to these restrictive loading and trace length requirements. The secondary PCI bus can support the full 10 loads (including the 21052) and essentially can be treated like a motherboard PCI bus.

Clocking

The following sections provide an overview of 21052 clocking requirements, and explain how to implement clocks by using the 21052 on a motherboard and an option card.

21052 Clocking

The 21052 has two clocking domains: one for the primary PCI interface and one for the secondary PCI interface. Both interfaces must operate at the same frequency and must be synchronous to each other.

The primary clocking domain is controlled by the **p_clk** input.

The secondary bus arbiter pins operate in the secondary interface clocking domain. The secondary clocking domain is controlled by the **s_clk** input.

The relationship between the **p_clk** and **s_clk** inputs has the following restrictions:

- Both interfaces must operate at the same frequency.
- The maximum clock frequency is 33 MHz.
- The maximum delay between **p_clk** and **s_clk** is 7 ns for both rising and falling edges.
- The minimum delay between **p_clk** and **s_clk** is 0 ns, that is, **s_clk** cannot precede **p_clk** for both rising and falling edges.
- The input duty cycle of **p_clk** must be between 43% to 57%, measured at 1.5 V.

21052 Output Clocks

The 21052 generates five secondary bus clock outputs, **s_clk_o<4:0>**. These clocks are derived from **p_clk**. Use one of the secondary clock outputs as the secondary clock input to the 21052, and use the four other outputs for devices on the secondary bus.

These **s_clk_o<4:0>** clock outputs are generated directly from **p_clk,** and their relationship to **p_clk** has the following properties:

- All clock outputs operate at the same frequency as **p_clk**.
- The maximum delay between **p_clk** and **s_clk_o** is 5 ns for both rising and falling edges.
- The minimum delay between **p_clk** and **s_clk_o** is 0 ns for both rising and falling edges.
- The maximum propagation delay difference from the clock going low to the clock going high, with respect to the primary clock, is ±750 ps. As a result, the secondary clock duty cycle skews by a maximum of 750 ps.
- Unused secondary clock signals must be terminated. This can be done with a 100-Ω series-termination resistor and a 50-pF shunt capacitor to ground.

Clock Implementation on the Motherboard

When the 21052 is implemented on a motherboard and the secondary bus is routed to one or more PCI expansion slots or to another 21052, you can use the 21052 secondary clock outputs **s_clk_o<4:0>**, provided the designer can guarantee that the output clocks will meet PCI duty cycle specifications. The designer must include the 750-ps skew added by the 21052 to the duty cycle in their analysis.

If you choose not to use the secondary clock outputs, you can substitute a low-skew clock buffer to generate secondary clocks. Table 3 lists some buffers that can be used.

Table 3 Low-Skew Clock Buffers

Vendor	Part Number
Texas Instruments	CDC328A
National Semiconductor	CGS74B2525
IDC	1DT74FCT805CT

One of the secondary clock signals must be connected to the 21052 secondary clock input **s_clk**. Plan the layout of clocks to minimize skew between the 21052 and other PCI devices.

Clock Implementation Using Secondary Clocks

Follow these guidelines when implementing secondary clock outputs:

- Connect one **s_clk_o<4:0>** output to the 21052 **s_clk** input.
- Route **s_clk_o<4:0>** to minimize skew, that is, use the same amount of etch for the **s_clk_o<4:0>** outputs that are used as **s_clk** inputs as you use for the other **s_clk_o<4:0>** signals. Digital recommends that the amount of delay due to etch be limited to less than 2 ns (10 in if etch has a delay of 200 ps/in).

- Digital recommends that you use a series-termination resistor to terminate the **s_clk_o<4:0>** outputs to limit reflections on the clock lines. The value of the resistor will depend on the transmission line impedance of the etch, which will depend on the board layout. As a result, it is difficult to recommend a precise value. Based on a 75-Ω transmission line, a 50-Ω series resistor provides acceptable clock waveforms.

- Digital recommends that you bury the secondary clocks in internal layers. The spacing of the clock signals to adjacent signals should be more than the minimum, to eliminate cross talk.

Secondary IDSEL Mapping

The *PCI Local Bus Specification Revision 2.0* defines two formats for configuration transactions in hierarchical systems:

- Type 0 configuration transactions are used to configure devices on the same PCI bus.

- Type 1 configuration transactions are used to configure devices that reside on a downstream PCI bus.

When the 21052 detects a Type 1 configuration transaction for a device connected to the secondary interface of the bridge, the 21052 translates the Type 1 transaction into a Type 0 transaction on the downstream interface. Instead of using IDSEL to identify the target of a configuration transaction, the Type 1 configuration format uses a 5-bit field at bits <15:11> in the address as a device number. A device number in the Type 1 format is translated by the 21052 into an IDSEL line for Type 0 transactions on the target interface.

As required by the *PCI-to-PCI Bridge Architecture Specification Revision 1.0,* the 21052 uses **s_ad<31:16>** as secondary IDSEL lines. The 21052 uses the mapping shown in Table 1 for translation of a device number to an **s_ad** pin. The 21052 supports mapping for device numbers 0 through F (hex). Configuration transactions with device numbers outside of this range are still forwarded, but no IDSEL mapping is performed (**s_ad<31:16>** are 00h).

Table 4 Device Number to Secondary IDSEL Mapping

Device Number p_ad<15:11>	Secondary IDSEL s_ad<31:16>	Device Number p_ad<15:11>	Secondary IDSEL s_ad<31:16>
00000	0000 0000 0000 0001	01010	0000 0100 0000 0000
00001	0000 0000 0000 0010	01011	0000 1000 0000 0000
00010	0000 0000 0000 0100	01100	0001 0000 0000 0000
00011	0000 0000 0000 1000	01101	0010 0000 0000 0000
00100	0000 0000 0001 0000	01110	0100 0000 0000 0000
00101	0000 0000 0010 0000	01111	1000 0000 0000 0000
00110	0000 0000 0100 0000	10000—11110	0000 0000 0000 0000
00111	0000 0000 1000 0000	11111	0000 0000 0000 0000 if **p_ad<7:2>** > 00h *Generate special cycle* if **p_ad<7:2>** = 00h
01000	0000 0001 0000 0000		
01001	0000 0010 0000 0000		

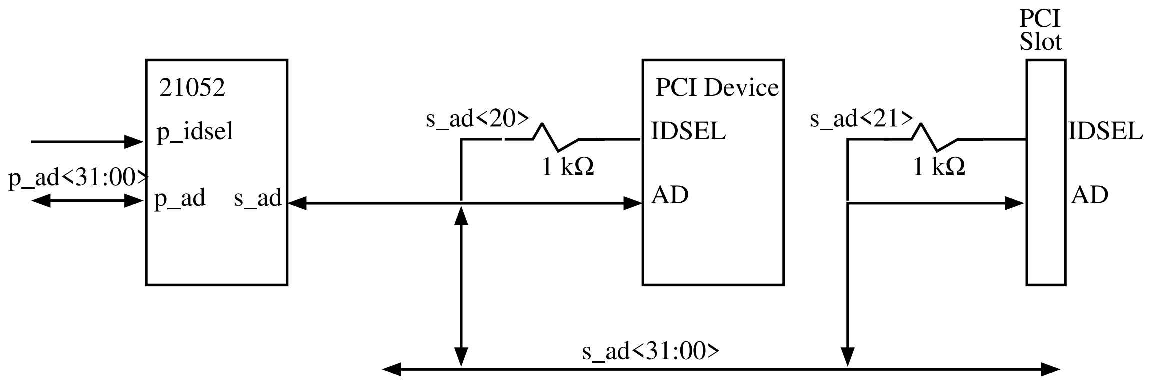
When implementing the 21052 in a system, each PCI device and PCI slot connected to the secondary PCI bus must have its IDSEL line connected to one of the **s_ad<31:16>** lines. To reduce the capacitive load on the **s_ad** line, this connection can be done through a 1-kΩ resistor, as shown in Figure 1. The 21052 supports address stepping on the address cycle of Type 0 transactions to allow additional time for the IDSEL signal to become valid when resistively coupled. This is the only time that the 21052 uses stepping, because the value of the IDSEL for data cycles and other transaction types is not important. If you think that the **s_ad** bus can support the additional capacitive load of the IDSEL pin, then you do not have to use the resistor.

When you assign secondary AD signals to secondary IDSEL signals on an expansion card, make sure that the device numbers are consistent with the required interrupt binding given in Table 2.

Note

Some early PCI host bridges automatically claim all configuration commands directed to device 0. Therefore, Digital recommends that you avoid using device 0, if this situation might apply.

Figure 2 Secondary IDSEL Implementation Example



Data Synchronization and Interrupts

The *PCI Local Bus Specification Revision 2.0* requires that either the interrupt handler (service routine) or the device that initiates the interrupt guarantees that all buffers are flushed between the device and the final destination. To accomplish this, the interrupt service routine of the device driver can perform a read of the device, or the device itself can perform a read of the last location written by the device. In either case, the read transaction forces buffers between the device and the final destination to be flushed.

Section 6.3.4 of the *PCI Local Bus Specification, Revision 2.0* states that device drivers are ultimately responsible for guaranteeing consistency of data and interrupts.

Interrupts originating from secondary bus devices are not routed through the 21052.

Controlling Write Posting

You can configure the 21052 to always enable or always disable memory write posting by resetting or setting respectively bit <1> in the chip control register in 21052 configuration space, address 40h.

Interrupt Binding on Option Cards

When PCI-to-PCI Bridges are used in option cards, a binding is required by the *PCI-to-PCI Bridge Architecture Specification Revision 1.0* between the device number (as given in the Type 1 configuration address and, therefore, the IDSEL line) and the INT*x*# line it uses when requesting an interrupt. The PCI connector has only four interrupt lines assigned to it: INTA#, INTB#, INTC#, and INTD#. Multiple devices might have to share these lines. Refer to Table 5.

Because only the BIOS knows how the PCI INT*x*# lines are routed to the system interrupt controller, a mechanism is required to inform the device driver of which IRQ its device will request an interrupt on. The interrupt line register stores this information. The BIOS code assumes the binding as listed in Table 4 behind a PCI-to-PCI Bridge and writes the IRQ number in each device. The interrupt binding defined in Table 4 is mandatory for option cards utilizing PCI-to-PCI Bridges.

Table 5 Interrupt Binding on Option Cards

Device Number on Secondary Bus	Interrupt Pin on Device	Interrupt Pin on Connector
0, 4, 8, 12, 16, 20, 24, 28	INTA#	INTA#
	INTB#	INTB#
	INTC#	INTC#
	INTD#	INTD#
1, 5, 9, 13, 17, 21, 25, 29	INTA#	INTB#
	INTB#	INTC#
	INTC#	INTD#
	INTD#	INTA#
2, 6, 10, 14, 18, 22, 26, 30	INTA#	INTC#
	INTB#	INTD#
	INTC#	INTA#
	INTD#	INTB#
3, 7, 11, 15, 19, 23, 27, 31	INTA#	INTD#
	INTB#	INTA#
	INTC#	INTB#
	INTD#	INTC#

Arbitration

The following sections describe primary and secondary bus arbitration.

Primary Bus Arbitration

The 21052 provides request (**p_req_l**) and grant (**p_gnt_l**) signals for primary bus access arbitration in accordance with the *PCI Local Bus Specification Revision 2.0*. These signals connect to external arbiter logic provided by the system.

Secondary Bus Arbitration

You can configure the 21052 to use either its internal arbiter or an external arbiter for secondary bus access arbitration. The 21052 internal secondary bus arbiter supports four bus masters plus the 21052. The 21052 has four request inputs (**s_req_l<3:0>**) and four grant outputs (**s_gnt_l<3:0>**). When the internal arbiter is used, 21052 request and grant lines are internal to the chip.

To use the 21052 secondary bus arbiter:

- Tie the central function pin, **s_cfn_l**, to ground (low) through a 1-kΩ resistor.
- Connect secondary bus masters to the request and grant line pairs. The 21052 has two arbitration modes, both rotating, so order is not important (initial highest priority is at **s_req_l<0>**, however). Add 5-kΩ resistors from **s_req_l<3:0>** to **s_vio**. You can leave unused **s_gnt_l** output pins unconnected. These signals are compliant with the REQ# and GNT# specifications in the *PCI Local Bus Specification Revision 2.0* and should be treated accordingly.
- The default mode rotates highest priority among all four external bus masters, but gives the 21052 highest priority on alternate transactions. You can select a second mode, which uses rotating priority among all bus masters including the 21052. Set this alternate mode by writing a 1 to the arbitration mode bit in the 21052 configuration space (cfg addr 40h, bit 0).

To use an external secondary bus arbiter:

- Tie the central function pin, **s_cfn_l**, high through a 1-kΩ resistor
- The 21052 reconfigures the **s_req_l<0>** and **s_gnt_l<0>** pins to act as external grant and request pins. Because **s_gnt_l<0>** is an output pin, the 21052 uses this pin as an external secondary bus request line. Connect this pin to one of the request lines of the external arbiter. Because **s_req_l<0>** is an input pin, the 21052 uses this pin as an external secondary grant line. Connect this pin to one of the grant lines of the external arbiter. Tie all **s_req_l<3:1>** high through a 1-kΩ resistor. You can leave **s_gnt_l<3:1>** unconnected.
- The external arbiter must use some kind of fairness algorithm. If a fixed priority algorithm is used, a deadlock condition might occur.
- The secondary bus arbiter selects the device that will park the secondary bus during normal operations by asserting a single GNT# line. However, the 21052 will always drive **s_ad**, **s_cbe_l**, and **s_par** low during reset, even if the internal arbiter is disabled.

Technical Support and Ordering Information

If you need technical support or help deciding which literature best meets your needs, call the Digital Semiconductor Information Line:

United States and Canada	**1-800-332-2717**
Outside North America	**+1-508-628-4760**

Ordering Digital Semiconductor Products

To order the DECchip 21052 PCI-to-PCI Bridge, contact your local distributor.

You can order the following Digital Semiconductor product from Digital:

Product	Order Number
DECchip 21052 PCI-to-PCI Bridge	21052-AA

Ordering Associated Literature

The following table lists some of the Digital Semiconductor literature that is available. For a complete list, contact the Digital Semiconductor Information Line.

Title	Order Number
DECchip 21052 PCI-to-PCI Bridge Data Sheet	EC–QHURA–TE
DECchip 21052 PCI-to-PCI Bridge Product Brief	EC–QHUQA–TE
DECchip 21052 PCI-to-PCI Bridge Configuration: An Application Note	EC–QLZBA–TE
DECchip 21052 PCI-to-PCI Bridge Evaluation Board User's Guide	EC–QKHJA–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, OR 97124-6497 (503) 696-2000