IDT PCI Express® 24-Port 32-Lane Gen 2 Switch Hardware Design Guide

Application Note AN-727

Notes

By Bryan Le

Introduction

This application note provides system design guidelines for IDT's PCI Express® 2.0 base specification compliant System Interconnect switch device family. Information provided in this document is applicable to the following devices: 89HPES32NT24A[B]G2, 89HPES32NT8A[B]G2, 89HPES24NT24[A]G2, 89HPES24NT6AG2, 89HPES22NT16G2, 89HPES16NT16G2, 89HPES16NT2G2 and 89HPES12NT12G2. In this document, the PES32NT24AG2 is used as the primary reference. The letters "G2" within the device names indicate that these devices are capable of GEN2 (5.0 GT/S) serial data speeds. The PES32NT24AG2 device offers 32 PCIe lanes divided into 24 ports. The PES24NT24G2 device offers 24 PCIe lanes divided into 24 ports, and so on.

This document also describes the following device interfaces and provides relevant board design recommendations:

1) PCI Express Interface
2) Reference Clock (REFCLK) Circuitry
3) Reset (Fundamental Reset) Schemes
4) SMBus Interfaces
5) GPIO and JTAG pins
6) Power and Decoupling Scheme
7) Switch Partitioning

PCI Express Interface

Port Configuration

Eight of the twenty four ports of the PES32NT24AG2 are statically allocated 2 lanes with ports labeled from 0 through 7 and the remaining 16 ports are statically allocated 1 lane with ports labeled from 8 through 23. In a default configuration, SWMODE[3:0] = 0x0, Port 0 is always the upstream port while the remaining ports are always downstream ports. In a Multi-partition configuration, SWMODE[3:0] = 0xC, or a Multi-partition with Serial EEPROM initialization configuration, SWMODE[3:0] = 0xD, all ports come up as unattached. Through a Serial EEPROM or Slave SMBus interface, ports can be configured as an upstream port, upstream port with NT function, upstream port with NT and DMA functions, NT function, NT with DMA functions, or as downstream ports. All ports can operate at a maximum link width of x2 (i.e. 2 lanes) or x1 (i.e. 1 lane) and support both 2.5 GT/S (Gen1) and 5.0 GT/S (Gen2) speeds.

Per the PCIe® specification, each switch port is viewed as a virtual PCI-PCI bridge device. In the PES32NT24AG2, PCI device numbering follows the port numbering. Port 0 corresponds to Device 0 on the upstream bus. Port 1 corresponds to Device 1 on the PES32NT24AG2 virtual PCI bus, Port 2 to Device 2, and so on.

> Note: Unused PCIe TX and RX lanes are not required to have a termination and can be left open.

Notes

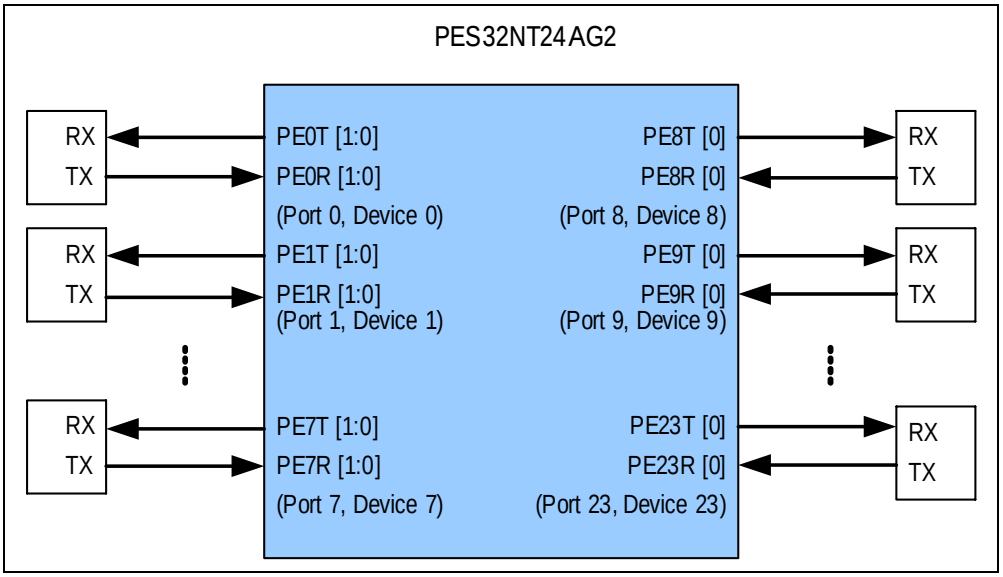


Figure 1 Port Numbering and Device Numbering

The switch contains four stack blocks labeled Stack 0, Stack 1, Stack 2, and Stack 3. Stacks 0 and 1 have four ports each, and stacks 2 and 3 have eight ports each. This provides a total of 24 ports in the device, labeled port 0 through port 23. Table 1 lists the ports associated with each stack.

Stack	Port Associated with the Stack
Stack 0	0, 1, 2, 3
Stack 1	4, 5, 6, 7
Stack 2	8, 9, 10, 11, 12, 13, 14, 15
Stack 3	16, 17, 18, 19, 20, 21, 22, 23

Table 1 Ports in Each Stack

Stacks 0 and 1 may be configured as four x2 ports, two x4 ports, one x8 port, and combinations in between. Stacks 2 and 3 may be configured as eight x1 ports, four x2 ports, two x4 ports, one x8 port, and combinations in between. The configuration of each stack is controlled by the Stack Configuration(STK[3:0]CFG) registers. These registers are located in the Switch Configuration and Status space (see Device User Manual Chapter 20). Stacks 0 and 1 support five possible configurations each. Stacks 2 and 3 support 26 possible configurations each. Tables 3.4 through 3.7 below show the possible configurations for each stack.

- Each STKxCFG register controls the configuration of the corresponding stack (e.g., STK0CFG controls the configuration of Stack 0, STK1CFG for Stack 1, etc.)
- Stack configurations not shown in the table are not allowed. Programming the STKxCFG register to values not shown in the table produces undefined results.

Notes

STKCFG Field in the STK0CFG Register		Stack Configuration			
Hex	Binary	Port 3	Port 2	Port 1	Port 0
0x0	0b00000				x8
0x1	0b00001		x4		x4
0x2	0b00010	x2	x2		x4
0x3	0b00011	x2	x2	x2	x2
0x6	0b00110		x4	x2	x2
Others		Reserved			

Table 2 Possible Configuration for Stack 0

STKCFG Field in the STK1CFG Register		Stack Configuration			
Hex	Binary	Port 7	Port 6	Port 5	Port 4
0x0	0b00000				x8
0x1	0b00001		x4		x4
0x2	0b00010	x2	x2		x4
0x3	0b00011	x2	x2	x2	x2
0x6	0b00110		x4	x2	x2
Others		Reserved			

Table 3 Possible Configuration for Stack 1

STKCFG Field in the STK2CFG Register		Stack Configuration							
Hex	Binary	P15	P14	P13	P12	P11	P10	P9	P8
0x0	0b00000								x8
0x1	0b00001				x4				x4
0x2	0b00010		x2		x2				x4
0x3	0b00011		x2		x2		x2		x2
0x6	0b00110				x4		x2		x2
0x8	0b01000				x4	x1	x1		x2
0x9	0b01001				x4	x1	x1	x1	x1
0xA	0b01010		x2		x2		x2	x1	x1
0xB	0b01011		x2		x2	x1	x1		x2
0xC	0b01100	x1	x2		x2				x4
0xD	0b01101	x1	x1	x1	x1				x4
0xE	0b01110		x2	x1	x1		x2		x2

Table 4 Possible Configuration for Stack 2 (Page 1 of 2)

Notes

STKCFG Field in the STK2CFG Register		Stack Configuration							
0xF	0b01111	x1	x1		x2		x2		x2
0x10	0b10000		x2		x2	x1	x1	x1x	1
0x11	0b10001		x2	x1	x1	x1	x1		x2
0x12	0b10010		x2	x1	x1	x1	x1	x1	x1
0x13	0b10011	x1	x1		x2	x1	x1	x1	x1
0x14	0b10100	x1	x1	x1	x1		x2		x2
0x15	0b10101	x1	x1		x2		x2	x1x	x1
0x16	0b10110	x1	x1	x1	x1		x2	x1	x1
0x17	0b10111	x1	x1	x1	x1	x1	x1		x2
0x18	0b11000				x4		x2	x1	x1
0x19	0b11001		x2	x1	x1		x2	x1	x1
0x1A	0b11010	x1	x1		x2	x1	x1		x2
0x1B	0b11011	x1	x1	x1	x1	x1	x1	x1	x1
0x1C	0b11100		x2	x1	x1				x4
Others		Reserved							

Table 4 Possible Configuration for Stack 2 (Page 2 of 2)

STKCFG Field in the STK3CFG Register		Stack Configuration							
Hex	Binary	P23	P22	P21	P20	P19	P18	P17	P16
0x0	0b00000								x8
0x1	0b00001				x4				x4
0x2	0b00010		x2		x2				x4
0x3	0b00011		x2		x2		x2		x2
0x6	0b00110				x4		x2		x2
0x8	0b01000				x4	x1	x1		x2
0x9	0b01001				x4	x1	x1	x1	x1
0xA	0b01010		x2		x2		x2	x1	x1
0xB	0b01011		x2		x2	x1	x1		x2
0xC	0b01100	x1	x2		x2				x4
0xD	0b01101	x1	x1	x1	x1				x4
0xE	0b01110		x2	x1	x1		x2		x2
0xF	0b01111	x1	x1		x2		x2		x2
0x10	0b10000		x2		x2	x1	x1	x1x	1
0x11	0b10001		x2	x1	x1	x1	x1		x2
0x12	0b10010		x2	x1	x1	x1	x1	x1	x1

Table 5 Possible Configuration for Stack 3 (Page 1 of 2)

Notes

STKCFG Field in the STK3CFG Register		Stack Configuration							
0x13	0b10011	x1	x1		x2	x1	x1	x1	x1
0x14	0b10100	x1	x1	x1	x1		x2		x2
0x15	0b10101	x1	x1		x2		x2	x1x	x1
0x16	0b10110	x1	x1	x1	x1		x2	x1	x1
0x17	0b10111	x1	x1	x1	x1	x1	x1		x2
0x18	0b11000				x4		x2	x1	x1
0x19	0b11001		x2	x1	x1		x2	x1	x1
0x1A	0b11010	x1	x1		x2	x1	x1		x2
0x1B	0b11011	x1	x1	x1	x1	x1	x1	x1	x1
0x1C	0b11100		x2	x1	x1				x4
Others		Reserved							

Table 5 Possible Configuration for Stack 3 (Page 2 of 2)

A stack may be configured statically using the corresponding Stack Configuration (STKxCFG) pins. These pins are sampled by the switch as part of the boot-configuration vector during switch fundamental reset. The STKxCFG pins determine the initial value of the STKCFG field in the corresponding STKxCFG register. The encoding of the STKxCFG pins is identical to that of the STKCFG field shown in Tables 2 through 5.

- For Stacks 0 and 1, the STKxCFG pins have 2 bits each (i.e., STK0CFG[1:0] and STK1CFG[1:0]). These bits correspond to the two least significant bits of the STKCFG field in the correspondingSTKxCFG register. Therefore, for these stacks, configurations 0x0 through 0x3 may be selected statically. Other configurations must be selected dynamically via EEPROM or Slave SMBus interface.
- For Stacks 2 and 3, the STKxCFG pins have 5 bits each. Therefore, all 26 possible configura-tionsmay be selected statically.
- Note that for all stacks the STKxCFG[2] pin can be used to select between a stack configuration and its mirror image. For example, when the STKxCFG pins are set to 0b00010, the stack iscon-figured per configuration 0x2 (ports are configured as x4, x2, x2). The setting 0b00110 yields the mirror image which corresponds to stack configuration 0x6 (ports are configured as x2, x2, x4).

Lane Reversal

The PES32NT24AG2 supports automatic lane reversal outlined in the PCIe specification. This allows trace routing flexibility to avoid crossovers and potentially reduces the number of trace vias required for signal routing. Lane reversal must be done for both the transmitter and the receiver of a port.

Lane reversal mappings for the various non-trivial x8 maximum link width configurations are illustrated in Figures 2 through 4. In the figures, PExRP[n] refers to the pin associated with lane 0 of port'x'.

Notes

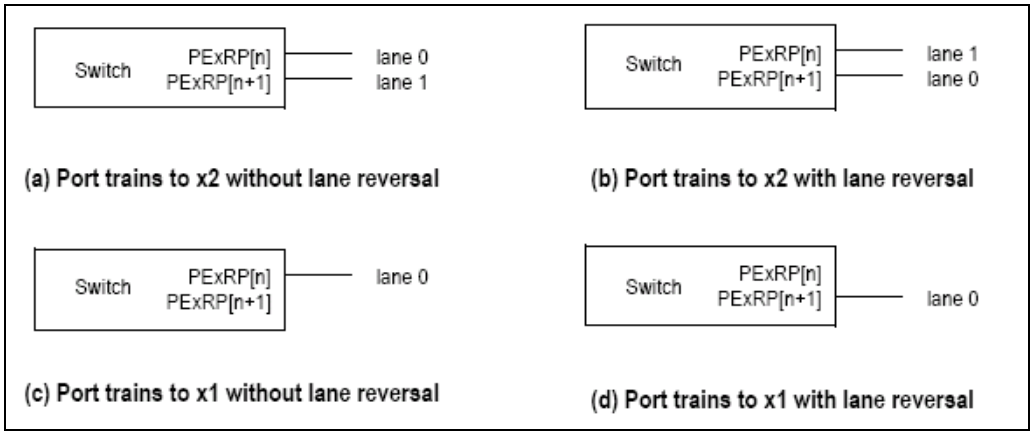


Figure 2 Lane Reversal for Highest Achievable Link Width of x2

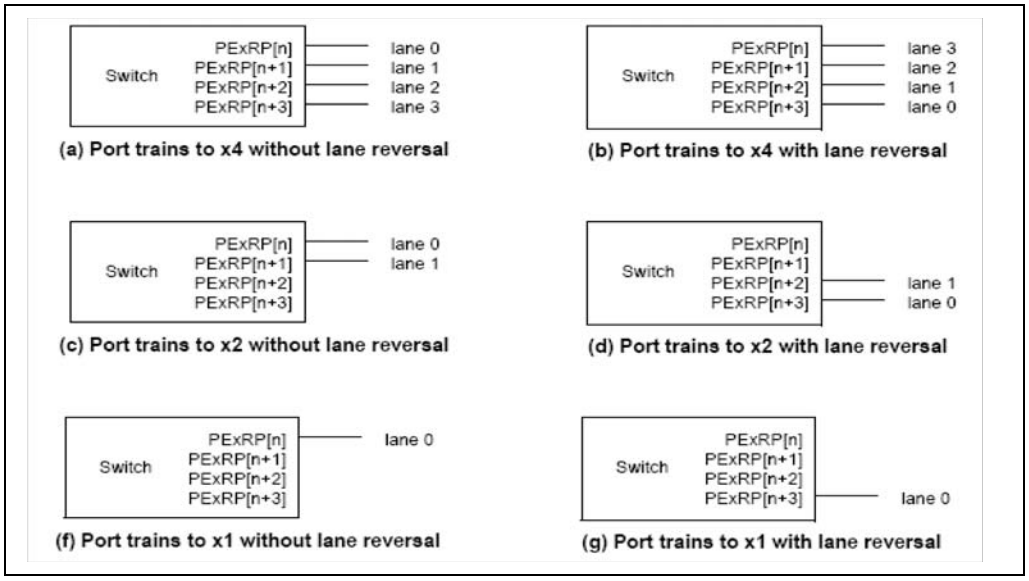


Figure 3 Lane Reversal for Highest Achievable Link Width of x4

Notes

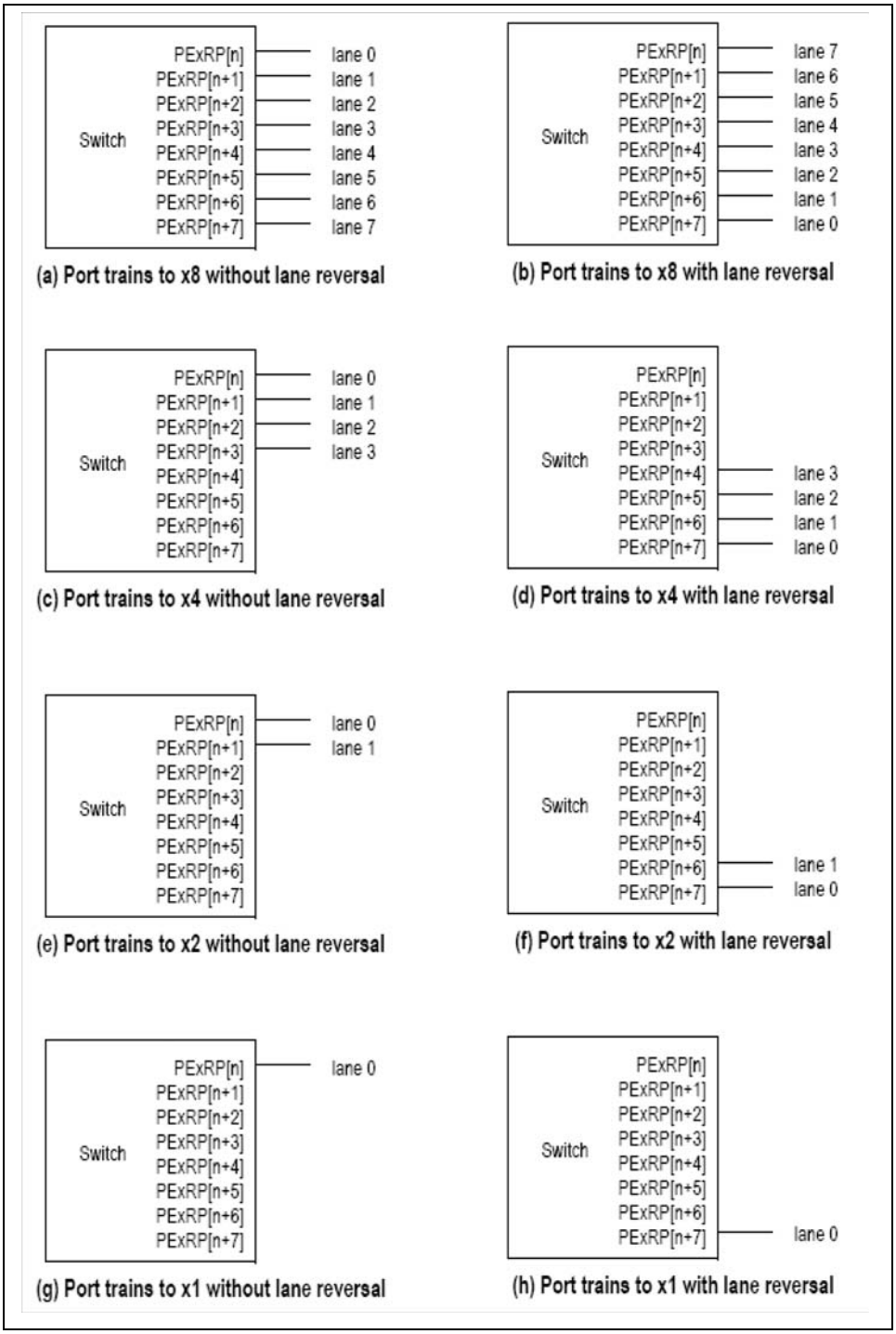


Figure 4 Lane Reversal for Highest Achievable Link Width of x8

Polarity Inversion

Each port of the PES32NT24AG2 supports automatic polarity inversion defined by the PCIe specification. This allows trace routing flexibility to avoid crossovers and potentially reduces the number of trace vias required for signal routing. Polarity inversion is a function of the receiver and not the transmitter. The transmitter never inverts its data. During link training, the receiver examines symbols 6 through 15 of the TS1 and TS2 ordered sets for inversion of the PExRP[n] and PExRN[n] signals. If an inversion is detected, then

Notes

logic for the receiving lane automatically inverts received data. Polarity inversion is a lane function and not a link function. Therefore, it is possible for some lanes of link to be inverted and for others not to be inverted.

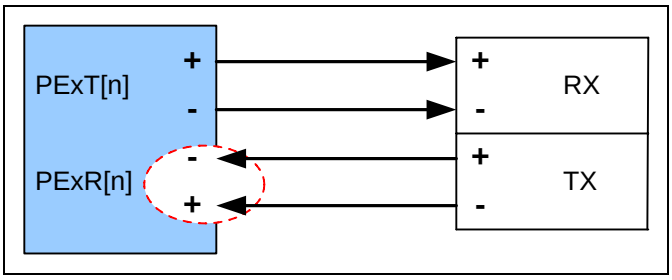


Figure 5 Polarity Inversion

Tx Capacitors

The PCIe specification requires that each lane of the link is AC coupled between its corresponding transmitter and receiver. The AC coupling capacitors for the PES32NT24AG2 are discrete components located along each transmitter link on the PCB. These AC coupling capacitors allow the transmitter and receiver on a link to be biased at separate voltages. Table 1 summarizes the guidelines for implementing the PCIe AC coupling capacitors on a system board.

Parameter	Implementation Guideline
AC Coupling	AC coupling capacitors are required on the Tx pairs originating from the PES32NT24AG2
Capacitor Value	Between 75nF and 200nF
Capacitor Tolerance	Specified minimum/maximum range must be met when capacitor tolerance is considered along with effects due to temperature and voltage
Capacitor Type	Size 603 ceramic capacitors are acceptable, however, size 402 capacitors are strongly encouraged. The smaller the package size, the less ESL is introduced into the topology. The same package and capacitor size should be used for each signal in a differential pair. Do not use capacitor packs for PCIe AC coupling.
Capacitor Pad Size	To minimize parasitic impacts, pad sizes for each capacitor should be the minimum allowed per PCB manufacturer.
Capacitor Placement	AC coupling capacitors should be located at the same place within the differential pair. They should not be staggered in distance from one trace of the differential pair to the other. Capacitors should be placed as close to each other as possible to avoid creating large uncoupled sections within the differential pair traces. Relative location from one differential pair to another is not important.
Capacitor Location – Chip-to-Connector Routing	Capacitors should be placed such that they are not located in the center point of a trace route (for example, capacitors should be placed next to the connector or 1/3 the distance between the connector and the PES32NT24AG2.
Capacitor Location – Chip-to-Chip Routing	Capacitors should be located off-center within the interconnect (for example, placing the capacitors next to the Rx pins of one device is generally better than locating the capacitors in the midpoint of the interconnect).

Table 6 PCIE Interface AC Coupling Capacitor Guidelines

Notes

Routing Differential Pairs

The switch includes 50 Ohm resistor on-die terminations on both the transmit and the receive pins. No external termination is required. Individual traces within a given differential pair (positive and negative) must be matched in length to a tolerance of 5 mils. Length matching within a differential pair should occur on a segment-by-segment basis, as opposed to length matching across the total distance of the overall route. In addition, the spacing between traces of adjacent pairs must be at least 20 mils edge-to-edge to reduce crosstalk effects.

Note that trace length matching between pairs is not required because the PCIe 2.0 specifications allow up to 8ns of skew between differential pairs. AC coupling capacitors are associated with the Tx differential pairs and should be located symmetrically on the top or bottom layer between the switch and the PCIe connectors/devices.

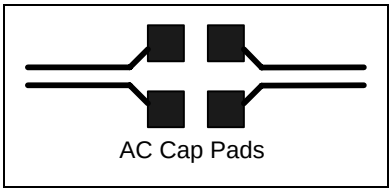


Figure 6 AC Capacitor Placement

Every effort should be made to avoid vias on the PCIe differential pairs since they can result in a signal loss of up to 0.25 dB. When a via is unavoidable, its pad size should be less than 25 mils, its hole size should be less than 14 mils, and its anti-pads should be 35 mils or smaller. No extra vias should be added over and above those needed for IC pads or a connector. Vias in a differential pair should always be at the same relative location and placed in a symmetrical fashion along the differential pair as shown in Figure 7.

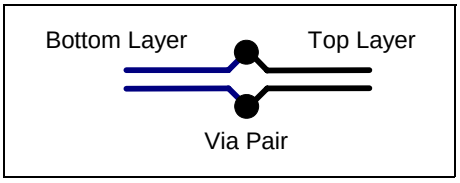


Figure 7 Via Placement

Avoid 90-degree bends or turns on traces. Wherever possible, the number of left and right bends should be matched as closely as possible. Alternating left and right turns helps to minimize length skew differences between each signal of a differential pair.

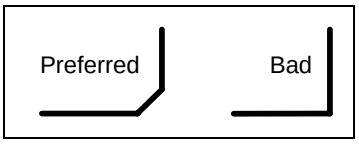


Figure 8 Acceptable Bends and Unacceptable Bends

Depending on the system topology and the maximum targeted trace length, regular FR4 material is appropriate dielectric material. In case of a backplane type of application, higher quality, lower loss material, such as Nelco 4000-13, may be used.

A HSPICE simulation kit for the switch can be requested by emailing ssdhelp@idt.com.

Device Breakout Area

Within the breakout areas of the PES32NT24AG2, the PCIe signal pairs should maximize the differential routing while minimizing any discontinuities or trace length skew. Length matching of the differential traces should occur as close as possible to the pad or pin while avoiding the addition of tight bends within the

Notes

traces. The breakout area should not exceed 250 mil in length for the PCIe interface. Within the breakout areas, the trace routing guidelines of the differential pairs can be slightly relaxed (if absolutely necessary) to facilitate successful breakout of the signals. A width and spacing geometry of 6/4.5/6 (6 mil trace width and 4.5 mil spacing edge to edge) are used on PES32NT24AG2 evaluation board.

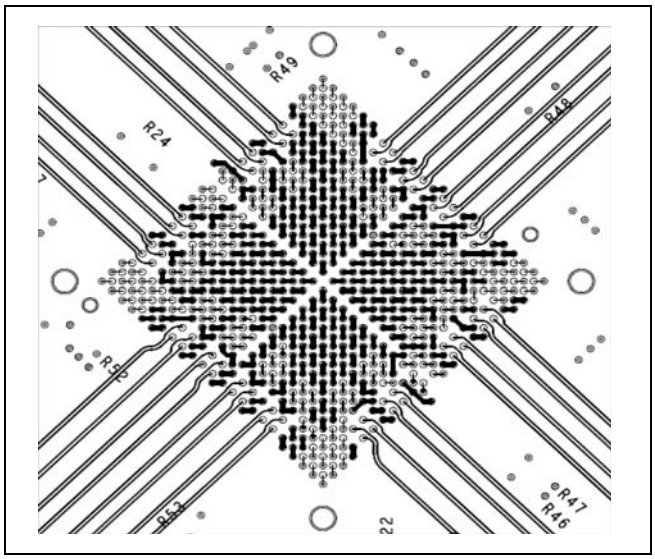


Figure 9 PCIe Differential Trace Breakout Bottom Layer

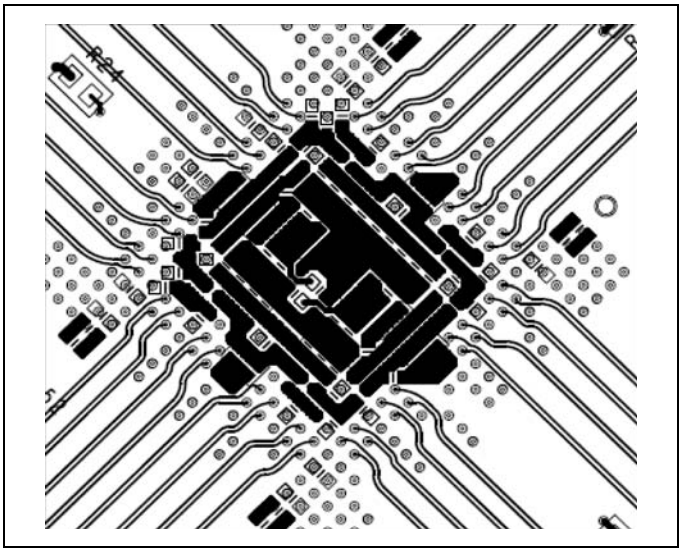


Figure 10 PCI Different Trace Breakout Top Layer

Serdes Reference Resistor Pins

The switch has one Serdes Reference Resistor pin per port. The 3.0K +/- 1% reference bias resistor should be located as close to these pins as possible and should be tapped immediately to the GROUND plane. This resistor must be isolated from any source of noise injection. One way to achieve this is to place the resistor on the back-side of the board, directly underneath the device. No bypass capacitors must be placed on these pins.

Notes

Reference Clock (REFCLK) Circuitry

The switch has two differential global reference clock inputs (GCLKP[1:0]/GCLKN[1:0]) that are used to generate all of the clocks required by the internal switch logic and the SerDes. The differential clock inputs require the signal source to drive a 0V common-mode and the REFCLK signal must meet the electrical specifications defined in the PCI Express Card Electromechanical Specification. AC coupling is not required on reference input clocks.

The reference clock inputs support spread spectrum clocking (SSC) for reducing EMI. The required method is to adjust the spread technique to prevent modulation above the nominal frequency. This technique is often called “down-spreading.” If SSC is used, all clocks must come from a single source. This includes the clock for the switch itself, the clock for the devices connected to the downstream ports of the switch, and the clocks for the root complex chipset or other devices (switch or bridge) connected to the upstream port of the switch. If SSC is not used, multiple clock sources are allowed for each PCI Express device in the tree.

Global Reference Clock Selection

The frequency of the global reference clock inputs can be either 100 MHz or 125 Mhz, and the Global Clock Frequency Select (GCLKFSEL) input is used to indicate the choice of frequency. The PCIe CEM specification requires a nominal frequency of 100 MHz for the reference clock pair. Thus, in the majority of applications, the 100 MHz clock input should be selected.

- For the 100 MHz clock input, GCLKSEL input pin must be asserted low.
- For the 125 MHz clock input, GCLKSEL input pin must be tied to a pull-up resistor of 3.3V power.

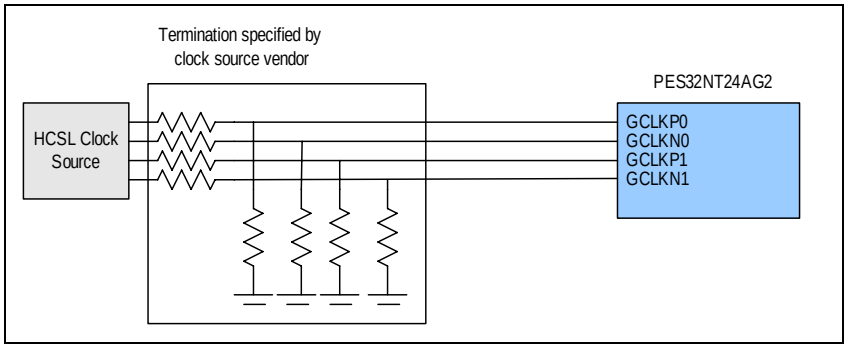


Figure 11 Simplified GCLK Circuit

The switch provides two clock operation modes for each side of the switch: Global Clock and Local Port Clock. System designers must configure the CLKMODE[2:0] pins depending on which mode is chosen for the upstream side and the downstream side of the switch.

Global Port Clocked Mode

Figures 12 and 13 show the implementation of common clock mode and non-common clock mode, respectively. The Spread Spectrum Clock must be disabled when the non-common clock is used on either the upstream port or downstream port.

> Note: The downstream port's reference clock must be controlled by the power good signal or an appropriate control signal if downstream slots support hot-plug operation.

Notes

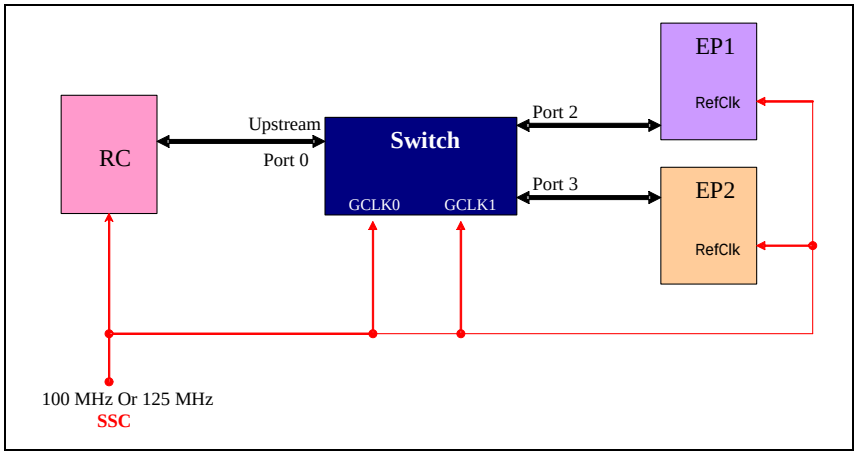


Figure 12 Example of Common Clock Mode with SSC

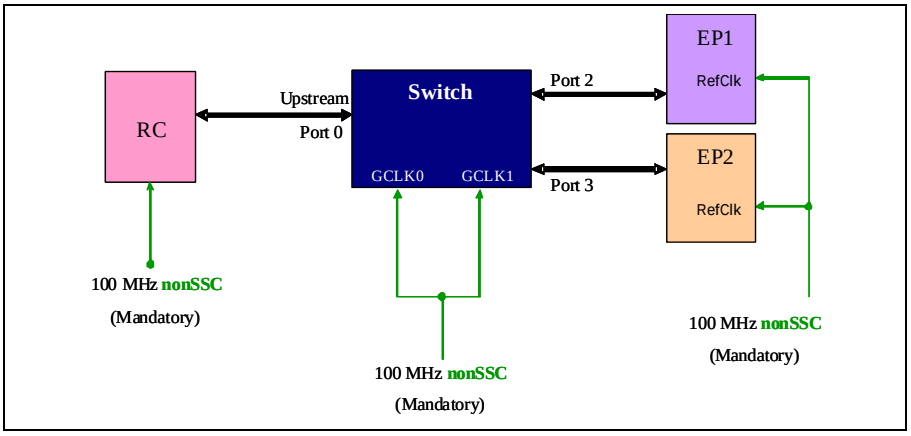


Figure 13 Example of Non-Common Clock Mode

Local Port Clocked Mode

Associated with some ports is a port reference clock input (PxCLK). Depending on the port clocking mode, a differential reference clock is driven into the device on the corresponding PxCLKP and PxCLKn pins. Table 7 lists the ports that must operate in the same port clock mode (i.e., Global Clocked or Local Port Clocked). This allows support for Spread Spectrum Clocking (SSC) either globally or independently on a per port basis. Figure 14 shows implementation of Local Port Clock mode with SSC on root complex via P0CLK.

Ports
0,1
2,3
4,5
6,7
8,9,10,11

Table 7 Ports that must operate with the same Port Clock Mode

Notes

Ports
12,13,14,15
16,17,18,19
20,21,22,23

Table 7 Ports that must operate with the same Port Clock Mode

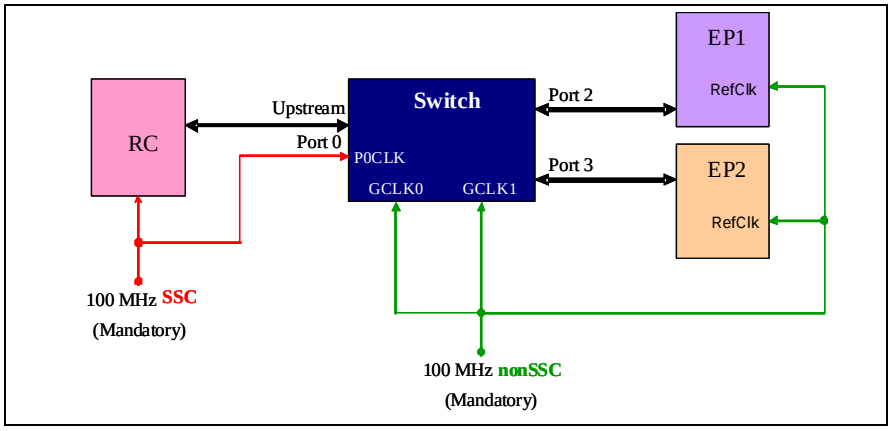


Figure 14 Example of Local Port Clock mode with SSC on Root Complex

Reset (Fundamental Reset) Schemes

The PERSTN pin is used to reset all logic inside the switch and is a Schmitt Trigger Input which can be connected to the PERST# from the system or a power-on reset circuit. In a system, the values of Tpvperl and Tperst-clk depend on the mechanical form factor in which the switch is used. For example, the PCIe Card Electromechanical Specification, Revision 2.0, specifies the minimum value of Tperst-clk=100µs and Tpvperl=100ms.

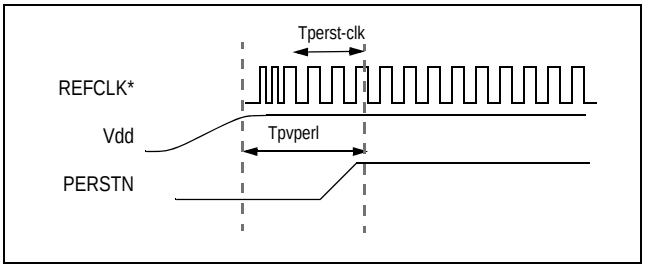


Figure 15 Fundamental Reset

For the reset signals to downstream ports, reset schemes listed below can be implemented.

- Simplified Reset Scheme
- Reset Scheme for Hot Plug Support

Simplified Reset Scheme

If Hot Plug support is not required, the simplified reset scheme can be implemented as shown in Figure 16. Add a buffer on the PERST# signal if the output from system is not able to drive a number of fan-out loads for all downstream ports.

Notes

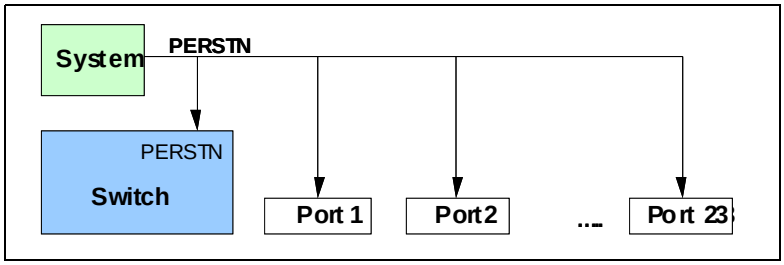


Figure 16 Simplified Reset Scheme

Reset Scheme for Hot Plug Support

Figure 17 shows an implementation where downstream endpoints have independent fundamental reset. This scheme should be used if Hot-Plug support is needed selectively on the downstream ports.

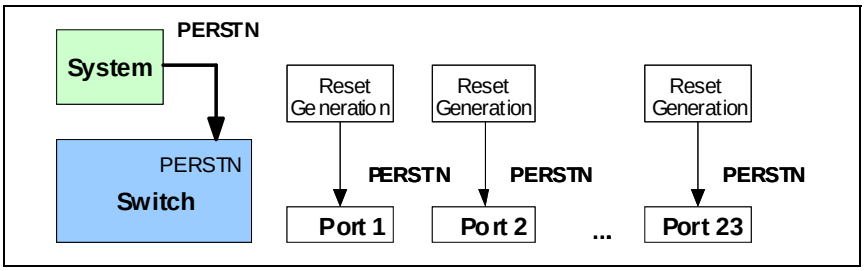


Figure 17 Reset Scheme for Hot Plug Support

RSTHALT

When this signal is asserted high during a PCI Express fundamental reset, the switch continuously returns Configuration Request Retry Completion Status (CRS) to Configuration Requests during the enumeration process. This allows the system BIOS via the SMBus to access internal registers before normal device operation begins. The device exits the RSTHALT state when the RSTHALT bit is cleared in the SWCTL register by a SMBus master. This RSTHALT mode is not required in most applications. The RSTHALT pin should be pulled down externally if the application does not use a SMBus master to initialize internal registers.

SMBus Interfaces

The switch provides two SMBus interfaces.

- The Master SMBus interface provides connection for an external serial EEPROM used for initialization and optional external I/O expanders.
- The slave SMBus interface provides full access to all software visible registers in the PES32NT24AG2, allowing every register in the device to be read or written by an external SMBus master. The slave SMBus may also be used to preload the serial EEPROM used for initialization.

The Master SMBus interface consists of an SMBus clock pin and data pin. The Slave SMBus interface consists of an SMBus clock pin, data pin and 2 SMBus address pins. In the slave interface, these address pins allow the SMBus address to which the device responds to be configured. In the master interface, the address of the serial configuration EEPROM from which data is loaded to be configured is fixed to 0x50. This interface operates using a basic I^2C protocol, at a rate of 400 KHz (i.e., I^2C Fast Mode). The SMBus address is set up on negation of PERSTN by sampling the corresponding address pins.

Notes

Initialization from Serial EEPROM

During a fundamental reset, an serial EEPROM is required to initialize any software visible register within the device. Serial EEPROM loading occurs if the Switch Mode (SWMODE [3:0]) field selects an operating mode that performs serial EEPROM initialization (e.g., Normal switch mode with Serial EEPROM initialization).

Any serial EEPROM compatible with those listed in Table 8 can be used to store switch initialization values. EEPROM space may not be fully utilized because some of these devices are larger than the total available PCI configuration space that can be initialized in the switch.

Serial EEPROM	Size
24C32	4 KB
24C64	8 KB
24C128	16 KB
24C256	32 KB
24C512	64 KB

Table 8 PES32NT24AG2 Compatible Serial EEPROMs

Configuring the I/O Expander Address

The switch utilizes external SMBus / I^2C-bus I/O expanders connected to the master SMBus interface for hot-plug and port status signals. The switch is designed to work with Phillips PCA9555 compatible I/O expanders (i.e., PCA9555, PCA9535, and PCA9539). See the Phillips PCA9555 data sheet for details on the operation of this device. For applications that require more than 8 I/O expanders, the MAX7311 is recommended since it is compatible with the Phillips PCA9555 and supports 64 slave addresses.

The switch supports up to 22 external I/O expanders numbered 0 to 22. Table 9 summarizes the allocation of functions to I/O expanders. Figure 18 illustrates an example of the interface for I/O expanders 0, 2, and 12. During switch initialization, the SMBus/I^2C-bus address allocated to each I/O expander used in that system configuration should be written to the corresponding I/O Expander Address (IOE[0,2,12]ADDR) field. The IOE[0,2]ADDR fields are contained in the I/O Expander Address 0 (IOEXPADDR0) register while the IOE[12]ADDR fields are contained in the SMBus I/O Expander Address 3 (IOEXPADDR3) register.

Hot-plug outputs and I/O expanders may be initialized via serial EEPROM. Since the I/O expanders and serial EEPROM both utilize the master SMBus, no I/O expander transactions are initiated until serial EEPROM initialization completes.

SMBus I/O Expander	Section	Functionality
0	Lower	Port 0 hot-plug
	Upper	Port 4 hot-plug
1	Lower	Port 8 hot-plug
	Upper	Port 16 hot-plug
2	Lower	Port 2 hot-plug
	Upper	Port 6 hot-plug

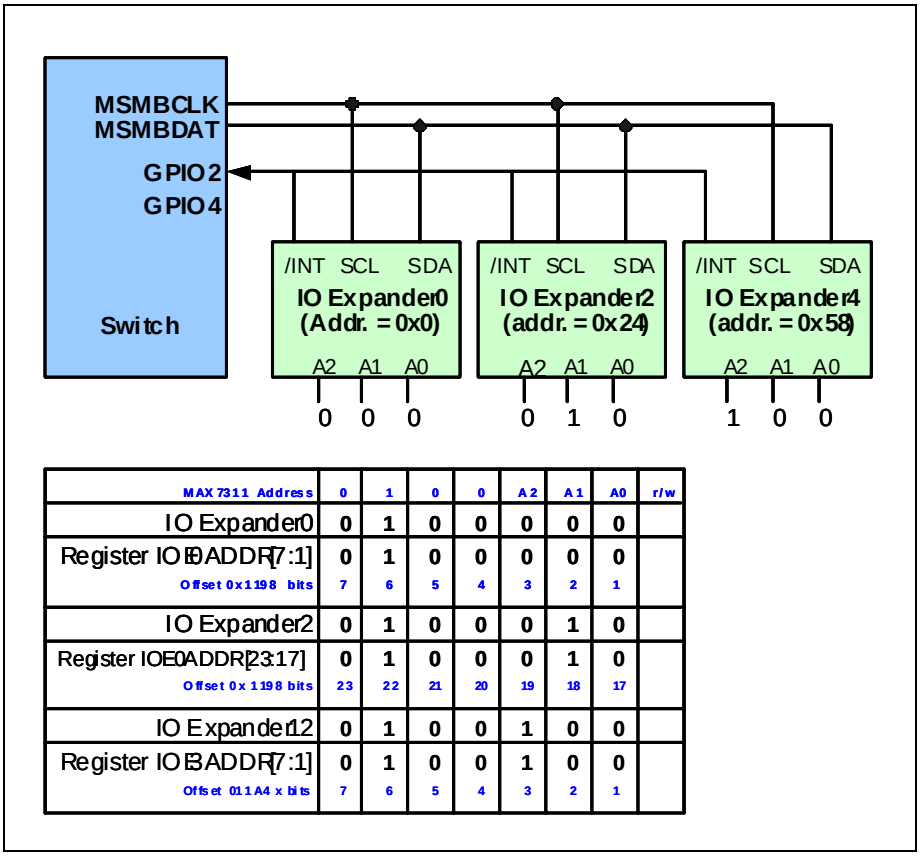
Table 9 I/O Expander Functionality Allocation (Page 1 of 2)

Notes

SMBus I/O Expander	Section	Functionality
3	Lower	Port 12 hot-plug
	Upper	Port 20 hot-plug
4	Lower	Port 1 hot-plug
	Upper	Port 3 hot-plug
5	Lower	Port 5 hot-plug
	Upper	Port 7 hot-plug
6	Lower	Port 10 hot-plug
	Upper	Port 14 hot-plug
7	Lower	Port 18 hot-plug
	Upper	Port 22 hot-plug
8	Lower	Port 9 hot-plug
	Upper	Port 11 hot-plug
9	Lower	Port 13 hot-plug
	Upper	Port 15 hot-plug
10	Lower	Port 17 hot-plug
	Upper	Port 19 hot-plug
11	Lower	Port 21 hot-plug
	Upper	Port 23 hot-plug
12	Lower / Upper	Hot-plug MRL inputs (port0 through 7, 8, 10, 12, 14, 16, 18, 20, 22)
13	Lower	Hot-plug MRL inputs (port9, 11, 13, 15, 17, 19, 21, 23)
14	Lower / Upper	Hot-plug electromechanical interlock (Ports 0, 2, 4, 6, 8, 12, 16, 20)
15	Lower / Upper	Hot-plug electromechanical interlock (Ports 1, 3, 5, 7, 10, 14, 18, 22)
16	Lower / Upper	Hot-plug electromechanical interlock (Ports 9, 11, 13, 15, 17, 19, 21, 23)
17	Lower / Upper	Link status (Ports 0 through 15)
18	Lower / Upper	Link activity (Ports 0 through 15)
19	Lower / Upper	Link status (Ports 16 through 23) / Link activity (Ports 16 through 23)
20	Lower / Upper	Port reset outputs (Ports 0, 2, 4, 6, 8, 12, 16, 20) / Partition fundamental reset inputs (partitions 0 to 7)
21	Lower / Upper	Port reset outputs (Ports 1, 3, 5, 7, 9, 10, 11, 13, 14, 15, 17, 18, 19, 21, 22, 23)

Table 9 I/O Expander Functionality Allocation (Page 2 of 2)

Notes



MAX7311 Address	0	1	0	0	A2	A1	A0	r/w
IO Expander0	0	1	0	0	0	0	0	
Register IOE0ADDR[7:1]	0	1	0	0	0	0	0	
Offset 0x1198 bits	7	6	5	4	3	2	1	
IO Expander2	0	1	0	0	0	1	0	
Register IOE0ADDR[23:17]	0	1	0	0	0	1	0	
Offset 0x1198 bits	23	22	21	20	19	18	17	
IO Expander12	0	1	0	0	1	0	0	
Register IOE3ADDR[7:1]	0	1	0	0	1	0	0	
Offset 011A4x bits	7	6	5	4	3	2	1	

Figure 18 Example of I/O Expander Interface

Slave SMBus Address Interface

The slave SMBus interface provides the switch with a configuration, management, and debug interface. Using the slave SMBus interface, an external master can read or write any software visible register in the device. The address is specified by the SSMBADDR[2:1] signals as shown in Table 10.

Address Bit	Address Bit Value
1	SSMBADDR[1]
2	SSMBADDR[2]
3	1
4	0
5	1
6	1
7	1

Table 10 Slave SMBus Address

Notes

Power and Decoupling Scheme

The switch has five different types of power supply pins:

1. V_{DD}CORE (1.0V) powers the digital core of the switch.
2. V_{DD}PEA (1.0V) power the SERDES core and analog circuits. V_{DD}PEA should have no more than 25mVpeak-peak AC power supply noise superimposed on the 1.0V nominal DC value.
3. V_{DD}PEHA (2.5V) power the SERDES core and analog circuits. V_{DD}PEHA should have no more than 50mVpeak-peak AC power supply noise superimposed on the 2.5V nominal DC value.
4. V_{DD}PETA (1.0V) is the termination voltage used on the SERDES TX lanes. V_{DD}PETA can be adjusted to modify the TX common mode voltage as well as the voltage swing.
5. V_{DD}I/O (3.3V) powers the low speed IOs of the switch.

V_{DD}CORE, V_{DD}PEA, and V_{DD}PETA can be derived from the same voltage source with appropriate bypass capacitors and a ferrite bead. If all voltages can not be handled by a single voltage regulator, they can be derived from separate voltage regulators.

A ferrite bead can be used to attenuate the power noise and improve the analog circuit performance in a noisy environment. The following three parameters should be considered when you select a ferrite bead for power rails:

- Very low DC resistance
- Impedance of 50 ~ 120 ohms at 100 MHz
- Provides enough DC current

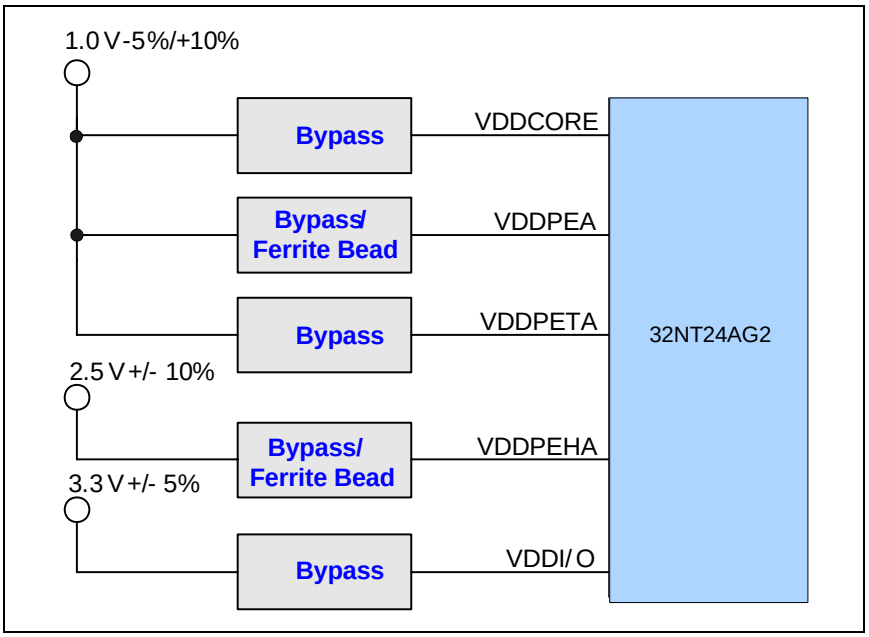


Figure 19 Board Power Supply Arrangement

Power Consumption

The typical and maximum power consumption can be found in the appropriate switch data sheet (see Reference Documents at the end of this guide).

Power-Up/Power-Down Sequence

During power supply ramp-up, V_{DD}I/O must remain at least 1.0V above V_{DD}CORE at all times. If V_{DD}I/O is brought up first, then there is no problem. There are no other power-up sequence requirements for the various operating supply voltages. The power-down sequence can occur in any order.

Notes

Decoupling Scheme

1) One bypass capacitor per power pin is recommended if board layout allows. 0402 package ceramic capacitors are recommended for 0.1µF and 0.01µF capacitors.

2) Bypass Capacitors must be placed as close to the device pins as possible based on space availability. Note that some of the vias need to be shared in order to create space for placing a capacitor next to a pin.

3) A bigger capacitor should be used to filter out low frequency noise. Larger 1µF and 47µF capacitors should be added around the part. Two bigger capacitors per voltage supply are appropriate. One option is to spread out the big capacitors at four corners, top and bottom layers of the chips.

4) Short and wide traces should be used to minimize resistance and inductance.

5) Prioritize the bypass capacitors in the following order for each power supply:

1. V_{DD}CORE
2. V_{DD}PEA / V_{DD}PETA
3. V_{DD}PEHA
4. V_{DD}I/O

GPIO and JTAG Pins

GPIO Pins

The switch has a number of General Purpose I/O (GPIO) pins that may be individually configured as general purpose inputs, general purpose outputs, or alternate functions. GPIO pins are controlled by the General Purpose I/O Function (GPIOFUNC), General Purpose I/O Configuration (GPIOCFG), and General Purpose I/O Data (GPIOD) registers in the upstream port's PCI configuration space. Please refer to the device data sheet for additional details.

The internal pull-up resistors value for the GPIO pins under typical condition is about 92K ohm.

JTAG Pins

The switch provides the JTAG Boundary Scan interface to test the interconnections between integrated circuit pins after they have been assembled onto a circuit board. For details of the interface, please refer to to the appropriate switch user manual (see Reference Documents below).

The JTAG_TRST_N pin must be asserted low when the switch is in normal operation mode (i.e. drive this signal low with an external pull-down or control logic on the board if the JTAG interface is not used).

Switch Partitioning

Switch partitioning allows the logical division of the PCIe switch into multiple partitions (up to 8), each of which is composed of a configurable number of ports, and each of which connects to a separate PCIe domain[1]. Each switch partition is logically isolated from the other partitions.

From the switch's perspective, a switch partition represents a logical container that contains switch ports associated with a PCIe domain. Any switch port can be configured to belong to one partition. The assignment of ports to partitions is left to the system designer. A partition may be configured with zero, one, or many ports, although certain rules apply (see below) regarding valid partition configurations. Figure 20 shows a PES32NT24AG2 configured with 3 partitions.

1. A PCIe domain is the collection of PCIe devices under a common processor/memory complex (i.e., root-complex), and sharing common PCIe memory, I/O, and configuration spaces.

Notes

A partition can be placed in one of three modes: disabled, active, and reset. When a partition is disabled, all ports associated with the partition (if any) are disabled and can't be used. When a partition is active, all ports associated with the partition are active. When a partition is reset, all ports associated with the partition are in PCIe fundamental reset.

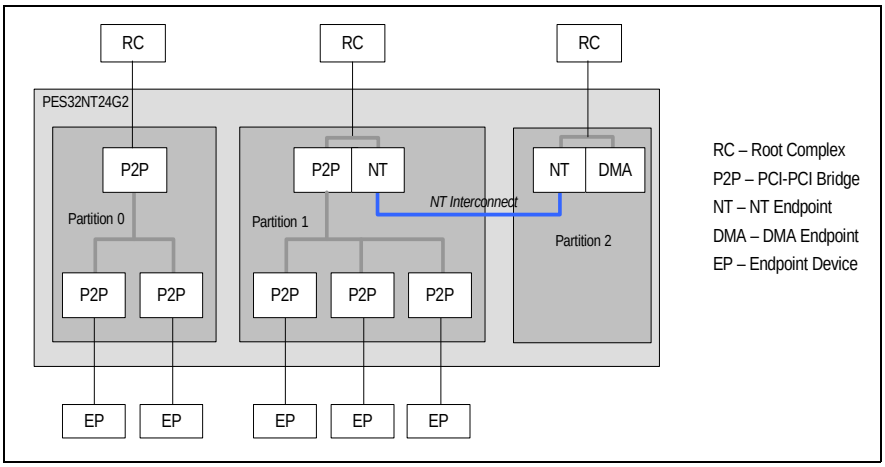


Figure 20 The PES32NT24AG2 Configured with 3 Switch Partitions

As shown in Figure 20, each port can be placed in one of several modes. Depending on the mode, ports may contain between one and three PCIe functions. The PCI-to-PCI bridge function allows the port to forward packets across the same partition. By placing two or more ports with a PCI-to-PCI bridge function within a partition, a PCI Express switch is logically created within that partition. On the other hand, the NT endpoint function serves as a gateway to other PCIe domains via the non-transparent interconnect (shown in blue in Figure 20). Finally, the DMA endpoint function provides a DMA engine to off-load the partition's host during data transfers. Table 11 lists the port operating modes and the PCIe functions in each mode.

Port Mode	PCIe Functions in the Port		
	Function 0	Function 1	Function 2
Disabled	The port is not operational		
Upstream switch port	PCI-to-PCI bridge		
Upstream switch port with NT function	PCI-to-PCI bridge	NT Endpoint	
Upstream switch port with NT and DMA functions	PCI-to-PCI bridge	NT Endpoint	DMA Endpoint
NT function	NT Endpoint		
NT with DMA function	NT Endpoint		DMA Endpoint
Downstream switch port	PCI-to-PCI bridge		
Unattached	The port is not associated with any partition. The port accepts configuration requests that allow for switch management.		

Table 11 Port Operating Modes[1]

1. Not all switch ports support all modes. Refer to the PES32NT24AG2 User Manual for further details.

Not all possible partition configurations are valid. For example, it is not valid to configure a partition with more than one upstream port. The following are the common partition configurations supported by the PES32NT24AG2:

- A switch partition with one upstream switch port (with or without NT or DMA functions) and one or more downstream switch ports.
- A switch partition with an NT endpoint port (with or without DMA function).

Notes

Partitions and ports can be configured at boot-time, and reconfigured dynamically during run-time by software or automatically by hardware as a result of a failover event. When a port's mode is re-configured, the change can be stateless (i.e., the port is reset during the change) or state-full (i.e., the port preserves its configuration during the change).

Reference Documents

89H32NT24AG2 Data Sheet and Device User Manual

PCI Express Base Specification, Revision 1.1 & 2.0, PCI-SIG

PCI Express Card Electromechanical Specification Revision 1.1 & 2.0, PCI-SIG

PCI to PCI Bridge Architecture Specification, Revision 1.2, PCI-SIG

SMBus Specification, Revision 2.0

Intel PCI Express Electrical Interconnect Design

Revision History

January 20, 2010: Initial publication.

May 5, 2011: In Figure 19, for VDDPEHA, changed to Bypass/Ferrite Bead.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.