# TSB12LV22
# OHCI-Lynx PCI-BASED IEEE 1394 HOST CONTROLLER

SLLS290 – JULY 1998

- **Designed to 1394 Open Host Controller Interface (OHCI) Specification**
- **IEEE 1394-1995 Compliant and Compatible with Proposal 1394A**
- **Compliant to Latest PCI Specification, PCI 2.2**
- **PCI Power Management Compliant**
- **3.3-V Core Logic with Universal PCI Interface Compatible with 3.3-V and 5-V PCI Signaling Environments**
- **Supports Serial Bus Data Rates of 100, 200, and 400 Mbits/s**
- **Provides Bus-Hold Buffers on Physical I/F for Low-Cost Single Capacitor Isolation**
- **Supports Physical Write Posting of Up to Three Outstanding Transactions**
- **Serial ROM Interface Supports 2-Wire Devices**
- **Supports External Cycle Timer Control for Customized Synchronization**
- **Implements PCI Burst Transfers and Deep FIFOs to Tolerate Large Host Latency**
- **Provides up to Four General Purpose I/Os**
- **Fabricated in Advanced Low-Power CMOS Process**
- **Packaged in 100 LQFP (PZP)**

## description

The Texas Instruments OHCI-Lynx™ is a PCI-to-1394 host controller compatible with the latest PCI, IEEE1394, and 1394 OHCI 1.00 specifications. The chip provides the IEEE1394 link function, and is compatible with serial bus data rates of 100 Mbits/s, 200 Mbits/s, and 400 Mbits/s.

As required by the 1394 OHCI Specification, internal control registers are memory mapped and non-prefetchable. The PCI configuration header is accessed through configuration cycles specified by PCI, and provides Plug-and-Play (PnP) compatibility. Furthermore, the OHCI-Lynx is compliant with the PCI Power Management Specification, per the PC ′98 requirements.

The OHCI-Lynx design provides PCI bus master bursting, and is capable of transferring a cacheline of data at 132 Mbytes/s after connection to the memory controller. Since PCI latency can be large even on a PCI Revision 2.1 system, deep FIFOs are provided to buffer 1394 data.

Physical write posting buffers are provided to enhance serial bus performance, and multiple isochronous channels are provided for simultaneous operation of real-time applications. The OHCI-Lynx also provides bus holding buffers on the phy interface for simple and cost effective single capacitor isolation.

An advanced CMOS process is used to achieve low power consumption while operating at PCI clock rates up to 33 MHz.

## OHCI-Lynx block diagram

A simplified block diagram of the OHCI-Lynx is provided in Figure 1.

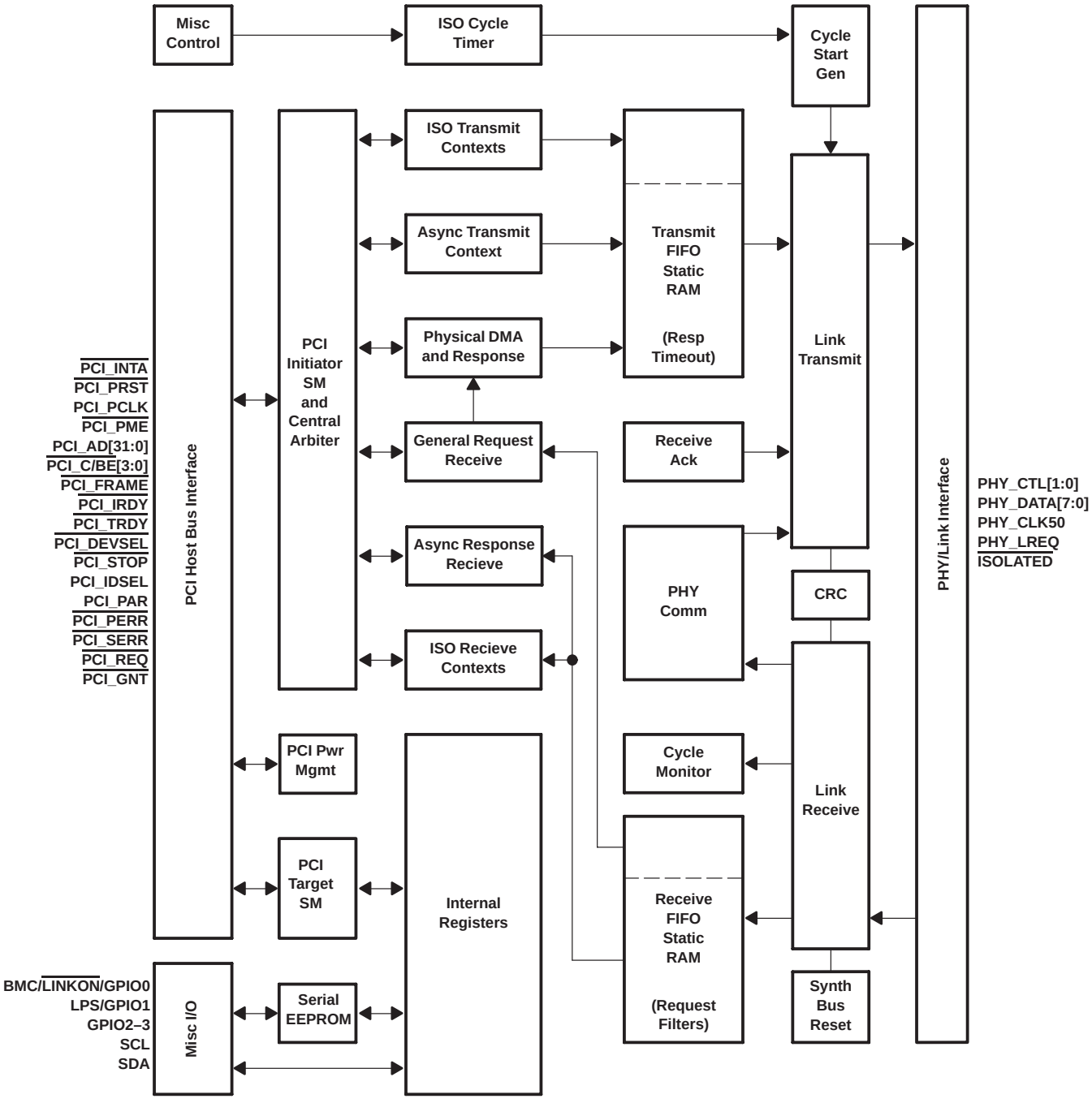



**Figure 1. OHCI-Lynx Block Diagram**

## terminal assignments

This section provides the terminal assignments for the OHCI-Lynx.

**PZP PACKAGE**
**(TOP VIEW)**

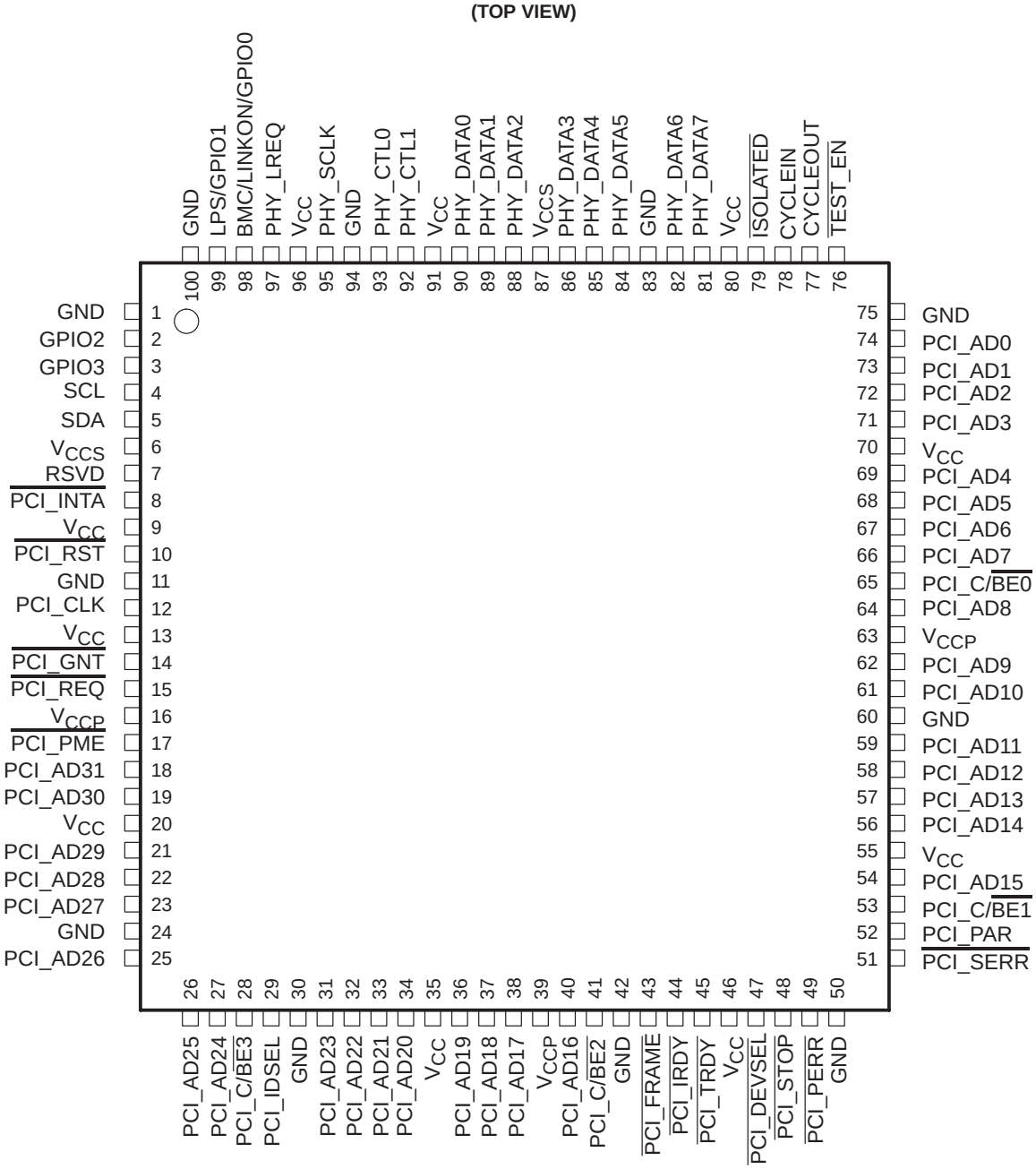



**Figure 2. Terminal Assignments**

**Table 1. Signals Sorted by Pin Number**

| NO | TERMINAL NAME | NO | TERMINAL NAME | NO | TERMINAL NAME | NO | TERMINAL NAME |
|---|---|---|---|---|---|---|---|
| 1 | GND | 26 | PCI_AD25 | 51 | $\overline{\text{PCI\_SERR}}$ | 76 | $\overline{\text{TEST\_EN}}$ |
| 2 | GPIO2 | 27 | PCI_AD24 | 52 | PCI_PAR | 77 | CYCLEOUT |
| 3 | GPIO3 | 28 | PCI_C/$\overline{\text{BE3}}$ | 53 | PCI_C/$\overline{\text{BE1}}$ | 78 | CYCLEIN |
| 4 | SCL | 29 | PCI_IDSEL | 54 | PCI_AD15 | 79 | $\overline{\text{ISOLATED}}$ |
| 5 | SDA | 30 | GND | 55 | $V_{CC}$ | 80 | $V_{CC}$ |
| 6 | $V_{CCS}$ | 31 | PCI_AD23 | 56 | PCI_AD14 | 81 | PHY_DATA7 |
| 7 | RSVD | 32 | PCI_AD22 | 57 | PCI_AD13 | 82 | PHY_DATA6 |
| 8 | $\overline{\text{PCI\_INTA}}$ | 33 | PCI_AD21 | 58 | PCI_AD12 | 83 | GND |
| 9 | $V_{CC}$ | 34 | PCI_AD20 | 59 | PCI_AD11 | 84 | PHY_DATA5 |
| 10 | $\overline{\text{PCI\_RST}}$ | 35 | $V_{CC}$ | 60 | GND | 85 | PHY_DATA4 |
| 11 | GND | 36 | PCI_AD19 | 61 | PCI_AD10 | 86 | PHY_DATA3 |
| 12 | PCI_CLK | 37 | PCI_AD18 | 62 | PCI_AD9 | 87 | $V_{CCS}$ |
| 13 | $V_{CC}$ | 38 | PCI_AD17 | 63 | $V_{CCP}$ | 88 | PHY_DATA2 |
| 14 | $\overline{\text{PCI\_GNT}}$ | 39 | $V_{CCP}$ | 64 | PCI_AD8 | 89 | PHY_DATA1 |
| 15 | $\overline{\text{PCI\_REQ}}$ | 40 | PCI_AD16 | 65 | PCI_C/$\overline{\text{BE0}}$ | 90 | PHY_DATA0 |
| 16 | $V_{CCP}$ | 41 | PCI_C/$\overline{\text{BE2}}$ | 66 | PCI_AD7 | 91 | $V_{CC}$ |
| 17 | $\overline{\text{PCI\_PME}}$ | 42 | GND | 67 | PCI_AD6 | 92 | PHY_CTL1 |
| 18 | PCI_AD31 | 43 | $\overline{\text{PCI\_FRAME}}$ | 68 | PCI_AD5 | 93 | PHY_CTL0 |
| 19 | PCI_AD30 | 44 | $\overline{\text{PCI\_IRDY}}$ | 69 | PCI_AD4 | 94 | GND |
| 20 | $V_{CC}$ | 45 | $\overline{\text{PCI\_TRDY}}$ | 70 | $V_{CC}$ | 95 | PHY_SCLK |
| 21 | PCI_AD29 | 46 | $V_{CC}$ | 71 | PCI_AD3 | 96 | $V_{CC}$ |
| 22 | PCI_AD28 | 47 | $\overline{\text{PCI\_DEVSEL}}$ | 72 | PCI_AD2 | 97 | PHY_LREQ |
| 23 | PCI_AD27 | 48 | $\overline{\text{PCI\_STOP}}$ | 73 | PCI_AD1 | 98 | BMC/LINKON/GPIO0 |
| 24 | GND | 49 | $\overline{\text{PCI\_PERR}}$ | 74 | PCI_AD0 | 99 | LPS/GPIO1 |
| 25 | PCI_AD26 | 50 | GND | 75 | GND | 100 | GND |

This section describes the OHCI-Lynx terminal functions. The terminals are grouped in tables by functionality for convenient reference.

## Terminal Functions

| TERMINAL NAME | TERMINAL NO. | I/O | DESCRIPTION |
|---|---|---|---|
| **Power Supply** | | | |
| GND | 1,11,24,30,42,50, 60,75,83,94,100 | I | Device ground terminals |
| $V_{CC}$ | 9,13,20,35,46,55, 70,80,91,96 | I | 3.3-V power supply terminals |
| $V_{CCS}$ | 6,87 | I | Clamp rail power input.; Provides 5 V tolerance for non PCI I/Os |
| $V_{CCP}$ | 16,39,63 | I | PCI signaling clamp rail power input. PCI signals clampled per PCI specification |
| **PCI System** | | | |
| PCI_CLK | 12 | I/O | PCI Bus Clock. Provides timing for all transactions on the PCI bus. All PCI signals are sampled at rising edge of PCLK. |
| PCI_RST | 10 | I/O | PCI Reset. When the PCI bus reset is asserted the OHIC1-Lynx 3-states all output buffers and resets internal registers. When asserted, the device is completely nonfunctional. After deasserting RST, the OHCI-Lynx is in its default state. |
| PCI_INTA | 8 | I/O | PCI Interrupt A. The OHCI-Lynx drives this shared interrupt signal low when there is a pending internal interrupt event that has occurred. |
| **PCI Address and Data** | | | |
| PCI_AD31 – PCI_AD0 | 18,19,21–23, 25–27,31–34, 36–38,40,54, 56–59,61,62,64, 66–69,71–74 | I/O | PCI Address/Data Bus. These signals make up the multiplexed PCI address and data bus on the PCI interface during the address phase of a PCI cycle, AD31:0 contains a 32-bit address or other destination information. During the data phase AD31:0 contains data. |
| PCI_C/BE0<br>PCI_C/BE1<br>PCI_C/BE2<br>PCI_C/BE3 | 65,<br>53,<br>41,<br>28 | I/O | PCI Bus Commands and Byte Enables. The command and byte enable signals are multiplexed on the same PCI terminals. During the address phase of a bus cycle C/BE3:0 defines the bus command. During the data phase, this four-bit bus is used as byte enables. |
| PCI_PAR | 52 | I/O | PCI Parity. In all PCI bus read and write cycles, the OHCI-Lynx calculates even parity across the AD and C/BE buses. As an initiator during PCI cycles, the OHCI-Lynx outputs this parity indicator with a one PCLK delay. As a target during PCI cycles, the calculated parity is compared to the initiator' parity indicator; a miscompare can result in a parity error assertion (PERR). |
| **PCI Interface Control** | | | |
| PCI_DEVSEL | 47 | I/O | PCI Device Select. The OHCI-Lynx will assert this signal to claim a PCI cycle as the target device. As a PCI initiator, the OHCI-Lynx monitors this signal until a target responds. If no target responds before time-out occurs, then the OHCI-Lynx will terminate the cycle with an initiator abort. |
| PCI_FRAME | 43 | I/O | PCI Cycle Frame. This signal is driven by the initiator of a PCI bus cycle. FRAME is asserted to indicate that a bus transaction is beginning, and data transfers continue until while this signal is asserted. When FRAME is deasserted, the PCI bus transaction is in the final data phase. |
| PCI_GNT | 14 | I | PCI Bus Grant. This signal is driven by the PCI bus arbiter to grant the OHCI-Lynx access to the PCI bus after the current data transaction has completed. This signal may or may not follow a PCI bus request depending upon the PCI bus parking algorithm. |
| PCI_IDSEL | 29 | I | Initialization Device Select. IDSEL selects the OHCI-Lynx during configuration space accesses. IDSEL can be connected to one of the upper 24 PCI address lines on the PCI bus. |
| PCI_IRDY | 44 | I/O | PCI Initiator Ready. IRDY indicates the PCI bus initiator's ability to complete the current data phase of the transaction. A data phase is completed upon a rising edge of PCLK where both IRDY and TRDY are asserted; until which wait states are inserted. |
| PCI_STOP | 48 | I/O | PCI Cycle Stop Signal. This signal is driven by a PCI target to request the initiator to stop the current PCI bus transaction. This signal is used for target disconnects, and is commonly asserted by target devices which do not support burst data transfers. |

## Terminal Functions (Continued)

| TERMINAL NAME | TERMINAL NO. | I/O | DESCRIPTION |
|---|---|---|---|
| **PCI Interface Control (Continued)** | | | |
| PCI_PERR | 49 | I/O | PCI Parity Error Indicator. This signal is driven by a PCI device to indicate that calculated parity does not match PAR when enabled through the command register. |
| PCI_PME | 17 | | |
| PCI_REQ | 15 | O | PCI Bus Request. Asserted by the OHCI-Lynx to request access to the bus as an initiator. The host arbiter will assert the GNT# signal when the OHCI-Lynx has been granted access to the bus. |
| PCI_SERR | 51 | O | PCI System Error. Output pulsed from the OHCI-Lynx when enabled indicating an address parity error has occurred. The OHCI-Lynx needs not be the target of the PCI cycle to assert this signal. |
| PCI_TRDY | 45 | I/O | PCI Target Ready. TRDY indicates the PCI bus target's ability to complete the current data phase of the transaction. A data phase is completed upon a rising edge of PCLK where both IRDY and TRDY are asserted; until which wait states are inserted. |
| **IEEE1394 PHY/Link** | | | |
| PHY_CTL1<br>PHY_CTL0 | 92<br>93 | I/O | Phy-link Interface Control. These bidirectional signals control passage of information between the two devices. The OHCI-Lynx can only drive these terminals after the PHY has granted permission following a link request (LREQ). |
| PHY_DATA7 –<br>PHY_DATA0 | 81,82,<br>84–86,<br>88–90 | I/O | Phy-link Interface Data. These bidirectional signals pass data between the OHCI-Lynx and the PHY device. These terminals are driven by the OHCI-Lynx on transmissions, and driven by the PHY on reception. Only DATA1:0 are valid for 100Mbit speeds, DATA4:0 are valid for 200Mbit speeds, and DATA7:0 are valid for 400 Mbit speeds. |
| PHY_SCLK | 95 | I | System Clock. This input from the PHY provides a 49.152 MHz clock signal for data synchronization. |
| PHY_LREQ | 97 | O | Link Request. This signal is driven by the OHCI-Lynx to initiate a request for the PHY to perform some service. |
| **Miscellaneous** | | | |
| SDA | 5 | I/O | Serial Data. The OHCI-Lynx determines whether a two-wire serial ROM, or no serial ROM is implemented at reset. If a two-wire serial ROM is detected, then this terminal provides the SDA serial data signaling. This terminal must be wired low to indicate no serial ROM is present. |
| SCL | 4 | I/O | Serial Clock. The OHCI-Lynx determines whether a two-wire, or no serial ROM is implemented at reset. If a two-wire serial ROM is implemented, then this terminal provides the SCL serial clock signaling. |
| ISOLATED | 79 | I | Phy–link Isolation Barrier Mode. This terminal should be asserted when the PHY device is electrically isolated from the OHCI-Lynx. This input controls bus-hold I/Os. |
| CYCLEIN | 78 | I | Cycle Input. This optional external 8KHz clock input may be used as the cycle timer clock, and can be used for synchronization with other system devices. |
| CYCLEOUT | 77 | O | Cycle Output. This optional 8 kHz output may be used for cycle timer synchronization. |
| GPIO3 | 3 | I/O | General Purpose I/O [3] |
| GPIO2 | 2 | I/O | General Purpose I/O [2] |
| LPS/GPIO1 | 99 | I/O | General Purpose I/O [1]/ Link Power Status Output. Link Power status indicates that link is powered and full functional. |
| BMC/LINKON/<br>GPIO0 | 98 | I/O | General Purpose I/O [0]/Bus Manager Contender Output/LINKON#.<br>LINKON. Receipt of a link-on packet. Once asserted LINKON shall remain asserted until LPS is asserted or the PHY register L bit is set to one |
| TEST_EN | 76 | I | |

## OHCI-Lynx controller programming model

This section describes the internal registers used to program the OHCI-Lynx, including both PCI configuration registers and Open HCI registers. All registers are detailed in the same format. A brief description is provided for each register, followed by the register offset and a bit–table describing the reset state for each register.

## OHCI-Lynx controller programming model (continued)

A bit description table is typically included that indicates bit field names, a detailed field description, and field access tags. The field access tags are described in Table 2.

**Table 2. Bit Field Access Tag Descriptions**

| ACCESS TAG | NAME | MEANING |
|---|---|---|
| r | read | Field may be read by software. |
| w | write | Field may be written by software to any value. |
| s | set | Field may be set by a write of one. Writes of zero have no effect. |
| c | clear | Field may be cleared by a write of one. Writes of zero have no effect. |
| u | update | Field may be autonomously updated by the OHCI-Lynx. |

### PCI configuration registers

The OHCI-Lynx configuration header is compliant with the PCI Specification as a standard header.Table 3 illustrates the PCI configuration header which includes both the predefined portion of the configuration space and the user definable registers. The registers that are labeled Reserved are read only returning zero when read, and are not applicable to the OHCI-Lynx design, or have been reserved by the PCI specification for future use.

**Table 3. PCI Configuration Register Map**

| REGISTER NAME | | | | OFFSET |
|---|---|---|---|---|
| Device ID | | Vendor ID | | 00h |
| Status | | Command | | 04h |
| Class Code | | | Revision ID | 08h |
| BIST | Header Type | Latency Timer | Cache Line Size | 0Ch |
| Open HCI Registers Base Address | | | | 10h |
| TI Extension Registers Base Address | | | | 14h |
| Reserved | | | | 18h |
| Reserved | | | | 1Ch |
| Reserved | | | | 20h |
| Reserved | | | | 24h |
| Reserved | | | | 28h |
| Subsystem ID | | Subsystem Vendor ID | | 2Ch |
| Reserved | | | | 30h |
| Reserved | | | Capabilites Pointer | 34h |
| Reserved | | | | 38h |
| Max Latency | Min Grant | Interrupt Pin | Interrupt Line | 3Ch |
| PCI OHCI Control Register | | | | 40h |
| Power Management Capabilities | | Next Item Pointer | Capability ID | 44h |
| PM Data | PMCSR_BSE | Power Management CSR | | 48h |
| Reserved | | | | 4C–F2h |
| Link_Enhancements Register | | | | F4h |
| Subsytem ID Alias | | Subsystem Vendor ID Alias | | F8h |
| GPIO3 | GPIO2 | GPIO1 | GPIO0 | FCh |

### *vendor ID register*

This 16-bit read only register contains a value allocated by the PCI SIG and identifies the manufacturer of the PCI device. The Vendor ID assigned to Texas Instruments is 104Ch. All bits in this register are read only.

**PCI register 00h**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |

### *device ID register*

This 16-bit read only register contains a value assigned to the OHCI-Lynx by Texas Instruments. The device identification for the OHCI-Lynx is 8009.

**PCI register 02h**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |

### *PCI command register*

The command register provides control over the OHCI-Lynx interface to the PCI bus. All bit functions adhere to the definitions in the PCI Local Bus Specification, as seen in the following bit descriptions.

**PCI register 04h**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 4. Bit Descriptions – PCI Command Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 9 | FBB_ENB | r | Fast Back-to-Back Enable. The OHCI-Lynx will not generate fast back to back transactions, thus this bit is read only and returns zero when read. |
| 8 | SERR_ENB | rw | SERR# Enable. When set, the OHCI-Lynx SERR# driver is enabled. SERR# can be asserted after detecting an address parity error on the PCI bus. |
| 7 | STEP_ENB | r | Address/Data Stepping Control. The OHCI-Lynx does not support address/data stepping, and this bit is hardwired to zero. |
| 6 | PERR_ENB | rw | Parity Error Enable. When set, the OHCI-Lynx is enabled to drive PERR# response to parity errors through the PERR# signal. |
| 5 | VGA_ENB | r | VGA Palette Snoop Enable. The OHCI-Lynx does not feature VGA palette snooping, and this bit is read only returning zeros when read. |
| 4 | MWI_ENB | rw | Memory Write and Invalidate Enable. When set, the OHCI-Lynx is enabled to generate MWI pci bus commands. If reset, the OHCI-Lynx will generate memory write commands instead. |
| 3 | SPECIAL | r | Special Cycle Enable. The OHCI-Lynx function does not respond to special cycle transactions, and this bit is read only and returns zero when read. |
| 2 | MASTER_ENB | rw | Bus Master Enable. When set, the OHCI-Lynx is enabled to initiate cycles on the PCI bus. |
| 1 | MEMORY_ENB | rw | Memory Response Enable. Setting this bit enables the OHCI-Lynx to respond to memory cycles on the PCI bus. This bit must be set to access OHCI registers. |
| 0 | IO_ENB | r | I/O Space Enable. The OHCI-Lynx does not implement any I/O mapped functionality; thus, this bit is read only and returns zeros when read. |

### *PCI status register*

PCI Bus Specification, as seen in the bit descriptions.

**PCI register 06h**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

**Table 5. Bit Descriptions – PCI Status Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15 | PAR_ERR | rcu | Detected Parity Error. This bit is set when a parity error is detected, either address or data parity errors. |
| 14 | SYS_ERR | rcu | Signaled System Error. This bit is set when $\overline{\text{SERR}}$ is enabled and the OHCI-Lynx signaled a system error to the host. |
| 13 | MABORT | rcu | Received Master Abort. This bit is set when a cycle initiated by the OHCI-Lynx on the PCI bus has been terminated by a master abort. |
| 12 | TABORT_REC | rcu | Received Target Abort. This bit is set when a cycle initiated by the OHCI-Lynx on the PCI bus was terminated by a target abort. |
| 11 | TABORT_SIG | rcu | Signaled Target Abort. This bit is set by the OHCI-Lynx when it terminates a transaction on the PCI bus with a target abort. |
| 10:9 | PCI_SPEED | r | DEVSEL Timing. These read only bits encode the timing of $\overline{\text{DEVSEL}}$ and are hardwired 01b indicating that the OHCI-Lynx asserts this signal at a medium speed on non-configuration cycle accesses. |
| 8 | DATAPAR | rcu | Data Parity Error Detected. This bit is set when the following conditions have been met:<br>a. PERR# was asserted by any PCI device including the OHCI-Lynx<br>b. The OHCI-Lynx was the bus master during the data parity error<br>c. The parity error response bit is set in the command register |
| 7 | FBB_CAP | r | Fast Back-to-Back Capable. The OHCI-Lynx cannot accept fast back to back transactions; thus, this bit is hardwired to zero. |
| 6 | UDF | r | UDF Supported. The OHCI-Lynx does not support the user definable features; thus, this bit is hardwired to zero. |
| 5 | 66MHZ | r | 66 MHz capable. The OHCI-Lynx operates at a maximum PCLK frequency of 33 MHz; therefore, this bit is hardwired to zero. |
| 4 | CAPLIST | r | Capabilities List. This bit is read only and returns one when read, and indicates that capabilities additional to standard PCI are implemented. The linked list of PCI Power Management capabilities is implemented in this function. |

### *class code and revision ID register*

This read only register categorizes the OHCI-Lynx as a serial bus controller (0Ch), controlling an IEEE1394 bus (00h), with an OHCI programming model (10h). Furthermore, the TI chip revision is indicated in the lower byte.

**PCI register 08h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Table 6. Bit Descriptions – Class Code and Revision ID Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:24 | BASECLASS | r | Base Class. This field returns 0Ch when read, which broadly classifies the function as a serial bus controller. |
| 23:16 | SUBCLASS | r | Sub Class. This field returns 00h when read, which specifically classifies the function as controlling a IEEE1394 serial bus. |
| 15:8 | PGMIF | r | Programming Interface. This field returns 10h when read, which indicates that the programming model is compliant with the 1394 OHCI specification. |
| 7:0 | CHIPREV | r | Silicon Revision. This field returns 01h when read, indicating the silicon revision of the OHCI-Lynx. |

### *latency timer and class cache line size register*

This register is programmed by host BIOS to indicate system cache line size and the latency timer associated with the OHCI-Lynx.

**PCI register 0Ch**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 7. Bit Descriptions – Latency Timer and Class Cache Line Size Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15:8 | LATENCY_TIMER | rw | PCI Latency Timer. The value in this register specifies the latency timer for the OHCI-Lynx, in units of PCI clock cycles. When the OHCI-Lynx is a PCI bus initiator and asserts $\overline{\text{FRAME}}$, the latency timer will begin counting from zero. If the latency timer expires before the OHCI-Lynx transaction has terminated, then the OHCI-Lynx will terminate the transaction when its $\overline{\text{GNT}}$ is deasserted. |
| 7:0 | CACHELINE_SZ | rw | Cache Line Size. This value is used by the OHCI-Lynx during Memory Write and Invalidate, Memory Read Line, and Memory Read Multiple transactions. |

### *header type and bist register*

This register indicates the OHCI-Lynx PCI header type, and indicates no built-in self test.

**PCI register 0Eh**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 8. Bit Descriptions – Header Type and Bist Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15:8 | BIST | r | Built-in Self Test. The OHCI-Lynx does not include a built-in self test, and this field returns zero when read. |
| 7:0 | HEADER_TYPE | r | PCI Header Type. The OHCI-Lynx includes the standard PCI header, and this is communicated by returning zero when this field is read. |

### *open HCI registers base address register*

This register is programmed with a base address referencing the memory mapped OHCI control. When BIOS writes all ones to this register, the value read back is FFFF F800h, indicating that at least 2 Kbytes of memory address space are required for the OHCI registers.

**PCI register 10h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 9. Bit Descriptions – Open HCI Registers Base Address Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:11 | OHCIREG_PTR | rw | Open HCI Register Pointer. Specifies the upper 20 bits of the 32-bit OHCI register base address. |
| 10:4 | OHCI_SZ | r | Open HCI Register Size. This read only field returns zeros when read, and indicates that the OHCI registers require a 2Kbyte region of memory. |
| 3 | OHCI_PF | r | OHCI Register Prefetch. This bit returns zero, indicating the OHCI registers are nonprefetchable. |
| 2:1 | OHCI_MEMTYPE | r | Open HCI Memory Type. This read only field returns zeros when read, and indicates that the base register is 32 bits wide and mapping can be done anywhere in the 32-bit memory space. |
| 0 | OHCI_MEM | r | OHCI Memory Indicator. This read only bit returns zero, indicating the OHCI registers are mapped into system memory space. |

### *TI extension base address register*

This register is programmed with a base address referencing the memory mapped TI extension registers. Refer to the description of the OHCI Base Address Register for bit field details.

**PCI register 14h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### *PCI subsystem identification register*

This register is used for system and option card identification purposes. This register can be initialized from the serial EEPROM.

**PCI register 2Ch**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 10. Bit Descriptions – Open HCI Registers Base Address Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:16 | | ru | Subsystem Device ID. This field indicates the subsystem device ID. |
| 15:0 | | ru | Subsystem Vendor ID. This field indicates the subsystem vendor ID. |

### *PCI power management capabilities pointer*

This register provides a pointer into the PCI configuration header where the PCI power management register block resides. OHCI-Lynx configuration header double-words at 44h and 48h provide the PM registers. This register is read only and returns 44h when read.

**PCI register 34h**

| BIT NUMBER | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |

### *interrupt line and pin registers*

This register is used to communicate interrupt line routing information.

**PCI register 3Ch**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 11. Bit Descriptions – Interrupt Line and Pin Registers**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15:8 | INTR_PIN | r | Interrupt Pin Register. This register is read only and returns 01h when read, indicating that the OHCI-Lynx PCI function signals interrupts on $\overline{\text{INTA}}$ pin. |
| 7:0 | INTR_LINE | rw | Interrupt Line Register. This register is programmed by the system and indicates to software to which interrupt line the OHCI-Lynx $\overline{\text{INTA}}$ is connected. |

### *MIN_GNT and MAX_LAT registers*

This register is used to communicate to the system the desired setting of the Latency Timer Register. If a serial ROM is detected, then the contents of this register are loaded through the serial ROM interface after a PCI reset. If no serial ROM is detected, then this register returns a default value that corresponds to the MIN_GNT = 2, MAX_LAT = 4.

**PCI register 3Eh**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

**Table 12. Bit Descriptions – MIN_GNT and MAX_LAT Registers**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15:8 | MAX_LAT | ru | Maximum Latency. The contents of this register may be used by host BIOS to assign an arbitration priority-level to the OHCI-Lynx. The default for this register indicates that the OHCI-Lynx may need to access the PCI bus as often as every 1/4 microsecond; thus, an extremely high priority level is requested. The contents of this field may also be loaded through the serial ROM. |
| 7:0 | MIN_GNT | ru | Minimum Grant. The contents of this register may be used by host BIOS to assign a Latency Timer Register value to the OHCI-Lynx. The default for this register indicates that the OHCI-Lynx may need to sustain burst transfers for nearly 64 microseconds; thus, requesting a large value be programmed in the OHCI-Lynx Latency Timer Register. |

### *PCI OHCI control register*

This register contains IEEE1394 Open HCI specific control bits. All bits in this register are read only and return zeros, since no OHCI specific control bits have been implemented.

**PCI register 40h**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### *capability ID and next item pointer registers*

This read only register identifies the linked list capability item, and provides a pointer to the next capability item.

**PCI register 44h**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Table 13. Bit Descriptions – Capability ID and Next Item Pointer Registers**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15:8 | NEXT_ITEM | r | Next Item Pointer. The OHCI-Lynx supports only one additional capability that is communicated to the system through the extended capabilities list; thus, this read only field returns 00h when read. |
| 7:0 | CAPABILITY_ID | r | Capability Identification. This field returns 01h when read, which is the unique ID assigned by the PCI SIG for PCI power management capability. |

### *power management capabilities*

This register indicates the capabilities of the OHCI-Lynx related to PCI power management. In summary, the D0, $D3_{hot}$ device states are supported.

**PCI register 46h**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

**Table 14. Bit Descriptions – Capability ID and Next Item Pointer Registers**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15:11 | PME_SUPPORT | r | PME Support. This five bit field indicates the power states from which the OHCI-Lynx may assert PME. A zero for any bit indicates that the OHCI-Lynx cannot assert PME signal from that power state. These five bits return a value of "01001b" when read . Each of these bits is described below:<br>Bit 15 – reads 0 indicating that the PME signal cannot be asserted from $D3_{cold}$ state.<br>Bit 14 – reads 1 indicating that the PME signal can be asserted from $D3_{hot}$ state.<br>Bit 13 – reads 0 indicating that the PME signal cannot be asserted from D2 state.<br>Bit 12 – reads 0 indicating that the PME signal cannot be asserted from D1 state.<br>Bit 11 – reads 0 indicating that the PME signal cannot be asserted from D0 state. |
| 10 | D2_SUPPORT | r | D2 Support. This bit returns a 0 when read, indicating that the OHCI-Lynx doesnot supports the D2 power state. |
| 9 | D1_SUPPORT | r | D1 Support. This bit returns a 0 when read, indicating that the OHCI-Lynx doesnot support the D1 power state. |
| 8 | DYN_DATA | r | Dynamic Data Support. This bit returns a zero when read, indicating that the OHCI-Lynx does not report dynamic power consumption data. |
| 7–6 | RESERVED | r | Reserved. This bit is read only and returns 0 when read. |
| 5 | DSI | r | Device Specific Initialization. This bit is read only and returns 0 when read, indicating that the OHCI-Lynx does not require special initialization beyond the standard PCI configuration header before a generic class driver is able to use it. |
| 4 | AUX_PWR | r | Auxiliary Power Source. Since the OHCI-Lynx does not support PME generation in the $D3_{COLD}$ device state, this bit returns zero when read. |
| 3 | PME_CLK | r | PME Clock. This bit is read only and returns a 0 when read indicating that no host bus clock is required for the OHCI-Lynx to generate PME. |
| 2:0 | PM_VERSION | r | Power Management Version. This field returns 001b when read, indicating that the OHCI-Lynx is compatible with the registers described in the revision 1.0 PCI Bus Power Management Specification. |

***power management control and status register***

This register implements the control and status of the PCI power management function. This register is not affected by the internally generated reset caused by the transition from the $D3_{HOT}$ to D0 state.

**PCI register 48h**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 15. Bit Descriptions – Power Management Control and Status Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15 | PME_STS | r c | This bit is set when the OHCI-Lynx would normally be asserting the PME signal, independent of the state of the PME_ENB bit. This bit is cleared by a write back of 0, and this also clears the PME# signal driven by the OHCI-Lynx. Writing a zero to this bit has no effect. |
| 14:9 | DYN_CTRL | r | Dynamic Data Control. This bit field returns zeros when read since the OHCI-Lynx does not report dynamic data. |
| 8 | PME_ENB | r w | PME Enable. This bit enables the function to assert PME. If the bit is cleared assertion of PME is disabled. |
| 4 | DYN_DATA | r | Dynamic Data. This bit field returns zeros when read since the OHCI-Lynx does not report dynamic data. |
| 1:0 | PWR_STATE | rw | Power State. This two bit field is used to set the OHCI-Lynx device power state, and is encoded as follows:<br>00 – Current power state is D0<br>01 – Current power state is D1<br>10 – Current power state is D2<br>11 – Current power state is $D3_{HOT}$ |

### *power management extension registers*

This register provides extended power management features not applicable to the OHCI-Lynx, thus it is read only and returns zero when read.

**PCI register 4Ah**

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 16. Bit Descriptions – Power Management Extension Registers**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15:8 | PM_DATA | r | Power Management Data. This bit field returns zeros when read since the OHCI-Lynx does not report dynamic data. |
| 7:0 | PMCSR_BSE | r | Power Management CSR – Bridge Support Extensions. This field returns zero since the OHCI-Lynx is not provide P2P bridging. |

### *link enhancement control register*

This implements TI proprietary bits that are initialized by software or by a serial EEPROM if present. After these bits are set, their functionality is enabled only if the aPhyEnhanceEnable bit in the HCControl register is set.

**PCI register F4h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 17. Bit Descriptions – OHCI Enhancements Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 7 | enab_unfair | rw | Enable asynchronous priority requests . When this bit is set to one , all async response packets are sent as priority requests without considering fairness protocol. This bit can be initialized from serial EEPROM. |
| 2 | enab_insert_idle | rw | Enable insert idle. When this bit is set to one, and the aPHYEnhanceEnable bit is set to one, the PHY grants the PHY/Link interface to the OHCI-Lynx with *grant*, the OHCI-Lynx takes control by asserting idle for one cycle before changing the state of CTL[0:1] to hold or transmit. |
| 1 | enab_accel | rw | Enable acceleration. When this bit is set to one and the aPhyEnhanceEnable bit is set to one, the OHCI-Lynx supports a 1394A PHY with enab_accel bit set (ack-accelerated and fly-by accelerations enabled). To support the 1394a PHY with enab_acel bit set the OHCI-Lynx :<br>● Uses Acceleration Control Request to enable or disable arbitration acceleration.<br>● Monitors the CTL[0:1] and cancels LREQ if the received packet is greater than 8 bits in length. |

### *PCI subsystem identification register*

This register is used for system and option card identification purposes. The contents of this register are aliased to Subsytem Identification register at address 2Ch.

**PCI register F8h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 18. Bit Descriptions – Open HCI Registers Base Address Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:16 | SUBDEV_ID | rw | Subsystem Device ID. This field indicates the subsystem device ID. |
| 15:0 | SUBVEN_ID | rw | Subsystem Vendor ID. This field indicates the subsystem vendor ID. |

### *GPIO control register*

This register has the control and status bits for GPIO0, GPIO1, GPIO2 and GPIO3 ports. Upon reset GPIO0 and GPIO1 default to Bus Manager Contender (BMC) and Link power status terminals respectively. The BMC terminal can be configured as GPIO0 by setting the disable_BMC bit to 1. The LPS terminal can be configured as GPIO1 by setting the disable_LPS bit to 1.

**PCI register FCh**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

**Table 19. Bit Descriptions – General Purpose Input/Output Control Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 29 | GPIO_INV3 | rw | GPIO3 Polarity Invert. This bit controls the input output polarity control of for GPIO3.<br>– 0 = noninverted (default)<br>– 1 = inverted |
| 28 | GPIO_ENB3 | rw | GPIO3 Enable Control. This bit controls the output enable for GPIO3<br>– 0 = output tristated (default)<br>– 1 = output enabled |
| 24 | GPIO_DATA3 | rw | GPIO3 Data. When GPIO3 output is enabled, the value written to this bit represents the logical data driven to the GPIO3 terminal. |
| 21 | GPIO_INV2 | rw | GPIO2 Polarity Invert. This bit controls the input output polarity control of for GPIO2.<br>– 0 = noninverted (default)<br>– 1 = inverted |
| 20 | GPIO_ENB2 | rw | GPIO2 Enable Control. This bit controls the output enable for GPIO2<br>– 0 = output 3-stated (default)<br>– 1 = output enabled |
| 16 | GPIO_DATA2 | rw | GPIO2 Data. When GPIO2 output is enabled, the value written to this bit represents the logical data driven to the GPIO3 terminal. |
| 15 | DISABLE_LPS | rw | DISABLE Link Power Status (LPS). This bit configures this terminals as<br>– 0 = LPS (default)<br>1 = GPIO1 |
| 13 | GPIO_INV1 | rw | GPIO1 Polarity Invert. When DISABLE_LPS bit is set to 1, this bit controls the input output polarity control of for GPIO1<br>– 0 = noninverted (default)<br>– 1 = inverted |
| 12 | GPIO_ENB1 | rw | GPIO1 Enable Control. When DISABLE_LPS bit is set to 1, this bit controls the output enable for GPIO1<br>– 0 = output tristated<br>– 1 = output enabled (default) |
| 8 | GPIO_DATA1 | rw | GPIO1 Data. When DISABLE_LPS bit is set to 1 and GPIO1 output is enabled, the value written to this bit represents the logical data driven to the GPIO1 terminal. |
| 7 | DISABLE_BMC | rw | Disable Bus Manager Contender (BMC). This bit configures this terminals as<br>– 0 = BMC (default)<br>– 1 = GPIO0 |

**Table 19. Bit Descriptions – General Purpose Input/Output Control Register(Continued)**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 5 | GPIO_INV0 | rw | GPIO0 Polarity Invert. When DISABLE_BMC bit is set to 1, this bit controls the input output polarity control of for GPIO0<br>– 0 = non-inverted (default)<br>– 1 = inverted |
| 4 | GPIO_ENB0 | rw | GPIO0 Enable Control. When DISABLE_BMC bit is set to 1, this bit controls the output enable for GPIO0<br>– 0 = output tristated<br>– 1 = output enabled (default) |
| 0 | GPIO_DATA0 | rw | GPIO0 Data. When DISABLE_BMC bit is set to 1 and GPIO0 output is enabled, the value written to this bit represents the logical data driven to the GPIO0 terminal. |

### open HCI registers

The Open HCI registers defined by the IEEE1394 Open HCI Specification are memory mapped into a 2Kbyte region of memory pointed to by the OHCI Base Address Register at offset 10h in PCI configuration space. These registers are the primary interface for controlling the OHCI-Lynx IEEE1394 link function.

This section provides the register interface and bit descriptions. There are several set and clear register pairs in this programming model, which are implemented to solve various issues with typical read-modify-write control registers. There are two addresses for a set/clear register: RegisterSet and RegisterClear. Refer to Table 20 for an illustration. A one bit written to RegisterSet causes the corresponding bit in the set/clear register to be set, while a zero bit leaves the corresponding bit unaffected. A one bit written to RegisterClear causes the corresponding bit in the set/clear register to be reset, while a zero bit leaves the corresponding bit in the set/clear register unaffected.

Typically, a read from either RegisterSet or RegisterClear returns the value of the set/clear register. However, sometimes reading the RegisterClear will provide a masked version of the set/clear register. The IntEvent register is an example of this behavior.

**Table 20. Open HCI Register Map**

| DMA CONTEXT | REGISTER NAME | ABBREVIATION | OFFSET |
|---|---|---|---|
| — | OHCI Version Register | Version | 00h |
| | Global Unique ID ROM Register | GUID_ROM | 04h |
| | Asynchronous Transmit Retries Register | ATRetries | 08h |
| | CSR Data Register | CSRData | 0Ch |
| | CSR Compare Data Register | CSRCompareData | 10h |
| | CSR Control Register | CSRControl | 14h |
| | Configuration ROM Header Register | ConfigROMhdr | 18h |
| | Bus Identification Register | BusID | 1Ch |
| | Bus Options Register | BusOptions | 20h |
| | Global Unique ID High Register | GUIDHi | 24h |
| | Global Unique ID Low Register | GUIDLo | 28h |
| | Reserved | — | 2Ch |
| | Reserved | — | 30h |
| | Configuration ROM Map Register | ConfigROMmap | 34h |
| | Posted Write Address Low Register | PostedWriteAddressLo | 38h |
| | Posted Write Address High Register | PostedWriteAddressHi | 3Ch |
| | Vendor Identification Register | VendorID | 40h |
| | Reserved | — | 44h – 4Ch |

**Table 20. Open HCI Register Map (Continued)**

| DMA CONTEXT | REGISTER NAME | ABBREVIATION | OFFSET |
|---|---|---|---|
| — | Host Controller Control Register | HCControlSet | 50h |
| | | HCControlClr | 54h |
| | Reserved | — | 58h |
| | Reserved | — | 5Ch |
| Self ID | Reserved | — | 60h |
| | Self ID Buffer Register | SelfIDBuffer | 64h |
| | Self ID Count Register | SelfIDCount | 68h |
| | Reserved | — | 6Ch |
| — | Isochronous Receive Channel Mask High Register | IRChannelMaskHiSet | 70h |
| | | IRChannelMaskHiClear | 74h |
| | Isochronous Receive Channel Mask Low Register | IRChannelMaskLoSet | 78h |
| | | IRChannelMaskLoClear | 7Ch |
| | Interrupt Event Register | IntEventSet | 80h |
| | | IntEventClear | 84h |
| | Interrupt Mask Register | IntMaskSet | 88h |
| | | IntMaskClear | 8Ch |
| | Isochronous Transmit Interrupt Event Register | IsoXmitIntEventSet | 90h |
| | | IsoXmitIntEventClear | 94h |
| | Isochronous Transmit Interrupt Mask Register | IsoXmitIntMaskSet | 98h |
| | | IsoXmitIntMaskClear | 9Ch |
| — | Isochronous Receive Interrupt Event Register | IsoRecvIntEventSet | A0h |
| | | IsoRecvIntEventClear | A4h |
| | Isochronous Receive Interrupt Mask Register | IsoRecvIntMaskSet | A8h |
| | | IsoRecvIntMaskClear | ACh |
| | Reserved | | B0–D8h |
| | Fairness Control Register | FairnessControl | DCh |
| | Link Control Register | LinkControlSet | E0h |
| | | LinkControlClear | E4h |
| | Node Identification Register | NodeID | E8h |
| | Phy Layer Control Register | PhyControl | ECh |
| | Isochronous Cycle Timer Register | IsoCycleTimer | F0h |
| | Reserved | — | F4h – FCh |
| | Asynchronous Request Filter High Register | AsyncRequestFilterHiSet | 100h |
| | | AsyncRequestFilterHiClear | 104h |
| | Asynchronous Request Filter Low Register | AsyncRequestFilterLoSet | 108h |
| | | AsyncRequestFilterloClear | 10Ch |
| | Physical Request Filter High Register | PhysicalRequestFilterHiSet | 110h |
| | | PhysicalRequestFilterHiClear | 114h |
| | Physical Request Filter Low Register | PhysicalRequestFilterLoSet | 118h |
| | | PhysicalRequestFilterloClear | 11Ch |
| | Physical Upper Bound Register | PhysicalUpperBound | 120h |
| | Reserved | — | 124h – 17Ch |

**Table 20. Open HCI Register Map (Continued)**

| DMA CONTEXT | REGISTER NAME | ABBREVIATION | OFFSET |
|---|---|---|---|
| Asychronous Request Transmit [ ATRQ ] | Context Control Register | ContextControlSet | 180h |
| | | ContextControlClear | 184h |
| | Reserved | — | 188h |
| | Command Pointer Register | CommandPtr | 18Ch |
| Asychronous Response Transmit [ ATRS ] | Reserved | — | 190h – 19Ch |
| | Context Control Register | ContextControlSet | 1A0h |
| | | ContextControlClear | 1A4h |
| | Reserved | — | 1A8h |
| | Command Pointer Register | CommandPtr | 1ACh |
| Asychronous Request Receive [ ARRQ ] | Reserved | — | 1B0h – 1BCh |
| | Context Control Register | ContextControlSet | 1C0h |
| | | ContextControlClear | 1C4h |
| | Reserved | — | 1C8h |
| | Command Pointer Register | CommandPtr | 1CCh |
| Asychronous Response Receive [ ARRS ] | Reserved | — | 1D0h – 1DCh |
| | Context Control Register | ContextControlSet | 1E0h |
| | | ContextControlClear | 1E4h |
| | Reserved | — | 1E8h |
| | Command Pointer Register | CommandPtr | 1ECh |
| Isochronous Transmit Context n n=0,1,2,3, …7 | Reserved | — | 1F0h – 1FCh |
| | Context Control Register | ContextControlSet | 200h + 16*n |
| | | ContextControlClear | 204h + 16*n |
| | Reserved | — | 208h + 16*n |
| Isochronous Receive Context n n=0,1,2,3, …4 | Command Pointer Register | CommandPtr | 20Ch + 16*n |
| | Context Control Register | ContextControlSet | 400h + 32*n |
| | | ContextControlClear | 404h + 32*n |
| | Reserved | — | 408h + 32*n |
| | Command Pointer Register | CommandPtr | 40Ch + 32*n |
| | Context Match Register | ContextMatch | 410h + 32*n |

### *OHCI version register*

This register indicates the OHCI version support, and whether or not the serial ROM is present.

**OHCI register 00h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | X | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 21. Bit Descriptions – OHCI Version Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 24 | GUID_ROM | r | The OHCI-Lynx sets this bit if the serial ROM is detected. If the serial ROM is present, then the Bus_Info_Block will be automatically loaded on h/w reset. |
| 23:16 | version | r | Major version of the Open HCI. The OHCI-Lynx is compliant with the OHCI specification version 1.00; thus, this field reads 8'h01. |
| 7:0 | revision | r | Minor version of the Open HCI. The OHCI-Lynx is compliant with the OHCI specification version 1.00; thus, this field reads 8'h00. |

### *GUID ROM register*

This register is used to access the serial ROM, and is only applicable if the Version.GUID_ROM bit is set.

**OHCI register 04h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 22. Bit Descriptions – GUID ROM Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | addrReset | rsu | Software sets this bit to reset the GUID ROM address to zero. When the OHCI-Lynx completes the reset, it clears this bit. The OHCI-Lynx does not automatically fill rdData with the $0^{th}$ byte. |
| 25 | rdStart | rsu | A read of the currently addressed byte is started when this bit is set. This bit is automatically cleared when the OHCI-Lynx completes the read of the currently addressed GUID ROM byte. |
| 23:16 | rdData | ru | This field represents the data read from the GUID ROM. |

### *asynchronous transmit retries register*

This register indicates the number of times the OHCI-Lynx will attempt a retry for asynchronous DMA request transmit and for asynchronous physical and DMA response transmit.

**OHCI register 08h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 23. Bit Descriptions – Asynchronous Transmit Retries Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:29 | secondLimit | r | The second limit field returns zeros when read, since outbound dual-phase retry is not implemented. |
| 28:16 | cycleLimit | r | The cycle limit field returns zeros when read, since outbound dual-phase retry is not implemeted. |
| 11:8 | maxPhysRespRetries | rw | The maxPhysRespRetries field tells the Physical Response Unit how many times to attempt to retry the transmit operation for the response packet when a busy acknowledge or ack_data_error is received from the target node. |
| 7:4 | maxATRespRetries | rw | The maxATRespRetries field tells the Asynchronous Transmit Response Unit how many times to attempt to retry the transmit operation for the response packet when a busy acknowledge or ack_data_error is received from the target node. |
| 3:0 | maxATReqRetries | rw | The maxATReqRetries field tells the Asynchronous Transmit DMA Request Unit how many times to attempt to retry the transmit operation for the response packet when a busy acknowledge or ack_data_error is received from the target node. |

### *CSR data register*

This register is used to access the bus management CSR registers from the host through compare-swap operations. This register contains the data to be stored in a CSR if the compare is successful.

**OHCI register 0Ch**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

### *CSR compare register*

This register is used to access the bus management CSR registers from the host through compare-swap operations. This register contains the data to be compared with the existing value of the CSR resource.

**OHCI register 10h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

### *CSR control register*

This register is used to access the bus management CSR registers from the host through compare-swap operations. This register is used to control the compare-swap operation and select the CSR resource.

**OHCI register 14h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x |

**Table 24. Bit Descriptions – CSR Control Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | csrDone | ru | This bit is set by the OHCI-Lynx when a compare-swap operation is complete. It is reset whenever this register is written. |
| 1:0 | csrSel | rw | This field selects the CSR resource as follows:<br>00 – BUS_MANAGER_ID<br>01 – BANDWIDTH_AVAILABLE<br>10 – CHANNELS_AVAILABLE_HI<br>11 – CHANNELS_AVAILABLE_LO |

### *configuration ROM header register*

This register externally maps to the first quadlet of the 1394 configuration ROM, offset 48'hFFFF_F000_0400.

**OHCI register 18h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

**Table 25. Bit Descriptions – Configuration ROM Header Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:24 | info_length | rw | IEEE1394 bus mgmt field. Must be valid when HCControl.linkEnable bit is set. |
| 23:16 | crc_length | rw | IEEE1394 bus mgmt field. Must be valid when HCControl.linkEnable bit is set. |
| 15:0 | rom_crc_value | rw | IEEE1394 bus management field. Must be valid at any time the HCControl.linkEnable bit is set. The reset value is undefined if no serial ROM is present. If a serial ROM is present, this field is loaded from the serial ROM. |

### *bus identification register*

This register externally maps to the first quadlet in the Bus_Info_Block, and contains the constant 32'h31333934, which is the ASCII value of 1394.

**OHCI register 1Ch**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |

### *bus options register*

This register externally maps to the second quadlet of the Bus_Info_Block.

**OHCI register 20h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | x | x | 0 | 0 | 0 | 0 | 1 | 0 |

**Table 26. Bit Descriptions – Bus Options Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | irmc | rw | Isochronous Resource Manager Capable. IEEE1394 bus management field. Must be valid when HCControl.linkEnable bit is set. |
| 30 | cmc | rw | Cycle Master Capable. IEEE1394 bus management field. Must be valid when HCControl.linkEnable bit is set. |
| 29 | isc | rw | Isochronous Support Capable. IEEE1394 bus management field. Must be valid when HCControl.linkEnable bit is set. |
| 28 | bmc | rw | Bus Manager Capable. IEEE1394 bus management field. Must be valid when HCControl.linkEnable bit is set. |
| 27 | pmc | rw | IEEE1394 bus management field. Must be valid when HCControl.linkEnable bit is set. |
| 23:16 | cyc_clk_acc | rw | Cycle Master Clock Accuracy. (accuracy in parts per million) IEEE1394 bus management field. Must be valid when HCControl.linkEnable bit is set. |
| 15:12 | max_rec | rw | IEEE 1394 bus management field. Hardware shall initialize max_rec to indicate the maximum number of bytes in a block request packet that is supported by the implementation. This value, max_rec_bytes must be 512 greater, and is calculated by 2 ^ (max_rec + 1). Software may change max_rec, however this field must be valid at any time the HCControl.linkEnable bit is set. A received block write request packet with a length grater than max_rec_bytes may generate an ack_type_error. This field is not affected by a soft reset, and defaults to value indicating 2048 bytes on hard reset. |
| 7:6 | g | rw | Generation Counter. This filed sahll be incremented is any portion the configuration ROM has incremented since the proir bus reset. |
| 2:0 | Lnk_spd | r | Link Speed. This field is implemented as read only and returns 010, indicating that the link speeds of 100, 200 and 400 Mbits/s are supported. |

### *GUID high register*

This register represents the upper quadlet in a 64-bit global unique ID (GUID) which maps to the third quadlet in the Bus_Info_Block. This register contains node_vendor_ID and chip_ID_hi fields. This register initializes to zeros on a hardware reset, which is an illegal GUID value. If a serial ROM is detected, then the contents of this register are loaded through the serial ROM interface after a PCI reset. At that point, the contents of this register cannot be changed. If no serial ROM is detected, then this register may be written once to set the value of this register. At that point, the contents of this register cannot be changed.

**OHCI register 24h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### *GUID low register*

This register represents the lower quadlet in a 64-bit global unique ID (GUID) which maps to chip_ID_lo in the Bus_Info_Block. This register initializes to zeros on a hardware reset, and behaves identical to the GUID High Register detailed in section 0.

**OHCI register 28h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### *configuration ROM mapping register*

This register contains the start address within system memory that will map to the start address of 1394 configuration ROM for this node.

**OHCI register 34h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 27. Bit Descriptions – Configuration ROM Mapping Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:10 | configROMaddr | rw | If a quadlet read request to 1394 offset 48'hFFFF_F000_0400 through offset 48'hFFFF_F000_07FF is received, then the low order 10 bits of the offset are added to this register to determine the host memory address of the read request. |

### *posted write address low register*

This register is used to communicate error information if a write request is posted and an error occurs while writing the posted data packet.

**OHCI register 38h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

**Table 28. Bit Descriptions – Posted Write Address Low Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:0 | offsetLo | ru | The low 32-bits of the 1394 destination offset of the write request that failed. |

### *posted write address high register*

This register is used to communicate error information if a write request is posted and an error occurs while writing the posted data packet.

**OHCI register 3Ch**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

**Table 29. Bit Descriptions – Posted Write Address High Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:16 | sourceID | ru | This bus and node number of the node that issued the write request that failed. |
| 15:0 | offsetHi | ru | The upper 16-bits of the 1394 destination offset of the write request that failed. |

### *vefndor ID register*

The vendor ID register holds the company ID of an organization that specifies any vendor-unique registers. The OHCI-Lynx does not implement Texas Instruments unique behavior with regards to Open HCI. Thus this register is read only returning zeros when read.

**OHCI register 40h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### *host controller control register*

This set/clear register pair provides flags for controlling the OHCI-Lynx.

**OHCI register 50h** **set register**
**OHCI register 54h** **clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 30. Bit Descriptions – Host Controller Control Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 30 | noByteSwapData | rsc | This bit is used to control whether physical accesses to locations outside the OHICLynx itself as well as any other DMA data accesses should be swapped. |
| 23 | programPhyEnable | rc | This bit informs upper level software that lower level software has consistently configured the p1394a enhancements in the Link and PHY. When 1 generic software such as the OHCI driver is responsible for configuring p1394a enhancements in the PHY and the aPhyEnhanceEnable bit in the OHCI-Lynx. When zero the generic software may not modify the p1394a enhancements in the OHCI-Lynx or PHY and cannot interpret the setting of aPhyEnhanceEnable. This bit can be initialized from serial EEPROM. |
| 22 | aPhyEnhanceEnable | rsc | When the programPhyenable is one and link enable is one , the OHCI driver can set this bit to use all p1394a enhancements. When programPhyEnable is set zero, the software shall not change PHY enhancements or the aPhyEnhanceEnable bit. |
| 19 | LPS | rsc | This bit is used to control the Link Power Status. Software must set LPS to 1 to permit the link-PHY communication. A zero prevents link-PHY communication. |
| 18 | postedWriteEnable | rsc | This bit is used to enable (1) or disable (0) posted writes. Software should change this bit only when linkEnable is 0. |
| 17 | linkEnable | rsc | This bit is cleared to 0 by a hardware reset or software reset. Software must set this bit to 1 when the system is ready to begin operation and then force a bus reset. This bit is necessary to keep other nodes from sending transactions before the local system is ready. When this bit is clear the OHCI-Lynx is logically and immediately disconnected from the 1394 bus, no packets will be received or processed nor will packets be transmitted. |
| 16 | SoftReset | rscu | When set to 1, all OHCI-Lynx state is reset, all FIFO's are flushed, and all OHCI registers are set to their hardware reset values unless otherwise specified. PCI registers are not affected by this bit. This bit remains set to 1 while the softReset is in progress, and reverts back to 0 when the reset has completed. |

### *self ID buffer pointer register*

This register points to the 2 Kbyte aligned base address of the buffer in host memory where the self ID packets will be stored during bus initialization. Bits 31:11 are read/write accessible.

**OHCI register 64h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### *self ID count register*

This register keeps a count of the number of times the bus self ID process has occurred, flags self ID packet errors, and keeps a count of the amount of self ID data in the self ID buffer.

**OHCI register 68h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 31. Bit Descriptions – Self ID Count Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | selfIDError | ru | When this bit is one, and error was detected during the most recent self ID packet reception. The contents of the self ID buffer are undefined. This bit is cleared after a self ID reception in which no errors are detected. Note that an error can be a hardware error or a host bus write error. |
| 23:16 | selfIDGeneration | ru | The value in this field increments each time a bus reset is detected. This field rolls over to 0 after reaching 255. |
| 10:2 | selfIDSize | ru | This field indicates the number of quadlets that have been written into the self ID buffer for the current selfIDGeneration. This includes the header quadlet and the self ID data. This field is cleared to zero when the self ID reception begins. |

### *ISO receive channel mask high register*

This set/clear register is used to enable packet receives from the upper 32 isochronous data channels. A read from either the set register or clear register returns the value of the IRChannelMaskHi register.

**OHCI register 70h set register**
**OHCI register 74h clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

**Table 32. Bit Descriptions – ISO Receive Channel Mask High Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | isoChannel63 | rsc | When set, the OHCI-Lynx is enabled to receive from iso channel number 63. |
| 30 | isoChannel62 | rsc | When set, the OHCI-Lynx is enabled to receive from iso channel number 62. |
| | | | Bits 29 through 2 follow the same pattern |
| 1 | isoChannel33 | rsc | When set, the OHCI-Lynx is enabled to receive from iso channel number 33. |
| 0 | isoChannel32 | rsc | When set, the OHCI-Lynx is enabled to receive from iso channel number 32. |

### *ISO receive channel mask low register*

This set/clear register is used to enable packet receives from the lower 32 isochronous data channels.

**OHCI register 78h set register**
**OHCI register 7Ch clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

**Table 33. Bit Descriptions –**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | isoChannel31 | rsc | When set, the OHCI-Lynx is enabled to receive from iso channel number 31. |
| 30 | isoChannel30 | rsc | When set, the OHCI-Lynx is enabled to receive from iso channel number 30. |
| | | | Bits 29 through 2 follow the same pattern |
| 1 | isoChannel1 | rsc | When set, the OHCI-Lynx is enabled to receive from iso channel number 1. |
| 0 | isoChannel0 | rsc | When set, the OHCI-Lynx is enabled to receive from iso channel number 0. |

### *interrupt event register*

This set/clear register reflects the state of the various OHCI-Lynx interrupt sources. The interrupt bits are set by an asserting edge of the corresponding interrupt signal, or by writing a one in the corresponding bit in the set register. The only mechanism to clear the bits in this register is to write a one to the corresponding bit in the clear register.

**OHCI register 80h set register**
**OHCI register 84h clear register [returns IntEvent and IntMask when read]**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | x | 0 | 0 | 0 | x | x | x | x | x | x | x | x | 0 | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x | x | x |

**Table 34. Bit Descriptions – Interruput Event Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 30 | vendorSpecific | | Vendor defined. |
| 26 | phyRegRcvd | rscu | The OHCI-Lynx has received a PHY register data byte which can be read from the PHY control register. |
| 25 | cycleTooLong | rscu | If LinkControl.cycleMaster is set, this indicated that over 125 μs elapsed between the start of sending a cycle start packet and the end of a subaction gap. LinkControl.cycleMaster is cleared by this event. |
| 24 | unrecoverableError | rscu | This event occurs when the OHCI-Lynx encounters any error that forces it to stop operations on any or all of its subunits. For example, when a DMA context sets its dead bit. While unrecoverableError is set, all normal interrupts for the context(s) that caused this interrupt will be blocked from being set. |
| 23 | cycleInconsistent | rscu | A cycle start was received that had an isochronous cycleTimer.seconds and isochronous cycleTimer.count different from the value in the CycleTimer register. |

**Table 34. Interrupt Event Register (Continued)**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 22 | cycleLost | rscu | A lost cycle is indicated when no cycle_start packet is sent/received between two successive cycleSynch events. A lost cycle can be predicted when a cycle_start packet does not immediately follow the first subaction gap after the cycleSynch event or if an arbitration reset gap is detected after a cycleSynch event without an intervening cycle start. CycleLost may be set either when it occurs or when logic predicts that it will occur. |
| 21 | cycle64Seconds | rscu | Indicates that the 7$^{th}$ bit of the cycle second counter has changed. |
| 20 | cycleSynch | rscu | Indicates that a new isochronous cycle has started. Set when the low order bit of the cycle count toggles. |
| 19 | phy | rscu | Indicates the PHY requests an interrupt through a status transfer. |
| 17 | busReset | rscu | Indicates that the PHY chip has entered bus reset mode. |
| 16 | selfDcomplete | rscu | A selfID packet stream has been received. It will be generated at the end of the bus initialization process. This bit is turned off simultaneously when IntEvent.busReset is turned on. |
| 9 | lockRespErr | rscu | Indicates that the OHCI-Lynx sent a lock response for a lock request to a serial bus register, but did not receive an ack_complete. |
| 8 | postedWriteErr | rscu | Indicates that a host bus error occurred while the OHCI-Lynx was trying to write a 1394 write request, which had already been given an ack_complete, into system memory. |
| 7 | isochRx | ru | Isochronous Receive DMA interrupt. Indicates that one or more isochronous receive contexts have generated and interrupt. This is not a latched event, it is the OR'ing of all bits in (isoRecvIntEvent & isoRecvIntMask). The isoRecvIntEvent register indicates which contexts have interrupted. |
| 6 | iscohTx | ru | Isochronous Transmit DMA interrupt. Indicates that one or more isochronous transmit contexts have generated and interrupt. This is not a latched event, it is the OR'ing of all bits in (isoXmitIntEvent & isoXmitIntMask). The isoXmitIntEvent register indicates which contexts have interrupted. |
| 5 | RSPkt | rscu | Indicates that a packet was sent to an asynchronous receive response context buffer and the descriptor's xferStatus and resCount fields have been updated. |
| 4 | RQPkt | rscu | Indicates that a packet was sent to an asynchronous receive request context buffer and the descriptor's xferStatus and resCount fields have been updated. |
| 3 | ARRS | rscu | Async Receive Response DMA interrupt. This bit is conditionally set upon completion of an ARRS context command descriptor. |
| 2 | ARRQ | rscu | Async Receive Request DMA interrupt. This bit is conditionally set upon completion of an ARRQ DMA context command descriptor. |
| 1 | respTxComplete | rscu | Asynchronous response transmit DMA interrupt. This bit is conditionally set upon completion of an ATRS DMA command. |
| 0 | reqTxComplete | rscu | Asynchronous request transmit DMA interrupt. This bit is conditionally set upon completion of an ATRQ DMA command. |

### *interrupt mask register*

This set/clear register is used to enable the various OHCI-Lynx interrupt sources. Reads from either the set register or the clear register always return IntMask. In all cases except masterIntEnable (bit 31), the enables for each interrupt event align with the event register bits detailed in Table 32.

**OHCI register 88h set register**
**OHCI register 8Ch clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | 0 | 0 | 0 | x | x | x | x | x | x | x | x | 0 | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x | x | x |

**Table 35. Bit Descriptions – Interrupt Mask Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | masterIntEnable | rsc | When set, external interrupts will be generated in accordance with the IntMask register. If clear, no external interrupts will be generated. |
| 30:0 | | | See Table 34 |

### *isochronous transmit interrupt event register*

This set/clear register reflects the interrupt state of the isochronous transmit contexts. An interrupt is generated on behalf of an isochronous transmit context if an OUTPUT_LAST command completes and its interrupt bits are set. Upon determining that the IntEvent.isochTx interrupt has occurred, software can check this register to determine which context(s) caused the interrupt. The interrupt bits are set by an asserting edge of the corresponding interrupt signal, or by writing a one in the corresponding bit in the set register. The only mechanism to clear the bits in this register is to write a one to the corresponding bit in the clear register.

**OHCI register 90h** **set register**
**OHCI register 84h** **clear register [returns IsoXmitEvent and IsoXmitMask when read]**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |

**Table 36. Bit Descriptions – Isochronous Transmit Interrupt Event Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 7 | isoXmit7 | rsc | Isochronous transmit channel 7 caused the isochTx interrupt. |
| 6 | isoXmit6 | rsc | Isochronous transmit channel 6 caused the isochTx interrupt. |
| 5 | isoXmit5 | rsc | Isochronous transmit channel 5 caused the isochTx interrupt. |
| 4 | isoXmit4 | rsc | Isochronous transmit channel 4 caused the isochTx interrupt. |
| 3 | isoXmit3 | rsc | Isochronous transmit channel 3 caused the isochTx interrupt. |
| 2 | isoXmit2 | rsc | Isochronous transmit channel 2 caused the isochTx interrupt. |
| 1 | isoXmit1 | rsc | Isochronous transmit channel 1 caused the isochTx interrupt. |
| 0 | isoXmit0 | rsc | Isochronous transmit channel 0 caused the isochTx interrupt. |

### *isochronous transmit interrupt mask register*

This set/clear register is used to enable the isochTx interrupt source on a per channel basis. Reads from either the set register or the clear register always return IsoXmitIntMask. In all cases the enables for each interrupt event align with the event register bits detailed in Table 36.

**OHCI register 98h** **set register**
**OHCI register 9Ch** **clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |

### *isochronous receive interrupt event register*

This set/clear register reflects the interrupt state of the isochronous receive contexts. An interrupt is generated on behalf of an isochronous receive context if an INPUT_* command completes and its interrupt bits are set. Upon determining that the IntEvent.isochRx interrupt has occurred, software can check this register to determine which context(s) caused the interrupt. The interrupt bits are set by an asserting edge of the corresponding interrupt signal, or by writing a one in the corresponding bit in the set register. The only mechanism to clear the bits in this register is to write a one to the corresponding bit in the clear register.

**OHCI register A0h set register**
**OHCI register A4h celar register [returns IsoRecvEvent and IsoRecvMask when read]**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x |

**Table 37. Bit Descriptions – Isochronous Receive Interrupt Event Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 3 | isoRecv3 | rsc | Isochronous receive channel 3 caused the isochRx interrupt. |
| 2 | isoRecv2 | rsc | Isochronous receive channel 2 caused the isochRx interrupt. |
| 1 | isoRecv1 | rsc | Isochronous receive channel 1 caused the isochRx interrupt. |
| 0 | isoRecv0 | rsc | Isochronous receive channel 0 caused the isochRx interrupt. |

### *isochronous receive interrupt mask register*

This set/clear register is used to enable the isochRx interrupt source on a per channel basis. Reads from either the set register or the clear register always return IsoRecvIntMask. In all cases the enables for each interrupt event align with the event register bits detailed in Table 37.

**OHCI register A8h set register**
**OHCI register ACh clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x |

### *fairness control register (optional register)*

This register provides a mechanism by which software can direct the Host Controller to transmit multiple asynchronous requests during a fairness interval.

**OHCI register DCh**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 38. Bit Descriptions – Link Control Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 7–0 | pri_req | r | This field specifies the maximum number of priority arbitration requests for asynchronous request packets that the link is permitted to make of the PHY during fairness interval. |

### *link control register*

This set/clear register provides the control flags that enable and configure the link core protocol portions of the OHCI-Lynx. It contains controls for the receiver and cycle timer.

**OHCI register E0h set register**
**OHCI register E4h clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | x | x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 39. Bit Descriptions – Link Control Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 22 | cycleSource | rsc | When one, the cycle timer will use an external source (CYCLEIN) to determine when to roll over the cycle timer. When zero, the cycle timer rolls over when the timer reaches 3072 cycles of the 24.576 MHz clock (125 us). |
| 21 | cycleMaster | rscu | When set, and the PHY has notified the OHCI-Lynx that it is root, the OHCI-Lynx will generate a cycle start packet every time the cycle timer rolls over, based on the setting of the cycleSource bit. When zero, the OHICLynx will accept received cycle start packets to maintain synchronization with the node which is sending them. This bit is automatically reset when the cycleTooLong event occurs and cannot be set until the IntEvent.cycleTooLong bit is cleared. |
| 20 | CycleTimerEnable | rsc | When one, the cycle timer offset will count cycles of the 24.576 MHz clock and roll over at the appropriate time based on the settings of the above bits. When zero, the cycle timer offset will not count. |
| 10 | RcvPhyPkt | rsc | When one, the receiver will accept incoming PHY packets into the AR request context if the AR request context is enabled. This does not control receipt of self-identification packets. |
| 9 | RcvSelfID | rsc | When one, the receiver will accept incoming self-identification packets. Before setting this bit to one, software must ensure that the self ID buffer pointer register contains a valid address. |

### *node identification register*

This register contains the address of the node on which the OHICLynx chip resides, and indicates the valid node number status. The 16-bit combination of busNumber and NodeNumber is referred to as the node ID.

**OHCI register E8h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | x | x | x | x | x | x |

**Table 40. Bit Descriptions – Node Identification Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | iDValid | ru | This bit indicates whether or not the OHCI-Lynx has a valid node number. It is cleared when a 1394 bus reset is detected and set when the OHCI-Lynx receives a new node number from the PHY. |
| 30 | root | ru | This bit is set during the bus reset process if the attached PHY is root. |
| 27 | CPS | ru | Set if the PHY is reporting that cable power status is OK (VP 8V). |
| 15:6 | busNumber | rwu | This number is used to identify the specific 1394 bus the OHCI-Lynx belongs to when multiple 1394-compatiable busses are connected via a bridge. |
| 5:0 | NodeNumber | ru | This number is the physical node number established by the PHY during self-identification. It is automatically set to the value received from the PHY after the self-identification phase. If the PHY sets the nodeNumber to 63, software should not set ContextControl.run for either of the AT DMA contexts. |

### *PHY control register*

This register is used to read or write a PHY register.

**OHCI register ECh**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | 0 | 0 | 0 | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x | x | x | x | x |

**Table 41. Bit Descriptions – PHY Control Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | rdDone | ru | This bit is cleared to 0 by the OHCI-Lynx when either rdReg or wrReg is set to 1. This bit is set to 1 when a register transfer is received from the PHY. |
| 27:24 | rdAddr | ru | This is the address of the register most recently received from the PHY. |
| 23:16 | rdData | ru | This field is the contents of a PHY register which has been read. |
| 15 | rdReg | rwu | This bit is set by software to initiate a read request to a PHY register, and is cleared by hardware when the request has been sent. The wrReg and rdReg bits must be used exclusively. |
| 14 | wrReg | rwu | This bit is set by software to initiate a write request to a PHY register, and is cleared by hardware when the request has been sent. The wrReg and rdReg bits must be used exclusively. |
| 11:8 | regAddr | rw | This field is the address of the PHY register to be written or read. |
| 7:0 | wrData | rw | This field is the data to be written to a PHY register, and is ignored for reads. |

### *isochronous cycle timer register*

This read/write register indicates the current cycle number and offset. When the OHCI-Lynx is cycle master, this register is transmitted with the cycle start message. When the OHCI-Lynx is not cycle master, this register is loaded with the data field in an incoming cycle start. In the event that the cycle start message is not received, the fields can continue incrementing on their own (if programmed) to maintain a local time reference.

**OHCI register F0h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

**Table 42. Bit Descriptions – Isochronous Cycle Timer Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:25 | cycleSeconds | rwu | This field counts seconds (cycleCount rollovers) modulo 128. |
| 24:12 | cycleCount | rwu | This field counts cycles (cycleOffset rollovers) modulo 8000. |
| 11:0 | cycleOffset | rwu | This field counts 24.576 MHz clocks modulo 3072, i.e., 125 µs. If an external 8 kHz clock configuration is being used, cycleOffset must be set to 0 at each tick of the external clock. |

### *asynchronous request filter high register*

This set/clear register is used to enable asynchronous receive requests on a per node basis, and handles the upper node IDs. When a packet is destined for either the physical request context or the ARRQ context, the source node ID is examined. If the bit corresponding to the node ID is not set in this register, then the packet is not acknowledged and the request is not queued. The node ID comparison is done if the source node is on the same bus as the OHCI-Lynx. All non-local bus sourced packets are not acknowledged unless bit 31 in this register is set.

**OHCI register 100h set register**
**OHCI register 104h clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 43. Bit Descriptions – Asynchronous Request Filter High Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | asynReqAllBuses | rsc | If set to one, all asynchronous requests received by the OHCI-Lynx from non–local bus nodes will be accepted. |
| 30 | asynReqResource62 | rsc | If set to one for local bus node number 62, asynchronous requests received by the OHCI-Lynx from that node will be accepted. |
| | | | Bits29 through 2 follow the same pattern as below. |
| 1 | asynReqResource33 | rsc | If set to one for local bus node number 33, asynchronous requests received by the OHCI-Lynx from that node will be accepted. |
| 0 | asynReqResource32 | rsc | If set to one for local bus node number 32, asynchronous requests received by the OHCI-Lynx from that node will be accepted. |

### *asynchronous request filter low register*

This set/clear register is used to enable asynchronous receive requests on a per node basis, and handles the lower node IDs. Other than filtering different node IDs, this register behaves identical to the Asynchronous Request Filter High Register described in section 0.

**OHCI register 108h set register**
**OHCI register 10Ch clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 44. Bit Descriptions – Asynchronous Request Filter Low Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | asynReqResource31 | rsc | If set to one for local bus node number 31, asynchronous requests received by the OHCI-Lynx from that node will be accepted. |
| | | | Bits 30 through 2 follow the same pattern as below. |
| 1 | asynReqResource1 | rsc | If set to one for local bus node number 1, asynchronous requests received by the OHCI-Lynx from that node will be accepted. |
| 0 | asynReqResource0 | rsc | If set to one for local bus node number 0, asynchronous requests received by the OHCI-Lynx from that node will be accepted. |

### *physical request filter high register*

This set/clear register is used to enable physical receive requests on a per node basis, and handles the upper node IDs. When a packet is destined for the physical request context, and the node ID has been compared against the ARRQ registers, then the comparison is done again with this register. If the bit corresponding to the node ID is not set in this register, then the request will be handled by the ARRQ context instead of the physical request context.

**OHCI register 110h set register**
**OHCI register 114h clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 45. Bit Descriptions – Physical Request Filter High Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | physReqAllBusses | rsc | If set to one, all asynchronous requests received by the OHCI-Lynx from non-local bus nodes will be accepted. |
| 30 | physReqResource62 | rsc | If set to one for local bus node number 62 physical requests received by the OHCI-Lynx from that node will be handled through the physical request context. |
| | | | Bits 29 through 2 follow the same pattern as below. |
| 1 | physReqResource33 | rsc | If set to one for local bus node number 33 physical requests received by the OHCI-Lynx from that node will be handled through the physical request context. |
| 0 | physReqResource32 | rsc | If set to one for local bus node number 32 physical requests received by the OHCI-Lynx from that node will be handled through the physical request context. |

### *physical request filter low register*

This set/clear register is used to enable physical receive requests on a per node basis, and handles the lower node IDs. When a packet is destined for the physical request context, and the node ID has been compared against the asynchronous request filter registers, then the node ID comparison is done again with this register. If the bit corresponding to the node ID is not set in this register, then the request will be handled by the asynchronous request context instead of the physical request context.

**OHCI register 118h set register**
**OHCI register 11Ch clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Table 46. Bit Descriptions – Physical Request Filter Low Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | physReqResource31 | rsc | If set to one for local bus node number 31 physical requests received by the OHCI-Lynx from that node will be handled through the physical request context. |
| 30 | physReqResource30 | rsc | If set to one for local bus node number 30 physical requests received by the OHCI-Lynx from that node will be handled through the physical request context. |
| | | | Bits 29 through 2 follow the same pattern as below. |
| 1 | physReqResource1 | rsc | If set to one for local bus node number 1 physical requests received by the OHCI-Lynx from that node will be handled through the physical request context. |
| 0 | physReqResource0 | rsc | If set to one for local bus node number 0 physical requests received by the OHCI-Lynx from that node will be handled through the physical request context. |

### *physical upper bound register (optional register)*

This register is an optional register and is not implemented. This register is read only and returns all zeros.

**OHCI register 120h**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

### *asynchronous context control register*

This set/clear register controls the state and indicates status of the DMA context.

**OHCI register 180h set register [ATRQ]**
**OHCI register 184h clear register [ATRQ]**
**OHCI register 1A0h set register [ATRS]**
**OHCI register 1A4h clear register [ATRS]**
**OHCI register 1C0h set register [ARRQ]**
**OHCI register 1C4h clear register [ARRQ]**
**OHCI register 1E0h set register [ATRS]**
**OHCI register 1E4h clear register [ATRS]**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | x | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |

**Table 47. Bit Descriptions – Asynchronous Context Control Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 15 | run | rscu | This bit is set by software to enable descriptor processing for the context and cleared by software to stop descriptor processing. The OHCI-Lynx will only change this bit on a hardware or software reset. |
| 12 | wake | rsu | Software sets this bit to cause the OHCI-Lynx to continue or resume descriptor processing. The OHCI-Lynx will clear this bit on every descriptor fetch. |
| 11 | dead | ru | The OHCI-Lynx sets this bit when it encounters a fatal error, and clears the bit when software resets the run bit. |
| 10 | active | ru | The OHCI-Lynx sets this bit to one when it is processing descriptors. |
| 7:5 | spd | ru | This field indicates the speed at which a packet was received or transmitted, and only contains meaningful information for receive contexts. This field is encoded as:<br>000b = 100 Mbits/sec,<br>001b = 200 Mbits/sec,<br>010b = 400 Mbits/sec, and all other values are reserved. |
| 4:0 | eventcode | ru | This field holds the acknowledge sent by the Link core for this packet, or an internally generated error code if the packet was not transferred successfully. |

### *asynchronous context command pointer register*

This register contains a pointer to the address of the first descriptor block that the OHCI-Lynx will access when software enables the context by setting the ContextControl.run bit.

**OHCI register 19Ch [ATRQ]**
**OHCI register 1ACh [ATRS]**

**OHCI register 1CCh [ATRQ]**
**OHCI register 1ECh [ATRS]**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

**Table 48. Bit Descriptions – Asynchronous Context Command Pointer Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31:4 | descriptorAddress | rwu | Contains the upper 28 bits of the address of a 16-byte aligned descriptor block. |
| 3:0 | Z | rwu | Indicates the number of contiguous descriptors at the address pointed to by the descriptor address. If Z is 0, it indicates that the descriptorAddress is not valid. |

### *isochronous transmit context control register*

This set/clear register controls options, state, and status for the isochronous transmit DMA contexts. The n value in the following register addresses indicates the context number (n=0,1,2,3,…).

**OHCI register 200h + (16 * n) set register**
**OHCI register 204h + (16 * n) clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | x | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |

**Table 49. Bit Descriptions – Isochronous Transmit Context Control Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | cycleMatchEnable | rscu | When set to one, processing will occur such that the packet described by the context's first descriptor block will be transmitted in the cycle whose number is specified in the cycleMatch field of this register. The 13-bit cycleMatch field must match the 13-bit cycleCount field in the cycle start packet that is sent or received immediately before isochronous transmission begins. |
| 30:16 | cycleMatch | rsc | Contains a 15-bit value, corresponding to the lower order 2 bits of cycleSeconds and 13-bit cycleCount field. If cycleMatchEnable is set, then this IT DMA context will become enabled for transmits when the bus cycleCount value equals the cycleMatch value. |
| 15 | run | rsc | This bit is set by software to enable descriptor processing for the context and cleared by software to stop descriptor processing. The OHCI-Lynx will only change this bit on a hardware or software reset. |
| 12 | wake | rsu | Software sets this bit to cause the OHCI-Lynx to continue or resume descriptor processing. The OHCI-Lynx will clear this bit on every descriptor fetch. |
| 11 | dead | ru | The OHCI-Lynx sets this bit when it encounters a fatal error, and clears the bit when software resets the run bit. |
| 10 | active | ru | The OHCI-Lynx sets this bit to one when it is processing descriptors. |
| 7:5 | spd | ru | This field in not meaningful for isochronous transmit contexts. |
| 4:0 | event code | ru | Following an OUTPUT_LAST* command, the error code is indicated in this field. Possible values are: ack_complete, evt_descriptor_read, evt_data_read, and evt_unknown. |

### *isochronous transmit context command pointer register*

This register contains a pointer to the address of the first descriptor block that the OHCI-Lynx will access when software enables an ISO transmit context by setting the ContextControl.run bit. The n value in the following register addresses indicates the context number (n=0,1,2,3,…). Refer to section 0 details.

**OHCI register 20Ch + (16 * n)**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | X | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

### *ischronous receive context control register*

This set/clear register controls options, state, and status for the isochronous receive DMA contexts. The n value in the following register addresses indicates the context number (n=0,1,2,3,…).

**OHCI register 400h + (32 * n)** **set register**
**OHCI register 404h + (32 * n)** **clear register**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | 0 | 0 | 0 | x | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |

**Table 50. Bit Descriptions – Isochronous Receive Context Control**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | bufferFill | rsc | When set, received packets are placed back-to-back to completely fill each receive buffer. When clear, each received packet is placed in a single buffer. If the multiChanMode bit is set to one, this bit must also be set to one. The value of bufferFill must not be changed while active or run are set. |
| 30 | isochHeader | rsc | When set to one, received isochronous packets will include the complete 4-byte isochronous packet header seen by the link layer. The end of the packet will be marked with a xferStatus in the first doublet, and a 16-bit timeStamp indicating the time of the most recently received (or sent) cycleStart packet. When clear, the packet header is stripped off of received isochronous packets. The packet header, if received, immediately precedes the packet payload. The value of isochHeader must not be changed while active or run are set. |
| 29 | cycleMatchEnable | rscu | When set, the context will begin running only when the 13-bit cycleMatch field in the contextMatch register matches the 13-bit cycleCount in the cycleStart packet. The effects of this bit however are impacted by the values of other bits in this register. Once the context has become active, hardware clears the cycleMatchEnable bit. The value of cycleMatchEnable must not be changed while active or run are set. |
| 28 | multiChanMode | rsc | When set, the corresponding isochronous receive DMA context will receive packets for all isochronous channels enabled in the IRChannelMaskHi and IRChannelMaskLo registers. The isochronous channel number specified in the IRDMA context match register is ignored. When zero, the IRDMA context will receive packets for that single channel. Only one IRDMA context may use the IRChannelMask registers. If more that one IRDMA context control register has the multiChanMode bit set, results are undefined. The value of multiChanMode must not be changed while active or run are set. |
| 15 | run | rscu | This bit is set by software to enable descriptor processing for the context and cleared by software to stop descriptor processing. The OHCI-Lynx will only change this bit on a hardware or software reset. |
| 12 | wake | rsu | Software sets this bit to cause the OHCI-Lynx to continue or resume descriptor processing. The OHCI-Lynx will clear this bit on every descriptor fetch. |
| 11 | dead | ru | The OHCI-Lynx sets this bit when it encounters a fatal error, and clears the bit when software resets the run bit. |
| 10 | active | ru | The OHCI-Lynx sets this bit to one when it is processing descriptors. |
| 7:5 | spd | ru | This field indicates the speed at which the packet was received.<br>3'b000 = 100 Mbits/sec,<br>3'b001 = 200 Mbits/sec, and<br>3'b010 = 400 Mbits/sec. All other values are reserved. |
| 4:0 | event code | ru | Following and INPUT* command, the error code is indicated in this field. |

### *isochronous receive context command pointer register*

This register contains a pointer to the address of the first descriptor block that the OHCI-Lynx will access when software enables an ISO receive context by setting the ContextControl.run bit. The n value in the following register addresses indicates the context number (n=0,1,2,3,…). Refer to section 0 for details.

**OHCI register 40Ch + (32 * n)**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

***isochronous receive context match register***

This register is used to start an isochronous receive context running on a specified cycle number, to filter incoming isochronous packets based on tag values, and to wait for packets with a specified sync value. The n value in the following register addresses indicates the context number (n=0,1,2,3,…).

**OHCI register 410Ch + (32 * n)**

| BIT NUMBER | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | 0 | 0 | 0 | x | x | x | x | x | x | x | x | x |

| BIT NUMBER | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reset State | x | x | x | x | x | x | x | x | 0 | x | x | x | x | x | x | x |

**Table 51. Bit Descriptions – Isochronous Receive Context Match Register**

| BIT | FIELD NAME | ACCESS | DESCRIPTION |
|---|---|---|---|
| 31 | tag3 | rw | If set, this context will match on iso receive packets with a tag field of 2’b11. |
| 30 | tag2 | rw | If set, this context will match on iso receive packets with a tag field of 2’b10. |
| 29 | tag1 | rw | If set, this context will match on iso receive packets with a tag field of 2’b01. |
| 28 | tag0 | rw | If set, this context will match on iso receive packets with a tag field of 2’b00. |
| 24:12 | cycleMatch | rw | Contains a 13-bit value, corresponding to the 13-bit cycleCount field in the cycleStart packet. If cycleMatchEnable is set, then this context is enabled for receives when the bus cycleCount value equals the cycleMatch value. |
| 11:8 | sync | rw | This field contains the 4-bit field which is compared to the sync field of each iso packet for this channel when the command descriptor’s w field is set to 2’b11. |
| 6 | tag1SyncFilter | rw | If set and the tag1 bit is set , then packets with tag2b01 shall be accepted into the context if the two most significant bits of the packets sync field are 2’b00. Packets with tag values other than 2’b01 shall be filtered according to tag0, tag2 and tag3 without any additional restrictions.<br>If clear, this context will match on isochronous receive packets as specified in the tag0–3 bits with no additional restrictions. |
| 5:0 | channelNumber | rw | This 6-bit field indicates the isochronous channel number for which this IR DMA context will accept packets. |

## GPIO interface

The GPIO interface consists of four general purpose input output ports. On power reset, GPIO0 and GPIO1 are enabled and are configured as Bus Manager Contender (BMC) and Link Power Status (LPS) , respectively. BMC and LPS outputs can be configured via the GPIO control register in PCI Configuration space, as GPIO0 and GPIO1. Figure 3 shows the schematic for GPIO0 and GPIO implementation.

GPIO2 and GPIO3 power up as general purpose inputs and are programmable via the GPIO control register. Figure 4 shows the schematic for GPIO2 and GPIO3 implementation.

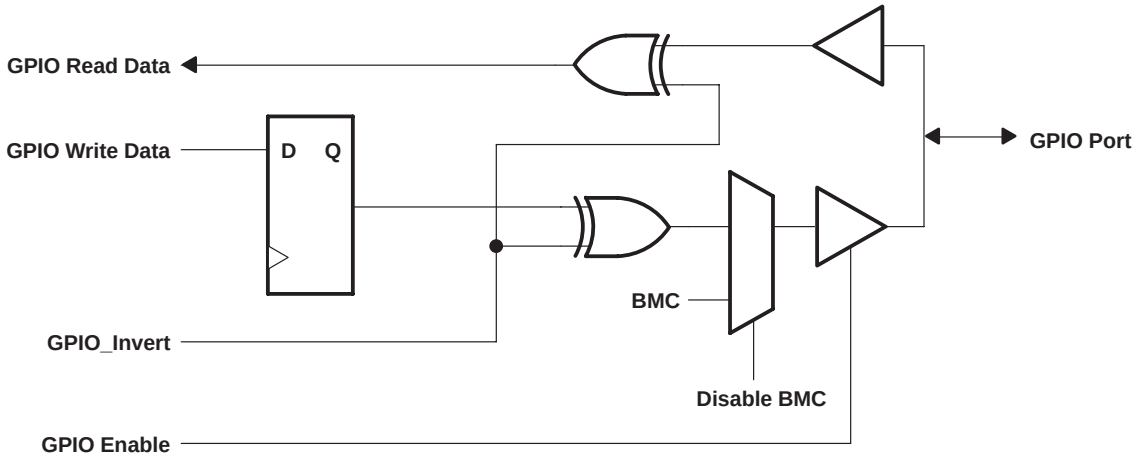



**Figure 3. BMC/LINKON/GPIO0**

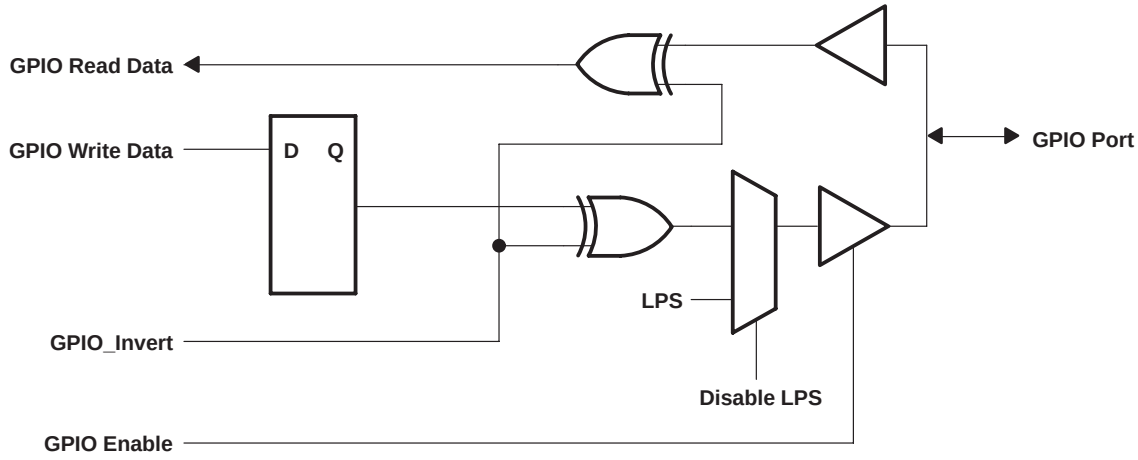



**Figure 4. LPS/GPIO1**

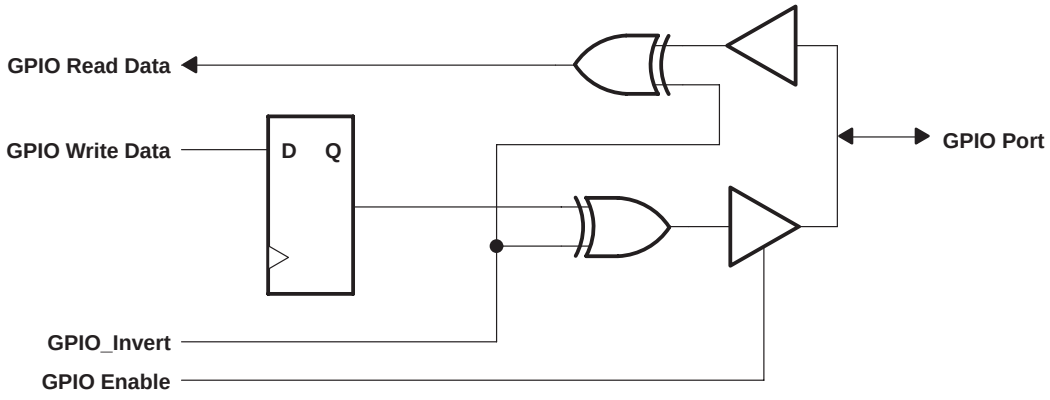



**Figure 5. GPIO2 and GPIO3**

## serial EEPROM

### serial bus interface

The OHCI-Lynx provides a serial bus interface to initiallize the 1394 Global Unique ID register and a few PCI configuration registers through a serial EEPROM. The OHCI-Lynx communicates with the serial EEPROM via the 2-wire serial interface.

After power-up the serial interface initializes the locations listed in Table 52. While the OHCI-Lynx is accessing the serial ROM , all incoming PCI Slave accesses are terminated with retry status. Table 53. shows the serial ROM memory map required for initializing the OHCI-Lynx registers.

**Table 52. Bit Descriptions – Registers and Bits Loadable through Serial EEPROM**

| OFFSET | REGISTER | BITS LOADED FROM EEPROM |
|---|---|---|
| OHCI Register (24h) | 1394 GlobalUniqueIDHi | 31–0 |
| OHCI Register(28h) | 1394 GlobalUniqueIDLo | 31–0 |
| OHCI Register (50h) | Host Control Register | 23 |
| PCI Register (2Ch) | PCI Subsytem ID | 15–0 |
| PCI Register (2Dh) | PCI Vendor ID | 15–0 |
| PCI Register (3Eh) | PCI Maximum Latency, PCI Minimum Grant | 15–0 |
| PCI Register (F4h) | Link Enhancements Control Register | 7, 2, 1 |

**Table 53. Bit Descriptions – Serial EEPROM Map**

| BYTE ADDRESS | BYTE DESCRIPTION | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 00 | PCI Maximum Latency (4'h0) | | | | PCI_Minimum Grant (4'h0) | | | |
| 01 | PCI Vendor ID | | | | | | | |
| 02 | PCI Vendor ID (msbyte) | | | | | | | |
| 03 | PCI Subsytem ID (ls byte) | | | | | | | |
| 04 | PCI Subsytem ID | | | | | | | |
| 05 | [7] Link_enhancement-Control.enab_unfair | [6] HCControl. ProgramPhy Enable | [5] RSVD | [4] RSVD | [3] RSVD | [2] Link_enhancement-Control.enab_ insert_idle | [1] Link_enhancement-Control.enab_accel | [0] RSVD |
| 06 | Reserved | | | | | | | |
| 07 | 1394 GlobalUniqueIDHi (lsbyte0) | | | | | | | |
| 08 | 1394 GlobalUniqueIDHi (byte 1) | | | | | | | |
| 09 | 1394 GlobalUniqueIDHi (byte 2) | | | | | | | |
| 0A | 1394 GlobalUniqueIDHi (msbyte 3) | | | | | | | |
| 0B | 1394 GlobalUniqueIDLo (lsbyte0) | | | | | | | |
| 0C | 1394 GlobalUniqueIDLo (byte 1) | | | | | | | |
| 0D | 1394 GlobalUniqueIDLo (byte 2) | | | | | | | |
| 0E | 1394 GlobalUniqueIDLo (msbyte 3) | | | | | | | |

### absolute maximum ratings over operating free-air temperature (unless otherwise noted)†

Supply voltage range, $V_{CC}$ . . . . . –0.5 V to 3.6 V
Supply voltage range, $V_{CCP}$ . . . . . –0.5 V to 5.5 V
Supply voltage range, $V_{CCS}$ . . . . . –0.5 V to 5.5 V
Input voltage range for PCI, $V_I$ . . . . . –0.5 to $V_{CCP}$ + 0.5 V
Input voltage range for Miscellaneous and Phy I/F, $V_I$ . . . . . –0.5 to $V_{CCI}$ + 0.5 V
Output voltage range for PCI, $V_O$ . . . . . –0.5 to $V_{CCP}$ + 0.5 V
Input voltage range for Miscellaneous and Phy I/F, $V_O$ . . . . . –0.5 to $V_{CCP}$ + 0.5 V
Input clamp current, $I_{IK}$ ($V_I$ < 0 or $V_I$ > $V_{CC}$) (see Note 1) . . . . . ±20 mA
Output clamp current, $I_{OK}$ ($V_O$ < 0 or $V_O$ > $V_{CC}$) (see Note 2) . . . . . ±20 mA
Storage temperature range . . . . . –65°C to 150°C

† Stresses beyond those listed under absolute maximum ratings may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under recommended operating conditions is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 1. Applies to external input and bidirectional buffers. For 5-V tolerant use $V_I > V_{CCI}$. For PCI use $V_I > V_{CCP}$.
2. Applies to external output and bi–directional buffers. For 5-V tolerant use $V_O > V_{CCI}$. For PCI use VO > $V_{CCP}$.

### recommended operating conditions

| | | | OPERATION | MIN | NOM | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| $V_{CC}$ | Core voltage | Commercial | | 3 | 3.3 | 3.6 | V |
| $V_{CCP}$ | PCI I/O clamping voltage | Commercial | | 3 | | 5.5 | V |
| $V_{CCS}$ | All other I/O clamping rail | Commercial | | 3 | | 5.5 | V |
| $V_{IH}$† | High-level input voltage | PCI | | 0.475 $V_{CCP}$ | | $V_{CCP}$ | V |
| | | PHY I/F | | 2 | | $V_{CCS}$ | |
| | | Miscellaneous‡ | | 2 | | $V_{CCS}$ | |
| $V_{IL}$† | Low-level input voltage | PCI | | 0 | | 0.325 $V_{CCS}$ | V |
| | | PHY I/F | | 0 | | 0.8 | |
| | | Miscellaneous‡ | | 0 | | 0.8 | |
| $V_I$ | Input voltage | PCI | | 0 | | $V_{CCP}$ | V |
| | | PHY I/F | | 0 | | $V_{CCS}$ | |
| | | Miscellaneous‡ | | 0 | | $V_{CCS}$ | |
| $V_O$§ | Output voltage | PCI | | 0 | | $V_{CCP}$ | V |
| | | PHY I/F | | 0 | | $V_{CCS}$ | |
| | | Miscellaneous‡ | | 0 | | $V_{CCS}$ | |
| $t_t$ | Input transition time ($t_r$ and $t_f$) | PCI | | 0 | | 6 | ns |
| $T_A$ | Operating ambient temperature | | | 0 | 25 | 70 | °C |
| $T_J$¶ | Virtual junction temperature | | | 0 | 25 | 115 | °C |

† Applies for external inputs and bidirectional buffers without hysteresis.
‡ Miscellaneous pins are: GPIO(0:3), SDA, SCL, CYCLEOUT.
§ Applies for external output buffers.
¶ The junction temperatures reflect simulation conditions. Customer is responsible for verifying junction temperature.

## electrical characteristics over recommended operating conditions

| | | | OPERATION | TEST CONDITIONS | MIN | MAX | UNIT |
|---|---|---|---|---|---|---|---|
| $V_{OH}$ | High-level output voltage | PCI | | $I_{OH}$ = – 0.5 mA | 0.9 $V_{CCP}$ | | V |
| | | | | $I_{OH}$ = – 2 mA | 2.4 | | |
| | | PHY I/F | | $I_{OH}$ = – 4 µA | 2.8 | | |
| | | | | $I_{OH}$ = – 8 mA | $V_{CC}$ – 0.6 | | |
| | | Miscellaneous‡ | | $I_{OH}$ = – 4 mA | $V_{CC}$ – 0.6 | | |
| $V_{OL}$† | Low-level output voltage | PCI | | $I_{OL}$ = 1.5 mA | | 0.1 $V_{CC}$ | V |
| | | | | $I_{OL}$ = 6 mA | 0 | 0.55 | |
| | | PHY I/F | | $I_{OL}$ = 4 mA | | 0.4 | |
| | | | | $I_{OL}$ = 8 mA | | | |
| | | Miscellaneous‡ | | $I_{OL}$ = 4 mA | | 0.5 | |
| $I_{OZ}$ | 3-state output high-impedance | Output pins | 3.6 V | $V_O$ = $V_{CC}$ or GND | | ±20 | µA |
| $I_{IL}$ | Low-level input current | Input pins | 3.6 V | $V_I$ = GND‡ | | ±20 | µA |
| | | I/O pins† | 3.6 V | $V_I$ = GND‡ | ±20 | | |
| $I_{IH}$ | High-level input current | PCI† | 3.6 V | $V_I$ = $V_{CC}$‡ | | ±20 | µA |
| | | OTHERS† | 3.6 V | $V_I$ = $V_{CC}$‡ | ±20 | | |

† For I/O pins, input leakage ($I_{IL}$ and $I_{IH}$) includes $I_{OZ}$ of the disabled output.
‡ Miscellaneous pins are: GPIO(0:3), SDA, SCL, CYCLEOUT.

## switching characteristics for PCI interface§

| PARAMETER | | MEASURED | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|
| $t_{su}$ | Setup time before PCLK | 50% to 50% | 7 | | | ns |
| $t_h$ | Hold time before PCLK | 50% to 50% | 0 | | | ns |
| $t_d$ | Delay time, PHI_CLK to data valid | 50% to 50% | 2 | | 11 | ns |

§ These parameters are ensured by design.

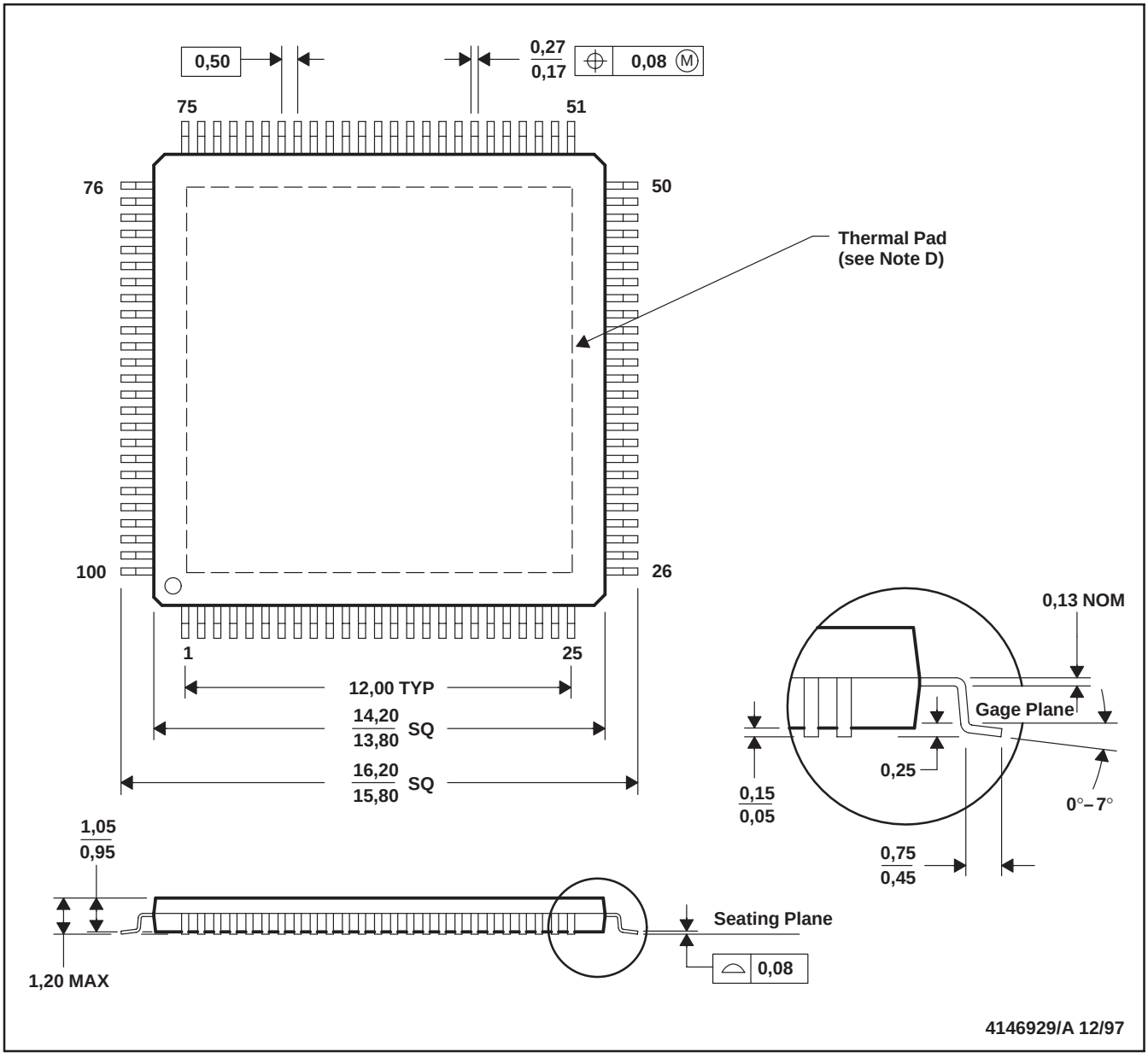
## switching characteristics for PHY-Link interface§

| PARAMETER | | MEASURED | MIN | TYP | MAX | UNIT |
|---|---|---|---|---|---|---|
| $t_{su}$ | Setup time, Dn, CTLn, LREQ to PHY_CLK | 50% to 50% | 6 | | | ns |
| $t_h$ | Hold time, Dn, CTLn, LREQ before PHY_CLK | 50% to 50% | 1 | | | ns |
| $t_d$ | Delay time, PHY_CLK to Dn, CTLn | 50% to 50% | 2 | | 11 | ns |

§ These parameters are ensured by design.

## MECHANICAL DATA

**PZP (S-PQFP-G100)** **PowerPAD™ PLASTIC QUAD FLATPACK**




NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Body dimensions include mold flash or protrusions.
D. The package thermal performance may be enhanced by bonding the thermal pad to an external thermal plane. This pad is electrically and thermally connected to the backside of the die and possibly selected leads.
E. Falls within JEDEC MS-026

PowerPAD is a trademark of Texas Instruments Incorporated.

## IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.