

## FW322 07 T100

### 1394a PCI PHY/Link Open Host Controller



## 1 Features

- 1394a-2000 OHCI link and PHY core function in a single device:
  - 100-pin TQFP package (also available in a lead-free package; see ordering information on page 86.)
  - Single-chip link and PHY enable smaller, simpler, more efficient motherboard and add-in card designs
  - Enables lower system costs
  - Leverages proven 1394a-2000 PHY core design
  - Demonstrated compatibility with current *Microsoft Windows*® drivers and common applications
  - Demonstrated interoperability with existing, as well as older, 1394 consumer electronics and peripherals products
  - Feature-rich implementation for high performance in common applications
  - Supports low-power system designs (CMOS implementation, power management features)
- OHCI:
  - Complies with the *1394 OHCI 1.1 Specification*
  - OHCI 1.0 backwards compatible—configurable via EEPROM to operate in either OHCI 1.0 or OHCI 1.1 mode
  - Complies with *Microsoft Windows* logo program system and device requirements
  - Listed on *Windows* hardware compatibility list <http://testedproducts.windowsmarketplace.com/>.
  - Compatible with *Microsoft Windows* and *MacOS*® operating systems
  - 4 Kbyte isochronous transmit FIFO
  - 2 Kbyte asynchronous transmit FIFO
  - 4 Kbyte isochronous receive FIFO
  - 2 Kbyte asynchronous receive FIFO
  - Dedicated asynchronous and isochronous descriptor-based DMA engines
  - Eight isochronous transmit contexts
  - Eight isochronous receive contexts
  - Prefetches isochronous transmit data
  - Supports posted write transactions
  - Supports parallel processing of incoming physical read and write requests
  - Supports notification (via interrupt) of a failed register access
- 1394a-2000 PHY core:
  - Compliant with *IEEE*® 1394a-2000, *Standard for a High Performance Serial Bus* (Supplement)
  - Provides two fully compliant cable ports, each supporting 400 Mb/s, 200 Mb/s, and 100 Mb/s traffic
    - Supports extended BIAS\_HANDSHAKE time for enhanced interoperability with camcorders
    - While unpowered and connected to the bus, will not drive TPBIAS on a connected port even if receiving incoming bias voltage on that port
    - Does not require external filter capacitor for PLL
    - Supports link-on as a part of the internal PHY core-link interface
    - 25 MHz crystal oscillator and internal PLL provide a 50 MHz internal link-layer controller clock as well as transmit/receive data at 100 Mb/s, 200 Mb/s, and 400 Mb/s
    - Interoperable across 1394 cable with 1394 physical layers (PHY core) using 5 V supplies
    - Provides node power-class information signaling for system power management
    - Supports ack-accelerated arbitration and fly-by concatenation
    - Supports arbitrated short bus reset to improve utilization of the bus
    - Fully supports suspend/resume
    - Supports connection debounce
    - Supports multispeed packet concatenation
    - Supports PHY pinging and remote PHY access packets
    - Reports cable power fail interrupt when voltage at CPS pin falls below 7.5 V
    - Provides separate cable bias and driver termination voltage supply for each port
- Link:
  - Cycle master and isochronous resource manager capable
  - Supports 1394a-2000 acceleration features
- PCI:
  - Revision 2.3 compliant
  - 33 MHz/32-bit operation
  - Programmable burst size thresholds for PCI data transfer
  - Supports optimized memory read line, memory read multiple, and memory write invalidate burst commands
  - Supports *PCI Bus Power Management Interface Specification* v.1.1

**Note:** This device does not support D3cold wakeup, CLKRUN protocol, *mini PCI*® applications, and CardBus applications. Use the FW322 06 120-pin TQFP device if one or more of these features are needed.

## Table of Contents

| Contents   | Page |
|--|------|
| 1 Features .....   | 1    |
| 1.1 Other Features .....   | 6    |
| 2 FW322 Functional Overview .....                                    | 6    |
| 3 FW322 Functional Description .....                                 | 7    |
| 3.1 PCI Core .....   | 7    |
| 3.2 OHCI Data Transfer .....   | 8    |
| 3.3 OHCI Isochronous Data Transfer .....                             | 8    |
| 3.4 Isochronous Register Access .....                                | 9    |
| 3.5 OHCI Asynchronous Data Transfer .....                            | 9    |
| 3.6 Asynchronous Register Access .....                               | 9    |
| 3.7 Link Core .....  | 11   |
| 3.8 PHY Core .....   | 13   |
| 4 Pin Information .....  | 15   |
| 5 Internal Registers .....   | 21   |
| 5.1 PCI Configuration Registers .....                                | 21   |
| 5.2 Vendor ID Register .....   | 22   |
| 5.3 Device ID Register .....   | 22   |
| 5.4 PCI Command Register .....                                       | 23   |
| 5.5 PCI Status Register .....  | 24   |
| 5.6 Class Code and Revision ID Registers .....                       | 25   |
| 5.7 Latency Timer and Cache Line Size Register .....                 | 25   |
| 5.8 Header Type and BIST Register .....                              | 26   |
| 5.9 OHCI Base Address Register .....                                 | 26   |
| 5.10 PCI Subsystem Identification Register .....                     | 27   |
| 5.11 PCI Power Management Capabilities Pointer Register .....        | 27   |
| 5.12 Interrupt Line and Pin Register .....                           | 28   |
| 5.13 MIN_GNT and MAX_LAT Register .....                              | 28   |
| 5.14 PCI OHCI Control Register .....                                 | 29   |
| 5.15 Capability ID and Next Item Pointer Register .....              | 29   |
| 5.16 Power Management Capabilities Register .....                    | 30   |
| 5.17 Power Management Control and Status Register .....              | 31   |
| 5.18 Power Management CSR PCI-to-PCI Bridge Support Extensions ..... | 32   |
| 5.19 Power Management Data .....                                     | 32   |
| 5.20 OHCI Registers .....  | 33   |
| 5.21 OHCI Version Register .....                                     | 36   |
| 5.22 GUID ROM Register .....   | 37   |
| 5.23 Asynchronous Transmit Retries Register .....                    | 37   |
| 5.24 CSR Data Register .....   | 38   |
| 5.25 CSR Compare Register .....                                      | 38   |
| 5.26 CSR Control Register .....                                      | 38   |
| 5.27 Configuration ROM Header Register .....                         | 39   |
| 5.28 Bus Identification Register .....                               | 40   |
| 5.29 Bus Options Register .....                                      | 40   |
| 5.30 GUID High Register .....  | 41   |
| 5.31 GUID Low Register .....   | 41   |
| 5.32 Configuration ROM Mapping Register .....                        | 42   |
| 5.33 Posted Write Address Low Register .....                         | 43   |
| 5.34 Posted Write Address High Register .....                        | 43   |
| 5.35 Vendor ID Register .....  | 43   |
| 5.36 Host Controller Control Register .....                          | 44   |
| 5.37 SelfID Buffer Pointer Register .....                            | 46   |

## Table of Contents (continued)

| Contents   | Page |
|--|------|
| 5.38 SelfID Count Register .....   | 46   |
| 5.39 Isochronous Receive Multiple Channel Mask High (IRMultiChanMaskHi) Register ..... | 47   |
| 5.40 Isochronous Receive Multiple Channel Mask Low (IRMultiChanMaskLo) Register .....  | 47   |
| 5.41 Interrupt Event (IntEvent) Register .....   | 48   |
| 5.42 Interrupt Mask (IntMask) Register .....   | 50   |
| 5.43 Isochronous Transmit Interrupt Event (isoXmitIntMask) Register .....              | 52   |
| 5.44 Isochronous Transmit Interrupt Mask (isoXmitIntMask) Register .....               | 53   |
| 5.45 Isochronous Receive Interrupt Event (isoRecvIntEvent) Register .....              | 54   |
| 5.46 Isochronous Receive Interrupt Mask (isoRecvIntMask) Register .....                | 55   |
| 5.47 Fairness Control Register .....   | 55   |
| 5.48 Link Control Register .....   | 56   |
| 5.49 Node Identification Register .....  | 57   |
| 5.50 PHY Core Layer Control Register .....   | 58   |
| 5.51 Isochronous Cycle Timer Register .....  | 58   |
| 5.52 Asynchronous Request Filter High Register .....                                   | 59   |
| 5.53 Asynchronous Request Filter Low Register .....                                    | 59   |
| 5.54 Physical Request Filter High Register .....                                       | 60   |
| 5.55 Physical Request Filter Low Register .....  | 60   |
| 5.56 Asynchronous Context Control Register .....                                       | 61   |
| 5.57 Asynchronous Context Command Pointer Register .....                               | 62   |
| 5.58 Isochronous Transmit Context Control (IT DMA ContextControl) Register .....       | 63   |
| 5.59 Isochronous Transmit Context Command Pointer Register .....                       | 64   |
| 5.60 Isochronous Receive Context Control (IR DMA ContextControl) Register .....        | 65   |
| 5.61 Isochronous Receive Context Command Pointer Register .....                        | 66   |
| 5.62 Isochronous Receive Context Match (IR DMA ContextMatch) Register .....            | 67   |
| 5.63 FW322 Vendor-Specific Registers .....   | 68   |
| 5.64 Isochronous DMA Control .....   | 68   |
| 5.65 Asynchronous DMA Control .....  | 69   |
| 5.66 Link Options .....  | 70   |
| 6 Internal Register Configuration .....  | 71   |
| 6.1 PHY Core Register Map .....  | 71   |
| 6.2 PHY Core Register Fields .....   | 72   |
| 7 Crystal Selection Considerations .....   | 77   |
| 7.1 Load Capacitance .....   | 77   |
| 7.2 Adjustment to Crystal Loading .....  | 77   |
| 7.3 Crystal/Board Layout .....   | 77   |
| 8 Serial EEPROM Interface .....  | 78   |
| 9 ac Characteristics of Serial EEPROM Interface Signals .....                          | 78   |
| 10 Solder Reflow and Handling .....  | 81   |
| 11 Absolute Maximum Voltage/Temperature Ratings .....                                  | 81   |
| 12 Electrical Characteristics .....  | 82   |
| 13 Timing Characteristics .....  | 84   |
| 14 Outline Diagrams .....  | 85   |
| 14.1 100-Pin TQFP .....  | 85   |
| 15 Ordering Information .....  | 86   |

## Table of Contents (continued)

| Figure   | Page |
|--|------|
| Figure 1. FW322 Conceptual Block Diagram.....        | 6    |
| Figure 2. PCI Core Block Diagram.....                | 7    |
| Figure 3. OHCI Core Block Diagram.....               | 8    |
| Figure 4. Link Core Block Diagram.....               | 11   |
| Figure 5. The PHY Core Block Diagram.....            | 12   |
| Figure 6. Pin Assignments for the FW322 07 T100..... | 15   |
| Figure 7. Crystal Circuitry.....                     | 77   |
| Figure 8. Bus Timing.....                            | 79   |
| Figure 9. Write Cycle Timing.....                    | 79   |
| Figure 10. Data Validity.....                        | 79   |
| Figure 11. Start and Stop Definition.....            | 80   |
| Figure 12. Output Acknowledge.....                   | 80   |

| Table  | Page |
|--|------|
| Table 1. Pin Descriptions.....   | 16   |
| Table 2. Bit-Field Access Tag Description.....                             | 21   |
| Table 3. PCI Configuration Register Map.....                               | 21   |
| Table 4. PCI Command Register Description.....                             | 23   |
| Table 5. PCI Status Register.....  | 24   |
| Table 6. Class Code and Revision ID Register Description.....              | 25   |
| Table 7. Latency Timer and Class Cache Line Size Register Description..... | 25   |
| Table 8. Header Type and BIST Register Description.....                    | 26   |
| Table 9. OHCI Base Address Register Description.....                       | 26   |
| Table 10. PCI Subsystem Identification Register Description.....           | 27   |
| Table 11. Interrupt Line and Pin Register Description.....                 | 28   |
| Table 12. MIN_GNT and MAX_LAT Register Description.....                    | 28   |
| Table 13. PCI OHCI Control Register Description.....                       | 29   |
| Table 14. Capability ID and Next Item Pointer Register Description.....    | 29   |
| Table 15. Power Management Capabilities Register Description.....          | 30   |
| Table 16. Power Management Control and Status Register Description.....    | 31   |
| Table 17. Power Management Data Register Description.....                  | 32   |
| Table 18. OHCI Register Map.....   | 33   |
| Table 19. OHCI Version Register Description.....                           | 36   |
| Table 20. GUID ROM Register Description.....                               | 37   |
| Table 21. Asynchronous Transmit Retries Register Description.....          | 37   |
| Table 22. CSR Data Register Description.....                               | 38   |
| Table 23. CSR Compare Register Description.....                            | 38   |
| Table 24. CSR Control Register Description.....                            | 38   |
| Table 25. Configuration ROM Header Register Description.....               | 39   |
| Table 26. Bus Identification Register Description.....                     | 40   |
| Table 27. Bus Options Register Description.....                            | 40   |
| Table 28. GUID High Register Description.....                              | 41   |
| Table 29. GUID Low Register Description.....                               | 41   |
| Table 30. Configuration ROM Mapping Register Description.....              | 42   |
| Table 31. Posted Write Address Low Register Description.....               | 43   |
| Table 32. Posted Write Address High Register Description.....              | 43   |
| Table 33. Vendor ID Register Description.....                              | 43   |
| Table 34. Host Controller Control Register Description.....                | 44   |
| Table 35. SelfID Buffer Pointer Register Description.....                  | 46   |

## Table of Contents (continued)

| Table   | Page |
|---|------|
| Table 36. SelfID Count Register Description .....                                 | 46   |
| Table 37. Isochronous Receive Channel Mask High Register Description .....        | 47   |
| Table 38. Isochronous Receive Channel Mask Low Register Description .....         | 47   |
| Table 39. Interrupt Event Register Description .....                              | 48   |
| Table 40. Interrupt Mask Register Description .....                               | 50   |
| Table 41. Isochronous Transmit Interrupt Event Register Description .....         | 52   |
| Table 42. Isochronous Transmit Interrupt Event Description .....                  | 53   |
| Table 43. Isochronous Receive Interrupt Event Description .....                   | 54   |
| Table 44. Fairness Control Register Description .....                             | 55   |
| Table 45. Link Control Register Description .....                                 | 56   |
| Table 46. Node Identification Register Description .....                          | 57   |
| Table 47. PHY Core Layer Control Register Description .....                       | 58   |
| Table 48. Isochronous Cycle Timer Register Description .....                      | 58   |
| Table 49. Asynchronous Request Filter High Register Description .....             | 59   |
| Table 50. Asynchronous Request Filter Low Register Description .....              | 59   |
| Table 51. Physical Request Filter High Register Description .....                 | 60   |
| Table 52. Physical Request Filter Low Register Description .....                  | 60   |
| Table 53. Asynchronous Context Control Register Description .....                 | 61   |
| Table 54. Asynchronous Context Command Pointer Register Description .....         | 62   |
| Table 55. Isochronous Transmit Context Control Register Description .....         | 63   |
| Table 56. Isochronous Transmit Context Command Pointer Register Description ..... | 64   |
| Table 57. Isochronous Receive Context Control Register Description .....          | 65   |
| Table 58. Isochronous Receive Context Command Pointer Register Description .....  | 66   |
| Table 59. Isochronous Receive Context Match Register Description .....            | 67   |
| Table 60. FW322 Vendor-Specific Registers Description .....                       | 68   |
| Table 61. Isochronous DMA Control Registers Description .....                     | 68   |
| Table 62. Asynchronous DMA Control Registers Description .....                    | 69   |
| Table 63. Link Options Register Description .....                                 | 70   |
| Table 64. PHY Core Register Map .....   | 71   |
| Table 65. PHY Core Register Fields .....  | 72   |
| Table 66. PHY Core Register Page 0: Port Status Page .....                        | 74   |
| Table 67. PHY Core Register Port Status Page Fields .....                         | 75   |
| Table 68. PHY Core Register Page 1: Vendor Identification Page .....              | 76   |
| Table 69. PHY Core Register Vendor Identification Page Fields .....               | 76   |
| Table 70. ac Characteristics of Serial EEPROM Interface Signals .....             | 78   |
| Table 71. Absolute Maximum Ratings .....  | 81   |
| Table 72. Analog Characteristics .....  | 82   |
| Table 73. Driver Characteristics .....  | 83   |
| Table 74. Device Characteristics .....  | 83   |
| Table 75. Switching Characteristics .....   | 84   |
| Table 76. Clock Characteristics .....   | 84   |

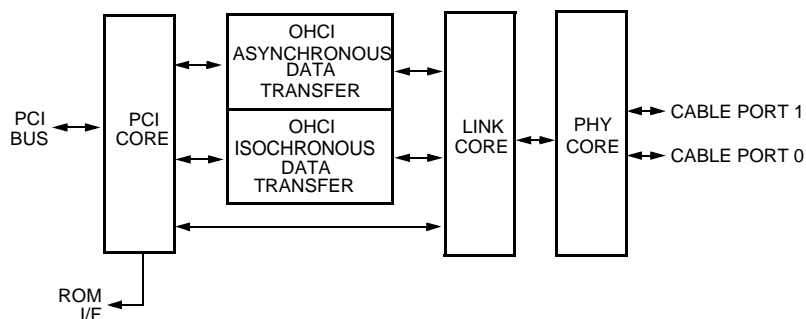
## 1 Features (continued)

### 1.1 Other Features

- I<sup>2</sup>C serial ROM interface
- CMOS process
- 3.3 V operation, 5 V tolerant inputs
- 100-pin TQFP package
- May be used without an EEPROM when the system BIOS is programmed with the EEPROM contents.

## 2 FW322 Functional Overview

The FW322 is a high-performance, PCI bus-based open host controller designed by Agere Systems for implementation of *IEEE* 1394a-2000 compliant systems and devices. Link-layer functions are handled by the FW322, utilizing the on-chip 1394a-2000 compliant link core and physical layer core. A high-performance and cost-effective solution for connecting and servicing multiple *IEEE* 1394 (both 1394-1995 and 1394a-2000) peripheral devices can be realized using this PHY/link OHCI device.



5-6250 (F).f

Figure 1. FW322 Conceptual Block Diagram

### 3 FW322 Functional Description

The FW322 is comprised of four major functional sections (see Figure 1): PCI core, OHCI isochronous and asynchronous data transfer, link core, and PHY core. The following is a general description of each of the major sections.

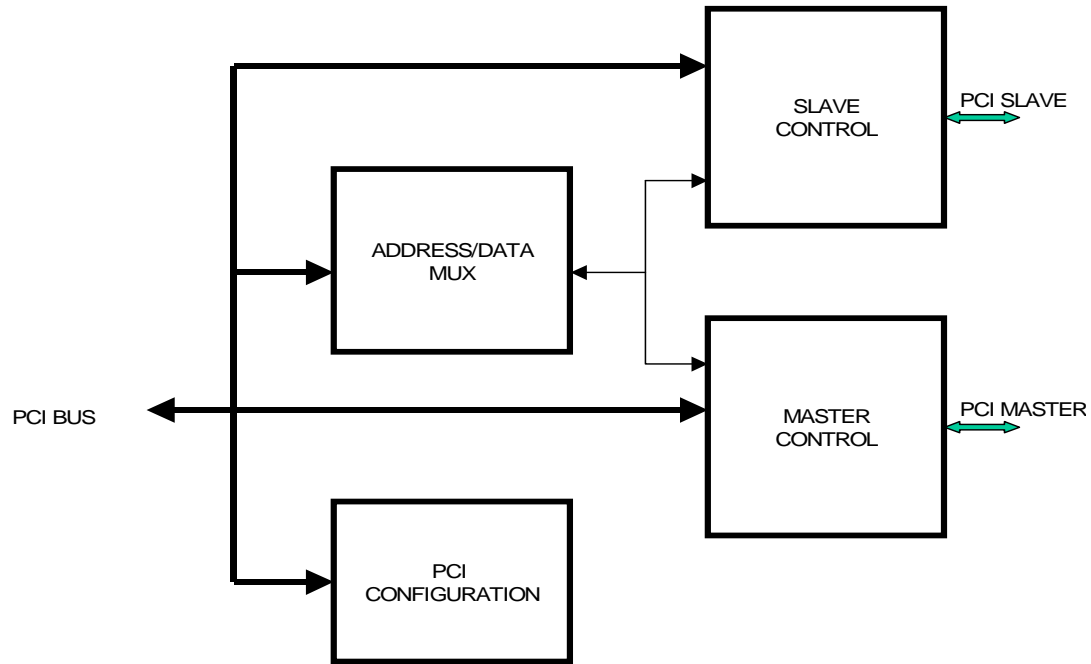


Figure 2. PCI Core Block Diagram

#### 3.1 PCI Core

The PCI core (shown in Figure 2) serves as the interface to the PCI bus. It contains the state machines that allow the FW322 to respond properly when it is the target of the transaction. Also, during 1394 packet transmission or reception, the PCI core arbitrates for the PCI bus and enables the FW322 to become the bus master for reading the different buffer descriptors and management of the actual data transfers to/from host system memory.

The PCI core also supports the *PCI Bus Power Management Interface Specification v.1.1*. Included in this support is a standard power management register interface accessible through the PCI configuration space. Through this register interface, software is able to transition the FW322 into four distinct power consumption states (D0, D1, D2, and D3hot). This permits software to selectively increase/decrease the power consumption of the FW322 for reasons such as periods of system inactivity or power conservation. In addition, the FW322 also includes support for waking up the system through the generation of a power management event (PME).

The FW322 supports generation of a power management event (PME) while in the D0, D1, D2, and D3hot power states.

### 3 FW322 Functional Description (continued)

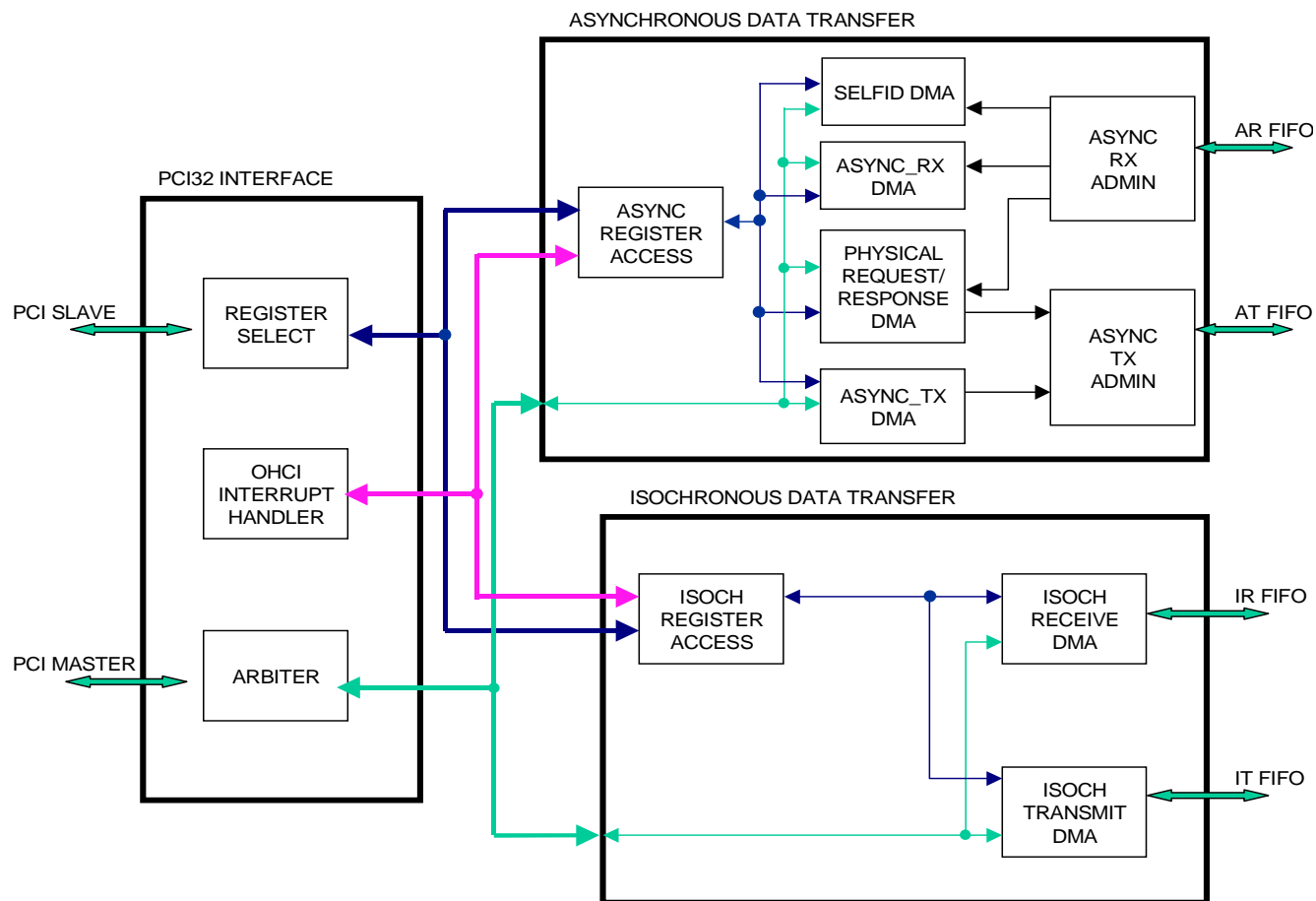


Figure 3. OHCI Core Block Diagram

#### 3.2 OHCI Data Transfer

The OHCI core consists of the three blocks shown in Figure 3: the PCI interface (PCI32\_interface), the isochronous data transfer, and the asynchronous data transfer blocks. The PCI interface provides an interface between the OHCI blocks and the PCI core. It contains an arbiter to select the appropriate OHCI data engine to gain access to the PCI core. In addition, the PCI interface includes a register select function to decode slave accesses to the OHCI core and select data from appropriate sources. The PCI interface also has an OHCI interrupt handler to service OHCI generated interrupts, which are ultimately translated into PCI interrupts.

#### 3.3 OHCI Isochronous Data Transfer

The isochronous data transfer logic, which is incorporated into the OHCI core, handles the transfer of isochronous data between the link core and the PCI interface module. It consists of the isochronous register access module, the isochronous transmit DMA module, the isochronous receive DMA module, the isochronous transmit (IT) FIFO, and the isochronous receive (IR) FIFO.



### 3 FW322 Functional Description (continued)

#### 3.4 Isochronous Register Access

The isochronous register access module services PCI slave accesses to OHCI registers within the isochronous block. The module also maintains the status of interrupts generated within the isochronous block and sends the isochronous interrupt status to the OHCI interrupt handler block.

##### 3.4.1 Isochronous Transmit DMA (ITDMA)

The isochronous transmit DMA (ITDMA) module moves data from host memory to the link core, which will then send the data via the PHY core to the 1394 bus. This module consists of eight isochronous transmit contexts, each of which is independently configurable by software, and is capable of sending data on a separate 1394 isochronous channel.

During each 1394 isochronous cycle, the ITDMA module will service each of the contexts and attempt to process one 1394 packet for each active context. While processing an active context, ITDMA will request access to the PCI bus. When granted PCI access, a descriptor block is fetched from host memory. This data is decoded by ITDMA to determine how much data is required and where in host memory the data resides. ITDMA initiates another PCI access to fetch this data, which is placed into the isochronous transmit FIFO for processing by the link core. If the context is not active, it is skipped by ITDMA for the current cycle.

After processing each context, ITDMA writes a cycle marker word in the transmit FIFO to indicate to the link core that there is no more data for this isochronous cycle. As a summary, the major steps for the FW322 ITDMA to transmit a packet are the following:

1. Fetch a descriptor block from host memory.
2. Fetch data specified by the descriptor block from host memory, and place it into the isochronous transmit FIFO.
3. Data in FIFO is read by the link and sent to the PHY core device interface.

##### 3.4.2 Isochronous Receive DMA (IRDMA)

The isochronous receive DMA (IRDMA) module moves data from the isochronous receive FIFO to host memory. It consists of eight isochronous contexts, each of which is independently controlled by software. Normally, each context can process data on a single 1394 isochronous channel. However, software can select one context to receive data on multiple channels.

When IRDMA detects that the link core has placed data into the receive FIFO, it immediately reads out the first word in the FIFO, which makes up the header of the isochronous packet. IRDMA extracts the channel number for the packet and packet filtering controls from the header. This information is compared with the control registers for each context to determine if any context is to process this packet.

If a match is found, IRDMA will request access to the PCI bus. When granted PCI access, a descriptor block is fetched from host memory. The descriptor provides information about the host memory block allocated for the incoming packet. IRDMA then reads the packet from the receive FIFO and writes the data to host memory via the PCI bus.

If no match is found, IRDMA will read the remainder of the packet from the receive FIFO, but not process the data in any way.

#### 3.5 OHCI Asynchronous Data Transfer

The asynchronous data transfer block within the OHCI core is functionally partitioned into blocks responsible for processing incoming SelfID packet streams, transmitting and receiving asynchronous 1394 packets, processing incoming physical request packets and outgoing physical response packets, and servicing accesses to OHCI registers within the respective asynchronous blocks.

#### 3.6 Asynchronous Register Access

The asynchronous register access module operates on PCI slave accesses to OHCI registers within the asynchronous block. The module also maintains the status of interrupts generated within the asynchronous block and sends the asynchronous interrupt status to the OHCI interrupt handler block.

### **3 FW322 Functional Description** (continued)

#### **3.6.1 Asynchronous Transmit DMA (ASYNC\_TX DMA, ASYNC\_TX\_ADMIN)**

The ASYNC\_TX DMA and ASYNC\_TX\_ADMIN blocks of the FW322 manage the asynchronous transmission of either request or response packets. The mechanism for asynchronous transmission of requests and responses is similar. The only difference is the system memory location of the buffer descriptor list when processing the two contexts. Therefore, the discussion below, which pertains to asynchronous transmit requests, parallels that of asynchronous transmit responses.

The FW322 asynchronous transmission of packets involves the following steps:

1. Fetch complete buffer descriptor block from host memory.
2. Get data from system memory and store into asynchronous transmit (AT) FIFO.
3. Request transfer of data from FIFO to the link core.
4. Handle retries, if any.
5. Handle errors in steps 1 to 4.
6. End the transfer if there are no errors.

#### **3.6.2 Asynchronous Receive DMA (ASYNC\_RX DMA, ASYNC\_RX\_ADMIN)**

The ASYNC\_RX DMA and ASYNC\_RX\_ADMIN blocks of the FW322 manage the processing of received packets. Data packets are parsed and stored in a dedicated asynchronous receive (AR) FIFO. Command descriptors are read through the PCI interface to determine the disposition of the data arriving through the 1394 link.

The header of the received packet is processed to determine, among other things, the following:

1. The type of packet received.
2. The source and destinations.
3. The data and size, if any.
4. Any required operation, for example, compare and swap operation.

The asynchronous data transfer block also handles DMA transfers of SelfID packets during the 1394 bus initialization phase and block transactions associated with physical requests.

#### **3.6.3 Physical Request/Response DMA**

The physical DMA block within the FW322 is responsible for processing incoming physical requests and outgoing physical responses. When an incoming asynchronous packet is received, the FW322 will process the packet automatically without software intervention if the packet meets a set of criteria defined within the OHCI specification. When the criteria are met, the asynchronous packet is reclassified as a physical packet. Requests that do not meet the criteria remain asynchronous packets and are processed as described above in Section 3.6.2. Processing packets as physical requests/responses allows the FW322 to either receive or transmit an asynchronous packet without the use of DMA descriptors. Instead, the FW322 directly writes or reads data to/from memory using the address defined within the packet header. Since physical packets can be processed independently of the system's software and CPU, processing a packet as physical results in a system performance optimization.

#### **3.6.4 SelfID DMA**

The SelfID DMA block within the FW322 is responsible for receiving SelfID packets during the bus initialization process. The received SelfID packets are written into a software-defined host memory buffer.

### 3 FW322 Functional Description (continued)

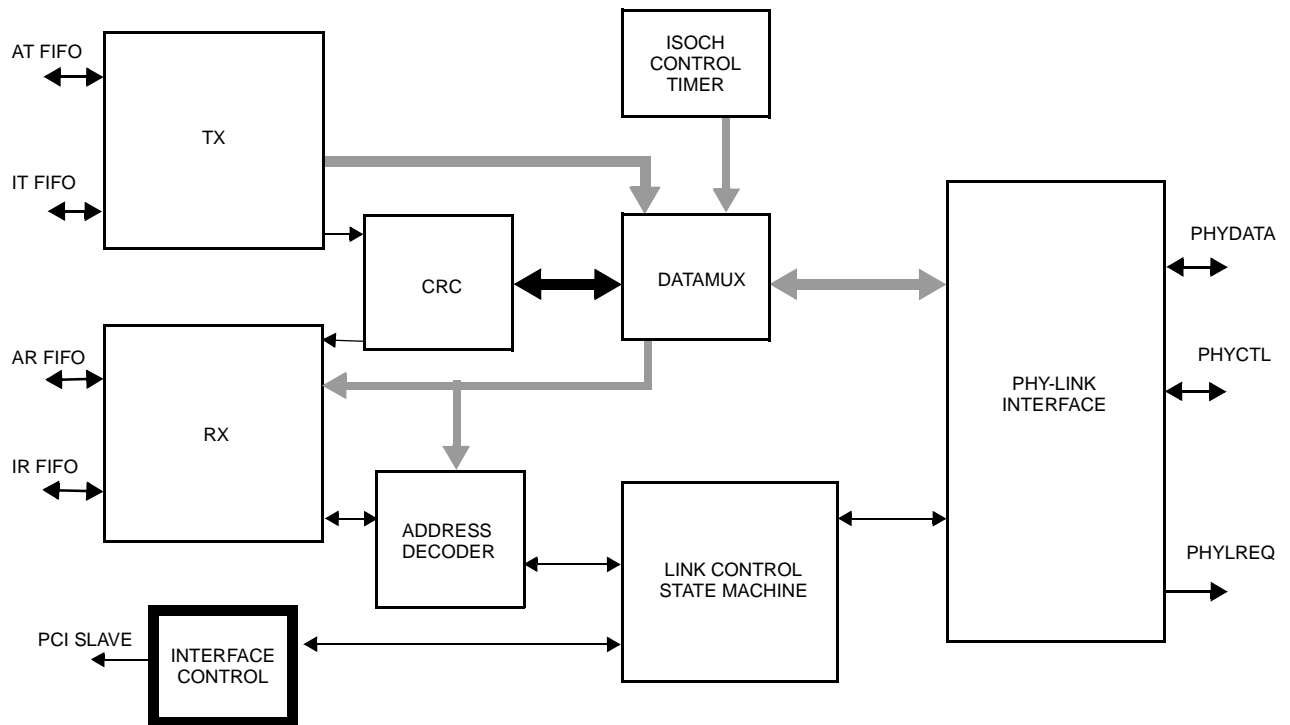


Figure 4. Link Core Block Diagram

#### 3.7 Link Core

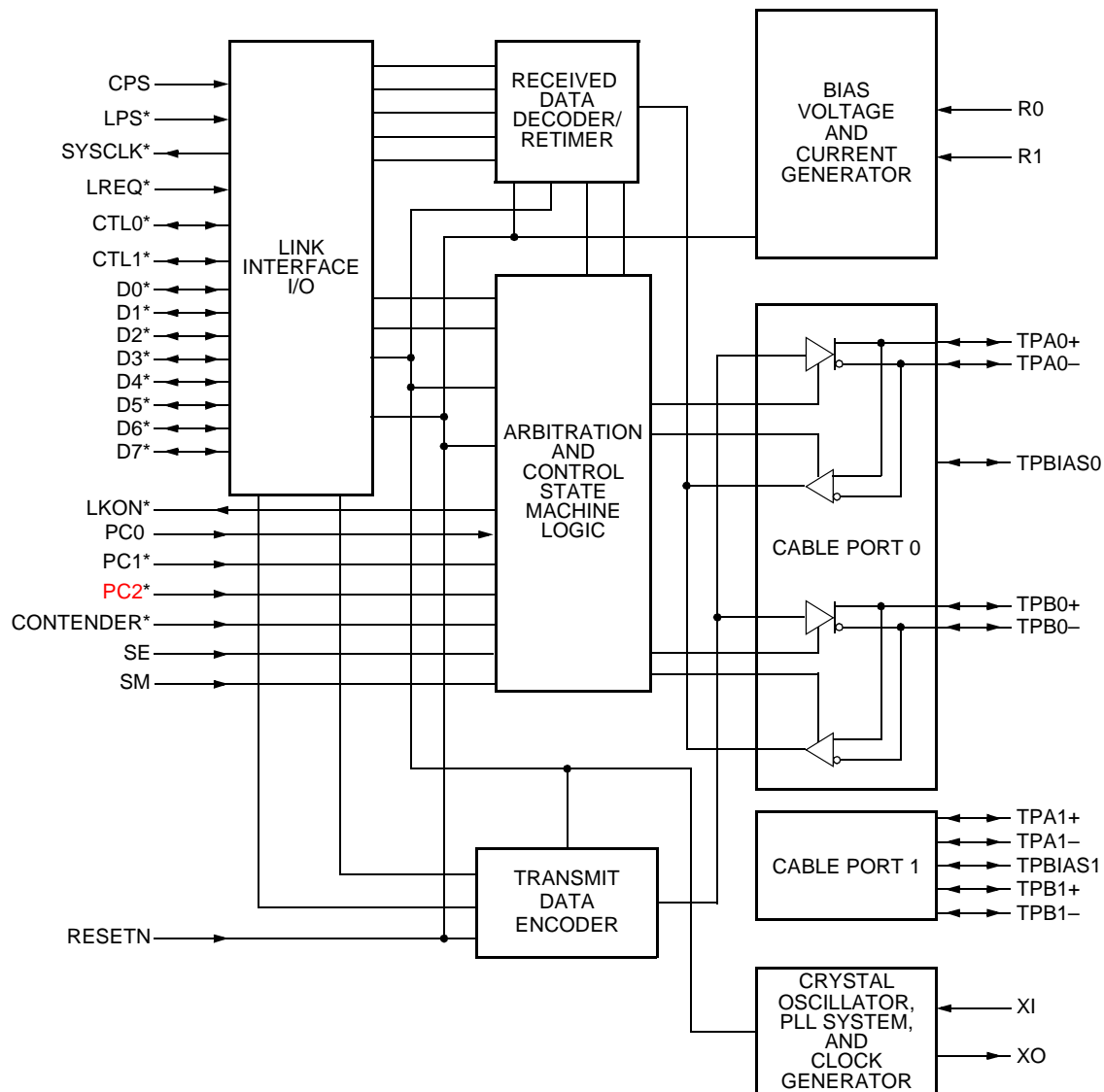
The link core shown in Figure 4 consists of the following blocks:

- **Link Control State Machine:** main link state machine that controls all other link core modules.
- **Transmit (TX):** reads from the AT and IT FIFOs and forms 1394 packets for transmit.
- **Receive (RX):** pipes incoming 1394 packet data to appropriate FIFO (if any).
- **Address Decoder:** decodes the destination ID of an incoming 1394 packet to determine if an acknowledge is needed.
- **CRC:** calculates and checks CRC on outgoing and incoming packets.
- **Isochronous Control Timer:** contains the logic for the 1394 cycle timer.
- **DataMUX:** pipes 1394 data to and from various modules.
- **Interface Control:** contains interrupt and registers for the link core. Interfaces with the slave control block of the PCI core.
- **PHY-Link Interface:** interfaces with the 1394 physical layer.

### 3 FW322 Functional Description (continued)

It is the responsibility of the link to ascertain if a received packet is to be forwarded to the OHCI for processing. If so, the packet is directed to a proper inbound FIFO for either the isochronous block or the asynchronous block to process. The link is also responsible for CRC generation on outgoing packets and CRC checking on received packets.

To become aware of data to be sent outbound on the 1394 bus, the link must monitor the OHCI FIFOs looking for packets in need of transmission. Based on data received from the OHCI block, the link will form packet headers for the 1394 bus. The link will alert the PHY core regarding the availability of the outbound data. It is the link's function to generate CRC for the outbound data. The link also provides PHY core register access for the OHCI.



5-5459.i(F) R.01

\* Internal points, inaccessible via external package pins.

Figure 5. The PHY Core Block Diagram

### 3 FW322 Functional Description (continued)

#### 3.8 PHY Core

The PHY core in Figure 5 provides the analog physical layer functions needed to implement a two-port node in a cable-based *IEEE* 1394-1995 and *IEEE* 1394a-2000 network.

Each cable port incorporates two differential line transceivers. The transceivers include circuitry to monitor the line conditions as needed for determining connection status, for initialization and arbitration, and for packet reception and transmission. The PHY core interfaces with the link core.

The PHY core requires either an external 24.576 MHz crystal or crystal oscillator. The internal oscillator drives an internal phase-locked loop (PLL), which generates the required 393.216 MHz reference signal. The 393.216 MHz reference signal is internally divided to provide the 49.152 MHz, 98.304 MHz, and 196.608 MHz clock signals that control transmission of the outbound encoded strobe and data information. The 49.152 MHz clock signal is also supplied to the associated link layer controller (LLC) for synchronization of the link with the PHY core and is used for resynchronization of the received data.

The PHY/link interface is a direct connection and does not provide isolation.

Data bits to be transmitted through the cable ports are received from the LLC on two, four, or eight data lines (D[0:7]), and are latched internally in the PHY in synchronization with the 49.152 MHz system clock. These bits are combined serially, encoded, and transmitted at 98.304 Mbits/s, 196.608 Mbits/s, or 393.216 Mbits/s as the outbound data-strobe information stream. During transmission, the encoded data information is transmitted differentially on the TPA and TPB cable pair(s).

During packet reception, the TPA and TPB transmitters of the receiving cable port are disabled, and the receivers for that port are enabled. The encoded data information is received on the TPA and TPB cable pair. The received data strobe information is decoded to recover the receive clock signal and the serial data bits. The serial data bits are split into two (for S100), four (for S200), or eight (for S400) parallel streams, resynchronized to the local system clock, and sent to the associated LLC. The received data is also transmitted (repeated) out of the other active (connected) cable ports.

Both the TPA and TPB cable interfaces incorporate differential comparators to monitor the line states during initialization and arbitration. The outputs of these comparators are used by the internal logic to determine the arbitration status. The TPA channel monitors the incoming cable common-mode voltage. The value of this common-mode voltage is used during arbitration to set the speed of the next packet transmission.

In addition, the TPB channel monitors the incoming cable common-mode voltage for the presence of the remotely supplied twisted-pair bias voltage. This monitor is called bias-detect.

The TPBIAS circuit monitors the value of incoming TPA pair common-mode voltage when local TPBIAS is inactive. Because this circuit has an internal current source and the connected node has a current sink, the monitored value indicates the cable connection status. The monitor is called connect-detect.

Both the TPB bias-detect monitor and TPBIAS connect-detect monitor are used in suspend/resume signaling and cable connection detection.

The PHY core provides a 1.86 V nominal bias voltage for driver load termination. This bias voltage, when seen through a cable by a remote receiver, indicates the presence of an active connection. The value of this bias voltage has been chosen to allow interoperability between transceiver chips operating from 5 V or 3 V nominal supplies. This bias voltage source should be stabilized by using an external filter capacitor of approximately 0.33  $\mu$ F.

The port transmitter circuitry and the receiver circuitry are disabled when the port is disabled, suspended, or disconnected.

### 3 FW322 Functional Description (continued)

The line drivers in the PHY core operate in a high-impedance current mode and are designed to work with external  $112\ \Omega$  line-termination resistor networks. One network is provided at each end of each twisted-pair cable. Each network is composed of a pair of series-connected  $56\ \Omega$  resistors. The midpoint of the pair of resistors that is directly connected to the twisted-pair A (TPA) signals is connected to the TPBIAS voltage signal. The midpoint of the pair of resistors that is directly connected to the twisted-pair B (TPB) signals is coupled to ground through a parallel RC network with recommended resistor and capacitor values of  $5\ \text{k}\Omega$  and  $220\ \text{pF}$ , respectively. The values of the external resistors are specified to meet the *1394a-2000 Specification* when connected in parallel with the internal receiver circuits.

An external resistor sets the driver output current, along with other internal operating currents. This resistor is connected between the R0 and R1 signals and has a value of  $2.49\ \text{k}\Omega \pm 1\%$ .

The **C** bit (20) in the SelfID packet (see Section 4.3.4.1 of the *IEEE 1394a-2000* specification for more details) has a default value of 0, which means this node is not a contender for bus manager. A value of 0 still allows this node to be considered by software for bus manager.

The least significant bit (23) in the Power Class (Pwr) field in the SelfID packet is set to the value of the PC2 pin. The two most significant bits (21 and 22) in the Pwr field are set to a default of 00. The PC2 pin is tied low or high to create a Pwr field of 000 (PC2 low) or 001 (PC2 high). See Section 4.3.4.1 of the *IEEE 1394a-2000* specification for additional details.

When the power supply of the PHY core is removed while the twisted-pair cables are connected, the PHY core transmitter and receiver circuitry has been designed to present a high impedance to the cable in order to not load the TPBIAS signal voltage on the other end of the cable.

Whenever the TPA $\pm$ /TPB $\pm$  signals are wired to a connector, they must be terminated using the normal termination network. This is required for reliable operation. For those applications when one or more of the FW322 ports are not wired to a connector, those unused ports may be left unconnected without normal termination. When a port does not have a cable connected, internal connect-detect circuitry will keep the port in a disconnected state.

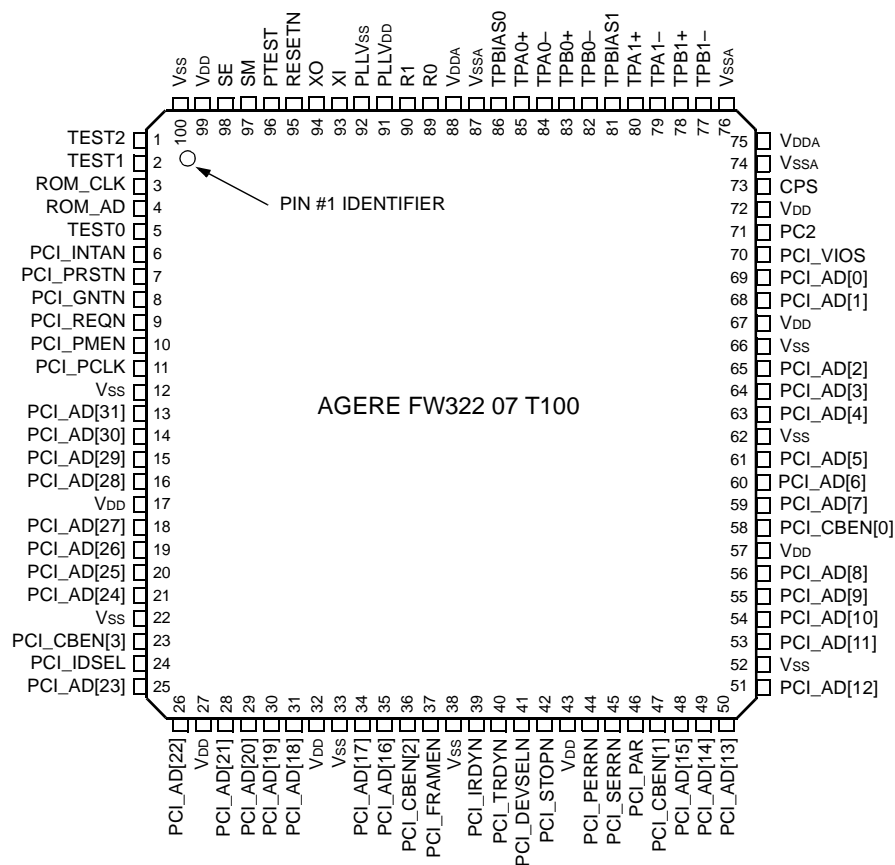
**Note:** All gap counts on all nodes of a 1394 bus must be identical. The software accomplishes this by issuing PHY core configuration packets (see Section 4.3.4.3 of *IEEE 1394-1995* and *1394a-2000* standards) or by issuing two bus resets, which resets the gap counts to the maximum level (3Fh).

The internal link power status (LPS) signal works with the internal LKON signal to manage the LLC power usage of the node. The LPS signal indicates if the LLC of the node is powered up or down. If LPS is inactive for more than  $1.2\ \mu\text{s}$  and less than  $25\ \mu\text{s}$ , the internal PHY/link interface is reset.

If LPS is inactive for greater than  $25\ \mu\text{s}$ , the PHY will disable the internal PHY/link interface to save power. The FW322 continues its repeater function even when the PHY/link interface is disabled. If the PHY then receives a link-on packet, the internal LKON signal is activated to output a  $6.114\ \text{MHz}$  signal, which can be used by the LLC to power itself up. Once the LLC is powered up, the internal LPS signal communicates this to the PHY and the internal PHY/link interface is enabled. The internal LKON signal is turned off when the LCtrl bit is set. (For more information on this bit, refer to the Table 65 on *PHY Core Register Fields* in this data sheet.)

Six of the FW322 pins are used to set up various test conditions used only during the device manufacturing process. These pins are SE, SM, TEST0, TEST1, TEST2, and PTEST.

## 4 Pin Information



5-7338 (F)r.2

Note: Active-low signals within this document are indicated by an N following the symbol names.

**Figure 6. Pin Assignments for the FW322 07 T100**

## 4 Pin Information (continued)

Table 1. Pin Descriptions

| Pin | Symbol*     | Type | Description   |
|-----|-------------|------|---|
| 1   | TEST2       | I    | <b>Test.</b> Used by Agere for device manufacturing testing. This pin must be tied high for normal operation. Do not tie this pin to Vss. |
| 2   | TEST1       | I    | <b>Test.</b> Used by Agere for device manufacturing testing. Tie to Vss for normal operation.   |
| 3   | ROM_CLK     | I/O  | <b>ROM Clock.</b>   |
| 4   | ROM_AD      | I/O  | <b>ROM Address/Data.</b>  |
| 5   | TEST0       | I    | <b>Test.</b> Used by Agere for device manufacturing testing. Tie to Vss for normal operation.   |
| 6   | PCI_INTAN   | O    | <b>PCI Interrupt (Active-Low).</b>  |
| 7   | PCI_PRSTN   | I    | <b>PCI Reset (Active-Low).</b>  |
| 8   | PCI_GNTN    | I    | <b>PCI Grant Signal (Active-Low).</b>   |
| 9   | PCI_REQN    | O    | <b>PCI Request Signal (Active-Low).</b>   |
| 10  | PCI_PMEN    | O    | <b>PCI Power Management Event (Active-Low).</b> A PCI power management event will be indicated if this signal is low.                     |
| 11  | PCI_PCLK    | I    | <b>PCI Clock Input.</b> 33 MHz.   |
| 12  | Vss         | —    | <b>Digital Ground.</b>  |
| 13  | PCI_AD[31]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 14  | PCI_AD[30]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 15  | PCI_AD[29]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 16  | PCI_AD[28]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 17  | VDD         | —    | <b>Digital Power.</b>   |
| 18  | PCI_AD[27]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 19  | PCI_AD[26]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 20  | PCI_AD[25]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 21  | PCI_AD[24]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 22  | Vss         | —    | <b>Digital Ground.</b>  |
| 23  | PCI_CBEN[3] | I/O  | <b>PCI Command/Byte Enable (Active-Low).</b>  |
| 24  | PCI_IDSEL   | I    | <b>PCI ID Select.</b>   |
| 25  | PCI_AD[23]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 26  | PCI_AD[22]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 27  | VDD         | —    | <b>Digital Power.</b>   |
| 28  | PCI_AD[21]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 29  | PCI_AD[20]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 30  | PCI_AD[19]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 31  | PCI_AD[18]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 32  | VDD         | —    | <b>Digital Power.</b>   |
| 33  | Vss         | —    | <b>Digital Ground.</b>  |
| 34  | PCI_AD[17]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 35  | PCI_AD[16]  | I/O  | <b>PCI Address/Data Bit.</b>  |
| 36  | PCI_CBEN[2] | I/O  | <b>PCI Command/Byte Enable Signal (Active-Low).</b>   |
| 37  | PCI_FRAMEN  | I/O  | <b>PCI Frame Signal (Active-Low).</b>   |

\* Active-low signals within this document are indicated by an N following the symbol names.



## 4 Pin Information (continued)

Table 1. Pin Descriptions (continued)

| Pin | Symbol*     | Type | Description   |
|-----|-------------|------|---|
| 38  | Vss         | —    | Digital Power.  |
| 39  | PCI_IRDYN   | I/O  | PCI Initiator Ready Signal (Active-Low).  |
| 40  | PCI_TRDYN   | I/O  | PCI Target Ready Signal (Active-Low).   |
| 41  | PCI_DEVSELN | I/O  | PCI Device Select Signal (Active-Low).  |
| 42  | PCI_STOPN   | I/O  | PCI Stop Signal (Active-Low).   |
| 43  | VDD         | —    | Digital Power.  |
| 44  | PCI_PERRN   | I/O  | PCI Parity Error Signal (Active-Low).   |
| 45  | PCI_SERRN   | I/O  | PCI System Error Signal (Active-Low).   |
| 46  | PCI_PAR     | I/O  | PCI Parity Signal.  |
| 47  | PCI_CBEN[1] | I/O  | PCI Command/Byte Enable Signal (Active-Low).  |
| 48  | PCI_AD[15]  | I/O  | PCI Address/Data Bit.   |
| 49  | PCI_AD[14]  | I/O  | PCI Address/Data Bit.   |
| 50  | PCI_AD[13]  | I/O  | PCI Address/Data Bit.   |
| 51  | PCI_AD[12]  | I/O  | PCI Address/Data Bit.   |
| 52  | Vss         | —    | Digital Ground.   |
| 53  | PCI_AD[11]  | I/O  | PCI Address/Data Bit.   |
| 54  | PCI_AD[10]  | I/O  | PCI Address/Data Bit.   |
| 55  | PCI_AD[9]   | I/O  | PCI Address/Data Bit.   |
| 56  | PCI_AD[8]   | I/O  | PCI Address/Data Bit.   |
| 57  | VDD         | —    | Digital Power.  |
| 58  | PCI_CBEN[0] | I/O  | PCI Command/Byte Enable Signal (Active-Low).  |
| 59  | PCI_AD[7]   | I/O  | PCI Address/Data Bit.   |
| 60  | PCI_AD[6]   | I/O  | PCI Address/Data Bit.   |
| 61  | PCI_AD[5]   | I/O  | PCI Address/Data Bit.   |
| 62  | Vss         | —    | Digital Ground.   |
| 63  | PCI_AD[4]   | I/O  | PCI Address/Data Bit.   |
| 64  | PCI_AD[3]   | I/O  | PCI Address/Data Bit.   |
| 65  | PCI_AD[2]   | I/O  | PCI Address/Data Bit.   |
| 66  | Vss         | —    | Digital Ground.   |
| 67  | VDD         | —    | Digital Power.  |
| 68  | PCI_AD[1]   | I/O  | PCI Address/Data Bit.   |
| 69  | PCI_AD[0]   | I/O  | PCI Address/Data Bit.   |
| 70  | PCI_VIOS    | —    | <b>PCI Signaling Indicator.</b> For PCI applications that use a universal expansion board (see <i>PCI Local Bus Specification</i> , Rev. 2.3, Section 4.1.1), connect this pin to the VI/O pin. For other cases, connect this pin to 3.3 V for PCI buses using 3.3 V signaling or to 5 V for PCI buses using 5 V signaling. |

\* Active-low signals within this document are indicated by an N following the symbol names.

## 4 Pin Information (continued)

Table 1. Pin Descriptions (continued)

| Pin | Symbol* | Type       | Description  |
|-----|---------|------------|--|
| 71  | PC2     | I          | <b>Power-Class Indicator.</b> On hardware reset (RESETN), this input sets the default value of the least significant bit in the Power Class field (Pwr) in the SelfID packet (see Section 4.3.4.1 of the <i>1394a-2000 Specification</i> ). This bit can be tied to VDD (high) or to VSS (low) as required for particular power consumption and source characteristics. The two most significant Pwr bits in the SelfID are internally set to zero. As an example, for a Pwr field value of 001, PC2 = 1.  |
| 72  | VDD     | —          | <b>Digital Power.</b>  |
| 73  | CPS     | I          | <b>Cable Power Status.</b> CPS is normally connected to the cable power through a 400 kΩ resistor. This circuit drives an internal comparator that detects the presence of cable power. This information is maintained in one internal register and is available to the LLC by way of a register read (see <i>IEEE 1394a-2000, Standard for a High Performance Serial Bus</i> , Sections 4.2.2.7 and 5B.1).<br><b>Note:</b> This pin can be left unconnected for applications that do not use 1394 bus power (VP). When this pin is grounded, the PWR_FAIL bit in PHY register 01012 will set. |
| 74  | VSSA    | —          | <b>Analog Circuit Ground.</b> All VSSA signals should be tied together to a low-impedance ground plane.  |
| 75  | VDDA    | —          | <b>Analog Circuit Power.</b> VDDA supplies power to the analog portion of the device.  |
| 76  | VSSA    | —          | <b>Analog Circuit Ground.</b> All VSSA signals should be tied together to a low-impedance ground plane.  |
| 77  | TPB1–   | Analog I/O | <b>Port 1, Port Cable Pair B.</b> TPB1± is the port B connection to the twisted-pair cable. Board traces from each pair of positive and negative differential signal pins should be kept matched and as short as possible to the external load resistors and to the cable connector. When the FW322's 1394 port pins are not wired to a connector, the unused port pins may be left unconnected. Internal connect-detect circuitry will keep the port in a disconnected state.   |
| 78  | TPB1+   |            |  |
| 79  | TPA1–   | Analog I/O | <b>Port 1, Port Cable Pair A.</b> TPA1± is the port A connection to the twisted-pair cable. Board traces from each pair of positive and negative differential signal pins should be kept matched and as short as possible to the external load resistors and to the cable connector. When the FW322's 1394 port pins are not wired to a connector, the unused port pins may be left unconnected. Internal connect-detect circuitry will keep the port in a disconnected state.   |
| 80  | TPA1+   |            |  |

\* Active-low signals within this document are indicated by an N following the symbol names.

## 4 Pin Information (continued)

Table 1. Pin Descriptions (continued)

| Pin | Symbol* | Type       | Description  |
|-----|---------|------------|--|
| 81  | TPBIAS1 | Analog I/O | <b>Port 1, Twisted-Pair Bias.</b> TPBIAS1 provides the 1.86 V nominal bias voltage needed for proper operation of the twisted-pair cable drivers and receivers and for sending a valid cable connection signal to the remote nodes. When the FW322's 1394 port pins are not wired to a connector, the unused port pins may be left unconnected. Internal connect-detect circuitry will keep the port in a disconnected state.  |
| 82  | TPB0–   | Analog I/O | <b>Port 0, Port Cable Pair B.</b> TPB0± is the port B connection to the twisted-pair cable. Board traces from each pair of positive and negative differential signal pins should be kept matched and as short as possible to the external load resistors and to the cable connector. When the FW322's 1394 port pins are not wired to a connector, the unused port pins may be left unconnected. Internal connect-detect circuitry will keep the port in a disconnected state. |
| 83  | TPB0+   |            |  |
| 84  | TPA0–   | Analog I/O | <b>Port 0, Port Cable Pair A.</b> TPA0± is the port A connection to the twisted-pair cable. Board traces from each pair of positive and negative differential signal pins should be kept matched and as short as possible to the external load resistors and to the cable connector. When the FW322's 1394 port pins are not wired to a connector, the unused port pins may be left unconnected. Internal connect-detect circuitry will keep the port in a disconnected state. |
| 85  | TPA0+   |            |  |
| 86  | TPBIAS0 | Analog I/O | <b>Port 0, Twisted-Pair Bias.</b> TPBIAS0 provides the 1.86 V nominal bias voltage needed for proper operation of the twisted-pair cable drivers and receivers and for sending a valid cable connection signal to the remote nodes. When the FW322's 1394 port pins are not wired to a connector, the unused port pins may be left unconnected. Internal connect-detect circuitry will keep the port in a disconnected state.  |
| 87  | VSSA    | —          | <b>Analog Circuit Ground.</b> All VSSA signals should be tied together to a low-impedance ground plane.  |
| 88  | VDDA    | —          | <b>Analog Circuit Power.</b> VDDA supplies power to the analog portion of the device.  |
| 89  | R0      | I          | <b>Current Setting Resistor.</b> An internal reference voltage is applied to a resistor connected between R0 and R1 to set the operating current and the cable driver output current. A low temperature-coefficient resistor (TCR) with a value of 2.49 kΩ ± 1% should be used to meet the IEEE 1394-1995 standard requirements for output voltage limits.   |
| 90  | R1      |            |  |
| 91  | PLLVD   | —          | <b>Power for PLL Circuit.</b> PLLVD supplies power to the PLL circuitry portion of the device.   |
| 92  | PLLVS   | —          | <b>Ground for PLL Circuit.</b> PLLVS is tied to a low-impedance ground plane.  |

\* Active-low signals within this document are indicated by an N following the symbol names.

## 4 Pin Information (continued)

Table 1. Pin Descriptions (continued)

| Pin | Symbol* | Type       | Description   |
|-----|---------|------------|---|
| 93  | XI      | Analog I/O | <b>Crystal Oscillator.</b> XI and XO connect to a 24.576 MHz parallel resonant fundamental mode crystal. Although when a 24.576 MHz clock source is used, it can be connected to XI with XO left unconnected. The optimum values for the external shunt capacitors are dependent on the specifications of the crystal used. It is necessary to add an external series resistor to the XO pin. The value of the resistor is nominally 400 $\Omega$ . For more details, refer to the Crystal Selection Considerations section in this data sheet. Note that it is very important to place the crystal as close as possible to the XO and XI pins, i.e., within 0.5 in./1.27 cm. For more important details regarding the crystal, refer to the <i>FW323/FW322 Hardware Implementation Design Guideline</i> Application Note.  |
| 94  | XO      |            |   |
| 95  | RESETN  | I          | <b>Reset (Active-Low).</b> When RESETN is asserted low (active), a 1394 bus reset condition is set on the active cable ports and the FW322 is reset to the reset start state. To guarantee that the PHY will reset, this pin must be held low for at least 2 ms. An internal pull-up resistor, connected to VDD, is provided, so only an external delay capacitor (0.1 $\mu$ F) and resistor (510 k $\Omega$ ), in parallel, are required to connect this pin to ground. This circuitry will ensure that the capacitor will be discharged when PHY power is removed. The input is a standard logic buffer and can also be driven by an open-drain logic output buffer. Do not leave this pin unconnected. This pin is also used with the EEPROM interface. It is the powerup reset pin. This pin is asserted low (active) to indicate a powerup reset. Refer to the <i>FW322 06/FW323 06 EEPROM Interface and Start-up Behavior</i> Application Note sections titled Initiation of EEPROM Load and Initial Powerup. |
| 96  | PTEST   | I          | <b>Test.</b> Used by Agere for device manufacturing testing. Tie to Vss for normal operation.   |
| 97  | SM      | I          | <b>Test Mode Control.</b> SM is used during Agere's manufacturing test and should be tied to Vss for normal operation.  |
| 98  | SE      | I          | <b>Test Mode Control.</b> SE is used during Agere's manufacturing test and should be tied to Vss for normal operation.  |
| 99  | VDD     | —          | <b>Digital Power.</b>   |
| 100 | VSS     | —          | <b>Digital Ground.</b>  |

\* Active-low signals within this document are indicated by an N following the symbol names.

Note: For those applications when one or more FW322 ports are not wired to a connector, those unused ports may be left unconnected without normal termination. When a port does not have a cable connected, internal connect-detect circuitry will keep the port in a disconnected state.

## 5 Internal Registers

This section provides a summary of the internal registers within the FW322, including both PCI configuration registers and OHCI registers. Register default values, registers and bits that have not been implemented in the FW322, and other information specific to the FW322 will be noted.

Please refer to the *PCI Local Bus Specification v.2.2*, *PCI Bus Power Management Interface Specification, v.1.1*, 1394 OHCI specification v.1.1, and the *IEEE standard 1394a-2000 Specification* for further details concerning these registers.

Table 2 describes the field access tags that are designated in the **Type** column of the register tables in this document.

**Table 2. Bit-Field Access Tag Description**

| Access Tag | Name   | Description   |
|------------|--------|---|
| 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 1. Writes of 0 have no effect.     |
| C          | Clear  | Field may be cleared by a write of 1. Writes of 0 have no effect. |
| U          | Update | Field may be autonomously updated by the FW322.                   |

### 5.1 PCI Configuration Registers

Table 3 illustrates the PCI configuration header that includes both the predefined portion of the configuration space and the user-definable registers.

**Table 3. PCI Configuration Register Map**

| Register Name [Default]                 |                      |                              |   | Offset |
|---|----------------------|------------------------------|---|--------|
| Device ID [5811h]                       |                      | Vendor ID [11C1h]            |   | 00h    |
| Status [02901h]                         |                      | Command [0000h]              |   | 04h    |
| Class Code [0C0010h]                    |                      |                              | Revision ID [7xh]*                                      | 08h    |
| BIST [00h]                              | Header Type [00h]    | Latency Timer† [00h]         | Cache Line Size† [00h]                                  | 0Ch    |
| OHCI Base Address Register [0000 0000h] |                      |                              |   | 10h    |
| Reserved                                |                      |                              |   | 14h    |
| Reserved                                |                      |                              |   | 18h    |
| Reserved                                |                      |                              |   | 1Ch    |
| Reserved                                |                      |                              |   | 20h    |
| Reserved                                |                      |                              |   | 24h    |
| CardBus CIS Pointer [0000 0000h]        |                      |                              |   | 28h    |
| Subsystem ID† [0000h]                   |                      | Subsystem Vendor ID† [0000h] |   | 2Ch    |
| Reserved                                |                      |                              |   | 30h    |
| Reserved                                |                      |                              | PCI Power Manage-<br>ment Capabilities<br>Pointer [44h] | 34h    |
| Reserved                                |                      |                              |   | 38h    |
| Maximum Latency†<br>[18h]               | Minimum Grant† [0Ch] | Interrupt Pin [01h]          | Interrupt Line [00h]                                    | 3Ch    |

\* x is a minor revision number of the FW322 07 T100 and may be any value from 0 hex to F hex.

† Values for this register can be loaded from a serial EEPROM during the powerup sequence.

## 5 Internal Registers (continued)

**Table 3. PCI Configuration Register Map** (continued)

| Register Name [Default]                 |                         |                               | Offset |
|---|-------------------------|-------------------------------|--------|
| PCI OHCI Control Register† [0000 0000h] |                         |                               | 40h    |
| Power Management Capabilities† [7E02h]  | Next Item Pointer [00h] | Capability ID [01h]           | 44h    |
| Pm Data† [00h]                          | Pmcsr_bse [00h]         | Power Management CSR‡ [0000h] | 48h    |
| Reserved                                |                         |                               | 4C—FCh |

† Values for this register can be loaded from a serial EEPROM during the powerup sequence.

### 5.2 Vendor ID Register

The vendor ID register contains a value allocated by the PCI SIG and identifies the manufacturer of the device. The vendor ID assigned to Agere is 11C1h.

Offset: 00h  
Default: 11C1h  
Type: Read only

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.1

### 5.3 Device ID Register

The device ID register contains a value assigned to the FW322 by Agere. The device identification for the FW322 is 5811h.

Offset: 02h  
Default: 5811h  
Type: Read only

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.1

## 5 Internal Registers (continued)

### 5.4 PCI Command Register

The command register provides control over the FW322 interface to the PCI bus. All bit functions adhere to the definitions in the *PCI Local Bus Specification*, as in the following bit descriptions.

Offset: 04h

Default: 0000h

Type: Read/write

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.2 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.1

**Table 4. PCI Command Register Description**

| Bit   | Field Name  | Type | Description  |
|-------|-------------|------|--|
| 15:11 | Reserved    | R    | <b>Reserved.</b> Bits 15:10 return 0s when read.   |
| 10    | INT_DISABLE | RW   | <b>Interrupt Disable.</b> This bit disables the FW322 from asserting PCI_INTAN. A value of 0 enables the assertion of the PCI_INTAN pin. A value of 1 disables assertion of the PCI_INTAN pin. The default value after reset is 0. |
| 9     | FBB_ENB     | R    | <b>Fast Back-to-Back Enable.</b> The FW322 does not generate fast back-to-back transactions; thus, this bit returns 0 when read.   |
| 8     | SERR_ENB    | RW   | <b>SERR Enable.</b> When this bit is set, the FW322 SERR driver is enabled. PCI_SERRN can be asserted after detecting an address parity error on the PCI bus.  |
| 7     | STEP_ENB    | R    | <b>Address/Data Stepping Control.</b> The FW322 does not support address/data stepping; thus, this bit is hardwired to 0.  |
| 6     | PERR_ENB    | RW   | <b>Parity Error Enable.</b> When this bit is set, the FW322 is enabled to drive PERR response to parity errors through the PCI_PERRN signal.   |
| 5     | VGA_ENB     | R    | <b>VGA Palette Snoop Enable.</b> The FW322 does not feature VGA palette snooping. This bit returns 0 when read.  |
| 4     | MWI_ENB     | RW   | <b>Memory Write and Invalidate Enable.</b> When this bit is set, the FW322 is enabled to generate MWI PCI bus commands. If this bit is reset, then the FW322 generates memory write commands instead.                              |
| 3     | SPECIAL     | R    | <b>Special Cycle Enable.</b> The FW322 function does not respond to special cycle transactions. This bit returns 0 when read.  |
| 2     | MASTER_ENB  | RW   | <b>Bus Master Enable.</b> When this bit is set, the FW322 is enabled to initiate cycles on the PCI bus.  |
| 1     | MEMORY_ENB  | RW   | <b>Memory Response Enable.</b> Setting this bit enables the FW322 to respond to memory cycles on the PCI bus. This bit must be set to access OHCI registers.   |
| 0     | IO_ENB      | R    | <b>I/O Space Enable.</b> The FW322 does not implement any I/O mapped functionality; thus, this bit returns 0 when read.  |

## 5 Internal Registers (continued)

### 5.5 PCI Status Register

The status register provides status information for PCI bus related events. All bit functions adhere to the definitions in the *PCI Local Bus Specification*, v.2.2, Table 6.2.

Offset: 06h  
Default: 0290h  
Type: Read/write

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.3 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.2

**Table 5. PCI Status Register**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 15   | PAR_ERR    | RCU  | <b>Detected Parity Error.</b> This bit must be set by the device whenever it detects a parity error, even if parity error handling is disabled.  |
| 14   | SYS_ERR    | RCU  | <b>Signaled System Error.</b> This bit must be set whenever the device asserts SERR#.  |
| 13   | MABORT     | RCU  | <b>Received Master Abort.</b> This bit must be set by a master device whenever its transaction (except for special cycle) is terminated with master-abort.   |
| 12   | TABORT_REC | RCU  | <b>Received Target Abort.</b> This bit must be set by a master device whenever its transaction is terminated with target-abort.  |
| 11   | TABORT_SIG | RCU  | <b>Signaled Target Abort.</b> This bit must be set by a target device whenever it terminates a transaction with target-abort.  |
| 10:9 | PCI_SPEED  | R    | <b>DEVSEL Timing.</b> Bits 9 and 10 encode the timing of DELSEL# (see Section 3.6.1 of the PCI Specification). These bits must indicate the slowest time that a device asserts DEVSEL# for any bus command except configuration read and configuration write. The default timing is 01 (medium).   |
| 8    | DATAPAR    | RCU  | <b>Master Data Parity Error.</b> See Table 6-2 of the PCI Specification for more information.  |
| 7    | FBB_CAP    | R    | <b>Fast Back-to-Back Capable.</b> Indicates whether or not the target is capable of accepting fast back-to-back transactions when the transactions are not to the same agent. The FW322 does not support back-to-back transactions.  |
| 6    | Reserved   | R    | <b>Reserved.</b>   |
| 5    | 66MHZ      | R    | <b>66 MHz Capable.</b> Indicates whether or not this device is capable of running at 66 MHz as defined in Chapter 7 of the PCI Specification. The FW322 reports a value of zero in this field indicating that 66 MHz functionality is not supported.   |
| 4    | CAPLIST    | R    | <b>Capabilities List.</b> Indicates whether or not this device implements the pointer for a New Capabilities linked list at offset 34h. A value of zero indicates that no New Capabilities linked list is available. A value of one indicates that the value read at offset 34h is a point in Configuration Space to a linked list of new capabilities. (See Section 6.7 of the PCI Specification for more details.) |
| 3    | INT_STATUS | R    | <b>Interrupt Status.</b> This read-only bit reflects the state of the interrupt pin in the FW322. Only when the Interrupt Disable bit in the command register is a 0 and this Interrupt Status bit is a 1 will the FW322's PCI_INTAN pin be asserted. Setting the Interrupt Disable bit to a 1 has no effect on the state of this pin.   |
| 2:0  | Reserved   | R    | <b>Reserved.</b>   |



## 5 Internal Registers (continued)

### 5.6 Class Code and Revision ID Registers

The class code register and revision ID register categorize the FW322 as a serial bus controller (0Ch), controlling an *IEEE* 1394 bus (00h), with an OHCI programming model (10h). Furthermore, the chip revision is indicated in the lower byte.

Offset: 08h

Default: 0C00 107xh\*

Type: Read only

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.1 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.3 and A.3.4.

**Table 6. Class Code and Revision ID Register Description**

| Bit   | Field Name | Type | Description  |
|-------|------------|------|--|
| 31:24 | BASECLASS  | R    | <b>Base Class.</b> This field returns 0Ch when read, which classifies the function as a serial bus controller.   |
| 23:16 | SUBCLASS   | R    | <b>Subclass.</b> This field returns 00h when read, which specifically classifies the function as an <i>IEEE</i> 1394 serial bus controller.  |
| 15:8  | PGMIF      | R    | <b>Programming Interface.</b> This field returns 10h when read, indicating that the programming model is compliant with the <i>1394 Open Host Controller Interface Specification</i> . |
| 7:0   | CHIPREV    | R    | <b>Silicon Revision.</b> This field returns 7xh* when read, indicating the silicon revision of the FW322.  |

\* x is a minor revision number of the FW322 07 T100 and may be any value from 0 hex to F hex.

### 5.7 Latency Timer and Cache Line Size Register

The latency timer and class cache line size register is programmed by host BIOS to indicate system cache line size and the latency timer associated with the FW322. If a serial EEPROM is detected, then the contents of this register are loaded from the serial EEPROM interface after a PCI reset. If no serial EEPROM is detected, then this register returns a default value of 0000h.

Offset: 0Ch

Default: 0000h

Type: Read/write

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.4

**Table 7. Latency Timer and Class Cache Line Size Register Description**

| Bit  | Field Name    | Type | Description   |
|------|---------------|------|---|
| 15:8 | LATENCY_TIMER | RW   | <b>PCI Latency Timer.</b> The value in this register specifies the latency timer, in units of PCI clock cycles, for the FW322. When the FW322 is a PCI bus initiator and asserts FRAME, the latency timer begins counting from zero. If the latency timer expires before the FW322 transaction has terminated, then the FW322 terminates the transaction when its PCI_GNTN is deasserted. |
| 7:0  | CACHELINE_SZ  | RW   | <b>Cache Line Size.</b> This value is used by the FW322 during memory write and invalidate, memory read line, and memory read multiple transactions.  |

## 5 Internal Registers (continued)

### 5.8 Header Type and BIST Register

The header type and BIST register indicates the FW322 PCI header type.

Offset: 0Eh  
Default: 0000h  
Type: Read only

Reference: *PCI Local Bus Specification*, Rev. 2.3, Sections 6.2.1 and 6.2.4

**Table 8. Header Type and BIST Register Description**

| Bit  | Field Name  | Type | Description  |
|------|-------------|------|--|
| 15:8 | BIST        | R    | <b>Built-In Self-Test.</b> The FW322 does not include a built-in self-test; thus, this field returns 00h when read.                    |
| 7:0  | HEADER_TYPE | R    | <b>PCI Header Type.</b> The FW322 includes the standard PCI header, and this is communicated by returning 00h when this field is read. |

### 5.9 OHCI Base Address Register

The OHCI base address register is programmed with a base address referencing the memory-mapped OHCI control. When BIOS writes all 1s to this register, the value read back is FFFF F000h, indicating that 4 Kbytes of memory address space are required for the OHCI registers.

Offset: 10h  
Default: 0000 0000h  
Type: Read/write

Reference: *PCI Local Bus Specification*, Rev. 2.3, Sections 6.2.5 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.5

**Table 9. OHCI Base Address Register Description**

| Bit   | Field Name   | Type | Description   |
|-------|--------------|------|---|
| 31:12 | OHCIREG_PTR  | RW   | <b>OHCI Register Pointer.</b> Specifies the upper 20 bits of the 32-bit OHCI base address.  |
| 11:4  | OHCI_SZ      | R    | <b>OHCI Register Size.</b> This field returns 0s when read, indicating that the OHCI registers require a 4 Kbyte region of memory.  |
| 3     | OHCI_PF      | R    | <b>OHCI Register Prefetch.</b> This bit returns 0 when read, indicating that the OHCI registers are not prefetchable.   |
| 2:1   | OHCI_MEMTYPE | R    | <b>OHCI Memory Type.</b> This field returns 0s when read, indicating that the OHCI base address register is 32 bits wide and mapping can be done anywhere in the 32-bit memory space. |
| 0     | OHCI_MEM     | R    | <b>OHCI Memory Indicator.</b> This bit returns 0 when read, indicating that the OHCI registers are mapped into system memory space.   |

## 5 Internal Registers (continued)

### 5.10 PCI Subsystem Identification Register

The PCI subsystem identification register is used to uniquely identify the card or system in which the FW322 resides. If a serial EEPROM is present, these values are loaded from the EEPROM during the powerup sequence. Subsystem vendor IDs can be obtained from the PCI SIG. Values for the subsystem ID are vendor specific.

By default, the PCI subsystem ID and PCI subsystem vendor ID registers are read only. However, if a serial EEPROM is not interfaced to the FW322 06, bit 0 (SubSystemWriteEn) of the PCI config register, offset 4Ch can be set to enable writes to the PCI subsystem ID and PCI subsystem vendor ID so that these registers can be customized to the correct ID values. After the IDs have been written, the SubSystemWriteEn bit should be reset to protect the data from being overwritten.

Offset: 2Ch  
Default: 0000 0000h  
Type: Read/write

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.4

**Table 10. PCI Subsystem Identification Register Description**

| Bit   | Field Name | Type | Description   |
|-------|------------|------|---|
| 31:16 | SSID       | RU   | <b>Subsystem ID.</b> This field indicates the subsystem ID.               |
| 15:0  | SSVID      | RU   | <b>Subsystem Vendor ID.</b> This field indicates the subsystem vendor ID. |

### 5.11 PCI Power Management Capabilities Pointer Register

The PCI power management capabilities pointer register provides a pointer into the PCI configuration header where the PCI power management register block resides. The FW322 configuration words at offsets 44h and 48h provide the power management registers. This register is read only and returns 44h when read.

Offset: 34h  
Default: 44h  
Type: Read only

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.4 and 6.7 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.6.

## 5 Internal Registers (continued)

### 5.12 Interrupt Line and Pin Register

The interrupt line and pin register is used to communicate interrupt line routing information.

Offset: 3Ch  
Default: 0100h  
Type: Read only

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.4 and 6.7

**Table 11. Interrupt Line and Pin Register Description**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 15:8 | INTR_PIN   | R    | <b>Interrupt Pin Register.</b> This register returns 01h when read, indicating that the FW322 PCI function signals interrupts on the INTA pin.           |
| 7:0  | INTR_LINE  | RW   | <b>Interrupt Line Register.</b> This register is programmed by the system and indicates to software to which interrupt line the FW322 INTA is connected. |

### 5.13 MIN\_GNT and MAX\_LAT Register

The MIN\_GNT and MAX\_LAT register is used to communicate to the system the desired setting of the latency timer register. If a serial EEPROM is detected, then the contents of this register are loaded from the serial EEPROM interface after a PCI reset. If no serial EEPROM is detected, then this register returns a default value that corresponds to the MIN\_GNT = 0Ch, MAX\_LAT = 18h.

Offset: 3Eh  
Default: 180Ch  
Type: Read only

Reference: *PCI Local Bus Specification*, Rev. 2.3, Section 6.2.4

**Table 12. MIN\_GNT and MAX\_LAT Register Description**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 15:8 | MAX_LAT    | RU   | <b>Maximum Latency.</b> The contents of this register may be used by host BIOS to assign an arbitration priority level to the FW322. The default for this register (18h) indicates that the FW322 may need to access the PCI bus as often as every 0.25 $\mu$ s; thus, an extremely high-priority level is requested. The contents of this field may also be loaded from the serial ROM.                         |
| 7:0  | MIN_GNT    | RU   | <b>Minimum Grant.</b> The contents of this register may be used by host BIOS to assign a latency timer register value to the FW322. The default (0Ch) for this register indicates that the FW322 may need to sustain burst transfers for nearly 64 $\mu$ s; thus, requesting a large value be programmed in the FW322 latency timer register. The contents of this field may also be loaded from the serial ROM. |

## 5 Internal Registers (continued)

### 5.14 PCI OHCI Control Register

The PCI OHCI control register is defined in Section A.3.7 of the *1394 Open Host Controller Interface Specification* and provides a bit for big-endian PCI support. Note that the GLOBAL\_SWAP bit is loaded from the serial EEPROM on powerup.

Offset: 40h  
Default: 0000 0000h  
Type: Read/write

Reference: *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.7

**Table 13. PCI OHCI Control Register Description**

| Bit  | Field Name  | Type | Description  |
|------|-------------|------|--|
| 31:1 | Reserved    | R    | <b>Reserved.</b> Bits 31:1 return 0s when read.  |
| 0    | GLOBAL_SWAP | RW   | When this bit is set, all quadlets read from the FW322 as well as any data written to the PCI bus by the FW322 is byte swapped. This excludes PCI config registers and CardBus function event registers (they are not swapped under any circumstances). However, OHCI registers are byte swapped when this bit is set. |

### 5.15 Capability ID and Next Item Pointer Register

The capability ID and next item pointer register identifies the linked list capability item and provides a pointer to the next capability item.

Offset: 44h  
Default: 0001h  
Type: Read only

Reference: *PCI Local Bus Specification*, Rev. 2.3, Sections 6.8.1.1, 6.8.1.2 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Sections A.3.8.1 and A.3.8.2

**Table 14. Capability ID and Next Item Pointer Register Description**

| Bit  | Field Name    | Type | Description  |
|------|---------------|------|--|
| 15:8 | NEXT_ITEM     | R    | <b>Next Item Pointer.</b> The FW322 supports only one additional capability that is communicated to the system through the extended capabilities list; thus, this field returns 00h when read. |
| 7:0  | CAPABILITY_ID | R    | <b>Capability Identification.</b> This field returns 01h when read, which is the unique ID assigned by the PCI SIG for PCI power management capability.  |

## 5 Internal Registers (continued)

### 5.16 Power Management Capabilities Register

The power management capabilities register indicates the capabilities of the FW322 related to PCI power management. The default value of this register can be selectively programmed by the serial EEPROM. However, the D3cold and AUX\_PWR fields cannot be changed from the default of 0 by the EEPROM.

Offset: 46h  
Default: 7E02h  
Type: Read only

Reference: *PCI Bus Power Management Interface Specification*, Rev. 1.1, Section 3.2.3 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.8.3

**Table 15. Power Management Capabilities Register Description**

| Bit | Field Name | Type | Description  |
|-----|------------|------|--|
| 15  | PME_D3COLD | R    | <b>PME Support from D3cold.</b> Set to 0, indicating that the FW322 cannot generate a PME event in D3cold.   |
| 14  | PME_D3HOT  | R    | <b>PME Support from D3hot.</b> Set to 1, indicating that the FW322 can generate a PME event in the D3hot state.  |
| 13  | PME_D2     | R    | <b>PME Support from D2.</b> Set to 1, indicating that the FW322 can generate a PME in D2.  |
| 12  | PME_D1     | R    | <b>PME Support from D1.</b> Set to 1, indicating that the FW322 can generate a PME in D1.  |
| 11  | PME_D0     | R    | <b>PME Support from D0.</b> Set to 1, indicating that the FW322 can generate a PME in D0.  |
| 10  | D2_SUPPORT | R    | <b>D2 Support.</b> This bit returns a 1 when read, indicating that the FW322 supports the D2 power state.  |
| 9   | D1_SUPPORT | R    | <b>D1 Support.</b> This bit returns a 1 when read, indicating that the FW322 supports the D1 power state.  |
| 8:6 | AUX_PWR    | R    | <b>Auxiliary Power Source.</b> This field reports the Vaux power requirements for the Open HCI function. This field is always 0.   |
| 5   | DSI        | R    | <b>Device-Specific Initialization.</b> This bit returns 0 when read, indicating that the FW322 does not require special initialization beyond the standard PCI configuration header before a generic class driver is able to use it. |
| 4   | Reserved   | R    | <b>Reserved.</b> Bit returns 0 when read.  |
| 3   | PME_CLK    | R    | <b>PME Clock.</b> This bit returns 0 when read, indicating that no host bus clock is required for the FW322 to generate PME.   |
| 2:0 | PM_VERSION | R    | <b>Power Management Version.</b> This field returns 010b when read, indicating that the FW322 is compatible with the registers described in the <i>PCI Power Management Interface Specification</i> , Rev.1.1.                       |

## 5 Internal Registers (continued)

### 5.17 Power Management Control and Status Register

The power management control and status 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 D3hot to D0 state. All bits within this register, will be reset by a PCI reset.

Offset: 48h

Default: XX00h

Type: Read/write

Reference: *PCI Bus Power Management Interface Specification*, Rev. 1.1, Section 3.2.4 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.8.4

**Table 16. Power Management Control and Status Register Description**

| Bit   | Field Name  | Type | Description  |
|-------|-------------|------|--|
| 15    | PME_STS     | RC   | This bit is set when the FW322 would normally be asserting the PME signal, independent of the state of the PME_ENB bit. This bit is cleared by a writeback of 1, and this also clears the PME signal driven by the FW322. Writing a 0 to this bit has no effect.                                     |
| 14:13 | DATA_SCALE  | R    | This 2-bit field indicates a scaling factor that is to be used when interpreting the value of the PM_DATA register within the Power management extension register. The value and meaning of this field will vary depending on the value that has been selected by the DATA_SELECT field.             |
| 12:9  | DATA_SELECT | R    | This 4-bit field is used to select which data values are to be reported through the PM_DATA field in the power management extension register and the DATA_SCALE fields. Valid values are 0—7, which map to power consumption/dissipation ratings for the FW322 within the PM_DATA/DATA_SCALE fields. |
| 8     | PME_ENB     | RW   | <b>PME Enable.</b> This bit enables the function to assert PME. If this bit is cleared, then assertion of PME is disabled.   |
| 7:5   | Reserved    | R    | <b>Reserved.</b> Bits 7:5 return 0s when read.   |
| 4     | DYN_DATA    | R    | <b>Dynamic Data.</b> This bit returns 0 when read, since the FW322 does not report dynamic data.   |
| 3:2   | Reserved    | R    | <b>Reserved.</b> Bits 3:2 return 0s when read.   |
| 1:0   | PWR_STATE   | RW   | <b>Power State.</b> This 2-bit field is used to set the FW322 device power state and is encoded as follows:<br><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.  |

## 5 Internal Registers (continued)

### 5.18 Power Management CSR PCI-to-PCI Bridge Support Extensions

This register returns 00h when read, since the FW322 does not provide PCI-to-PCI bridging.

Offset: 4Ah  
Default: 00h  
Type: Read only

Reference: *PCI Bus Power Management Interface Specification*, Rev. 1.1, Section 3.2.5 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.8.5 and A.3.8.6

### 5.19 Power Management Data

The power management (PM) data register set is comprised of 16 eight-bit registers, providing more detailed power management information about the device. All 16 registers will return 00h by default. The first eight registers are assigned to single function devices, and the second eight are reserved for use by multifunction devices (see Table 17). The FW322 supports programmability, via the serial EEPROM, of the first eight registers in the PM data complex. Software uses the DATA\_SELECT and DATA\_SCALE fields within the power management control and status register to select and scale the desired PM data entry. Note that if the serial EEPROM is used to program nonzero values into PM DATA, then the AUX\_PWR field should be programmed to a zero value via the serial EEPROM and vice versa. This is to comply with the *PCI Specification*, which states that these two functions must be implemented mutually exclusive of one another. The FW322 does **not** enforce this, and therefore, it is up to the creator of the serial EEPROM image to ensure that these two fields are used mutually exclusive of one another.

Offset: 4Bh  
Default: 00h  
Type: Read Only

Reference: *PCI Bus Power Management Interface Specification*, Rev. 1.1, Section 3.2.6 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.8.5 and A.3.8.6

**Table 17. Power Management Data Register Description**

(Derived from Table 10 of the *PCI Power Management Interface Specification*, Revision 1.1.)

| Value in Data_Select | Data Reported  | Data_Scale Interpretation                          | Units/Accuracy |
|----------------------|--|--|----------------|
| 0                    | D0 Power Consumed  | 0 = Unknown<br>1 = 0.1x<br>2 = 0.01x<br>3 = 0.001x | W              |
| 1                    | D1 Power Consumed  |  |                |
| 2                    | D2 Power Consumed  |  |                |
| 3                    | D3 Power Consumed  |  |                |
| 4                    | D0 Power Dissipated                                      |  |                |
| 5                    | D1 Power Dissipated                                      |  |                |
| 6                    | D2 Power Dissipated                                      |  |                |
| 7                    | D3 Power Dissipated                                      |  |                |
| 8—15                 | Reserved (unused by FW322 and will return 00h when read) | Reserved   | TBD            |



## 5 Internal Registers (continued)

### 5.20 OHCI Registers

The OHCI registers defined by the *1394 Open Host Controller Interface Specification* are memory mapped into a 2 Kbyte region of memory pointed to by the OHCI base address register located at offset 10h in PCI configuration space. These registers are the primary interface for controlling the FW322 *IEEE 1394 OHCI* function. This section provides a summary of the registers within this interface and a description of the individual bit fields within each register. For more details regarding these registers and bits, please refer to the *1394 Open Host Controller Interface Specification*, Rev. 1.1.

In addition to regular read/write registers, there are several pairs of set and clear registers implemented within the OHCI register interface. For each pair of set and clear registers, there are two addresses that correspond to individual set/clear registers: RegisterSet and RegisterClear. Refer to Table 18 for a listing of these registers. A 1 bit written to RegisterSet causes the corresponding bit in the register to be set, while a 0 bit leaves the corresponding bit unaffected. A 1 bit written to RegisterClear causes the corresponding bit in the register to be reset, while a 0 bit leaves the corresponding bit unaffected. Typically, a read from either RegisterSet or RegisterClear returns the contents of the set or clear register. However, in some instances, reading the RegisterClear provides a masked version of the set or clear register. The interrupt event register is an example of this behavior.

The following FW322 OHCI register definitions are based on version 1.1 of the *1394 Open Host Controller Specification*.

**Table 18. OHCI Register Map**

| DMA Context | Register Name                 | Abbreviation         | Offset   | OHCI Specification Reference |
|-------------|-------------------------------|----------------------|----------|------------------------------|
| —           | OHCI Version                  | Version              | 00h      | 5.2                          |
|             | Global Unique ID ROM          | GUID_ROM             | 04h      | 5.3                          |
|             | Asynchronous Transmit Retries | ATRetries            | 08h      | 5.4                          |
|             | CSR Data                      | CSRData              | 0Ch      | 5.5.1                        |
|             | CSR Compare Data              | CSRCompareData       | 10h      |                              |
|             | CSR Control                   | CSRControl           | 14h      |                              |
|             | Configuration ROM Header      | ConfigROMhdr         | 18h      | 5.5.2                        |
|             | Bus Identification            | BusID                | 1Ch      | 5.5.3                        |
|             | Bus Options                   | BusOptions           | 20h      | 5.5.4                        |
|             | Global Unique ID High         | GUIDHi               | 24h      | 5.5.5                        |
|             | Global Unique ID Low          | GUIDLo               | 28h      |                              |
|             | Reserved                      | —                    | 2Ch      | —                            |
|             | Reserved                      | —                    | 30h      | —                            |
|             | Configuration ROM Map         | ConfigROMmap         | 34h      | 5.5.6                        |
|             | Posted Write Address Low      | PostedWriteAddressLo | 38h      | 13.2.8.1                     |
|             | Posted Write Address High     | PostedWriteAddressHi | 3Ch      |                              |
|             | Vendor Identification         | VendorID             | 40h      | 5.6                          |
|             | Reserved                      | —                    | Reserved | —                            |
|             | Host Controller Control       | HCCControlSet        | 50h      | 5.7                          |
|             |                               | HCCControlClear      | 54h      |                              |
|             | Reserved                      | —                    | 58h      | —                            |
|             | Reserved                      | —                    | 5Ch      | —                            |
|             | Reserved                      | —                    | 60h      | —                            |

## 5 Internal Registers (continued)

Table 18. OHCI Register Map (continued)

| DMA Context | Register Name                         | Abbreviation                 | Offset    | OHCI Specification Reference |
|-------------|---------------------------------------|------------------------------|-----------|------------------------------|
| SelfID      | SelfID Buffer                         | SelfIDBuffer                 | 64h       | 11.1                         |
|             | SelfID Count                          | SelfIDCount                  | 68h       | 11.2                         |
|             | Reserved                              | —                            | 6Ch       | —                            |
| —           | Isochronous Receive Channel Mask High | IRChannelMaskHiSet           | 70h       | 10.4.1.1                     |
|             |                                       | IRChannelMaskHiClear         | 74h       |                              |
|             | Isochronous Receive Channel Mask Low  | IRChannelMaskLoSet           | 78h       |                              |
|             |                                       | IRChannelMaskLoClear         | 7Ch       |                              |
|             | Interrupt Event                       | IntEventSet                  | 80h       | 6.1                          |
|             |                                       | IntEventClear                | 84h       |                              |
|             | Interrupt Mask                        | IntMaskSet                   | 88h       | 6.2                          |
|             |                                       | IntMaskClear                 | 8Ch       |                              |
|             | Isochronous Transmit Interrupt Event  | IsoXmitIntEventSet           | 90h       | 6.3.1                        |
|             |                                       | IsoXmitIntEventClear         | 94h       |                              |
|             | Isochronous Transmit Interrupt Mask   | IsoXmitIntMaskSet            | 98h       | 6.3.2                        |
|             |                                       | IsoXmitIntMaskClear          | 9Ch       |                              |
| —           | Isochronous Receive Interrupt Event   | IsoRecvIntEventSet           | A0h       | 6.4.1                        |
|             |                                       | IsoRecvIntEventClear         | A4h       |                              |
|             | Isochronous Receive Interrupt Mask    | IsoRecvIntMaskSet            | A8h       | 6.4.2                        |
|             |                                       | IsoRecvIntMaskClear          | ACh       |                              |
|             | Bus Management CSR Initialization     | InitialBandwidthAvailable    | B0h       | 5.8                          |
|             |                                       | InitialChannelsAvailableHi   | B4h       |                              |
|             |                                       | InitialChannelsAvailableLo   | B8h       |                              |
|             | Reserved                              | —                            | BCh:D8h   | —                            |
|             | Fairness Control                      | FairnessControl              | DCh       | 5.9                          |
|             | Link Control                          | LinkControlSet               | E0h       | 5.10                         |
|             |                                       | LinkControlClear             | E4h       |                              |
|             | Node Identification                   | NodeID                       | E8h       | 5.11                         |
|             | PHY Core Layer Control                | PhyControl                   | ECh       | 5.12                         |
|             | Isochronous Cycle Timer               | IsoCycTimer                  | F0h       | 5.13                         |
|             | Reserved                              | —                            | F4h:FCh   | —                            |
|             | Asynchronous Request Filter High      | AsyncRequestFilterHiSet      | 100h      | 5.14.1                       |
|             |                                       | AsyncRequestFilterHiClear    | 104h      |                              |
|             | Asynchronous Request Filter Low       | AsyncRequestFilterLoSet      | 108h      |                              |
|             |                                       | AsyncRequestFilterLoClear    | 10Ch      |                              |
|             | Physical Request Filter High          | PhysicalRequestFilterHiSet   | 110h      | 5.4.2                        |
|             |                                       | PhysicalRequestFilterHiClear | 114h      |                              |
|             | Physical Request Filter Low           | PhysicalRequestFilterLoSet   | 118h      |                              |
|             |                                       | PhysicalRequestFilterLoClear | 11Ch      |                              |
|             | Physical Upper Bound                  | PhysicalUpperBound           | 120h      | 5.15                         |
|             | Reserved                              | —                            | 124h:17Ch | —                            |

## 5 Internal Registers (continued)

Table 18. OHCI Register Map (continued)

| DMA Context                               | Register Name   | Abbreviation        | Offset                        | OHCI Specification Reference |
|---|-----------------|---------------------|-------------------------------|------------------------------|
| Asynchronous Request Transmit [ATRQ]      | Context Control | ContextControlSet   | 180h                          | 3.1, 7.2.2                   |
|   |                 | ContextControlClear | 184h                          |                              |
|   | Reserved        | —                   | 188h                          | —                            |
|   | Command Pointer | CommandPtr          | 18Ch                          | 3.1.2, 7.2.1                 |
| Asynchronous Response Transmit [ATRS]     | Reserved        | —                   | 190h:19Ch                     | —                            |
|   | Context Control | ContextControlSet   | 1A0h                          | 3.1, 7.2.2                   |
|   |                 | ContextControlClear | 1A4h                          |                              |
|   | Reserved        | —                   | 1A8h                          | —                            |
|   | Command Pointer | CommandPtr          | 1ACh                          | 3.1.2, 7.2.1                 |
| Asynchronous Request Receive [ARRQ]       | Reserved        | —                   | 1B0h:1BCh                     | —                            |
|   | Context Control | ContextControlSet   | 1C0h                          | 3.1, 8.3.2                   |
|   |                 | ContextControlClear | 1C4h                          |                              |
|   | Reserved        | —                   | 1C8h                          | —                            |
|   | Command Pointer | CommandPtr          | 1CCh                          | 3.1.2, 8.3.1                 |
| Asynchronous Response Receive [ARRS]      | Reserved        | —                   | 1D0h:1DFh                     | —                            |
|   | Context Control | ContextControlSet   | 1E0h                          | 3.1, 8.3.2                   |
|   |                 | ContextControlClear | 1E4h                          |                              |
|   | Reserved        | —                   | 1E8h                          | —                            |
|   | Command Pointer | CommandPtr          | 1ECh                          | 3.1.2, 8.3.1                 |
| Isochronous Transmit Context n<br>n = 0:7 | Reserved        | —                   | 1F0h:1FFh                     | —                            |
|   | Context Control | ContextControlSet   | 200h + 16 * n                 | 3.1, 9.2.2                   |
|   |                 | ContextControlClear | 204h + 16 * n                 |                              |
|   | Reserved        | —                   | 208h + 16 * n                 | —                            |
|   | Command Pointer | CommandPtr          | 20Ch + 16 * n                 | 3.1.2, 9.2.1                 |
| Isochronous Receive Context n<br>n = 0:7  | Context Control | ContextControlSet   | 400h + 32 * n                 | 3.1, 10.3.2                  |
|   |                 | ContextControlClear | 404h + 32 * n                 |                              |
|   | Reserved        | —                   | 408h + 32 * n                 | —                            |
|   | Command Pointer | CommandPtr          | 40Ch + 32 * n                 | 3.1.2, 10.3.1                |
|   | Context Match   | ContextMatch        | 410h + 32 * n                 | 10.3.3                       |
|   | Reserved        | —                   | 414h + 32 * n – 41Ch + 32 * n | —                            |

## 5 Internal Registers (continued)

### 5.21 OHCI Version Register

This register indicates the OHCI version supported, and whether or not the serial EEPROM is present. To support backwards compatibility with existing hardware and software, the version and revision fields default to 8'h01 and 8'h00 respectively. These values denote compatibility with version 1.0 of the OHCI specification. However, both the version and revision fields are programmable via the serial EEPROM. This functionality allows these fields to be optionally updated to 8'h01 and 8'h10 respectively to indicate compatibility with version 1.1 of the OHCI specification. Note that if the version and revision fields are programmed with OHCI 1.1 values, then the LinkOptions register (see Table 60) should also be programmed to properly enable OHCI 1.1 features within the FW322.

Offset: 00h  
Default: 0X01 0000h  
Type: Read/write

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.2

**Table 19. OHCI Version Register Description**

| Bit   | Field Name | Type | Description   |
|-------|------------|------|---|
| 31:25 | Reserved   | R    | <b>Reserved.</b>  |
| 24    | GUID_ROM   | R    | The FW322 sets this bit if the serial EEPROM is detected.   |
| 23:16 | Version    | R    | <b>Major Version of the OHCI.</b> The FW322 is compliant with both version 1.0 and version 1.1 of the <i>1394 Open Host Controller Interface Specification</i> . This field defaults to 01h, but can be reconfigured via the serial EEPROM. |
| 15:8  | Reserved   | R    | <b>Reserved.</b>  |
| 7:0   | Revision   | R    | <b>Minor Version of the OHCI.</b> The FW322 is compliant with both version 1.0 and version 1.1 of the <i>1394 Open Host Controller Interface Specification</i> . This field defaults to 00h, but can be reconfigured via the serial EEPROM. |

## 5 Internal Registers (continued)

### 5.22 GUID ROM Register

The GUID ROM register is used to access the serial EEPROM, and is only applicable if bit 24 (GUID\_ROM) in the OHCI version register is set.

Offset: 04h

Default: 00XX 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.3

**Table 20. GUID ROM Register Description**

| Bit   | Field Name | Type | Description   |
|-------|------------|------|---|
| 31    | addrReset  | RWU  | Software sets this bit to reset the GUID ROM address to 0. When the FW322 completes the reset, it clears this bit.  |
| 30:26 | Reserved   | R    | <b>Reserved.</b> Bits 30:26 return 0s when read.  |
| 25    | rdStart    | RWU  | A read of the currently addressed byte is started when this bit is set. This bit is automatically cleared when the FW322 completes the read of the currently addressed GUID ROM byte. |
| 24    | Reserved   | R    | <b>Reserved.</b> Bit 24 returns 0 when read.  |
| 23:16 | rdData     | RU   | This field represents the data read from the GUID ROM and is only valid when rdStart = 0.   |
| 15:8  | Reserved   | R    | <b>Reserved.</b> Bits 15:8 return 0s when read.   |
| 7:0   | miniROM    | R    | Indicates the first byte location of the miniROM image in the GUID ROM. A value of 0 is returned if no miniROM is implemented.  |

### 5.23 Asynchronous Transmit Retries Register

The asynchronous transmit retries register indicates the number of times the FW322 attempts a retry for asynchronous DMA request transmit and for asynchronous physical and DMA response transmit.

Offset: 08h

Default: 0000 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.4

**Table 21. Asynchronous Transmit Retries Register Description**

| Bit   | Field Name         | Type | Description   |
|-------|--------------------|------|---|
| 31:29 | secondLimit        | R    | The second limit field returns 0s when read, since outbound dual-phase retry is not implemented.  |
| 28:16 | cycleLimit         | R    | The cycle limit field returns 0s when read, since outbound dual-phase retry is not implemented.   |
| 15:12 | Reserved           | R    | <b>Reserved.</b> Bits 15:12 return 0s when read.  |
| 11:8  | maxPhysRespRetries | RW   | This field tells the physical response unit how many times to attempt to retry the transmit operation for the response.                         |
| 7:4   | maxATRespRetries   | RW   | This field tells the asynchronous transmit <b>DMA</b> response unit how many times to attempt to retry the transmit operation for the response. |
| 3:0   | maxATReqRetries    | RW   | This field tells the asynchronous transmit DMA request unit how many times to attempt to retry the transmit operation for the response.         |

## 5 Internal Registers (continued)

### 5.24 CSR Data Register

The CSR data 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.

Offset: 0Ch  
Default: XXXX XXXXh

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Sections 5.5.1.

**Table 22. CSR Data Register Description**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 31:0 | csrData    | RWU  | At start of operation, the data to be stored if the compare is successful. |

### 5.25 CSR Compare Register

The CSR compare 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.

Offset: 10h  
Default: XXXX XXXXh

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.5.1

**Table 23. CSR Compare Register Description**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 31:0 | csrCompare | RW   | The data to be compared with the existing value of the CSR resource. |

### 5.26 CSR Control Register

The CSR control register is used to access the bus management CSR registers from the host through compare-swap operations. Bits in this register are used to initiate a compare-and-swap operation on a selected resource and signal when that operation is complete.

Offset: 14h  
Default: 8000 000Xh

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.5.1

**Table 24. CSR Control Register Description**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 31   | csrDone    | RU   | This bit is set by the FW322 when a compare-swap operation is complete. It is reset whenever this register is written.   |
| 30:2 | Reserved   | R    | <b>Reserved.</b> Bits 30:2 return 0s when read.  |
| 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 |

## 5 Internal Registers (continued)

### 5.27 Configuration ROM Header Register

The configuration ROM header register externally maps to the first quadlet of the 1394 configuration ROM, offset 48'hFFFF\_F000\_0400.

Offset: 18h

Default: 0000 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.5.2

**Table 25. Configuration ROM Header Register Description**

| Bit   | Field Name    | Type | Description   |
|-------|---------------|------|---|
| 31:24 | info_length   | RW   | <b>IEEE 1394 Bus Management Field.</b> Must be valid when bit 17 (linkEnable) of the host controller control register is set (see Table 34).        |
| 23:16 | crc_length    | RW   | <b>IEEE 1394 Bus Management Field.</b> Must be valid when bit 17 (linkEnable) of the host controller control register is set (see Table 34).        |
| 15:0  | rom_crc_value | RW   | <b>IEEE 1394 Bus Management Field.</b> Must be valid at any time bit 17 (linkEnable) of the host controller control register is set (see Table 34). |

## 5 Internal Registers (continued)

### 5.28 Bus Identification Register

The bus identification register externally maps to the first quadlet in the Bus\_Info\_Block and is addressable at FFFF\_F000\_0404. This register is read locally at the offset specified below.

Offset: 1Ch  
Default: 3133 3934h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.5.3

**Table 26. Bus Identification Register Description**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 31—0 | busID      | R    | Contains the constant 32'h31333934, which is the ASCII value for 1394. |

### 5.29 Bus Options Register

The Bus options register externally maps to the second quadlet of the Bus\_Info\_Block and is 1394 addressable at FFFF\_F000\_0408.

Offset: 20h  
Default: 0000 A002h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.5.4

**Table 27. Bus Options Register Description**

| Bit   | Field Name | Type    | Description  |
|-------|------------|---------|--|
| 31:16 | Reserved   | R or RW | <b>Reserved.</b> Bits return 0s when read; 23:16 and 31:27 are RW and undefined.   |
| 15:12 | max_rec    | RW      | <b>IEEE 1394 Bus Management Field.</b> Hardware initializes this field 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 or greater and is calculated by $2^{(\text{max\_rec} + 1)}$ . Software may change this field; however, this field must be valid at any time bit 17 (linkEnable) of the host controller control register is set. A received block write request packet with a length greater 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 a hard reset. |
| 11:3  | Reserved   | R       | <b>Reserved.</b> Bits 11:3 return 0s when read.  |
| 2:0   | Lnk_spd    | R       | <b>Link Speed.</b> This field returns 010, indicating that the link speeds of 100 Mbits/s, 200 Mbits/s, and 400 Mbits/s are supported.   |



## 5 Internal Registers (continued)

### 5.30 GUID High Register

The GUID high register represents the upper quadlet in a 64-bit global unique ID (GUID), which maps to the third quadlet in the Bus\_Info\_Block 1394, addressable at FFFF\_F000\_0410. This register contains node\_vendor\_ID and chip\_ID\_hi fields. This register initializes to 0s on a hardware reset, which is an illegal GUID value. If a serial EEPROM is detected, then the contents of this register are loaded through the serial EEPROM interface after a PCI reset. If no serial EEPROM is detected, then the contents of this register can be loaded with a single PCI write to either of two configuration registers, executed after a PCI reset. The two configuration registers are located at offset 0x70, for new PCI applications, and offset 0x80, for backward compatibility with FW322 05 PCI applications only. After one of these load mechanisms has completed, this register becomes read only.

Offset: 24h

Default: 0000 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.5.5

**Table 28. GUID High Register Description**

| Bit  | Field Name     | Type | Description   |
|------|----------------|------|---|
| 31:8 | node_vendor_ID | RWU  | <b>IEEE 1394 Bus Management Fields.</b> Firmware or hardware must ensure that this register is valid whenever HCCControl.linkEnable bit is set. |
| 7:0  | chip_ID_hi     | RWU  | Firmware or hardware must ensure that this register is valid whenever HCCControl.linkEnable bit is set.   |

### 5.31 GUID Low Register

The GUID low register represents the lower quadlet in a 64-bit global unique ID (GUID), which maps to chip\_ID\_lo in the Bus\_Info\_Block 1394, addressable at FFFF\_F000\_0414. This register initializes to 0s on a hardware reset and behaves identical to the GUID high register. If no serial EEPROM is detected, then the contents of this register can be loaded with a PCI configuration write to either offset 0x74 or 0x84, as described above.

Offset: 28h

Default: 0000 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.5.5

**Table 29. GUID Low Register Description**

| Bit  | Field Name | Type | Description   |
|------|------------|------|---|
| 31:0 | chip_ID_lo | R    | <b>IEEE 1394 Bus Management Fields.</b> Firmware or hardware must ensure that this register is valid whenever HCCControl.linkEnable bit is set. |

## 5 Internal Registers (continued)

### 5.32 Configuration ROM Mapping Register

The configuration ROM mapping register contains the start address within system memory that maps to the start address of 1394 configuration ROM for this node.

Offset: 34h  
Default: 0000 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.5.6

**Table 30. Configuration ROM Mapping Register Description**

| Bit   | Field Name    | Type | 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. |
| 9:0   | Reserved      | R    | <b>Reserved.</b> Bits 9:0 return 0s when read.   |

## 5 Internal Registers (continued)

### 5.33 Posted Write Address Low Register

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

Offset: 38h  
Default: XXXX XXXXh

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 13.2.8.1.

**Table 31. Posted Write Address Low Register Description**

| Bit  | Field Name | Type | Description   |
|------|------------|------|---|
| 31:0 | offsetLo   | RU   | The lower 32 bits of the 1394 destination offset of the write request that was posted and failed. |

### 5.34 Posted Write Address High Register

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

Offset: 3Ch  
Default: XXXX XXXXh

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 13.2.8.1.

**Table 32. Posted Write Address High Register Description**

| Bit   | Field Name | Type | Description   |
|-------|------------|------|---|
| 31:16 | sourceID   | RU   | This field is the bus and node number of the node that issued the write request that was posted and failed. |
| 15:0  | offsetHi   | RU   | The upper 16 bits of the 1394 destination offset of the write request that was posted and failed.           |

### 5.35 Vendor ID Register

The vendor ID register holds the company ID of an organization that specifies any vendor-unique registers.

Offset: 40h  
Default: 0000 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.6

**Table 33. Vendor ID Register Description**

| Bit   | Field Name      | Type | Description  |
|-------|-----------------|------|--|
| 31:24 | vendorUnique    | R    | Returns 0 when read, since the FW322 does not specify any vendor-unique registers. |
| 23:0  | vendorCompanyID | R    | Returns 0 when read, since the FW322 does not specify any vendor-unique registers. |

## 5 Internal Registers (continued)

### 5.36 Host Controller Control Register

The host controller control set/clear register pair provides flags for controlling the OHCI portion of the FW322.

Offset: 50h set register  
54h clear register  
Default: X08X 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 5.7

**Table 34. Host Controller Control Register Description**

| Bit   | Field Name        | Type | Description   |
|-------|-------------------|------|---|
| 31    | BIBimageValid     | RSU  | This bit is used to enable both OHCI response to block read requests to host configuration ROM and the OHCI mechanism for automatically updating configuration ROM. When this bit is 0, the OHCI returns a <code>ack_type_error</code> on block read requests to configuration ROM and does not update the <code>configROMmap</code> register or <code>ConfigROMheader</code> and <code>BusOptions</code> registers when a 1394 bus reset occurs. When this bit is 1, the physical response unit handles block reads of host configuration ROM and the mechanism for automatically updating configuration ROM is enabled.   |
| 30    | noByteSwapData    | RSC  | This bit is used to control byte swapping during host bus accesses involving the data portion of 1394 packets. Data is swapped if equal to 0, not swapped when equal to 1.  |
| 29    | ackTardyEnable    | RSC  | This bit is used to control the acknowledgment of <code>ack_tardy</code> . When this bit is set to one, <code>ack_tardy</code> may be returned as an acknowledgement to configuration ROM accesses from 1394 to OHCI including accesses to the <code>bus_info_block</code> . The host controller will return <code>ack_tardy</code> to all other asynchronous packets addressed to the OHCI node.   |
| 28:24 | Reserved          | R    | <b>Reserved.</b> Bits 28:24 return 0s when read.  |
| 23    | programPhyEnable  | RC   | This bit informs upper-level software that lower-level software has consistently configured the 1394a-2000 enhancements in the link and PHY core. When this bit is 1, generic software such as the OHCI driver is responsible for configuring 1394a-2000 enhancements in the PHY core and bit 22 ( <code>aPhyEnhanceEnable</code> ) in the FW322. When this bit is 0, the generic software may not modify the 1394a-2000 enhancements in the FW322 and cannot interpret the setting of bit 22 ( <code>aPhyEnhanceEnable</code> ). When a serial EEPROM is present, this bit is initialized from the EEPROM. Otherwise, this bit is initialized to the hardware default value. |
| 22    | aPhyEnhanceEnable | RSC  | When bits 23 ( <code>programPhyEnable</code> ) is 1 and 17 ( <code>linkEnable</code> ) is 0, the OHCI driver can set this bit to use all 1394a-2000 enhancements. When bit 23 ( <code>programPhyEnable</code> ) is set to 0, the software does not change PHY enhancements or this bit.   |
| 21:20 | Reserved          | R    | <b>Reserved.</b> Bits 21:20 return 0s when read.  |
| 19    | LPS               | RSU  | <b>Link Power Status.</b> This bit drives the LPS signal to the PHY core within the FW322 (see Section 5.7 of the <i>OHCI 1.1 Specification</i> for additional details).  |
| 18    | postedWriteEnable | RSC  | This bit is used to enable (1) or disable (0) posted writes. Software should change this bit only when bit 17 ( <code>linkEnable</code> ) is 0.   |

## 5 Internal Registers (continued)

Table 34. Host Controller Control Register Description (continued)

| Bit  | Field Name | Type | Description   |
|------|------------|------|---|
| 17   | linkEnable | RSU  | This bit is cleared to 0 by either a hardware 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 cleared, the FW322 is logically and immediately disconnected from the 1394 bus, no packets are received or processed, and no packets transmitted. |
| 16   | SoftReset  | RSU  | When this bit is set, all FW322 states are reset, all FIFOs 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 while the softReset is in progress and reverts back to 0 when the reset has completed.  |
| 15:0 | Reserved   | R    | <b>Reserved.</b> Bits 15:0 return 0s when read.   |

## 5 Internal Registers (continued)

### 5.37 SelfID Buffer Pointer Register

The SelfID buffer pointer register points to the 2 Kbyte aligned base address of the buffer in host memory where the SelfID packets are stored during bus initialization. Bits 31:11 are read/write accessible.

Offset: 64h  
Default: XXXX XX00h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 11.1

**Table 35. SelfID Buffer Pointer Register Description**

| Bit   | Field Name      | Type | Description  |
|-------|-----------------|------|--|
| 31:11 | SelfIDBufferPtr | RW   | Contains the 2 Kbyte aligned base address of the buffer in host memory where received SelfID packets are stored. |
| 10:0  | Reserved        | R    | Reserved.  |

### 5.38 SelfID Count Register

The SelfID count register keeps a count of the number of times the SelfID process has occurred. The register also flags any SelfID errors and maintains a count of the amount of SelfID data in the SelfID buffer.

Offset: 68h  
Default: X0XX 0000h

Reference: 1394 Open Host Controller Interface Specification, Rev. 1.1, Section 11.2

**Table 36. SelfID Count Register Description**

| Bit   | Field Name       | Type | Description  |
|-------|------------------|------|--|
| 31    | selfIDError      | RU   | When this bit is 1, an error was detected during the most recent SelfID packet reception. The contents of the SelfID buffer are undefined. This bit is cleared after a SelfID reception in which no errors are detected. Note that an error can be a hardware error or a host bus write error. |
| 30:24 | Reserved         | R    | <b>Reserved.</b> Bits 30:24 return 0s when read.   |
| 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.   |
| 15:11 | Reserved         | R    | <b>Reserved.</b> Bits 15:11 return 0s when read.   |
| 10:2  | selfIDSize       | RU   | This field indicates the number of quadlets that have been written into the SelfID buffer for the current bits 23:16 (SelfIDGeneration field). This includes the header quadlet and the SelfID data. This field is cleared to 0 when the SelfID reception begins.                              |
| 1:0   | Reserved         | R    | <b>Reserved.</b> Bits 1:0 return 0s when read.   |

## 5 Internal Registers (continued)

### 5.39 Isochronous Receive Multiple Channel Mask High (IRMultiChanMaskHi) Register

The isochronous receive multiple channel mask high 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 content of the isochronous receive multiple channel mask high register.

Offset:           70h                   set register  
                  74h                   clear register  
Default:        XXXX XXXXh

Reference:      1394 Open Host Controller Interface Specification, Rev. 1.1, Section 10.4.1.1

**Table 37. Isochronous Receive Channel Mask High Register Description**

| Bit  | Field Name         | Type | Description  |
|------|--------------------|------|--|
| 31:0 | isoChannel(N + 32) | RSC  | If bit N (where N = a bit number 0—31) is set, iso channel number (N + 32) is enabled. |

### 5.40 Isochronous Receive Multiple Channel Mask Low (IRMultiChanMaskLo) Register

The isochronous receive channel mask low set/clear register is used to enable packet receives from the lower 32 isochronous data channels.

Offset:           78h                   set register  
                  7Ch                   clear register  
Default:        XXXX XXXXh

Reference:      1394 Open Host Controller Interface Specification, Rev. 1.1, Section 10.4.1.

**Table 38. Isochronous Receive Channel Mask Low Register Description**

| Bit  | Field Name   | Type | Description   |
|------|--------------|------|---|
| 31:0 | isoChannel N | RSC  | If bit N (where N = a bit number 0—31) is set, iso channel number N is enabled. |

## 5 Internal Registers (continued)

### 5.41 Interrupt Event (IntEvent) Register

The interrupt event set/clear register reflects the state of the various FW322 interrupt sources. The interrupt bits are set by an asserting edge of the corresponding interrupt signal or by writing a 1 in the corresponding bit in the set register. The only mechanism to clear the bits in this register is to write a 1 to the corresponding bit in the clear register. Reading the IntEventSet register returns the current state of the IntEvent register. Reading the IntEventClear register returns the **masked** version of the IntEvent register, i.e., the bit-wise AND function of IntEvent and IntMask.

Offset:           80h               set register  
                  84h               clear register

Default:       XXXX 0XXXh

Reference:      1394 Open Host Controller Interface Specification, Rev. 1.1, Section 6.1.

**Table 39. Interrupt Event Register Description**

| Bit | Field Name         | Type | Description   |
|-----|--------------------|------|---|
| 31  | Reserved           | R    | <b>Reserved.</b> Bit 31 returns 0 when read.  |
| 30  | vendorSpecific     | RSCU | This vendor-specific interrupt event is reported when serial EEPROM read is complete.   |
| 29  | SoftInterrupt      | RSC  | <b>Soft Interrupt.</b> This bit may be used by software to generate a host controller interrupt for its own use.  |
| 28  | Reserved           | R    | <b>Reserved.</b> Bit 28 returns 0 when read.  |
| 27  | ack_Tardy          | RSCU | This bit will be set when the ackTardyEnable bit of the HC control register (see Table 34) is set to 1 and any of the following conditions occur: <ul style="list-style-type: none"> <li>■ Data is present in a FIFO that is to be delivered to the host.</li> <li>■ The physical response unit is busy processing requests or sending responses.</li> <li>■ The host controller sent an ack_tardy acknowledgment.</li> </ul>                       |
| 26  | phyRegRcvd         | RSCU | The FW322 has received a PHY core register data byte, which can be read from the PHY core layer control register.   |
| 25  | cycleTooLong       | RSCU | If bit 21 (cycleMaster) of the link control register (see Table 45) is set, then this indicates that over 125 $\mu$ s have elapsed between the start of sending a cycle start packet and the end of a subaction gap. The link control register bit 21 (cycleMaster) is cleared by this event.   |
| 24  | unrecoverableError | RSCU | This event occurs when the FW322 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 this bit is set, all normal interrupts for the context(s) that caused this interrupt are blocked from being set.  |
| 23  | cycleInconsistent  | RSCU | A cycle start was received that had values for cycleSeconds and cycleCount fields that are different from the values in bits 31:25 (cycleSeconds field) and bits 24:12 (cycleCount field) of the isochronous cycle timer register (see Table 48).   |
| 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. This bit may be set either when it occurs or when logic predicts that it will occur. |



## 5 Internal Registers (continued)

Table 39. Interrupt Event Register Description (continued)

| Bit   | Field Name      | Type | Description  |
|-------|-----------------|------|--|
| 21    | cycle64Seconds  | RSCU | Indicates that the seventh bit of the cycleSeconds (see Table 48) counter has changed.   |
| 20    | cycleSynch      | RSCU | Indicates that a new isochronous cycle has started. This bit is set when the low-order bit of the cycleCount (see Table 48) toggles.   |
| 19    | PHY             | RSCU | Indicates the PHY core requests an interrupt through a status transfer.  |
| 18    | regAccessFail   | RSCU | Indicates that an OHCI register access failed due to a missing SCLK clock signal from the PHY. When a register access fails, this bit will be set before the next register access.   |
| 17    | busReset        | RSCU | Indicates that the PHY core chip has entered bus reset mode.   |
| 16    | selfIDcomplete  | RSCU | A SelfID packet stream has been received. It is generated at the end of the bus initialization process. This bit is turned off simultaneously when bit 17 (busReset) is turned on.   |
| 15    | SelfIDcomplete2 | RSCU | Secondary indication of the end of a SelfID packet stream. This bit will be set by the OHCI when it sets SelfIDcomplete, and will retain state independent of the busReset bit of this register.   |
| 14:10 | Reserved        | R    | <b>Reserved.</b> Bits 14:10 return 0s when read.   |
| 9     | lockRespErr     | RSCU | Indicates that the FW322 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 FW322 was trying to write a 1394 write request, which had already been given an ack_complete, into system memory.   |
| 7     | isochRx         | RU   | <b>Isochronous Receive DMA Interrupt.</b> Indicates that one or more isochronous receive contexts have generated an interrupt. This is not a latched event; it is the <b>ORing</b> of all bits in the isochronous receive interrupt event and isochronous receive interrupt mask registers. The isochronous receive interrupt event register (see Table 43) indicates which contexts have interrupted.                     |
| 6     | isochTx         | RU   | <b>Isochronous Transmit DMA Interrupt.</b> Indicates that one or more isochronous transmit contexts have generated an interrupt. This is not a latched event; it is the <b>ORing</b> of all bits in the isochronous transmit interrupt event (see Table 41) and isochronous transmit interrupt mask (see Table 42) registers. The isochronous transmit interrupt event 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 | <b>Asynchronous Receive Response DMA Interrupt.</b> This bit is conditionally set upon completion of an ARRS DMA context command descriptor.   |
| 2     | ARRQ            | RSCU | <b>Asynchronous Receive Request DMA Interrupt.</b> This bit is conditionally set upon completion of an ARRQ DMA context command descriptor.  |
| 1     | respTxComplete  | RSCU | <b>Asynchronous Response Transmit DMA Interrupt.</b> This bit is conditionally set upon completion of an ATRS DMA command.   |
| 0     | reqTxComplete   | RSCU | <b>Asynchronous Request Transmit DMA Interrupt.</b> This bit is conditionally set upon completion of an ATRQ DMA command.  |

## 5 Internal Registers (continued)

### 5.42 Interrupt Mask (IntMask) Register

The Interrupt Mask set/clear register is used to enable/disable the various FW322 interrupt sources. Reads from either the set register or the clear register always return the contents of the interrupt mask register. In all cases except masterIntEnable (bit 31), the enables for each interrupt event align with the Interrupt Event (IntEvent) register bits (see Table 39). A **one** bit in the IntMask register enables the corresponding IntEvent register bit to generate a processor interrupt. A **zero** bit in IntMask disables the corresponding IntEvent register bit from generating a processor interrupt. A bit is set in the IntMask register by writing a one to the corresponding bit in the IntMaskSet address and cleared by writing a one to the corresponding bit in the IntMaskClear address.

Offset:           88h               set register  
                  8Ch               clear register  
Default:       XXXX 0XXXh

Reference:       1394 Open Host Controller Interface Specification, Rev. 1.1, Section 6.2.

**Table 40. Interrupt Mask Register Description**

| Bit   | Field Name         | Type | Description   |
|-------|--------------------|------|---|
| 31    | masterIntEnable    | RSCU | <b>Master Interrupt Enable.</b> If this bit is set, then external interrupts are generated in accordance with the Interrupt Mask register. If this bit is cleared, then external interrupts are not generated, regardless of the Interrupt Mask register settings. The value of masterIntEnable has no effect on the value returned by reading the IntEventClear. |
| 30    | vendorSpecific     | RSC  | When this bit is set, this vendor-specific interrupt mask enables interrupt generation when bit 30 (vendorSpecific) of the Interrupt Event register (Table 39) is set.  |
| 29    | softInterrupt      | RSC  | <b>Soft Interrupt.</b> This bit may be used by software to generate a host controller interrupt for its own use. When set, this bit enables the corresponding IntEvent register bit to generate a processor interrupt.  |
| 28    | Reserved           | R    | <b>Reserved.</b> Bit 28 returns 0 when read.  |
| 27    | ack_Tardy          | RSCU | A <b>one</b> bit enables the corresponding IntEvent register bit to generate a processor interrupt. A <b>zero</b> bit disables the corresponding IntEvent register bit from generating a processor interrupt.   |
| 26    | phyRegRcvd         |      |   |
| 25    | cycleTooLong       |      |   |
| 24    | unrecoverableError |      |   |
| 23    | cycleInconsistent  |      |   |
| 22    | cycleLost          |      |   |
| 21    | cycle64Seconds     |      |   |
| 20    | cycleSynch         |      |   |
| 19    | PHY                |      |   |
| 18    | regAccessFail      |      |   |
| 17    | busReset           |      |   |
| 16    | selfIDcomplete     |      |   |
| 15    | SelfIDcomplete2    |      |   |
| 14:10 | Reserved           | R    | <b>Reserved.</b> Bits 14:10 return 0s when read.  |

## 5 Internal Registers (continued)

**Table 40. Interrupt Mask Register Description** (continued)

| Bit | Field Name     | Type | Description   |
|-----|----------------|------|---|
| 9   | lockRespErr    | RSCU | When set, these bits enable the corresponding IntEvent register bits to generate a processor interrupt. |
| 8   | postedWriteErr |      |   |
| 7   | isochRx        |      |   |
| 6   | isochTx        |      |   |
| 5   | RSPkt          |      |   |
| 4   | RQPkt          |      |   |
| 3   | ARRS           |      |   |
| 2   | ARRQ           |      |   |
| 1   | respTxComplete |      |   |
| 0   | reqTxComplete  |      |   |

## 5 Internal Registers (continued)

### 5.43 Isochronous Transmit Interrupt Event (isoXmitIntMask) Register

The isochronous transmit interrupt event (isoXmitIntMask) 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 interrupt event register isoTx (bit 6) (see Table 39) 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 1 in the corresponding bit in the set register. The only mechanism to clear the bits in this register is to write a 1 to the corresponding bit in the clear register.

Reading the isoXmitIntEventSet register returns the current state of the isoXmitIntEvent register. Reading the isoXmitIntEventClear register returns the **masked** version of the isoXmitIntEvent register, i.e., the bit-wise AND function of **isoXmitIntEvent** and **isoXmitIntMask**.

Offset:           90h               set register  
                  94h               clear register  
Default:        0000 00XXh

Reference:      1394 Open Host Controller Specification, Rev. 1.1, Section 6.3

**Table 41. Isochronous Transmit Interrupt Event Register Description**

| Bit  | Field Name | Type | Description   |
|------|------------|------|---|
| 31:8 | Reserved   | R    | <b>Reserved.</b> Bits 31:8 return 0s when read.   |
| 7    | isoXmit7   | RSCU | Isochronous transmit channel 7 caused the interrupt event register bit 6 (isoTx) interrupt. |
| 6    | isoXmit6   | RSCU | Isochronous transmit channel 6 caused the interrupt event register bit 6 (isoTx) interrupt. |
| 5    | isoXmit5   | RSCU | Isochronous transmit channel 5 caused the interrupt event register bit 6 (isoTx) interrupt. |
| 4    | isoXmit4   | RSCU | Isochronous transmit channel 4 caused the interrupt event register bit 6 (isoTx) interrupt. |
| 3    | isoXmit3   | RSCU | Isochronous transmit channel 3 caused the interrupt event register bit 6 (isoTx) interrupt. |
| 2    | isoXmit2   | RSCU | Isochronous transmit channel 2 caused the interrupt event register bit 6 (isoTx) interrupt. |
| 1    | isoXmit1   | RSCU | Isochronous transmit channel 1 caused the interrupt event register bit 6 (isoTx) interrupt. |
| 0    | isoXmit0   | RSCU | Isochronous transmit channel 0 caused the interrupt event register bit 6 (isoTx) interrupt. |

## 5 Internal Registers (continued)

### 5.44 Isochronous Transmit Interrupt Mask (isoXmitIntMask) Register

The isochronous transmit interrupt mask 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 the contents of the isochronous transmit interrupt mask register. In all cases, the enables for each interrupt event align with the event register bits detailed in Table 42.

Offset:            98h                    set register  
                     9Ch                    clear register (returns IsoXmitEvent and IsoXmitMask when read)  
Default:           0000 00XXh  
Reference:        *1394 Open Host Controller Specification*, Rev. 1.1, Section 6.3

**Table 42. Isochronous Transmit Interrupt Event Description**

| Bit  | Field Name        | Type | Description   |
|------|-------------------|------|---|
| 31:8 | Reserved          | R    | <b>Reserved.</b> Bits 31:8 return 0s when read.   |
| 7:0  | isoXmit7:isoXmit0 | RSCU | Setting one of these bits enables the corresponding interrupt event in the isoXmitIntEvent register. Clearing a bit in this register disables the corresponding interrupt event in the isoXmitInt-Event register. |

## 5 Internal Registers (continued)

### 5.45 Isochronous Receive Interrupt Event (isoRecvIntEvent) Register

The isochronous receive interrupt event 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 interrupt event register isoChRx (bit 7) 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 1 in the corresponding bit in the set register. The only mechanism to clear the bits in this register is to write a 1 to the corresponding bit in the clear register.

The isoRecvIntMask register is ANDed with the isoRecvIntEvent register to enable selected bits to generate processor interrupts. Reading the isoRecvIntEventSet register returns the current state of the isoRecvIntEvent register. Reading isoRecvIntEventClear register returns the **masked** version of the isoRecvIntEvent register, i.e., the bit-wise AND function of **isoRecvIntEvent** and **isoRecvIntMask**.

Offset:           A0h               set register  
                  A4h               clear register  
Default:        0000 0000h

Reference:      1394 Open Host Controller Specification, Rev. 1.1, Section 6.4

**Table 43. Isochronous Receive Interrupt Event Description**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 31:8 | Reserved   | R    | <b>Reserved.</b> Bits 31:8 return 0s when read.  |
| 7    | isoRecv7   | RSCU | Isochronous receive context 7 caused the interrupt event register bit 7 (isoChRx) interrupt. |
| 6    | isoRecv6   | RSCU | Isochronous receive context 6 caused the interrupt event register bit 7 (isoChRx) interrupt. |
| 5    | isoRecv5   | RSCU | Isochronous receive context 5 caused the interrupt event register bit 7 (isoChRx) interrupt. |
| 4    | isoRecv4   | RSCU | Isochronous receive context 4 caused the interrupt event register bit 7 (isoChRx) interrupt. |
| 3    | isoRecv3   | RSCU | Isochronous receive context 3 caused the interrupt event register bit 7 (isoChRx) interrupt. |
| 2    | isoRecv2   | RSCU | Isochronous receive context 2 caused the interrupt event register bit 7 (isoChRx) interrupt. |
| 1    | isoRecv1   | RSCU | Isochronous receive context 1 caused the interrupt event register bit 7 (isoChRx) interrupt. |
| 0    | isoRecv0   | RSCU | Isochronous receive context 0 caused the interrupt event register bit 7 (isoChRx) interrupt. |

## 5 Internal Registers (continued)

### 5.46 Isochronous Receive Interrupt Mask (isoRecvIntMask) Register

The isochronous receive interrupt mask 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 the contents of the isochronous transmit interrupt mask register. In all cases, the enables for each interrupt event correspond to the isoRecvIntEvent register bits. Setting a bit in this register enables the corresponding interrupt event in the isoRecvIntEvent register. Clearing a bit in this register disables the corresponding interrupt event in the isoRecvIntEvent register.

Offset:           A8h               set register  
                  ACh               clear register  
Default:           0000 000Xh

Reference: *1394 Open Host Controller Specification*, Rev. 1.1, Section 6.4

### 5.47 Fairness Control Register

The fairness control register provides a mechanism by which software can direct the host controller to transmit multiple asynchronous requests during a fairness interval, as specified by the *IEEE-1394a Specification*.

Offset:           DCh  
Default:           0000 0000h

Reference:       *1394 Open Host Controller Specification*, Rev. 1.1, Section 5.9

**Table 44. Fairness Control Register Description**

| Bit  | Field Name | Type | Description  |
|------|------------|------|--|
| 31:8 | Reserved   | R    | <b>Reserved.</b> Bits 31:8 return 0s when read.  |
| 7:0  | pri_req    | RW   | This field specifies the maximum number of priority arbitration requests for asynchronous request packets that the link is permitted to make of the PHY core during the fairness interval. |

## 5 Internal Registers (continued)

### 5.48 Link Control Register

The link control register provides flags to enable and configure the link core cycle timer and receiver portions of the FW322.

Offset: E0h set register  
E4h clear register  
Default: 00X0 0X00h

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Section 5.10

**Table 45. Link Control Register Description**

| Bit   | Field Name         | Type | Description  |
|-------|--------------------|------|--|
| 31:23 | Reserved           | R    | <b>Reserved.</b> Bits 31:23 return 0s when read.   |
| 22    | cycleSource        | RSC  | Set to 0, since the FW322 does not support an external cycle timer.  |
| 21    | cycleMaster        | RSCU | When this bit is set, and the FW322 PHY core has notified the OHCI core that it is root, the OHCI generates a cycle start packet every time the cycle timer rolls over, based on the setting of bit 22. When this bit is cleared, the OHCI accepts received cycle start packets to maintain synchronization with the node that is sending them. This bit is automatically reset when bit 25 (cycleTooLong) of the Interrupt Event register (see Table 39) is set and cannot be set until bit 25 (cycleTooLong) is cleared. |
| 20    | CycleTimerEnable   | RSC  | When this bit is set, the cycle timer offset counts cycles of the 24.576 MHz clock and rolls over at the appropriate time based on the settings of the above bits. When this bit is cleared, the cycle timer offset does not count.  |
| 19:11 | Reserved           | R    | <b>Reserved.</b> Bits 19:11 return 0s when read.   |
| 10    | RcvPhyPkt          | RSC  | When this bit is set, the receiver accepts incoming PHY core packets into the AR request context if the AR request context is enabled. This does not control receipt of self-identification packets received outside of the SelfID phase of bus initialization.  |
| 9     | RcvSelfID          | RSC  | When this bit is set, the receiver accepts incoming self-identification packets. Before setting this bit to 1, software must ensure that the SelfID Buffer Pointer register contains a valid address.  |
| 8:7   | Reserved           | R    | <b>Reserved.</b>   |
| 6     | tag1SyncFilterLock | RS   | When this bit is set, the tag1SyncFilter bit of the IR Context Match register (see Table 59) equals one for all IR contexts. When this bit is cleared, the tag1SynchFilter bit has read/write access. A hardware reset clears this bit to 0. A soft reset has no effect.   |
| 5:0   | Reserved           | R    | <b>Reserved.</b> Bits 5:0 return 0s when read.   |



## 5 Internal Registers (continued)

### 5.49 Node Identification Register

The node identification register contains the address of the node on which the OHCI resides, and indicates the valid node number status. The 16-bit combination of the busNumber field (bits 15:6) and the NodeNumber field (bits 5:0) is referred to as the node ID.

Offset: E8h  
Default: 0000 FFXh

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Section 5.11

**Table 46. Node Identification Register Description**

| Bit   | Field Name | Type | Description   |
|-------|------------|------|---|
| 31    | iDValid    | RU   | This bit indicates whether or not the FW322 has a valid node number. It is cleared when a 1394 bus reset is detected and set when the FW322 receives a new node number from the PHY core.   |
| 30    | root       | RU   | This bit is set during the bus reset process if the attached PHY core is root.  |
| 29:28 | Reserved   | R    | <b>Reserved.</b> Bits 29:28 return 0s when read.  |
| 27    | CPS        | RU   | Set if the PHY core is reporting that cable power status is OK.   |
| 26:16 | Reserved   | R    | <b>Reserved.</b> Bits 26:16 return 0s when read.  |
| 15:6  | busNumber  | RWU  | This number is used to identify the specific 1394 bus to which the FW322 belongs when multiple 1394-compatible buses are connected via a bridge.  |
| 5:0   | NodeNumber | RU   | This number is the physical node number established by the PHY core during self-identification. It is automatically set to the value received from the PHY core after the self-identification phase. If the PHY core sets the nodeNumber to 63, then software should not set the run bit of the ContextControl register for either of the AT DMA contexts (see Table 53). |

## 5 Internal Registers (continued)

### 5.50 PHY Core Layer Control Register

The PHY core layer control register is used to read or write a PHY core register.

Offset: ECh  
Default: 0000 0000h

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Section 5.12

**Table 47. PHY Core Layer Control Register Description**

| Bit   | Field Name | Type | Description   |
|-------|------------|------|---|
| 31    | rdDone     | RU   | This bit is cleared to 0 by the FW322 when either bit 15 (rdReg) or bit 14 (wrReg) is set. This bit is set when a register transfer is received by the OHCI core from the PHY core and rdData is updated. |
| 30:28 | Reserved   | R    | <b>Reserved.</b> Bits 30:28 return 0s when read.  |
| 27:24 | rdAddr     | RU   | This is the address of the register most recently received from the PHY core.   |
| 23:16 | rdData     | RU   | This field is the contents of a PHY Core register, which have been read at rdAddr.  |
| 15    | rdReg      | RWU  | This bit is set by software to initiate a read request to a PHY Core register and is cleared by hardware when the request has been sent. Bit 14 (wrReg) must not be set when bit 15 (rdReg) is set.       |
| 14    | wrReg      | RWU  | This bit is set by software to initiate a write request to a PHY Core register and is cleared by hardware when the request has been sent. Bit 15 (rdReg) must not be set when bit 14 (wrReg) is set.      |
| 13:12 | Reserved   | R    | <b>Reserved.</b> Bits 13:12 return 0s when read.  |
| 11:8  | regAddr    | RW   | This field is the address of the PHY Core register to be written or read.   |
| 7:0   | wrData     | RW   | This field is the data to be written to a PHY Core register and is ignored for reads.   |

### 5.51 Isochronous Cycle Timer Register

The isochronous cycle timer register indicates the current cycle number and offset. When the FW322 is cycle master, this register is transmitted with the cycle start message. When the FW322 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.

Offset: F0h  
Default: XXXX XXXXh

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Section 5.13

**Table 48. Isochronous Cycle Timer Register Description**

| Bit   | Field Name   | Type | Description   |
|-------|--------------|------|---|
| 31:25 | cycleSeconds | RWU  | This field counts seconds [rollovers from bits 24:12 (cycleCount field)] modulo 128.  |
| 24:12 | cycleCount   | RWU  | This field counts cycles [rollovers from bits 11:0 (cycleOffset field)] modulo 8000.  |
| 11:0  | cycleOffset  | RWU  | This field counts 24.576 MHz clocks modulo 3072, i.e., 125 ms. If an external 8 kHz clock configuration is being used, then this bit must be set to 0 at each tick of the external clock. |

## 5 Internal Registers (continued)

### 5.52 Asynchronous Request Filter High Register

The asynchronous request filter high 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 FW322. All nonlocal bus-sourced packets are not acknowledged unless bit 31 in this register is set.

Offset: 100h set register  
104h clear register

Default: 0000 0000h

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Section 5.14

**Table 49. Asynchronous Request Filter High Register Description**

| Bit  | Field Name         | Type | Description   |
|------|--------------------|------|---|
| 31   | asynReqResourceAll | RSCU | If this bit is set, then all asynchronous requests received by the FW322 from nonlocal bus nodes are accepted and the values of all asynReqResourceN bits will be ignored. Set/clear operations to this register while the IntEvent.busReset bit (see Table 39) is asserted will have no effect. A bus reset will not affect the value of the asynReqResourceAll bit. |
| 30:0 | asynReqResourceN   | RSCU | If this bit is set, then asynchronous requests received from node N (where N = the bit number + 32) on local bus are accepted by FW322. All asynReqResourceN bits will be cleared to zero when a bus reset occurs. Set/clear operations to this register while the IntEvent.busReset bit (see Table 39) is asserted will have no effect.                              |

### 5.53 Asynchronous Request Filter Low Register

The asynchronous request filter low 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 identically to the asynchronous request filter high register.

Offset: 108h set register  
10Ch clear register

Default: 0000 0000h

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Section 5.14

**Table 50. Asynchronous Request Filter Low Register Description**

| Bit  | Field Name       | Type | Description  |
|------|------------------|------|--|
| 31:0 | asynReqResourceN | RSCU | If this bit is set for local bus node number N (where N = the bit number from 0 to 31), then asynchronous requests received by the FW322 from that node are accepted. All asynReqResourceN bits will be cleared to zero when a bus reset occurs. Set/clear operations to this register while the IntEvent.busReset bit (see Table 39) is asserted will have no effect. |

## 5 Internal Registers (continued)

### 5.54 Physical Request Filter High Register

The physical request filter high set/clear register is used to enable physical receive requests on a per-node basis and handle 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 is handled by the ARRQ context instead of the physical request context.

Offset:            110h            set register  
                    114h            clear register

Default:           0000 0000h

Reference:        *1394 Open Host Controller Specification*, Rev. 1.1, Section 5.14.2

**Table 51. Physical Request Filter High Register Description**

| Bit  | Field Name              | Type | Description   |
|------|-------------------------|------|---|
| 31   | physReqResourceAllBuses | RSC  | If this bit is set, then all asynchronous requests received by the FW322 from nonlocal bus nodes are accepted. Set/clear operations to this register while the IntEvent.busReset bit (see Table 39) is asserted will have no effect. A bus reset will not affect the value of the physReqResourceAllBuses bit.  |
| 30:0 | physReqResourceN        | RSC  | If this bit is set, requests received by the FW322 from local bus node N (where N = bit number + 32) will be handled through the physical request context. Set/clear operations to this register while the IntEvent.busReset bit (see Table 39) is asserted will have no effect. All physReqResourceN bits will be cleared to zero when a bus reset occurs. |

### 5.55 Physical Request Filter Low Register

The physical request filter low set/clear register is used to enable physical receive requests on a per-node basis and handle 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 is handled by the asynchronous request context instead of the physical request context.

Offset:            118h            set register  
                    11Ch            clear register

Default:           0000 0000h

Reference:        *1394 Open Host Controller Specification*, Rev. 1.1, Section 5.14.2

**Table 52. Physical Request Filter Low Register Description**

| Bit  | Field Name       | Type | Description  |
|------|------------------|------|--|
| 31:0 | physReqResourceN | RSC  | If this bit is set, requests received by the FW322 from local bus node N (where N = bit number) will be handled through the physical request context. Set/clear operations to this register while the IntEvent.busReset bit (see Table 39) is asserted will have no effect. All physReqResourceN bits will be cleared to zero when a bus reset occurs. |

## 5 Internal Registers (continued)

### 5.56 Asynchronous Context Control Register

The asynchronous context control set/clear register controls the state and indicates status of the DMA context.

Offset:           180h           set register (ATRQ)  
                   184h           clear register (ATRQ)  
                   1A0h           set register (ATRS)  
                   1A4h           clear register (ATRS)  
                   1C0h           set register (ARRQ)  
                   1C4h           clear register (ARRQ)  
                   1E0h           set register (ARRS)  
                   1E4h           clear register (ARRS)

Default:           0000 X0XXh

Reference:        1394 Open Host Controller Specification, Rev. 1.1, Section 7.22, 8.3.2, 3.1.1

**Table 53. Asynchronous Context Control Register Description**

| Bit   | Field Name  | Type | Description   |
|-------|---|------|---|
| 31:16 | Reserved  | R    | <b>Reserved.</b> Bits 31:16 return 0s when read.  |
| 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 FW322 changes this bit (i.e., sets it to 0) only on a hardware or software reset.  |
| 14:13 | Reserved  | R    | <b>Reserved.</b> Bits 14:13 return 0s when read.  |
| 12    | wake  | RSU  | Software sets this bit to cause the FW322 to continue or resume descriptor processing. The FW322 clears this bit on every descriptor fetch.   |
| 11    | dead  | RU   | The FW322 sets this bit when it encounters a fatal error and clears the bit when software resets bit 15 (run).  |
| 10    | active  | RU   | The FW322 sets this bit to 1 when it is processing descriptors.   |
| 9:8   | Reserved  | R    | <b>Reserved.</b> Bits 9:8 return 0s when read.  |
| 7:5   | spd<br>(These bits are reserved, undefined for the ATRQ and ATRS contexts.) | 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><br>000 = 100 Mbits/s.<br>001 = 200 Mbits/s.<br>010 = 400 Mbits/s.<br><br>All other values are reserved. Software should not attempt to interpret the contents of this field while the active or wake bits are set. |
| 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.  |

## 5 Internal Registers (continued)

### 5.57 Asynchronous Context Command Pointer Register

The asynchronous context command pointer register contains a pointer to the address of the first descriptor block that the FW322 accesses when software enables the context by setting the asynchronous context control register bit 15 (run).

Offset:           18Ch           (ATRQ)  
                  1ACh           (ATRS)  
                  1CCh           (ARRQ)  
                  1ECh           (ARRS)  
Default:        XXXX XXXXh

Reference:       1394 Open Host Controller Specification, Rev. 1.1, Sections 3.1.2, 7.2.1, 8.3.1

**Table 54. Asynchronous Context Command Pointer Register Description**

| Bit  | Field Name        | Type | 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, then it indicates that the descriptorAddress field (bits 31:4) is not valid. Valid values for z are context specific. Refer to the OHCI specification for more details. |

## 5 Internal Registers (continued)

### 5.58 Isochronous Transmit Context Control (IT DMA ContextControl) Register

The isochronous transmit context control 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 ( $n = 0:7$ ).

Offset:             $200h + (16 * n)$                             set register  
                       $204h + (16 * n)$                             clear register  
 Default:           XXXX X0XXh

Reference:        *1394 Open Host Controller Specification*, Rev. 1.1, Sections 9.2, 3.1.1

**Table 55. Isochronous Transmit Context Control Register Description**

| Bit   | Field Name       | Type | Description   |
|-------|------------------|------|---|
| 31    | cycleMatchEnable | RSCU | When this bit is set to 1, processing occurs such that the packet described by the context's first descriptor block is transmitted in the cycle whose number is specified in the cycleMatch field (bits 30:16). The cycleMatch field (bits 30:16) must match the low-order 2 bits of cycleSeconds and the 13-bit cycleCount field in the cycle start packet that is sent or received immediately before isochronous transmission begins. Since the isochronous transmit DMA controller may work ahead, the processing of the first descriptor block may begin slightly in advance of the actual cycle in which the first packet is transmitted. The effects of this bit, however, are impacted by the values of other bits in this register and are explained in the <i>1394 Open Host Controller Interface Specification</i> . Once the context has become active, hardware clears this bit. |
| 30:16 | cycleMatch       | RSC  | Contains a 15-bit value, corresponding to the low-order 2 bits of the bus isochronous cycle time register cycleSeconds field (bits 31: 25) and the cycleCount field (bits 24:12) (see Table 48). If bit 31 (cycleMatchEnable) is set, then this isochronous transmit DMA context becomes enabled for transmits when the low-order 2 bits of the bus isochronous cycle timer register cycleSeconds field (bits 31:25) and the cycleCount field (bits 24:12) value equal this field's (cycleMatch) value.   |
| 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 FW322 changes this bit only on a hardware or software reset.   |
| 14:13 | Reserved         | R    | <b>Reserved.</b> Bits 14:13 return 0s when read.  |
| 12    | wake             | RSU  | Software sets this bit to cause the FW322 to continue or resume descriptor processing. The FW322 clears this bit on every descriptor fetch.   |
| 11    | dead             | RU   | The FW322 sets this bit when it encounters a fatal error and clears the bit when software resets bit 15 (run).  |
| 10    | active           | RU   | The FW322 sets this bit to 1 when it is processing descriptors.   |
| 9:5   | Reserved         | R    | <b>Reserved.</b> Bits 9:5 return 0s when read.  |
| 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.   |

## 5 Internal Registers (continued)

### 5.59 Isochronous Transmit Context Command Pointer Register

The isochronous transmit context command pointer register contains a pointer to the address of the first descriptor block that the FW322 accesses when software enables an isochronous transmit context by setting the isochronous transmit context control register bit 15 (run). The n value in the following register addresses indicates the context number (n = 0:7).

Offset:  $20Ch + (16 * n)$

Default: XXXX XXXXh

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Sections 9.2.1, 3.1.2

**Table 56. Isochronous Transmit Context Command Pointer Register Description**

| Bit  | Field Name        | Type | Description   |
|------|-------------------|------|---|
| 31:4 | descriptorAddress | RWU  | Address of the context program, which will be executed when a DMA context is started. All descriptors are 16-byte aligned, so the four least significant bits of any descriptor address must be zero. |
| 3:0  | Z                 | RWU  | Indicates how many physically contiguous descriptors are pointed to by descriptorAddress.   |



## 5 Internal Registers (continued)

### 5.60 Isochronous Receive Context Control (IR DMA ContextControl) Register

The isochronous receive context control 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 = 0:7).

Offset:            400h + (32 \* n)                      set register  
                      404h + (32 \* n)                      clear register  
 Default:           X000 X0XXh

Reference:        1394 Open Host Controller Specification, Rev. 1.1, Sections 10.3, 3.1.2

**Table 57. Isochronous Receive Context Control Register Description**

| Bit   | Field Name       | Type | Description  |
|-------|------------------|------|--|
| 31    | bufferFill       | RSC  | When this bit is set, received packets are placed back-to-back to completely fill each receive buffer. When this bit is cleared, each received packet is placed in a single buffer. If bit 28 (multiChanMode) is set to 1, then this bit must also be set to 1. The value of this bit must not be changed while bit 10 (active) or bit 15 (run) is set.  |
| 30    | isochHeader      | RSC  | When this bit is 1, received isochronous packets include the complete 4-byte isochronous packet header seen by the link layer. The end of the packet is marked with an xferStatus in the first doublet, and a 16-bit timeStamp indicating the time of the most recently received (or sent) cycleStart packet. When this bit is cleared, the packet header is stripped off of received isochronous packets. The packet header, if received, immediately precedes the packet payload. The value of this bit must not be changed while bit 10 (active) or bit 15 (run) is set.  |
| 29    | cycleMatchEnable | RSCU | When this bit is set, the context begins running only when the 15-bit cycleMatch field (bits 26:12) in the IRContext Match register (see Table 59) matches the two low-order bits of the CycleSeconds field and the 13-bit CycleCount field in the CycleTimer register. 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 this bit. The value of this bit must not be changed while bit 10 (active) or bit 15 (run) is set.   |
| 28    | multiChanMode    | RSC  | When this bit is set, the corresponding isochronous receive DMA context receives packets for all isochronous channels enabled in the isochronous receive channel mask high and isochronous receive channel mask low registers. The isochronous channel number specified in the isochronous receive DMA Context Match register is ignored. When this bit is cleared, the isochronous receive DMA context receives packets for the channel number specified in the context match register. Only one isochronous receive DMA context may use the isochronous receive channel mask registers. If more than one isochronous receive context control register has this bit set, then results are undefined. The value of this bit must not be changed while bit 10 (active) or bit 15 (run) is set to 1. |
| 27    | dualBufferMode   | RSC  | When this bit is set, received packets are separated into first and second payload and streamed independently to the first buffer series and second buffer series (see OHCI v.1.1, 10.2.3). Both multiChanMode and buffer fill must be programmed to zero when this bit is set. The value of dualBufferMode will not be changed while active or run is set.  |
| 26:16 | Reserved         | R    | <b>Reserved.</b> Bits 26:16 return 0s when read.   |

## 5 Internal Registers (continued)

Table 57. Isochronous Receive Context Control Register Description (continued)

| Bit   | Field Name | Type | 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 FW322 changes this bit only on a hardware or software reset. |
| 14:13 | Reserved   | R    | <b>Reserved.</b> Bits 14:13 return 0s when read.  |
| 12    | wake       | RSU  | Software sets this bit to cause the FW322 to continue or resume descriptor processing. The FW322 clears this bit on every descriptor fetch.   |
| 11    | dead       | RU   | The FW322 sets this bit when it encounters a fatal error and clears the bit when software resets bit 15 (run).  |
| 10    | active     | RU   | The FW322 sets this bit to 1 when it is processing descriptors.   |
| 9:8   | Reserved   | R    | <b>Reserved.</b> Bits 9:8 return 0s when read.  |
| 7:5   | spd        | RU   | This field indicates the speed at which the packet was received.<br>000 = 100 Mbits/s.<br>001 = 200 Mbits/s.<br>010 = 400 Mbits/s.<br>All other values are reserved.                                |
| 4:0   | event code | RU   | Following an INPUT_* command, the error or status code is indicated in this field (see OHCI v.1.1, 10.3.2 and Table 3-2).   |

### 5.61 Isochronous Receive Context Command Pointer Register

The isochronous receive context command pointer register contains a pointer to the address of the first descriptor block that the FW322 accesses when software enables an isochronous receive context by setting the isochronous receive context control register bit 15 (run). The n value in the following register addresses indicates the context number (n = 0:7).

Offset: 40Ch + (32 \* n)  
Default: XXXX XXXXh

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Sections 10.3, 3.1.2

Table 58. Isochronous Receive Context Command Pointer Register Description

| Bit  | Field Name        | Type | Description  |
|------|-------------------|------|--|
| 31:4 | descriptorAddress | RWU  | Address of the context program that will be executed when a DMA context is started.  |
| 3:0  | Z                 | RWU  | Indicates how many physically contiguous descriptors are pointed to by descriptor address. In buffer full mode, Z will be either one or zero. In packet-per-buffer mode, Z will be from zero to eight. |

## 5 Internal Registers (continued)

### 5.62 Isochronous Receive Context Match (IR DMA ContextMatch) Register

The isochronous receive context match register is used to control on which isochronous cycle the context should start. The register is also used to control which packets are accepted by the context.

Offset: 410Ch + (32 \* n)

Default: XXXX XXXXh

Reference: 1394 Open Host Controller Specification, Rev. 1.1, Section 10.3. 3.

**Table 59. Isochronous Receive Context Match Register Description**

| Bit   | Field Name     | Type | Description  |
|-------|----------------|------|--|
| 31    | tag3           | RW   | If this bit is set, then this context matches on iso receive packets with a tag field of 11b.  |
| 30    | tag2           | RW   | If this bit is set, then this context matches on iso receive packets with a tag field of 10b.  |
| 29    | tag1           | RW   | If this bit is set, then this context matches on iso receive packets with a tag field of 01b.  |
| 28    | tag0           | RW   | If this bit is set, then this context matches on iso receive packets with a tag field of 00b.  |
| 27    | Reserved       | R    | <b>Reserved.</b> Bit 27 returns a 0 when read.   |
| 26:12 | cycleMatch     | RW   | Contains a 15-bit value, corresponding to the low-order 2 bits of cycleSeconds and the 13-bit cycleCount field in the cycleStart packet. If isochronous receive context control register bit 29 (cycleMatchEnable) is set, then this context is enabled for receives when the 2 low-order bits of the bus isochronous cycle timer register cycleSeconds field (bits 31:25) and cycleCount field (bits 24:12) value equal this field's (cycleMatch) value.  |
| 11:8  | sync           | RW   | This field contains the 4-bit field, which is compared to the sync field of each isochronous packet for this channel when the command descriptor's w field is set to 11b.  |
| 7     | Reserved       | R    | <b>Reserved.</b> Bit 7 returns 0 when read.  |
| 6     | tag1SyncFilter | RW   | If this bit and bit 29 (tag1) are set, then packets with tag2'b01 are accepted into the context if the two most significant bits of the packets sync field are 00b. Packets with tag values other than 01b are filtered according to tag0, tag2, and tag3 (bits 28, 30, and 31, respectively) without any additional restrictions. If this bit is cleared, then this context matches on isochronous receive packets as specified in bits 28:31 (tag0:tag3) with no additional restrictions. If the tag1SyncFilterLock bit of the Link Control register is set, then this bit is read only and is set to one by the OHCI. |
| 5:0   | channelNumber  | RW   | This 6-bit field indicates the isochronous channel number for which this isochronous receive DMA context accepts packets.  |

## 5 Internal Registers (continued)

### 5.63 FW322 Vendor-Specific Registers

The FW322 contains a number of vendor-defined registers used for diagnostics and control of low-level hardware functionality. These registers are addressable in the upper 2K of the 4K region defined by PCI base address register 0 (registers defined by the OHCI specification reside in the lower 2K of this region). These registers are also programmable via the serial EEPROM. These control registers should not be changed when the link is enabled.

**Table 60. FW322 Vendor-Specific Registers Description**

| Offset  | Register Name | Description   |
|---------|---------------|---|
| 12'h800 | IsoDMACtrl    | Controls PCI access for the isochronous DMA engines. Initial values are loaded from serial EEPROM, if present (see Table 61 of this data sheet).                        |
| 12'h808 | AsyDMACtrl    | Controls PCI access and AT FIFO threshold for the asynchronous DMA engines. Initial values are loaded from serial EEPROM, if present (see Table 62 of this data sheet). |
| 12'h840 | LinkOptions   | Controls low-level functionality of the link core. Initial values are loaded from serial EEPROM, if present (see Table 63 of this data sheet).                          |

### 5.64 Isochronous DMA Control

The fields in this register control when the isochronous DMA engines access the PCI bus and how much data they will attempt to move in a single PCI transaction. The actual PCI burst sizes will also be affected by 1394 packet size, host memory buffer size, FIFO constraints, and the PCI cache line size.

Offset:800h

Default:0000 7373h

**Table 61. Isochronous DMA Control Registers Description**

| Bits  | Field            | Description  |
|-------|------------------|--|
| 15:12 | IT Maximum Burst | The maximum number of quadlets that will be fetched by the IT DMA in one PCI transaction. The maximum burst is $16 * (n + 1)$ quadlets; <b>n</b> defaults to 7 (128 quadlets). Max value of <b>n</b> is 0xf.   |
| 11:8  | IT Threshold     | This field defines the amount of available space that is needed in the IT FIFO, before the IT DMA will request access to the PCI bus. The threshold is $16 * (n + 1)$ quadlets; <b>n</b> defaults to 3 (64 quadlets). Note, however, that the IT DMA may request access to the PCI bus sooner if the amount of data to be fetched from memory is less than the amount of space available in the IT FIFO. Max value of <b>n</b> is 0xf.   |
| 7:4   | IR Maximum Burst | The maximum number of quadlets that will be written by the IR DMA in one PCI transaction. The maximum burst is $16 * (n + 1)$ quadlets; <b>n</b> defaults to 7 (128 quadlets). Max value of <b>n</b> is 0xf.   |
| 3:0   | IR Threshold     | This field defines the amount of available data that is needed in the IR FIFO, before the IR DMA will request access to the PCI bus. The threshold is $16 * (n + 1)$ quadlets; <b>n</b> defaults to 0 (16 quadlets). Note, however, that the IR DMA may request access to the PCI bus sooner if the amount of data available in the FIFO exceeds the space remaining in the current host memory buffer or a complete packet resides in the FIFO. Max value of <b>n</b> is 0xf. |

## 5 Internal Registers (continued)

### 5.65 Asynchronous DMA Control

The fields in this register control the functionality within the asynchronous and physical DMA engines. Accesses to the PCI bus and how much data the DMA engines will attempt to move in a single PCI transaction can be controlled. The actual PCI burst sizes will also be affected by 1394 packet size, host memory buffer size, FIFO constraints, and the PCI cache line size.

Offset: 808h  
Default: 0010 7373h

**Table 62. Asynchronous DMA Control Registers Description**

| Bits  | Field                          | Description   |
|-------|--------------------------------|---|
| 24    | Retry Threshold<br>Max. Enable | When this bit is set, a packet being retried, e.g., due to an ack_busy on the initial attempt, will behave as if the AT FIFO threshold value was set to the maximum ( $n = 0x20$ ). The purpose of this feature is to prevent a packet that previously experienced a FIFO underrun on the initial transmit attempt from failing again due to a FIFO underrun on the retry attempt. If this bit is not set, retried packets will use the same AT FIFO threshold as the initial transmit attempt. The default value of this field is 0x0. |
| 23:16 | AT FIFO Threshold              | This field defines the number of quadlets of packet data that must be available in the AT FIFO before the link will be notified that there is an asynchronous packet to be transmitted. (The link will also be signaled that a packet is available for transmission if the entire packet is in the FIFO, regardless of its size.) The threshold is $16 * n$ quadlets; <b>n</b> defaults to a value of 0x10 (256 quadlets). Max size of <b>n</b> is 0x20 (512 quadlets).   |
| 15:12 | AT Maximum Burst               | The maximum number of quadlets that will be fetched by the AT or physical read response DMAs in one PCI transaction. The maximum burst is $16 * (n + 1)$ quadlets; <b>n</b> defaults to 7 (128 quadlets). Max value of <b>n</b> is 0xf.   |
| 11:8  | AT Threshold                   | This field defines the amount of available space that is needed in the AT FIFO, before the AT or physical read response units will request access to the PCI bus. The threshold is $16 * (n + 1)$ quadlets; <b>n</b> defaults to 0 (16 quadlets). Note, however, that the AT or physical DMAs may request access to the PCI bus sooner if the amount of data to be fetched from memory is less than the amount of space available in the AT FIFO. Max value of <b>n</b> is 0xf.   |
| 7:4   | AR Maximum Burst               | The maximum number of quadlets that will be written by the AR and physical write DMAs in one PCI transaction. The maximum burst is $16 * (n + 1)$ quadlets; <b>n</b> defaults to 7 (128 quadlets). Max value of <b>n</b> is 0xf.  |
| 3:0   | AR Threshold                   | This field defines the amount of available data that is needed in the AR FIFO, before the AR DMA will request access to the PCI bus. The threshold is $16 * (n + 1)$ quadlets; <b>n</b> defaults to 0 (16 quadlets). However, the AR DMA may request access to the PCI bus sooner if the amount of data available in the FIFO exceeds the space remaining in the current host memory buffer or a complete packet resides in the FIFO. Max value of <b>n</b> is 0xf.   |

## 5 Internal Registers (continued)

### 5.66 Link Options

The values in this register provide low-level control of configurable features within the FW322 that are beyond those stated in 1394 and OHCI specifications.

Offset: 840h  
Default: 0000 0020h

**Table 63. Link Options Register Description**

| Bits | Field               | Description   |
|------|---------------------|---|
| 31   | OHCI1.1En           | Enables general features of OHCI 1.1 that are not covered by any of the bits below.   |
| 30   | Reserved            | Reserved for internal use by the FW322. Must be set to 0x0.   |
| 29   | RegAccessFailEn     | Enables RegAccessFailEn interrupt for SCLK register accesses that fail.   |
| 28   | InitBMEnable        | Enables usage of initial registers for loading Bus Management registers on a bus reset.   |
| 27   | RetryEnable         | Enables retry processing as defined in OHCI 1.1.  |
| 26   | ConfigROMEnable     | Enables config ROM management, including config ROM block reads, as defined in OHCI 1.1.  |
| 25   | DualBufferEnable    | Enables IR dual-buffer mode processing as defined in OHCI 1.1.  |
| 24   | ITChangeEnable      | Enables skip and FIFO underrun processing in the IT context as defined in OHCI 1.1.   |
| 23   | Reserved            | Reserved for internal use by the FW322. Must be set to 0x0.   |
| 22   | Reserved            | Read-only status bit. Reserved for internal use by the FW322. Will read back as 0x0.  |
| 21   | Reserved            | Reserved for internal use by the FW322. Must be set to 0x0.   |
| 20   | Reserved            | Reserved for internal use by the FW322. Must be set to 0x0.   |
| 19:6 | Reserved            | Reserved for internal use by the FW322. Must be set to 0x0.   |
| 5:3  | Posted Writes       | Number of physical posted writes the link is allowed to queue in the asynchronous receive FIFO. These three bits [5:3] default to 100b, which is the maximum value. Values greater than 100b will disable all physical posted writes.   |
| 2:0  | Cycle Timer Control | Selects the value the FW322 will use for its isochronous cycle period when the FW322 is the root node. This value is for debugging purposes only and should not be set to any value other than its default value in a real 1394 network. This value defaults to 0.<br>If 0, cycle = 125 $\mu$ s.<br>If 1, cycle = 62.5 $\mu$ s.<br>If 2, cycle = 31.25 $\mu$ s.<br>If 3, cycle = 15.625 $\mu$ s.<br>If 4, cycle = 7.8125 $\mu$ s. |

## 6 Internal Register Configuration

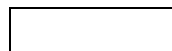
### 6.1 PHY Core Register Map

The PHY core register map is shown below in Table 64.

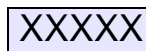
Reference: IEEE Standard 1394a-2000, Annex J2

**Table 64. PHY Core Register Map**

| Address | Contents               |           |           |          |             |            |            |            |
|---------|------------------------|-----------|-----------|----------|-------------|------------|------------|------------|
|         | Bit 0                  | Bit 1     | Bit 2     | Bit 3    | Bit 4       | Bit 5      | Bit 6      | Bit 7      |
| 00002   | Physical_ID            |           |           |          |             |            | R          | PS         |
| 00012   | RHB                    | IBR       | Gap_count |          |             |            |            |            |
| 00102   | Extended (7)           |           |           | XXXXXX   | Total_ports |            |            |            |
| 00112   | Max_speed              |           |           | XXXXXX   | Delay       |            |            |            |
| 01002   | LCtrl                  | Contender | Jitter    |          |             | Pwr_class  |            |            |
| 01012   | Watchdog               | ISBR      | Loop      | Pwr_fail | Timeout     | Port_event | Enab_accel | Enab_multi |
| 01102   | XXXXXX                 | XXXXXX    | XXXXXX    | XXXXXX   | XXXXXX      | XXXXXX     | XXXXXX     | XXXXXX     |
| 01112   | Page_select            |           |           | XXXXXX   | Port_select |            |            |            |
| 10002   | Register 0 Page_select |           |           |          |             |            |            |            |
| ⋮       | ⋮                      |           |           |          |             |            |            |            |
| 11112   | Register 7 Page_select |           |           |          |             |            |            |            |



REQUIRED



RESERVED

## 6 Internal Register Configuration (continued)

### 6.2 PHY Core Register Fields

Table 65. PHY Core Register Fields

| Field       | Size (In Bits) | Type | Power Reset Value | Description   |
|-------------|----------------|------|-------------------|---|
| Physical_ID | 6              | R    | 000000            | The address of this node is determined during self-identification. A value of 63 indicates a misconfigured bus; therefore, the link will not transmit any packets.  |
| R           | 1              | R    | 0                 | When set to one, indicates that this node is the root.  |
| PS          | 1              | R    | —                 | <b>Cable Power Active.</b> The PHY core sets this bit when cable power measured at the connector is at least 7.5 V. The PHY core clears this bit when the detectable voltage is below this value.   |
| RHB         | 1              | RW   | 0                 | <b>Root Hold-Off Bit.</b> When set to one, the force_root variable is TRUE. This instructs the PHY core to attempt to become the root during the next tree identify process.  |
| IBR         | 1              | RW   | 0                 | <b>Initiate Bus Reset.</b> When set to one, instructs the PHY core to set ibr TRUE and reset_time to RESET_TIME. These values, in turn, cause the PHY core to initiate a bus reset without arbitration; the reset signal is asserted for 166 $\mu$ s. This bit is self-clearing.  |
| Gap_count   | 6              | RW   | 3F16              | Used to configure the arbitration timer setting to optimize gap times according to the topology of the bus. See Section 4.3.6 of <i>IEEE Standard 1394a-2000</i> for the encoding of this field.  |
| Extended    | 3              | R    | 7                 | This field has a constant value of seven, which indicates the extended PHY core register map.   |
| Total_ports | 4              | R    | 2                 | The number of ports implemented by this PHY core. This count reflects the number.   |
| Max_speed   | 3              | R    | 0102              | Indicates the speed(s) this PHY core supports:<br>0002 = 98.304 Mbits/s.<br>0012 = 98.304 and 196.608 Mbits/s.<br>0102 = 98.304, 196.608, and 393.216 Mbits/s.<br>0112 = 98.304, 196.608, 393.216, and 786.43 Mbits/s.<br>1002 = 98.304, 196.608, 393.216, 786.432, and 1,572.864 Mbits/s.<br>1012 = 98.304, 196.608, 393.216, 786.432, 1,572.864, and 3,145.728 Mbits/s.<br>All other values are reserved for future definition. |
| Delay       | 4              | R    | 0000              | Worst-case repeater delay; total worst-case repeater delay = $[144 + (\text{delay} * 20)]$ ns.  |
| LCtrl       | 1              | RW   | 1                 | <b>Link Active.</b> Cleared or set by software to control the value of the L bit transmitted in the node's SelfID packet 0, which will be the logical AND of this bit and LPS active.   |
| Contender   | 1              | RW   | See description   | Cleared or set by software to control the value of the C bit transmitted in the SelfID packet. Powerup reset value is 0.  |
| Jitter      | 3              | R    | 000               | The difference between the fastest and slowest repeater data delay = $[(\text{Jitter} + 1) * 20]$ ns.   |
| Pwr_class   | 3              | RW   | See description   | <b>Power Class.</b> Controls the value of the pwr field transmitted in the SelfID packet. See Section 4.3.4.1 of <i>IEEE Standard 1394a-2000</i> for the encoding of this field. The PC2 pin determines the least significant bit of the power reset value. The two most significant bits default to a power reset value of 00.   |



## 6 Internal Register Configuration (continued)

Table 65. PHY Core Register Fields (continued)

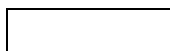
| Field       | Size | Type | Power Reset Value | Description   |
|-------------|------|------|-------------------|---|
| Watchdog    | 1    | RW   | 0                 | When set to one, the PHY core will set Port_event to one if resume operations commence for any port.  |
| ISBR        | 1    | RW   | 0                 | <b>Initiate Short (Arbitrated) Bus Reset.</b> A write of one to this bit instructs the PHY core to set ISBR true and reset_time to SHORT_RESET_TIME. These values, in turn, cause the PHY core to arbitrate and issue a short bus reset. This bit is self-clearing.   |
| Loop        | 1    | RW   | 0                 | <b>Loop Detect.</b> A write of one to this bit clears it to zero.   |
| Pwr_fail    | 1    | RW   | 1                 | <b>Cable Power Failure Detect.</b> Set to one when the PS bit changes from one to zero. A write of one to this bit clears it to zero.   |
| Timeout     | 1    | RW   | 0                 | <b>Arbitration State Machine Time-Out.</b> A write of one to this bit clears it to zero (see MAX_ARB_STATE_TIME).   |
| Port_event  | 1    | RW   | 0                 | <b>Port Event Detect.</b> The PHY core sets this bit to one if any of connected, bias, disabled, or fault change for a port whose Int_enable bit is one. The PHY core also sets this bit to one if resume operations commence for any port and watchdog bit is one. A write of one to this bit clears it to zero. |
| Enab_accel  | 1    | RW   | 0                 | <b>Enable Arbitration Acceleration.</b> When set to one, the PHY core will use the enhancements specified in clause 4.4 of <i>1394a-2000 Specification</i> . PHY core behavior is unspecified if the value of Enab_accel is changed while a bus request is pending.   |
| Enab_multi  | 1    | RW   | 0                 | <b>Enable Multispeed Packet Concatenation.</b> When set to one, the link will signal the speed of all packets to the PHY core.  |
| Page_select | 3    | RW   | 000               | Selects which of eight possible PHY Core register pages are accessible through the window at PHY Core register addresses 10002 through 11112, inclusive.  |
| Port_select | 4    | RW   | 0000              | If the page selected by Page_select presents per-port information, this field selects which port's registers are accessible through the window at PHY Core register addresses 10002 through 11112, inclusive. Ports are numbered monotonically starting at zero, p0.  |

## 6 Internal Register Configuration (continued)

The port status page is used to access configuration and status information for each of the PHY core's ports. The port is selected by writing zero to Page\_select and the desired port number to Port\_select in the PHY core register at address 01112. The format of the port status page is illustrated by Table 66 below; reserved fields are shown as XXXXX. The meanings of the register fields in the port status page are defined as type RSC.

**Table 66. PHY Core Register Page 0: Port Status Page**

| Address | Contents         |        |        |            |        |           |        |          |
|---------|------------------|--------|--------|------------|--------|-----------|--------|----------|
|         | Bit 0            | Bit 1  | Bit 2  | Bit 3      | Bit 4  | Bit 5     | Bit 6  | Bit 7    |
| 10002   | AStat            |        | BStat  |            | Child  | Connected | Bias   | Disabled |
| 10012   | Negotiated_speed |        |        | Int_enable | Fault  | XXXXXX    | XXXXXX | XXXXXX   |
| 10102   | XXXXXX           | XXXXXX | XXXXXX | XXXXXX     | XXXXXX | XXXXXX    | XXXXXX | XXXXXX   |
| 10112   | XXXXXX           | XXXXXX | XXXXXX | XXXXXX     | XXXXXX | XXXXXX    | XXXXXX | XXXXXX   |
| 11002   | XXXXXX           | XXXXXX | XXXXXX | XXXXXX     | XXXXXX | XXXXXX    | XXXXXX | XXXXXX   |
| 11012   | XXXXXX           | XXXXXX | XXXXXX | XXXXXX     | XXXXXX | XXXXXX    | XXXXXX | XXXXXX   |
| 11102   | XXXXXX           | XXXXXX | XXXXXX | XXXXXX     | XXXXXX | XXXXXX    | XXXXXX | XXXXXX   |
| 11112   | XXXXXX           | XXXXXX | XXXXXX | XXXXXX     | XXXXXX | XXXXXX    | XXXXXX | XXXXXX   |



REQUIRED



RESERVED

## 6 Internal Register Configuration (continued)

The meaning of the register fields in the port status page are defined by Table 68 below.

**Table 67. PHY Core Register Port Status Page Fields**

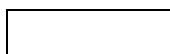
| Field            | Size | Type | Power Reset Value | Description   |
|------------------|------|------|-------------------|---|
| AStat            | 2    | R    | —                 | TPA line state for the port:<br>002 = invalid.<br>012 = 1.<br>102 = 0.<br>112 = Z.  |
| BStat            | 2    | R    | —                 | TPB line state for the port (same encoding as AStat).   |
| Child            | 1    | R    | 0                 | If this bit is equal to one, the port is a child; otherwise, a parent. The meaning of this bit is undefined from the time a bus reset is detected until the PHY core transitions to state T1: child handshake during the tree identify process (see Section 4.4.2.2 in <i>IEEE Standard 1394a-2000</i> ). |
| Connected        | 1    | R    | 0                 | If equal to one, the port is connected.   |
| Bias             | 1    | R    | 0                 | If equal to one, incoming TPBIAS is detected.   |
| Disabled         | 1    | RW   | 0                 | If equal to one, the port is disabled.  |
| Negotiated_speed | 3    | R    | 000               | Indicates the maximum speed negotiated between this PHY core port and its immediately connected port; the encoding is the same as for the PHY Core Register Max_speed field (see Table 65).   |
| Int_enable       | 1    | RW   | 0                 | <b>Enable Port Event Interrupts.</b> When set to one, the PHY core will set Port_event to one if any of connected, bias, disabled, or fault (for this port) change state.   |
| Fault            | 1    | RW   | 0                 | Set to one if an error is detected during a suspend or resume operation. A write of one to this bit clears it to zero.  |

## 6 Internal Register Configuration (continued)

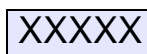
The vendor identification page is used to identify the PHY core's vendor and compliance level. The page is selected by writing one to Page\_select in the PHY core register at address 01112. The format of the vendor identification page is shown in Table 68; reserved fields are shown as XXXXX.

**Table 68. PHY Core Register Page 1: Vendor Identification Page**

| Address | Contents         |        |        |        |        |        |        |        |
|---------|------------------|--------|--------|--------|--------|--------|--------|--------|
|         | Bit 0            | Bit 1  | Bit 2  | Bit 3  | Bit 4  | Bit 5  | Bit 6  | Bit 7  |
| 10002   | Compliance_level |        |        |        |        |        |        |        |
| 10012   | XXXXXX           | XXXXXX | XXXXXX | XXXXXX | XXXXXX | XXXXXX | XXXXXX | XXXXXX |
| 10102   |                  |        |        |        |        |        |        |        |
| 10112   | Vendor_ID        |        |        |        |        |        |        |        |
| 11002   |                  |        |        |        |        |        |        |        |
| 11012   |                  |        |        |        |        |        |        |        |
| 11102   | Product_ID       |        |        |        |        |        |        |        |
| 11112   |                  |        |        |        |        |        |        |        |



REQUIRED



RESERVED

Note: The meaning of the register fields within the vendor identification page are defined by Table 70.

**Table 69. PHY Core Register Vendor Identification Page Fields**

| Field            | Size | Type | Description   |
|------------------|------|------|---|
| Compliance_level | 8    | r    | Standard to which the PHY core implementation complies:<br>0 = not specified<br>1 = <i>IEEE</i> 1394a-2000<br>Agere's FW322 compliance level is 1.<br>All other values reserved for future standardization.   |
| Vendor_ID        | 24   | r    | The company ID or organizationally unique identifier (OUI) of the manufacturer of the PHY core. Agere's vendor ID is <b>00601Dh</b> . This number is obtained from the <i>IEEE</i> registration authority committee (RAC). The most significant byte of Vendor_ID appears at PHY core register location 10102 and the least significant at 11002. |
| Product_ID       | 24   | r    | The meaning of this number is determined by the company or organization that has been granted Vendor_ID. Agere's FW322 PHY core product ID is 03227x16*. The most significant byte of Product_ID appears at PHY core register location 11012 and the least significant at 11112.  |

\* x is a minor revision number of the FW322 07 T100 and may be any value from 0 hex to F hex.

Note: The vendor-dependent page provides access to information used in the manufacturing test of the FW322.

## 7 Crystal Selection Considerations

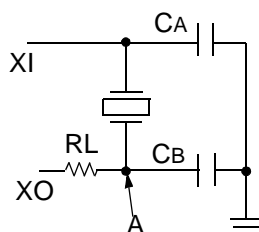
The FW322 is designed to use an external 24.576 MHz parallel resonant fundamental mode crystal connected between the XI and XO terminals to provide the reference for an internal oscillator circuit. The *IEEE* 1394a-2000 standard requires that FW322 have less than  $\pm 100$  ppm total variation from the nominal data rate, which is directly influenced by the crystal. To achieve this, it is recommended that an oscillator with a nominal 50 ppm or less frequency tolerance be used.

The total frequency variation must be kept below  $\pm 100$  ppm from nominal with some allowance for error introduced by board and device variations. Trade-offs between frequency tolerance and stability may be made as long as the total frequency variation is less than  $\pm 100$  ppm.

### 7.1 Load Capacitance

The frequency of oscillation is dependent upon the load capacitance specified for the crystal, in parallel resonant mode crystal circuits. Total load capacitance (CL) is a function of not only the discrete load capacitors, but also capacitances from the FW322 board traces and capacitances of the other FW322 connected components. The values for load capacitors (CA and CB) should be calculated using this formula:

$$CA = CB = (CL - C_{\text{stray}}) \times 2$$



Where:

CL = load capacitance specified by the crystal manufacturer.

Cstray = capacitance of the board and the FW322, typically 2 pF—3 pF.

RL = load resistance; nominal value is 400  $\Omega$ ; the best value to be used can be determined by customer testing.

Figure 7. Crystal Circuitry

### 7.2 Adjustment to Crystal Loading

The resistor (RL) in Figure 7 is recommended for fine-tuning the crystal circuit. The nominal value for this resistor is approximately 400  $\Omega$ . A more precise value for this resistor is dependent on the specific crystal used. Please refer to the crystal manufacturer's data sheet and application notes to determine an appropriate value for RL. A more precise value for this resistor can be obtained by placing different values of RL on a production board and using an oscilloscope to view the resultant clock waveform at node A for each resistor value. The desired waveform should have the following characteristics: the waveform should be sinusoidal, with an amplitude as large as possible, but not greater than 3.3 V or less than 0 V.

### 7.3 Crystal/Board Layout

The layout of the crystal portion of the PHY circuit is important for obtaining the correct frequency and minimizing noise introduced into the FW322 PLL. The crystal and two load capacitors (CA + CB) should be considered as a unit during layout. They should be placed as close as possible to one another, while minimizing the loop area created by the combination of the three components. Minimizing the loop area minimizes the effect of the resonant current that flows in this resonant circuit. This layout unit (crystal and load capacitors) should then be placed as close as possible to the PHY XI and XO terminals to minimize trace lengths. Vias should not be used to route the XI and XO signals.

## 8 Serial EEPROM Interface

The FW322 features an I<sup>2</sup>C compliant serial ROM interface that allows for the connection of an external serial EEPROM. The interface provides a mechanism to store configurable data such as the global unique identification (GUID) within an external EEPROM.

The interface consists of the ROM\_AD and ROM\_CLK pins. ROM\_CLK is an output clock provided by the FW322 to the external EEPROM. ROM\_AD is bidirectional and is used for serial data/control transfers between the FW322 and the external EEPROM.

The FW322 uses this interface to read the contents of the serial EEPROM in response to the first PCI reset after powerup. The FW322 also makes the serial ROM interface visible to software through the OHCI defined GUID ROM register. When the FW322 is operational, the GUID ROM register allows software to initiate reads to the external EEPROM.

The FW322 EEPROM interface has a number of new and updated features that were not present in earlier revisions of the chip:

1. Improved external interface timing.
2. OHCI soft reset and D3-to-D0 device state transition will not initiate an EEPROM load.
3. Retry of PCI/OHCI/vendor register accesses during an EEPROM load.
4. Load failure and quick EEPROM detection.
5. EEPROM cache.
6. Automatic update of the OHCI 1394 MiniROM field of the GUID\_ROM register.
7. Additional EEPROM image formats.

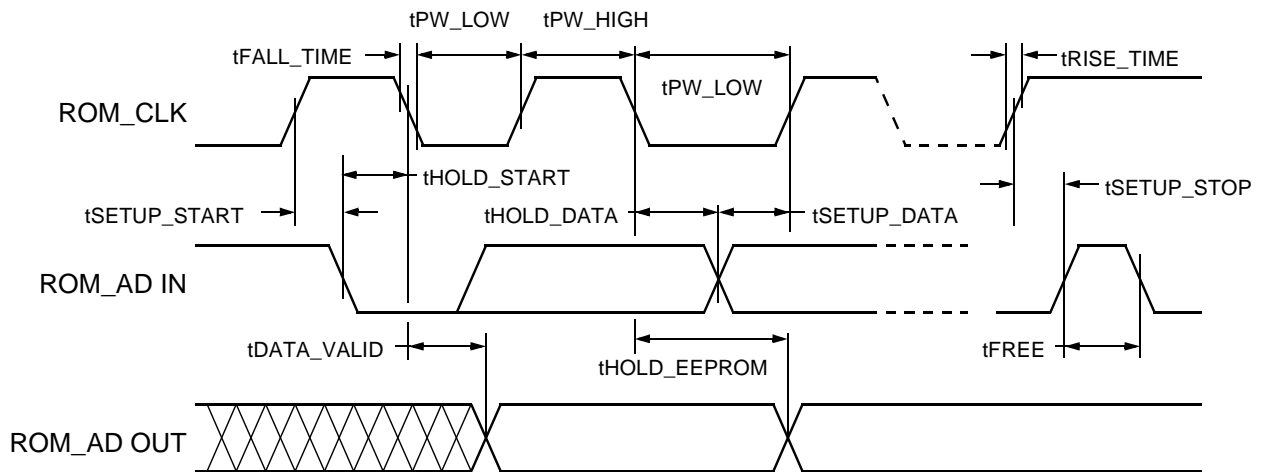
For additional detail, refer to the *FW322/FW323 06 EEPROM Interface and Start-Up Behavior* Application Note.

## 9 ac Characteristics of Serial EEPROM Interface Signals

**Table 70. ac Characteristics of Serial EEPROM Interface Signals**

| Symbol                   | Parameter  | Min | Max | Unit |
|--------------------------|--|-----|-----|------|
| f <sub>ROM_CLK</sub>     | Frequency of Serial Clock  | —   | 100 | kHz  |
| t <sub>PW_LOW</sub>      | Width of Serial Clock Pulse Low  | 4.7 | —   | μs   |
| t <sub>PW_HIGH</sub>     | Width of Serial Clock Pulse High   | 4.0 | —   | μs   |
| t <sub>DATA_VALID</sub>  | Time from When Serial Clock Transitions Low Until EEPROM Returns Valid Data    | 0.1 | 4.5 | μs   |
| t <sub>FREE</sub>        | Time I <sup>2</sup> C Bus Must Be Idle Before a New Transaction Can Be Started | 4.7 | —   | μs   |
| t <sub>HOLD_START</sub>  | FW322 Hold Time for a Valid Start Condition                                    | 4.0 | —   | μs   |
| t <sub>SETUP_START</sub> | FW322 Setup Time for a Valid Start Condition                                   | 4.7 | —   | μs   |
| t <sub>HOLD_DATA</sub>   | Data Out Hold Time for the FW322   | 0   | —   | μs   |
| t <sub>SETUP_DATA</sub>  | Data Out Setup Time for the FW322  | 200 | —   | ns   |
| t <sub>RISE_TIME</sub>   | Rise Time for Serial Clock and Data Out from the FW322                         | —   | 1.0 | μs   |
| t <sub>FALL_TIME</sub>   | Fall Time for Serial Clock and Data Out from the FW322                         | —   | 300 | ns   |
| t <sub>SETUP_STOP</sub>  | FW322 Setup Time for a Valid Stop Condition                                    | 4.7 | —   | μs   |
| t <sub>HOLD_EEPROM</sub> | Data Out Hold Time for EEPROM  | 100 | —   | ns   |

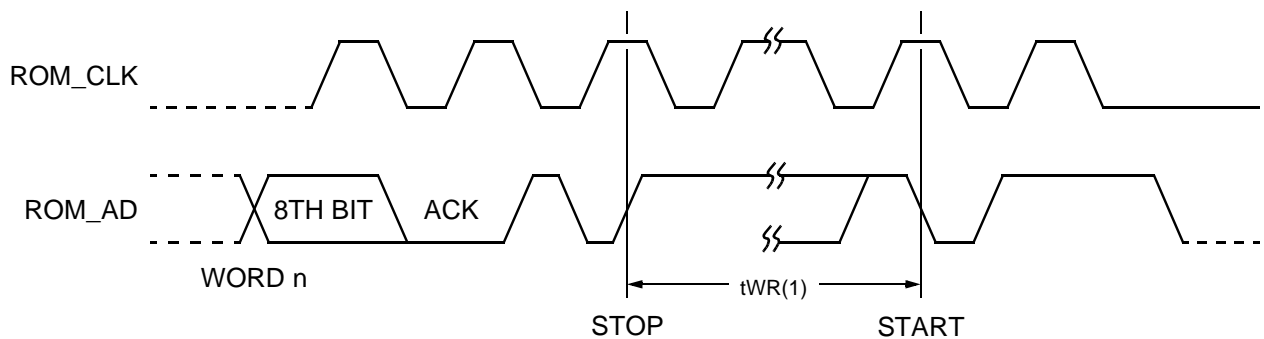
## 9 ac Characteristics (continued)



ROM\_CLK: serial clock, ROM\_AD: serial data I/O.

1313 (F) R.02

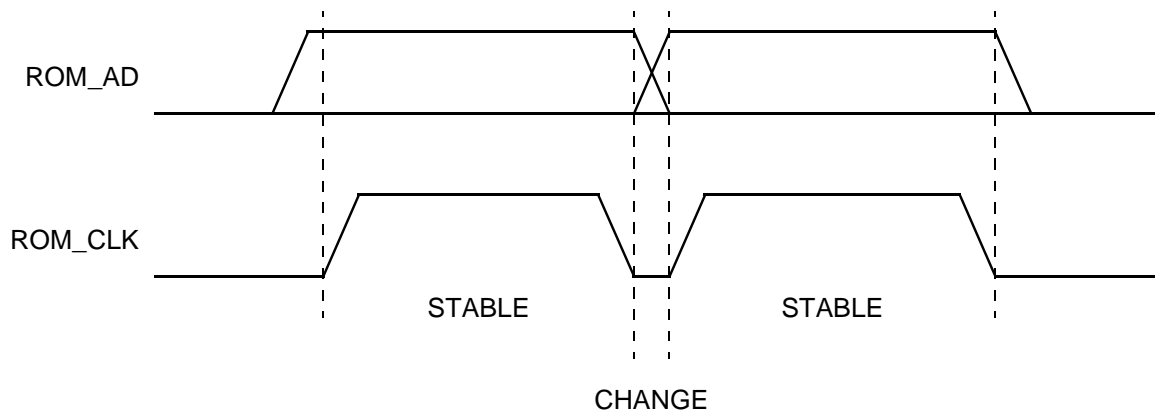
Figure 8. Bus Timing



ROM\_CLK: serial clock, ROM\_AD: serial data I/O.

1314 (F) R.02

Figure 9. Write Cycle Timing

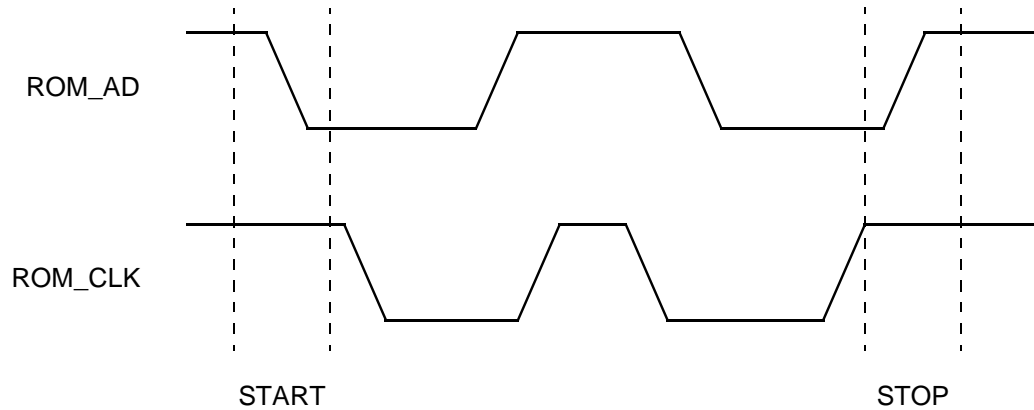


ROM\_CLK: serial clock, ROM\_AD: serial data I/O.

1310 (F) R.02

Figure 10. Data Validity

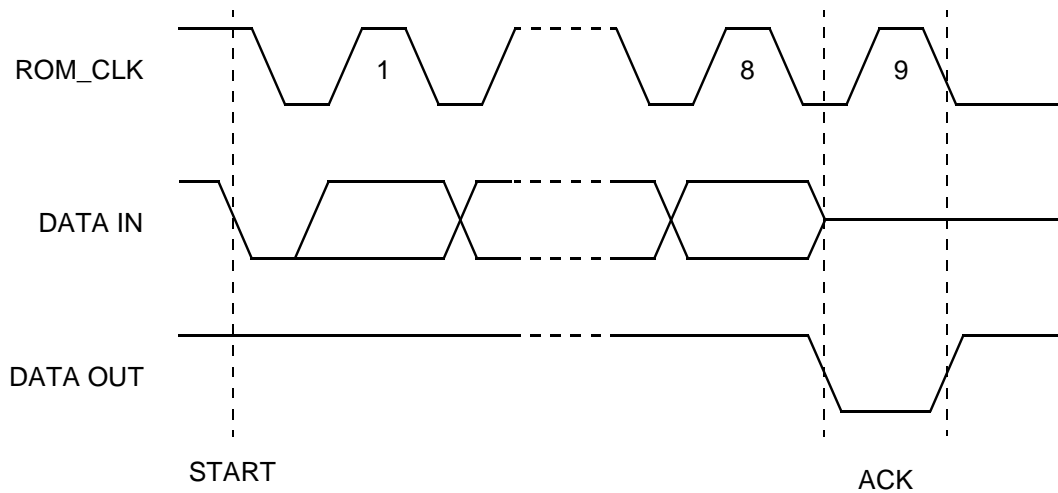
9 ac Characteristics (continued)



ROM\_CLK: serial clock, ROM\_AD: serial data I/O.

1311 (F) R.02

Figure 11. Start and Stop Definition



ROM\_CLK: serial clock.

1312 (F) R.02

Figure 12. Output Acknowledge



## 10 Solder Reflow and Handling

The FW322 has a moisture sensitivity classification of 3, which is determined in accordance with the standard IPC/JEDEC J-STD-020, Revision A, titled *Moisture/Reflow Sensitivity Classification for Nonhermetic Solid State Surface Mount Devices*. Handling of this device should be in accordance with standard IPC/JEDEC J-STD-033, titled *Standard for Handling, Packing, Shipping and Use of Moisture/Reflow Sensitive Surface Mount Devices*.

Up to three reflows may be performed using a temperature profile that meets the requirements of Table 3 in standard IPC/JEDEC J-STD-020. The requirements of IPC/JEDEC J-STD-033 must be met. The maximum allowable body temperature for the FW322 is 220 °C—225 °C. This is the actual tolerance that Agere uses to test the devices during preconditioning.

## 11 Absolute Maximum Voltage/Temperature Ratings

Stresses in excess of the absolute maximum ratings can cause permanent damage to the device. These are absolute stress ratings only. Functional operation of the device is not implied at these or any other conditions in excess of those given in the operational sections of the data sheet. Exposure to absolute maximum ratings for extended periods can adversely affect device reliability.

**Table 71. Absolute Maximum Ratings**

| Parameter                            | Symbol           | Min  | Max                   | Unit |
|--------------------------------------|------------------|------|-----------------------|------|
| Supply Voltage Range                 | V <sub>DD</sub>  | 3.0  | 3.6                   | V    |
| Input Voltage Range                  | V <sub>I</sub>   | −0.5 | V <sub>DD</sub> + 0.5 | V    |
| Output Voltage Range at Any Output   | V <sub>O</sub>   | −0.5 | V <sub>DD</sub> + 0.5 | V    |
| Operating Free Air Temperature       | T <sub>A</sub>   | 0    | 70                    | °C   |
| Storage Temperature Range            | T <sub>stg</sub> | −65  | 150                   | °C   |
| Operating Junction Temperature       | T <sub>JC</sub>  | —    | 125                   | °C   |
| Thermal Resistance (junction to air) | θ <sub>JA</sub>  | —    | 38                    | °C/W |

## 12 Electrical Characteristics

Table 72. Analog Characteristics

| Parameter   | Test Conditions  | Symbol      | Min   | Typ | Max   | Unit |
|---|--|-------------|-------|-----|-------|------|
| Supply Voltage  | Source power node  | VDD—SP      | 3.0   | 3.3 | 3.6   | V    |
| Differential Input Voltage                                    | Cable inputs, 100 Mbits/s operation                        | VID—100     | 142   | —   | 260   | mV   |
|   | Cable inputs, 200 Mbits/s operation                        | VID—200     | 132   | —   | 260   | mV   |
|   | Cable inputs, 400 Mbits/s operation                        | VID—400     | 100   | —   | 260   | mV   |
|   | Cable inputs, during arbitration                           | VID—ARB     | 168   | —   | 265   | mV   |
| Common-mode Voltage<br>Source Power Mode                      | TPB cable inputs,<br>speed signaling off                   | VCM         | 1.165 | —   | 2.515 | V    |
|   | TPB cable inputs,<br>S100 speed signaling on               | VCM—SP—100  | 1.165 | —   | 2.515 | V    |
|   | TPB cable inputs,<br>S200 speed signaling on               | VCM—SP—200  | 0.935 | —   | 2.515 | V    |
|   | TPB cable inputs,<br>S400 speed signaling on               | VCM—SP—400  | 0.532 | —   | 2.515 | V    |
| Common-mode Voltage<br>Nonsource Power Mode*                  | TPB cable inputs,<br>speed signaling off                   | VCM         | 1.165 | —   | 2.015 | V    |
|   | TPB cable inputs,<br>S100 speed signaling on               | VCM—NSP—100 | 1.165 | —   | 2.015 | V    |
|   | TPB cable inputs,<br>S200 speed signaling on               | VCM—NSP—200 | 0.935 | —   | 2.015 | V    |
|   | TPB cable inputs,<br>S400 speed signaling on               | VCM—NSP—400 | 0.532 | —   | 2.015 | V    |
| Receive Input Jitter  | TPA, TPB cable inputs,<br>100 Mbits/s operation            | —           | —     | —   | 1.08  | ns   |
|   | TPA, TPB cable inputs,<br>200 Mbits/s operation            | —           | —     | —   | 0.5   | ns   |
|   | TPA, TPB cable inputs,<br>400 Mbits/s operation            | —           | —     | —   | 0.315 | ns   |
| Receive Input Skew  | Between TPA and TPB cable inputs,<br>100 Mbits/s operation | —           | —     | —   | 0.8   | ns   |
|   | Between TPA and TPB cable inputs,<br>200 Mbits/s operation | —           | —     | —   | 0.55  | ns   |
|   | Between TPA and TPB cable inputs,<br>400 Mbits/s operation | —           | —     | —   | 0.5   | ns   |
| Positive Arbitration<br>Comparator Input<br>Threshold Voltage | —  | VTH+        | 89    | —   | 168   | mV   |
| Negative Arbitration<br>Comparator Input<br>Threshold Voltage | —  | VTH—        | –168  | —   | –89   | mV   |
| Speed Signal Input<br>Threshold Voltage                       | 200 Mbits/s  | VTH—S200    | 45    | —   | 139   | mV   |
|   | 400 Mbits/s  | VTH—S400    | 266   | —   | 445   | mV   |
| Output Current  | TPBIAS outputs   | IO          | –5    | —   | 2.5   | mA   |
| TPBIAS Output Voltage   | At rated I/O current                                       | VO          | 1.665 | —   | 2.015 | V    |
| Current Source for<br>Connect Detect Circuit                  | —  | ICD         | —     | —   | 76    | μA   |

\* For a node that does not source power (see Section 4.2.2.2 in *IEEE 1394-1995 Standard*).

## 12 Electrical Characteristics (continued)

**Table 73. Driver Characteristics**

| Parameter   | Test Conditions                      | Symbol            | Min   | Typ | Max   | Unit |
|---|--------------------------------------|-------------------|-------|-----|-------|------|
| Differential Output Voltage                         | 56 $\Omega$ load                     | V <sub>OD</sub>   | 172   | —   | 265   | mV   |
| Off-state Common-mode Voltage                       | Drivers disabled                     | V <sub>OFF</sub>  | —     | —   | 20    | mV   |
| Driver Differential Current, TPA+, TPA–, TPB+, TPB– | Driver enabled, speed signaling off* | I <sub>DIFF</sub> | –1.05 | —   | 1.05  | mA   |
| Common-mode Speed Signaling Current, TPB+, TPB–     | 200 Mb/s speed signaling enabled     | I <sub>SP</sub>   | –2.53 | —   | –4.84 | mA   |
|   | 400 Mb/s speed signaling enabled     | I <sub>SP</sub>   | –8.1  | —   | –12.4 | mA   |

\* Limits are defined as the algebraic sum of TPA+ and TPA– driver currents. Limits also apply to TPB+ and TPB– as the algebraic sum of driver currents.

**Table 74. Device Characteristics**

| Parameter  | Test Conditions   | Symbol          | Min                   | Typ | Max                 | Unit    |
|--|---|-----------------|-----------------------|-----|---------------------|---------|
| Supply Current*:   |   |                 |                       |     |                     |         |
| D0, 2 Ports Active   | V <sub>DD</sub> = 3.3 V   | I <sub>DD</sub> | —                     | 112 | —                   | mA      |
| D0, 1 Port Active  | V <sub>DD</sub> = 3.3 V   | I <sub>DD</sub> | —                     | 97  | —                   | mA      |
| D0, No Ports Active  | V <sub>DD</sub> = 3.3 V   | I <sub>DD</sub> | —                     | 78  | —                   | mA      |
| System in Standby (suspend mode = S1)  | V <sub>DD</sub> = 3.3 V PCI or V <sub>DD</sub> = 3.3 V <sub>aux</sub> | I <sub>DD</sub> | —                     | 30  | —                   | mA      |
| System in Hibernate (suspend mode = S1 or S3) or System in Standby (suspend mode = S3) | V <sub>DD</sub> = 3.3 V <sub>aux</sub>                                | I <sub>DD</sub> | —                     | 1.3 | —                   | mA      |
| High-level Output Voltage  | I <sub>OH</sub> max, V <sub>DD</sub> = min                            | V <sub>OH</sub> | V <sub>DD</sub> – 0.4 | —   | —                   | V       |
| Low-level Output Voltage   | I <sub>OL</sub> min, V <sub>DD</sub> = max                            | V <sub>OL</sub> | —                     | —   | 0.4                 | V       |
| High-level Input Voltage   | CMOS inputs   | V <sub>IH</sub> | 0.7 V <sub>DD</sub>   | —   | —                   | V       |
| Low-level Input Voltage  | CMOS inputs   | V <sub>IL</sub> | —                     | —   | 0.2 V <sub>DD</sub> | V       |
| Pull-up Current, RESETN Input  | V <sub>I</sub> = 0 V  | I <sub>I</sub>  | 11                    | —   | 32                  | $\mu$ A |

\* This I<sub>DD</sub> value may differ depending on the system board into which the FW322 07 T100 PCI add-in card is inserted.

## 13 Timing Characteristics

**Table 75. Switching Characteristics**

| Symbol | Parameter                     | Measured               | Test Conditions                               | Min | Typ | Max  | Unit |
|--------|-------------------------------|------------------------|---|-----|-----|------|------|
| —      | Jitter, Transmit              | TPA, TPB               | —   | —   | —   | 0.15 | ns   |
| —      | Transmit Skew                 | Between<br>TPA and TPB | —   | —   | —   | ±0.1 | ns   |
| $t_r$  | Rise Time, Transmit (TPA/TPB) | 10% to 90%             | $R_l = 56\ \Omega$ ,<br>$C_l = 10\ \text{pF}$ | —   | —   | 1.2  | ns   |
| $t_f$  | Fall Time, Transmit (TPA/TPB) | 90% to 10%             | $R_l = 56\ \Omega$ ,<br>$C_l = 10\ \text{pF}$ | —   | —   | 1.2  | ns   |

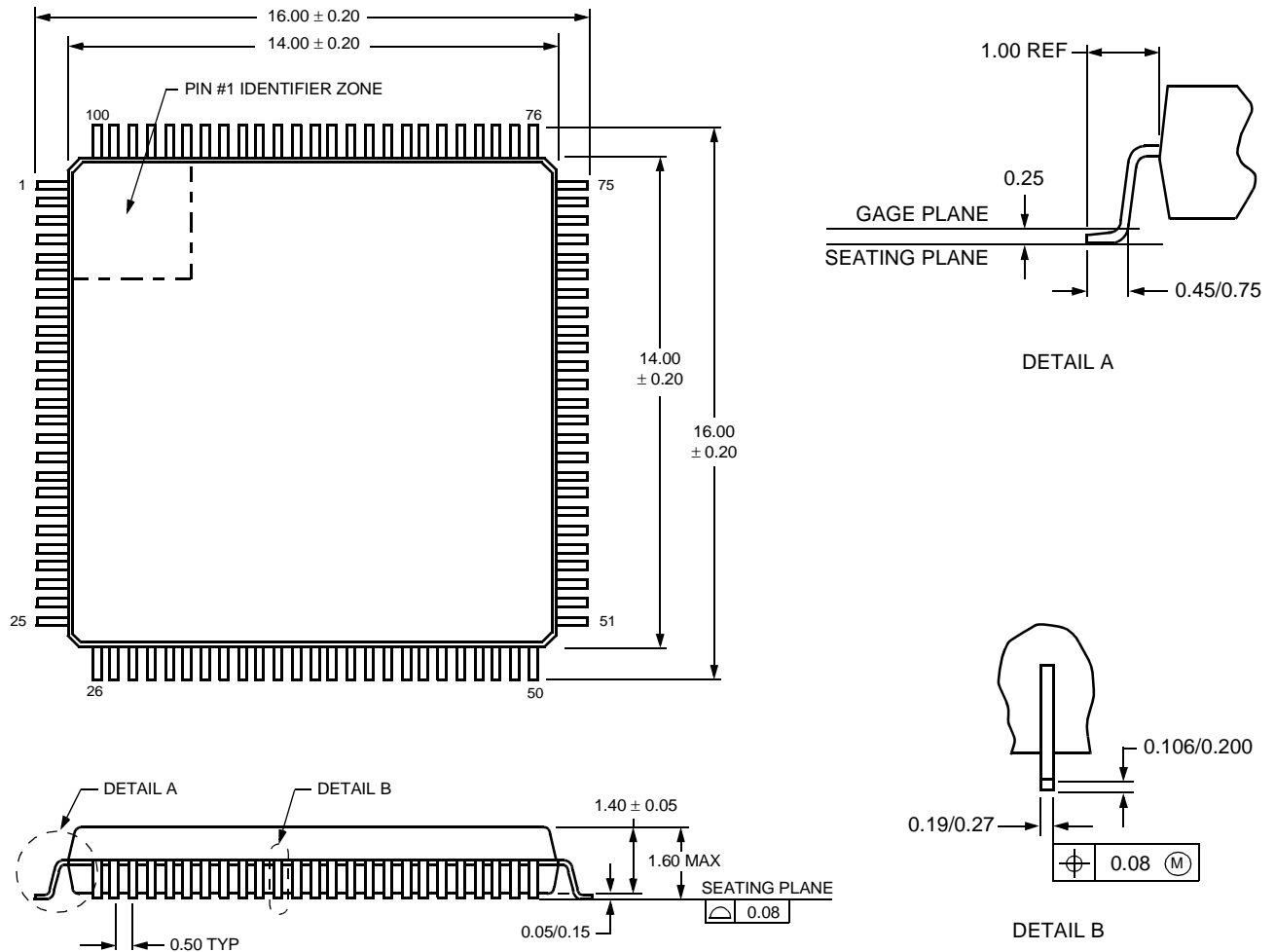
**Table 76. Clock Characteristics**

| Parameter                       | Symbol | Min     | Typ     | Max     | Unit |
|---------------------------------|--------|---------|---------|---------|------|
| External Clock Source Frequency | f      | 24.5735 | 24.5760 | 24.5785 | MHz  |

## 14 Outline Diagrams

### 14.1 100-Pin TQFP

Dimensions are in millimeters.



5-2146 (F)

## 15 Ordering Information

| Device Code          | Package      | Comcode   |
|----------------------|--------------|-----------|
| L-FW322 07 T100-DB*† | 100-Pin TQFP | 711008303 |
| L-FW322 07 T100-DT*† | 100-Pin TQFP | 711008304 |

\* Lead-free: No intentional addition of lead, and less than 1000 ppm.

† Agere Systems lead-free devices are fully compliant with the Restriction of Hazardous Substances (RoHS) directive that restricts the content of six hazardous substances in electronic equipment in the European Union. Beginning July 1, 2006, electronic equipment sold in the European Union must be manufactured in accordance with the standards set by the RoHS directive.

*Microsoft* and *Windows* are registered trademarks of Microsoft Corporation.

The *FireWire* logo is a trademark and *MacOS* is a registered trademark of Apple Computer, Inc.

*I.Link* is a registered trademark of Sony Corporation.

*IEEE* is a registered trademark of The Institute of Electrical and Electronics Engineers, Inc.

*Mini PCI* is a registered trademark of PCI-SIG, Inc.

---

For additional information, contact your Agere Systems Account Manager or the following:

INTERNET: Home: <http://www.agere.com> Sales: <http://www.agere.com/sales>

E-MAIL: [docmaster@agere.com](mailto:docmaster@agere.com)

N. AMERICA: Agere Systems Inc., Lehigh Valley Central Campus, Room 10A-301C, 1110 American Parkway NE, Allentown, PA 18109-9138

**1-800-372-2447**, FAX 610-712-4106 (In CANADA: **1-800-553-2448**, FAX 610-712-4106)

ASIA: CHINA: **(86) 21-54614688** (Shanghai), **(86) 755-25881122** (Shenzhen), **(86) 10-65391096** (Beijing)

JAPAN: **(81) 3-5421-1600** (Tokyo), KOREA: **(82) 2-767-1850** (Seoul), SINGAPORE: **(65) 6741-9855**, TAIWAN: **(886) 2-2725-5858** (Taipei)

EUROPE: Tel. **(44) 1344 865 900**

---

Agere Systems Inc. reserves the right to make changes to the product(s) or information contained herein without notice. No liability is assumed as a result of their use or application. Agere, Agere Systems, and the Agere logo are registered trademarks of Agere Systems Inc.