agere systems

Data Sheet
November 2005

FW321 NV129 1394a
PCI PHY/Link Open Host Controller Interface

1 Features

- 129-ball VTFSBGA lead-free package.
- 1394a-2000 OHCI link and PHY core function in a single device:
  - — Single-chip link and PHY enable smaller, simpler, more efficient motherboard and add-in card designs.
  - — Compatibility with current *Microsoft Windows*® drivers and common applications.
  - — Interoperability with existing, as well as older, 1394 consumer electronics and peripherals products.
  - — Support low-power system designs (CMOS implementation and power management features).
  - — LPS, LKON, and CNA outputs to support legacy power management implementations.
- OHCI:
  - — Complies with the *1394 OHCI 1.1 Specification*.
  - — OHCI 1.0 backwards compatible: configurable via PCI bus commands to operate in either OHCI 1.0 or OHCI 1.1 mode.
  - — Listed on *Windows* hardware compatibility list http://www.microsoft.com/hcl/results.asp.
  - — 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/receive contexts.
  - — Prefetches isochronous transmit data.
  - — Supports posted write transactions.
  - — Supports parallel processing of incoming physical read and write requests.
  - — May be used without an EEPROM when the system BIOS is programmed with the EEPROM contents.
- 1394a-2000 PHY core:
  - — Compliant with *IEEE*® *1394*a-2000, *Standard for a High Performance Serial Bus*.
  - — Provides one fully compliant cable port, supporting 400 Mbits/s, 200 Mbits/s, and 100 Mbits/s traffic.
  - — Does not require external filter capacitor for PLL.
  - — Supports link-on as a part of the internal PHY core-link interface.
  - — Supports arbitrated short bus reset to improve utilization of the bus.
  - — 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.
- PCI:
  - — Revision 2.2 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.
  - — Supports CLKRUN# protocol per PCI Mobile Design Guide.
  - — Supports *Mini PCI Specification* v1.0, including *Mini PCI*® power requirements.
  - — CardBus support per PC card standard release 8.0, including 128 bytes of on-chip tuple memory.

1.1 Other Features

- CMOS process.
- 3.3 V operation, 5 V tolerant inputs.
- I^2C serial ROM interface.

Table of Contents

Contents

Page

Table of Contents (continued)

Contents **Page**

List of Figures

Figure **Page**

List of Tables

Table **Page**

List of Tables (continued)

Table **Page**

2 FW321 NV129 Functional Overview

Agere Systems' FW321 NV129 is a high-performance PCI bus-based open host controller for implementation of *IEEE* 1394a compliant system or device.

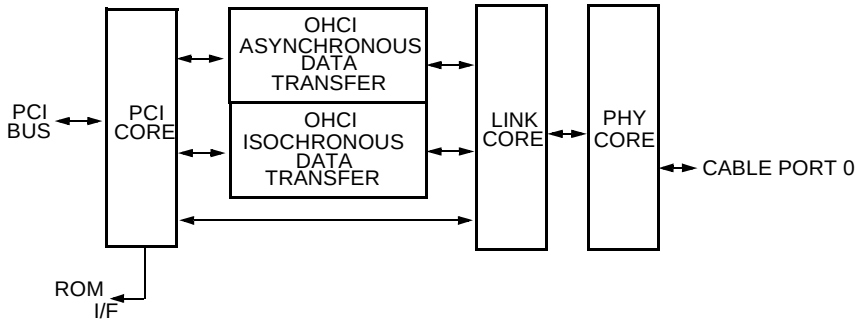


Figure 1. FW321 NV129 1394a Mode Block Diagram

3 FW321 Functional Description

The FW321 is comprised of four major functional sections (see Figure 2): 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.

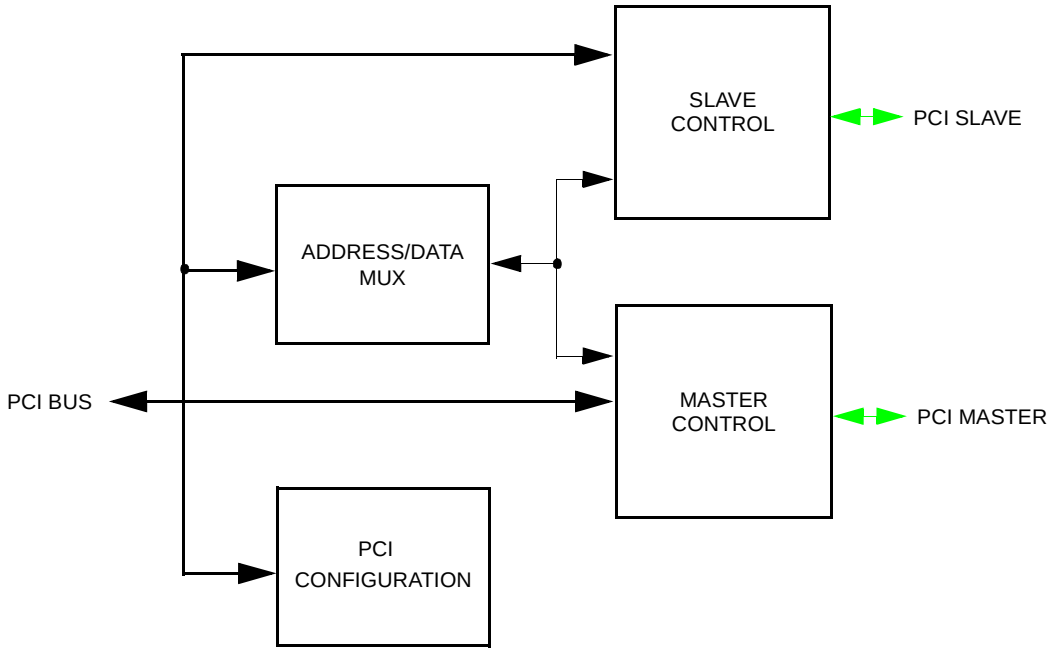


Figure 2. PCI Core Block Diagram

3 FW321 Functional Description (continued)

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 FW321 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 FW321 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 FW321 into four distinct power states (D0, D1, D2, and D3hot). This permits software to selectively increase/decrease the power consumption of the FW321 for reasons such as periods of system inactivity or power conservation. In addition, the FW321 also includes support for waking up the system through the generation of a power management event (PME).

The FW321 supports generation of a power management event (PME) while in the D0, D1, D2, and D3hot power states.

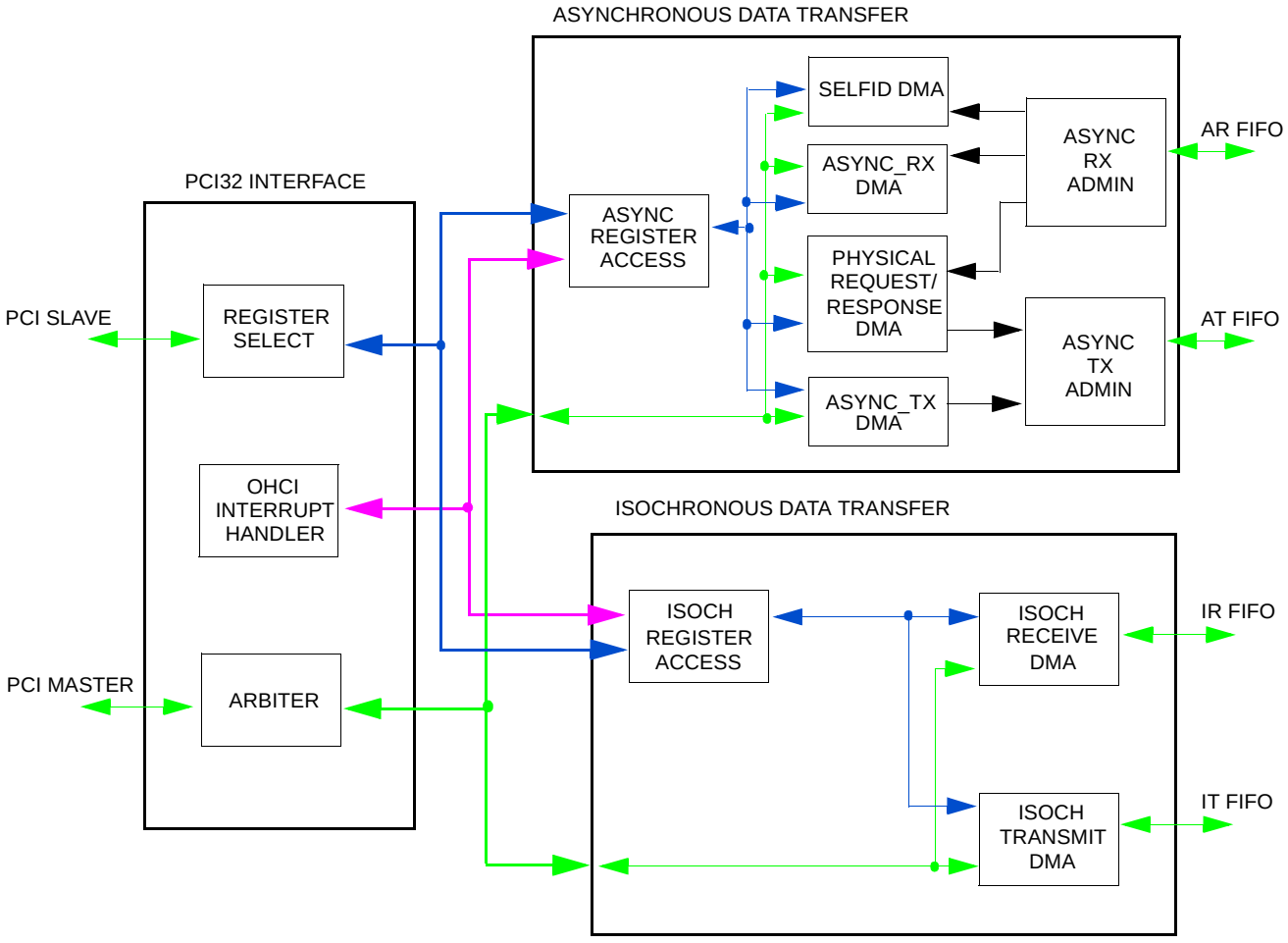


Figure 3. OHCI Core Block Diagram

3 FW321 Functional Description (continued)

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.

3.4 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 FW321 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.

3 FW321 Functional Description (continued)

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.

3.6.1 Asynchronous Transmit DMA (ASYNC_TX DMA, ASYNC_TX_ADMIN)

The ASYNC_TX DMA and ASYNC_TX_ADMIN blocks of the FW321 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 FW321 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 FW321 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.

3 FW321 Functional Description (continued)

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 FW321 is responsible for processing incoming physical requests and outgoing physical responses. When an incoming asynchronous packet is received, the FW321 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 FW321 to either receive or transmit an asynchronous packet without the use of DMA descriptors. Instead, the FW321 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 FW321 is responsible for receiving SelfID packets during the bus initialization process. The received SelfID packets are written into a software-defined host memory buffer.

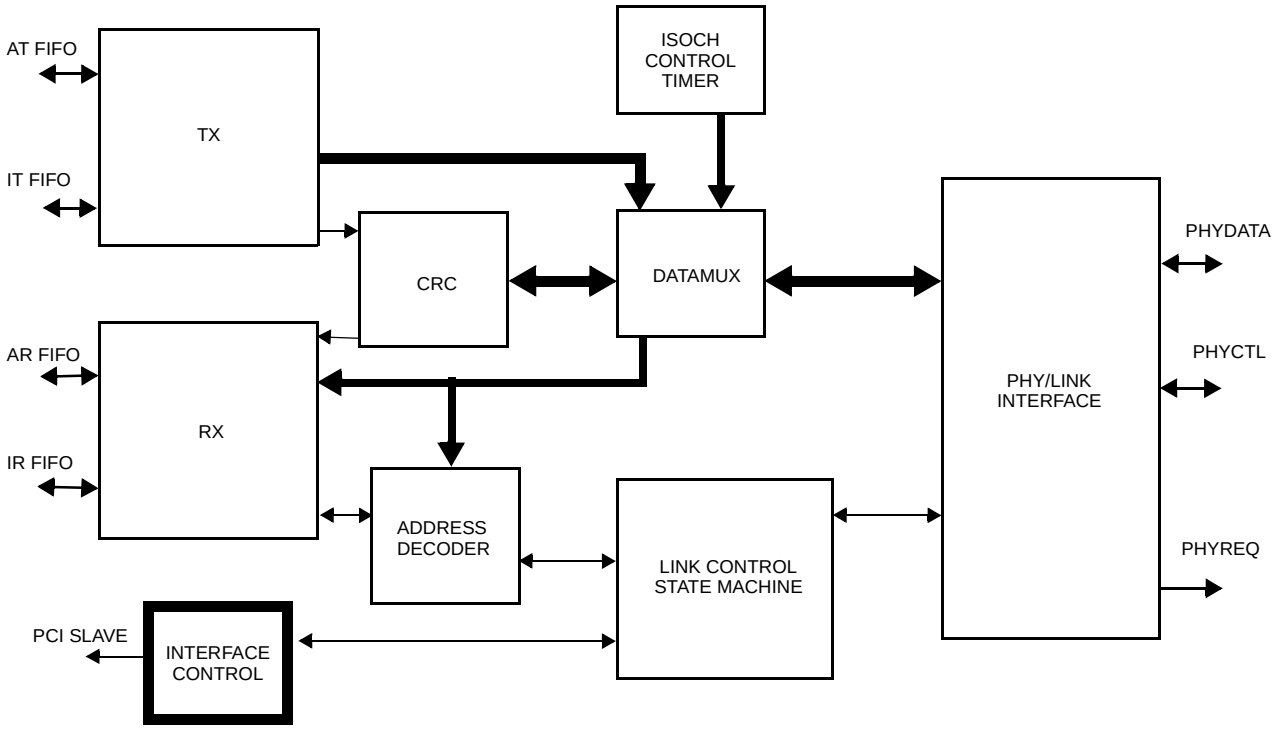


Figure 4. Link Core Block Diagram

3 FW321 Functional Description (continued)

3.7 Link Core

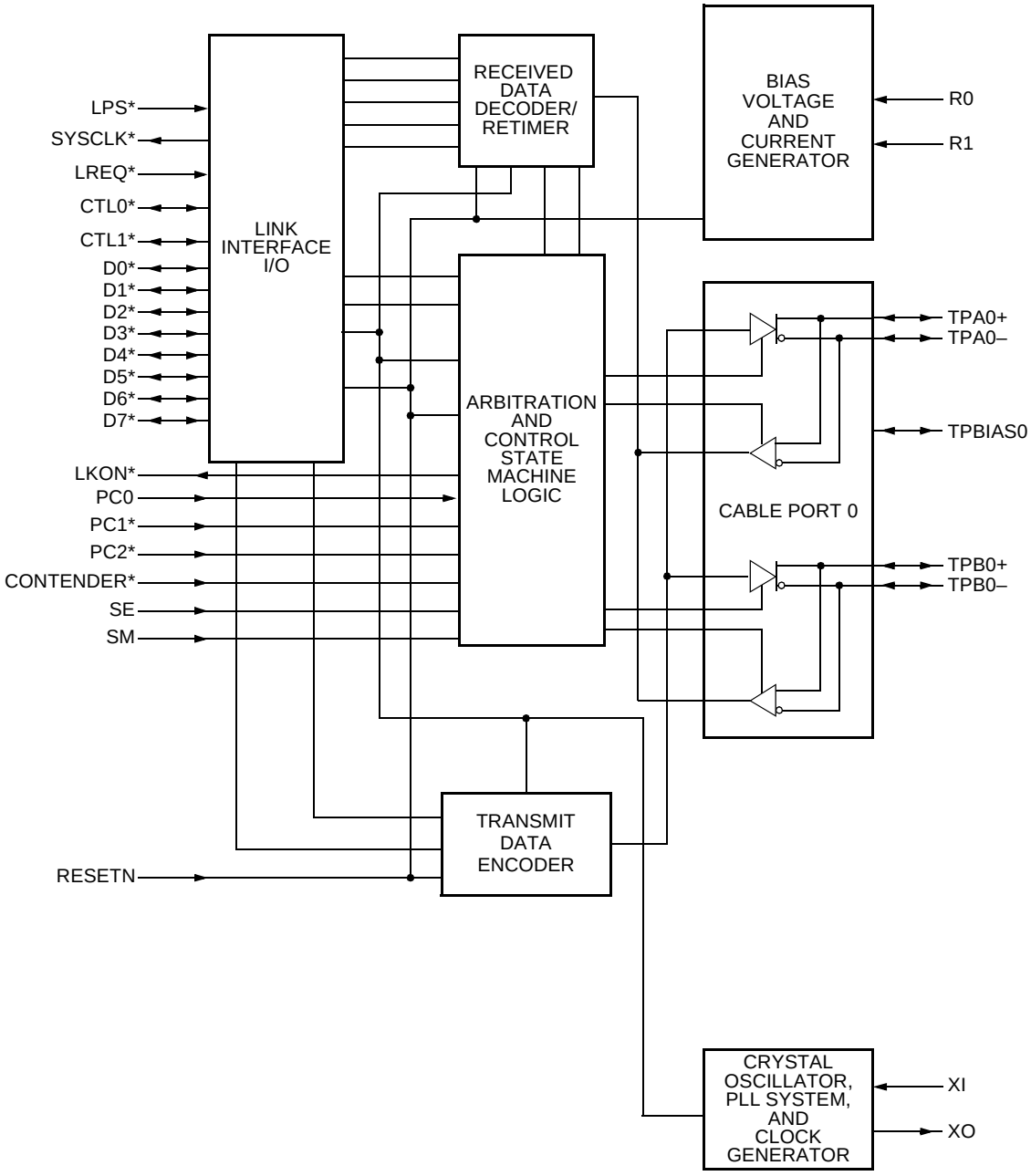
The link core shown in Figure 4 consists of the following blocks:

- 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.

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.

3 FW321 Functional Description (continued)



* Internal points, inaccessible via external package pins.

Figure 5. The PHY Core Block Diagram

3 FW321 Functional Description (continued)

3.8 PHY/Link Interface

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.

The internal link power status (L:S) signal works with the internal LinkOn signal to manage the LLC power usage of the node. The LPS signal indicates that the LLC of the node is powered up or down. If LPS is inactive for more than 1.2 µs and less than 25 µs, the internal PHY/link interface is reset.

If LPS is inactive for greater than 25 ms, the PHY disables the internal PHY/link interface to save power. The FW321 NV129 continues its repeater function. If the PHY receives a link-on packet, the internal LinkOn signal is activated to output a 6.114 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. Internal LinkOn signal is turned off when LCtrl bit is set.

3.9 PHY Core

The PHY core in Figure 5 on the preceding page 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) that 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.

3 FW321 Functional Description (continued)

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 µF.

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

The line drivers in the PHY core operate in a high-impedance current mode and are designed to work with external 112 Ω 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 Ω 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 kΩ and 220 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 kΩ ± 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 PC0 pin of the device can be tied to V_{SS} to set the power class to 0, or it can be tied to V_{DD} to set the power class to 4. The PC1 and PC2 bits (two least significant bits of the pwr_class field of the SelfID packet) are internally tied to 0.

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±/TPB± 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 FW321 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.

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 µs and less than 25 µs, the internal PHY/link interface is reset.

If LPS is inactive for greater than 25 µs, the PHY will disable the internal PHY/link interface to save power. The FW321 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 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.

Five of the FW321 pins are used to set up various test conditions used only during the device manufacturing process. These pins are SE, SM, TEST0, TEST1, and TEST2.

4 Ball Information

13	12	11	10	9	8	7	6	5	4	3	2	1	
$V_{SS}A$	NC	NC	TPB0–	TPA0+	$V_{DD}A$	$V_{DD}A$	R[1]	XO	SM	VAUX_ PRESENT	V_{DD}	V_{SS}	A
NC	NC	NC	TPB0+	TPA0–	TPBIAS0	R[0]	$PLLV_{DD}$	XI	PTEST	SE	V_{SS}	CARD-BUSN	B
NC	NC	NC	—	—	—	—	—	—	—	V_{SS}	CNA	NAND TREE	C
$V_{DD}A$	CPS	—	NC	$V_{SS}A$	NC	$V_{SS}A$	$PLLV_{SS}$	RESETN	V_{SS}	—	ROM_ CLK	ROM_ AD	D
MPCI ACTN_ 321	LPS	—	$V_{SS}A$	$V_{DD}A$	—	—	—	V_{SS}	TEST1	—	TEST0	PCI INTAN	E
PC0	PC1	—	LKON	—	V_{SS}	V_{SS}	V_{SS}	—	CLK RUNN	—	PCI GNTN	PCI RSTN	F
PCI VIOS	CONTEN DER	—	PC2	—	V_{SS}	V_{SS}	V_{SS}	—	PCI PMEN	—	V_{DD}	PCI REQN	G
PCI AD[1]	V_{DD}	—	PCI AD[0]	—	V_{SS}	V_{SS}	V_{SS}	—	PCICLK	—	PCI AD[30]	PCI AD[31]	H
PCI AD[4]	PCI AD[3]	—	PCI AD[2]	PCI AD[5]	—	—	—	V_{SS}	PCI AD[29]	—	PCI AD[27]	PCI AD[28]	J
PCI AD[6]	PCI CBEN[0]	—	PCI AD[7]	V_{SS}	V_{SS}	V_{DD}	V_{DD}	V_{DD}	PCI AD[25]	—	PCI AD[24]	PCI AD[26]	K
V_{DD}	PCI AD[8]	PCI AD[10]	—	—	—	—	—	—	—	PCI AD[22]	PCI IDSEL	PCI CBEN[3]	L
PCI AD[9]	PCI AD[11]	PCI AD[12]	PCI AD[14]	PCI CBEN[1]	PCI PERRN	PCI STOPN	PCI IRDYN	PCI CBEN[2]	PCI AD[16]	PCI AD[19]	PCI AD[20]	PCI AD[23]	M
V_{SS}	PCI AD[13]	PCI AD[15]	PCIPAR	PCI SERRN	PCI DEVSELN	PCI TRDYN	PCI FRAMEN	PCI AD[17]	PCI AD[18]	PCI AD[21]	V_{DD}	V_{DD}	N

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

Figure 6. Ball Assignments for the FW321 NV129

4 Ball Information (continued)

Table 1. Cable Ports

Signal Listing	Pin Name	Type	Name/Description
A10	TPB0–	Analog I/O	**Port 0, Port Cable Pair 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 FW321'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.
B10	TPB0+		
B9	TPA0–	Analog I/O	**Port 0, Port Cable Pair A.** 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 FW321'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.
A9	TPA0+		
D12	CPS	I	**Cable Power Status.** 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 *IEEE* 1394a-2000, *Standard for a High Performance Serial Bus*, Sections 4.2.2.7 and 5B.1). **Note:** 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 0101_2 will set.
B8	TPBIAS0	Analog I/O	**Port 0, Twisted-Pair Bias.** 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 FW321'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.

4 Ball Information (continued)

Table 2. PCI Signals

Signal Listing	Pin Name	Type	Name/Description
H1, H2, H10, H13, J1, J2, J4, J9, J10, J12, J13, L3, L11, L12, K1, K2, K4, K10, K13, M1, M2, M3, M4, M10, M11, M12, M13, N3, N4, N5, N11, N12	PCIAD[31:0]	I/O	**PCI Address/Data Bit.**
L1, M5, M9, K12	PCICBEN[3:0]	I/O	**PCI Command/Byte Enable Signal (Active-Low).**
H4	PCICLK	I	**PCI Clock Input. 33 MHz.**
N8	PCIDEVSELN	I/O	**PCI Device Select Signal (Active-Low).**
N6	PCIFRAMEN	I/O	**PCI Frame Signal (Active-Low).**
F2	PCIGNTN	I	**PCI Grant Signal (Active-Low).**
L2	PCIIDSEL	I	**PCI ID Select.**
E1	PCIINTAN	O	**PCI Interrupt (Active-Low).**
M6	PCIIRDYN	I/O	**PCI Initiator Ready Signal (Active-Low).**
N10	PCIPAR	I/O	**PCI Parity Signal.**
M8	PCIPERRN	I/O	**PCI Parity Error Signal (Active-Low).**
G4	PCIPMEN	O	**PCI Power Management Event (Active-Low).** A PCI power management event will be indicated if this signal is low.
G1	PCIREQN	O	**PCI Request Signal (Active-Low).**
F1	PCIRSTN	I	**PCI Reset (Active-Low).**
N9	PCISERRN	I/O	**PCI System Error Signal (Active-Low).**
M7	PCISTOPN	I/O	**PCI Stop Signal (Active-Low).**
N7	PCITRDYN	I/O	**PCI Target Ready Signal (Active-Low).**
G13	PCIVIOS	—	**PCI Signaling Indicator.** For PCI applications that use a universal expansion board (see *PCI Local Bus Specification*, Rev. 2.2, 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.
B1	CARDBUSN	I	**CardBusN (Active-Low).** Selects mode of operation for PCI output buffers. Connect this pin to ground for CardBus operation; connect to V_{DD} for PCI operation.
E13	MPCIACTN_321	O	**Mini PCI Function Active.** Output is either asserted or tristated. If output is 0, chip is in a communication state and requires full system performance. Otherwise, the output is tristated (active-low).
F4	CLKRUNN	I/O	**CLKRUNN (Active-Low).** Optional signal for PCI mobile computing environment. If not used, CLKRUNN pin needs to be pulled down to V_{SS} for correct operation.

4 Ball Information (continued)

Table 3. Test, Reset, Clock, and Configuration Signals

Signal Listing	Pin Name	Type	Name/Description
B4	PTEST	I	**Test.** Used by Agere for device manufacturing testing. Tie to VSS for normal operation.
D1	ROM_AD	I/O	**ROM Address/Data.**
F10	LKON	O	**Link On.** Signal from the internal PHY core to the internal link core. This signal is provided as an output for use in legacy power management systems.
E12	LPS	O	**Link Power Status.** Signal from the internal link core to the internal PHY core. LPS is provided as an output for use in legacy power management systems.
G12	CONTENDER	I	**Contender.** On hardware reset (RESETN), this input sets the default value of the CONTENDER bit indicated during SelfID. This bit can be tied to VDD (high), so it will be considered for bus manager or to ground (low) to not be considered for bus manager.
G10 F12 F13	PC2 PC1 PC0	I	**Power-Class Indicators.** On hardware reset (RESETN), these inputs set the default value of the power class indicated during SelfID. These bits can be tied to VDD (high) or to ground (low) as required for particular power consumption and source characteristics. In SelfID packet (see Section 4.3.4.1 of the 1394a-2000 specification), PC0, the most significant bit of this 3-bit field, corresponds to bit 21, PC1 corresponds to bit 22, and PC2 corresponds to bit 23. As an example, for a Power_Class value of 001, PC0 = 0, PC1 = 0, and PC2 = 1.
A6 B7	R[1] R[0]	I	**Current Setting Resistor.** 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.
D5	RESETN	I	**Reset (Active-Low).** When RESETN is asserted low (active), a 1394 bus reset condition is set on the active cable ports and the FW321 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 µF) and resistor (510 kΩ), 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.
A4	SM	I	**Test Mode Control.** SM is used during Agere's manufacturing test and should be tied to VSS for normal operation.
B3	SE	I	**Test Mode Control.** SE is used during Agere's manufacturing test and should be tied to VSS for normal operation.
C2	CNA	O	**Cable Not Active.** CNA output is provided for use in legacy power management systems. CNA is asserted high when none of the PHY ports is receiving an incoming bias voltage. This circuit remains active during the powerdown mode. The CNA pin is TTL-compatible. This pin can source and sink up to a 6 mA load.
C1	NANDTREE	O	**NAND Tree Test Output.** When the chip is placed into the NAND tree test mode, the pin is the output of the NAND tree logic. This pin is not used during normal operation.
D2	ROM_CLK	I/O	**ROM Clock.**

4 Ball Information (continued)

Table 3. Test, Reset, Clock, and Configuration Signals (continued)

Signal Listing	Pin Name	Type	Name/Description
E2	TEST0	I	**Test**. Used by Agere for device manufacturing testing. Tie to VSS for normal operation.
E4	TEST1	I	**Test**. Used by Agere for device manufacturing testing. Tie to VSS for normal operation.
B5	XI	Analog I/O	**Crystal Oscillator.** 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 Ω. 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 *FW323/ FW322 Hardware Implementation Design Guideline* Application Note.
A5	XO		

Table 4. Power Signals

Signal Listing	Pin Name	Type	Name/Description
A1, B2, C3, D4, E5, F6, F7, F8, G6, G7, G8, H6, H7, H8, J5, K8, K9, N13	VSS	—	**Digital Ground.**
A2, G2, H12, K5, K6, K7, L13, N1, N2	VDD	—	**Digital Power.**
A7, A8, D13, E9	VDDA	—	**Analog Circuit Power.** VDDA supplies power to the analog portion of the device.
A13, D9, D7, E10	VSSA	—	**Analog Circuit Ground.** All VSSA signals should be tied together to a low-impedance ground plane.
B6	PLLVDD	—	**Power for PLL Circuit.** PLLVDD supplies power to the PLL circuitry portion of the device.
D6	PLLVSS	—	**Ground for PLL Circuit.** PLLVSS is tied to a low-impedance ground plane.
A3	VAUX_PRESENT	I	**3.3 VAUX Present.** An active-high input indicating whether the FW321 is powered via an auxiliary power supply (e.g., PCI 3.3 VAUX). An internal pull-down resistor connected to VSS is provided, so an external pull-up is only required when the device is being powered by an auxiliary power supply. Note that VAUX_PRESENT is not an actual power supply pin to the device. Rather, this pin is an indicator of whether the FW321 is powered via an auxiliary power supply (VAUX_PRESENT = 1) or the regular PCI power supply (VAUX_PRESENT = 0). This input is used by the FW321 to properly support the D3cold power management functionality.
A11, A12, B11, B12, B13, C11, C12, C13, D8, D10	NC	—	**No Connect.**

5 Internal Registers

This section provides a summary of the internal registers within the FW321, including both PCI configuration registers and OHCI registers. Register default values, registers, and bits that have not been implemented in the FW321, and other information specific to the FW321 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 5 describes the field access tags that are designated in the **Type** column of the register tables in this document.

Table 5. 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 FW321.

5.1 PCI Configuration Registers

Table 6 illustrates the PCI configuration header that includes both the predefined portion of the configuration space and the user-definable registers.

Table 6. 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 [6xh]*	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 Management Capabilities Pointer [44h]	34h
Reserved				38h
Maximum Latency† [18h]	Minimum Grant† [0Ch]	Interrupt Pin [01h]	Interrupt Line [00h]	3Ch

* x is a minor revision number of the FW321 T100 and may be any value from 0 hex to F hex.
† The hardware default value for this register can be altered with a PCI bus command.

5 Internal Registers (continued)

Table 6. 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

* x is a minor revision number of the FW321 T100 and may be any value from 0 hex to F hex.
† The hardware default value for this register can be altered with a PCI bus command.

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. 2.2, Section 6.2.1

5.3 Device ID Register

The device ID register contains a value assigned to the FW321 by Agere. The device identification for the FW321 is 5811h.

Offset: 02h
Default: 5811h
Type: Read only
Reference: *PCI Local Bus Specification*, Rev. 2.2, Section 6.2.1

5 Internal Registers (continued)

5.4 PCI Command Register

The command register provides control over the FW321 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. 2.2, Section 6.2.2 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.1

Table 7. PCI Command Register Description

Bit	Field Name	Type	Description
15:10	Reserved	R	**Reserved.** Bits 15:10 return 0s when read.
9	FBB_ENB	R	**Fast Back-to-Back Enable.** The FW321 does not generate fast back-to-back transactions; thus, this bit returns 0 when read.
8	SERR_ENB	RW	**SERR Enable.** When this bit is set, the FW321 SERR driver is enabled. PCI_SERRN can be asserted after detecting an address parity error on the PCI bus.
7	STEP_ENB	R	**Address/Data Stepping Control.** The FW321 does not support address/data stepping; thus, this bit is hardwired to 0.
6	PERR_ENB	RW	**Parity Error Enable.** When this bit is set, the FW321 is enabled to drive PERR response to parity errors through the PCI_PERRN signal.
5	VGA_ENB	R	**VGA Palette Snoop Enable.** The FW321 does not feature VGA palette snooping. This bit returns 0 when read.
4	MWI_ENB	RW	**Memory Write and Invalidate Enable.** When this bit is set, the FW321 is enabled to generate MWI PCI bus commands. If this bit is reset, then the FW321 generates memory write commands instead.
3	SPECIAL	R	**Special Cycle Enable.** The FW321 function does not respond to special cycle transactions. This bit returns 0 when read.
2	MASTER_ENB	RW	**Bus Master Enable.** When this bit is set, the FW321 is enabled to initiate cycles on the PCI bus.
1	MEMORY_ENB	RW	**Memory Response Enable.** Setting this bit enables the FW321 to respond to memory cycles on the PCI bus. This bit must be set to access OHCI registers.
0	IO_ENB	R	**I/O Space Enable**. The FW321 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. 2.2, Section 6.2.3 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.2

Table 8. PCI Status Register

Bit	Field Name	Type	Description
15	PAR_ERR	RCU	**Detected Parity Error.** 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	**Signaled System Error.** This bit must be set whenever the device asserts SERR#.
13	MABORT	RCU	**Received Master Abort.** 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	**Received Target Abort.** This bit must be set by a master device whenever its transaction is terminated with target-abort.
11	TABORT_SIG	RCU	**Signaled Target Abort.** This bit must be set by a target device whenever it terminates a transaction with target-abort.
10:9	PCI_SPEED	R	**DEVSEL Timing.** 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	**Master Data Parity Error.** See Table 6-2 of the PCI Specification for more information.
7	FBB_CAP	R	**Fast Back-to-Back Capable.** This bit 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 FW321 does not support back-to-back transactions.
6	Reserved	R	**Reserved**.
5	66MHZ	R	**66 MHz Capable.** This bit indicates whether or not this device is capable of running at 66 MHz as defined in Chapter 7 of the PCI specification. The FW321 reports a value of zero in this field indicating that 66 MHz functionality is not supported.
4	CAPLIST	R	**Capabilities List.** This bit 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.)
3:0	Reserved	R	**Reserved**.

5 Internal Registers (continued)

5.6 Class Code and Revision ID Registers

The class code register and revision ID register categorize the FW321 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 106xh
Type: Read only
Reference: *PCI Local Bus Specification*, Rev. 2.2, Section 6.2.1 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.3 and A.3.4

Table 9. Class Code and Revision ID Register Description

Bit	Field Name	Type	Description
31:24	BASECLASS	R	**Base Class.** This field returns 0Ch when read, which classifies the function as a serial bus controller.
23:16	SUBCLASS	R	**Subclass.** This field returns 00h when read, which specifically classifies the function as an *IEEE* 1394 serial bus controller.
15:8	PGMIF	R	**Programming Interface.** This field returns 10h when read, indicating that the programming model is compliant with the *1394 Open Host Controller Interface Specification*.
7:0	CHIPREV	R	**Silicon Revision.** This field returns 6xh* when read, indicating the silicon revision of the FW321.

* x is a minor revision number of the FW321 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 FW321. On powerup, the hardware default value will be loaded, but an alternative value may be loaded using a PCI bus command.

Offset: 0Ch
Default: 0000h
Type: Read/write
Reference: *PCI Local Bus Specification*, Rev. 2.2, Section 6.2.4

Table 10. Latency Timer and Class Cache Line Size Register Description

Bit	Field Name	Type	Description
15:8	LATENCY_TIMER	RW	**PCI Latency Timer.** The value in this register specifies the latency timer, in units of PCI clock cycles, for the FW321. When the FW321 is a PCI bus initiator and asserts FRAME, the latency timer begins counting from zero. If the latency timer expires before the FW321 transaction has terminated, then the FW321 terminates the transaction when its PCI_GNTN is deasserted.
7:0	CACHELINE_SZ	RW	**Cache Line Size.** This value is used by the FW321 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 FW321 PCI header type.

Offset: 0Eh
Default: 0000h
Type: Read only
Reference: *PCI Local Bus Specification*, Rev. 2.2, Sections 6.2.1 and 6.2.4

Table 11. Header Type and BIST Register Description

Bit	Field Name	Type	Description
15:8	BIST	R	**Built-In Self-Test.** The FW321 does not include a built-in self-test; thus, this field returns 00h when read.
7:0	HEADER_TYPE	R	**PCI Header Type.** The FW321 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. 2.2, Sections 6.2.5 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.5

Table 12. OHCI Base Address Register Description

Bit	Field Name	Type	Description
31:12	OHCIREG_PTR	RW	**OHCI Register Pointer.** This field specifies the upper 20 bits of the 32-bit OHCI base address.
11:4	OHCI_SZ	R	**OHCI Register Size.** This field returns 0s when read, indicating that the OHCI registers require a 4 Kbyte region of memory.
3	OHCI_PF	R	**OHCI Register Prefetch.** This bit returns 0 when read, indicating that the OHCI registers are not prefetchable.
2:1	OHCI_MEMTYPE	R	**OHCI Memory Type.** 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	**OHCI Memory Indicator.** 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 FW321 resides. Subsystem vendor IDs can be obtained from the PCI SIG. Values for the subsystem ID are vendor specific.

On powerup, the hardware default value will be loaded. However, a nonzero value should be loaded into this register by the BIOS, operating system, etc., using a PCI bus command. By default, the PCI subsystem ID and PCI subsystem vendor ID registers are read-only. In order to load this register, first set the SubSystemWriteEn bit, bit 0 of the PCI configuration register offset 4Ch, which will enable writes to the PCI subsystem ID and PCI subsystem vendor ID registers. 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. 2.2, Section 6.2.4

Table 13. PCI Subsystem Identification Register Description

Bit	Field Name	Type	Description
31:16	SSID	RU	**Subsystem ID.** This field indicates the subsystem ID.
15:0	SSVID	RU	**Subsystem Vendor ID.** 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 FW321 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. 2.2, Section 6.2.4 and 6.7 and *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.6

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. 2.2, Section 6.2.4 and 6.7

Table 14. Interrupt Line and Pin Register Description

Bit	Field Name	Type	Description
15:8	INTR_PIN	R	**Interrupt Pin Register.** This register returns 01h when read, indicating that the FW321 PCI function signals interrupts on the INTA pin.
7:0	INTR_LINE	RW	**Interrupt Line Register.** This register is programmed by the system and indicates to software to which interrupt line the FW321 INTA is connected.

5 Internal Registers (continued)

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. On powerup, the hardware default value will be loaded, but an alternative value may be loaded using a PCI bus command.

Offset: 3Eh
Default: 180Ch
Type: Read only
Reference: *PCI Local Bus Specification*, Rev. 2.2, Section 6.2.4

Table 15. MIN_GNT and MAX_LAT Register Description

Bit	Field Name	Type	Description
15:8	MAX_LAT	RU	**Maximum Latency.** The contents of this register may be used by host BIOS to assign an arbitration priority level to the FW321. The default for this register (18h) indicates that the FW321 may need to access the PCI bus as often as every 0.25 µs; thus, an extremely high-priority level is requested. The contents of this field may also be loaded using a PCI bus command.
7:0	MIN_GNT	RU	**Minimum Grant.** The contents of this register may be used by host BIOS to assign a latency timer register value to the FW321. The default (0Ch) for this register indicates that the FW321 may need to sustain burst transfers for nearly 64 µs, thus requesting a large value be programmed in the FW321 latency timer register. The contents of this field may also be loaded using a PCI bus command.

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. On powerup, the hardware default value will be loaded, but an alternative value may be loaded using a PCI bus command.

Offset: 40h
Default: 0000 0000h
Type: Read/write
Reference: *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section A.3.7

Table 16. PCI OHCI Control Register Description

Bit	Field Name	Type	Description
31:1	Reserved	R	**Reserved.** Bits 31:1 return 0s when read.
0	GLOBAL_SWAP	RW	When this bit is set, all quadlets read from the FW321 as well as any data written to the PCI bus by the FW321 is byte swapped. This excludes PCI configuration registers (they are not swapped under any circumstances). However, OHCI registers are byte-swapped when this bit is set.

5 Internal Registers (continued)

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. 2.2, 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 17. Capability ID and Next Item Pointer Register Description

Bit	Field Name	Type	Description
15:8	NEXT_ITEM	R	**Next Item Pointer.** The FW321 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	**Capability Identification.** 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 FW321 related to PCI power management. On powerup, the default value will be loaded.

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 18. Power Management Capabilities Register Description

Bit	Field Name	Type	Description
15	PME_D3COLD	R	**PME Support from D3cold.** This bit is set to 0, indicating that the FW321 cannot generate a PME event in D3cold.
14	PME_D3HOT	R	**PME Support from D3hot.** This bit is set to 1, indicating that the FW321 can generate a PME event in the D3hot state.
13	PME_D2	R	**PME Support from D2.** This bit is set to 1, indicating that the FW321 can generate a PME in D2.
12	PME_D1	R	**PME Support from D1.** This bit is set to 1, indicating that the FW321 can generate a PME in D1.
11	PME_D0	R	**PME Support from D0.** This bit is set to 1, indicating that the FW321 can generate a PME in D0.
10	D2_SUPPORT	R	**D2 Support.** This bit returns a 1 when read, indicating that the FW321 supports the D2 power state.
9	D1_SUPPORT	R	**D1 Support.** This bit returns a 1 when read, indicating that the FW321 supports the D1 power state.
8:6	AUX_PWR	R	**Auxiliary Power Source**. This field reports the VAUX power requirements for the Open HCI function. This field is always 0.
5	DSI	R	**Device-Specific Initialization.** This bit returns 0 when read, indicating that the FW321 does not require special initialization beyond the standard PCI configuration header before a generic class driver is able to use it.
4	Reserved	R	**Reserved**. Bit returns 0 when read.
3	PME_CLK	R	**PME Clock.** This bit returns 0 when read, indicating that no host bus clock is required for the FW321 to generate PME.
2:0	PM_VERSION	R	**Power Management Version.** This field returns 010b when read, indicating that the FW321 is compatible with the registers described in the *PCI Power Management Interface Specification*, 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 19. Power Management Control and Status Register Description

Bit	Field Name	Type	Description
15	PME_STS	RC	This bit is set when the FW321 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 FW321. 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 FW321 within the PM_DATA/DATA_SCALE fields.
8	PME_ENB	RW	**PME Enable.** This bit enables the function to assert PME. If this bit is cleared, then assertion of PME is disabled.
7:5	Reserved	R	**Reserved.** Bits 7:5 return 0s when read.
4	DYN_DATA	R	**Dynamic Data**. This bit returns 0 when read, since the FW321 does not report dynamic data.
3:2	Reserved	R	**Reserved.** Bits 3:2 return 0s when read.
1:0	PWR_STATE	RW	**Power State**. This 2-bit field is used to set the FW321 device power state and is encoded as follows: 00 = current power state is D0. 01 = current power state is D1. 10 = current power state is D2. 11 = current power state is D3.

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

This register returns 00h when read since the FW321 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 Internal Registers (continued)

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 20). The FW321 supports programmability, via PCI bus commands, 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.

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 20. Power Management Data Register Description*

Value in Data_Select	Data Reported	Data_Scale Interpretation	Unit/Accuracy
0	D0 Power Consumed	0 = Unknown 1 = 0.1x 2 = 0.01x 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 FW321 and will return 00h when read)	Reserved	TBD

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

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 FW321 *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. 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 FW321 OHCI register definitions are based on version 1.1 of the *1394 Open Host Controller Specification*.

5 Internal Registers (continued)

Table 21. 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	HCControlSet	50h	5.7
		HCControlClear	54h	
	Reserved	—	58h	—
	Reserved	—	5Ch	—
	Reserved	—	60h	—
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	

* Since the FW321 V129 does not support an EEPROM, this register is not used.

5 Internal Registers (continued)

Table 21. OHCI Register Map (continued)

DMA Context	Register Name	Abbreviation	Offset	OHCI Specification Reference
—	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	—
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

* Since the FW321 V129 does not support an EEPROM, this register is not used.

5 Internal Registers (continued)

Table 21. OHCI Register Map (continued)

DMA Context	Register Name	Abbreviation	Offset	OHCI Specification Reference
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 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 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	—

* Since the FW321 V129 does not support an EEPROM, this register is not used.

5 Internal Registers (continued)

5.21 OHCI Version Register

This register indicates the OHCI version supported. 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 PCI bus commands. 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 Section 5.62) should also be programmed to properly enable OHCI 1.1 features within the FW321.

Offset: 00h
Default: 0X01 0000h
Type: Read/write
Reference: *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section 5.2

Table 22. OHCI Version Register Description

Bit	Field Name	Type	Description
31:25	Reserved	R	**Reserved.**
24	GUID_ROM	R	This field will return a 0 when read, indicating an EEPROM is not present.
23:16	Version	R	**Major Version of the OHCI.** The FW321 is compliant with both version 1.0 and version 1.1 of the *1394 Open Host Controller Interface Specification.* This field defaults to 01h, but can be reconfigured via PCI bus commands.
15:8	Reserved	R	**Reserved.**
7:0	Revision	R	**Minor Version of the OHCI.** The FW321 is compliant with both version 1.0 and version 1.1 of the *1394 Open Host Controller Interface Specification*. This field defaults to 00h, but can be reconfigured via PCI bus commands.

5.22 Asynchronous Transmit Retries Register

The asynchronous transmit retries register indicates the number of times the FW321 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 23. 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	**Reserved.** Bits 15:12 return 0s when read.
11:8	maxPhys RespRetries	RW	This field tells the physical response unit how many times to attempt to retry the transmit operation for the response.
7:4	maxAT RespRetries	RW	This field tells the asynchronous transmit DMA response unit how many times to attempt to retry the transmit operation for the response.
3:0	maxAT ReqRetries	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)

5.23 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 24. 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.

5.24 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 25. 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.

5.25 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 26. CSR Control Register Description

Bit	Field Name	Type	Description
31	csrDone	RU	This bit is set by the FW321 when a compare-swap operation is complete. It is reset whenever this register is written.
30:2	Reserved	R	**Reserved.** Bits 30:2 return 0s when read.
1:0	csrSel	RW	This field selects the CSR resource as follows: 00 = BUS_MANAGER_ID 01 = BANDWIDTH_AVAILABLE 10 = CHANNELS_AVAILABLE_HI 11 = CHANNELS_AVAILABLE_LO

5 Internal Registers (continued)

5.26 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 27. Configuration ROM Header Register Description

Bit	Field Name	Type	Description
31:24	info_length	RW	***IEEE* 1394 Bus Management Field.** Must be valid when bit 17 (linkEnable) of the host controller control register is set (see Section 5.35).
23:16	crc_length	RW	***IEEE* 1394 Bus Management Field.** Must be valid when bit 17 (linkEnable) of the host controller control register is set (see Section 5.35).
15:0	rom_crc_value	RW	***IEEE* 1394 Bus Management Field.** Must be valid at any time bit 17 (linkEnable) of the host controller control register is set (see Section 5.35).

5.27 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 28. 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 Internal Registers (continued)

5.28 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 29. Bus Options Register Description

Bit	Field Name	Type	Description
31:16	Reserved	R or RW	**Reserved**. Bits return 0s when read; 23:16 and 31:27 are RW and undefined.
15:12	max_rec	RW	***IEEE* 1394 Bus Management Field.** 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 $2^{(max\_rec + 1)}$. 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	**Reserved.** Bits 11:3 return 0s when read.
2:0	Lnk_spd	R	**Link Speed.** This field returns 010, indicating that the link speeds of 100 Mbits/s, 200 Mbits/s, and 400 Mbits/s are supported.

5.29 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. 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 FW321 05 PCI applications only. After one PCI configuration write 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 30. GUID High Register Description

Bit	Field Name	Type	Description
31:8	node_vendor_ID	RWU	***IEEE* 1394 Bus Management Fields.** 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 Internal Registers (continued)

5.30 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 identically to the GUID high register. 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 31. GUID Low Register Description

Bit	Field Name	Type	Description
31:0	chip_ID_lo	RW	***IEEE* 1394 Bus Management Fields.** Firmware or hardware must ensure that this register is valid whenever HCCControl.linkEnable bit is set.

5.31 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 32. 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	**Reserved.** Bits 9:0 return 0s when read.

5.32 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 33. 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 Internal Registers (continued)

5.33 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 34. 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.

5.34 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 35. Vendor ID Register Description

Bit	Field Name	Type	Description
31:24	vendorUnique	R	Returns 0 when read, since the FW321 does not specify any vendor unique registers.
23:0	vendorCompanyID	R	Returns 0 when read, since the FW321 does not specify any vendor unique registers.

5 Internal Registers (continued)

5.35 Host Controller Control Register

The host controller control set/clear register pair provides flags for controlling the OHCI portion of the FW321.

Offset: 50h set register
54h clear register
Default: X08X 0000h
Reference: *1394 Open Host Controller Interface Specification*, Rev. 1.1, Section 5.7

Table 36. 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 ack_type_error on block read requests to configuration ROM and does not update the configROMmap register or ConfigROMheader and BusOptions 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 ack_tardy. When this bit is set to one, ack_tardy may be returned as an acknowledgement to configuration ROM accesses from 1394 to OHCI including accesses to the bus_info_block. The host controller will return ack_tardy to all other asynchronous packets addressed to the OHCI node.
28:24	Reserved	R	**Reserved.** 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 (aPhyEnhanceEnable) in the FW321. When this bit is 0, the generic software may not modify the 1394a-2000 enhancements in the FW321 and cannot interpret the setting of bit 22 (aPhyEnhanceEnable). On powerup, the hardware default value will be loaded, but an alternative value may be loaded using a PCI bus command.
22	aPhyEnhanceEnable	RSC	When bit 23 (programPhyEnable) is 1 and bit 17 (linkEnable) is 0, the OHCI driver can set this bit to use all 1394a-2000 enhancements. When bit 23 (programPhyEnable) is set to 0, the software does not change PHY enhancements or this bit.
21:20	Reserved	R	**Reserved.** Bits 21:20 return 0s when read.
19	LPS	RSU	**Link Power Status.** This bit drives the LPS signal to the PHY core within the FW321 (see Section 5.7 of the *OHCI 1.1 Specification* 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 (linkEnable) is 0.

5 Internal Registers (continued)

Table 36. 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 FW321 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 FW321 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	**Reserved.** Bits 15:0 return 0s when read.

5.36 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 37. 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 Internal Registers (continued)

5.37 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 38. 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	**Reserved.** 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	**Reserved.** 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	**Reserved.** Bits 1:0 return 0s when read.

5.38 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 39. 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 Internal Registers (continued)

5.39 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 40. 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)

5.40 Interrupt Event (IntEvent) Register

The interrupt event set/clear register reflects the state of the various FW321 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 IntEvent-Clear 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 41. Interrupt Event Register Description

Bit	Field Name	Type	Description
31	Reserved	R	**Reserved.** Bit 31 returns 0 when read.
30	vendorSpecific	RSCU	This vendor-specific interrupt event is reported when serial EEPROM read is complete. **Note:** This bit should always be reset, since there is no EEPROM.
29	SoftInterrupt	RSC	**Soft Interrupt.** This bit may be used by software to generate a host controller interrupt for its own use.
28	Reserved	R	**Reserved.** 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 Section 5.35) is set to 1 and any of the following conditions occur: a. Data is present in a FIFO that is to be delivered to the host. b. The physical response unit is busy processing requests or sending responses. c. The host controller sent an ack_tardy acknowledgment.
26	phyRegRcvd	RSCU	The FW321 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 47) is set, then this indicates that over 125 µs 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 FW321 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 (cycle-Seconds field) and bits 24:12 (cycleCount field) of the isochronous cycle timer register (see Table 50).
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 41. Interrupt Event Register Description (continued)

Bit	Field Name	Type	Description
21	cycle64Seconds	RSCU	This bit indicates that the seventh bit of the cycleSeconds (see Table 50) counter has changed.
20	cycleSynch	RSCU	This bit indicates that a new isochronous cycle has started. This bit is set when the low-order bit of the cycleCount (see Table 50) toggles.
19	PHY	RSCU	This bit indicates the PHY core requests an interrupt through a status transfer.
18	regAccessFail	RSCU	This bit 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	This bit 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	**Reserved.** Bits 14:10 return 0s when read.
9	lockRespErr	RSCU	This bit indicates that the FW321 sent a lock response for a lock request to a serial bus register, but did not receive an ack_complete.
8	postedWriteErr	RSCU	This bit indicates that a host bus error occurred while the FW321 was trying to write a 1394 write request, which had already been given an ack_complete, into system memory.
7	isochRx	RU	**Isochronous Receive DMA Interrupt.** This bit indicates that one or more isochronous receive contexts have generated an interrupt. This is not a latched event; it is the **OR**ing of all bits in the isochronous receive interrupt event and isochronous receive interrupt mask registers. The isochronous receive interrupt event register (see Table 45) indicates which contexts have interrupted.
6	isochTx	RU	**Isochronous Transmit DMA Interrupt.** This bit indicates that one or more isochronous transmit contexts have generated an interrupt. This is not a latched event; it is the **OR**ing of all bits in the isochronous transmit interrupt event register (see Table 43) and isochronous transmit interrupt mask (see Table 44) register. The isochronous transmit interrupt event register indicates which contexts have interrupted.
5	RSPkt	RSCU	This bit 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	This bit 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	**Asynchronous Receive Response DMA Interrupt.** This bit is conditionally set upon completion of an ARRS DMA context command descriptor.
2	ARRQ	RSCU	**Asynchronous Receive Request DMA Interrupt.** This bit is conditionally set upon completion of an ARRQ DMA context command descriptor.
1	respTxComplete	RSCU	**Asynchronous Response Transmit DMA Interrupt.** This bit is conditionally set upon completion of an ATRS DMA command.
0	reqTxComplete	RSCU	**Asynchronous Request Transmit DMA Interrupt.** This bit is conditionally set upon completion of an ATRQ DMA command.

5 Internal Registers (continued)

5.41 Interrupt Mask (IntMask) Register

The interrupt mask set/clear register is used to enable/disable the various FW321 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 41). 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 42. Interrupt Mask Register Description

Bit	Field Name	Type	Description
31	masterIntEnable	RSCU	**Master Interrupt Enable.** 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 41) is set.
29	softInterrupt	RSC	**Soft Interrupt.** 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	**Reserved.** Bit 28 returns 0 when read.
27	ack_Tardy	RSCU	A **one** bit enables the corresponding IntEvent register bit to generate a processor interrupt. A **zero** 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	**Reserved.** Bits 14:10 return 0s when read.
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)

5.42 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 isochTx (bit 6) (see Table 41) 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 43. Isochronous Transmit Interrupt Event Register Description

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

5 Internal Registers (continued)

5.43 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 41.

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 44. Isochronous Transmit Interrupt Event Description

Bit	Field Name	Type	Description
31:8	Reserved	R	**Reserved.** 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 isoXmitIntEvent register.

5 Internal Registers (continued)

5.44 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 **isoRecvtIntEvent** and **isoRecvtIntMask**).

Offset: A0h set register
A4h clear register
Default: 0000 0000h
Reference: *1394 Open Host Controller Specification*, Rev. 1.1, Section 6.4

Table 45. Isochronous Receive Interrupt Event Description

Bit	Field Name	Type	Description
31:8	Reserved	R	**Reserved.** 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)

5.45 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

5.46 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 46. Fairness Control Register Description

Bit	Field Name	Type	Description
31:8	Reserved	R	**Reserved.** 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)

5.47 Link Control Register

The link control register provides flags to enable and configure the link core cycle timer and receiver portions of the FW321.

Offset: E0h set register
E4h clear register
Default: 00X0 0X00h
Reference: *1394 Open Host Controller Specification*, Rev. 1.1, Section 5.10

Table 47. Link Control Register Description

Bit	Field Name	Type	Description
31:23	Reserved	R	**Reserved.** Bits 31:23 return 0s when read.
22	cycleSource	RSC	This bit is set to 0, since the FW321 does not support an external cycle timer.
21	cycleMaster	RSCU	When this bit is set, and the FW321 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 41) 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	**Reserved.** 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	**Reserved.**
6	tag1SyncFilterLock	RS	When this bit is set, the tag1SyncFilter bit of the IR context match register (see Table 61) 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	**Reserved.** Bits 5:0 return 0s when read.

5 Internal Registers (continued)

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

Table 48. Node Identification Register Description

Bit	Field Name	Type	Description
31	iDValid	RU	This bit indicates whether or not the FW321 has a valid node number. It is cleared when a 1394 bus reset is detected and set when the FW321 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	**Reserved.** 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	**Reserved.** Bits 26:16 return 0s when read.
15:6	busNumber	RWU	This number is used to identify the specific 1394 bus to which the FW321 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 context control register for either of the AT DMA contexts (see Table 55).

5 Internal Registers (continued)

5.49 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 49. PHY Core Layer Control Register Description

Bit	Field Name	Type	Description
31	rdDone	RU	This bit is cleared to 0 by the FW321 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	**Reserved.** 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	**Reserved.** 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.

5.50 Isochronous Cycle Timer Register

The isochronous cycle timer register indicates the current cycle number and offset. When the FW321 is cycle master, this register is transmitted with the cycle start message. When the FW321 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 50. 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)

5.51 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 FW321. 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 Specificatio*n, Rev. 1.1, Section 5.14

Table 51. 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 FW321 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 41) 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 FW321. 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 41) is asserted will have no effect.

5.52 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 52. 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 FW321 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 41) is asserted will have no effect.

5 Internal Registers (continued)

5.53 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 53. 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 FW321 from nonlocal bus nodes are accepted. Set/Clear operations to this register while the IntEvent.busReset bit (see Table 41) 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 FW321 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 41) is asserted will have no effect. All physReqResourceN bits will be cleared to zero when a bus reset occurs.

5.54 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 54. Physical Request Filter Low Register Description

Bit	Field Name	Type	Description
31:0	physReqResourceN	RSC	If this bit is set, requests received by the FW321 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 41) is asserted will have no effect. All physReqResourceN bits will be cleared to zero when a bus reset occurs.

5 Internal Registers (continued)

5.55 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 55. Asynchronous Context Control Register Description

Bit	Field Name	Type	Description
31:16	Reserved	R	**Reserved.** 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 FW321 changes this bit (i.e., sets it to 0) only on a hardware or software reset.
14:13	Reserved	R	**Reserved.** Bits 14:13 return 0s when read.
12	wake	RSU	Software sets this bit to cause the FW321 to continue or resume descriptor processing. The FW321 clears this bit on every descriptor fetch.
11	dead	RU	The FW321 sets this bit when it encounters a fatal error and clears the bit when software resets bit 15 (run).
10	active	RU	The FW321 sets this bit to 1 when it is processing descriptors.
9:8	Reserved	R	**Reserved.** Bits 9:8 return 0s when read.
7:5	spd **Note:** 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 follows: 000 = 100 Mbits/s. 001 = 200 Mbits/s. 010 = 400 Mbits/s. 011 = 800 Mbits/s. 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)

5.56 Asynchronous Context Command Pointer Register

The asynchronous context command pointer register contains a pointer to the address of the first descriptor block that the FW321 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 56. 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	These bits indicate 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)

5.57 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 x n) set register
204h + (16 x n) clear register
Default: XXXX X0XXh
Reference: *1394 Open Host Controller Specification*, Rev. 1.1, Sections 9.2, 3.1.1

Table 57. 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 *1394 Open Host Controller Interface Specification*. 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 50). 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 FW321 changes this bit only on a hardware or software reset.
14:13	Reserved	R	**Reserved.** Bits 14:13 return 0s when read.
12	wake	RSU	Software sets this bit to cause the FW321 to continue or resume descriptor processing. The FW321 clears this bit on every descriptor fetch.
11	dead	RU	The FW321 sets this bit when it encounters a fatal error and clears the bit when software resets bit 15 (run).
10	active	RU	The FW321 sets this bit to 1 when it is processing descriptors.
9:5	Reserved	R	**Reserved.** 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)

5.58 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 FW321 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 x n)
Default: XXXX XXXXh
Reference: *1394 Open Host Controller Specification*, Rev. 1.1, Sections 9.2.1, 3.1.2

Table 58. 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	These bits indicate how many physically contiguous descriptors are pointed to by descriptorAddress.

5.59 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 x n) set register
404h + (32 x n) clear register
Default: X000 X0XXh
Reference: *1394 Open Host Controller Specification*, Rev. 1.1, Sections 10.3, 3.1.2

Table 59. 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 61) matches the two low-order bits of the CycleSeconds field and the 13-bit CycleCount field in the cycle timer 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.

5 Internal Registers (continued)

Table 59. Isochronous Receive Context Control Register Description (continued)

Bit	Field Name	Type	Description
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	**Reserved.** Bits 26:16 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 FW321 changes this bit only on a hardware or software reset.
14:13	Reserved	R	**Reserved.** Bits 14:13 return 0s when read.
12	wake	RSU	Software sets this bit to cause the FW321 to continue or resume descriptor processing. The FW321 clears this bit on every descriptor fetch.
11	dead	RU	The FW321 sets this bit when it encounters a fatal error and clears the bit when software resets bit 15 (run).
10	active	RU	The FW321 sets this bit to 1 when it is processing descriptors.
9:8	Reserved	R	**Reserved.** Bits 9:8 return 0s when read.
7:5	spd	RU	This field indicates the speed at which the packet was received. 000 = 100 Mbits/s. 001 = 200 Mbits/s. 010 = 400 Mbits/s. 011 = 800 Mbits/s. 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, Section 10.3.2 and Table 3-2).

5 Internal Registers (continued)

5.60 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 FW321 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 x n)
Default: XXXX XXXXh
Reference: *1394 Open Host Controller Specification*, Rev. 1.1, Sections 10.3, 3.1.2

Table 60. 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	These bits indicate 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)

5.61 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 x n)
Default: XXXX XXXXh
Reference: *1394 Open Host Controller Specification*, Rev. 1.1, Section 10.3. 3

Table 61. 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	**Reserved.** 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 two 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	**Reserved.** 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)

5.62 FW321 Vendor-Specific Registers

The FW321 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). On powerup, the hardware default value will be loaded, but an alternative value may be loaded using a PCI bus command. These control registers should not be changed when the link is enabled.

Table 62. FW321 Vendor-Specific Registers Description

Offset	Register Name	Description
12'h800	IsoDMACtrl	Controls PCI access for the isochronous DMA engines. Initial values are loaded from hardware defaults (see Table 63).
12'h808	AsyDMACtrl	Controls PCI access and AT FIFO threshold for the asynchronous DMA engines. Initial values are loaded from hardware defaults (see Table 63).
12'h840	LinkOptions	Controls low-level functionality of the link core. Initial values are loaded from hardware defaults (see Table 63).

5.63 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 63. 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 x (n + 1) quadlets; **n** defaults to 7 (128 quadlets). Max value of **n** 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 x (n + 1) quadlets; **n** 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 **n** 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 x (n + 1) quadlets; **n** defaults to 7 (128 quadlets). Max value of **n** 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 x (n + 1) quadlets; **n** 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 **n** is 0xf.

5 Internal Registers (continued)

5.64 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 64. Asynchronous DMA Control Registers Description

Bits	Field	Description
24	Retry Threshold Maximum 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 x n quadlets; **n** defaults to a value of 0x10 (256 quadlets). Max size of **n** 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 x (n + 1) quadlets; **n** defaults to 7 (128 quadlets). Max value of **n** 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 x (n + 1) quadlets; **n** 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 **n** 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 x (n + 1) quadlets; **n** defaults to 7 (128 quadlets). Max value of **n** 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 x (n + 1) quadlets; **n** 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 **n** is 0xf.

5 Internal Registers (continued)

5.65 Link Options

The values in this register provide low-level control of configurable features within the FW321 that are beyond those stated in 1394 and OHCI specifications.

Offset: 840h
Default: 0000 0020h

Table 65. 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 FW321. 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 FW321. Must be set to 0x0.
22	Reserved	Read-only status bit. Reserved for internal use by the FW321. Will read back as 0x0.
21	Reserved	Reserved for internal use by the FW321. Must be set to 0x0.
20	Reserved	Reserved for internal use by the FW321. Must be set to 0x0.
19:6	Reserved	Reserved for internal use by the FW321. 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 FW321 will use for its isochronous cycle period when the FW321 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. If 0, cycle = 125 µs. If 1, cycle = 62.5 µs. If 2, cycle = 31.25 µs. If 3, cycle = 15.625 µs. If 4, cycle = 7.8125 µs.

6 Internal Register Configuration

6.1 PHY Core Register Map

The PHY core register map is shown below in Figure 7.

Reference: *IEEE* Standard 1394a-2000, Annex J2

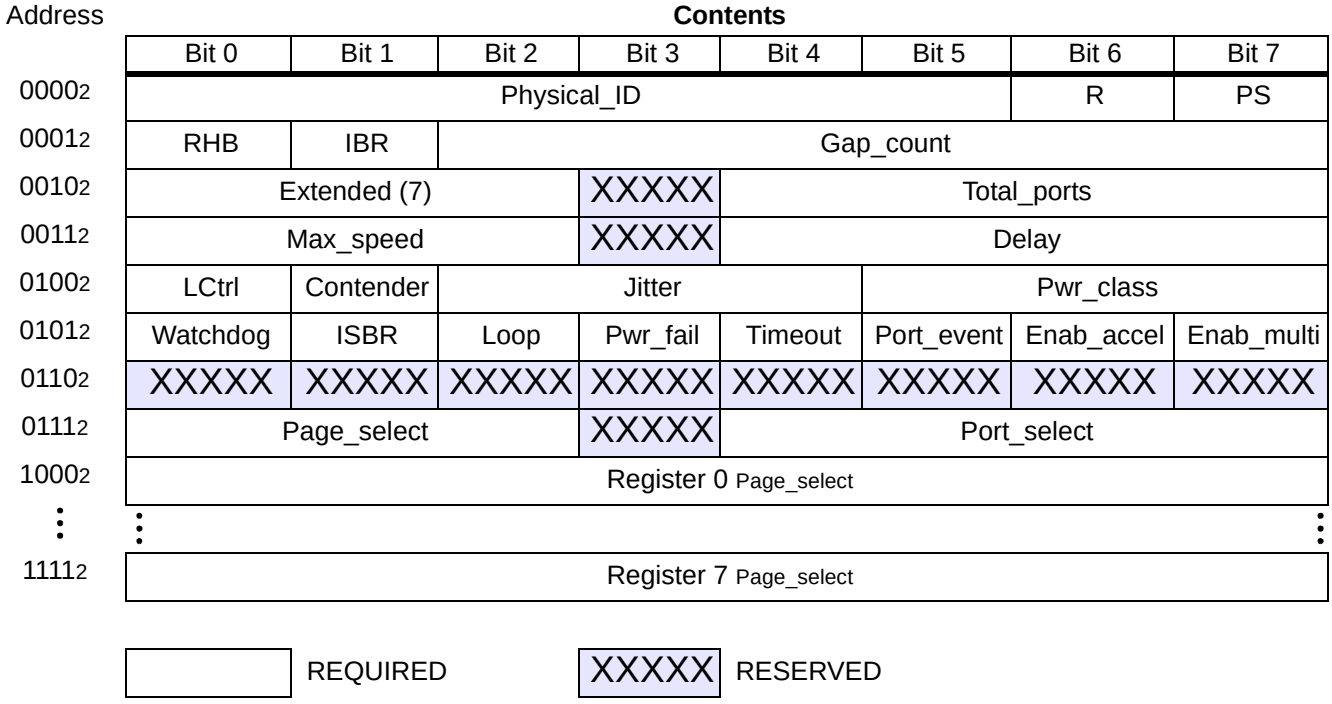


Figure 7. PHY Core Register Map

6 Internal Register Configuration (continued)

6.2 PHY Core Register Fields

Table 66. 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	—	**Cable Power Active.** 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	**Root Hold-Off Bit.** 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	**Initiate Bus Reset.** 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 µs. This bit is self-clearing.
Gap_count	6	RW	$3F_{16}$	Used to configure the arbitration timer setting to optimize gap times according to the topology of the bus. See Section 4.3.6 of *IEEE* Standard 1394a-2000 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	3	The number of ports implemented by this PHY core. This count reflects the number.
Max_speed	3	R	010_2	This bit indicates the speed(s) this PHY core supports: 000_2 = 98.304 Mbits/s. 001_2 = 98.304 and 196.608 Mbits/s. 010_2 = 98.304, 196.608, and 393.216 Mbits/s. 011_2 = 98.304, 196.608, 393.216, and 786.43 Mbits/s. 100_2 = 98.304, 196.608, 393.216, 786.432, and 1,572.864 Mbits/s. 101_2 = 98.304, 196.608, 393.216, 786.432, 1,572.864, and 3,145.728 Mbits/s. All other values are reserved for future definition.
Pwr_class	3	RW	See description	**Power Class.** Controls the value of the pwr field transmitted in the SelfID packet. See Section 4.3.4.1 of *IEEE* Standard 1394a-2000 for the encoding of this field. The PC0 pin determines the most significant bit of the power reset value. The two least significant bits, PC1 and PC2, are internally set to 0. The PC0 pin can be tied to V_{DD} for power class 4, or tied to V_{SS} for power class 0.
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	**Initiate Short (Arbitrated) Bus Reset.** 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	**Loop Detect.** A write of one to this bit clears it to zero.
Pwr_fail	1	RW	1	**Cable Power Failure Detect.** This bit is set to one when the PS bit changes from one to zero. A write of one to this bit clears it to zero.

6 Internal Register Configuration (continued)

Table 66. PHY Core Register Fields (continued)

Field	Size (In Bits)	Type	Power Reset Value	Description
Timeout	1	RW	0	**Arbitration State Machine Time-Out.** A write of one to this bit clears it to zero (see MAX_ARB_STATE_TIME).
Port_event	1	RW	0	**Port Event Detect.** 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	**Enable Arbitration Acceleration.** When set to one, the PHY core will use the enhancements specified in clause 4.4 of 1394*a-2000 Specification*. PHY core behavior is unspecified if the value of Enab_accel is changed while a bus request is pending.
Enab_multi	1	RW	0	**Enable Multispeed Packet Concatenation.** 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 1000_2 through 1111_2, 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 1000_2 through 1111_2, inclusive. Ports are numbered monotonically starting at zero, p0.
Delay	4	R	0000	Worst-case repeater delay; total worst-case repeater delay = [144 + (delay x 20)] ns.
LCtrl	1	RW	1	**Link Active.** 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 = [(Jitter + 1) x 20] ns.

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 0111_2. The format of the port status page is illustrated in Figure 8; reserved fields are shown as XXXXX. The meanings of the register fields in the port status page are defined as type RSC.

Address	Contents							
	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
1000_2	AStat		BStat		Child	Connected	Bias	Disabled
1001_2	Negotiated_speed			Int_enable	Fault	XXXXX	XXXXX	XXXXX
1010_2	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
1011_2	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
1100_2	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
1101_2	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
1110_2	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX
1111_2	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX	XXXXX

REQUIRED XXXXX RESERVED

Figure 8. PHY Core Register Page 0: Port Status Page

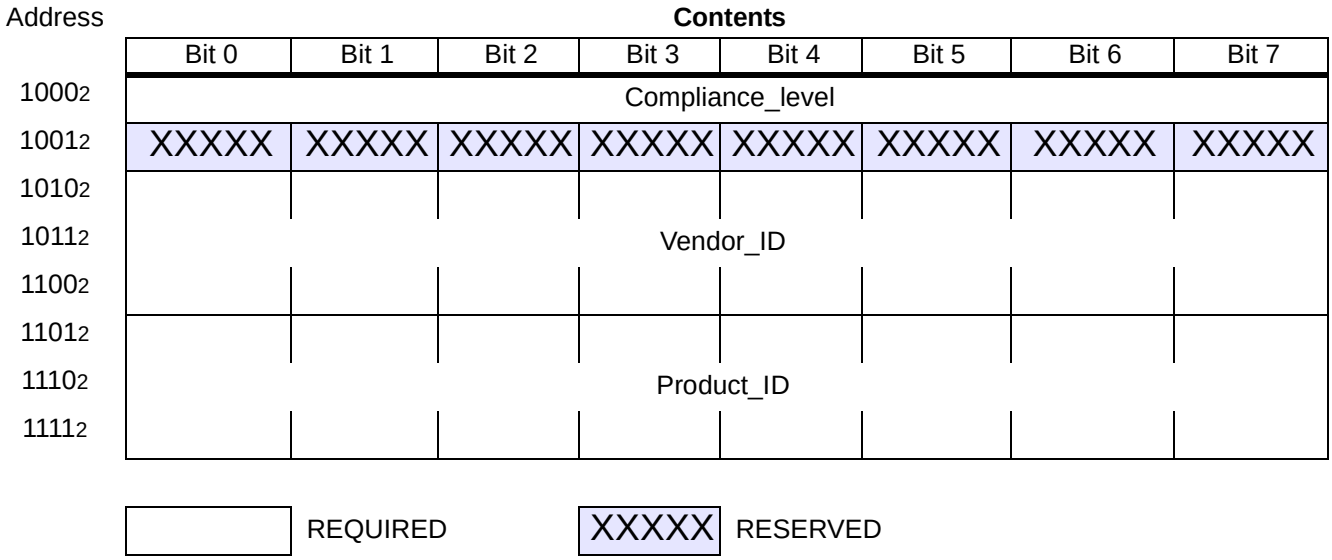
The meaning of the register fields in the port status page are defined by Table 67.

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: 00_2 = invalid. 01_2 = 1. 10_2 = 0. 11_2 = 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 hand-shake during the tree identify process (see Section 4.4.2.2 of *IEEE* Standard 1394a-2000).
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	This 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 66).
Int_enable	1	RW	0	**Enable Port Event Interrupts.** 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	This bit is 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 0111_2. The format of the vendor identification page is shown in Figure 9; reserved fields are shown as XXXXX.



Note: The meaning of the register fields within the vendor identification page are defined by Table 68.

Figure 9. PHY Core Register Page 1: Vendor Identification Page

Table 68. PHY Core Register Vendor Identification Page Fields

Field	Size	Type	Description
Compliance_level	8	r	Standard to which the PHY core implementation complies: 0 = not specified. 1 = *IEEE* 1394a-2000. Agere's FW321 compliance level is 1. 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 **00601Dh**. This number is obtained from the *IEEE* registration authority committee (RAC). The most significant byte of Vendor_ID appears at PHY core register location 1010_2 and the least significant at 1100_2.
Product_ID	24	r	The meaning of this number is determined by the company or organization that has been granted Vendor_ID. Agere's FW321 PHY core product ID is $03226x_{16}$.* The most significant byte of Product_ID appears at PHY core register location 1101_2 and the least significant at 1111_2.

* x is a minor revision number of the FW321 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 FW321.

7 Crystal Selection Considerations

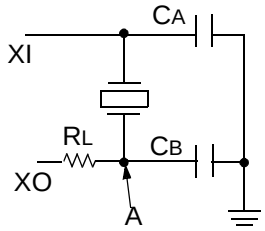
The FW321 is designed to use an external 24.576 MHz parallel resonant fundamental mode crystal connected between the XI and XO terminals to provide the reference for an internal oscillator circuit. The *IEEE* 1394a-2000 standard requires that FW321 have less than ±100 ppm total variation from the nominal 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 ±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 ±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 (C_L) is a function of not only the discrete load capacitors, but also capacitances from the FW321 board traces and capacitances of the other FW321 connected components. The values for load capacitors (C_A and C_B) should be calculated using this formula:

$$C_A = C_B = (C_L - C_{stray}) \times 2$$



Where:
C_L = load capacitance specified by the crystal manufacturer.
C_{stray} = capacitance of the board and the FW321, typically 2 pF—3 pF.
R_L = load resistance; nominal value is 400 Ω; the best value to be used can be determined by customer testing.

Figure 10. Crystal Circuitry

7.2 Adjustment to Crystal Loading

The resistor (R_L) in Figure 10 is recommended for fine-tuning the crystal circuit. The nominal value for this resistor is approximately 400 Ω. 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 R_L. 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 FW321 PLL. The crystal and two load capacitors (C_A + C_B) 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 NAND Tree Testing

The FW321 can be placed into a NAND tree mode of operation to enable board-level production testing. The NAND tree mode is designed to allow board-level contact testing of the digital pins of the FW321. To place the FW321 into NAND tree mode, pins 10 (TEST0), 7 (TEST1), and 124 (PTEST) should all be forced high. (In normal mode, these inputs are forced low.) The output for NAND tree is pin 6. No clocks are required for NAND tree testing. When NAND tree is enabled, the NAND tree logic follows the signal ordering in Table 69.

To run the test, force all of the inputs in the table below high. At this point, the NAND tree output should be verified to be high. In the order listed below, force each input low, while keeping previously tested inputs low. After each input is forced low, the NAND tree output should be verified, and the correct value should be the opposite of the previous value. Therefore, after forcing the first input low, the NAND tree output should be low, after forcing the second input low (and keeping the first input low), NAND tree output should be high, etc.

Table 69. NAND Tree Testing

Input Order	Pin #	Pin Name	Input Order	Pin #	Pin Name
1	B1	CARDBUSN	35	M7	PCI_STOPN
2	C2	CNA	36	N8	PCI_DEVSELN
3	A3	VAUX_PRESENT	37	N7	PCI_TRDYN
4	B3	SE	38	M6	PCI_IRDYN
5	A4	SM	39	N6	PCI_FRAMEN
6	D5	RESETN	40	M5	PCI_CBEN[2]
7	E13	MPCIACTN	41	M4	PCI_AD[16]
8	E12	LPS	42	N5	PCI_AD[17]
9	F10	LKON	43	N4	PCI_AD[18]
10	F13	PC0	44	M3	PCI_AD[19]
11	F12	PC1	45	M2	PCI_AD[20]
12	G10	PC2	46	N3	PCI_AD[21]
13	G12	CONTENDER	47	L3	PCI_AD[22]
14	H10	PCI_AD[0]	48	M1	PCI_AD[23]
15	H13	PCI_AD[1]	49	L2	PCI_IDSEL
16	J10	PCI_AD[2]	50	L1	PCI_CBEN[3]
17	J12	PCI_AD[3]	51	K2	PCI_AD[24]
18	J13	PCI_AD[4]	52	K4	PCI_AD[25]
19	J9	PCI_AD[5]	53	K1	PCI_AD[26]
20	K13	PCI_AD[6]	54	J2	PCI_AD[27]
21	K10	PCI_AD[7]	55	J1	PCI_AD[28]
22	K12	PCI_CBEN[0]	56	J4	PCI_AD[29]
23	L12	PCI_AD[8]	57	H2	PCI_AD[30]
24	M13	PCI_AD[9]	58	H1	PCI_AD[31]
25	L11	PCI_AD[10]	59	H4	PCI_CLK
26	M12	PCI_AD[11]	60	G1	PCI_REQN
27	M11	PCI_AD[12]	61	F2	PCI_GNTN
28	N12	PCI_AD[13]	62	F1	PCI_RSTN
29	M10	PCI_AD[14]	63	E1	PCI_INTAN
30	N11	PCI_AD[15]	64	F4	CLKRUNN
31	M9	PCI_CBEN[1]	65	D1	ROM_AD
32	N10	PCI_PAR	66	D2	ROM_CLK
33	N9	PCI_SERRN	Output	C1	NANDTREE
34	M8	PCI_PERRN			

8 NAND Tree Testing (continued)

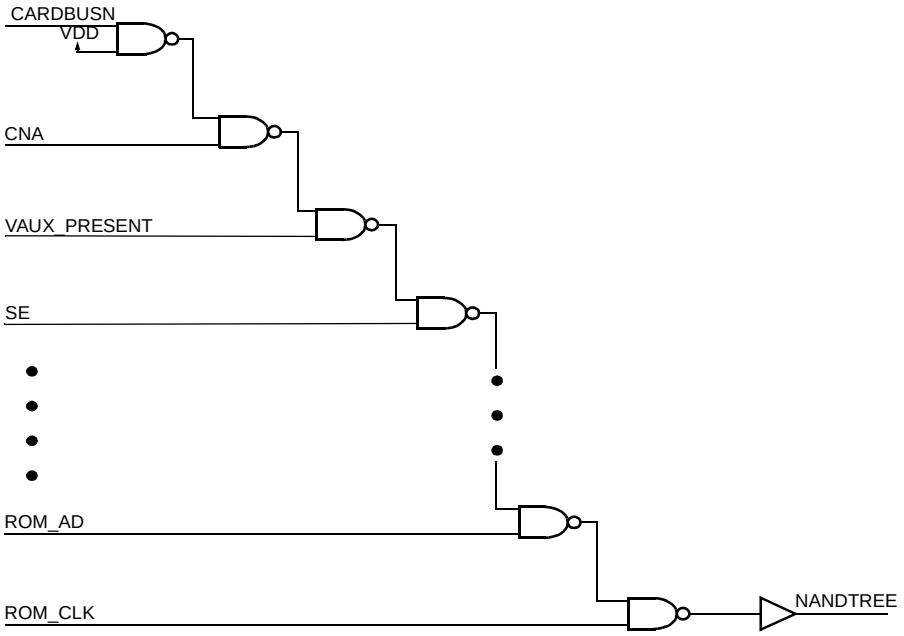


Figure 11. NAND Tree Logic Structure

9 Solder Reflow and Handling

The FW321 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 FW321 is 220 °C—225 °C. This is the actual tolerance that Agere uses to test the devices during preconditioning.

10 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 70. Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Unit
Supply Voltage Range	V_{DD}	3.0	3.6	V
Input Voltage Range	V_I	–0.5	V_{DD} + 0.5	V
Output Voltage Range at Any Output	V_O	–0.5	V_{DD} + 0.5	V
Operating Free Air Temperature	T_A	0	70	°C
Storage Temperature Range	T_{stg}	–65	150	°C

11 Electrical Characteristics

Table 71. Analog Characteristics

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

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

11 Electrical Characteristics (continued)

Table 72. Driver Characteristics

Parameter	Test Conditions	Symbol	Min	Typ	Max	Unit
Differential Output Voltage	56 Ω load	V_{OD}	172	—	265	mV
Off-state Common-mode Voltage	Drivers disabled	V_{OFF}	—	—	20	mV
Driver Differential Current, TPA+, TPA–, TPB+, TPB–	Driver enabled, speed signaling off*	I_{DIFF}	–1.05	—	1.05	mA
Common-mode Speed Signaling Current, TPB+, TPB–	200 Mbits/s speed signaling enabled	I_{SP}	–2.53	—	–4.84	mA
	400 Mbits/s speed signaling enabled	I_{SP}	–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 73. Device Characteristics

Parameter	Test Conditions	Symbol	Min	Typ	Max	Unit
Supply Current:						
D0, 1 Port Active	V_{DD} = 3.3 V	I_{DD}	—	97	—	mA
D0, No Ports Active	V_{DD} = 3.3 V	I_{DD}	—	78	—	mA
System in Standby (suspend mode = S1)	V_{DD} = 3.3 V PCI or V_{DD} = 3.3 V_{AUX}	I_{DD}	—	30	—	mA
System in Hibernate (suspend mode = S1 or S3) or System in Standby (suspend mode = S3)	V_{DD} = 3.3 V_{AUX}	I_{DD}	—	1.3	—	mA
High-level Output Voltage	I_{OH} max, V_{DD} = min	V_{OH}	V_{DD} – 0.4	—	—	V
Low-level Output Voltage	I_{OL} min, V_{DD} = max	V_{OL}	—	—	0.4	V
High-level Input Voltage	CMOS inputs	V_{IH}	0.7 V_{DD}	—	—	V
Low-level Input Voltage	CMOS inputs	V_{IL}	—	—	0.2 V_{DD}	V
Pull-up Current, RESETN Input	V_I = 0 V	I_I	11	—	32	µA

* This I_{DD} value may differ depending on the system board into which the FW321 06 T100 PCI add-in card is inserted.

12 Timing Characteristics

Table 74. Switching Characteristics

Symbol	Parameter	Measured	Test Conditions	Min	Typ	Max	Unit
—	Jitter, Transmit	TPA, TPB	—	—	—	0.15	ns
—	Transmit Skew	Between TPA and TPB	—	—	—	±0.1	ns
t_r	Rise Time, Transmit (TPA/TPB)	10% to 90%	R_l = 56 Ω, C_l = 10 pF	—	—	1.2	ns
t_f	Fall Time, Transmit (TPA/TPB)	90% to 10%	R_l = 56 Ω, C_l = 10 pF	—	—	1.2	ns

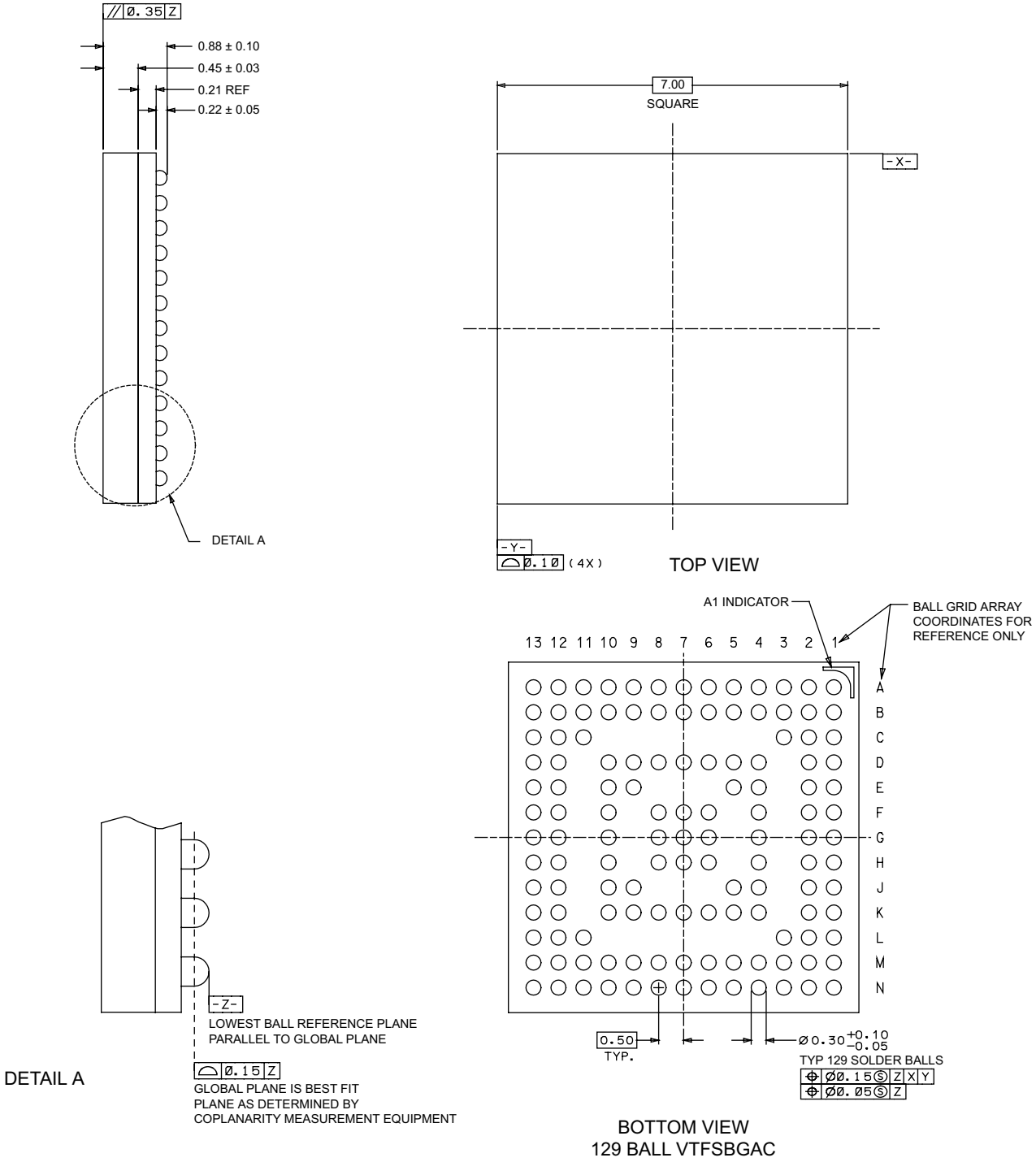
Table 75. Clock Characteristics

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

13 Outline Diagram

13.1 129-Ball VTFSBGAC

Dimensions are in millimeters.



14 Ordering Information

Device Code	Package	Comcode
L-FW321-06-NV129-DB*†	129-Ball VTFSBGAC	711007047
L-FW321-06-NV129-DT*†	129-Ball VTFSBGAC	711007048

* 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.

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**
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 296 400**

November 2005
DS06-025CMPR