DATA SHEET

NOTICE:
SEE ERRATA (REVISED 8/24/2000) AT WWW.SEMICONDUCTORS.COM/1394/SUPPRT FOR INFORMATION REGARDING CHANGED SPECIFICATIONS

NOT RECOMMENDED FOR NEW DESIGNS

PDI1394L11
1394 AV link layer controller

Product specification

1997 Oct 21

Philips Semiconductors

PHILIPS

1.0 FEATURES

- IEEE 1394–1995 Standard Link Layer Controller
- Hardware Support for the IEC61883 International Standard of Digital Interface for Consumer Electronics
- Interface to any IEEE 1394–1995 Physical Layer Interface
- 5V Tolerant I/Os
- Single 3.3V supply voltage

2.0 DESCRIPTION

The PDI1394L11, Philips Semiconductors 1394 Audio/Video (AV) Link Layer Controller, is an IEEE 1394–1995 compliant link layer controller featuring an embedded AV layer interface. The AV layer is designed to pack and un-pack application data packets for transmission over an IEEE 1394 bus using isochronous data transfers.

The application data is packetized according to the IEC 61883 International Standard of Digital Interface for Consumer Electronic Audio/Video Equipment. The AV layer interface is a byte-wide port capable of accommodating various MPEG–2 and DVC codecs. An 80C51 compatible byte-wide host interface is provided for internal register configuration as well as performing asynchronous data transfers.

The PDI1394L11 is powered by a single 3.3V power supply and the inputs and outputs are 5V tolerant. It is available in the PQFP80 package.

3.0 QUICK REFERENCE DATA

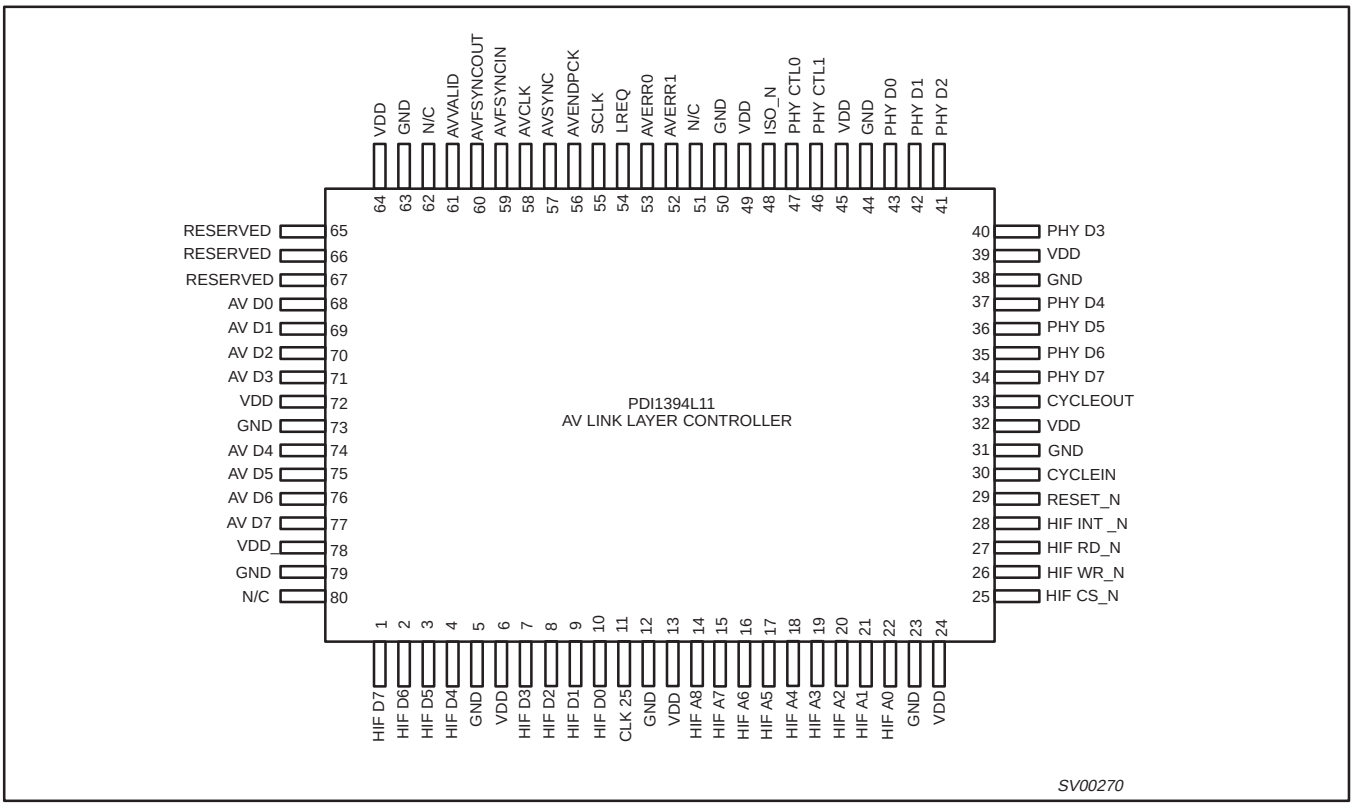
GND = 0V; T_{amb} = 25°C

SYMBOL	PARAMETER	CONDITIONS	MIN	TYP	MAX	UNIT
V_{DD}	Functional supply voltage range		3.0	3.3	3.6	V
I_{DD}	Supply current @ V_{DD}=3.3V			20		mA
SCLK	Device clock		49.147	49.152	49.157	MHz

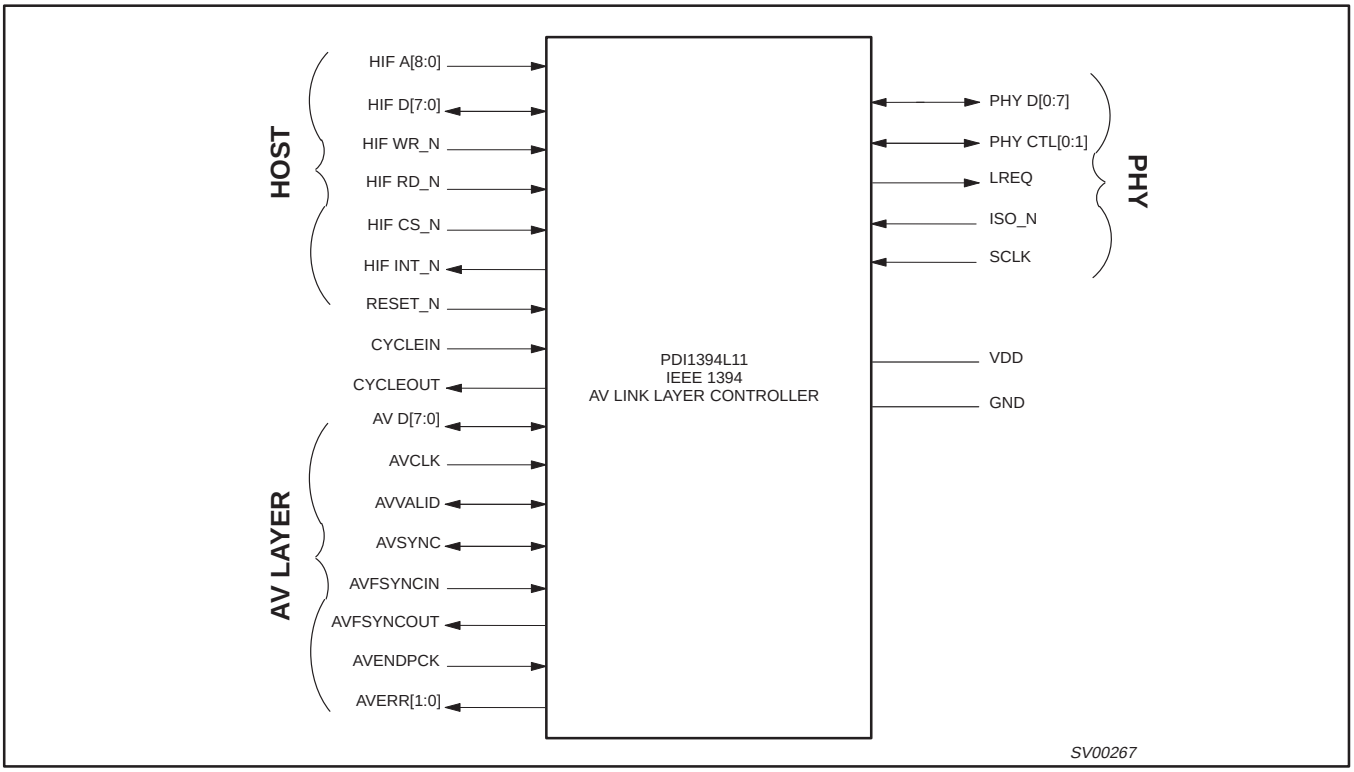
4.0 ORDERING INFORMATION

PACKAGES	TEMPERATURE RANGE	OUTSIDE NORTH AMERICA	NORTH AMERICA	PKG. DWG. #
80-pin plastic PQFP80	0°C to +70°C	PDI1394L11 BA	PDI1394L11 BA	SOT318-2

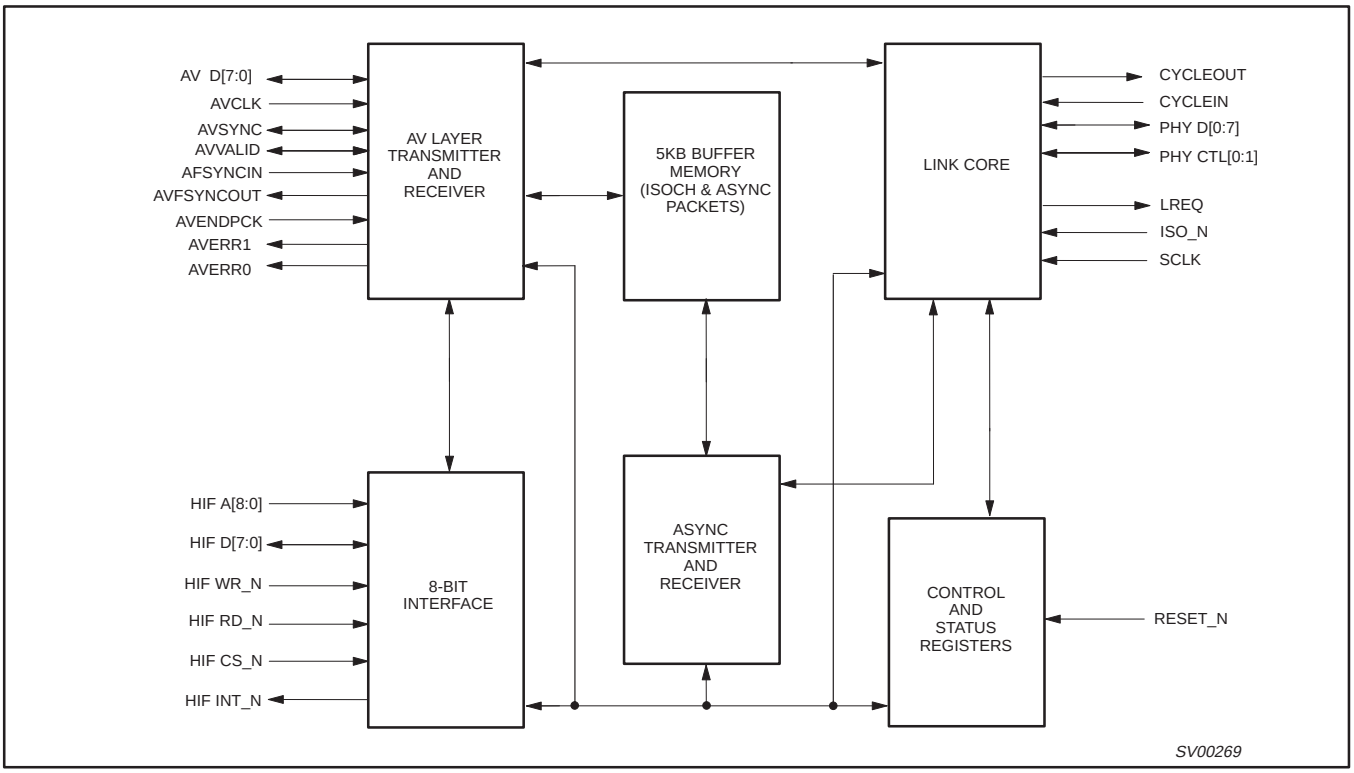
5.0 PIN CONFIGURATION



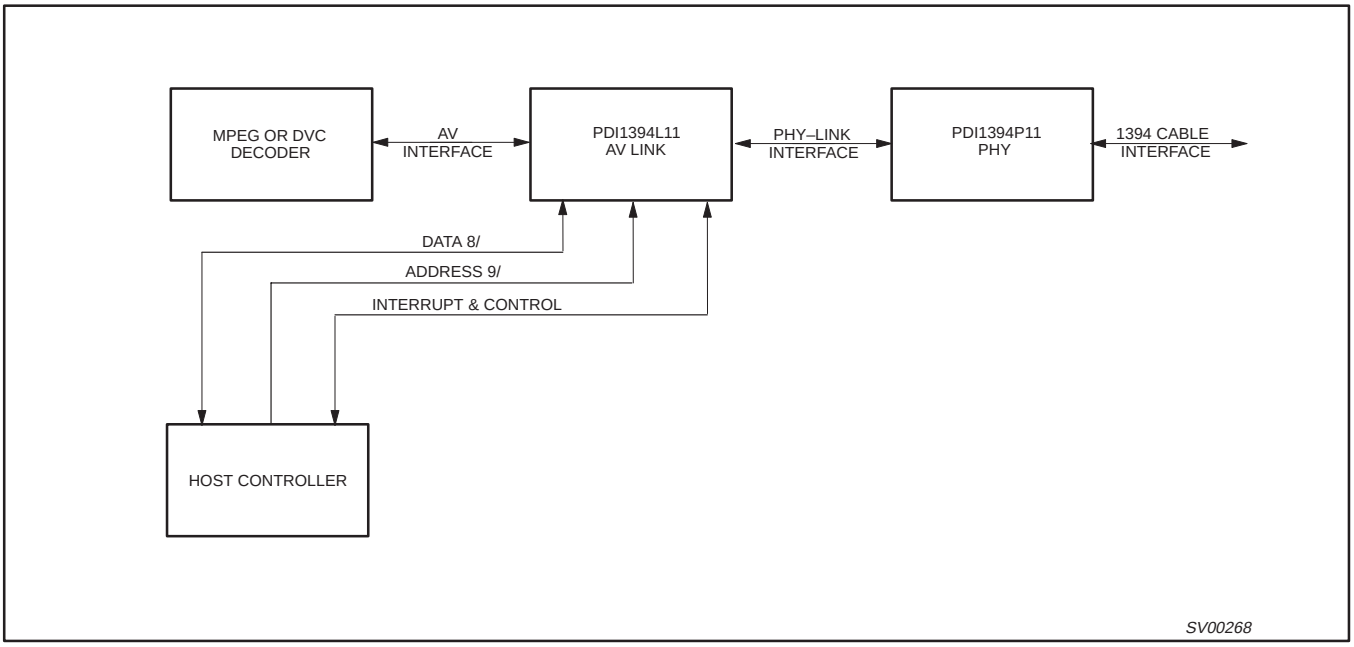
6.0 FUNCTIONAL DIAGRAM



7.0 INTERNAL BLOCK DIAGRAM



8.0 APPLICATION DIAGRAM



9.0 PIN DESCRIPTION

9.1 Host Interface

PIN No.	PIN SYMBOL	I/O	NAME AND FUNCTION
14, 15, 16, 17, 18, 19, 20, 21, 22	HIF A[8:0]	I	Host Interface Address 0 through 8. Provides the host with a byte wide interface to internal registers. See description of Host Interface for addressing rules.
1, 2, 3, 4, 7, 8, 9, 10	HIF D[7:0]	I/O	Host Interface Data 7 (MSB) through 0. Byte wide data path to internal registers.
26	HIF WR_N	I	Write enable. When asserted (LOW) in conjunction with HIF CS_N, a write to the PDI1394L11 internal registers is requested. (NOTE: HIF WR_N and HIF RD_N : if these are both LOW in conjunction with HIF CS_N, then a write cycle takes place. This can be used to connect CPUs that use R/W_N line rather than separate RD_N and WR_N lines. In that case, connect the R/W_N line to the HIF WR_N and tie HIF RD_N LOW.)
27	HIF RD_N	I	Read enable. When asserted (LOW) in conjunction with HIF CS_N, a read of the PDI1394L11 internal registers is requested.
25	HIF CS_N	I	Chip Select (active LOW). Host bus control signal to enable access to the FIFO and control and status registers.
28	HIF INT_N	O	Interrupt (active LOW). Indicates a interrupt internal to the PDI1394L11. Read the General Interrupt Register for more information. This pin is open drained and requires a 1KΩ pullup resistor.
29	RESET_N	I	Reset (active LOW). The asynchronous master reset to the PDI1394L11.
6, 13, 24, 32, 39, 45, 49, 64, 72, 78	V_{DD}		3.3V ± 0.3V power supply
5, 12, 23, 31, 38, 44, 50, 63, 73, 79	GND		Ground reference

9.2 AV Interface

PIN No.	PIN SYMBOL	I/O	NAME AND FUNCTION
77, 76, 75, 74, 71, 70, 69, 68	AV D[7:0]	I/O	Audio/Video Data 7 (MSB) through 0. Byte-wide interface to the AV layer.
58	AVCLK	I	External application clock. Rising edge active.
57	AVSYNC	I/O	Start of packet indicator; should only be used when AVVALID is active.
59	AVFSYNCIN	I	Frame sync input. Used for Digital Video (DV). The signal is time stamped and transmitted in the SYT field of ITXHQ2.
60	AVFSYNCOUT	O	Frame sync output. Signal is derived from SYT field of IRXHQ2.
56	AVENDPCK	I	End of application packet indication from data source. Required only if input packet is not multiple of 4 bytes. It can be tied LOW for data packets that are 4*N in size.
61	AVVALID	I/O	Indicates data on AV D [7:0] is valid
53	AVERR0	O	CRC error, indicates bus packet containing AV D [7:0] had a CRC error, the current AV packet is unreliable.
52	AVERR1	O	Sequence Error. Indicates at least one source packet was lost before the current AV D [7:0]

9.3 Phy Interface

PIN No.	PIN SYMBOL	I/O	NAME AND FUNCTION
34, 35, 36, 37, 40, 41, 42, 43	PHY D[0:7]	I/O	Data 0 (MSB) through 7 (NOTE: To preserve compatibility to the specified Link-Phy interface of the IEEE 1394–1995 standard, Annex J, bit 0 is the most significant bit). Data is expected on AV D[0:1] for 100Mb/s, AV D[0:3] for 200Mb/s, and AV D[0:7] for 400Mb/s. See IEEE 1394–1995 standard, Annex J for more information.
46, 47	PHY CTL[0:1]	I/O	Control Lines between Link and Phy. See 1394 Specification for more information.
48	ISO_N	I	Isolation barrier. This terminal is asserted (LOW) when an isolation barrier is present. See IEEE 1394–1995 standard, Annex J for more information (used to request arbitration or read/write PHY registers).
54	LREQ	O	Link Request. Bus request to access the PHY. See IEEE 1394–1995 standard, Annex J for more information.
55	SCLK	I	System clock. 49.152MHz input from the PHY (the PHY-LINK interface operates at this frequency).

9.4 Other Pins

PIN No.	PIN SYMBOL	I/O	NAME AND FUNCTION
65, 66, 67	RESERVED	NA	These pins are reserved for factory testing. For normal operation they should be connected to ground.
51, 62, 80	N/C	NA	These are test mode pins and should not be connected or terminated.
30	CYCLEIN	I	Provides the capability to supply an external cycle timer signal for the beginning of 1394 bus cycles.
33	CYCLEOUT	O	Reproduces the 8kHz cycle clock of the cycle master.
11	CLK 25	O	Auxiliary clock, value is SCLK/2 (usually 24.576 MHz)

10.0 RECOMMENDED OPERATING CONDITIONS

SYMBOL	PARAMETER	CONDITIONS	LIMITS		UNIT
			MIN.	MAX.	
V_{CC}	DC supply voltage		3.0	3.6	V
V_I	Input voltage		0	5	V
V_{IH}	High-level input voltage		2.0		V
V_{IL}	Low-level input voltage			0.8	V
I_{OH}	High-level output current			8	mA
I_{OL}	Low-level output current			–8	mA
dT/dV	Input transition rise or fall time		0	20	ns/V
T_{amb}	Operating ambient temperature range		0	+70	°C
SCLK	System clock		49.147	49.157	MHz
AVCLK	AV interface clock		0	24	MHz
t_r	Input rise time			10	ns
t_f	input fall time			10	ns

11.0 ABSOLUTE MAXIMUM RATINGS[1, 2]

In accordance with the Absolute Maximum Rating System (IEC 134). Voltages are referenced to GND (ground = 0V)

SYMBOL	PARAMETER	CONDITIONS	LIMITS		UNIT
			MIN	MAX	
V_{DD}	DC supply voltage		–0.5	+4.6	V
I_{IK}	DC input diode current		–	–50	mA
V_I	DC input voltage		–0.5	+5.5	V
I_{OK}	DC output diode current		–	±50	mA
V_O	DC output voltage		–0.5	V_{DD} +0.5	V
I_O	DC output source or sink current		–	±50	mA
I_{GND}, I_{CC}	DC V_{CC} or GND current		–	±150	mA
T_{stg}	Storage temperature range		–60	150	°C
T_{amb}	Operating ambient temperature		0	70	°C
P_{tot}	Power dissipation per package			0.6	W

NOTES:

1. Stresses beyond those listed may cause permanent damage to the device. These are stress ratings only and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
2. The performance capability of a high-performance integrated circuit in conjunction with its thermal environment can create junction temperatures which are detrimental to reliability. The maximum junction temperature of this integrated circuit should not exceed 150°C.

11.1 Buffer Memory Sizes

BUFFER MEMORY	SIZE (Quadlets)
Asynchronous Receive Transaction Response FIFO	64
Asynchronous Receive Transaction Request FIFO	64
Asynchronous Transmit Transaction Response FIFO	64
Asynchronous Transmit Transaction Request FIFO	64
AV Transmit/Receive Buffer	1024

12.0 FUNCTIONAL DESCRIPTION

12.1 Overview

The PDI1394L11 is an IEEE 1394–1995 compliant link layer controller. It provides a direct interface between a 1394 bus and various MPEG–2 and DVC codecs. Via this interface, the AV Link maps and unmaps these AV datastreams from these codecs onto 1394 isochronous bus packets. The AV Link also provides an 8051 compatible microcontroller interface for an attached host controller. Through the host interface port, the host controller can configure the AV layer for transmission or reception of AV datastreams. The host interface port also allows the host controller to transmit and receive 1394 asynchronous data packets.

12.2 AV interface and AV layer

The AV interface and AV layer allow AV packets to be transmitted from one node to another. The AV transmitter and receiver within the AV layer perform all the functions required to pack and unpack AV packet data for transfer over a 1394 network. Once the AV layer is properly configured for operation, no further host controller service should be required. The operation of the AV layer is half-duplex, i.e., the AV layer can either receive or transmit AV packets at a particular time.

12.2.1 The AV Interface

The AV Link provides an 8 bit data path to the AV layer. The 8 bit data path is designed with associated clock and control signals to be compatible with various MPEG–2 and DVC codecs.

The AV interface port buffer, if so programmed, can time stamp incoming AV packets. The AV packet data is stored in the embedded memory buffer, along with its time stamp information. After the AV packet has been written into the AV layer, the AV layer creates an isochronous bus packet with the appropriate CIP header. The bus packet along with the CIP header is transferred over the appropriate isochronous channel/packet. The size and configuration of isochronous data packet payload transmitted is determined by the AV layer's configuration registers accessible through the host interface.

The AV interface port waits for the assertion for AVVALID and AVSYNC. Note: Do not assert AVSYNC without AVVALID. AVSYNC is aligned with the rising edge of AVCLK and the first byte of data on AVDATA[7:0]. The duration of AVSYNC is one AVCLK cycle. AVSYNC signals the AV layer that the transfer of an AV packet has begun. At the time the AVSYNC is asserted, the AV layer creates a new time stamp in the buffer memory. (This only happens if so configured. The DVC format does not require these time stamps). The time stamp is then transmitted as part of the standard packet header. This allows the AV receiver to provide the AV packet for output at the appropriate time.

When the DV video is enabled (via the format code of the CIP header), the frame synchronization signal AVFSYNCIN is time stamped and placed in the SYT field. The timestamp value is 3 cycle times (duration of 125μs) in the future and is transmitted in the SYT field of the current CIP header. On the receiver side, when the SYT stamp matches the cycle timer register, a pulse is generated on the AVFSYNCOUT output. The timing for AVFSYNCIN and AVFSYNCOUT are independent of AV clock.

12.2.2 IEC 61883 International Standard

The PDI1394L11 is specifically designed to support the proposed IEC61883 International Standard of Digital Interface for Consumer Electronic Audio/Video Equipment. The IEC specification defines a scheme for mapping various types of AV datastreams onto 1394 isochronous data packets. The standard also defines a software protocol for managing isochronous connections in a 1394 bus called Connection Management Protocol (CMP). It also provides a framework for transfer of functional commands, called Function Control Protocol (FCP).

12.2.3 CIP Headers

A feature of the IEC61883 International Standard is the definition of Common Isochronous Packet (CIP) headers. These CIP headers contain information about the source and type of datastream mapped onto the isochronous packets.

The AV Layer supports the use of CIP headers. CIP headers are added to transmitted isochronous data packets at the AV data source. When receiving isochronous data packets, the AV layer automatically analyzes their CIP headers. The analysis of the CIP headers determines the method the AV layer uses to unpack the AV data from the isochronous data packets.

The information contained in the CIP headers is accessible via registers in the host interface.

(See IEC61883 International Standard of Digital Interface for Consumer Electronic Audio/Video Equipment for more details on CIP headers).

12.3 The host interface

The host interface allows an 8 bit CPU to access all registers and the asynchronous packet queues. It is specifically designed for an 8051 microcontroller but can also be used with other CPUs. There are 64 register addresses (for quadlet wide registers). To access

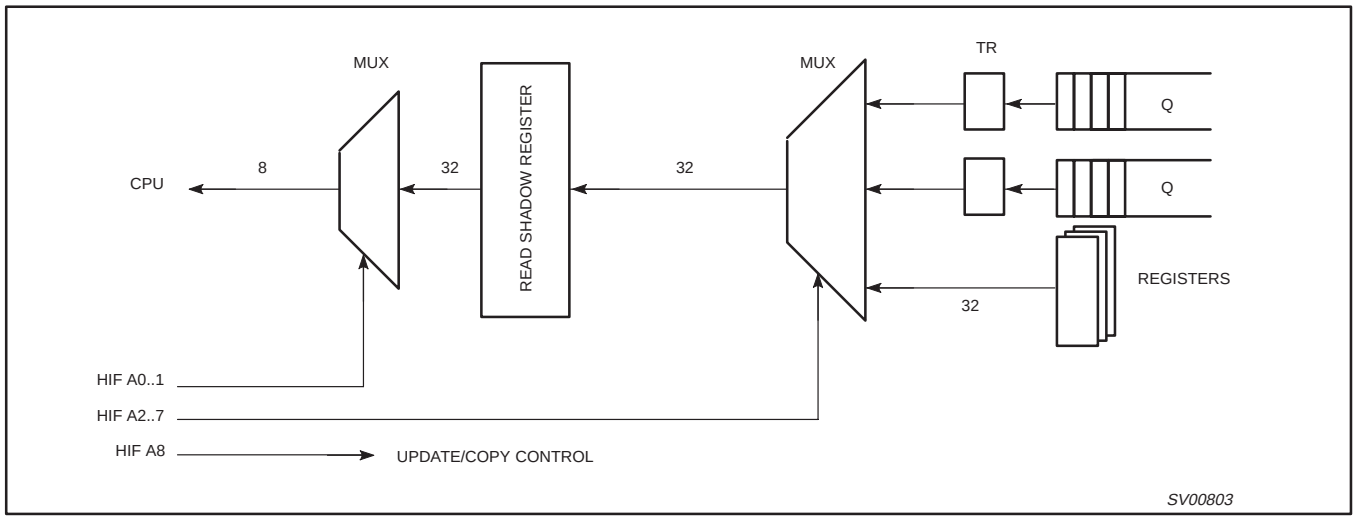
bytes rather than quadlets the address spaces is 256 bytes, requiring 8 address lines.

The use of an 8 bit interface introduces an inherent problem that must be solved: register fields can be more than 8 bits wide and be used (control) or changed (status) at every internal clock tick. If such a field is accessed through an 8 bit interface it requires more than one read or write cycle, and the value should not change in between to maintain consistency. To overcome this problem accesses to the chip's internal register space are always 32 bits, and the host interface must act as a converter between the internal 32 bit accesses and external 8 bit accesses. This is where the shadow registers come in.

12.3.1 Read accesses

To read an internal register the host interface can make a snapshot (copy) of that specific register which is then made available to the CPU 8 bits at a time. The register that holds the snapshot copy of the real register value inside the host interface is called the **read shadow register**. During a read cycle address lines HIF A0 and HIF A1 are used to select which of the 4 bytes currently stored in the **read shadow register** is output onto the CPU data bus. This selection is done by combinatorial logic only, enabling external hardware to toggle these lines through values 0 to 3 while keeping the chip in a read access mode to get all 4 bytes out very fast (in a single extended read cycle), for example into an external quadlet register.

This solution requires a control line to direct the host interface to make a snapshot of an internal register when needed, as well as the internal address of the target register. The register address is connected to input address lines HIF A2..HIF A7, and the update control line to input address line HIF A8. To let the host interface take a new snapshot the target address must be presented on HIF A2..HIF A7 and HIF A8 must be raised while executing a read access. The new value will be stored in the **read shadow register** and the selected byte (HIF A0, HIF A1) appears on the output.



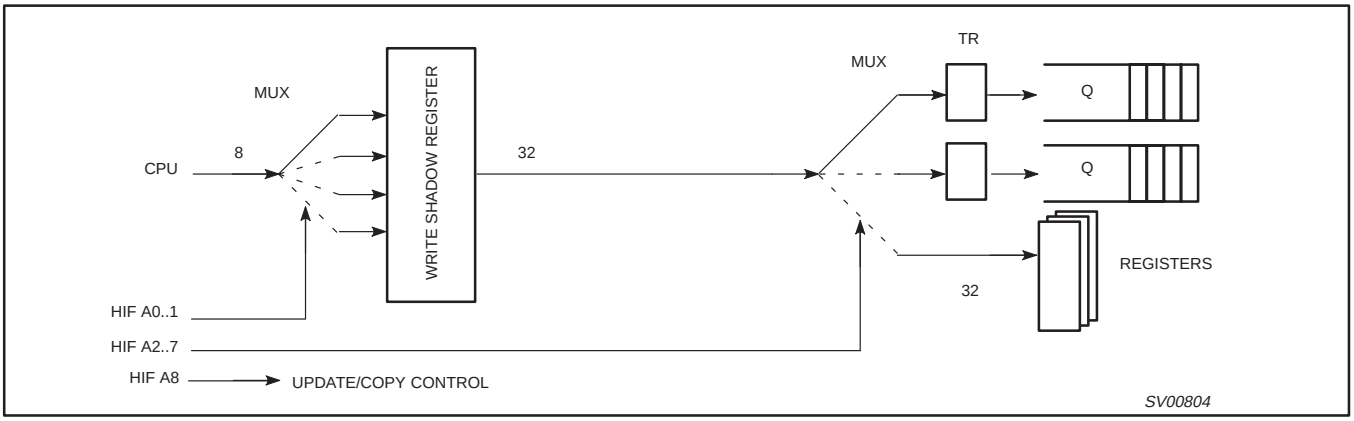
NOTES:

1. It is not required to read all 4 bytes of a register before reading another register. For example, if only byte 2 of register 0x54 is required a read of byte address 0x100 + (0×54) + 2 = 0x156 is sufficient.
2. The update control line does not necessarily have to be connected to the CPU address line HIF A8. This input could also be controlled by other means, for example a combinatorial circuit that activates the update control line whenever a read access is done for byte 0. This makes the internal updating automatic for quadlet reading.
3. Reading the bytes of the read shadow register can be done in any order and as often as needed.

12.3.2 Write accesses

To write to an internal register the host interface must collect the 4 byte values into a 32 bit value and then write the result to the target register in a single clock tick. This requires a register to hold the 32 bit value being compiled until it is ready to be written to the actual target register. This temporary register inside the host interface is called the **write shadow register**. During all write cycles address lines HIF A0 and HIF A1 are used to select which of the 4 bytes of the **write shadow register** is to be written with the value on the CPU data bus. Only one byte can be written in a single write access cycle.

This solution requires a control line to direct the host interface to copy the **write shadow register** to the actual destination register when ready, as well as the internal address of that register. The destination register address is connected to input address lines HIF A2..HIF A7, and the update control line to input address line HIF A8. To let the host interface make the internal transfer the target address must be presented on HIF A2..HIF A7 and HIF A8 must be raised while executing a write access. The current value on the CPU data bus will be stored in the **write shadow register** at the selected byte (HIF A0, HIF A1) and the result will be copied into the specified destination register.



NOTES:

1. It is not required to write all 4 bytes of a register: those bytes that are either reserved (undefined) or don't care do not have to be written in which case they will be assigned the value that was left in the corresponding byte of the **write shadow register** from a previous write access. For example, to acknowledge an interrupt for the isochronous receiver (external address 0x04C), a single byte write to location 0x100+(0x4C)+3 = 0x14F is sufficient. The value 256 represents setting HIF A8=1. The host interface cannot directly access the FIFOs, but instead reads from/writes into a transfer register (shown as TR in the Figures above). Data is moved between FIFO and TR by internal logic as soon as possible without CPU intervention.
2. The update control line does not necessarily have to be connected to the CPU address line HIF A8. This input could also be controlled by other means, for example a combinatorial circuit that activates the update control line whenever a write access is done for byte 3. This makes the internal updating automatic for quadlet writing.
3. Writing the bytes of the read shadow register can be done in any order and as often as needed (new writes simply overwrite the old value).

12.3.3 Byte order

The bytes in each quadlet are numbered 0..3 from left (most significant) to right (least significant) as shown in Figure 1.
To access a register at internal address N the CPU should use addresses E:

E = 4 N ; to access the upper 8 bits of the register.

E = 4 N + 1 ; to access the upper middle 8 bits of the register.

E = 4 N + 2 ; to access the lower middle 8 bits of the register.

E = 4 N + 3 ; to access the lower 8 bits of the register.

12.3.4 Accessing the packet queues

Although entire incoming packets are stored in the receiver buffer memory they are not randomly accessible. These buffers act like fifos and only the frontmost (oldest) data quadlet entry is accessible for reading. Therefore only one location (register address) is allocated to each of the two receiver queues. Reading this location returns the head entry of the queue, and at the same time removes it from the queue, making the next stored data quadlet accessible.

With the current host interface such a read is in fact a move operation of the data quadlet from the queue to the read shadow register. Once the data is copied into the read shadow register it is no longer available in the queue itself so the CPU should always read all 4 bytes before attempting any other read access (be careful with interrupt handlers for AVLink!).

A similar argument applies to the transmitter queues. Data cannot be written arbitrarily, but only to the next available free location. Since the transmitter needs to know when the packet is complete (all data stored in memory, so that it may start the arbitration process on the 1394 bus) two separate register locations are reserved per transmitter queue: one to write all but the last packet quadlet to, and one to write the last quadlet of every packet to. Writing to any of these register locations stores the data in the queue and makes the next memory location accessible for writing.

NOTE:

1. Because of the way it is implemented memory access is not always immediate; consequently it may take some time before the next data quadlet in the queue is accessible after reading or writing the current one. Status flags are provided to the CPU to indicate availability.

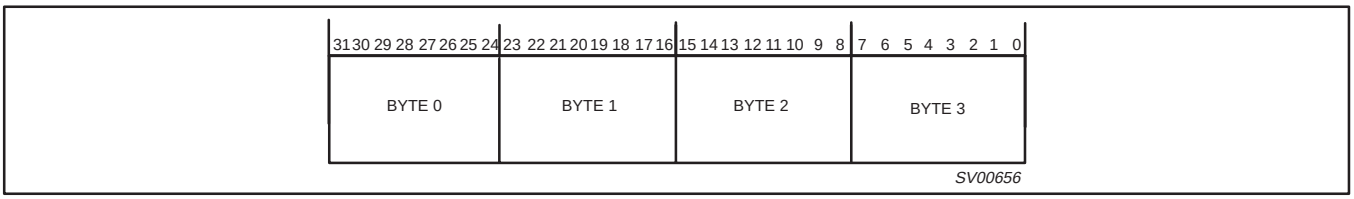


Figure 1. Byte order in quadlets as implemented in the host interface

12.3.5 The CPU bus interface signals

The CPU interface is directly compatible with an 8051 microcontroller. It uses a separate HIF RD_N and HIF WR_N inputs and a HIF CS_N chip select line, all of which are active LOW. There are 9 address inputs (HIF A0..HIF A8) and 8 data in/out lines HIF D0..HIF D7. An open drain HIF INT_N output is used to signal interrupts to the CPU.

The CPU is not required to run at a clock that is synchronous to the 1394 base clock. The control signals will be resampled by the host interface before being used internally.

An access through the host interface starts when HIF CS_N = 0 and either HIF WR_N = 0 or HIF RD_N = 0. Typically the chip select signal is derived from the upper address lines of the CPU (address decode stage), but it could also be connected to a port pin of the CPU to avoid the need for an external address decoder in very simple CPU systems. When both HIF CS_N = 0 and HIF RD_N = 0 the host interface will start a read access cycle, so the cycle is triggered at the falling edge of either HIF CS_N or HIF RD_N, whichever is later.

Very shortly after the start of the cycle, the selected byte in the **read shadow register** will be output (indicated in Figure 2 as RSR_O). If HIF A8 is asserted then the target register value will be copied into the **read shadow register**, leading to a new value RSR_n some time later in the read cycle. If HIF A8 is LOW, then the **read shadow register** will not change.

A write access starts when the later of HIF CS_N and HIF WR_N becomes LOW (see Figure 3). Data is written to the shadow register, following which, if HIF A8 is asserted, the shadow register value is copied to the addressed register.

NOTES:

1. The time between the end of any access and the start of the next access must be at least t_{CH} which needs to be greater than (2 x SCLK).
2. When HIF A8 = 0 for either write or read access the address bits HIF A2..HIF A7 are ignored.
3. If both HIF WR_N = 0 and HIF RD_N = 0 while HIF CS_N = 0, then a write cycle takes place.

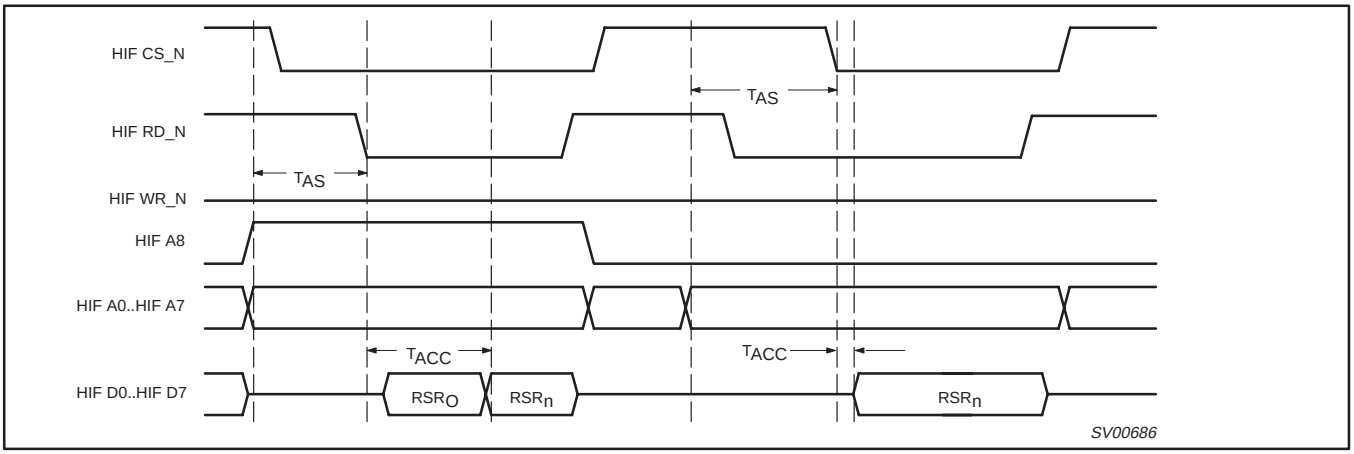


Figure 2. Read cycle signal timing (2 independent read cycles)

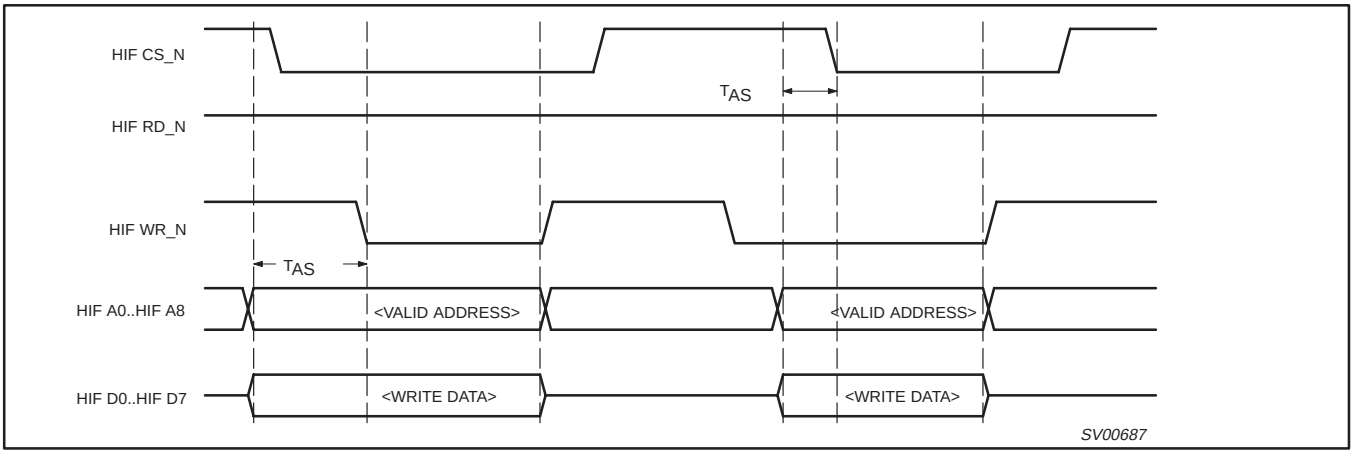


Figure 3. Write cycle signal timing (2 independent write cycles)

12.4 The Asynchronous Packet Interface

The PDI1394L11 provides an interface to asynchronous data packets through the registers in the host interface. The format of the asynchronous packets is specified in the following sections.

12.4.1 Reading an Asynchronous Packet

Upon reception of a packet, the packet data is stored in the appropriate receive FIFO, either the Request or Response FIFO. The location of the packet is indicated by either the RREQQQAV or RRSPQAV status bit being set in the Asynchronous Interrupt Acknowledge (ASYINTACK) register. The packet is transferred out of the FIFO by successive reads of the Asynchronous Receive Request (RREQ) or Asynchronous Receive Response (RRSP) register. The end of the packet (the last quadlet) is indicated by either the RREQQLASTQ or RRSPQLASTQ bit set in ASYINTACK. Attempting to read the FIFO when either RREQQQAV bit or RRSPQQAV bit is set to 0 (in the Asynchronous RX/TX interrupt acknowledge (ASYINTACK) register) will result in a queue read error.

12.4.2 Writing an Asynchronous Packet

An asynchronous packet intended for transmission is first stored in the appropriate Transmitter FIFO. Once writing to the FIFO is complete, the link layer controller arbitrates for the bus to transmit the packet.

To generate an asynchronous packet, the first and next to last quadlets of the packet must be written to the Asynchronous Transmit Request Next (TX_RQ_NEXT) register, for request type packets, or the Asynchronous Transmit Response Next (TX_RP_NEXT) register, for response type packets. The last quadlet of the packet is written to the Asynchronous Transmit Request Last (TX_RQ_LAST) register, for request type packets, or the Asynchronous Transmit Response Last (TX_RP_LAST) register, for response type packets. After writing the last quadlet, the packet is automatically queued by the AVlink layer controller for transmission over the bus.

12.5 Link Packet Data Formats

The data formats for transmission and reception of data are shown below. The transmit format describes the expected organization for data presented to the link at the asynchronous transmit, physical response, or isochronous transmit FIFO interfaces.

12.5.1 Asynchronous Transmit Packet Formats

These sections describe the formats in which packets need to be delivered to the queues (FIFOs) for transmission. There are four basic formats as follows:

ITEM	FORMAT	USAGE	TRANSACTION CODE (tCode)
1	No-packet data	Quadlet read requests	4
		Quadlet/block write responses	2
2	Quadlet packet	Qaudlet write requests	0
		Quadlet read responses	6
		Block read requests	5
3	Block Packet	Block write requests	1
		Block read responses	7
		Lock requests	9
		Lock responses	B_{hex}
4	Unformatted transmit	Concatenated self-ID / PHY packets	E_{hex}

Each packet format uses several fields (see names and descriptions below). More information about these fields (not the format) can be found in the 1394 specification. Grey fields are reserved and should be set to zero values.

12.5.1.1 No-data Transmit

The no-data transmit formats are shown in Figures 4 and 5. The first quadlet contains packet control information. The second and third quadlets contain 16-bit destination ID and either the 48-bit, quadlet aligned destination offset (for requests) or the response code (for responses).

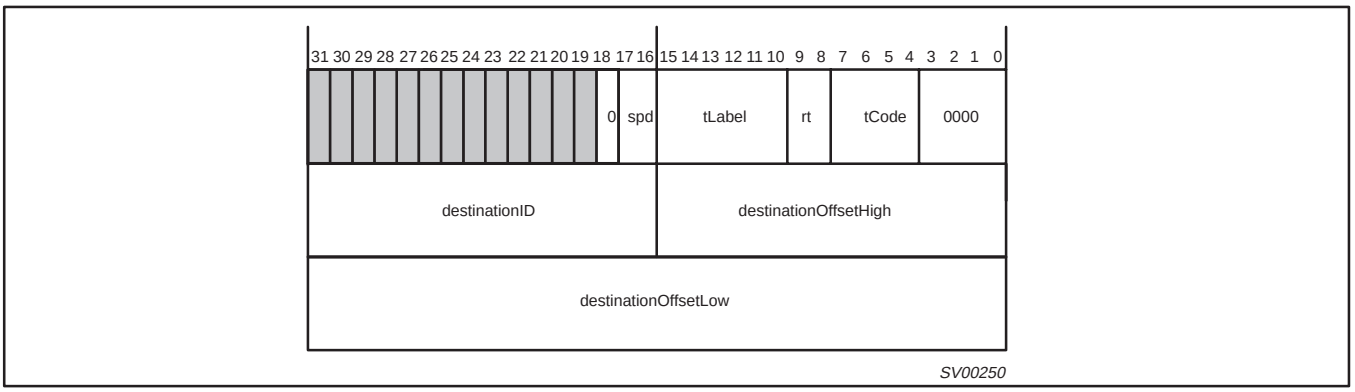


Figure 4. Quadlet Read Request Transmit Format

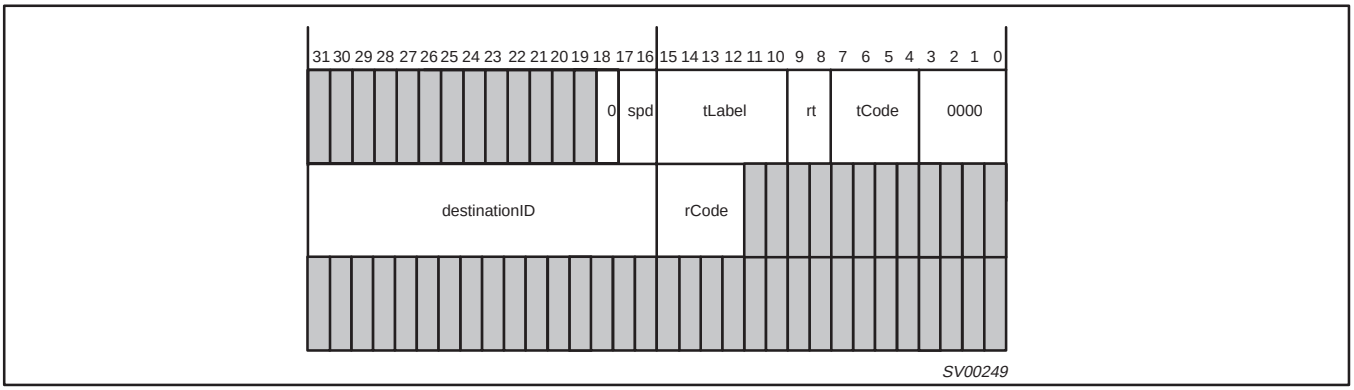


Figure 5. Quadlet/Block Write Response Packet Transmit Format

Table 1. No-Data Transmit Format

Field Name	Description
spd	This field indicates the speed at which this packet is to be sent. 00=100 Mbs, 01=200 Mbs, and 10=400 Mbs. 11 = undefined
tLabel	This field is the transaction label, which is used to pair up a response packet with its corresponding request packet.
rt	The retry code for this packet. Supported values are: 00=retry1, and 01=retryX.
tCode	The transaction code for this packet.
DestinationID	Contains a node ID value.
DestinationOffsetHigh DestinationOffsetLow	The concatenation of these two field addresses a quadlet in the destination node's address space.
rCode	Response code for write response packet.

12.5.1.2 Quadlet Transmit

Three quadlet transmit formats are shown below. In these figures: The first quadlet contains packet control information. The second and third quadlets contain 16-bit destination ID and either the 48-bit quadlet-aligned destination offset (for requests) or the response code (for responses).

The fourth quadlet contains the quadlet data for read response and write quadlet request formats, or the upper 16 bits contain the data length for the block read request format.

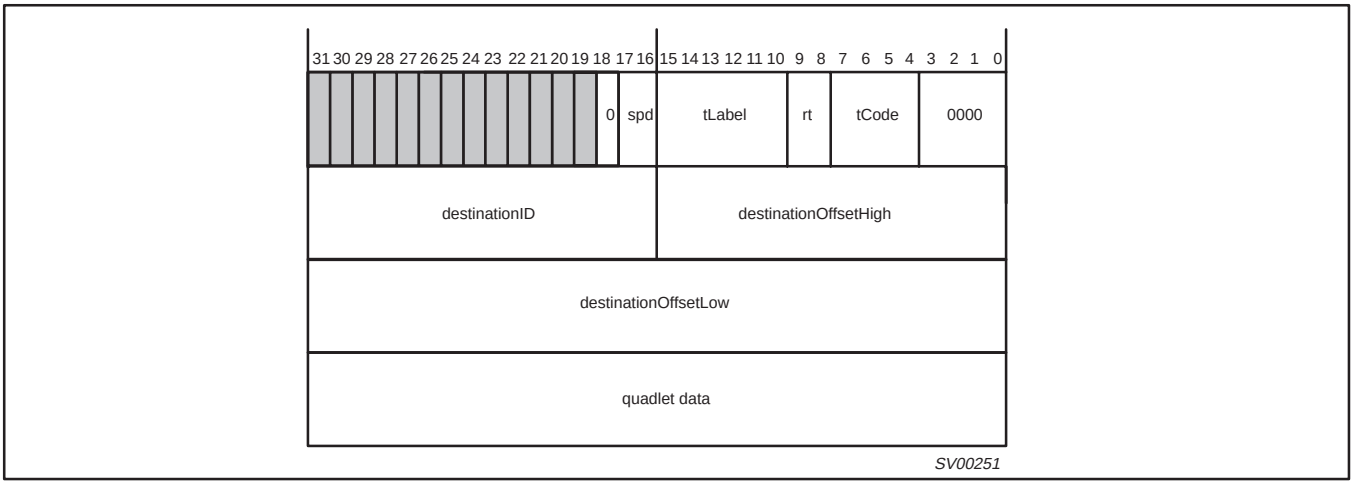


Figure 6. Quadlet Write Request Transmit Format

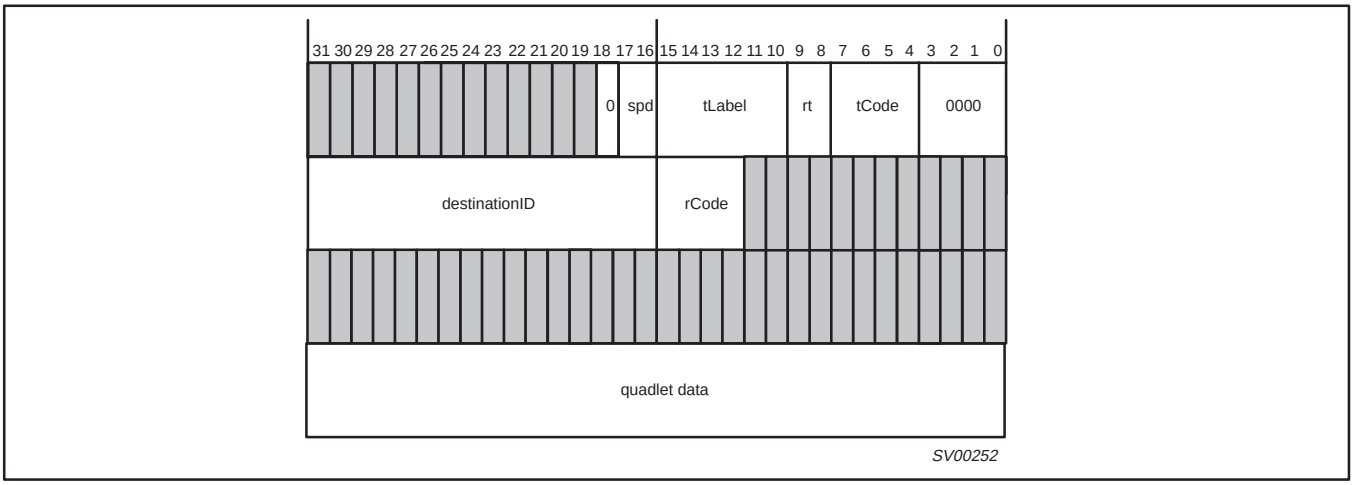


Figure 7. Quadlet Read Response Transmit Format

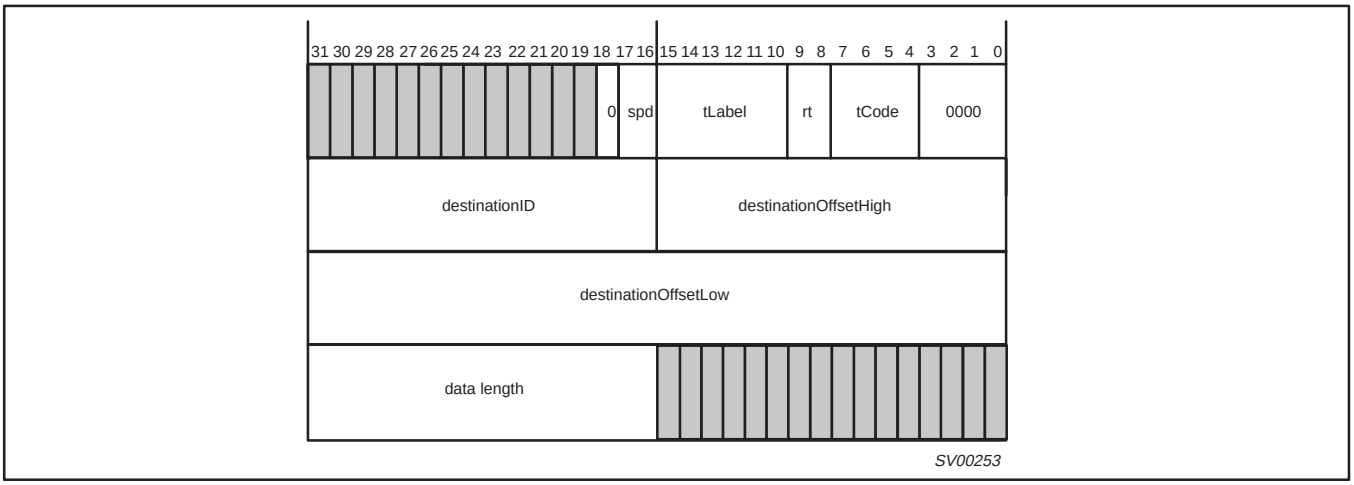


Figure 8. Block Read Request Transmit Format

Table 2. Quadlet Transmit Fields

Field Name	Description
spd, tLabel, rt, tCode, destinationID, destinationOffsetHigh, destinationOffsetLow, rCode	See Table 1.
Quadlet data	For quadlet write requests and quadlet read responses, this field holds the data to be transferred
Data length	The number of bytes requested in a block read request

12.5.1.3 Block Transmit

The block transmit format is shown below, this is the generic format for reads and writes. The first quadlet contains packet control information. The second and third quadlets contain the 16-bit destination node ID and either the 48-bit destination offset (for requests) or the response code and reserved data (for responses). The fourth quadlet contains the length of the data field and the extended transaction code (all zeros except for lock transaction). The block data, if any, follows the extended transaction code.

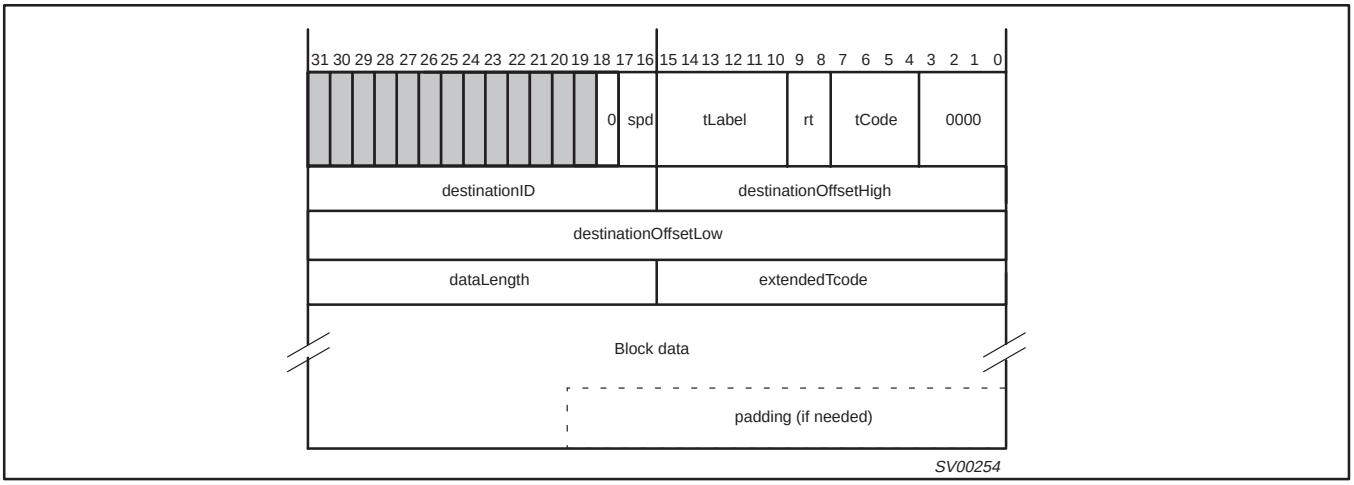


Figure 9. Block Packet Transmit Format

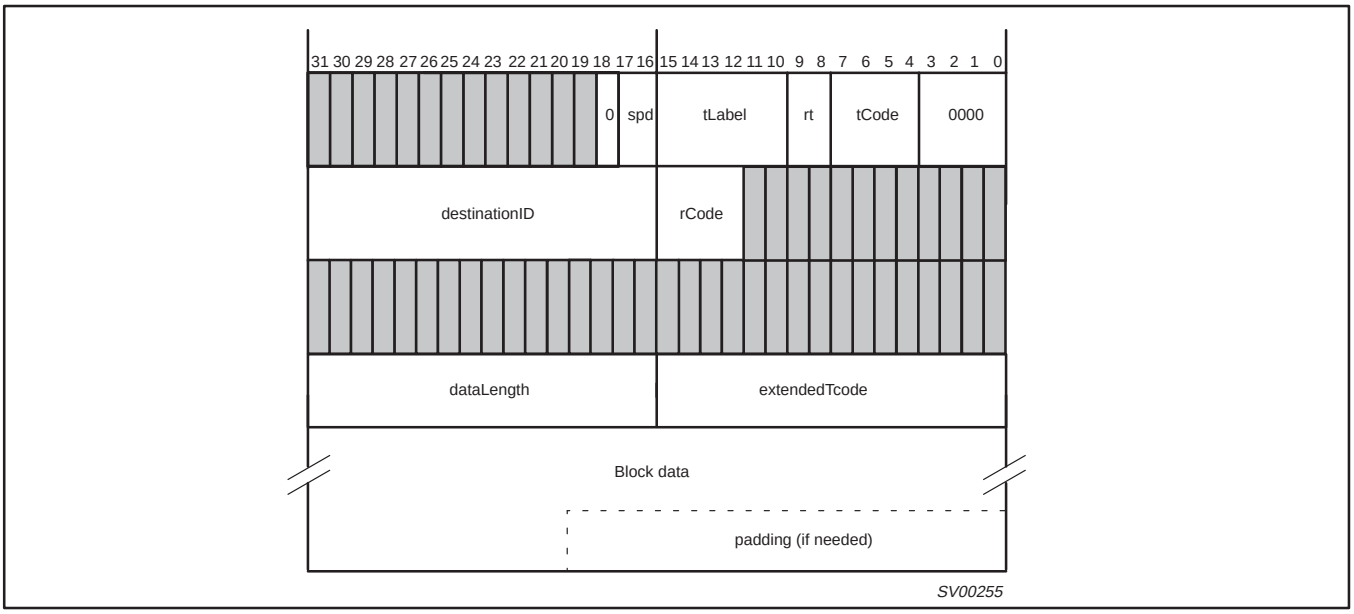


Figure 10. Block Read or Lock Response Transmit Format

Table 3. Block Transmit Field

Field Name	Description
spd, tLabel, rt, tCode, destinationID, destinationOffsetHigh, destinationOffsetLow, rCode	See Table 2.
dataLength	The number of bytes of data to be transmitted in this packet
extendedTcode	The tCode indicates a lock transaction, this specifies the actual lock action to be performed with the data in this packet.
block data	The data to be sent. If dataLength=0, no data should be written into the FIFO for this field. Regardless of the destination or source alignment of the data, the first byte of the block must appear in the high order byte of the first quadlet.
padding	If the dataLength mod 4 is not zero, then zero–value bytes are added onto the end of the packet to guarantee that a whole number of quadlets is sent.

12.5.1.4 Unformatted Transmit

The unformatted transmit format is shown in Figure 11. The first quadlet contains packet control information. The remaining quadlets contain data that is transmitted without any formatting on the bus. No CRC is appended on the packet, nor is any data in the first quadlet sent. This is used to send PHY configuration and Link-on packets. Note that the bit-inverted check quadlet must be included in the FIFO since the AV Link core will not generate it.

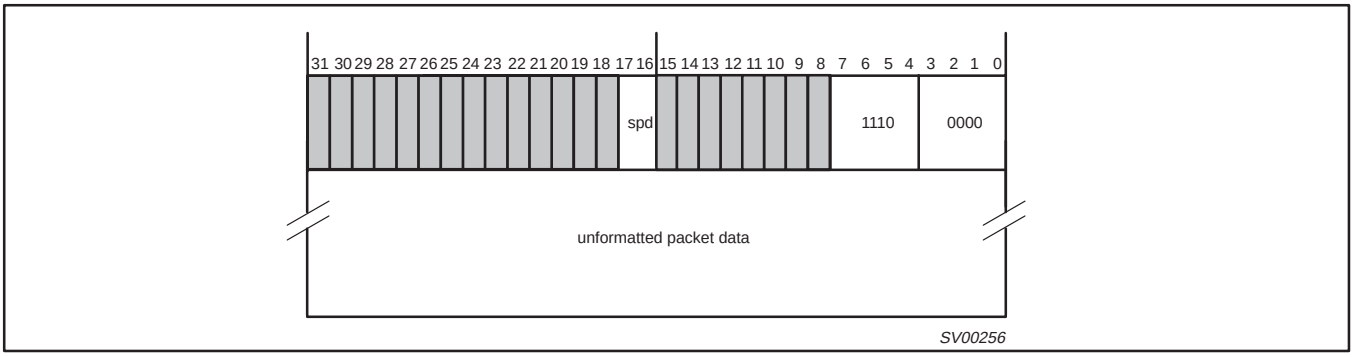


Figure 11. Unformatted Transmit Format

12.5.2 Asynchronous Receive Packet Formats

This section describes the asynchronous receive packet formats. Four basic asynchronous data packet formats and one confirmation format exist:

Table 4. Asynchronous Data Packet Formats

ITEM	FORMAT	USAGE	TRANSACTION CODE
1	No-packet data	Quadlet read requests	4
		Quadlet/block write responses	2
2	Quadlet packet	Qaudlet write requests	0
		Quadlet read responses	6
3	Block Packet	Block read requests	5
		Block write requests	1
		Block read responses	7
		Lock requests	9
		Lock responses	B_{hex}
4	Self-ID / PHY packet	Concatenated self-ID / PHY packets	E_{hex}
5	Confirmation packet	Confirmation of packet transmission	8

Each packet format uses several fields. More information about most of these fields can be found in the 1394 specification.

Table 5. Asynchronous Receive Fields

Field Name	Description
destinationID	This field is the concatenation of busNumbers (or all ones for “local bus”) and nodeNumbers (or all ones for broadcast) for this node.
tLabel	This field is the transaction label, which is used to pair up a response packet with its corresponding request packet.
rt	The retry code for this packet. 00=retry1, 01=retryX, 10=retryA, 11=retryB.
tCode	The transaction code for this packet.
priority	The priority level for this packet (0000 for cable environment).
sourceID	This is the node ID of the sender of this packet.
destinationOffsetHigh, destinationOffsetLow	The concatenation of these two field addresses a quadlet in this node’s address space.
rCode	Response code for response packets.
quadlet data	For quadlet write requests and quadlet read responses, this field holds the data received.
dataLength	The number of bytes of data to be received in a block packet.
extendedTcode	If the tCode indicates a lock transaction, this specifies the actual lock action to be performed with the data in this packet.
block data	The data received. If dataLength=0, no data will be written into the FIFO for this field. Regardless of the destination or source alignment of the data, the first byte of the block will appear in the high order byte of the first quadlet.
padding	If the dataLength mod 4 is not zero, then zero–value bytes are added onto the end of the packet to guarantee that a whole number of quadlets is sent.
u	Unsolicited response tag bit. This bit is set to one (1) if the received response was unsolicited.
ackSent	This field contains the acknowledge code that the link layer returned to the sender of the received packet. For packets that do not need to be acknowledged (such as broadcasts) the field contains the acknowledge value that would have been sent if an acknowledge had been required. The values for this field are listed in Table 6 (they also can be found in the IEEE 1394 standard).

Table 6. Acknowledge codes

Code	Name	Description
0001	ack_complete	The node has successfully accepted the packet. If the packet was a request subaction, the destination node has successfully completed the transaction and no response subaction shall follow.
0010	ack_pending	The node has successfully accepted the packet. If the packet was a request subaction, a response subaction will follow at a later time. This code shall not be returned for a response subaction.
0100	ack_busy_X	The packet could not be accepted. The destination transaction layer may accept the packet on a retry of the subaction.
0101	ack_busy_A	The packet could not be accepted. The destination transaction layer will accept the packet when the node is not busy during the next occurrence of retry phase A.
0110	ack_busy_B	The packet could not be accepted. The destination transaction layer will accept the packet when the node is not busy during the next occurrence of retry phase B.
1101	ack_data_error	The node could not accept the block packet because the data field failed the CRC check, or because the length of the data block payload did not match the length contained in the dataLength field. This code shall not be returned for any packet that does not have a data block payload.
1110	ack_type_error	A field in the request packet header was set to an unsupported or incorrect value, or an invalid transaction was attempted (e.g., a write to a read-only address).
0000, 0011, 0111 – 1100, and 1111	reserved	

12.5.2.1 No-Data Receive

The no-data receive formats are shown below. The first quadlet contains the destination node ID and the rest of the packet header. The second and third quadlet contain 16-bit source ID and either the 48-bit, quadlet-aligned destination offset (for requests) or the response code (for responses). The last quadlet contains packet reception status.

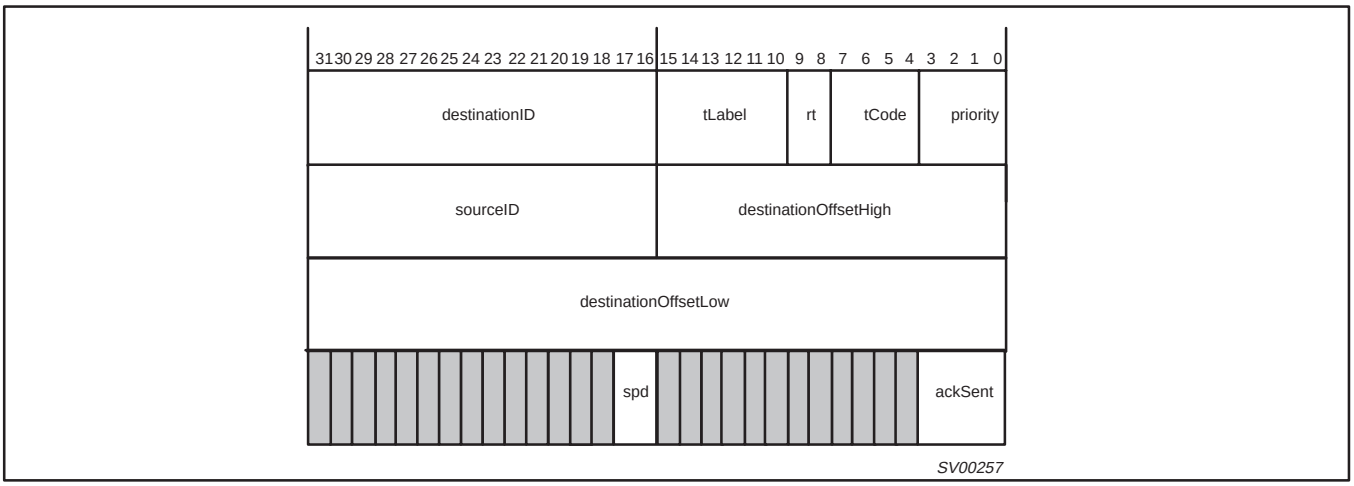


Figure 12. Quadlet Read Request Receive Format

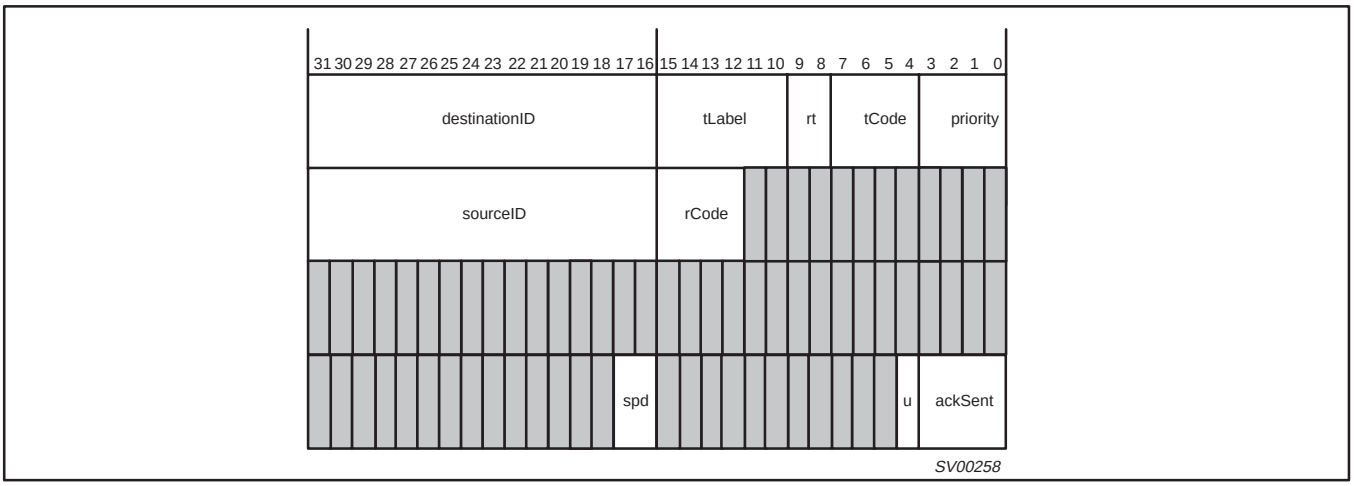


Figure 13. Write Response Receive Format

12.5.2.2 Quadlet Receive

The quadlet receive formats are shown below. The first quadlet contains the destination node ID and the rest of the packet header. The second and third quadlets contain 16-bit source ID and either the 48-bit, quadlet-aligned destination offset (for requests) or the response code (for responses). The fourth quadlet is the quadlet data for read responses and write quadlet requests, and is the data length and reserved for block read requests. The last quadlet contains packet reception status.

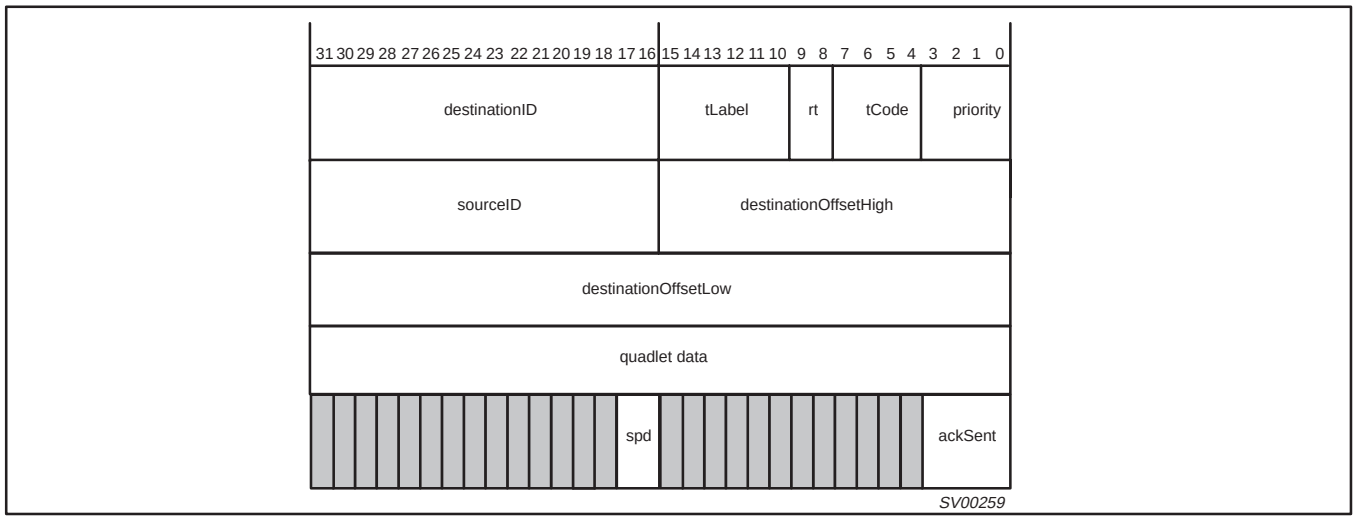


Figure 14. Quadlet Write Request Receive Format

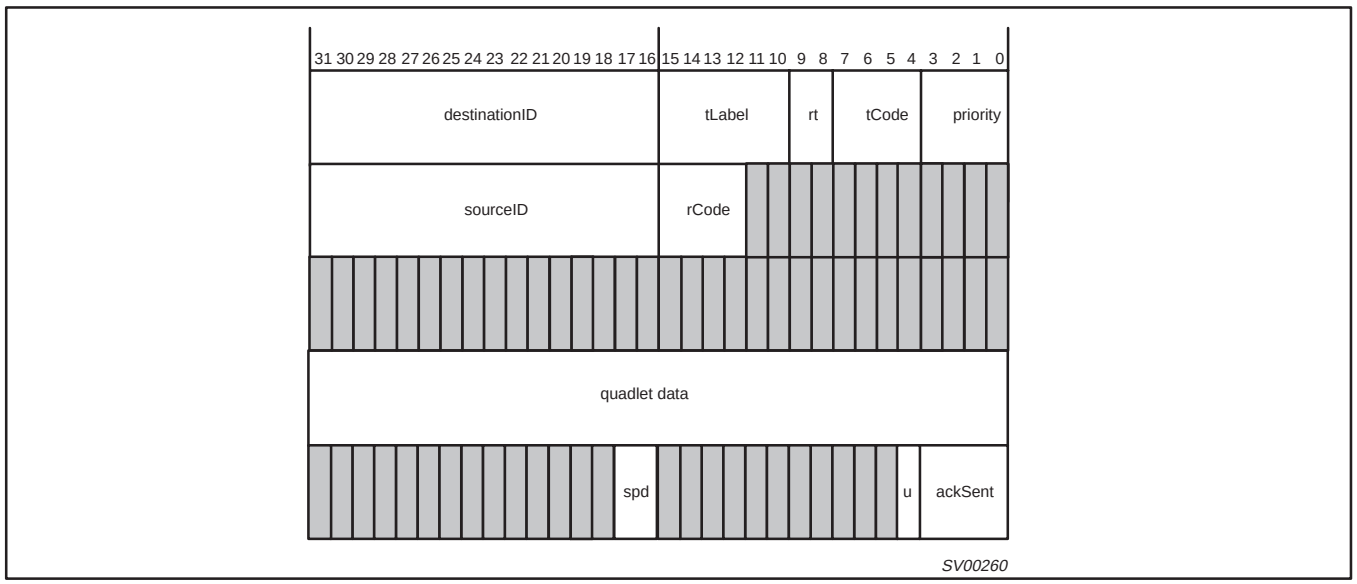


Figure 15. Quadlet Read Response Receive Format

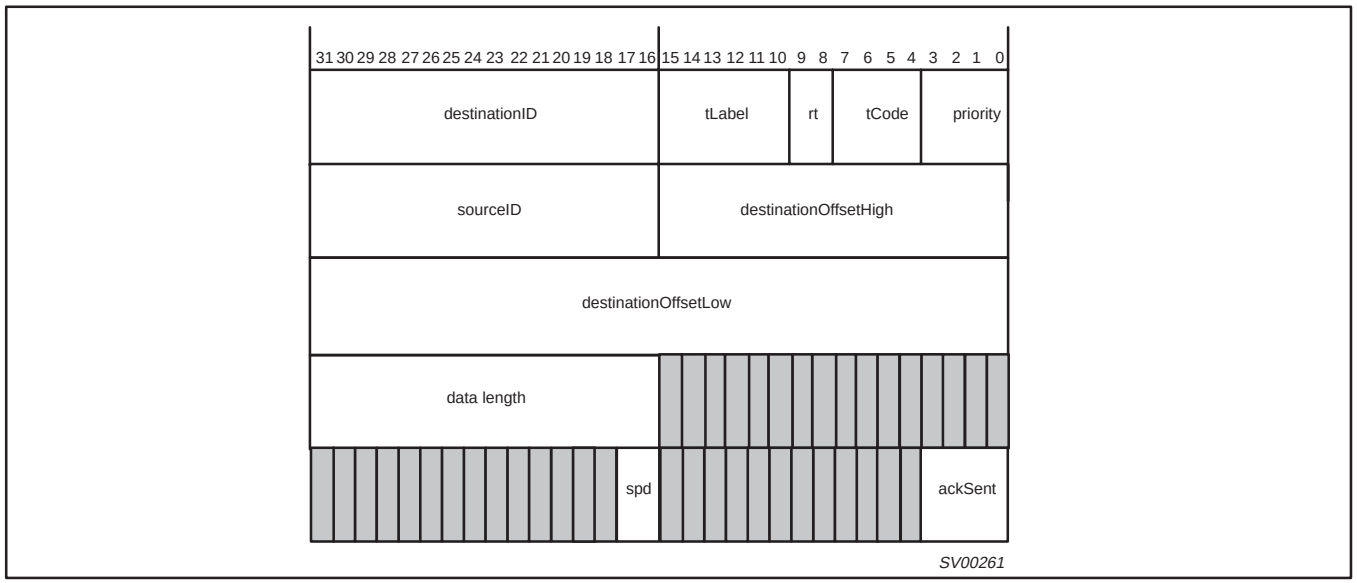


Figure 16. Block Read Request Receive Format

12.5.2.3 Block receive

The block receive format is shown below. The first quadlet contains the destination node ID and the rest of the packet header. The second and third quadlets contain 16-bit sourceID and either the 48-bit destination offset (for requests) or the response code and reserved data (for responses). The fourth quadlet contains the length of the data field and the extended transaction code (all zeros except for lock transactions). The block data, if any, follows the extended code. The last quadlet contains packet reception status.

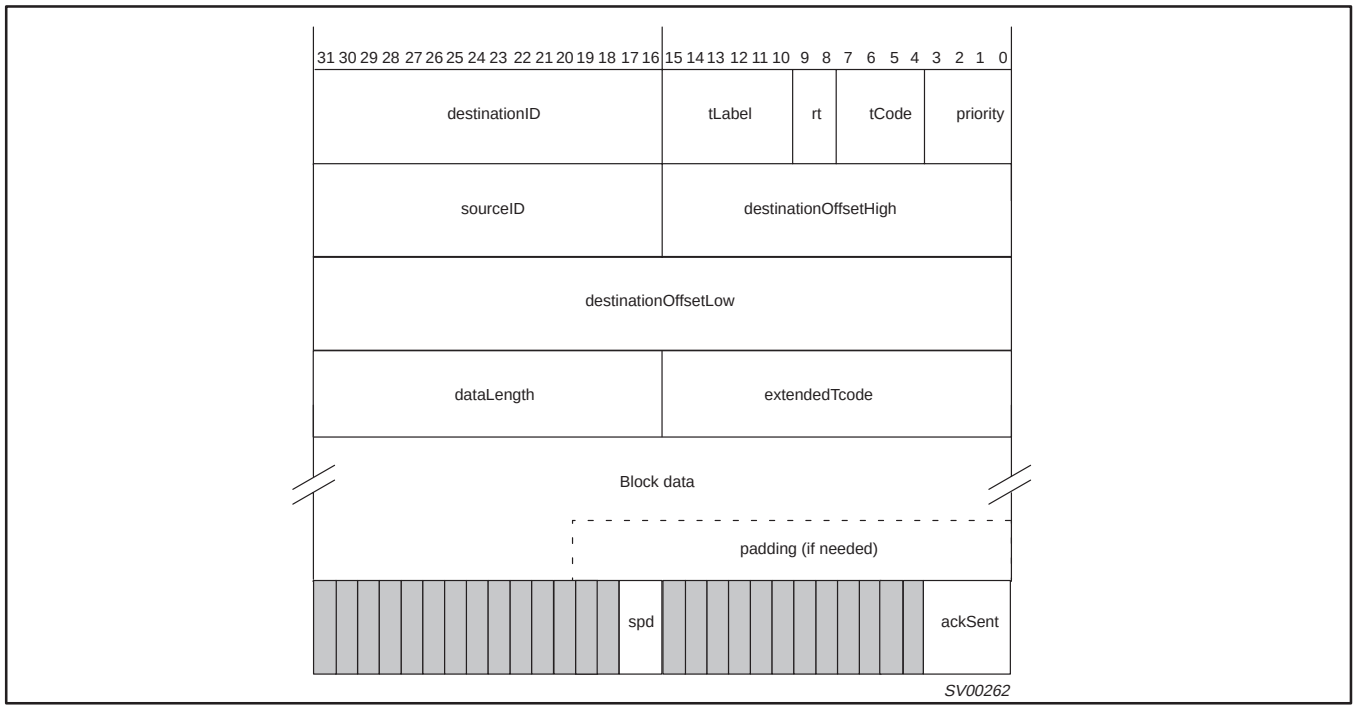


Figure 17. Block Write or Lock Request Receive Format

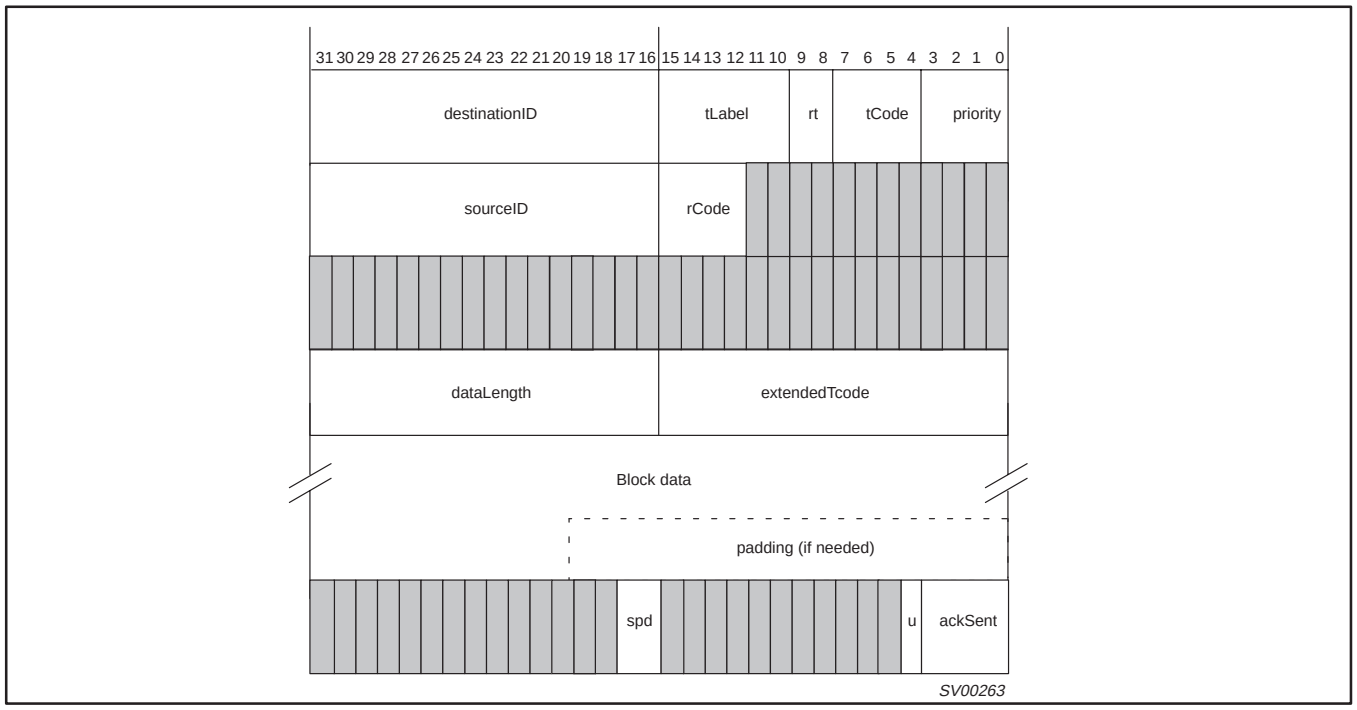


Figure 18. Block Read or Lock Response Receive Format

12.5.2.4 Self-ID and PHY packets receive

The self-ID and PHY packet receive formats are shown below. The first quadlet contains a synthesized packet header with a tCode of 0xE (hex). For self-ID information, the remaining quadlets contain data that is received from the time a bus reset ends to the first subaction gap. This is the concatenation of all the self-ID packets received. Note that the bit-inverted check quadlet is included in the Read Request FIFO and the application must check it.

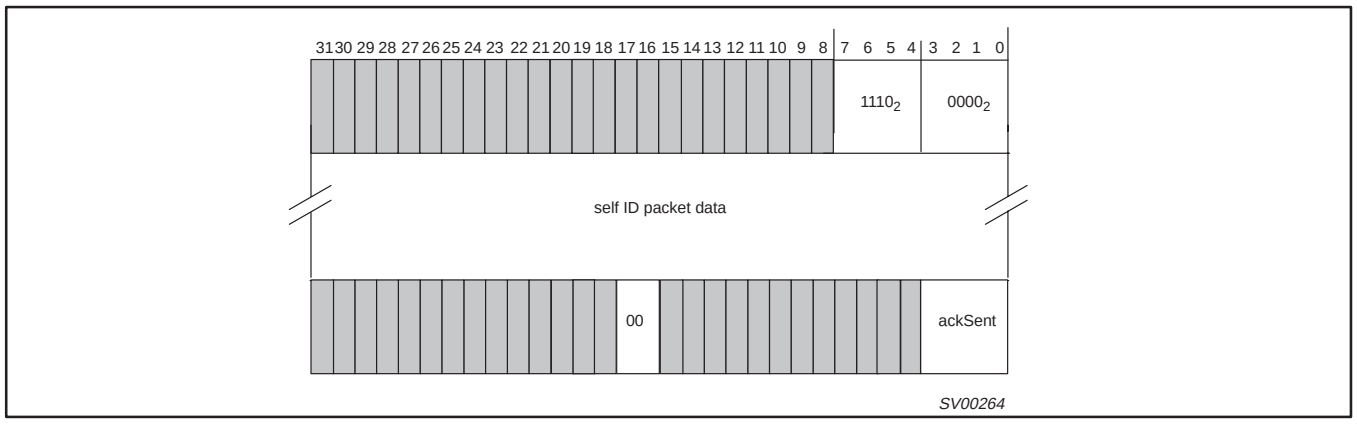


Figure 19. Self-ID Receive Format

The “ackSent” field will either be “ACK_DATA_ERROR” if a non-quadlet-aligned packet is received or there was a data overrun, or “ACK_COMPLETE” if the entire string of self-ID packets was received.

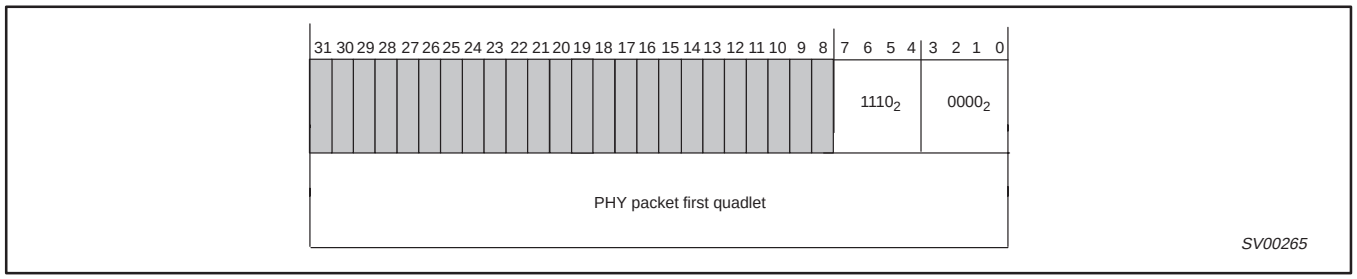


Figure 20. PHY Packet Receive Format

For PHY packets, there is a single following quadlet which is the first quadlet of the PHY packet. The check quadlet has already been verified and is not included.

12.5.2.5 Transaction data confirmation formats

After a packet from one of the queues has been transmitted, the asynchronous transmitter assembles a confirmation (see Figure 21) which is used to confirm the result of the transmission to the higher layers. Separate confirmations are assembled for request and response transmissions. Request confirmations are written into the request queue and response confirmations are written into the response queue.

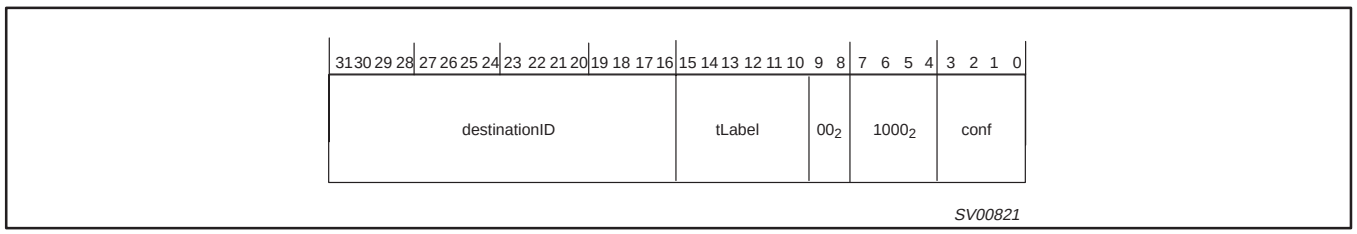


Figure 21. Request and response confirmation format

Table 7. Confirmation codes

CODE[1]	DESCRIPTION
0	Non-broadcast packet transmitted; addressed node returned no acknowledge.
1	Broadcast packet transmitted or non-broadcast packet transmitted; addressed node returned an acknowledge complete.
2	Non-broadcast packet transmited; addressed node returned an acknowledge pending.
4	Retry limit exceeded; destination node hasn't accepted the non-broadcast packet within the maximum number of retries.
D_{16}	Acknowledge data error received (transaction complete).
E_{16}	Acknowledge type error received (transaction complete).

NOTE:
1. All other codes are reserved.

For every packet written in a transmitter queue by the CPU, there will be one confirmation written in the corresponding receiver queue by the AV layer logic.

12.5.3 Interrupts

The PDI1394L11 provides a single interrupt line (HIF INT_N) for connection to a host controller. Status indications from four major areas of the device are collected and ORed together to activate HIF INT_N. Status from four major areas of the device are collected in four status registers; LNKPHYINTACK, ITXINTACK, IRXINTACK, and ASYINTACK. At this level, each individual status can be enabled to generate a chip-level interrupt by activating HIF INT_N. To aid in determining the source of a chip-level interrupt, the major area of the device generating an interrupt is indicated in the lower 4 bits of the GLOBCSR register. These bits are non-latching Read-Only status bits and do not need to be acknowledged. To acknowledge and clear a standing interrupt, the bit in LNKPHYINTACK, ITXINTACK, IRXINTACK, or ASYINTACK causing the interrupt status has to be written to a logic '1'; Note: Writing a value of '0' to the bit has no effect.

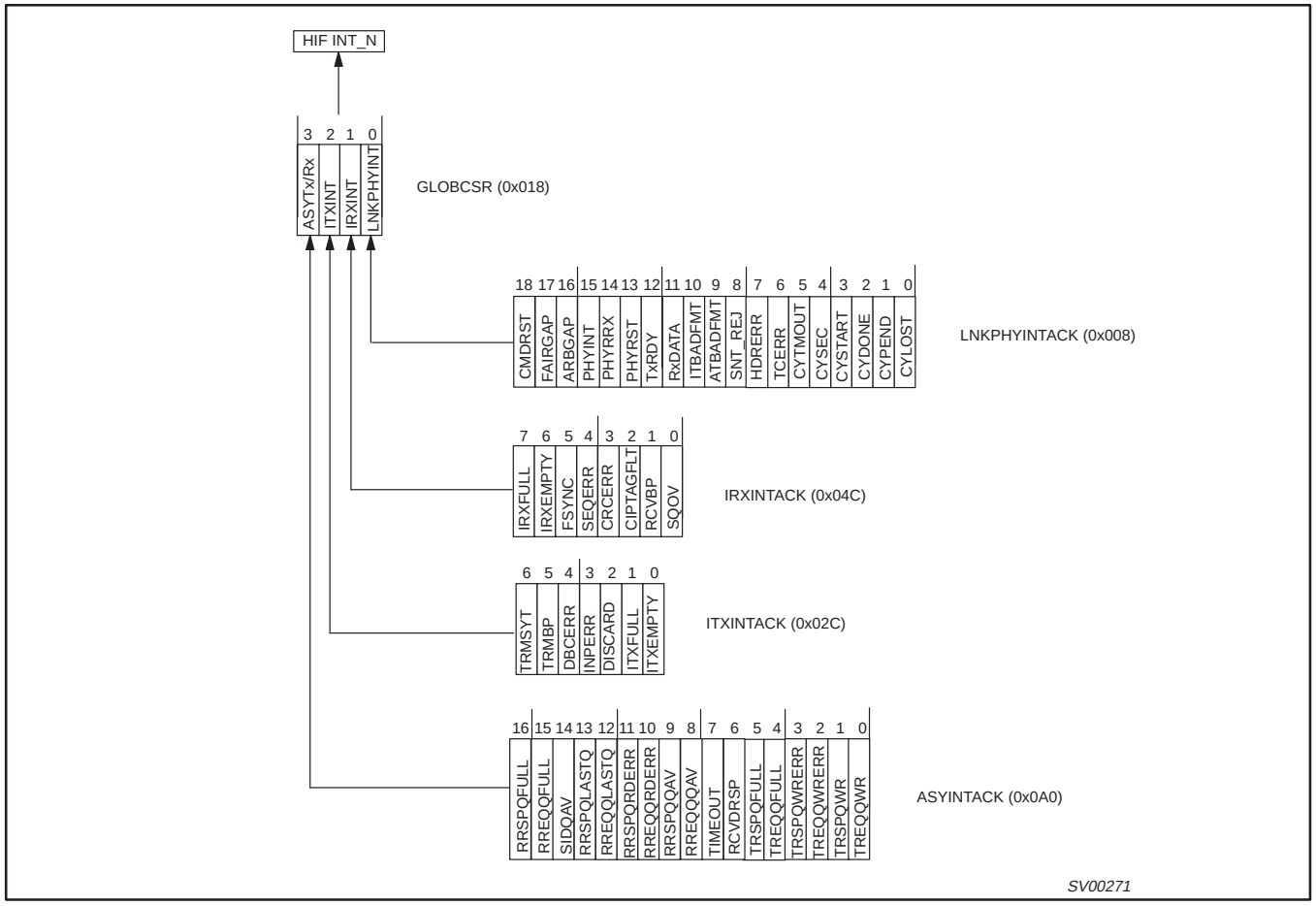
12.5.3.1 Determining and Clearing Interrupts

When responding to an interrupt event generated by the PDI1394L11, or operating in polled mode, the first register examined is the GLOBCSR register. The least significant nibble contains interrupt status bits from general sections of the device; the link layer controller, the AV transmitter, the AV receiver, and the asynchronous transceiver. The bits in GLOBCSR[3:0] are self clearing status bits. They represent the logical OR of all the enabled interrupt status bits in their section of the AV Link Layer Controller.

Once an interrupt, or status is detected in GLOBCSR, the appropriate interrupt status register needs to be read, see the Interrupt Hierarchy diagram for more detail. After all the interrupt indications are dealt with in the appropriate interrupt status register, the interrupt status indication will automatically clear in the GLOBCSR.

All interrupt status bits in the various interrupt status registers are latching unless otherwise noted.

12.5.3.2 Interrupt Hierarchy



13.0 REGISTER MAP

Registers are 32 bits (quadlet) wide and all accesses are always done on a quadlet basis. This means that it is not possible to write just the lower 8 bits, and leave the other bits unaffected (see Section 12.3.2 for more information). The values written to undefined fields/bits are ignored and thus DON'T CARE.

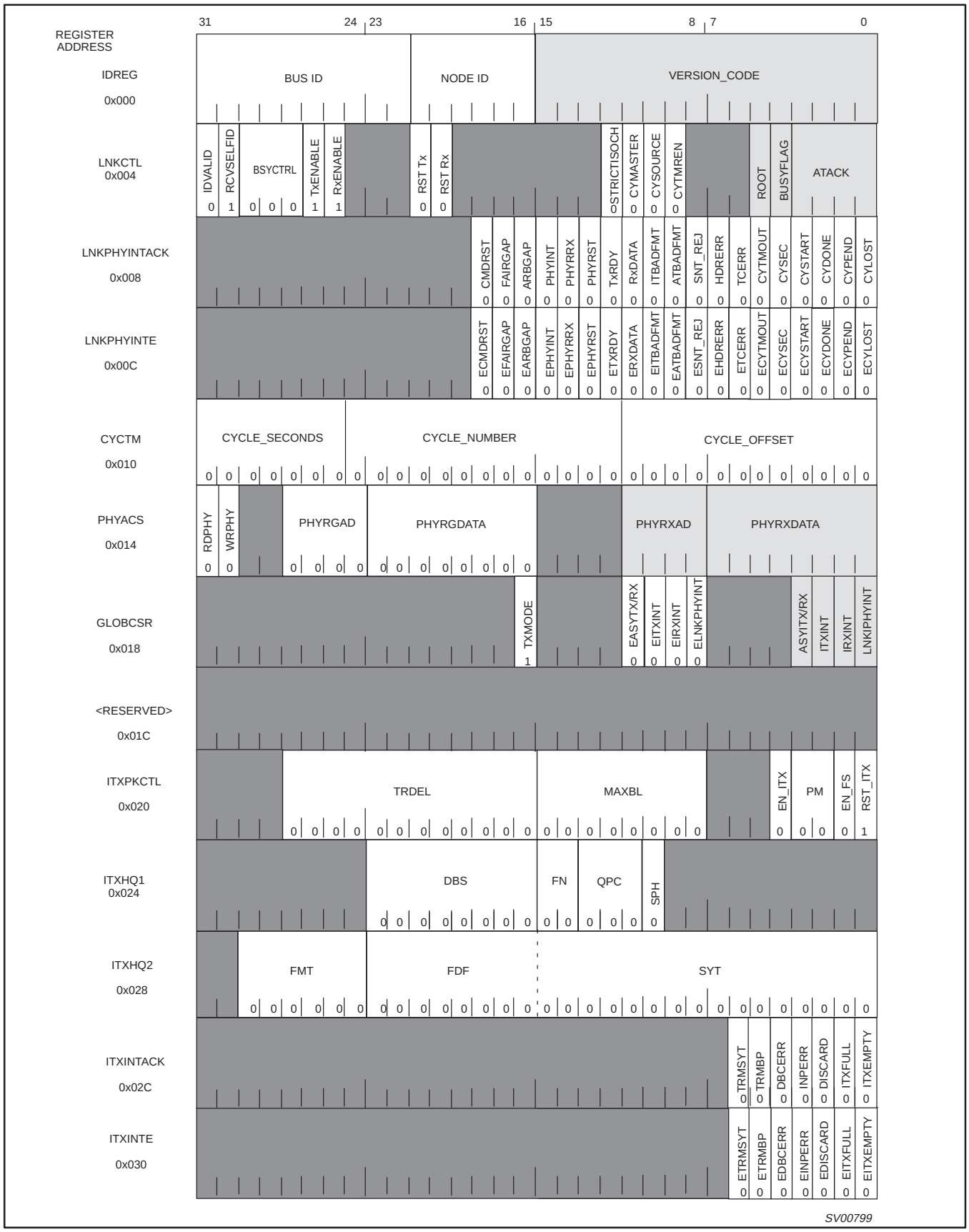
A full bitmap of all registers is listed in Table 8. The meaning of shading and bit cell values is as follows:

A bit/field with no name written in it and dark shading is reserved and not used.

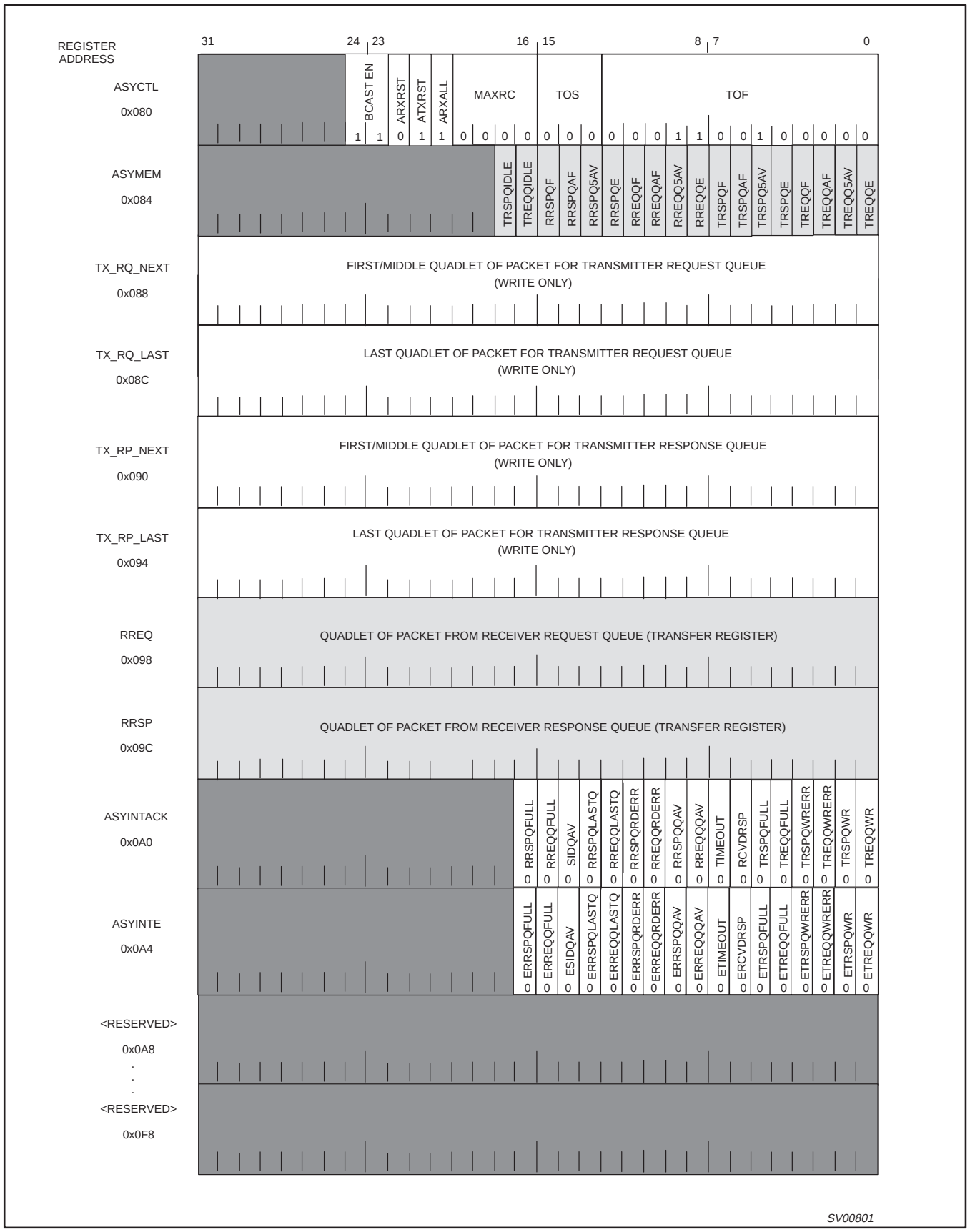
A bit/field with a name in it and light shading is a READ ONLY (status) bit/field.

A one bit value (0 or 1) written at the bottom of a writable (control) bit is the default value after power-on-reset.

Table 8. Full Bitmap of all Registers (consists of three tables shown on the following pages)


REGISTER ADDRESS
31
24
23
16
15
8
7
0
IDREG 0x000
BUS ID
NODE ID
VERSION_CODE
LNKCTL 0x004
IDVALID 0
RCVSELFID 1
BSYCTRL 0 0 0
TxENABLE 1
RxENABLE 1
RST Tx 0
RST Rx 0
STRICTISOCH 0
CYMASTER 0
CYSOURCE 0
CYTMREN 0
ROOT
BUSYFLAG
ATACK
LNKPHYINTACK 0x008
CMDRST 0
FAIRGAP 0
ARBGAP 0
PHYINT 0
PHYRRX 0
PHYRST 0
TxRDY 0
RxDATA 0
ITBADFMT 0
ATBADFMT 0
SNT_REJ 0
HDRERR 0
TCERR 0
CYTMOUT 0
CYSEC 0
CYSTART 0
CYDONE 0
CYPEND 0
CYLOST 0
LNKPHYINTE 0x00C
ECMDRST 0
EFAIRGAP 0
EARBGAP 0
EPHYINT 0
EPHYRRX 0
EPHYRST 0
ETXRDY 0
ERXDATA 0
EITBADFMT 0
EATBADFMT 0
ESNT_REJ 0
EHDRERR 0
ETCERR 0
ECYTMOUT 0
ECYSEC 0
ECYSTART 0
ECYDONE 0
ECYPEND 0
ECYLOST 0
CYCTM 0x010
CYCLE_SECONDS
CYCLE_NUMBER
CYCLE_OFFSET
PHYACS 0x014
RDPHY 0
WRPHY 0
PHYRGAD 0 0 0 0
PHYRGDATA
PHYRXAD
PHYRXDATA
GLOBCSR 0x018
TXMODE 1
EASYTX/RX 0
EITXINT 0
EIRXINT 0
ELNKPHYINT 0
ASYITX/RX
ITXINT
IRXINT
LNKIPHYINT
<RESERVED> 0x01C
ITXPKCTL 0x020
TRDEL
MAXBL
EN_ITX 0
PM 0 0
EN_FS 0
RST_ITX 1
ITXHQ1 0x024
DBS
FN 0 0
QPC 0 0 0
SPH 0
ITXHQ2 0x028
FMT 0 0 0 0 0 0
FDF
SYT
ITXINTACK 0x02C
TRMSYT 0
TRMBP 0
DBCERR 0
INPERR 0
DISCARD 0
ITXFULL 0
ITXEMPTY 0
ITXINTE 0x030
ETRMSYT 0
ETRMBP 0
EDBCERR 0
EINPERR 0
EDISCARD 0
EITXFULL 0
EITXEMPTY 0
SV00799

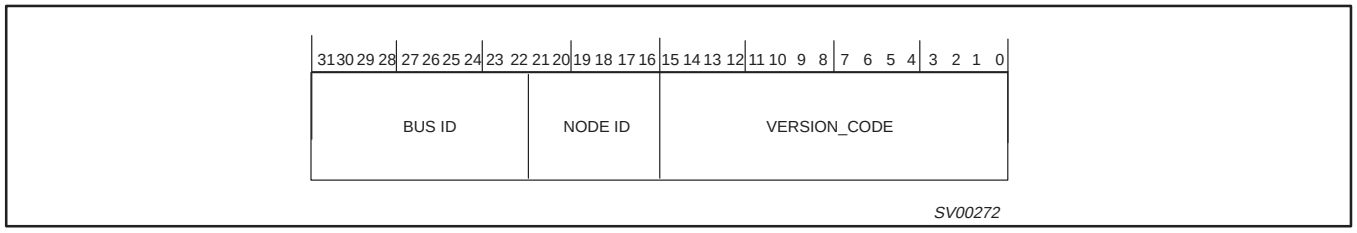

REGISTER ADDRESS
31
24
23
16
15
8
7
0
ITXCTL
0x034
TAG
CHANNEL
SPD
SYNC
ITXMEM
0x038
ITXMF
ITXMAF
ITXM5AV
ITXME
<RESERVED>
0x03C
IRXPKCTL
0x040
AP_CLK
RMVUAP
SPAV
EN_IRX
BPAD
EN_FS
RST_IRX
IRXHQ1
0x044
E0
F0
SID
DBS
FN
QPC
SPH
IRXHQ2
0x048
E1
F1
FMT
FDF
SYT
IRXINTACK
0x04C
IRXFULL
IRXEMPTY
FSYNC
SEQERR
CRCERR
CIPTAGFLT
RCVBP
SQOV
IRXINTE
0x050
EIRXFULL
EIRXEMPTY
EFSYNC
ESEQERR
ECRCERR
ECIPTAGFLT
ERCVBP
ESQOV
IRXCTL
0x054
SPD
TAG
CHANNEL
ERR
SYNC
IRXMEM
0x058
IRXMF
IRXMAF
IRXM5AV
IRXME
<RESERVED>
0x05C
<RESERVED>
0x07C
SV00800


REGISTER ADDRESS
31
24
23
16
15
8
7
0
ASYCTL
0x080
BCAST EN
ARXRST
ATXRST
ARXALL
MAXRC
TOS
TOF
ASYMEM
0x084
TRSPQIDLE
TREQQIDLE
RRSPQF
RRSPQAF
RRSPQ5AV
RRSPQE
RREQQF
RREQQAF
RREQQ5AV
RREQQE
TRSPQF
TRSPQAF
TRSPQ5AV
TRSPQE
TREQQF
TREQQAF
TREQQ5AV
TREQQE
TX_RQ_NEXT
0x088
FIRST/MIDDLE QUADLET OF PACKET FOR TRANSMITTER REQUEST QUEUE (WRITE ONLY)
TX_RQ_LAST
0x08C
LAST QUADLET OF PACKET FOR TRANSMITTER REQUEST QUEUE (WRITE ONLY)
TX_RP_NEXT
0x090
FIRST/MIDDLE QUADLET OF PACKET FOR TRANSMITTER RESPONSE QUEUE (WRITE ONLY)
TX_RP_LAST
0x094
LAST QUADLET OF PACKET FOR TRANSMITTER RESPONSE QUEUE (WRITE ONLY)
RREQ
0x098
QUADLET OF PACKET FROM RECEIVER REQUEST QUEUE (TRANSFER REGISTER)
RRSP
0x09C
QUADLET OF PACKET FROM RECEIVER RESPONSE QUEUE (TRANSFER REGISTER)
ASYINTACK
0x0A0
RRSPQFULL
RREQQFULL
SIDQAV
RRSPQLASTQ
RREQQLASTQ
RRSPQRDERR
RREQQRDERR
RRSPQQAV
RREQQQAV
TIMEOUT
RCVDRSP
TRSPQFULL
TREQQFULL
TRSPQWRERR
TREQQWRERR
TRSPQWR
TREQQWR
ASYINTE
0x0A4
ERRSPQFULL
ERREQQFULL
ESIDQAV
ERRSPQLASTQ
ERREQQLASTQ
ERRSPQRDERR
ERREQQRDERR
ERRSPQQAV
ERREQQQAV
ETIMEOUT
ERCVDRSP
ETRSPQFULL
ETREQQFULL
ETRSPQWRERR
ETREQQWRERR
ETRSPQWR
ETREQQWR
<RESERVED>
0x0A8
<RESERVED>
0x0F8
SV00801

13.1 Link Control Registers

13.1.1 ID Register (IDREG) – Base Address: 0x000

The ID register is automatically updated by the attached PHY with the proper Node ID after completion of the bus reset.



Reset Value 0xFFFF0002

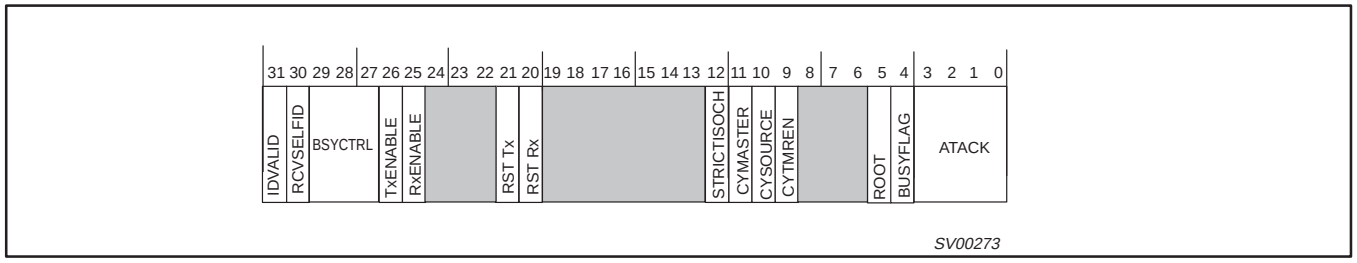
Bit 31..22: R/W Bus ID:The 10-bit bus number that is used with the Node ID in the source address for outgoing packets and used to accept or reject incoming packets. This field reverts to all ‘1’s (0x3FF) upon bus reset.

Bit 21..16: R/W Node ID: Used in conjunction with Bus ID in the source address for outgoing packets and used to accept or reject incoming packets. This register auto-updates with the node ID assigned after the 1394 bus Tree-ID sequence.

Bit 15..0: R Version_Code: Version of the PDI1394L11

13.1.2 General Link Control (LNKCTL) – Base Address: 0x004

The General Link control register is used to program the Link Layer isochronous transceiver, as well as the overall link transceiver. It also provides general link status.



Reset Value 0x46000000

Bit 31: R/W ID Valid: When equal to one, the PDI1394L11 accepts the packets addressed to this node. This bit is automatically set after selfID complete and node ID is updated.

Bit 30: R/W Receive Self ID: When asserted, the self-identification packets, generated by each PHY device on the bus, during bus initialization are received and placed into the asynchronous request queue as a single packet.

Bit 29..27: R/W Busy Control: These bits control what busy status the chip returns to incoming packets. The field is defined below:

- 000 = use protocol requested by received packet (either dual phase or single phase)
- 001 = send busy A when it is necessary to send a busy acknowledge (testing/diagnostics)
- 010 = send a busy B when it is necessary to send a busy acknowledge (testing/diagnostics)
- 011 = use single phase retry protocol
- 100 = use protocol requested in packet, always send a busy ack (for all packets)
- 101 = busy A all incoming packets
- 110 = busy B all incoming packets are ‘1’
- 111 = use single phase retry protocol, always send a busy ack

Bit 26: R/W Transmitter Enable: When this bit is set, the link layer transmitter will arbitrate and send packets.

Bit 25: R/W Receiver Enable: When this bit is set, the link layer receiver will receive and respond to bus packets.

Bit 21: R/W Reset Transmitter: When set to one, this synchronously resets the transmitter within the link layer.

Bit 20: R/W Reset Receiver: When set to one, this synchronously resets the receiver within the link layer.

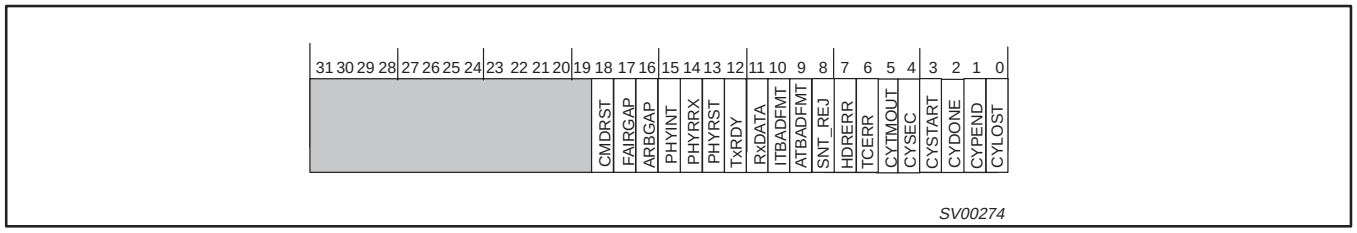
Bit 12: R/W Strict Isochronous: Used to accept or reject isochronous packets sent outside of specified isochronous cycles (between a Cycle Start and subaction gap). A ‘1’ rejects packets sent outside the specified cycles, a “0” accepts isochronous packets sent outside the specified cycle.

Bit 11: R/W Cycle Master: When asserted and the PDI1394L11 is attached to the root PHY (ROOT bit = 1), and the cycle_count field of the cycle timer register increments, the transmitter sends a cycle-start packet. Cycle Master function will be disabled if a cycle timeout is detected (CYTMOUT bit 5 in LNKPHYINTACK). To restart the Cycle Master function in such a case, first reset CYMASTER, then set it again.

Bit 10:	R/W	Cycle Source: When asserted, the cycle_count field increments and the cycle_offset field resets for each positive transition of CYCLEIN. When deasserted, the cycle count field increments when the cycle_offset field rolls over.
Bit 9:	R/W	Cycle Timer Enable: When asserted, the cycle offset field increments.
Bit 5:	R	Root: Indicates this device is the root on the bus. This automatically updates after the self_ID phase.
Bit 4:	R	Busy Flag: The type of busy acknowledge which will be sent next time an acknowledge is required. 0 = Busy A, 1 = Busy B (only meaningful during a dual-phase busy/retry operation).
Bit 3..0:	R	AT acknowledge received: The last acknowledge received by the transmitter in response to a packet sent from the transmit-FIFO interface while the ATF is selected (diagnostic purposes).

13.1.3 Link /Phy Interrupt Acknowledge (LNKPHYINTACK) – Base Address: 0x008

The Link/Phy Interrupt Acknowledge register indicates various status and error conditions in the Link and Phy which can be programmed to generate an interrupt. The interrupt enable register (LNKPHYINTE) is a mirror of this register. Acknowledgment of an interrupt is accomplished by writing a '1' to a bit in this register that is set. This action reset the bit indication to a '0'. Writing a '1' to a bit that is already "0" will have no effect on the register.



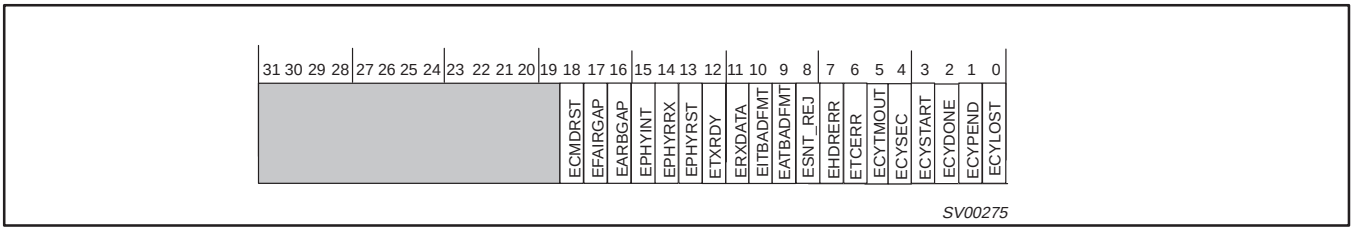
Reset Value 0x00000000

Bit 18:	R/W	Command Reset Received: A write request to RESET-START has been received.
Bit 17:	R/W	Fair Gap: The serial bus has been idle for a fair-gap time (called subaction gap in the IEEE 1394 specification).
Bit 16:	R/W	Arbitration Reset Gap: The serial bus has been idle for an arbitration reset gap.
Bit 15:	R/W	Phy Chip Int: The Phy chip has signaled an interrupt through the Phy interface.
Bit 14:	R/W	Phy Register Information Received: A register has been transferred by the Physical Layer device into the Link.
Bit 13:	R/W	Phy Reset Started: A Phy-layer reconfiguration has started. This interrupt clears the ID valid bit. (Called Bus Reset in the IEEE 1394 specification).
Bit 12:	R/W	Transmitter Ready: The transmitter is idle and ready.
Bit 11:	R/W	Receiver has Data: The receiver has confirmed data to the receiver response/request FIFO. Used for diagnostic purposes only.
Bit 10:	R/W	Isochronous Transmitter is Stuck: The transmitter has detected invalid data at the transmit-FIFO interface when the ITF is selected.
Bit 9:	R/W	Asynchronous Transmitter is Stuck: The transmitter expected start of new async packet in queue, but found other data (out of sync with user). Reset to clear.
Bit 8:	R/W	Busy Acknowledge Sent by Receiver: The receiver was forced to send a busy acknowledge to a packet addressed to this node because the receiver response/request FIFO overflowed.
Bit 7:	R/W	Header Error: The receiver detected a header CRC error on an incoming packet that may have been addressed to this node.
Bit 6:	R/W	Transaction Code Error: The transmitter detected an invalid transaction code in the data at the transmit FIFO interface.
Bit 5:	R/W	Cycle Timed Out. ISOCH cycle lasted more than 125µs from Cycle-Start to Fair Gap: Disables cycle master function
Bit 4:	R/W	Cycle Second incremented: The cycle second field in the cycle-timer register incremented. This occurs approximately every second when the cycle timer is enabled.
Bit 3:	R/W	Cycle Started: The transmitter has sent or the receiver has received a cycle start packet.
Bit 2:	R/W	Cycle Done: A fair gap has been detected on the bus after the transmission or reception of a cycle start packet. This indicates that the isochronous cycle is over; Note: Writing a value of '0' to the bit has no effect.
Bit 1:	R/W	Cycle Pending: Cycle pending is asserted when cycle timer offset is set to zero (rolled over or reset) and stays asserted until the isochronous cycle has ended.
Bit 0:	R/W	Cycle Lost: The cycle timer has rolled over twice without the reception of a cycle start packet. This only occurs when cycle master is not asserted.

13.1.4 Link / Phy Interrupt Enable (LNKPHYINTE) – Base Address: 0x00C

This register is a mirror of the Link/Phy Interrupt Acknowledge (LNKPHYINTACK) register. Enabling an interrupt is accomplished by writing a '1' to the bit corresponding to the interrupt desired.

This register enables the interrupts described in the Link /Phy Interrupt Acknowledge register (LNKPHYINTACK) description. A one in any of the bits enables that function to create an interrupt. A zero disables the interrupt, however the status is readable in the Link /Phy Interrupt Acknowledge register.

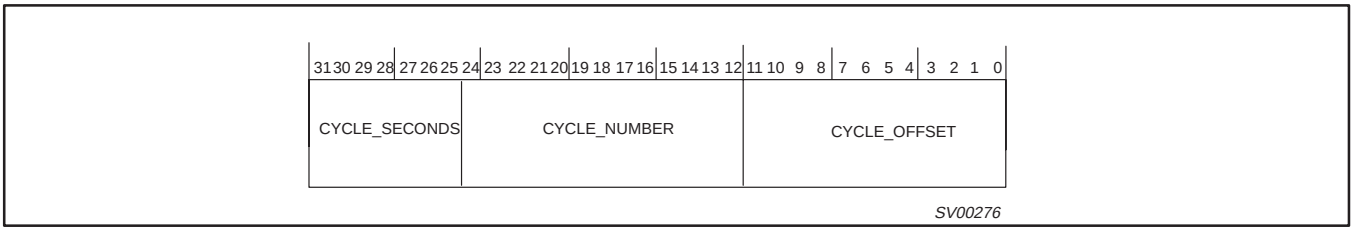


Reset Value 0x00000000

Bits 18..0 are interrupt enable bits for the Link/Phy Interrupt Acknowledge (LNKPHYINTACK).

13.1.5 Cycle Timer Register (CYCTM) – Base Address: 0x010

Cycle Timer Register operation is controlled by the Cycle Timer Enable (CYTMREN) bit in the General Control Register.

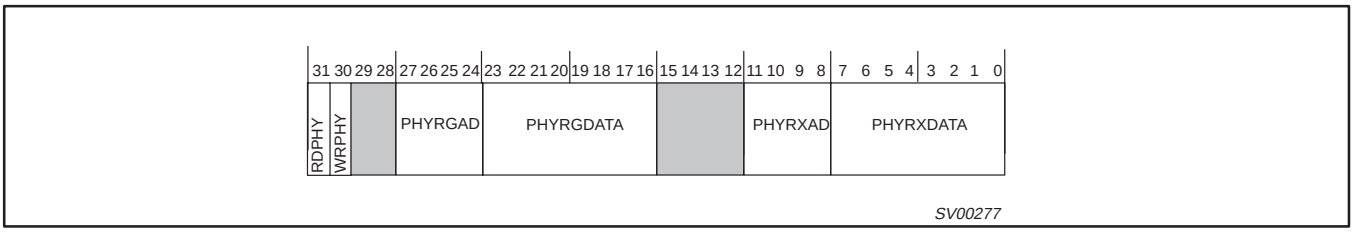


Reset Value 0x00000000

Bit 31..25:	R/W	Seconds count: 1-Hz cycle timer counter.
Bit 24..12:	R/W	Cycle Number: 8kHz cycle timer counter.
Bit 11..0:	R/W	Cycle Offset: 24.576MHz cycle timer counter.

13.1.6 Phy Register Access (PHYACS) – Base Address: 0x014

This register provides access to the internal registers on the Phy. There are special considerations when reading or writing to this register. When reading a PHY register, the address of the register is written to the PHYRGAD field with the RDPHY bit set. The PHY data will be valid when the PHYRRX bit (LNKPHYINTACK register bit 14) is set. Once this happens the register data is available in the PHYRXDATA, the address of the register just read is also available in the PHYRXAD fields. When writing a Phy register, the address of the register to be written is set in the PHYRGAD field and the data to be written to the register is set in PHYRGDATA, along with the WRPHY bit being set. Once the write is complete, the WRPHY bit will be cleared. Do not write a new Read/Write command until the previous one has been completed. After the Self-ID phase, PHY register 0 will be read automatically.



Reset Value 0x00000000

Bit 31:	R/W	Read Phy Chip Register: When asserted, the PDI1394L11 sends a read register request with address equal to Phy Rg Ad to the Phy interface. This bit is cleared when the request is sent.
Bit 30:	R/W	Write Phy Chip Register: When asserted, the PDI1394L11 sends a write register request with address equal to Phy Rg Ad to the Phy interface. This bit is cleared when the request is sent.
Bit 27..24:	R/W	Phy Chip Register Address: This is the address of the Phy-chip register that is to be accessed.
Bit 23..16:	R/W	Phy Chip Register Data: This is the data to be written to the Phy-chip register indicated in Phy Rg Ad.
Bit 11..8:	R	Phy Chip Register Received Address: Address of register from which Phy Rx Data came.
Bit 7..0:	R	Phy Chip Register Received Data: Data from register addressed by Phy Rx Ad.

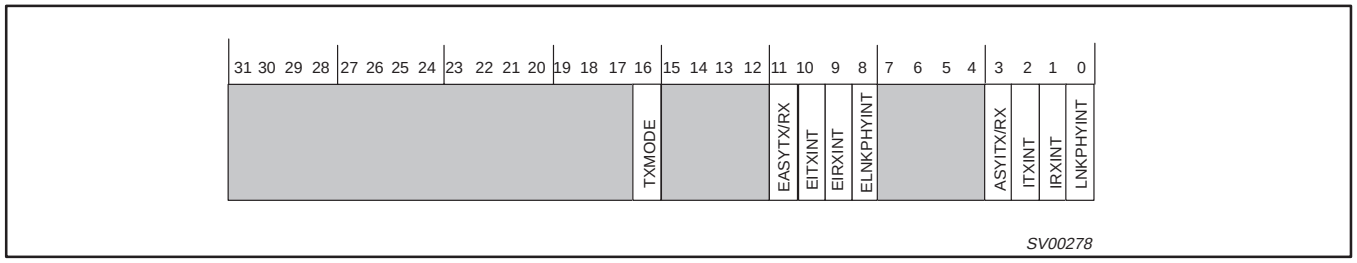
13.1.7 Global Interrupt Status and TX Control (GLOBCSR) – Base Address: 0x018

This register is the top level interrupt status register. If the external interrupt line is set, this register will indicate which major portion of the AV Link generated the interrupt. There is no interrupt acknowledge required at this level. These bits auto clear when the interrupts in the appropriate section of the device are cleared or disabled. Control of the AV transceiver is also provided by this register.

Bits 0 to 3 are used to identify which internal modules are currently generating an interrupt. After identifying the module, the appropriate register in that module must be read to determine the exact cause of the interrupt.

NOTES

1. There can be more than one interrupt source active at the same time.
2. The HIF INT_N signal (pin 28) remains active as long as there is at least one more enabled active interrupt status bit.



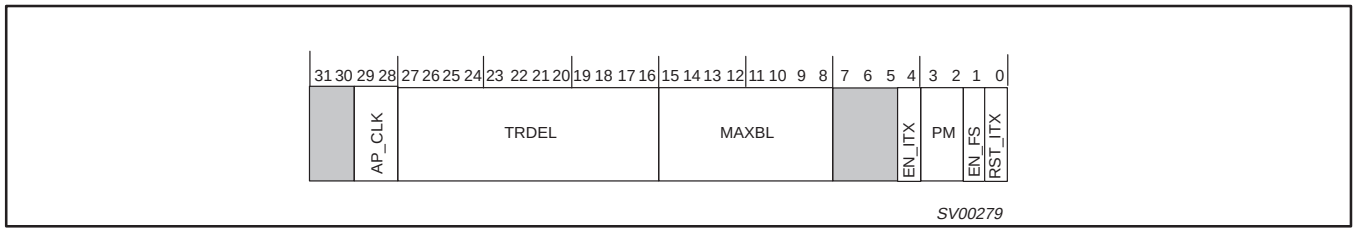
Reset Value 0x00010000

Bit 16:	R/W	Transmit Mode: Control bit, a ‘1’ enables the AV transmitter. A ‘0’ enables the AV receiver. The register defaults to ‘1’ on reset. Also this bit directly controls the direction of the bi-directional pins of the AV interface.
Bit 11:	R/W	Enables generation of external interrupt by asynchronous transmitter and receiver module (ASYTX/RX, bit 3) when set (1). Disables such interrupts when clear (0) (regardless of ASYINTE contents).
Bit 10:	R/W	Enables generation of external interrupt by the isochronous transmitter module (ITXINT, bit 2) when set (1). Disables such interrupts when clear (0) (regardless of ITXINTE contents).
Bit 9:	R/W	Enables generation of external interrupt by the isochronous receiver module (IRXINT, bit 1) when set (1). Disables such interrupts when clear (0) (regardless of IRXINTE contents).
Bit 8:	R/W	Enables generation of external interrupt by general link/phy module (LKPHYINT, bit 0) when set (1). Disables such interrupts when clear (0) (regardless of LNKPHYINTE contents).
Bit 3:	R	Asynchronous Transmitter/ Receiver Interrupt: Interrupt source is in the Asynchronous Transmitter/ Receiver Interrupt Acknowledge/Source register.
Bit 2:	R	AV Transmitter Interrupt: Interrupt source is in the AV Transmitter Interrupt Acknowledge/Source register.
Bit 1:	R	AV Receiver Interrupt: Interrupt source is in the AV Receiver Interrupt Acknowledge/Source register.
Bit 0:	R	Link-Phy Interrupt: Interrupt source is in the Link Phy Interrupt Acknowledge register.

13.2 AV (Isochronous) Transmitter and Receiver Registers

13.2.1 Isochronous Transmit Packing Control and Status (ITXPKCTL) – Base Address: 0x020

This register allows the user to set up the appropriate AV packets from data entered into the AV interface. The packing and control parameters (TRDEL, MAXBL, DBS, FN, QPC, and SPH) should never be changed while the transmitter is operating. The only exception to this is the MAXBL parameter when in MPEG-2 packing mode.



Reset Value 0x00000001

Bit 27..16:	R/W	TRDEL: Transport delay. Value added to cycle timer to produce time stamps. Lower 4 bits add to upper 4 bits of cycle_offset, (Cycle Timer Register, CYCTM). Remainder adds to cycle_count field.
Bit 15..8:	R/W	MAXBL: The (maximum) number of data blocks to be put in a payload.
Bit 4:	R/W	EN_ITX: Enable receipt of new application packets and generation of isochronous bus packets in every cycle. This bit also enables the Link Layer to arbitrate for the transmitter in each subsequent bus cycle. When this bit is disabled (0), the current packet will be transmitted and then the transmitter will shut down.

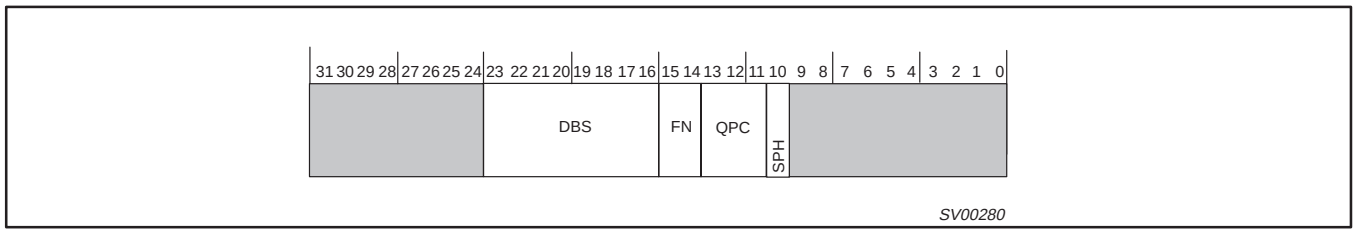
Bit 3..2: R/W PM: packing mode:
00 = variable sized bus packets, most generic mode.
01 = fixed size bus packets.
10 = MPEG–2 packing mode.
11 = No data, just CIP headers are transmitted.

Bit 1: R/W EN_FS:enable generation/insertion of SYT stamps (Time Stamps) in CIP header.

Bit 0: R/W Reset Isochronous Transmitter: causes transmitter to be reset when ‘1’. In order for synchronous reset of ITX to work properly, the application must supply an AVCLK and ensure that the reset bit is kept (programmed) HIGH for at least the duration of one AVCLK period. Failure to do so may cause the application interface of this module to be improperly reset (or not reset at all).

13.2.2 Common Isochronous Transmit Packet Header Quadlet 1 (ITXHQ1) – Base Address: 0x024

The AV Transmit Packing Control register holds the specification for the packing scheme used on the AV data stream. This information is included in Common Isochronous Packet (CIP) header quadlet 1.



Reset Value 0x00000000

Bit 10: R/W SPH: Indicates that a 25-bit CYCTM based time stamp has to be inserted before each application packet.

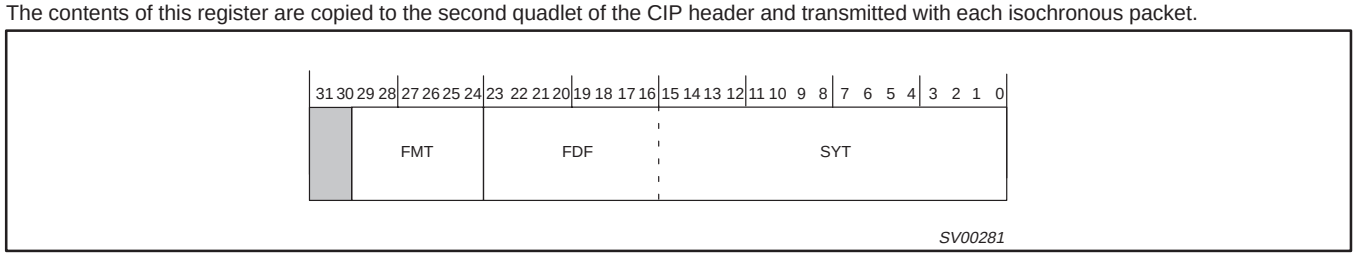
Bit 11..13: R/W QPC: Number of dummy quadlets to append to each source packet before it is divided into data blocks of the specified size. The value QPC must be less than DBS and less than 2^{FN}.

Bit 14..15: R/W FN: (Fraction Number) The encoding for the number of data blocks into which each source packet shall be divided (00 = 1, 01 = 2, 10 = 4, 11 = 8).

Bit 16..23: R/W DBS: Size of the data blocks from which AV payload is constructed. The value 0 represents a length of 256 quadlets.

13.2.3 Common Isochronous Transmit Packet Header Quadlet 2 (ITXHQ2) – Base Address: 0x028

The contents of this register are copied to the second quadlet of the CIP header and transmitted with each isochronous packet.



Reset Value 0x00000000

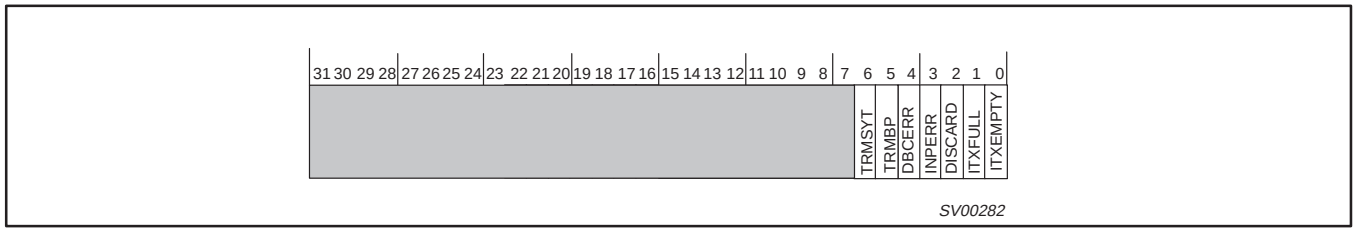
Bit 29..24: R/W FMT: Value to be inserted in the FMT field in the AV header.

Bit 23..0: R/W FDF/SYT: Value to be inserted in the FDF field. When the EN_FS bit in the Transmit Control and Status Register (ITXPKCTL) is set (=1), the lower 16 bits of this register are replaced by an SYT stamp if a rising edge on AVFSYNCIN has been detected or all ‘1’s if no such edge was detected since the previous packet. The upper 8 bits of the register are sent as they appear in the FDF register. When the EN_FS bit in the Transmit Control and Status Register is unset (=0), the full 24 bits can be set to any application specified value.

13.2.4 Isochronous Transmitter Interrupt Acknowledge (ITXINTACK) – Base Address: 0x02C

The AV Transmitter Interrupt Control and Status register is the interrupt register for the AV transmitter.

Bits 2, 3, and 4 ”auto repair” themselves, i.e. AVLINK will detect the situation and attempt to recover on its own. The host controller still needs to clear these interrupts to be alerted the next time.



Reset Value 0x00000000

Bits 6 .. 0 are interrupt acknowledge bits; and are defined as:

Bit 6: R/W TRMSYT: Interrupt on transmission of a SYT in CIP header quadlet 2

Bit 5: R/W TRMBP: Interrupt on payload transmission/discard complete.

Bit 4: R/W DBCERR: Acknowledge interrupt on Data Block Count (DBC) synchronization loss.

Bit 3: R/W INPERR: Acknowledge interrupt on input error (input data discarded).

Bit 2: R/W DISCARD: Interrupt on lost cycle (payload discarded).

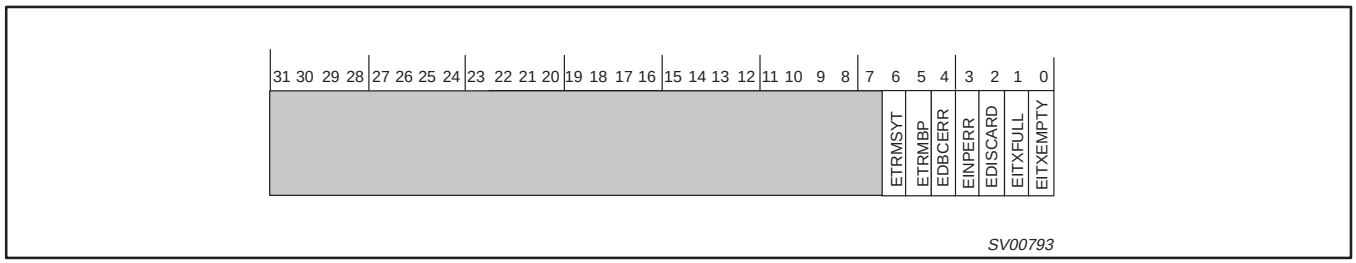
Bit 1: R/W ITXFULL: Interrupt on isochronous memory bank full. This is a fatal error, the recommended action is to reset and re-initialize the transmitter.

Bit 0: R/W ITXEMPTY: Interrupt on isochronous memory bank empty.

Other bits will always read ‘0’.

13.2.5 Isochronous Transmitter Interrupt Enable (ITXINTE) – Base Address: 0x030

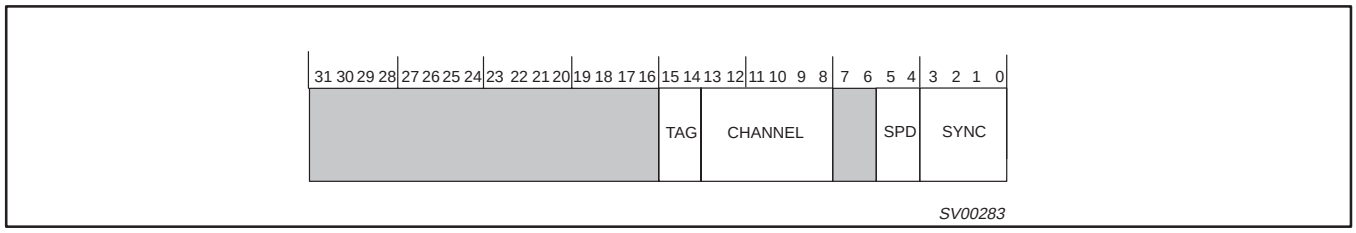
These are the enabled bits for the AV Transmitter Control.



Reset Value 0x00000000

Bits 6..0 are interrupt enable bits for the Isochronous Transmitter Interrupt Acknowledge register (ITXINTACK).

13.2.6 Isochronous Transmitter Control Register (ITXCTL) – Base Address: 0x34



Reset Value 0x00000000

Bit 15..14: R/W Tag: Tag code to insert in isochronous bus packet header. Should be ‘01’ for IEC 61883 International Standard data.

Bit 13..8: R/W Channel: Isochronous channel number.

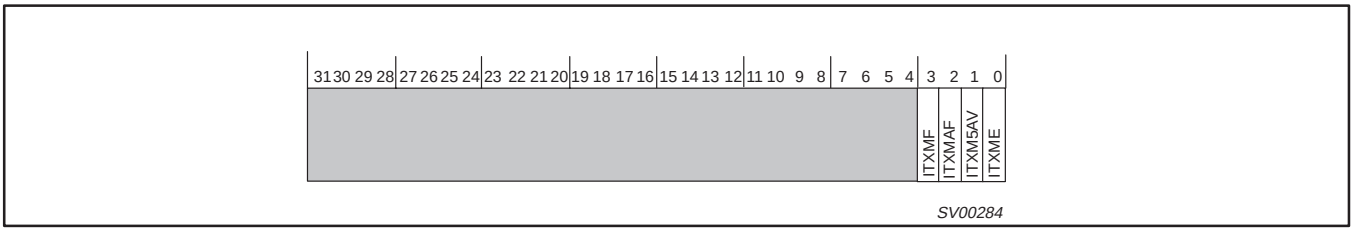
Bit 5..4: R/W Speed: Cable transmission speed (S100, S200, S400).

00=100Mbs
01=200Mbs
10=400Mbs
11=reserved

Bit 3..0 R/W Sync: Code to insert in SY field of isochronous bus packet header.

13.2.7 Isochronous Transmitter Memory Status (ITXMEM) – Base Address: 0x038

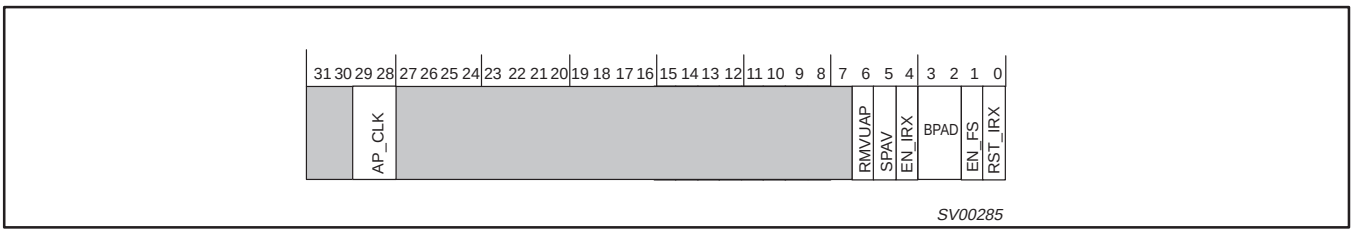
The AV Transmitter Memory Status register reports on the condition of the internal memory buffer used to store incoming AV data streams before transmission over the 1394 bus.



Reset Value 0x00000003

Bit 3:	R	ITXMF: memory is completely full, no storage available.
Bit 2:	R	ITXMAF: almost full, exactly one quadlet of storage available.
Bit 1:	R	ITXM5AV: at least 5 more quadlets of storage available.
Bit 0:	R	ITXME: memory bank is empty (zero quadlets stored).

13.2.8 Isochronous Receiver Unpacking Control (IRXPKCTL) – Base Address: 0x040



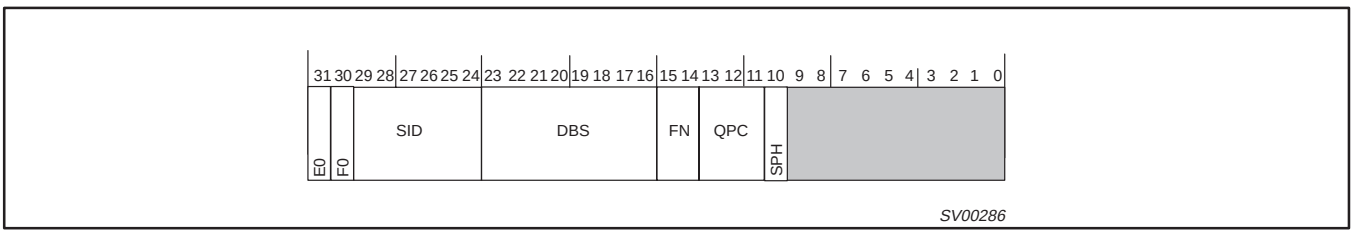
Reset Value 0x00000041

AV Receiver Control Bits.

Bit 6:	R/W	RMVUAP: Remove unreliable packets from memory, do not attempt delivery
Bit 5:	R	SPAV: Source packet available for delivery in buffer memory.
Bit 4:	R/W	EN_IRX: Enable receiver operation. Value is only checked whenever a new bus packet arrives, so enable/disable while running is ‘graceful’.
Bit 2..3:	R/W	BPAD: Value indicating the amount of byte padding to be removed from the last data quadlet of each source packet, from 0 to 3 bytes. This is in addition to quadlet padding as defined in IEC 61883 International Standard.
Bit 1:	R/W	EN_FS: Enable processing of SYT stamps.
Bit 0:	R/W	RST_IRX: causes the receiver to be reset when ‘1’. In order for synchronous reset of IRX to work properly, the application must supply an AVCLK and ensure that the reset bit is kept (programmed) HIGH for at least the duration of one AVCLK period. Failure to do so may cause the application interface of this module to be improperly reset (or not reset at all).

13.2.9 Common Isochronous Receiver Packet Header Quadlet 1 (IRXHQ1) – Base Address: 0x044

This quadlet represents the last received header value when AV receiver is operating.

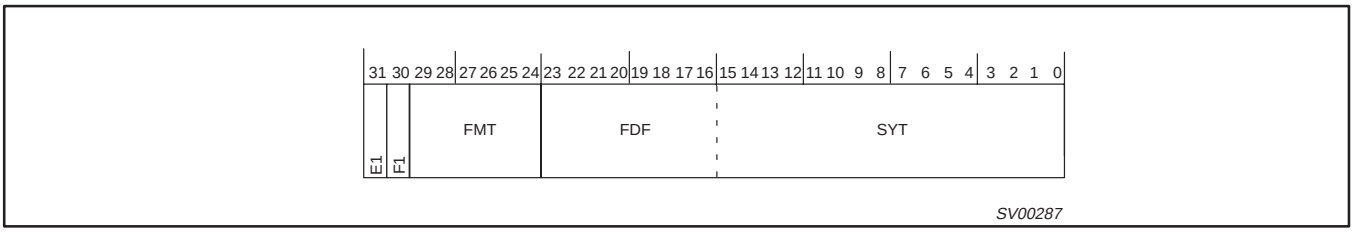


Reset Value 0x00000000

Bit 31..30:	R	E0: End of Header, F0: Format: Always set to 00 for first AV header quadlet
Bit 29..24	R	SID: Source ID, contains the node address of the sender of the isochronous data.
Bit 23.16:	R	DBS: Size of the data blocks from which AV payload is constructed. The value 0 represents a length of 256 quadlets.
Bit 15..14:	R	FN (Fraction Number): The encoding for the number of data blocks into which each source packet has been divided (00 = 1, 01 = 2, 10 = 4, 11 = 8) by the transmitter of the packet.
Bit 13..11:	R	QPC: Number of dummy quadlets appended to each source packet before it was divided into data blocks of the specified size.

Bit 10: R SPH: Indicates that a CYCTM based time stamp is inserted before each application packet (25 bits specified in the IEC 61883 International Standard).

13.2.10 Common Isochronous Receiver Packet Header Quadlet 2 (IRXHQ2) – Base Address: 0x048



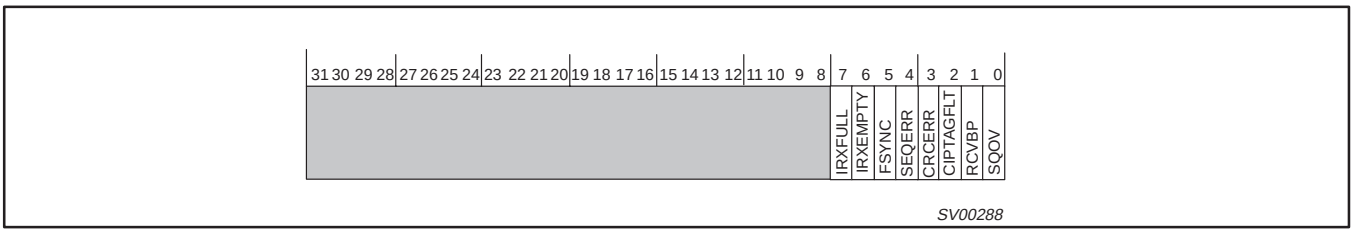
Reset Value 0x0000FFFF

Bit 31..30: R E1: End of Header, F1: Format: Should be set to 10 for second AV header quadlet.

Bit 29..24: R FMT: Value inserted in the Format field.

Bit 23..0: R FDF/SYT: If “EN FS” in Register IRXPKCTL (0x040) is set to ‘1’, then lower 16-bits are interpreted as SYT.

13.2.11 Isochronous Receiver Interrupt Acknowledge (IRXINTACK) – Base Address: 0x04C



Reset Value 0x00000000

Bit 7: R/W IRXFULL: Isochronous data memory bank has become full. This is a fatal error, the recommended action is to reset and re-initialize the transmitter.

Bit 6: R/W IRXEMPTY: Isochronous data memory bank has become empty.

Bit 5: R/W FSYNC: Pulse at fsync output.

Bit 4: R/W SEQERR: Sequence error of data blocks.

Bit 3: R/W CRCERR: CRC error in bus packet.

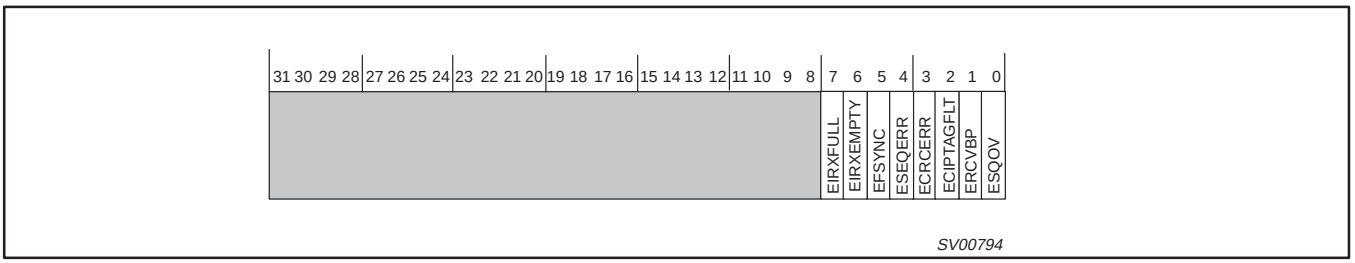
Bit 2: R/W CIPTAGFLT: Faulty CIP header tag (E,F bits). i.e.: The CIP header did not meet the standard and the whole packet is ignored.

Bit 1: R/W RCVBP: Bus packet processing complete.

Bit 0: R/W SQOV: Status queue overflow. This is a fatal error, the recommended action is to reset and re-initialize the transmitter.

13.2.12 Isochronous Receiver Interrupt Enable (IRXINTE) – Base Address: 0x050

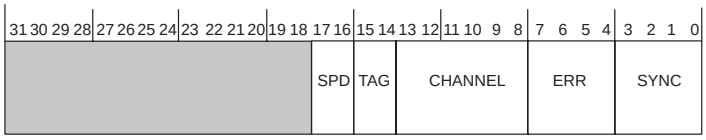
Interrupt enable bits for AV Receiver.



Reset Value 0x00000000

Bit 7..0 are interrupt enable bits for the Isochronous Receiver Interrupt Acknowledge (IRXINTACK).

13.2.13 Isochronous Receiver Control Register (IRXCTL) – Base Address: 0x054



Reset Value 0x00000000

Bit 17..16: R SPD: Speed of last received isochronous packet (S100 .. S400).
00 = 100 Mbps
01 = 200 Mbps
10 = 400 Mbps
11 = Reserved

Bit 15..14: R/W TAG: Isochronous tag value (must match) for AV format, ‘01’ for IEC 61883 International Standard data.

Bit 13..8: R/W CHAN: Channel number to receive isochronous data.

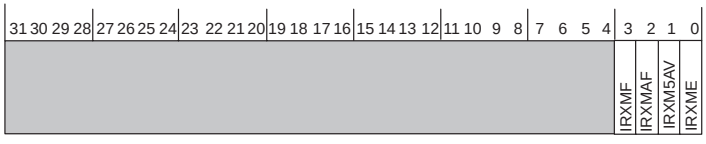
Bit 7..4: R ERR: Error code for last received isochronous AV packet.

Table 9. Error Codes

Code	Name	Meaning
0000	reserved	
0001	ack_complete	The node has successfully accepted the packet. If the packet was a request subaction, the destination node has successfully completed the transaction and no response subaction shall follow.
0010 through 1100	reserved	
1101	ack_data_error	The node could not accept the block packet because the data field failed the CRC check, or because the length of the data block payload did not match the length contained in the dataLength field. this code shall not be returned for any packet that does not have a data block payload.
1110 and 1111	reserved	

Bit 3..0: R SYNC: Last received SY code in isochronous bus packet header.

13.2.14 Isochronous Receiver Memory Status (IRXMEM) – Base Address: 0x058



Reset Value 0x00000003

Bit 3: R IRXMF: Full: no space available.

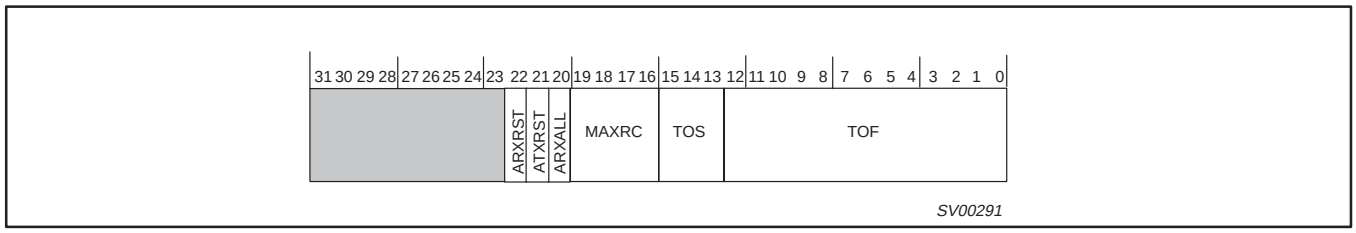
Bit 2: R IRXMAF: Almost full: exactly one quadlet of storage available.

Bit 1: R IRXM5AV: At least 5 more quadlets of storage available.

Bit 0: R RXME: Memory bank is empty (no data committed).

13.3 Asynchronous Control and Status Interface

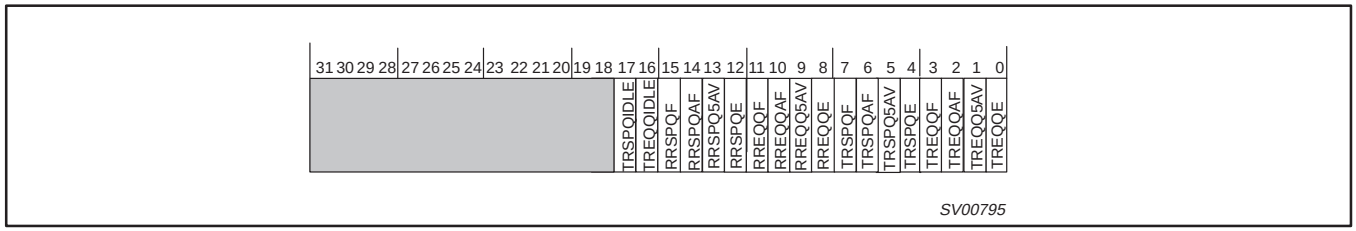
13.3.1 Asynchronous RX/TX Control (ASYCTL) – Base Address: 0x080



Reset Value 0x00300320

Bit 22:	R/W	ARXRST: Asynchronous receiver reset. This bit will auto clear when the link layer state machine is idle.
Bit 21:	R/W	ATXRST: Asynchronous transmitter reset
Bit 20:	R/W	ARXALL: Receive and filter only RESPONSE packets. When set (1), all responses are stored. When clear (0), only solicited responses are stored.
Bit 19..16:	R/W	MAXRC: Maximum number of asynchronous transmitter single phase retries
Bit 15..13:	R/W	TOS: Time out seconds, integer of 1 second
Bit 12..0:	R/W	TOF: Time out fractions, integer of 1/8000 second

13.3.2 Asynchronous RX/TX Memory Status (ASYMEM) – Base Address: 0x084

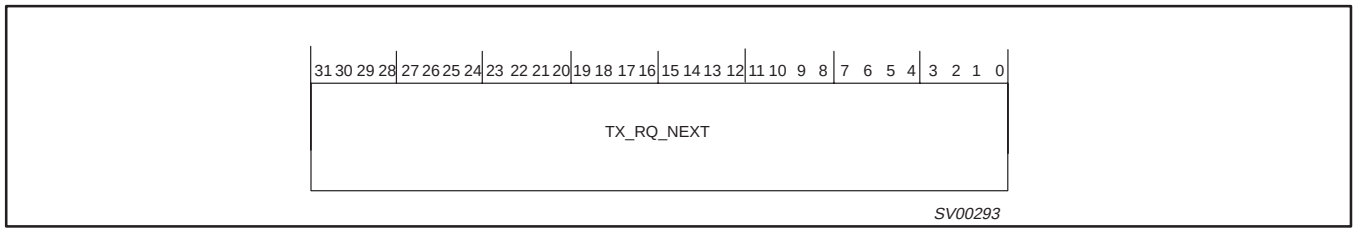


Reset Value 0x00000000

Unused bits read ‘0’. The information in this register is primarily used for diagnostics.

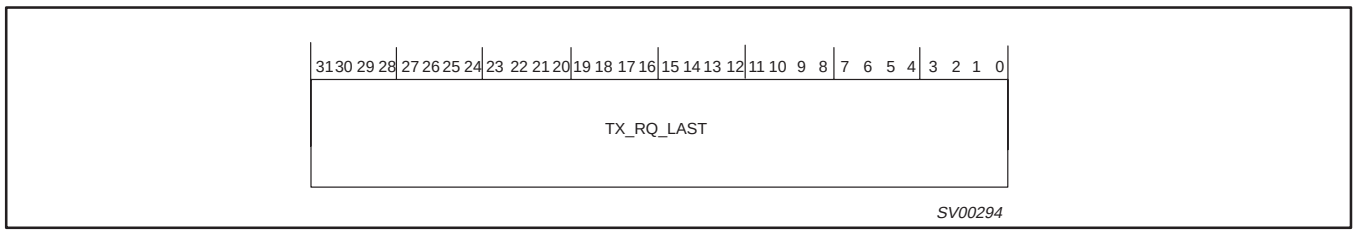
Bit 17:	R	TRSPQIDLE: Transmitter response queue idle. Indicates that the transfer register for this queue is empty.
Bit 16:	R	TREQQIDLE: Transmitter request queue idle. Indicates that the transfer register for this queue is empty.
Bit 15:	R	RRSPQF: Receiver response queue full.
Bit 14:	R	RRSPQAF: Receiver response queue almost full (precisely 1 more quadlet available).
Bit 13:	R	RRSPQ5AV: Receiver response queue at least 5 quadlets available.
Bit 12:	R	RRSPQE: Receiver response queue empty.
Bit 11:	R	RREQQF: Receiver request queue full.
Bit 10:	R	RREQQAF: Receiver request queue almost full (precisely 1 more quadlet available).
Bit 9:	R	RREQQ5AV: Receiver request queue at least 5 quadlets available.
Bit 8:	R	RREQQE: Receiver request queue empty.
Bit 7:	R	TRSPQF: Transmitter response queue full.
Bit 6:	R	TRSPQAF: Transmitter response queue almost full (precisely 1 more quadlet available).
Bit 5:	R	TRSPQ5AV: Transmitter response queue at least 5 quadlets available.
Bit 4:	R	TRSPQE: Transmitter response queue empty.
Bit 3:	R	TREQQF: Transmitter response queue full.
Bit 2:	R	TREQQAF: Transmitter request queue almost full (precisely 1 more quadlet available).
Bit 1:	R	TREQQ5AV: Transmitter response queue at least 5 quadlets available.
Bit 0:	R	TREQQE: Transmitter request queue empty.

13.3.3 Asynchronous Transmit Request Next (TX_RQ_NEXT) – Base Address: 0x088



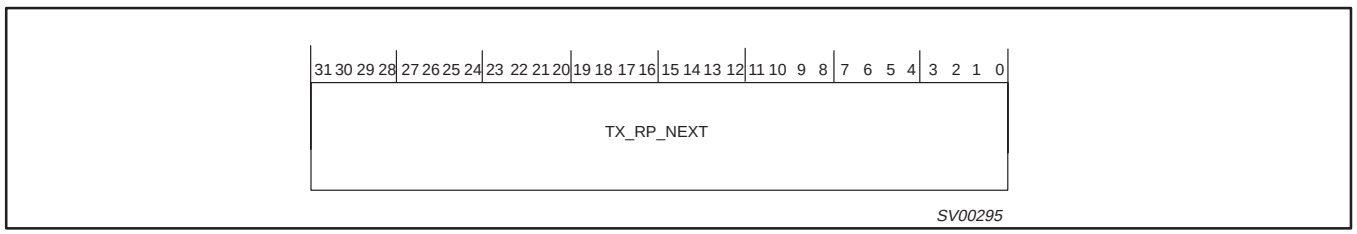
Bit 31..0: W TX_RQ_NEXT: First/middle quadlet of packet for transmitter request queue (write only).
Writing this register will clear the TREQQWR flag until the quadlet has been written to its queue.

13.3.4 Asynchronous Transmit Request Last (TX_RQ_LAST) – Base Address: 0x08C



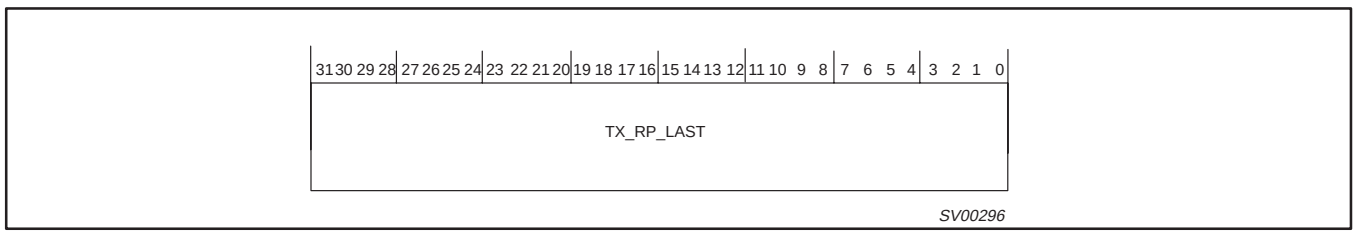
Bit 31..0: W TX_RQ_LAST: Last quadlet of packet for transmitter request queue (write only).
Writing this register will clear the TREQQWR flag until the quadlet has been written to its queue.

13.3.5 Asynchronous Transmit Response Next (TX_RP_NEXT) – Base Address: 0x090



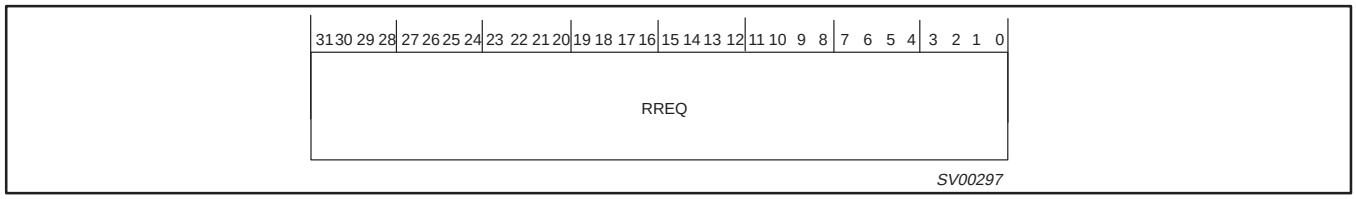
Bit 31..0: W TX_RP_NEXT: First/middle quadlet of packet for transmitter response queue (write only).
Writing this register will clear the TRSPQWR flag until the quadlet has been written to its queue.

13.3.6 Asynchronous Transmit Response Last (TX_RP_LAST) – Base Address: 0x094



Bit 31..0: W TX_RP_LAST: Last quadlet of packet for transmitter response queue (write only).
Writing this register will clear the TRSPQWR flag until the quadlet has been written to its queue.

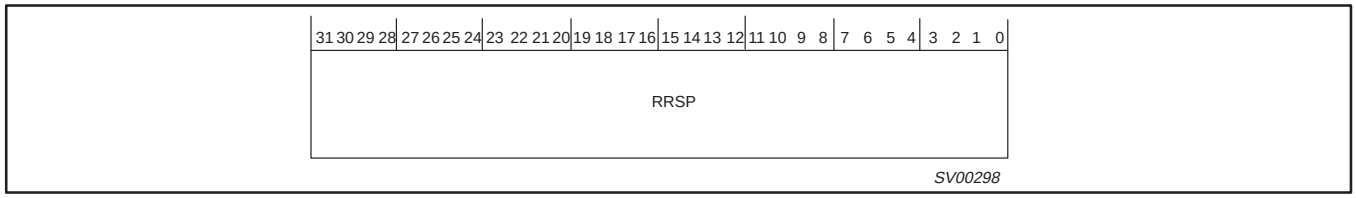
13.3.7 Asynchronous Receive Request (RREQ) – Base Address: 0x098



Reset Value 0x00000000

Bit 31..0: R RREQ:Quadlet of packet from receiver request queue (transfer register).
Reading this register will clear the RREQQQAV flag until the next received quadlet is available for reading.

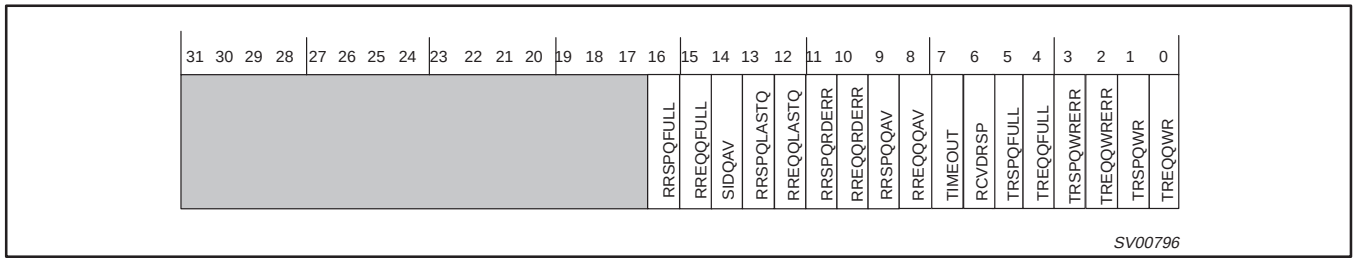
13.3.8 Asynchronous Receive Response (RRSP) – Base Address: 0x09C



Reset Value 0x00000000

Bit 31..0: R RRSP:Quadlet of packet from receiver response queue (transfer register).
Reading this register will clear the RRSPQQAV flag until the next received quadlet is available for reading.

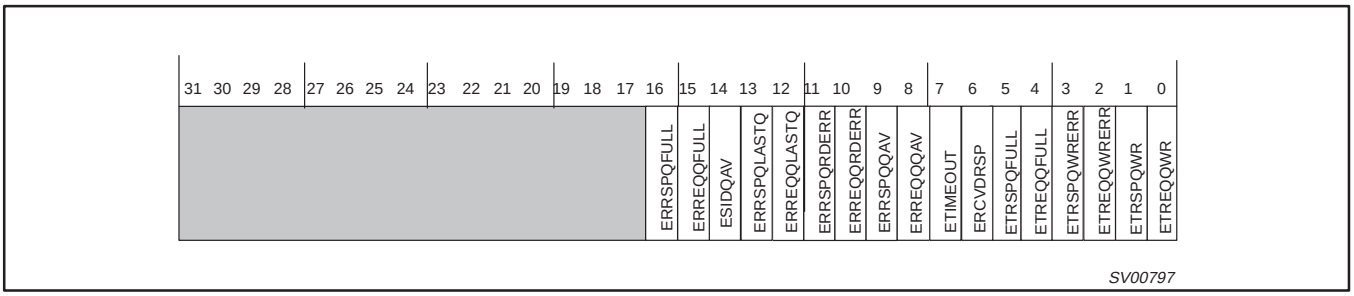
13.3.9 Asynchronous RX/TX Interrupt Acknowledge (ASYINTACK) – Base Address: 0x0A0



Reset Value 0x00000000

Bit 31..17: R/W Unused bits read ‘0’
Bit 16: R/W RRSPQFULL: Receiver response queue did become full.
Bit 15: R/W RREQQFULL: Receiver request queue did become full.
Bit 14: R/W SIDQAV: Current quadlet in RREQ is selfID data.
Bit 13: R/W RRSPQLASTQ: Current quadlet in RRSP is last quadlet of packet.
Bit 12: R/W RREQQLASTQ: Current quadlet in RREQ is last quadlet of packet.
Bit 11: R/W RRSPQRDERR: Receiver response queue read error (transfer error) or bus reset occurred. When set (1), this queue is blocked for read access.
Bit 10: R/W RREQQRDERR: Receiver request queue read error (transfer error) or bus reset occurred. When set (1), this queue is blocked for read access.
Bit 9: R/W RRSPQQAV: Receiver response queue quadlet available (in RRSP).
Bit 8: R/W RREQQQAV: Receiver request queue quadlet available (in RREQ).
Bit 7: R/W TIMEOUT: Split transaction response timeout.
Bit 6: R/W RCVDRSP: Solicited response received (within timeout interval).
Bit 5: R/W TRSPQFULL: Transmitter response queue did become full.
Bit 4: R/W TREQQFULL: Transmitter request queue did become full.
Bit 3: R/W TRSPQWRERR: Transmitter response queue write error (transfer error).
Bit 2: R/W TREQQWRERR: Transmitter request queue write error (transfer error).
Bit 1: R/W TRSPQWR: Transmitter response queue written (transfer register emptied).
Bit 0: R/W TREQQWR: Transmitter request queue written (transfer register emptied).

13.3.10 Asynchronous RX/TX Interrupt Enable (ASYINTE) – Base Address: 0x0A4



Reset Value 0x00000000

Bits16..0 are interrupt enable bits for the Asynchronous RX/TX Interrupt Acknowledge (ASYINTACK).

14.0 DC ELECTRICAL CHARACTERISTICS

Table 10. DC Electrical Characteristics

NAME	DESCRIPTION	MIN	MAX	UNIT	NOTE
V_{IL}	LOW input voltage		0.8	V	Pin categories 1, 2, 3
V_{IH}	HIGH input voltage	2.0		V	Pin categories 1, 2, 3
$V_{IT1}+$	Input threshold, rising edge	$V_{DD}/2 + 0.12$	$V_{DD}/2 + 0.66$	V	Pin categories 6, 8 LOW to HIGH transition
$V_{IT1}-$	Input threshold, falling edge	$V_{DD}/2 – 0.66$	$V_{DD}/2 – 0.12$	V	Pin categories 6, 8 HIGH to LOW transition
V_{OH1}	HIGH output voltage	2.4		V	Pin category 1 I_{OH} = 8mA I_{OL} = 8mA
V_{OL1}	LOW output voltage		0.4	V	Pin category 1 I_{OH} = 8mA I_{OL} = 8mA
V_{OH2}	HIGH output voltage	2.4		V	Pin categories 4, 6, 7 I_{OH} = 4mA
V_{OL2}	LOW output voltage		0.4	V	Pin categories 4, 5, 6, 7 I_{OL} = 4mA
I_L	Input leakage current		±1	μA	Pin categories 1, 2, 3 V_I = 5.5V or 0V
			100	μA	Pin category 8 V_I = 5.5V or 0V
I_{OZ}	3-State output current		±5	μA	Pin categories 1, 7 V_I = 5.5V or 0V
			200	μA	Pin category 6 V_I = 5.5V or 0V
I_{DD}	Active supply current		150	mA	Under idle conditions, the average value is 20 mA

14.1 Pin Categories

Table 11. Pin Categories

Category 1: Input/Output	Category 2: Input	Category 3: Input	Category 4: Output	Category 5: Output	Category 6: Input/Output	Category 7:	Category 8:
HIF D[7:0]	HIF A[8:0]	RESET_N	CYCLEOUT	HIF INT_N	PHY D[0:7]	LREQ	SCLK
AVSYNC	HIF CS_N	CYCLEIN	AVERR0		PHY CTL[0:1]		
AVVALID	HIF WR_N	AVCLK	AVERR1				
AV D[7:0]	HIF RD_N	ISO_N	CLK25				
	AVFSYNCIN		AVFSYNCOUT				
	AVENDPCK						

15.0 AC CHARACTERISTICS

GND = 0V, C_L = 50pF

SYMBOL	PARAMETER	TEST CONDITIONS	WAVEFORMS	LIMITS			UNIT
				T_{amb} = 0°C to +70°C			
				MIN	TYP	MAX	
t_{PERIOD}	AV clock period		Figure 23	41.67			ns
t_{SU}	AV clock setup time		Figure 23	20			ns
t_{IH}	AV clock input hold time		Figure 23	3			ns
t_{OD}	AV clock output delay time		Figure 23	3		20	ns
t_{WHIGH}	AV clock pulse width HIGH		Figure 23				
t_{WLOW}	AV clock pulse width LOW		Figure 23				
t_{PWFSO}	AVFSYNCOUT pulse width HIGH		Figure 24	100		140	ns
t_{PWFSI}	AVFSYNCIN pulse width HIGH		Figure 25	100			ns
t_{SUP}	PHY-link setup time		Figure 26	6.0			ns
t_{HP}	PHY-link hold time		Figure 26	2.5			ns
$t_{SCLKPER}$	SCLK period		Figure 26	20.347	20.345	20.343	ns
t_{DP}	PHY-link output delay	Note: C_L = 20pF	Figure 27	2.0		14.0	ns
t_{AS}	Host address setup time		Figure 28	0			ns
t_{AH}	Host address hold time		Figure 28	0			ns
t_{CL}	Host chip select pulse width LOW		Figure 28	115			ns
t_{CH}	Host chip select pulse width HIGH		Figure 28	42			ns
t_{RP}	Host read pulse width		Figure 28	115			ns
t_{ACC}	Host access time		Figure 28			115	ns
t_{DH}	Host data hold time		Figure 28	0			ns
t_{DS}	Host data setup time		Figure 28	0			ns
t_{WRP}	Host write pulse width		Figure 28	115			ns
t_{CWH}	CYCLEIN HIGH pulse width		Figure 29	200			ns
t_{CWL}	CYCLEIN LOW pulse width		Figure 29	200			ns
t_{CP}	CYCLEIN cycle period		Figure 29	125			µs
t_{CD}	CYCLEOUT cycle delay		Figure 30			20	ns
t_{RESET}	RESET_N pulse width LOW		Figure 31	10			µs

16.0 TIMING DIAGRAMS

16.1 AV Interface Operation

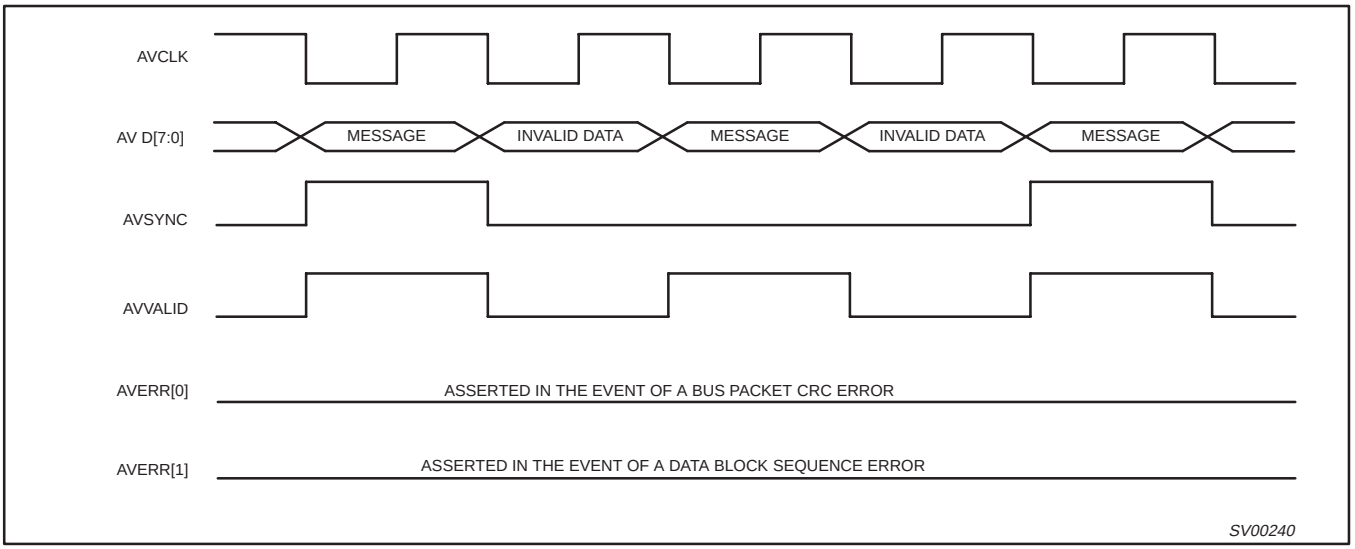


Figure 22. AV Interface Operation Diagram

16.2 AV Interface Critical Timings

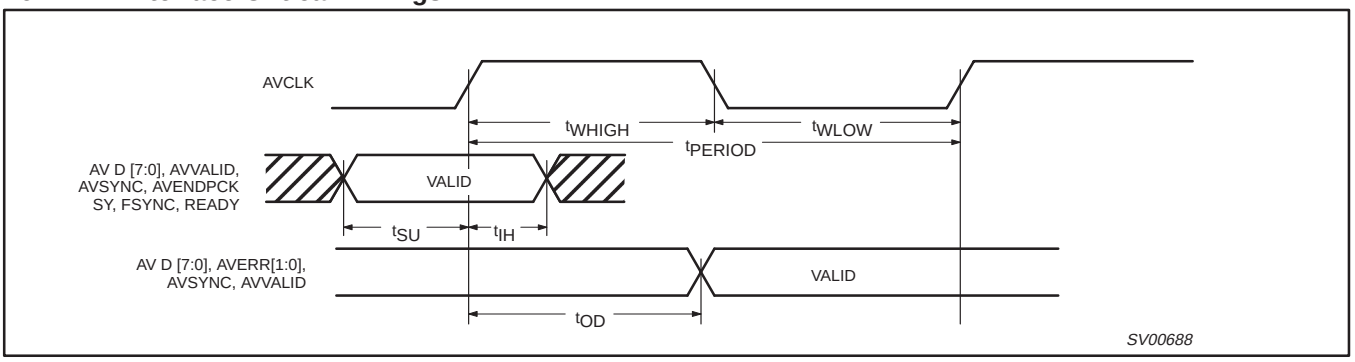


Figure 23. AV Interface Timing Diagram

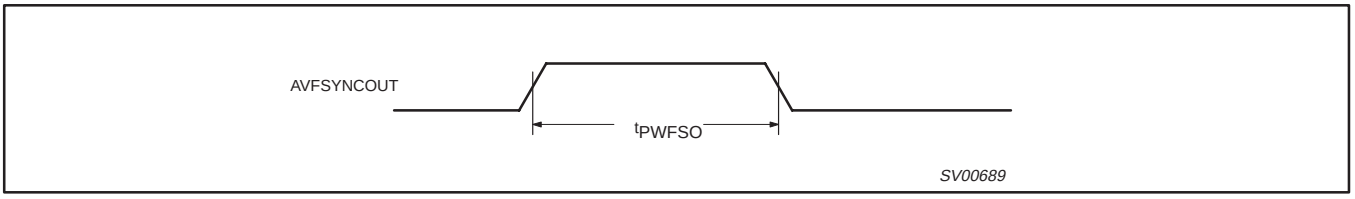


Figure 24. AVFSYNCOUT Timing Diagram

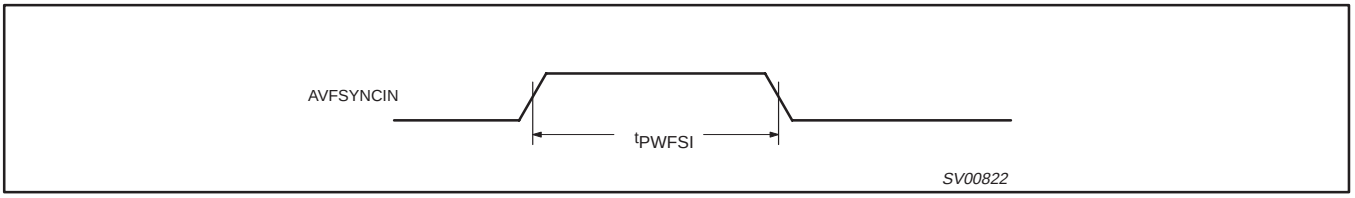


Figure 25. AVFSYNCIN Timing Diagram

16.3 PHY-Link Interface Critical Timings

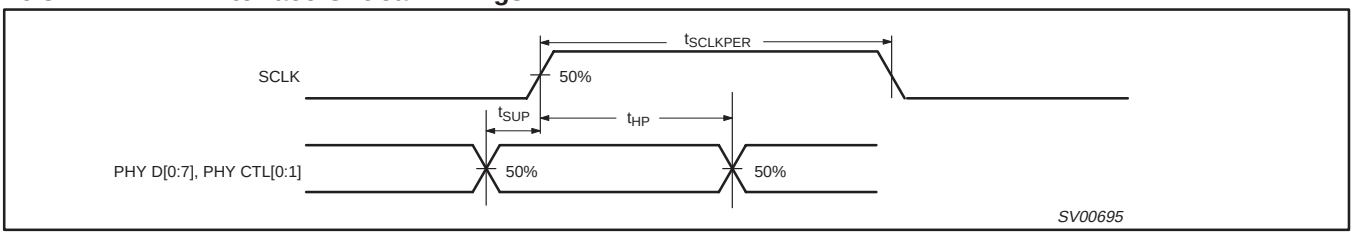


Figure 26. PHY D[0:7], PHY CTL[0:1] Input Setup and Hold Timing Waveforms

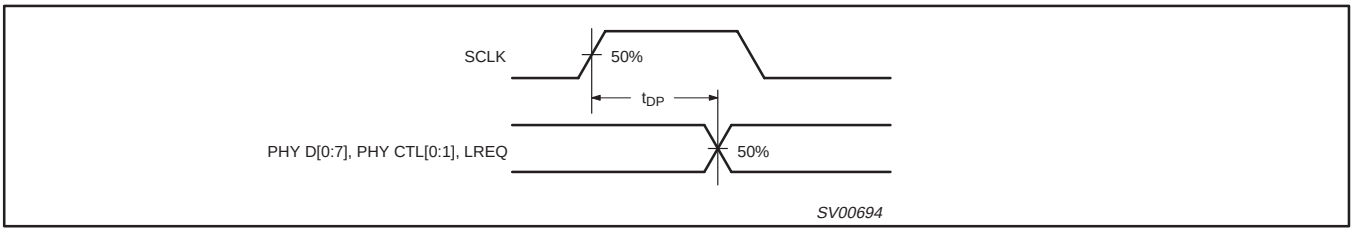


Figure 27. PHY D[0:7], PHY CTL[0:1], and LREQ Output-Delay Timing Waveforms

16.4 Host Interface Critical Timings

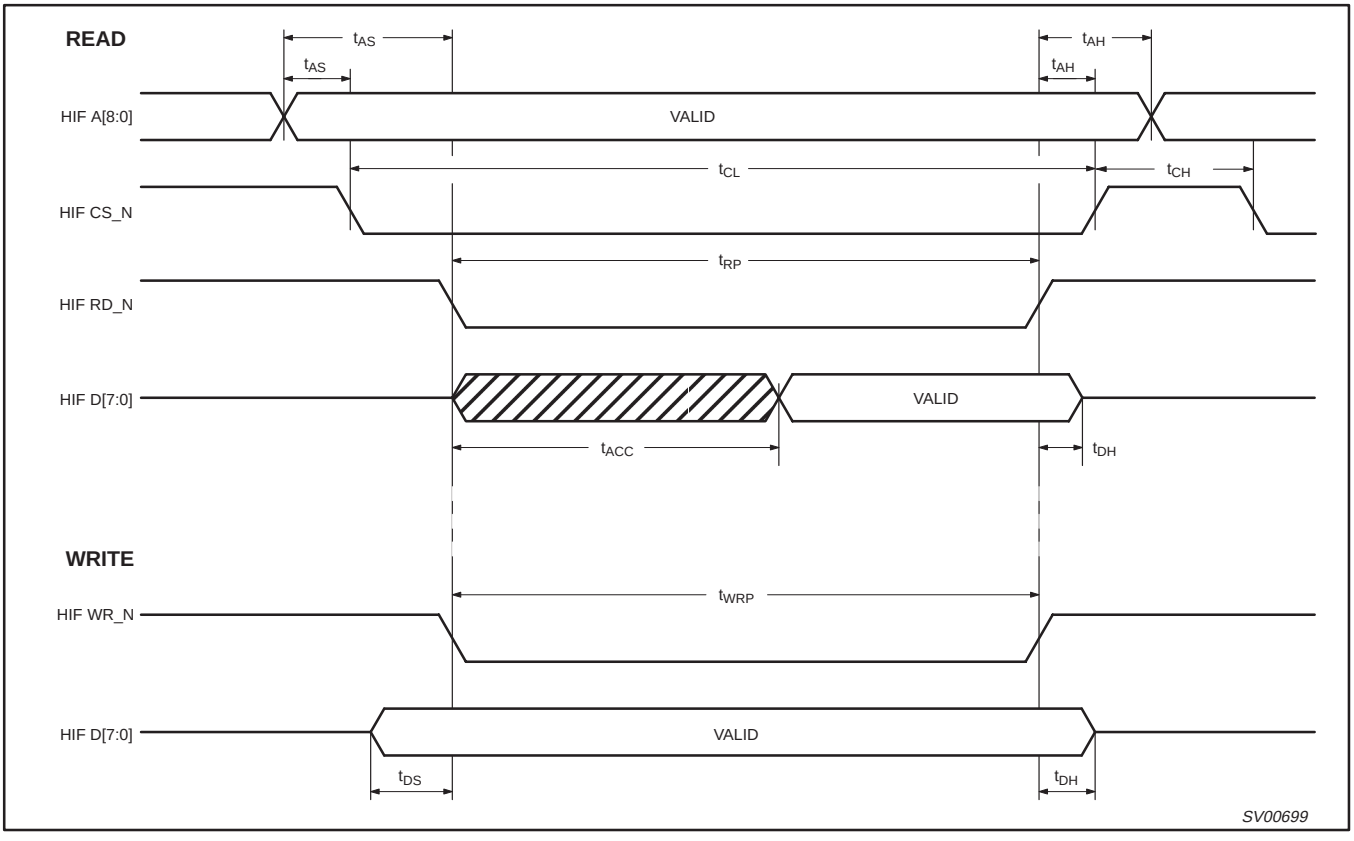


Figure 28. Host Interface Timing Waveforms

16.5 CYCLEIN/CYCLEOUT Timings

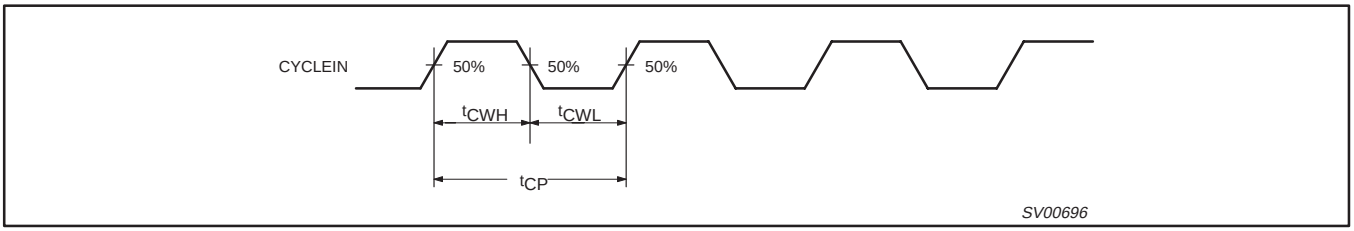


Figure 29. CYCLEIN Waveform

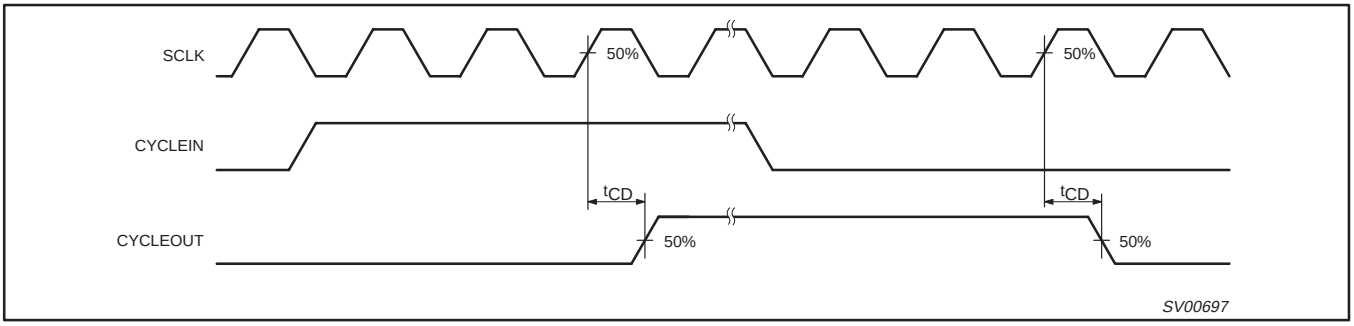


Figure 30. CYCLEOUT Waveforms

16.6 $\overline{\text{RESET}}$ Timings

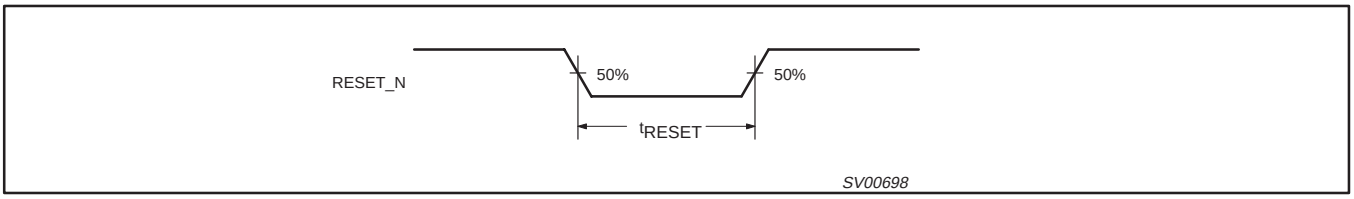
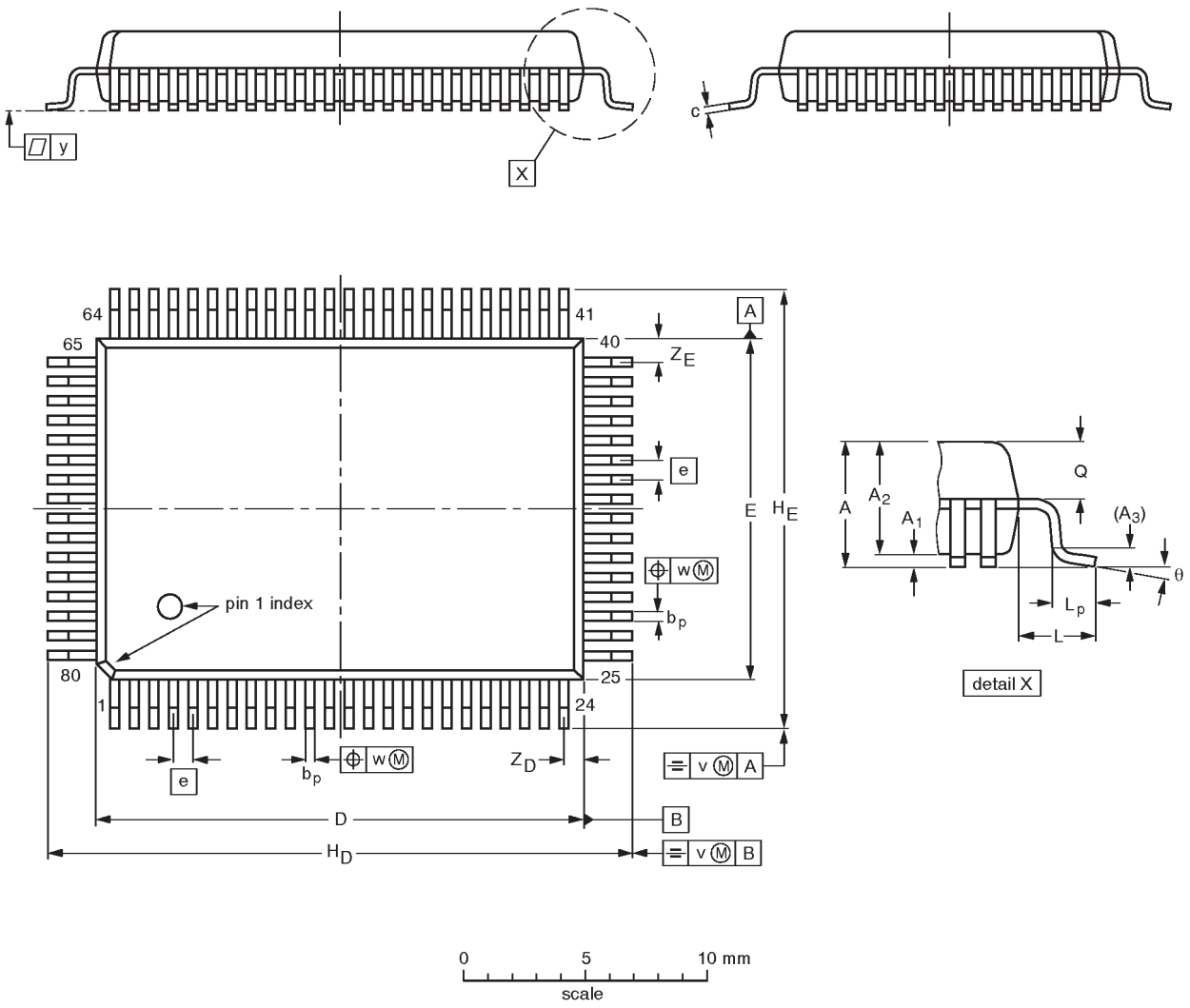


Figure 31. RESET_N Waveform

QFP80: plastic quad flat package; 80 leads (lead length 1.95 mm); body 14 x 20 x 2.8 mm SOT318-2



DIMENSIONS (mm are the original dimensions)

UNIT	A max.	A_1	A_2	A_3	b_p	c	$D^{(1)}$	$E^{(1)}$	e	H_D	H_E	L	L_p	Q	v	w	y	$Z_D^{(1)}$	$Z_E^{(1)}$	θ
mm	3.2	0.25 0.05	2.90 2.65	0.25	0.45 0.30	0.25 0.14	20.1 19.9	14.1 13.9	0.8	24.2 23.6	18.2 17.6	1.95	1.0 0.6	1.4 1.2	0.2	0.2	0.1	1.0 0.6	1.2 0.8	7° 0°

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT318-2						~~92-12-15~~ 95-02-04

ERRATA LIST FOR THE PHILIPS PDI1394L11 1394 AV LINK LAYER CONTROLLER

(These errata refer to the data sheet dated 1997 Oct 21)

Chip Errata:

E-1. **Receive and filter only RESPONSE packets, ARXALL:**

Description of expected operation: The ARXALL bit is designed to filter out unsolicited response packets. When set (1), all responses are stored. When cleared (0) only solicited responses are stored.

Description of observed behavior: When ARXALL is set (1), it works as expected. When ARXALL is cleared (0) AND the node is waiting for a response, only solicited responses are stored. When cleared (0) AND the node is NOT waiting for a response, all responses are stored, including unsolicited responses.

Solution or work around: No work around.

E-2. **Cycle time out, CYTMOUT, occasionally set incorrectly after bus reset:**

Description of expected operation: The Link generates cycle timeout indications during bus reset. This signal indicates that the time between cycle start packet and the first subsequent subaction gap exceeded 125 microseconds.

Description of observed behavior: If the AV transmitter or AV receiver is active during a bus reset, a cycle time (CYTMOUT) out interrupt will generally gets set. If this node is cycle master then it will no longer send cycle start packets.

Solution or work around: Cycle master functionality can be restored by the following procedure: clear CYMASTER (0) (bit 11 of LNKCTL register), acknowledge CYTMOUT (1), then reset CYMASTER (1).

E-3. **Confirmation packet not seen when the acknowledge code is not received for a transmitted request:**

Note: Originally this error was thought to be true only for Broadcast Packets, however, it is now found to apply whenever an acknowledge code is not received for a transmitted packet.

Description of expected operation: When a Request packet is transmitted and no acknowledge code is received, then a synthesized confirmation packet should be generated and placed at the Receiver Request Queue. Also, the Receiver Request Queue Quadlet Available (RREQQQAV) interrupt should be set.

Description of observed behavior: The confirmation packet is stored in the link layer controller but it is not available in the Receiver Request Queue. Several of these confirmation packets can accumulate. These packets become available in the Receiver Request Queue if a packet is received at the Receiver Request Queue immediately after the request is transmitted.

Solution or work around: If a confirmation packet is not received on time after a request packet is transmitted, then assume that no acknowledge code was received. Also, if you receive confirmation packets unexpectedly, disregard them.

ERRATA LIST FOR THE PHILIPS PDI1394L11 1394 AV LINK LAYER CONTROLLER

(These errata refer to the data sheet dated 1997 Oct 21)

E-4. **SIDQAV is not always set correctly:**

Description of expected operation: SIDQAV is set upon receipt of a self-ID packet.

Description of observed behavior: Depending upon precisely when a bus reset is seen, occasionally the SIDQAV interrupt bit may not be set.

Solution or work around: Use the bus reset interrupt bit along with the RREQQQAV interrupt to determine when self ID quadlets are in the queue. See also the work around for E-5.

E-5. **Self-ID packet issues:**

Description of expected operation: After bus reset, the queues are flushed and a self-ID packet should be placed in the Read Request Queue. Refer to section 12.5.2.4 (Self-ID and PHY packets receive) in the ″PDI1394L11 1394 AV Link Layer Controller″ data sheet for the proper format of a self-ID packet.

Description of observed behavior: The following outcomes have been observed:
(1) A correct self-ID packet.
(2) A self-ID packet missing the 0x000000E0 header, but otherwise correct.
(3) Nothing in the Read Request Queue (not even self-IDs).
(4) Self-ID data delivered in the individual PHY packet format.
(5) A 0x000000E0 header and 0x0000000D acknowledge code but no self-ID between them.

Solution or work around: If there are no quadlets in the queue, i.e., RREQQQQAV=0, then it can be assumed that there is a self-ID error, and a bus reset should be issued.

If there is a single quadlet and its value is 0x00000001, this would indicate that this is an isolated node missing the header 0x000000E0. If there is a single quadlet and its value is not 0x0000001 then a bus reset should be issued. If there are two or more quadlets, then a check for a correct self-ID sequence should be conducted. If the first quadlet is 0x000000E0, discard it and continue. Now look for zero or more quadlet pairs where the second is the bitwise inverse of the first. When there is one quadlet left, and the pairing of each quadlet with a bitwise inverse has succeeded up to this point, then check the last quadlet for the value of 0x00000001. If this is the case, a correct self-ID sequence has been detected, otherwise do a bus reset.

E-6. **Incorrect reception of PHY packets:**

Description of expected operation: A proper PHY packet consists of two quadlets. These two quadlets are (1) the synthesized header with value 0x000000E0 and (2) the first (non-inverted) quadlet as transmitted on the bus. Refer to section 12.5.2.4 (Self-ID and PHY packets receive) in the ″PDI1394L11 1394 AV Link Layer Controller″ data sheet for a description of a PHY packet.

Description of observed behavior: In some cases only the first header (value 0x000000E0) is transferred to the FIFO with the PHY packet being lost. The header will be prepended to and included as part of any subsequent received request packet causing that packet appear to be incorrectly framed. This has been seen to occur when a PHY packet arrives immediately (1) after a cycle-start packet, (2) after any packet which has been aborted due to insufficient FIFO space, or (3) after the retry protocol requires a busy_ack. The headers do not trigger interrupt RREQQQAV.

Solution or work around: Disregard spurious header (value 0x000000E0) quadlets.

ERRATA LIST FOR THE PHILIPS PDI1394L11 1394 AV LINK LAYER CONTROLLER

(These errata refer to the data sheet dated 1997 Oct 21)

E-7. **Incorrect confirmation packet received when broadcasting to an even numbered bus ID:**

Description of expected operation: When a broadcast packet is sent to any bus ID other than the local bus (3FF), the resulting confirmation packet should contain an acknowledge code of ack_complete.

Description of observed behavior: When a broadcast packet is sent to an even numbered bus ID, the resulting confirmation packet contains an incorrect acknowledge code of ack_missing.

Solution or work around: Expect an acknowledge code of ack_missing for even numbered bus ID addresses.

E-8. **Zero length lock response packets are not visible:**

Description of expected operation: All lock response packets should be visible at the receiver FIFO.

Description of observed behavior: Lock response packets with response code (rcode) value other than ″resp_complete″ (specified by the standard to be of zero length) are not visible on the receiver FIFO.

Solution or work around: Typically the response code for a lock subaction is ″resp_complete″. If a lock response is not received as expected, then it can be assumed that there was an error and the response code is other then ″resp_complete″.

E-9. **Wrong detection of CIPTAGFLT and RCVBP bits (register IRXINTACK 0x4C):**

Description of expected operation: When the Isochronous Receiver is disabled, interrupt bits CIPTAGFLT and RCVBP should be inactive.

Description of observed behavior: Occasionally, when the Isochronous Receiver is disabled, the CIPTAGFLT and RCVBP interrupt bits get set. When the Isochronous Receiver is enabled, then these bits work as expected.

Solution or work around: Keep the receiver in RESET state when not in use; this will stop these bits from being set erroneously.

E-10. **Error transmitting large asynchronous packets:**

Description of expected operation: Packets of up to 64 quadlets (256 bytes) could be transmitted.

Description of observed behavior: When transmitting asynchronous requests (or responses), if the combined size of any two consecutive packets is greater than 64 quadlets, then interrupt bit RREQQFULL (or RRSPQFULL) gets set as soon as the 64th quadlet is written to the FIFO, and the transmission is halted.

Solution or work around: When transmitting large packets make sure that the size of any two consecutive packets is less or equal to 64 quadlets. One way to do so is to transmit a small packet after transmitting a large packet. Another way is to keep the size of the transmitted packets 32 quadlets or less.

ERRATA LIST FOR THE PHILIPS PDI1394L11 1394 AV LINK LAYER CONTROLLER

(These errata refer to the data sheet dated 1997 Oct 21)

E-11. **Wrong detection of TRSPQIDLE and TREQQIDLE when the FIFO is full:**

Description of expected operation: TRSPQIDLE and TREQQIDLE indicate that the transfer registers for the transmitter response and transmitter request queues, respectively, are idle. When a quadlet is written to either transfer register, the corresponding idle bit, TRSPQIDLE or TREQQIDLE is cleared. When the quadlet is transferred to the FIFO the idle bit gets set. This would be an indication to the application that the transfer register is ready to be written with the next quadlet.

Description of observed behavior: If a quadlet is written to the transfer register when the FIFO is full, the quadlet will remain in the transfer register and will not be transferred to the FIFO. The idle bits get set while they should remain cleared.

Solution or work around: Before writing to the transfer register, the application should make sure that the FIFO is not full and that the previous write to the transfer register has been successfully transferred to the FIFO. The application can check bits TREQQF and TRSPQF (bits 3 and 7 of ASYMEN 0x084) to determine if the FIFO is full. It can check bits REQQWR and TRSPQWR (bits 1 and 2 of ASYINTACK 0x0A0) to determine if the previous write to the transfer register has been successfully transferred to the FIFO.

E-12. **Dual phase fairness interval not compliant with 1394-1995 standards:**

Description of expected operation: Section 3.6.2.4 of the 1394-1995 standard specifies that dual phase destination nodes shall assume all retry_A (or retry_B) subactions have been sent when four fairness intervals pass with no retry_A (or retry_B) subactions having been busied.

Description of observed behavior: For inbound transactions, one fairness interval is implemented instead of four.

Solution or work around: The application should specify that only single-phase protocol will be used for inbound. This is done by setting the Busy Control field (BSYCTRL: bits 27–29 of LNKCTL 0X004) to 011.

E-13. **Acknowledge packet is not properly received when the actual payload size of a transmitted packet is zero:**

Description of expected operation: When a Block Request Packet with actual payload size of zero is transmitted, the PDI1394L11 should expect an acknowledge code from the receiver. It should also generate a confirmation packet with the acknowledge code.

Description of observed behavior: When a packet with actual payload size of zero is transmitted, the PDI1394L11 does not properly receive the acknowledge code. Thus it behaves as if not acknowledge code is received (See E-3). That is, it generates a confirmation packet specifying that now acknowledge code is received while the receiver has actually sent an acknowledge code. Also, the confirmation packet is stored in the link layer controller but it is not available in the Receiver Request Queue. Several of these confirmation packets can accumulate. These packets become available in the Receiver Request Queue if a packet is received at the Receiver Request Queue immediately after the request is transmitted.

Solution or work around: When a packet of payload size equals to zero is transmitted, do not expect a confirmation packet. Also, if you receive confirmation packets unexpectedly, disregard them.

ERRATA LIST FOR THE PHILIPS PDI1394L11 1394 AV LINK LAYER CONTROLLER

(These errata refer to the data sheet dated 1997 Oct 21)

E-14. **No acknowledge code is sent when the actual size of a received payload is greater than the one specified in the header:**

Description of expected operation: Whenever a block-payload packet addressed to the current node is received, and the block length given in the header mismatches the actual length, the Link Layer should send back an acknowledge code of ack_data_error.

Description of observed behavior: If the actual size of the received payload is greater than the one specified in the header, then the PDI1394L11 does not send an acknowledge code.

Solution or work around: No work around.

E-15. **Link / PHY interface hang-up:**

Description of expected operation: When the subject node is root and also an isochronous talker, the transmission of isochronous data packets will continue on the bus through all bus reset events and thereafter.

Description of observed behavior: When root and an isochronous talker, after some bus reset events the L11 ceases to transmit isochronous data on the bus. The L11 actually requests access to the bus too late in the bus cycle for isochronous transmission, this causes the PHY to ignore the request while the L11 waits for the PHY to grant the isochronous access. The L11 waits in this state indefinitely, or until the next bus reset event. If another isochronous talker is present on the bus, the L11 hang-up will not occur.

Solution or work around: If the L11 is used with Philips P11A PHY there is a chance of this misoperation occurring. The L11 has been tested with the Philips P21, 400 Mbps PHY and the problem does not occur. If the L11 is an isochronous talker, but not root, the problem does not occur. If it is desired to send isochronous data on the bus, first send a PHY configuration packet to another node on the bus followed by a bus reset to make that node root (before starting isochronous transmission).

E-16. **Link / PHY interface does not initialize properly when single capacitor galvanic isolation is used:**

Description of expected operation: The link / PHY interface initializes with all opposing pins coming out of reset with low levels set on them. Pins that initialize as inputs are configured as inputs with the pull-down bus hold resistance active, thus weakly holding each input pin in the low state. I/Os are configured as outputs with low states being held on these pins during reset, and with these pins being configured as inputs with low state bus hold resistors active just after the chip reset is released.

Description of observed behavior: Occasionally a pin of the L11 has been observed to be high just after the release of the reset signal. This operation can cause the node to ″lock-up″ because proper link / PHY interface operation is not possible when opposite states appear across any galvanic isolation coupling capacitor.

Solution or work around: Place a 39 K Ω resistor at each capacitor coupled link / PHY interface input and I/O pin of the L11 (SYSCLK, CTL0, CTL1, D0...D7) as a pull-down device. The slight additional pin to ground current will set the proper pin state at the release of reset.

NOTES

DEFINITIONS

Data Sheet Identification	Product Status	Definition
Objective Specification	Formative or in Design	This data sheet contains the design target or goal specifications for product development. Specifications may change in any manner without notice.
Preliminary Specification	Preproduction Product	This data sheet contains preliminary data, and supplementary data will be published at a later date. Philips Semiconductors reserves the right to make changes at any time without notice in order to improve design and supply the best possible product.
Product Specification	Full Production	This data sheet contains Final Specifications. Philips Semiconductors reserves the right to make changes at any time without notice, in order to improve design and supply the best possible product.

Philips Semiconductors
811 East Arques Avenue
P.O. Box 3409
Sunnyvale, California 94088–3409
Telephone 800-234-7381

Date of release: 10-97

Document order number: 9397 750 07547

Let's make things better.

Philips Semiconductors

PHILIPS