AMCC
APPLIED MICRO CIRCUITS CORPORATION

S4804CBI41

Rhine Datasheet

Revision 2.1

May 24, 2002

APPLIED MICRO CIRCUITS CORPORATION

Dear customer,

Thank you for choosing an AMCC device. We appreciate your confidence in our products.

To ensure your complete satisfaction with our products and technologies, we have prepared this publication to provide you additional information, which will help you use the device more efficiently. (This publication may be held to our confidential and proprietary requirements and non-disclosure agreements.)

If you have any questions, concerns, or comments regarding this publication or device, or if you'd like to be added to our distribution list for future publications, send us an email addressed to **pubs_info@amcc.com**. Please indicate the devices you are interested in. If you want to be removed from a device's distribution list, please send an email to pubs_info@amcc.com and include REMOVE in the subject line.

AMCC

APPLIED MICRO CIRCUITS CORPORATION

Part Number - S4804CBI41

Product Brief Version 2.0 - January 2002

Datasheet Revision 2.1

RHINE

PRODUCT BRIEF

OC-48 / 4xOC-12 / 16xOC-3 SONET/SDH FRAMER AND POS/ATM MAPPER

Features

- Provides a SONET/SDH STS-48/STM-16, 4 STS-12/STM-4, or 16 STS-3/STM-1 line interfaces.
- STS-48/STM-16 data stream supports a single STS-48c/AU-4-16c, or any valid combination of STS-12c/AU-4-4c and/or STS-3c/AU-4 SONET/SDH payloads.
- Each STS-12/STM-4 data stream supports a single STS-12c/AU-4-4c or 4 STS-3c/AU-4 SONET/SDH payloads.
- Each STS-3/STM-1 data stream supports a single STS-3c/AU-4 SONET/SDH payload.
- Supports mixed STS-3 / STS-12 line rates
- Provides full-duplex mapping of ATM cells or packets in each payload tributary.
- Supports termination of mixed ATM and POS tributaries.
- Terminates/generates SONET/SDH section, line, and path layers with transport/section E1, E2, F1, and DCC overhead interfaces in both transmit and receive directions.
- APS port to support protection-switching configurations between two RHINE devices.
- 16-bit, bus interface at 155 MHz for STS-48/STM-16 mode, or serial interfaces operating at 622/155 MHz for STS-12/3 (STM-4/1) modes on the line side.
- 32-bit, parallel interface (FlexBus-3^{TM}) operating at 100 MHz on the system side.
- .25 micron, 2.5V core, and 3.3V tolerant I/O.
- Packaged in a 624 Pin CBGA.

The S4804 is a highly-integrated VLSI device that provides full-duplex mapping of packets or ATM cells to SONET/SDH payloads. It provides support for both uni-directional and bi-directional rings.

The S4804 provides full section, line, and path overhead processing, and supports framing, scrambling/descrambling, alarm signal insertion/detection, and bit-interleaved parity (B1/B2/B3) processing.

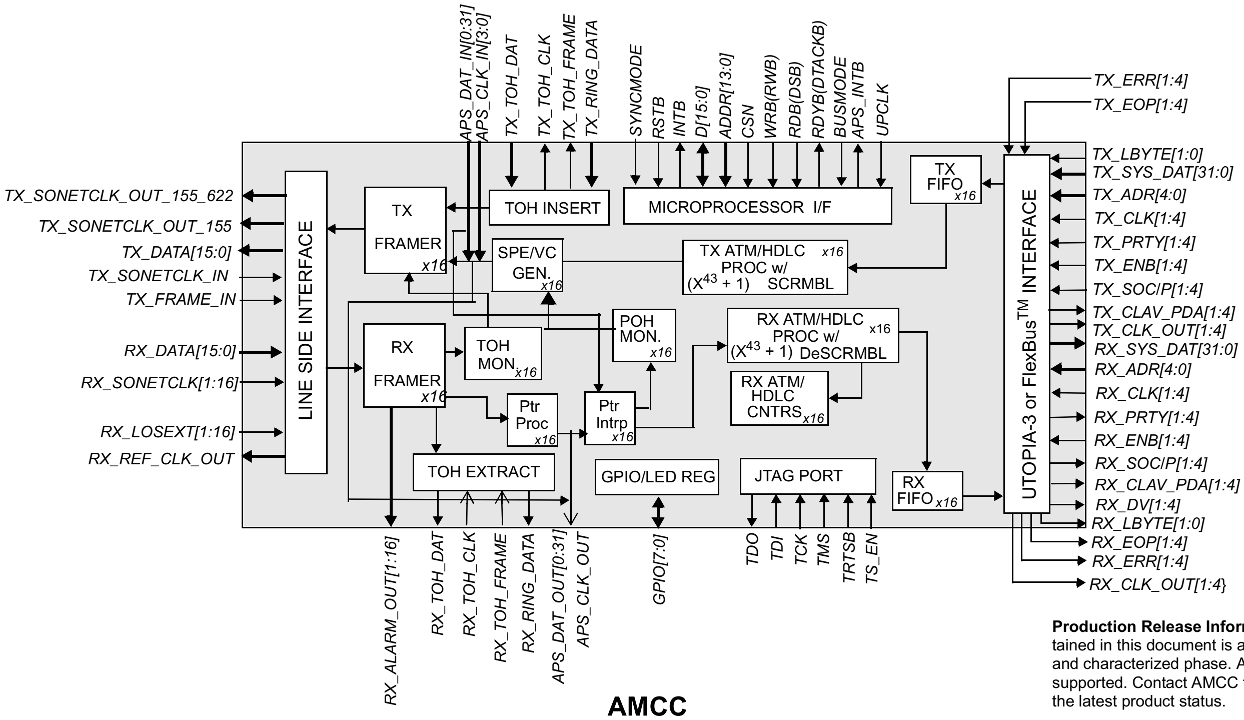
The S4804 is SONET/SDH standards compliant with Bellcore GR-253, ITU G.707, ITU-T 432.1, ANSI T1.105 -1995, and IETF RFCs 1619/1661/1662/2615.

A general purpose 8-bit or 16-bit microprocessor interface is provided for control and monitoring. The interface supports both Intel^{TM} and Motorola^{TM} type microprocessors, and is capable of operating in either an interrupt-driven or polled-mode configuration. In addition, a standard 5 signal IEEE 1149.1 JTAG Test Port is provided for Boundary Scan test purposes.

Applications

- ATM switches
- Packet over SONET Routers and Switches
- SONET/SDH Add Drop Multiplexers, Terminal Multiplexers, and Digital Cross Connects
- Test equipment

S4804CBI Block Diagram



Production Release Information - The information contained in this document is about a product in its fully tested and characterized phase. All features described herein are supported. Contact AMCC for updates to this document and the latest product status.

Datasheet Revision 2.1

Product Brief Version 2.0 - January 2002

S4804CBI41: RHINE

STS-48 POS/ATM SONET MAPPER

PRODUCT BRIEF

Overview and Applications

Sonet/SDH Processing

The S4804 implements SONET/SDH processing and full-duplex ATM/packet-mapping functions for STS-48/STM-16, STS-12/STM-4, or STS-3/STM-1 data streams. It can support either a single STS-48c/AU-4-16c or any valid combination of STS-12c/AU-4-4c or STS-3c/AU-4 signals within an STS-48/STM-16. The S4804 also supports 4 STS-12/STM-4 signals (each containing a single STS-12c/AU-4-4c or 4 STS-3c/AU-4), or 16 STS-3c/STM-1 signals each containing an STS-3c/AU-4.

A TOH/SOH interface provides direct add/drop capability for E1, E2, F1, and both Section and Line DCC channels. The S4804 also includes a clear channel mode that enables the direct transmission of system payload from the system interface to the line-side interface.

On the transmit side, the S4804 generates section, line, and path overhead. It performs framing pattern insertion (A1, A2), scrambling, alarm-signal insertion, and generates section, line, and path Bit Interleaved Parity (B1/B2/B3) for far-end performance monitoring.

On the receive side, the S4804 processes section, line, and path overhead. It performs framing (A1, A2), descrambling, alarm detection, pointer processing, Bit Interleaved Parity monitoring (B1/B2/B3), and error-count accumulation for performance monitoring.

ATM Processing

When configured for ATM cell processing, the S4804's transmit ATM processor(s) will perform all necessary cell processing as defined by ATM UNI3.1, ITU-T I.432.1, and I.432.2.

HDLC Processing

When configured for POS mode, the S4804's HDLC processor(s) provides HDLC packet processing as defined by IETF RFCs 1619, 1662 and 2615. In addition, the S4804 optionally performs scrambling (X^{43}+1).

Direct Map Mode

Direct Map Mode allows to map any protocol directly into the Sonet/SDH Synchronous Payload Envelope, by-passing the ATM and HDLC processing circuitry.

Automatic Protection Switching

The S4804 provides APS input and output interfaces to convey signals between two S4804 devices configured for APS operation. This configuration supports both 1+1 and 1:1 configurations.

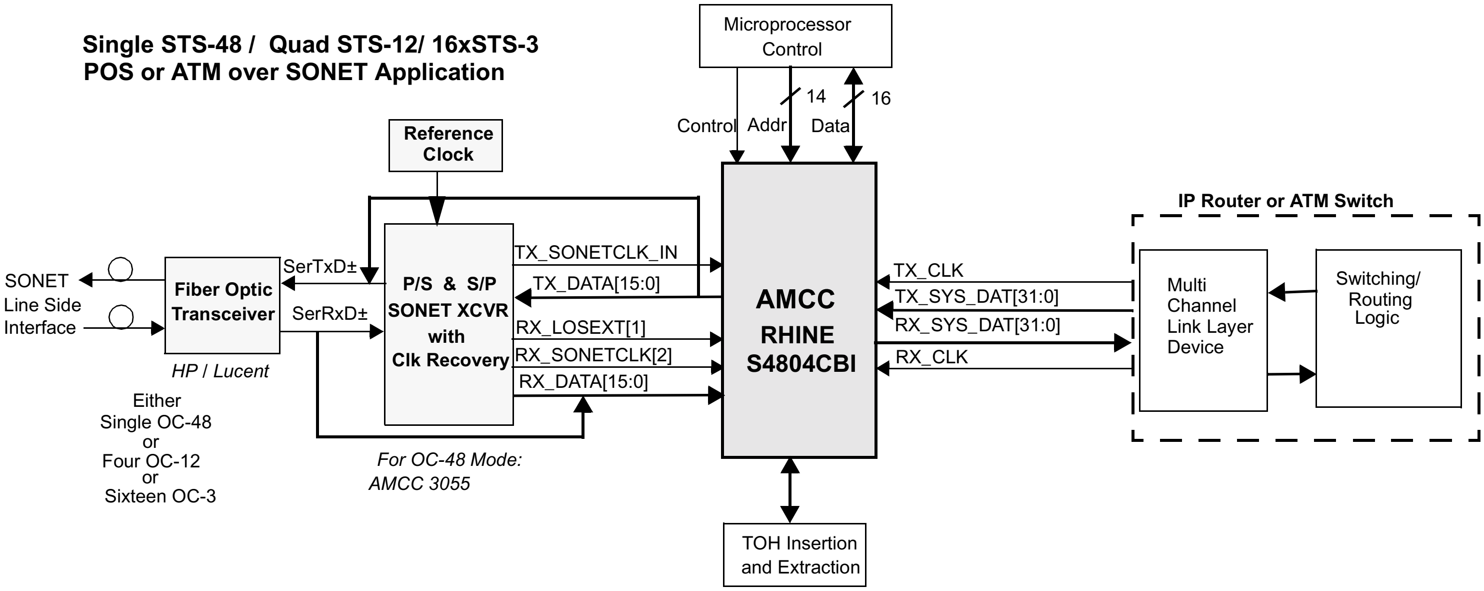
Line-side Interface

On the line side, the S4804 supports a 16-bit parallel interface, operating at 155MHz for a single OC-48 optical interface. It provides serial interfaces at either 622 MHz or 155 MHz for OC-12 and/or OC-3 optical interfaces. Mixed OC-3 / OC-12 line rates are supported.

System Interface

The S4804 supports a 32-bit, 100-MHz system interface. For ATM cell transfers, the S4804 supports Utopia Level 3 interface. For packet transfers, the S4804 supports FlexBus™ interface. The S4804 also provides support for a quad, 8-bit extension of the Utopia 3.

TYPICAL APPLICATIONS: S4804CBI - RHINE in ATM or POS System



Specification History

Rev.	Date	Description of Change
1.0	5/24/99	Incorporated Engineering Requirements document and I/O (Rev 14) document.
1.1	6/15/99	Incorporated review updates and new I/O (Rev 14) document changes.
1.1	6/15/99	Incorporated review updates and new I/O (Rev 14) document changes.
1.2	8/25/99	Updated document.
1.3	9/30/99	Included Table 9.
1.4	12/2/99	Incorporated the final I/0 spec; added appendix with corrections to memory maps; fixed typos based on customer feedback and reviews.
1.5	7/14/00	This is a major revision of RHINE. Change bars appear on the outside edge; they reflect any information that has changed.
1.6	12/13/00	• Note: Capabilities new in Rhine 4.0 are shown in blue with double underline. Change bars on the outside edge designate corrections and editorial changes. • Updated AC timing diagrams and DC specifications. • Inserted Power figures • Described GPIO when SYS_SYNC_OUT_EN=1. • Removed J1_3_MATCH functionality. • Deleted B3 SF/SD monitoring blocks since not adequate for mixed payloads. • Added TX_CLK_OUT_INH_[1:4] • Renamed TX_SEL_ENB to RX_SYS_B2B_ALLOW. • Updated default reset values in register map. • Removed TSI cross-connect functionality (HPIN_SEL_x[3:0]). • Corrected J0, J1 Framing description (search order). • Modified Flexbus-3 to be Utopia-3 compliant. • Removed built-in support for UPSR: RDI/REI Ring channel, dual-feed and auto path switching, RX_LAIS_SW_INH_x • Removed references to TX_FIFO_ERR_E • Removed framer bypass functionality. • Added provisionable send EOP, LBYTE in ATM mode; Made interpretation of LBYTE programmable. • Corrected mislabeled “Do Not Use” table. • Corrected RX, TX_SING_CLAV reset value (=1). • Removed restriction on single flags between packets. • Corrected B2 SD Group Clear value. • Added SYS_SYNC_OUT_EN functionality • Added APS_DELAY_EN to accommodate > 156ns delay. • Deleted RX_LOS_INH inhibit for PAIS gen based on LOS. Documented current use of RX_LOSEXT_INH_x to inhibit PAIS gen due to LOS. • Corrected location of RX_PG_FIFO_i_j_E bits in reg map. • Removed PLL support. • Corrected APS K1K2 Insert text • Corrected TX_, RX_TRIB_INH functional descriptions. • Deleted TX_FOUT_BYTE[2:0], TX_FRAME_IN_INH_[x] since they’re not used to control frame alignment. Specified frame sync alignment after resync. • Corrected K2 capture for 5 MSBs in K2[7:3]; 3 LSBs separately. • Corrected Table 4 and section 7.3 for only one APS_CLK_OUT. • Removed Sonet R to T Loopback support for STS-3 and STS-12

Rev.	Date	Description of Change
1.7	1/5/01	• Corrected errors in Pin Description section and Register Map section • Added Tables 6, 7, and 8 that were missing from Rev 1.6. • Corrected Tables 38 and 42: updated STS-3C table values.
1.8	7/2/01	• Corrected interpretation of fifofull, fifoempty for POS. • Deferred RX_LAIS_DATA_DIS_INH_x; Not implemented and not essential. • (Editorial) Changed “Read Only, cleared by uP” to “Read / cleared by uP” • APS input expects, checks only a single B1 byte (section 8.2.3) • Changed Pin signal names for TOH interface to remove _RING_ (for consistency w/sumd/s) • Corrected PATH_J1_SUMD equation to eliminate PATH_J1_SUMD_MASK used in next higher level. • J1 trace order reverted to TX_J1_COMMON_[47:0]_[7:0]. • Corrected resolution of FIFOFULL, FIFOEMPTY regs to 32 bytes in STS-3c. • Removed refs to TX_POH_SCR_INH and RX_POH_DSCR_INH. • Per Rhine 4.0 timing, updated tPRXI min/max from 2.5/8ns to 2/6ns; Changed T_{HTX15} and T_{HTX15} from .75ns to .5ns. Combined STS48C T_{STX15} and T_{HTX15} to a common spec for TX_DATA[0:15] • All references to FILL_CHECK tagged with NotInRhine4.0. • Tagged remaining refs to LINE_RING and PATH_RING as NotInRhine4.0. • Updated Power numbers in DC spec section to provide measured values for each mode • Removed section 12 (since it is redundant to section 2 and has outdated information. • Changed TOH I/O type in pin desc section to 3.3V tolerant • Conditionalized references to TX_POS_FIFOUNDER_E • Corrected ESD rating units from kV to V. • ATM_IDLE_CAP_E no longer supported; Text tagged as NotInRhine4.0. • RX_PI_PAIS_x, RX_PI_LOP_x = undefined for concatenated tribs. • BIT_BLKCNT does not apply to B2 and M1 in Rhine 4.1 to comply with SONET/SDH specs (B1, B3/G1 can count bit or block errs; B2/M1 always count bit errs). • UDF2-UDF4 undefined in RX direction. • Changed TXFRAMEIN timing diagram and spec from a setup and hold time to a pulse width requirement since TXFRAMEIN is asynchronous to TXSONETCLKIN.
1.9	11/27/01	• TX P-RDI is NOT removed within one frame of removal of received P-AIS. Document exact behaviour. • Increase Max FlexBus3 clock spec to 104Mhz • Fix minor typographical errors in reg map • Corrected error in first line of RX_REF_CLK_OUT selection table • Add text to description of TX/RX_APS_SEL • Add description of RX_ENB toggle at initialization requirement • Added user notes for Rhine 4.1 as an appendix to the datasheet.

Rev.	Date	Description of Change
2.0	1/8/02	• Changed document to Production Released document - no longer an NDA document status. • **Added final User Notes, see section called Appendix A: "User Notes".**
2.1	5/24/02	• Editorial changes. • Added 1 ball (pin A1) to Mechanical Packaging Information, section 3.0.

Note: Features and changes applicable to Rhine 4.0 and later are indicated by blue double-underscored text.

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

1.0 Applicable Documents

1. ATM Forum, “Utopia Specification Level 1,” Version 2.01, af-phy-0017.001, March 1994.
2. ATM Forum, “Utopia Specification Level 2,” Version 1.0, af-phy-0039.000, June 1995.
3. ATM Forum, “Utopia Specification Level 3,” af-phy--136.000, November 1999.
4. ATM Forum “ATM User-Network Interface Specification,” V3, 1af-uni-0010.002, 1994.
5. “PPP Over SONET/SDH,” IETF RFC 2615, June 1999.
6. “PPP in HDLC-Like Framing,” IETF RFC 1662, July 1994.
7. “The Point-to-Point Protocol,” IETF RFC 1661, July 1994.
8. ISO/IEC 3309:1991(E), “Information Technology - Telecommunications & information exchange between systems - High Level Data Link Control (HDLC) procedures - Frame structure,” International Organization for Standardization, Fourth Edition 1991-06-01.
9. ISO/IEC 3309:1991/Amd.2:1992(E), “Information Technology - Telecommunications & information exchange between systems - High Level Data Link Control (HDLC) procedures - Frame structure - Amendment 2: Extended transparency options for start/stop transmission,” International Organization for Standardization, 1992-01-15.
10. Bellcore Specification “SONET Transport Systems: Common Generic Criteria,” GR-253-CORE, Issue 2, Rev. 1, December 1997.
11. ITU-T Recommendation G.707, “Network Node Interface for the Synchronous Digital Hierarchy,” March 1996.
12. ITU-T I.432.1, “Series 1: Integrated Services Digital Network: B-ISDN user-network interface - Physical layer specification: General characteristics,” August 1996.
13. ITU-T I.432.2, “Series 1:Integrated Services Digital Network: B-ISDN user-network interface - Physical layer specification: 155 520 kbit/s and 622 080 kbit/s operation,” August 1996.
14. Bellcore Specification “SONET Dual-Fed Unidirectional Path Switched Ring (UPSR) Equipment Generic Criteria,” GR-1400-CORE, Revision 1, October 1995.

2.0 Pin Assignments and Descriptions

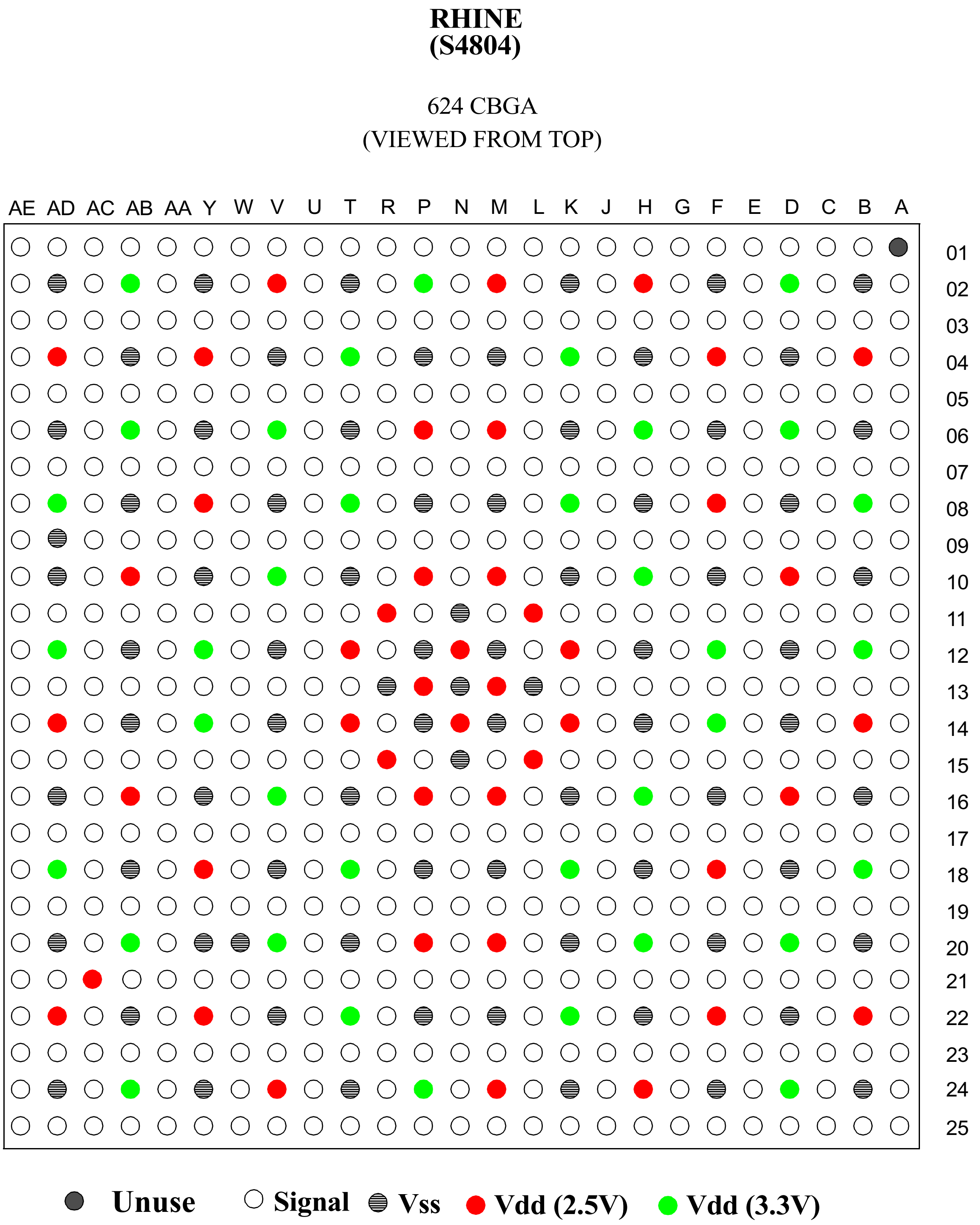


Figure 1: RHINE Pin Assignments

2.1 Pin Assignments

Table 1. SONET Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
155/622 Mb/s Serial I/O's (OC-3/12)			LVPECL	**TX_DATA_[n]†:** Line side transmit output data signals used when Rhine's Line side interface is configured for either STS-12/STM-4 or STS-3/STM-1 (serial) modes. With the line side interface configured in STS-12/STM-4 mode, the TX_DATA[1,5,9,13] outputs are updated on the rising edge of TX_SONETCLK_OUT_155_622. With the line side interface configured in STS-3/STM-1 mode, TX_DATA[1,5,9,13] are updated on the rising edge of TX_SONETCLK_OUT_155_622 while the other TX_DATA_[n] outputs are updated on the rising edge of TX_SONETCLK_OUT_155. **RX_DATA_IN[n]:** Line side receive input data signals used when Rhine's Line side interface is configured in STS-12/STM-4 or STS-3/STM-1 (serial) modes. Data is updated on the rising edge of RX_SONETCLK[n]. **RX_SONETCLK[n]:** 622 MHz clock for STS-12/STM-4 modes; 155 MHz clock for STS-48/STM-16 and STS-3/STM-1 mode. Unused RX_DATA and RX_SONETCLK differential input pairs should be tied off low. This is accomplished by connecting the - terminal to VDD_2.5 via a pull up resistor and the + terminal to GND via a pull down resistor. The resistor values should be chosen to ensure that the differential voltage between the + and - terminals is greater than +/- 200 mV. When calculating the 200mV differential, the internal 100 ohm termination resistor between the +/- terminals must be considered. Unused TX_DATA differential output pairs should be left floating.
TX_DATA[1]-	V21	O		
TX_DATA[1]+	W24	O		
RX_DATA[1]-	U19	I		
RX_DATA[1]+	W21	I		
RX_SONETCLK[1]-	V25	I		
RX_SONETCLK[1]+	U22	I		
TX_DATA[5]-	U24	O		
TX_DATA[5]+	T21	O		
RX_DATA[5]-	R17	I		
RX_DATA[5]+	U21	I		
RX_SONETCLK[5]-	N20	I		
RX_SONETCLK[5]+	N19	I		
TX_DATA[9]-	R25	O		
TX_DATA[9]+	T23	O		
RX_DATA[9]-	R21	I		
RX_DATA[9]+	P25	I		
RX_SONETCLK[9]-	P19	I		
RX_SONETCLK[9]+	P21	I		
TX_DATA[13]-	N23	O		
TX_DATA[13]+	N24	O		
RX_DATA[13]-	N18	I		
RX_DATA[13]+	N21	I		
RX_SONETCLK[13]-	R24	I		
RX_SONETCLK[13]+	N17	I		
155 Mb/s Serial I/O's			LVPECL	
TX_DATA[2]-	M23	O		
TX_DATA[2]+	L23	O		
RX_DATA[2]-	N25	I		
RX_DATA[2]+	N22	I		
RX_SONETCLK[2]-	H23	I		
RX_SONETCLK[2]+	J20	I		
TX_DATA[3]-	L22	O		
TX_DATA[3]+	L25	O		
RX_DATA[3]-	L21	I		
RX_DATA[3]+	K25	I		
RX_SONETCLK[3]-	G24	I		
RX_SONETCLK[3]+	J22	I		
TX_DATA[4]-	H25	O		
TX_DATA[4]+	J23	O		
RX_DATA[4]-	F25	I		
RX_DATA[4]+	K19	I		
RX_SONETCLK[4]-	F23	I		
RX_SONETCLK[4]+	H21	I		

Table 1. SONET Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
155 Mb/s Serial I/O’s (OC-3)			LVPECL	
TX_DATA[6]-	E22	O		
TX_DATA[6]+	D23	O		
RX_DATA[6]-	B25	I		
RX_DATA[6]+	B23	I		
RX_SONETCLK[6]-	F21	I		
RX_SONETCLK[6]+	C24	I		
TX_DATA[7]-	A22	O		
TX_DATA[7]+	B21	O		
RX_DATA[7]-	E24	I		
RX_DATA[7]+	G21	I		
RX_SONETCLK[7]-	G22	I		
RX_SONETCLK[7]+	J19	I		
TX_DATA[8]-	D19	O		
TX_DATA[8]+	E18	O		
RX_DATA[8]-	D17	I		
RX_DATA[8]+	C18	I		
RX_SONETCLK[8]-	C20	I		
RX_SONETCLK[8]+	E19	I		
TX_DATA[10]-	A18	O		
TX_DATA[10]+	H15	O		
RX_DATA[10]-	B19	I		
RX_DATA[10]+	A20	I		
RX_SONETCLK[10]-	B17	I		
RX_SONETCLK[10]+	C17	I		
TX_DATA[11]-	C14	O		
TX_DATA[11]+	A13	O		
RX_DATA[11]-	A16	I		
RX_DATA[11]+	D15	I		
RX_SONETCLK[11]-	A15	I		
RX_SONETCLK[11]+	C16	I		
TX_DATA[12]-	D13	O		
TX_DATA[12]+	C13	O		
RX_DATA[12]-	B13	I		
RX_DATA[12]+	A14	I		
RX_SONETCLK[12]-	E13	I		
RX_SONETCLK[12]+	G13	I		
TX_DATA[14]-	A11	O		
TX_DATA[14]+	E11	O		
RX_DATA[14]-	B11	I		
RX_DATA[14]+	B15	I		
RX_SONETCLK[14]-	C12	I		
RX_SONETCLK[14]+	A12	I		

Table 1. SONET Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
155 Mb/s Serial I/O's (OC-3)				
TX_DATA[15]-	C10	O	LVPECL	
TX_DATA[15]+	A09	O		
RX_DATA[15]-	A08	I		
RX_DATA[15]+	C09	I		
RX_SONETCLK[15]-	F09	I		
RX_SONETCLK[15]+	G10	I		
TX_DATA[16]-	C06	O		
TX_DATA[16]+	E08	O		
RX_DATA[16]-	C08	I		
RX_DATA[16]+	B07	I		
RX_SONETCLK[16]-	B05	I		
RX_SONETCLK[16]+	E07	I		
155 Mb/s Parallel Outputs (OC-48)				
TX_DATA[15]-	V21	O	LVPECL	**TX_DATA_[n]**: Line side transmit output data signals used when Rhine's Line side interface is configured for STS-48/STM-16 (parallel) mode. TX_DATA[15] is the MSB; TX_DATA[0] is the LSB. **Important** See Application Note "System Design Consideration When Using Rhine 3.0 in OC-48 Line Interface Mode" When using Rhine in OC-48 Line Side Interface Mode. This applications note is not required when designing with Rhine 4.0 or later.
TX_DATA[15]+	W24	O		
TX_DATA[14]-	U24	O		
TX_DATA[14]+	T21	O		
TX_DATA[13]-	R25	O		
TX_DATA[13]+	T23	O		
TX_DATA[12]-	N23	O		
TX_DATA[12]+	N24	O		
TX_DATA[11]-	M23	O		
TX_DATA[11]+	L23	O		
TX_DATA[10]-	L22	O		
TX_DATA[10]+	L25	O		
TX_DATA[9]-	H25	O		
TX_DATA[9]+	J23	O		
TX_DATA[8]-	E22	O		
TX_DATA[8]+	D23	O		
TX_DATA[7]-	A22	O		
TX_DATA[7]+	B21	O		
TX_DATA[6]-	D19	O		
TX_DATA[6]+	E18	O		
TX_DATA[5]-	A18	O		
TX_DATA[5]+	H15	O		
TX_DATA[4]-	C14	O		
TX_DATA[4]+	A13	O		
TX_DATA[3]-	D13	O		
TX_DATA[3]+	C13	O		
TX_DATA[2]-	A11	O		
TX_DATA[2]+	E11	O		
TX_DATA[1]-	C10	O		
TX_DATA[1]+	A09	O		
TX_DATA[0]-	C06	O		
TX_DATA[0]+	E08	O		

Table 1. SONET Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
155 Mb/s Parallel Inputs			LVPECL	**RX_DATA_IN[n]:** Line side receive input data signals used when Rhine’s Line side interface is configured in STS-48/STM-16 (parallel) mode. RX_DATA[15] is the MSB; RX_DATA[0] is the LSB. Data is sampled on the rising edge of RX_SONETCLK[2].
RX_DATA[15]-	G24	I		
RX_DATA[15]+	J22	I		
RX_DATA[14]-	L21	I		
RX_DATA[14]+	K25	I		
RX_DATA[13]-	F23	I		
RX_DATA[13]+	H21	I		
RX_DATA[12]-	F25	I		
RX_DATA[12]+	K19	I		
RX_DATA[11]-	F21	I		
RX_DATA[11]+	C24	I		
RX_DATA[10]-	B25	I		
RX_DATA[10]+	B23	I		
RX_DATA[9]-	G22	I		
RX_DATA[9]+	J19	I		
RX_DATA[8]-	E24	I		
RX_DATA[8]+	G21	I		
RX_DATA[7]-	C20	I		
RX_DATA[7]+	E19	I		
RX_DATA[6]-	D17	I		
RX_DATA[6]+	C18	I		
RX_DATA[5]-	B17	I		
RX_DATA[5]+	C17	I		
RX_DATA[4]-	B19	I		
RX_DATA[4]+	A20	I		
RX_DATA[3]-	A15	I		
RX_DATA[3]+	C16	I		
RX_DATA[2]-	A16	I		
RX_DATA[2]+	D15	I		
RX_DATA[1]-	E13	I		
RX_DATA[1]+	G13	I		
RX_DATA[0]-	B13	I		
RX_DATA[0]+	A14	I		
TX_SONETCLK_IN-	R19	I	LVPECL	**TRANSMIT SONET REFERENCE CLOCK INPUT:** Must be 622 MHz for STS-12/STM-4 and for mixed STS12/STM-4 and STS-3/STM1 mode operation; Must be 155 MHz for STS-48/STM-16 and STS-3/STM-1 line modes.
TX_SONETCLK_IN+	U25	I		
TX_SONETCLK_OUT_155-	H13	O	LVPECL	**TRANSMIT SONET CLOCK OUT:** TX_SONETCLK_OUT_155 is the line side transmit clock output . It provides a timing source which is synchronous to the TX_DATA[2:4,6:8,10:12,14:16] outputs in STS-3 mode and TX_DATA[15:0] in STS-48 mode. TX_SONETCLK_155 operates at 155.52 MHz.
TX_SONETCLK_OUT_155+	F13	O		

Table 1. SONET Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
TX_SONETCLK_OUT_155_622- TX_SONETCLK_OUT_155_622+	L19 K23	O O	LVPECL	**TRANSMIT SONET CLOCK OUT:** TX_SONETCLK_OUT_155_622 is the line side transmit clock output . It provides a timing source which is synchronous to the TX_DATA[1,5,9,13] outputs in STS-3/STS-12 line interface modes. TX_SONETCLK_155_ 622 operates at 155.52 MHz if none of the line interfaces are in STS-12 mode; it operates at 622.08 MHz if any of the line interfaces are STS-12/STM-4.
TX_FRAME_IN- TX_FRAME_IN+	W19 AB23	I I	LVPECL	**TRANSMIT FRAMER START-OF-FRAME INDICATION:** This signal is nominally 8 kHz and is used for aligning the frames of multiple RHINE devices (including Rhines connected via the APS port). Should be terminated, if not used, using 470 ohm pull-up on TX_FRAME_IN- to VDD_2.5 and 470 ohm pull-down on TX_FRAME_IN+ to Ground.
RX_REF_CLK_OUT	Y25	O	LVTTL 50 ohm	**REFERENCE CLOCK OUTPUT:** RX_REF_CLK_OUT provides a reference clock signal output that can be generated via various input clock sources to RHINE (see section 6.4).

[a] All unused differential input pairs should be tied off low. This is accomplished by connecting the - terminal to VDD_2.5 via a pull up resistor and the + terminal to GND via a pull down resistor. The resistor values should be chosen to ensure that the differential voltage between the + and - terminals is greater than +/- 200 mV. All differential LVPECL inputs on the RHINE include an internal 100 ohm termination resistor between the + and - terminals. This resistor must be included in the calculation of the external resistor values necessary to ensure a >200mV voltage between the two terminals. Unused differential output pairs should be left floating.

All single ended LVTTL and CMOS inputs should be passively tied off if unused (unless designated with an internal pull-up/down or as a no-connect pin). A 10Kohm pull up/down resistor value is sufficient unless a specific value is specified. Unused signals should be tied off to a inactive logic level.

†. Certain SONET interface data and clock pins support different signal types/numbers when configuring the SONET interface in serial (STS-3/STM-1 or STS-12/STM-4) mode vs. parallel (STS-48/STM-16) mode. For example pin G13 is used as the RX_SONETCLK[12]+ input in STS-3/STM-1mode and as RX_DATA[1]+ in STS-48/STM-16 mode. By assigning a single signal label (which includes both signals names) to a common pin in a Rhine schematic symbol, a common symbol can be used in Rhine designs supporting both serial and parallel SONET side configurations.

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode RX_ADR[4] RX_ADR[3] RX_ADR[2] RX_ADR[1] RX_ADR[0] Quad 8-bit Mode RX_ADR[4][2] RX_ADR[4][1] RX_ADR[4][0] RX_ADR[3][2] RX_ADR[3][1] RX_ADR[3][0] RX_ADR[2][2] RX_ADR[2][1] RX_ADR[2][0] RX_ADR[1][2] RX_ADR[1][1] RX_ADR[1][0]	AA08 AC08 Y07 AE08 V09 AC06 W07 AB07 AE06 AD07 W08 V07 AA08 AC08 Y07 AE08 V09	I	LVTTL	**Receive Port Address.** Driven by the Link Layer device to RHINE to select a specific PHY port. In both direct status and multiplexed status addressing modes, the port address driven on RX_ADR_y, when a falling edge occurs on RX_ENB, is the port that will be selected to transfer data over the RX Flexbus interface. In multiplexed status polling mode, RX_ADR_y is also used to poll ports via the RX_CLAV_PDA_y signal (independent of the state of RX_ENB_y). In Single, 32- bit mode, a port is selected using *RX_ENB_[1] and RX_ADR_[4]* is the MSB. In Quad, 8-bit mode, a port is selcted using *RX_ENB_y and RX_ADR_y_[2]* is the MSB.
Single 32-bit Mode RX_CLAV_PDA Quad 8-bit Mode RX_CLAV_PDA[4] RX_CLAV_PDA[3] RX_CLAV_PDA[2] RX_CLAV_PDA[1]	W09 Y09 AA09 AB09 W09	O	LVTTL 20 ohm	**Receive Cell/Packet available:** Provides an indication of whether a specified number of cells (in ATM mode) or bytes (in POS/direct mapped modes) are available in the RX FIFO for an addressed port.
Single 32-bit Mode RX_CLK Quad 8-bit Mode RX_CLK[4] RX_CLK[3] RX_CLK[2] RX_CLK[1]	AE10 AC10 AA10 W10 AE10	I	LVTTL	**RX Data transfer/synchronization input clock:** Provided by the Link Layer to RHINE for synchronizing transfers on RX_SYS_DAT. RX_CLK_[1] is used for single, 32-bit, bus operation; RX_CLK_[1:4] are used for quad, 8-bit, bus operation (RX_CLK_[1] for RX_SYS_DAT_[31:24].... ...RX_CLK_[4] for RX_SYS_DAT_[7:0]
Single 32-bit Mode RX_CLK_OUT Quad 8-bit Mode RX_CLK_OUT[4] RX_CLK_OUT[3] RX_CLK_OUT[2] RX_CLK_OUT[1]	AE12 AC12 AA12 W12 AE12	O	LVTTL 50 ohm	**RX Data transfer/synchronization output clock:** An gate version of the RX_CLK[N] input signal. Provides a timing source that is synchronous with the Rx FlexBus data and control output signals. Not required by Utopia. **RX_CLK_OUT_INH_[N]** register bit enables/inhibits each RX_CLK_OUT_[N] output signal to reduce power and EMI if the output is not used.

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode RX_DV Quad 8-bit Mode RX_DV[4] RX_DV[3] RX_DV[2] RX_DV[1]	U09 T09 R09 P09 U09	O	LVTTL 20 ohm	**FlexBus-3^{TM}:** Receive data valid signal. When high, RX_DV_y indicates that the receive data output signals current driven on the RX system interface (RX_SYS_DAT, RX_SOC/P_y, RX_EOP_y, RX_LBYTE, RX_PRTY_y and RX_ERR_y) are valid. Since RX_DV_y changes state on the same clk edge as the other Rx system interface outputs, it can be sampled by the Link layer device each time the output signals are latched. This allows the link layer device to qualify the output signals on each clk cycle. RX_DV_[1] provides receive data valid signal in single, 32-bit, bus mode; RX_DV_[1:4] used for quad, 8-bit, bus operation (RX_DV_[1] for RX_SYS_DAT_[31:24]...and RX_DV_[4] for RX_SYS_DAT_[7:0]).
Single 32-bit Mode RX_ENB Quad 8-bit Mode RX_ENB[4] RX_ENB[3] RX_ENB[2] RX_ENB[1]	U10 R10 N10 L10 U10	I	LVTTL	**UTOPIA: Receive Enable**. Provides ATM layer with flow control. Active low signal asserted by the ATM Layer device to indicate that RX_SYS_DAT and RX_SOC will be sampled at the end of the next cycle. When deasserted, data transfer is halted. The address, presented on RX_ADR, the clock cycle immediately preceding the transition of RX_ENB from 1 to 0, becomes the selected address for the data transfer that is initiated by the assertion of RX_ENB. This data transfer commences two clock cycles after the assertion of RX_ENB. **FlexBUS-3^{TM}:** Receive enable. Active-low signal asserted by the Link Layer device to start packet-data transfer from a Receive FIFO. When deasserted, data transfer is halted. The address, presented on RX_ADR, the clock cycle immediately preceding the transition of RX_ENB from 1 to 0, becomes the selected address for the data transfer that is initiated by the assertion of RX_ENB. This data transfer commences two clock cycles after the assertion of RX_ENB. RX_ENB_[1] provides the transfer enable signal in single, 32-bit, bus operation; RX_ENB_[1:4] are used for quad, 8-bit bus operation (RX_ENB_[1] for RX_SYS_DAT_[31:24]...RX_ENB_[4] for RX_SYS_DAT_[7:0]).

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode RX_EOP Quad 8-bit Mode RX_EOP[4] RX_EOP[3] RX_EOP[2] RX_EOP[1]	AE11 AD11 AC11 AB11 AE11	O	LVTTL 20 ohm	**UTOPIA:** Not Used. **FlexBUS-3^{TM}:** Receive end-of-packet. Active-high When *RX_EOP* is high, the last byte (in 4x8 bit mode) or word (32 bit mode) of a packet is being transferred on the *RX_SYS_DAT* bus. To support any number of bytes in a packet, *RX_LBYTE* indicates how many bytes in this last word are valid. *RX_SOC/P* can be high at the same time *RX_EOP* is high for packets less than 5 bytes in size (in 32-bit mode) or 1 byte in size (for 4x8 bit mode). RX_EOP_[1] provides the receive end-of-packet signal for 32-bit bus operation; RX_EOP_[1:4] are used for quad, 8-bit bus operation (RX_EOP_[1] for RX_SYS_DAT_[31:24]...RX_EOP_[4] for RX_SYS_DAT_[7:0]).
Single 32-bit Mode RX_ERR Quad 8-bit Mode RX_ERR[4] RX_ERR[3] RX_ERR[2] RX_ERR[1]	AA11 Y11 W11 V11 AA11	O	LVTTL 20 ohm	**UTOPIA:** Not Used. **FlexBUS-3^{TM}:** Receive error. RX_ERR is asserted simultaneously with RX_EOP to indicate to the link layer device that the packet currently being transferred should be aborted by the link layer device. RX_ERR_[1] provides the receive error signal for 32-bit bus operation; RX_ERR_[1:4] are used for quad, 8-bit bus operation (RX_ERR_[1] for RX_SYS_DAT_[31:24]...RX_ERR_[4] for RX_SYS_DAT_[7:0]).
Single 32-bit Mode RX_LBYTE[1] RX_LBYTE[0]	U11 T11	O	LVTTL 20 ohm	**FlexBus-3^{TM}:** RX_LBYTE_[1:0] is available in single, 32-bit, bus mode only, to provide support for the transfer of packets of any size. It is only valid during the last word of a packet transfer (RX_EOP_[1] high). It provides the number of valid bytes within the final word of a packet transfer. (See Table 40.)

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode RX_PRTY Quad 8-bit Mode RX_PRTY[4] RX_PRTY[3] RX_PRTY[2] RX_PRTY[1]	U05 U06 N06 N09 U05	O	LVTTL 20 ohm	**UTOPIA:** Parity. RX_PRTY is the parity bit over RX_SYS_DAT data bus, which is driven by RHINE. **RX_PRTY_MODE** = 0 indicates odd parity; **RX_PRTY_MODE** = 1 indicates even parity. **FlexBUS-3^{TM}:** RX_PRTY is the parity bit over the entire RX_SYS_DAT data bus, which is driven by RHINE. **RX_PRTY_MODE[N]** = 0 indicates odd parity; **RX_PRTY_MODE[N]** = 1 indicates even parity. RX_PRTY_[1] provides the parity signal for 32-bit bus operation; RX_PRTY_[1:4] are used for quad, 8-bit bus operation (RX_PRTY_[1] for RX_SYS_DAT_[31:24]...RX_PRTY_[4] for RX_SYS_DAT_[7:0]).
Single 32-bit Mode RX_SOC/P Quad 8-bit Mode RX_SOC/P[4] RX_SOC/P[3] RX_SOC/P[2] RX_SOC/P[1]	P11 AE09 R12 U12 P11	O	LVTTL 20 ohm	**UTOPIA:** Start of cell. Active-high signal asserted by RHINE when RX_SYS_DAT contains the first valid bytes of a cell. **FlexBUS-3^{TM}:** Start of packet. Active-high signal asserted by RHINE when RX_SYS_DAT contains the first valid bytes of a packet. RX_SOC/P_[1] provides the start of cell/packet signal for 32-bit bus operation; RX_SOC/P_[1:4] are used for quad, 8-bit bus operation (RX_SOC/P_[1] for RX_SYS_DAT_[31:24]...RX_SOC/P_[4] for RX_SYS_DAT_[7:0]).

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
RX_SYS_DAT[31] RX_SYS_DAT[30] RX_SYS_DAT[29] RX_SYS_DAT[28] RX_SYS_DAT[27] RX_SYS_DAT[26] RX_SYS_DAT[25] RX_SYS_DAT[24] RX_SYS_DAT[23] RX_SYS_DAT[22] RX_SYS_DAT[21] RX_SYS_DAT[20] RX_SYS_DAT[19] RX_SYS_DAT[18] RX_SYS_DAT[17] RX_SYS_DAT[16] RX_SYS_DAT[15] RX_SYS_DAT[14] RX_SYS_DAT[13] RX_SYS_DAT[12] RX_SYS_DAT[11] RX_SYS_DAT[10] RX_SYS_DAT[9] RX_SYS_DAT[8] RX_SYS_DAT[7] RX_SYS_DAT[6] RX_SYS_DAT[5] RX_SYS_DAT[4] RX_SYS_DAT[3] RX_SYS_DAT[2] RX_SYS_DAT[1] RX_SYS_DAT[0]	V01 W01 Y01 AA01 AB01 AC01 AD01 AE01 W02 AA02 AC02 AE02 V03 W03 Y03 AA03 AB03 AC03 AD03 AE03 W04 AA04 AC04 AE04 V05 W05 Y05 AA05 AB05 AD05 W06 AA06	O	LVTTL 20 ohm	**UTOPIA/FlexBus-3^{TM}:** Receive System Data. Four byte-wide, true data driven from RHINE to Link Layer. RX_SYS_DAT_[31] is the MSB. OR Four, 8-bit wide, data buses driven from RHINE to Link Layer. RX_SYS_DAT_[31] and [23] and [15] and [7] are the MSBs.

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode TX_ADR[4] TX_ADR[3] TX_ADR[2] TX_ADR[1] TX_ADR[0] Quad 8-bit Mode TX_ADR[4][2] TX_ADR[4][1] TX_ADR[4][0] TX_ADR[3][2] TX_ADR[3][1] TX_ADR[3][0] TX_ADR[2][2] TX_ADR[2][1] TX_ADR[2][0] TX_ADR[1][2] TX_ADR[1][1] TX_ADR[1][0]	T17 W18 R14 R18 P17 U13 T15 U16 T13 R16 U17 U14 T17 W18 R14 R18 P17	I	LVTTL	**Receive Port Address.** Driven by the Link Layer device to RHINE to select a specific PHY port. In both direct status and multiplexed status addressing modes, the port address driven on TX_ADR_y when a falling edge occurs on TX_ENB is the port that will be selected to transfer data over the TX Flexbus interface. In multiplexed status polling mode, TX_ADR_y is also used to poll ports via the TX_CLAV_PDA_y signal (independent of the state of TX_ENB_y). In SIngle, 32- bit mode, a port is selcted using *TX_ENB_[1] and TX_ADR_[4]* is the MSB. In Quad, 8-bit mode, a port is selcted using *TX_ENB_y and TX_ADR_y_[2]* is the MSB
Single 32-bit Mode TX_CLAV_PDA Quad 8-bit Mode TX_CLAV_PDA[4] TX_CLAV_PDA[3] TX_CLAV_PDA[2] TX_CLAV_PDA[1]	AB19 AC18 AC19 AC20 AB19	O	LVTTL 20 ohm	**Transmit Cell/Packet available:** Provides an indication of whether space is available in the TX FIFO to accept a specified number of cells (in ATM mode) or bytes (in POS/direct mapped modes) for an addressed port.
Single 32-bit Mode TX_CLK Quad 8-bit Mode TX_CLK[4] TX_CLK[3] TX_CLK[2] TX_CLK[1]	AD19 AE18 AE19 AE20 AD19	I	LVTTL	**TX Data transfer/synchronization input clock:** Provided by the Link Layer to RHINE for synchronizing transfers on TX_SYS_DAT. TX_CLK_[1] is used for single, 32-bit, bus operation; TX_CLK_[1:4] are used for quad, 8-bit, bus operation (TX_CLK_[1] for TX_SYS_DAT_[31:24]......TX_CLK_[4] for TX_SYS_DAT_[7:0]
Single 32-bit Mode TX_CLK_OUT Quad 8-bit Mode TX_CLK_OUT[4] TX_CLK_OUT[3] TX_CLK_OUT[2] TX_CLK_OUT[1]	AE24 AA21 AE23 AB21 AE24	O	LVTTL 65 ohm	**TX Data transfer/synchronization output clock:** An gate version of the TX_CLK[N] input signal. Provides a timing source that is synchronous with the Rx FlexBus data and control output signals. Not required by Utopia. **TX_CLK_OUT_INH_[N]** register bit enables/inhibits each TX_CLK_OUT_[N] output signal to reduce power and EMI if the output is not used.

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode TX_ENB Quad 8-bit Mode TX_ENB[4] TX_ENB[3] TX_ENB[2] TX_ENB[1]	Y19 AA18 AA19 AA20 Y19	I	LVTTL	**UTOPIA Transmit Enable**: Provides ATM Layer with flow control. Active low signal asserted by the ATM Layer device during cycles when *TX_SYS_DAT* contains valid data. When deasserted, data transfer is halted. In multi-port bus operation, the address presented on *TX_ADR_y* the clock cycle immediately preceding the transition of *TX_ENB_y* from 1 to 0 becomes the selected address for the data transfer that is initiated by this assertion of *TX_ENB _y.* In both a single-port and multi-port modes, when *TX_ENB_y* is asserted, data is transferred to the selected PHY-layer port. When *TX_ENB_y* is deasserted, *TX_SYS_DAT* is ignored. **FlexBUS-3^{TM} Transmit Enable:** Active-low signal asserted by the Link Layer device to start packet-data transfer to an addressed TX FIFO. When deasserted, data transfer is halted. The address, presented on TX_ADR, the clock cycle immediately preceding the transition of TX_ENB from 1 to 0, becomes the selected address for the data transfer that is initiated by the assertion of TX_ENB. When TX_ENB is asserted, data is transferred to the selected RHINE port. When TX_ENB is deasserted, TX_SYS_DAT is ignored. *TX_ENB_[1]* provides transfer enable indications in a single, 32-bit, bus operation; *TX_ENB_[1:4]* are used for quad, 8-bit, bus operation (*TX_ENB_[1]* for *TX_SYS_DAT_[31:24] ..., TX_ENB_[4]* for *TX_SYS_DAT_[7:0]).*

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode TX_EOP Quad 8-bit Mode TX_EOP[4] TX_EOP[3] TX_EOP[2] TX_EOP[1]	V19 U18 U20 R20 V19	I	LVTTL	**UTOPIA:** Not Used. **FlexBUS-3™: Transmit end-of-packet.** Active-high When TX_EOP is high, the last byte (in 4x8 bit mode) or word (32 bit mode) of a packet is being transferred on the TX_SYS_DAT bus. To support any number of bytes in a packet, TX_LBYTE indicate is provided to indicate which bytes in the last word of the packet transferred are valid. *TX_SOC/P* can be high at the same time *TX_EOP* is high for packets less than 5 bytes in size (in 32-bit mode) or 1 byte in size (for 4x8 bit mode). *TX_EOP_[1]* provides transmit end of packet indication in a single, 32-bit, bus operation; *TX_EOP_[1:4]* are used for quad, 8-bit, bus operation (*TX_EOP_[1]* for *TX_SYS_DAT_[31:24] ..., TX_EOP_[4]* for *TX_SYS_DAT_[7:0]).*
Single 32-bit Mode TX_ERR Quad 8-bit Mode TX_ERR[4] TX_ERR[3] TX_ERR[2] TX_ERR[1]	L16 M17 L17 K17 L16	I	LVTTL	**UTOPIA:** Not Used. **FlexBUS-3™: Transmit error.** *TX_ERR* is asserted by the link layer device to indicate to the Rhine that the packet currently being transferred over the TX system interface should be aborted by the Rhine. If the packet being transferred over the system interface is completely contained within the TX FIFO, then the packet will be deleted by the Rhine. If the packet has started the mapping process into the SONET SPE, then the packet will be aborted. TX_EOP must be asserted simultaneously with TX_ERR for the Rhine to correctly recognize the TX_ERR signal.
Single 32-bit Mode TX_LBYTE[1] TX_LBYTE[0]	AD21 AC22	I	LVTTL	**FlexBus-3™:** TX_LBYTE_[1:0] is available in single, 32-bit, bus mode only, to provide support for the transfer of packets of any size. It is only valid during the last word of a packet transfer (TX_EOP_[1] high). It provides the number of valid bytes within the final word of a packet transfer. (See Table 40.)

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode TX_PRTY Quad 8-bit Mode TX_PRTY[4] TX_PRTY[3] TX_PRTY[2] TX_PRTY[1]	U23 T19 V23 T25 U23	I	LVTTL	**UTOPIA:** Parity over TX_SYS_DAT, driven by the ATM Layer. TX_PRTY_MODE = 0 indicates odd parity; TX_PRTY_MODE = 1 indicates even parity. **FlexBUS-3^{TM}:** TX_PRTY is the parity bit over entire TX_SYS_DAT data bus, which is driven by the Link Layer. **TX_PRTY_MODE[N]** = 0 indicates odd parity; **TX_PRTY_MODE[N]** = 1 indicates even parity. TX_PRTY is considered valid only when TX_ENB is asserted. TX_PRTY_[1] provides the parity signal for 32-bit bus operation; TX_PRTY_[1:4] are used for quad, 8-bit bus operation (TX_PRTY_[1] for TX_SYS_DAT_[31:24]...TX_PRTY_[4] for TX_SYS_DAT_[7:0]).
Single 32-bit Mode TX_SOC/P Quad 8-bit Mode TX_SOC/P[4] TX_SOC/P[3] TX_SOC/P[2] TX_SOC/P[1]	AB25 AC25 Y21 AA22 AB25	I	LVTTL	**UTOPIA:** Start of cell. Active-high signal asserted by the ATM Layer device when TX_SYS_DAT contains the first valid byte of the cell. **FlexBUS-3^{TM}:** Start-of-packet. Active-high signal asserted by the Link Layer device when TX_SYS_DAT contains the first valid byte of the packet. Considered valid only when TX_ENB is asserted. TX_SOC/P_[1] provides the start of cell/packet signal for 32-bit bus operation; TX_SOC/P_[1:4] are used for quad, 8-bit bus operation (TX_SOC/P_[1] for TX_SYS_DAT_[31:24]...TX_SOC/P_[4] for TX_SYS_DAT_[7:0]).

Table 2. FlexBus Interface[a]

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
Single 32-bit Mode		I	LVTTL	**UTOPIA/FlexBus-3^{TM}:** Transmit System Data. Four byte-wide, true data driven from Link Layer to RHINE. TX_SYS_DAT_[31] is the MSB. OR Four, 8-bit wide, data buses driven from Link Layer to RHINE. TX_SYS_DAT_[31] and [23] and [15] and [7] are the MSBs.
TX_SYS_DAT[31]	V13			
TX_SYS_DAT[30]	W13			
TX_SYS_DAT[29]	Y13			
TX_SYS_DAT[28]	AA13			
TX_SYS_DAT[27]	AB13			
TX_SYS_DAT[26]	AC13			
TX_SYS_DAT[25]	AD13			
TX_SYS_DAT[24]	AE13			
TX_SYS_DAT[23]	W14			
TX_SYS_DAT[22]	AA14			
TX_SYS_DAT[21]	AC14			
TX_SYS_DAT[20]	AE14			
TX_SYS_DAT[19]	V15			
TX_SYS_DAT[18]	W15			
TX_SYS_DAT[17]	Y15			
TX_SYS_DAT[16]	AA15			
TX_SYS_DAT[15]	AB15			
TX_SYS_DAT[14]	AC15			
TX_SYS_DAT[13]	AD15			
TX_SYS_DAT[12]	AE15			
TX_SYS_DAT[11]	W16			
TX_SYS_DAT[10]	AA16			
TX_SYS_DAT[9]	AC16			
TX_SYS_DAT[8]	AE16			
TX_SYS_DAT[7]	V17			
TX_SYS_DAT[6]	W17			
TX_SYS_DAT[5]	Y17			
TX_SYS_DAT[4]	AA17			
TX_SYS_DAT[3]	AB17			
TX_SYS_DAT[2]	AC17			
TX_SYS_DAT[1]	AD17			
TX_SYS_DAT[0]	AE17			

[a] All of RHINE's single ended LVTTL and CMOS inputs should be passively tied off if unused (unless designated with an internal pull-up/down or as a no-connect pin). A 10Kohm pull up/down resistor value is sufficient unless a specific value is specified. Unused signals should be tied off to a inactive logic level. All unused LVTTL/CMOS outputs should be left floating

Table 3. Auto Protection Switching Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
APS_DATA_OUT[31]	U01	O	2.5V CMOS 35 ohm	Parallel APS DATA interface. Updated on the rising edge of APS_CLK_OUT. The **APS_OUT_INH** bit can be set to turn off the APS_DATA_OUT and APS_CLK_OUT logic (to reduce power/EMI) if the APS interface is not used.
APS_DATA_OUT[30]	U02	O		
APS_DATA_OUT[29]	U03	O		
APS_DATA_OUT[28]	U04	O		
APS_DATA_OUT[27]	M01	O		
APS_DATA_OUT[26]	M03	O		
APS_DATA_OUT[25]	U07	O		
APS_DATA_OUT[24]	U08	O		
APS_DATA_OUT[23]	T01	O		
APS_DATA_OUT[22]	T03	O		
APS_DATA_OUT[21]	T05	O		
APS_DATA_OUT[20]	T07	O		
APS_DATA_OUT[19]	R01	O		
APS_DATA_OUT[18]	R02	O		
APS_DATA_OUT[17]	R03	O		
APS_DATA_OUT[16]	R04	O		
APS_DATA_OUT[15]	R05	O		
APS_DATA_OUT[14]	R06	O		
APS_DATA_OUT[13]	R07	O		
APS_DATA_OUT[12]	R08	O		
APS_DATA_OUT[11]	P01	O		
APS_DATA_OUT[10]	P03	O		
APS_DATA_OUT[9]	P05	O		
APS_DATA_OUT[8]	P07	O		
APS_DATA_OUT[7]	N01	O		
APS_DATA_OUT[6]	N02	O		
APS_DATA_OUT[5]	N03	O		
APS_DATA_OUT[4]	N04	O		
APS_DATA_OUT[3]	N05	O		
APS_DATA_OUT[2]	M11	O		
APS_DATA_OUT[1]	N07	O		
APS_DATA_OUT[0]	N08	O		
APS_CLK_OUT	F01	O	2.5V CMOS 35 ohm	Reference clock for APS_DATA_OUT operates at 77.76 MHz.

Table 3. Auto Protection Switching Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
APS_DATA_IN[31]	L01	I	2.5V CMOS 3.3V Tolerant	Four byte-wide data input stream for transmit side APS DATA interface. Updated on the rising edge of APS_CLK_IN[N]. APS_DA TA_IN{31:24] is clocked using APS_CLK_IN[1]. APS_DA TA_IN{23:16] is clocked using APS_CLK_IN[2]. APS_DA TA_IN{15:8] is clocked using APS_CLK_IN[3]. APS_DA TA_IN[7:0] is clocked using APS_CLK_IN[4]. **APS_IN_INH** bit in the register map inhibits APS_DATA_IN[N] to provide a means of reducing power consumption if the APS input is not used. As single ended, CMOS inputs, the APS_DATA_IN and APS_CLK_IN inputs should be passively tied off (high or low) if unused.
APS_DATA_IN[30]	L02	I		
APS_DATA_IN[29]	L03	I		
APS_DATA_IN[28]	L04	I		
APS_DATA_IN[27]	L05	I		
APS_DATA_IN[26]	L06	I		
APS_DATA_IN[25]	L07	I		
APS_DATA_IN[24]	L08	I		
APS_DATA_IN[23]	K01	I		
APS_DATA_IN[22]	K03	I		
APS_DATA_IN[21]	K05	I		
APS_DATA_IN[20]	K07	I		
APS_DATA_IN[19]	J01	I		
APS_DATA_IN[18]	J02	I		
APS_DATA_IN[17]	J03	I		
APS_DATA_IN[16]	J04	I		
APS_DATA_IN[15]	J05	I		
APS_DATA_IN[14]	J06	I		
APS_DATA_IN[13]	J07	I		
APS_DATA_IN[12]	J08	I		
APS_DATA_IN[11]	H01	I		
APS_DATA_IN[10]	H03	I		
APS_DATA_IN[9]	H05	I		
APS_DATA_IN[8]	H07	I		
APS_DATA_IN[7]	G01	I		
APS_DATA_IN[6]	G02	I		
APS_DATA_IN[5]	G03	I		
APS_DATA_IN[4]	G04	I		
APS_DATA_IN[3]	G05	I		
APS_DATA_IN[2]	G06	I		
APS_DATA_IN[1]	G07	I		
APS_DATA_IN[0]	G08	I		
APS_CLK_IN[1] APS_CLK_IN[2] APS_CLK_IN[3] APS_CLK_IN[4]	M05 M07 M09 L09	I	2.5V CMOS 3.3V Tolerant	Reference clocks driven by the APS_CLK_OUT of the chip sending the APS_DATA operating at 77.76 MHz.

Table 4. Drop/Insert Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
RX_TOH_DATA	G12	O	2.5V CMOS 65 ohm	Serial-outputs for received TOH/SOH bytes (E1, E2, D1-D12, K1, K2, and F1). Clocked on the falling edge of RX_TOH_CLK.
RX_TOH_CLK	E25	I	2.5V CMOS 3.3V Tolerant	A 19.44-MHz clock reference for RX TOH serial channels.
RX_TOH_FRAME	E23	I	2.5V CMOS 3.3V Tolerant	Byte alignment indication for received TOH channels.

Table 4. Drop/Insert Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
TX_TOH_DATA	E14	I	2.5V CMOS 3.3V Tolerant	Serial input for transmitted TOH/SOH bytes (E1, E2, D1-D12, K1, K2, and F1). Clocked on the rising edge of TX_TOH_CLK.
TX_TOH_CLK	E12	O	2.5V CMOS 65 ohm	A 19.44-MHz clock reference for TX TOHchannels.
TX_TOH_FRAME	G14	O	2.5V CMOS	Byte alignment indication for transmitted TOHchannels.

Table 5. Microprocessor Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
ADDR[13] ADDR[12] ADDR[11] ADDR[10] ADDR[9] ADDR[8] ADDR[7] ADDR[6] ADDR[5] ADDR[4] ADDR[3] ADDR[2] ADDR[1] ADDR[0]	A02 B03 D03 A04 W20 E05 F05 A07 C07 D07 F07 A06 D09 E09	I	LVTTL	**ADDRESS BUS:** Allows host microprocessor to perform register selection within the S4804. Since each register address stores a 8 bit value, the addressing scheme differs depending if the microprocessor is configured in 8-bit, asynchronous mode or 16-bit, synchronous mode. In 8-bit, asynchronous mode, each byte wide register in the register map can be addressed uniquely. In 16-bit, synchronous mode, only even addresses are valid with D[7:0] data bus transferred to the even address (n)and D[15:8] transferred to the odd address (n+1).
APS_INTB	G09	O	LVTTL 65 ohm	**APS INTERRUPT:** Active-low output from the S4804 triggered by an APS event. APS_INTB is an open-drain output that is tri-stated when the RX_APS_INT interrupt register bit is not set, is masked or has been cleared back to 0.
BUSMODE	H09	I	LVTTL	**BUS INTERFACE MODE:** This signal allows the data-transfer operations to be compatible with most microprocessor interfaces. When Busmode=1, data transfer occurs in “Intel mode” (RDB,WRB,RDYB); when Busmode=0, data transfer occurs in “Motorola mode” (DSB, RWB, DTACKB).
CSN	K09	I	LVTTL	**CHIP SELECT:** Active-low chip select to the S4804 used to validate the address bus for read-and-write transfers.

Table 5. Microprocessor Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
D[15] D[14] D[13] D[12] D[11] D[10] D[9] D[8] D[7] D[6] D[5] D[4] D[3] D[2] D[1] D[0]	J14 E15 F15 G15 E16 J15 K15 C15 G16 J16 A17 G17 J17 G18 J18 H19	I/O	LVTTL 65 ohm	**DATA BUS:** Allows transfer of data between host microprocessor and the S4804. Only the lower 8 bits (D[7:0]) are used on asynchronous mode. In synchronous mode, D[7:0] is written to even addresses and D[15:8] is written to odd addresses.
INTB	J09	O	LVTTL 65 ohm	**INTERRUPT:** Active-low output from the S4804 triggered by an event that caused one of the internal interrupts to become activated. INTB is an open-drain output that is tri-stated when the RX/TXSUM_INT interrupt register bits are not set, are masked or has been cleared back to 0.
RDB/DSB	B09	I	LVTTL	Read or data strobe. If Busmode=1, the active low RDB input is low to enable read data from the addressed location on the data bus. If Busmode=0 the active low DSB input is LOW to enable read data from RHINE, or strobe write data into RHINE This signal is only valid for 8-bit asynchronous operation.
RDYB/DTACKB	E10	O	LVTTL 65 ohm	**Ready/Data Acknowledge:** RDYB/DTACKB goes low to acknowledge the end of data transfers over the data bus. The RDYB/DTACKB signal is a tri-stated output and operates the same for both BUSMODE settings.
RSTB	AC23	I	LVTTL	**RESET:** Active-low input to reset the S4804.
SYNCMODE	H11	I	LVTTL	**Sync Mode:** Selects the mode of operation of the Microprocessor interface. SYNCMODE=1 is for the 16-bit synchronous operation; SYNCMODE=0 is for the 8-bit asynchronous operation.
uPCLK	J12	I	LVTTL	**Microprocessor Clock:** Microprocessor system clock. Maximum rate is 50 MHz. This signal must be driven with a valid clock signal in both synchronous and asynchronous processor modes. (See12.13 (page 217) and 12.14 (page 218).

Table 5. Microprocessor Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
WRB/RWB	J13	I	LVTTL	Write/Read or Read/Write. If Busmode=1, the WRB input is low for write and high for read. If Busmode=0, the RWB input defines the access as read if “1” or a write if “0.”

Table 6. GPIO/LOS/ALARM Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
GPIO[7] GPIO[6] GPIO[5] GPIO[4] GPIO[3] GPIO[2] GPIO[1] GPIO[0]	D11 F11 K11 L14 K13 A10 G11 H17	I/O	LVTTL 65 ohm	**GENERAL PURPOSE I/O:** The GPIO register allows the user to define each grouping (GPIO[0,1], GPIO[2,3], GPIO[4,5], GPIO[6,7]) as either input or output bits. These bits can be used for user-defined input control. GPIOCTL1 bit controls GPIO[1:0] GPIOCTL2 bit controls GPIO[3:2] GPIOCTL3 bit controls GPIO[5:4] GPIOCTL4 bit controls GPIO[7:6] Specific restrictions apply when configuring certain GPIO signals as input or outputs (see section 11.1.1.6 for details)
RX_ALARM_OUT[1] RX_ALARM_OUT[2] RX_ALARM_OUT[3] RX_ALARM_OUT[4] RX_ALARM_OUT[5] RX_ALARM_OUT[6] RX_ALARM_OUT[7] RX_ALARM_OUT[8] RX_ALARM_OUT[9] RX_ALARM_OUT[10] RX_ALARM_OUT[11] RX_ALARM_OUT[12] RX_ALARM_OUT[13] RX_ALARM_OUT[14] RX_ALARM_OUT[15] RX_ALARM_OUT[16]	F19 E20 L20 K21 J24 G20 C22 L18 P15 N16 L24 C23 D25 G23 C25 G25	O	LVTTL 65 ohm	**RECEIVE ALARM OUT SIGNAL:** Reports SONET receive alarms, OOF, LOF, LOC, LAIS, and LOS for each line interface. Contribution of each alarm may be inhibited using INH bits within the register map. (addr 220Bh, 224Bh, 228Bh, ... 25CB)
RX_LOSEXT[1] RX_LOSEXT[2] RX_LOSEXT[3] RX_LOSEXT[4] RX_LOSEXT[5] RX_LOSEXT[6] RX_LOSEXT[7] RX_LOSEXT[8] RX_LOSEXT[9] RX_LOSEXT[10] RX_LOSEXT[11] RX_LOSEXT[12] RX_LOSEXT[13] RX_LOSEXT[14] RX_LOSEXT[15] RX_LOSEXT[16]	R22 W22 Y23 J25 J21 E21 A24 A25 F17 E17 C11 J11 J10 E06 E04 U15	I	LVTTL	External Loss of Signal indication to RHINE. The interpretation of the sense of the signal is dependant on RX_LOSEXT_LEVEL_x register bit. Use of these inputs is optional since the RHINE itself supports internal LOS monitoring.

Table 7. Test Interface

Pin Signal Name	Pin #	(I/O)	Signal Type	Signal Description
TCK	A03	I	LVTTL	**TEST CLOCK:** JTAG input clock used to sample data on the TDI and TDO pins. Internal pull-up.
TDI	C01	I	LVTTL	**TEST DATA IN:** Input pin for serial-data stream to be sent to the S4804. TDI is sampled on the rising edge of TCK. Internal pull-up.
TDO	C03	O	LVTTL 65 ohm	**TEST DATA OUT:** Output pin for serial-data stream sent from the S4804. TDO is updated on the falling edge of TCK.
TMS	E01	I	LVTTL	**TEST MODE SELECT:** Controls the operating mode of the JTAG interface. TMS is sampled on the rising edge of TCK. Should be pulled high when JTAG interface is not in use. Internal pull-up.
TRSTB	F03	I	LVTTL	**TEST PORT RESET:** Active-low input used to reset the JTAG interface. Must be pulsed low for >30ns after power up (or tied low via a 220ohm resistor if unused) for the device to operate correctly. Includes a weak internal pull-up.
TRISTATE_EN	B01	I	LVTTL	**TRI-STATE ENABLE:** When TRISTATE_EN is brought high, all output and input/output pins are tri-stated. Internal pull-down.

Table 8. Power, Ground, No Connect Pins

Chip Signal Name	Pin #	Signal Type
VDD_2.5	H02, M02, V02, B04, F04, Y04, AD04, M06, P06, F08, Y08, D10, M10, P10, AB10, L11, R11, K12, N12, T12, M13, P13, B14, K14, N14, T14, AD14, L15, R15, D16, M16, P16, AB16, F18, Y18, M20, P20, B22, F22, Y22, AD22, H24, M24, V24, W23	2.5V power supply
VDD_3.3	D02, P02, AB02, K04, T04, D06, H06, V06, AB06, B08, K08, T08, AD08, H10, V10, B12, F12, Y12, AD12, F14, Y14, H16, V16, B18, K18, T18, AD18, D20, H20, V20, AB20, K22, T22, D24, P24, AB24, AC21	3.3V power supply

Table 8. Power, Ground, No Connect Pins

Chip Signal Name	Pin #	Signal Type
Ground	B02, F02, K02, T02, Y02, AD02, D04, H04, M04, P04, V04, AB04, B06, F06, K06, T06, Y06, AD06, D08, H08, M08, P08, V08, AB08, B10, F10, K10, T10, Y10, AD10, N11, D12, H12, M12, P12, V12, AB12, L13, N13, R13, D14, H14, M14, P14, V14, AB14, N15, B16, F16, K16, T16, Y16, AD16, D18, H18, M18, P18, V18, AB18, B20, F20, K20, T20, Y20, AD20, D22, H22, M22, P22, V22, AB22, B24, F24, K24, T24, Y24, AD24, P23, AA25, AC24, AD09	Ground
NO CONNECT	L12, AC09, AA24, AA07, G19, D21, C02, AE22, AD23 A23, AE05, AC05, C05, C21, A19, E03, AE21, C19, M25, AA23, M15, M19, M21, A21, AC07, AE07, E02, D01, AE25, AD25, A05, R23	Leave disconnected
EXTERNAL PULL-DOWN	W25	Provide 220 ohm pull-down to ground
RESERVED	D05	Provide 470 ohm pull-up to VDD_2.5
RESERVED	C04	Provide 470 ohm pull-down to ground

3.0 Mechanical Packaging Information

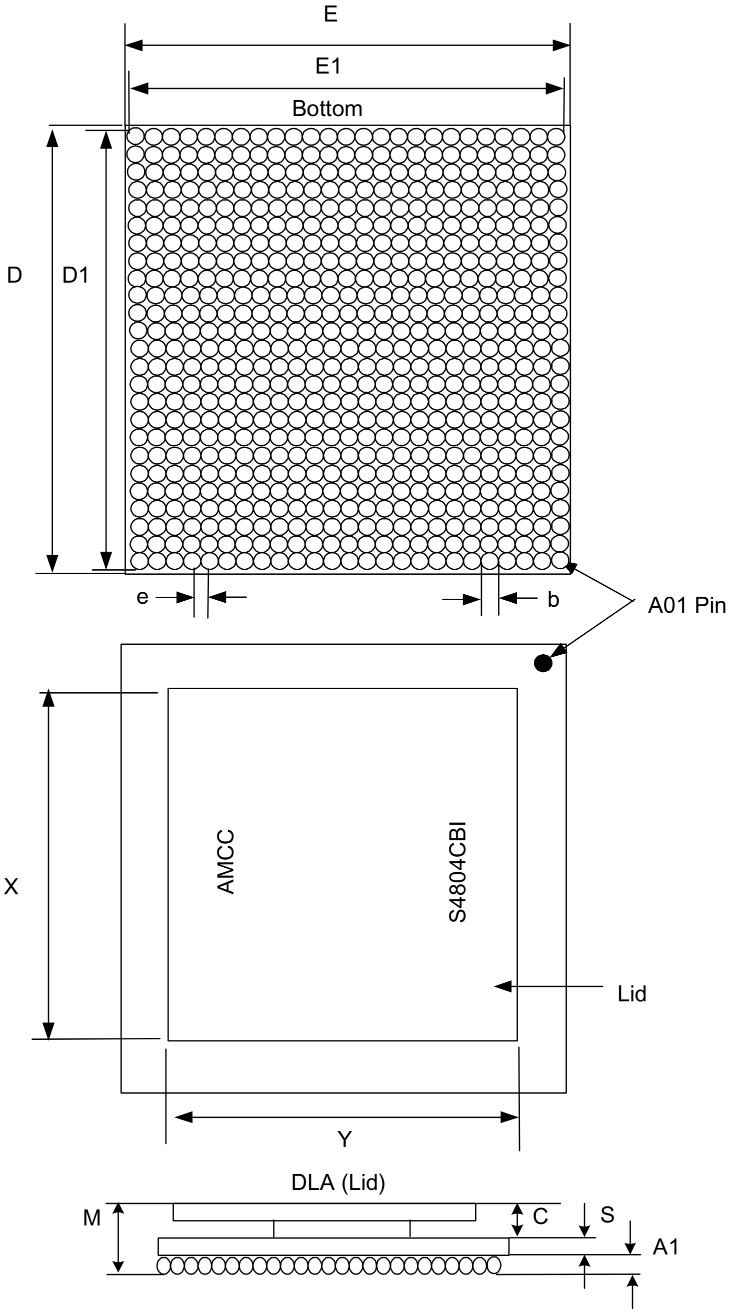


Table 9. Mechanical Package Specifications

Dimension	Min	Nom	Max
A1	0.80	0.90	1.00
C	2.638	2.83	3.022
b	.82	.89	.93
e		1.27	
D	32.3	32.5	32.7
D_1	30.18	30.48	30.68
E	31.3	32.5	32.7
E_1	30.28	30.48	30.68
S		1.65	
M		5.38	5.672
X		30.5	
Y		30.5	
Weight (grams)		10.37	

1. All dimensions are in millimeters.
2. Total module height, M, is the summation of C, S, and A1.
3. No I/O at A01 location.
4. Outline conforms to JEDEC MO-156.

Table 10. Thermal Performance

Theta $_{JC}$(°C/W)	Thermal Performance Theta $_{JA}$(°C/W) @ Airflow					
	0 FPM	100 FPM	200 FPM	300 FPM	400 FPM	600 FPM
.47	11.8	10.6	9.4	8.5	7.7	6.7

4.0 Functional Description

4.1 Common Conventions, Controls, and Configuration

Note: Bytes within the SONET or SDH signal are transmitted by the MSB first. By convention, the MSBs of SONET/SDH bytes are referred to as bit 1 and the LSBs as bit 8; however, the register words that provision and monitor SONET/SDH bytes are defined with bit 7 or 15 as the MSB and bit 0 as the LSB.

The S4804 supports SONET and SDH modes. For brevity, SONET terminology will be used as the main protocol for discussion unless otherwise indicated.

The interface where data flows from the ATM or Link Layer device to the S4804 device is labeled the Transmit System Interface. The interface where data flows from the S4804 to the ATM or Link Layer device is labeled the Receive System Interface.

4.2 Monitors and Control Interface

For performance-monitoring purposes, RHINE contains a number of “delta” bits, “event” bits, “second event” bits, and error counters.

Delta bits are set (to logic “1”) by RHINE when a monitored-parameter changes state. The delta bit then stays high until the controller clears (to logic “0”) the bit by writing a 1 to the bit. If a write-to-1 occurs simultaneously with a parameter state change, the delta bit is set. Delta bits are indicated by a **_D** suffix.

Event bits are similar to delta bits, but they do not have a corresponding status bit. Event bits are set (to logic “1”) by RHINE when the associated event occurs (such as FIFO overflow). The event bit then stays high, regardless of whether or not the event reoccurs, until the controller clears (to logic “0”) the bit by writing a 1 to the bit. If a write to 1 occurs simultaneously with the event occurrence, the event bit is set. Event bits are indicated by a **_E** suffix.

There are several “events” that are monitored for occurrence each second. This allows the controller to accumulate the number of seconds that contain a particular event, for example, the number of seconds that at least 1 error was detected in a received signal by monitoring the B1 bytes. For this purpose, RHINE creates a 1 Hz signal, SEC_EVENT. Alternatively, the timing for “second events” can be controlled by the **LATCH_CNT** register.

When RHINE detects a rising edge of SEC_EVENT (and **CNT_SEC_EN** = 1) or when **LATCH_CNT**, in register 0x2000, is written from a 0 to a 1 (and **CNT_SEC_EN** = 0), it produces a pulse on an internal signal, LATCH_EVENT. When a pulse occurs on LATCH_EVENT, the ***_SECE** register bits are set if their corresponding internal-current, second-event, monitoring-bits are active. Like delta and event bits, the ***_SECE** bits are not cleared until they are written to 1 by the controller. The microprocessor is notified via **LATCH_E** when a pulse occurs on LATCH_EVENT.

All the internal performance-monitoring counter blocks are comprised of a running error counter and a holding register that presents stable results to the controller. The counts in all of the running counters are latched into the hold registers and the running counters cleared when a pulse occurs on LATCH_EVENT. To prevent missing a count that occurs when latching occurs, a counter is set to 1 rather than 0 — if the clear signal is simultaneous with an increment.

After being latched, the results are held to be read by the microprocessor. All internal counters have the ability to store more than the maximum possible count in a 1-second interval for a bit error rate of 10^{-3}. As long as the count values are latched (and the results read) every second, no counts will be lost. In case this doesn’t happen, all the running counters will hold their maximum value rather than roll over to 0 (zero).

Summary delta/event/second event-bits provide a consolidated view of the various individual delta/event/second event-bits, grouped either by function or tributary. Summary delta/event/second events are therefore a function of the other delta/event/second event-bits in the register maps. These summary bits do not behave as their individual counterparts do, in that the summary bits are NOT cleared when written to a 1 by the uP. The summary bits are read-only, and should only clear when all delta/event/second event-bits that contribute to them are cleared.

The summary bits, as well as some delta, event and second event-bits, are “NORed” to form RHINE interrupt outputs: *INTB* and *APS_INTB*. The contribution of any of these bits to the summary interrupts can be deleted by setting the corresponding “mask” bit.

4.3 Configuration and Conventions

RHINE has many configuration and monitoring registers that are accessible through the microprocessor interface. There can be up to 48 different tributaries with an STS-48 SONET interface signal. The terminology used here is based on the “STS-3#/STS-1#” tributary indexing convention defined in GR-253. Pointer processor and pointer interpreter registers are provided on an STS-1 basis; therefore, 48 are required. These are referred to using the Bellcore double indexing, ranging from [1]_[1] to [16]_[3].

Most parameters in RHINE are tied to an STS-3 tributary; they require only 16-separate provisioning and monitoring registers. In the following descriptions, an ‘x’ is used in the names of these registers to indicate the register for a particular SONET/SDH, ATM, or HDLC signal, where the value of ‘x’ ranges from [1] to [16] (the first index in the Bellcore terminology) the second index, a ‘j’ ranges in values from [1] to [3]. The 16 possible data-streams that are input/output via the system interface will be referred to throughout the document as ‘tributaries.’ There are up to 16 ‘tributaries’ that can be supported by RHINE. When configured for STS-12 tributaries, the values of x that correspond to the ‘active’ tributaries are [1], [5], [9], and [13].

As RHINE supports an 8-bit, Utopia Level 3 bus, there are other registers and pins that must be replicated 4 times. In the following descriptions, a ‘**y**’ is used in the names of these registers and pins, where ‘**y**’ ranges from [1] to [4].

When RHINE is configured to interface to 1 STS-48c/STM-16 tributary, only 1 provisioning and/or monitoring register for each parameter (x = 1) is used. In other configurations, other provisioning and monitoring registers can also be unused. In all modes, the unused provisioning registers are “don’t care” and should be left at their default values; the unused monitoring registers should be ignored.

The **TX_LINE_CONFIG_[4:0]** register determines the format and rate of the SONET/SDH output signal. The coding of **TX_LINE_CONFIG_[4:0]** is given in Table 11.

Table 11. TX_LINE_CONFIG_[4:0] Provisioning

TX/RX_LINE_CONFIG_[4:0]	Line-Side Interface Configuration
1 XXXX	1 STS-48/STM-16 signal.
0 1XXX	Quadrant 1 configured for 1STS-12/STM-4 signal.
0 0XXX	Quadrant 1 configured for 4 STS-3c/STM-1 signals.
0 X1XX	Quadrant 2 configured for 1 STS-12/STM-4 signal.
0 X0XX	Quadrant 2 configured for 4 STS-3c/STM-1 signals.
0 XX1X	Quadrant 3 configured for 1 STS-12/STM-4 signal.
0 XX0X	Quadrant 3 configured for 4 STS-3c/STM-1 signals.
0 XXX1	Quadrant 4 configured for 1 STS-12/STM-4 signal.

Table 11. TX_LINE_CONFIG_[4:0] Provisioning

TX/RX_LINE_CONFIG_[4:0]	Line-Side Interface Configuration
0 XXX0	Quadrant 4 configured for 4 STS-3c/STM-1 signals.

TX_LINE_CONFIG_[4] = 1 (the default) instructs RHINE to output 1 STS-48/STM-16 signal in 2 byte-wide format at 155.52 MHz. Setting any **TX_LINE_CONFIG_[3:0]** = 1 places RHINE in STS-12/STM-4 mode for that quadrant, where the device outputs a serial signal at 622.08 MHz. Setting **TX_LINE_CONFIG_[3:0]** = 0 places the corresponding quadrant of RHINE in quad STS-3c/STM-1 mode, where the device outputs 4 serial signals at 155.52 MHz. Setting **TX_LINE_CONFIG_[3:0]** = 1111 places RHINE in STS-12/STM-4 mode for all four quadrants, where the device outputs 4 serial signals at 622.08 MHz. Setting **TX_LINE_CONFIG_[3:0]** = 0000 places RHINE in STS-3c/STM-1 mode, where the device outputs 16 serial signals (4 from each quadrant) at 155.52 MHz.

A comparable register exists on the Receive side, named **RX_LINE_CONFIG_[4:0]**. Its interpretation is identical to the **TX_LINE_CONFIG** register defined in Table 11.

The configuration of the tributaries within the line interface signals is defined by the **TX_CONFIG_[20:0]** register. The contents of **TX_CONFIG_[20:0]** have the following definition:

Table 12. TX_CONFIG_[20:0] Values

TX_CONFIG_[20:0]	Contents of Tributary x
1_xxxx_xxxx_xxxx_xxxx_xxxx	STS-48c/AU-4-16c payload (Tributary index = 1,1)
0_1xxx_xxxx_xxxx_xxxx_xxxx	STS-12c/AU-4-4c payload in first quadrant (Tributary index = 1,1)
0_x1xx_xxxx_xxxx_xxxx_xxxx	STS-12c/AU-4-4c payload in second quadrant (Tributary index = 5,1)
0_xx1x_xxxx_xxxx_xxxx_xxxx	STS-12c/AU-4-4c payload in third quadrant (Tributary index = 9,1)
0_xxx1_xxxx_xxxx_xxxx_xxxx	STS-12c/AU-4-4c payload in fourth quadrant (Tributary index = 13,1)
0_0xxx_1xxx_xxxx_xxxx_xxxx	STS-3c/AU-4 payload in first STS-3 tributary group of first quadrant (Tributary index = 1,1)
0_0xxx_0xxx_xxxx_xxxx_xxxx	STS-3/3xAU-3 payloads in first STS-3 tributary group of first quadrant (Tributary indexes = (1,1), (1,2), (1,3))
0_0xxx_x1xx_xxxx_xxxx_xxxx	STS-3c/AU-4 payload in second STS-3 tributary group of first quadrant (Tributary index = 2,1)
0_0xxx_x0xx_xxxx_xxxx_xxxx	STS-3/3xAU-3 payloads in second STS-3 tributary group of first quadrant (Tributary indexes = (2,1), (2,2), (2,3))
. . .	
0_xxx0_xxxx_xxxx_xxxx_xxx1	STS-3c/AU-4 payload in fourth STS-3 tributary group of fourth quadrant (Tributary index = 16,1)

Table 12. TX_CONFIG_[20:0] Values

TX_CONFIG_[20:0]	Contents of Tributary x
0_xxx0_xxxx_xxxx_xxxx_xxx0	STS-3/3xAU-3 payloads in fourth STS-3 tributary group of fourth quadrant (Tributary indexes = (16,1), (16,2), (16,3))

A comparable register exists on the Receive side, named **RX_PP_CONFIG_[20:0]**. Its interpretation is identical to the **TX_CONFIG** register defined in Table 12.

RHINE cannot terminate STS-3 tributaries; that is, RHINE cannot process an STS-3/STM-1 containing STS-1/AU-3 paths and extract payloads from them. Any received STS-3 tributaries will be discarded unless they are hairpinned back out the SONET Transmit interface.

For STS-12/STM-4 and STS-3c/STM-1 line-side configurations, RHINE can be provisioned for additional functionality. Each STS-12/STM-4 or STS-3/STM-1 line can be enabled/disabled via the management interface. Disabled lines have their circuitry shut down in order to minimize power dissipation. STS-12/STM-4 or STS-3/STM-1 lines are disabled by setting **TX_TRIB_INH_x** = 1. The default is enabled (**TX_TRIB_INH_x** = 0). Identical controls are provided in the receive direction, **RX_TRIB_INH_x**.

4.4 Power Minimization[1]

In general, power dissipation is a key concern for all customers. In order to minimize power consumption in each mode of operation, RHINE disables all unused circuitry. The guidelines used for this disabling are as follows:

- In each mode of operation, RHINE disables all unused circuitry: (that is, for STS-48c/STM-16 mode, RHINE disables all unused framers, TOH/POH monitors, HDLC/ATM processors, and so on).
- In all modes, RHINE powers down all unused tributary logic and I/O. This shut-down is controlled by the user via **TX/RX_TRIB_INH_x**.
- Rhine disables the APS interface inputs and outputs via the APS_IN_INH and APS_OUT_INH bits respectively.
- In configurations in which the path/line switching functionality is enabled, the unused HDLC/ATM processing blocks are disabled.
- RHINE disables any clock and data outputs that are unused in the current configuration.
- The user should not clock any unused clock-inputs, in order to minimize power requirements.
- After **MASTER_RESET** or **RX/TX_PROV_RESET** has been set, RHINE activates all circuitry to be consistent with its default configuration. When RHINE is reconfigured to another operating mode, RHINE activates/deactivates circuitry accordingly.
- After **RX/TX_STATE_RESET** has been set, RHINE will again return to its ‘configured’ state, which is maintained through a **STATE_RESET** (see section 11.1.1.1) and deactivate circuitry accordingly.

[1] Even with the RHINE’s ability to disable unused logic and I/O, all unused inputs (both differential and single ended) must be tied off externally to prevent unstable input conditions (see pin description section)

5.0 Transmit Direction

In the transmit direction, RHINE provides for the insertion of packets, ATM cells, or direct data into the STS/STM SPE of all paths. The operating mode of these tributaries is provisionable through the management interface. The register value **TX_POS_x** = 1 instructs the device to perform POS processing for tributary x. The register value **TX_POS_x** = 0 selects ATM mode for these path(s).

Optionally, RHINE can accept raw data from the system interface and directly map it to the SONET/SDH SPE of local paths. This mode of operation is selected by setting **TX_DIRECT_MAP_x** = 1. The value of **TX_POS_x** is ignored when **TX_DIRECT_MAP_x** = 1. The default value for **TX_DIRECT_MAP_x** and **TX_POS_x** is 0, which places RHINE in ATM mode.

The provisioning of the **TX_LINE_CONFIG**, **TX_POS_x**, and **TX_DIRECT_MAP_x** registers (and their Receive side counterparts **RX_LINE_CONFIG**, **RX_POS_x**, and **RX_DIRECT_MAP_x**) must be set to the same values to ensure that the loopback modes will operate correctly. While it is expected that most applications will configure both the RX and TX side configurations to the same settings, in non-loopback mode, it is possible to configured each side independently. For example, the TX side quadrant 1 tributaries could be configured to map POS traffic into an STS12 while the RX side is configured to de-map 4 STS3Cs containing direct mapped data.

5.1 Transmit FIFO Interface

In ATM mode of operation, the Transmit system interface operates as a Utopia Level 3, compliant interface. In POS mode, the Transmit system interface operates as an Utopia-like interface (FlexBus 3) for packet applications. In the direct mapping mode-of-operation, the Transmit system interface operates as a variant of the packet interface, with the *TX_SOC/P_y*, *TX_EOP_y, TX_LBYTE,* and *TX_ERR_y* signals disabled.

In addition to the 32-bit wide, Utopia Level 3 interface, RHINE supports a derivative of this interface, which provides 4 parallel, 8-bit wide, buses for STS-12 and STS-3 line configurations. To support this mode, there are 4 sets of the following signals: *TX_SOC/P_y, TX_EOP_y, TX_ERR_y, TX_ENB_y, TX_CLK_y,* and *TX_PRTY_y.* Individual instantiations of these signals would apply to ¼ of the data bus: *TX_SYS_DAT_[31:24], [23:16], [15:8],* and *[7:0].* These modes are selectable via the management interface. A detailed description of the system interface is presented in section 9.0

5.1.1 Transmit Data Parity Check

In all 3 modes (packet, ATM, and direct data), RHINE calculates the parity of each word received over the appropriate Transmit system interface data bits *(TX_SYS_DAT),* and compares it to the received parity (*TX_PRTY_y)*. Parity errors are reported to the management interface by setting the **TX_PRTY_ERR_y_E** register to 1. **TX_PRTY_MODE_y** = 0 (the default) indicates that odd parity is used for this calculation. **TX_PRTY_MODE_y** = 1 indicates that even parity is used.

RHINE does not treat a cell/packet received with parity errors as an errored cell/packet. It does not alter the cell/packet but simply notifies the management interface of the parity error.

5.1.2 Transmit FIFO

The Link Layer device provides an interface clock to RHINE for synchronizing all interface transfers. This convention requires RHINE to incorporate rate-matching buffers (namely, FIFOs). FIFOs [1], [5], [9], and [13] handle the tributaries, which depending on the configuration, can carry STS-48c/AU-4-16c or STS-12c/AU-4-4c tributaries. In these configurations, these FIFOs are 1024 octets deep. FIFOs that handle STS-3c/AU-4 tributaries are 256 octets deep.

RHINE also maintains the packet/cell status through the FIFOs.

5.1.2.1 Transmit FIFO Error

In POS and ATM modes, the state of the FIFOs are monitored by RHINE. A FIFO error-condition is declared whenever the following occur: 1) a *TX_SOC/P_y* is received prior to the end of a cell or end of a packet (*TX_EOP_y* indication) or 2) the *TX_ENB_y* is active while the FIFO is full (i.e. a TX FIFO overflow).

For each tributary, RHINE contains an 8-bit, FIFO, error-counter that counts every cell/packet affected by a FIFO error-event. When the performance-monitoring counters are latched, the value of these counters are latched to the **TX_FIFOERR_CNT_x_[7:0]** registers and the FIFO error-counters are cleared. (See section 4.2.)

If there has been at least 1 FIFO error-event since the last rising edge of LATCH_EVENT*,* then the FIFO error event-bit, **TX_FIFOERR_x_SECE**, is set.

In POS mode (**TX_POS_x** = 1), RHINE deletes or aborts the errored packets, depending on the size. (See section 5.1.3.)

In ATM mode (**TX_POS_x** = 0 and **TX_DIRECT_MAP_x** = 0), any ATM cells corrupted by FIFO error-events are deleted.

5.1.3 POS Errored Packet Handling

In POS mode-of-operation, (**TX_POS_x** = 1**),** the following errored, packet-handling procedures are provided:

5.1.3.1 *TX_ERR_y* Link Layer Indication

The Transmit system interface provides a method by which the Link Layer device can indicate to RHINE when a particular packet contains errors, and should be aborted or discarded. (See definition of *TX_ERR_y* in section 9.0.)

For each tributary, RHINE contains an 8-bit, link layer, error-counter that counts every packet received from the Link Layer that is marked as errored. (See *TX_ERR_y,* section 9.0) When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters is latched to the **TX_POS_LLPKT_ERRCNT_x_[7:0]** registers, and the link layer, packet, error-counters are cleared. (See section 4.2.)

If there has been at least 1 link-layer packet-error since the last rising edge of LATCH_EVENT*,* then the link layer, packet, error-event bit, **TX_POS_LLPKT_ERR_x_SECE**, is set.

5.1.3.2 Minimum/Maximum Packet Sizes

RHINE also, as an option, views a packet as being errored and does not transmit it, or aborts it if it violates minimum or maximum packet sizes. The packet sizes refer to the size of the packet — only, and does not include the bytes inserted by RHINE (such as flag sequence, address, control, FIFO underflow, transparency, or the FCS bytes). These minimum and maximum sizes are programmable via the management interface. Register **TX_POS_PMIN_[3:0]** contains the minimum packet size. The default value of this register is 0.

Register **TX_POS_PMAX_[15:0]** contains the maximum packet size. The default value of this register is 0x05DE (RFC 1661, on page 4). The minimum valid size for **TX_POS_PMAX_[15:0]** is 4 bytes. Max packet length checking is unreliable if **TX_POS_PMAX_[15:0]** is set smaller than 4.

RHINE disables/enables minimum and maximum packet-size checking on a per-tributary basis when instructed to through the management interface. If **TX_POS_PMIN_ENB_x** or **TX_POS_PMAX_ENB_x** = 1, packet-abort that is due to a violation of the packet-size restriction is enabled. If = 0 (the default), packet size violations are ignored.

For each tributary, RHINE contains two, 8-bit, error-counters that count every violation of the maximum and minimum packet-size limits. When the performance-monitoring counters are latched, the value of these counters is latched to the **TX_POS_PMIN_ERRCNT_x_[7:0]** and **TX_POS_PMAX_ERRCNT_x_[7:0]** registers, and the packet-size violation-counters are cleared. (See section 4.2.)

If there has been at least 1 packet-size violation-error since the last rising edge of LATCH_EVENT, then the appropriate, packet-size violation, second event-bit, **TX_POS_PMIN_ERR_x_SECE** or **TX_POS_PMAX_ERR_x_SECE**, is set.

5.1.3.3 Errored Packet Abort

RHINE cannot delete packets if the error condition is received or detected after the packet has begun transmission. These packets are therefore aborted. RHINE supports 2 options for aborting an errored packet. The default option is to abort a packet by inserting the abort sequence, 0x7d7e. Reception of this code at the far end should cause the receiver to discard this packet.

As an alternative, RHINE can also abort an errored packet by simply inverting the FCS bytes. The Abort mode is controlled via the management interface. **TX_POS_FCSABRT_ENB_x** = 1 enables the FCS inversion method; **TX_POS_FCSABRT_ENB_x** = 0 (the default) disables it.

5.1.4 System Side Cell/Packet Loopback

For debug purposes, the RHINE provides a test mode called system R/T loopback. When enabled (**SYS_R_TO_T_LOOP_x=1)**, ATM cells or packets within a given tributary (x) are extracted from the incoming RX SONET/SDH frames and processed normally into the RX FIFO. However, instead of the data being transferred from the RX FIFO to the RX system interface, the tributary (x) data in the RX FIFO is rerouted directly to the TX HDLC/ATM processor (where it replaces the tributary (x) data received from the TX System Interface). From this point, the loopback data is mapped normally into outgoing TX SONET/SDH frames.

When **SYS_R_TO_T_LOOP** is 0, the loopback is inhibited and independent TX and RX operation is performed.

NOTE: Since the SYS_R_TO_T_LOOP test mode passes data directly from the RX FIFO to the TX HDLC processor, there is no flow control provided to prevent the RX FIFO from underflowing when the data bytes extracted from the RX SONET SPE encompasses multiple rows of the SONET frame. Therefore, the use of system T/R test loopback is limited to a specific packet size range of 20-60 bytes with a inter-packet gap of >12 bytes. This system T/R loopback limitation is only applicable to POS mode since in ATM mode, the TX ATM Cell processor will not map cells until a complete cell has been taken from the RX FIFO.

An alternative R/T loopback scheme, without the above packet size and packet gap liminations, can be performed using the RHINE’s hairpin feature described in section 5.7.3

5.2 Transmit HDLC Processing (POS)

Following the Transmit system interface, RHINE performs the following processing when in POS mode (**TX_POS_x** = 1).

RHINE performs a subset of this processing when in direct mapping mode (**TX_DIRECT_MAP_x** = 1). Specifically, in direct mapping mode RHINE optionally performs the transparency processing in order to support the insertion of the FIFO, underflow, stuff-byte into the data stream.

5.2.1 Transmit Valid Packet Count

In POS mode (**TX_POS_x** = 1), RHINE contains 16 packet-counters that count every valid packet that is transmitted at the line interface. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **TX_POS_PKT_CNT_x_[22:0]** registers, and the packet counters are cleared. (See section 4.2.)

5.2.2 Pre-HDLC Scrambling

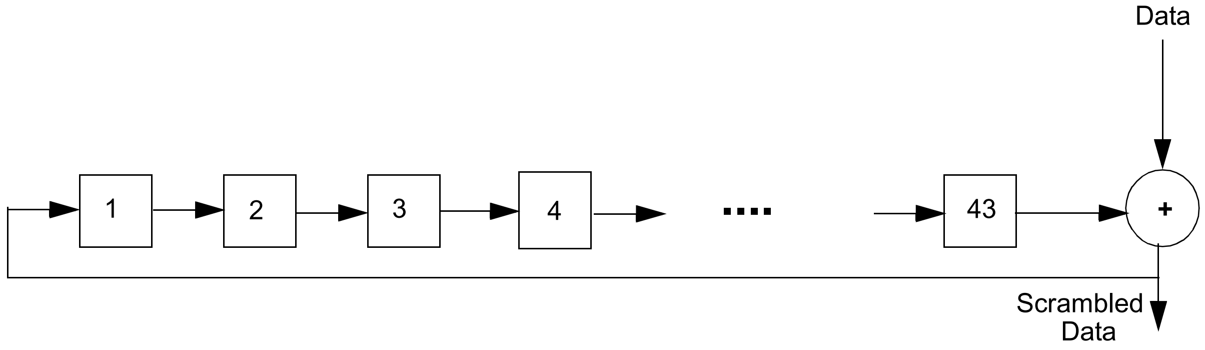
For both direct mapping and POS modes, prior to HDLC processing, the data can be scrambled using a self-synchronizing X^{43} + 1 scrambler. Registers **TX_PREHDLC_SCR_INH_x** control the operation of the 16 pre-HDLC scramblers. When **TX_PREHDLC_SCR_INH_x** = 0, the scrambler for tributary x is enabled. When **TX_PREHDLC_SCR_INH_x** = 1 (the default), operation of the scrambler is inhibited. The scrambling operates on the data in big-endian bit-order. The payload scrambler (section 5.4) has priority over the pre-HDLC scrambler and must be disabled if the pre-HDLC scrambler is enabled.

RHINE provides 16 pre-HDLC, self-synchronizing scramblers based on the following generator polynomial:

$$X^{43} + 1$$

This scrambler is illustrated in the following figure:

Figure 2. X^{43} + 1 Scrambler



In POS mode (**TX_POS_x** = 1), this scrambler scrambles the entire, incoming, packet stream. In Direct Data mode (**TX_DIRECT_MAP_x** = 1), this scrambler randomizes the entire, incoming, direct, data stream.

5.2.3 Encapsulation of Packets in HDLC Frame

The HDLC frame defined for Packet-Over-Sonet (POS) applications is illustrated in Figure 3. In packet mode (**TX_POS_x** = 1), each packet received from the Link Layer is delineated using the Flag Sequence defined in RFC 1662, which is used to indicate both the beginning and end of an HDLC frame. The value of this Flag Sequence is 01111110 (hexadecimal 0x7e).

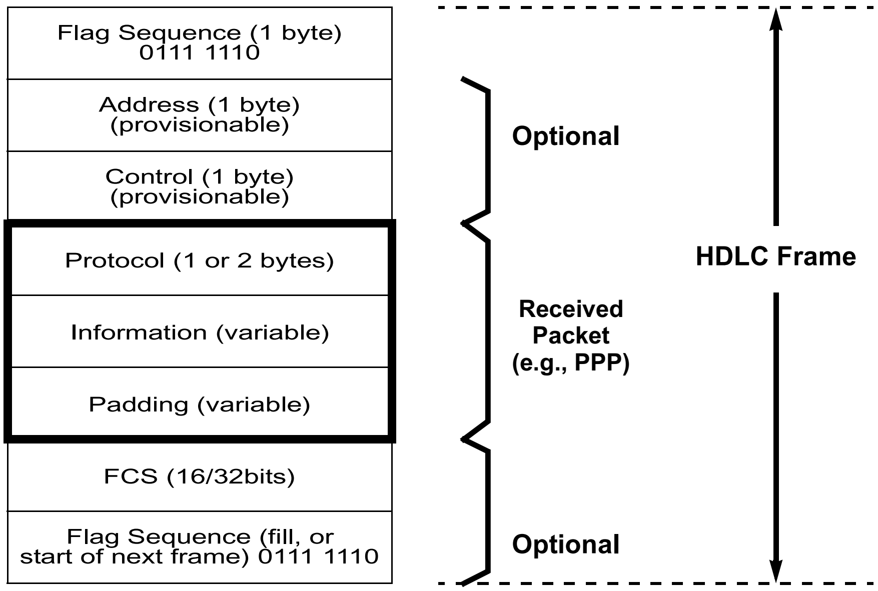

Flag Sequence (1 byte) 0111 1110
Address (1 byte) (provisionable)
Control (1 byte) (provisionable)
Protocol (1 or 2 bytes)
Information (variable)
Padding (variable)
FCS (16/32bits)
Flag Sequence (fill, or start of next frame) 0111 1110
Optional
Received Packet (e.g., PPP)
Optional
HDLC Frame

Figure 3. HDLC Encapsulation for Packet over SONET

As an option, RHINE can insert a single flag to indicate both the end of one frame and the start of the following frame. This is controlled via the management interface; if **TX_POS_EOP_FLAG_x** = 1, RHINE inserts separate flags in tributary x to indicate the start-of-frame and end-of-frame. If **TX_POS_EOP_FLAG_x** = 0 (the default), only 1 Flag Sequence can be inserted.

For STS-48c payloads, if both the FCS and address/control removal are inhibited, reception of a continuous stream of either 1- or 2-byte packets with only a single flag between these bytes (resulting in an average of more than one packet per 78 MHz clock cycle) will result in packets being periodically dropped.

5.2.4 Address and Control Fields

The HDLC POS standards specify 2 fields immediately following the start-of-frame Flag Sequence: an Address byte and a Control byte. The values of both of these bytes are provisionable on a per-tributary basis via registers **TX_POS_ADDRESS_x_[7:0]** and **TX_POS_CONTROL_x_[7:0]**. In POS mode (**TX_POS_x** = 1), RHINE will optionally insert these fields in tributary x, if **TX_POS_ADRCTL_INS_x** = 1. It will not insert these fields if **TX_POS_ADRCTL_INS_x** = 0 (the default).

5.2.5 Frame Check Sequence (FCS) Field

In POS mode (**TX_POS_x** = 1) as an option, an FCS field is then calculated and inserted at the end of each frame. This option is controlled by register **TX_POS_FCS_INH**. A value of **TX_POS_FCS_INH** = 0 (the default) enables the FCS in all tributaries. A value of **TX_POS_FCS_INH** = 1 disables it.

Two types of FCS fields have been defined in RFC 1662, a 16-bit check sequence (FCS-16) and a 32-bit check sequence (FCS-32). The device supports both types. **TX_POS_FCS_MODE_x** = 0 places the device in FCS-32 mode and is the default. **TX_POS_FCS_MODE_x** = 1 places the device in FCS-16 mode.

RHINE provides FCS-16 functionality, using the following generator polynomial:

$$1 + X^{5} + X^{12} + X^{16}$$

RHINE provides FCS-32 functionality, using the following generator polynomial:

$$1 + X + X^{2} + X^{4} + X^{5} + X^{7} + X^{8} + X^{10} + X^{11} + X^{12} + X^{16} + X^{22} + X^{23} + X^{26} + X^{32}$$

The FCS field is calculated over all bits of the original packet, as well as the Address and Control fields. It does not include the Flag Sequence, nor the FCS field itself. It also does not include any added FIFO underflow bytes. (See section 5.2.7.3.)

If **TX_POS_FCS_BIT_ORDR** = 0, the packet data is read into the FCS shift register in big-endian bit order (MSB first). If **TX_POS_FCS_BIT_ORDR** = 1 (the default), the packet data is read into the shift register in little-endian bit order (LSB first). In either case, the packet data is restored to big-endian order for processing after the FCS calculation.

The FCS is inverted and then transmitted (most significant octet first), which contains the coefficient of the highest term. If **TX_POS_FCS_BIT_ORDR** = 0, the FCS is transmitted in big-endian order. If **TX_POS_FCS_BIT_ORDR** = 1, the FCS is transmitted in little-endian order.

Figure 4 illustrates the CRC calculation for the FCS-16 functionality:

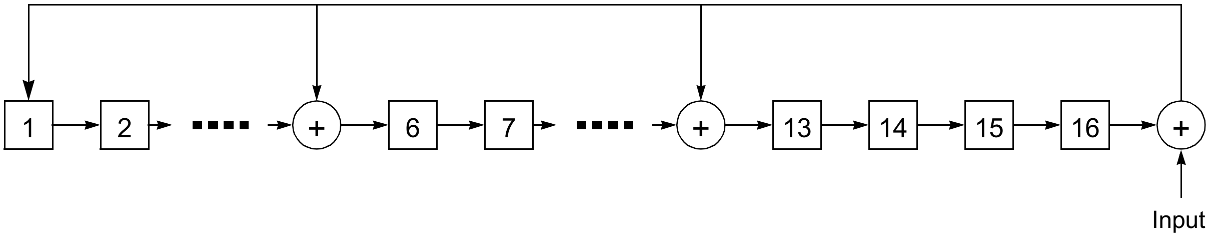


Figure 4. CRC Calculation for FCS-16 Computation

In the preceding figure, all feedback contributions are shown: the shift register inputs that are not shown come directly from the preceding shift register. The location of the addition operators between the shift registers and the feedback are determined by the coefficients of the FCS polynomial. A coefficient of 0 indicates that there is no feedback contribution to the input of the corresponding flip-flop in the shift register.

The shift register implementation for the FCS-32 functionality is similar, with location of the feedback contribution to the shift register input determined by the FCS-32 generator polynomial. For the FCS-32, addition operators are located above registers 2, 3, 5, 6, 8, 9, 11, 12, 13, 17, 23, 24, and 27.

The FCS calculation begins by initializing the shift register with all 1s. The data over which the FCS calculation is to be performed is loaded into the shift register from the bottom right. After the last data bit has entered the shift register, the contents of the shift register contain the 16/32-bit FCS, with the MSB in register 16 or 32.

5.2.6 Transparency

5.2.6.1 POS mode

In the POS mode (**TX_POS_x** = 1, **TX_DIRECT_MAP_x** = 0), an octet stuffing procedure is performed at this point, which is referred to as Transparency Processing. A specific octet, Control Escape (01111101 or hexadecimal 0x7d) is used as a marker to indicate bytes that will require specific processing at the Receive side. Control Escape is used to mark any occurrence of specific codes (see the following table) in the frame data.

After FCS computation, RHINE examines the entire frame between any 2 Flag Sequences. Each occurrence of any code identified in Table 13 is replaced by a 2-octet sequence consisting of the Control Escape octet followed by the original octet exclusive-or’d with hexadecimal 0x20.

RHINE performs transparency processing on the following byte sequences. Occurrences of 0x7e in the payload (between Flag Sequences) are processed as described.

Table 13. Octet Values Handled by Transparency Processing

Octet Value (Hex) or Register	Name
0x7e	Flag Sequence
0x7d	Control Escape
TX_POS_FIFOUNDER_BYTE_ [7:0]	FIFO Underflow

As an example:

0x7e is encoded as 0x7d, 0x5e

0x7d is encoded as 0x7d, 0x5d

The transparency byte stuffing for the FIFO, underflow, byte code is controlled on a per-tributary basis by the **TX_POS_FIFOUNDER_MODE_x** register. (See section 5.2.7.3.) If **TX_POS_FIFOUNDER_MODE_x** = 1, the FIFO byte code **TX_POS_FIFOUNDER_BYTE_[7:0]** is inserted during periods of FIFO underflow, and the transparency byte stuffing for the FIFO byte code is enabled. If **TX_POS_FIFOUNDER_MODE_x** = 0, the FIFO byte code is not inserted and the byte stuffing of the FIFO byte code is not performed.

5.2.6.2 Direct Map Mode

In the direct mapping mode (**TX_DIRECT_MAP_x** = 1), transparency byte stuffing on the Flag Sequence (0x7e) is not performed. If **TX_POS_FIFOUNDER_MODE_x** = 1, transparency byte stuffing is performed on tributary x on the Control Escape (0x7d) and FIFO underflow code (**TX_POS_FIFOUNDER_BYTE_[7:0]**). If **TX_POS_FIFOUNDER_MODE_x** = 0, all transparency processing within tributary x is disabled.

5.2.7 SPE Creation

5.2.7.1 Packet Operation (TX_POS_x = 1)

The packet stream is then mapped to the payload of the SONET/SDH Synchronous Payload Envelope (SPE). The packet octet boundaries are aligned with the SPE octet boundaries. As packet frames are variable in length, they are allowed to cross SPE boundaries.

When, during operation, there are no HDLC frames available for immediate insertion into the SPE, the Flag Sequence is transmitted to fill the time between HDLC frames. This is done only between complete frames. (See section 5.2.7.3 for the case where a FIFO underflow occurs prior to the end of a packet.)

The available information rate for Packet over SONET/SDH for STS-3c/STM-1 is 149.760 Mbps, which is the SPE rate with section, line, and path overhead removed. The available information rate for Packet over SONET/SDH for STS-12c/STM-4 (622.080 Mb/s) applications is 599.040 Mbps; for STS-48c/STM-16 applications, the available information rate is 2396.16 Mbps.

5.2.7.2 Direct Data Mapping (TX_DIRECT_MAP_x = 1)

The direct data is mapped into the payload of the SONET/SDH Synchronous Payload Envelope (SPE). The data octet boundaries are aligned with the SPE octet boundaries.

In direct data mapping applications, the Link Layer device is responsible for insuring there is enough data in RHINE to fill the SONET/SDH SPE. (See section 5.2.7.3 for the case where a FIFO underflow occurs in direct mapping mode.)

5.2.7.3 FIFO Underflow

In POS mode (**TX_POS_x** = 1), the transmit FIFO will become empty as a matter of course between packets, but should not become empty during a packet transmission: that is, after a *TX_SOC/P_y* indication has been received but before a *TX_EOP_y* indication has been received. If it does, RHINE provides 2 options for handling FIFO underflow: the packet can be aborted, using the abort methods described in section 5.1.3.3; or a special code can be transmitted, **TX_POS_FIFOUNDER_BYTE_[7:0],** filling the SPE until valid data is once again in the FIFO. Register **TX_POS_FIFOUNDER_MODE_x** controls the response on a per-tributary basis; **TX_POS_FIFOUNDER_MODE_x** = 0 indicates that the packet will be aborted. This is the default value. **TX_POS_FIFOUNDER_MODE_x** = 1 indicates that the special FIFO underflow code, **TX_POS_FIFOUNDER_BYTE_[7:0]** will be transmitted while the underflow condition exists. **TX_POS_FIFOUNDER_BYTE_[7:0]** defaults to 0x50.

In direct mapping mode, (**TX_DIRECT_MAP_x** = 1), the **TX_POS_FIFOUNDER_MODE_x** and **TX_POS_FIFOUNDER_BYTE_[7:0]** can be enabled as well. If **TX_POS_FIFOUNDER_MODE_x** = 0, RHINE does nothing except report FIFO underflow events; it is the responsibility of the user to ensure that the correct amount of data is delivered to RHINE in a timely manner. If **TX_POS_FIFOUNDER_MODE_x** = 1, RHINE will insert the **TX_POS_FIFOUNDER_BYTE_[7:0]** into the SONET/SDH SPE until valid data is once again present in the FIFO.

RHINE contains an 8-bit FIFO underflow counter for each tributary that counts every packet affected by a FIFO underflow event (in the direct mapping case, each new FIFO underflow event is counted). When the performance-monitoring counters are latched, the value of these counters are latched to the **TX_POS_FIFOUNDER_ERRCNT_x_[7:0]** registers, and the FIFO, underflow, error counters are cleared. (See section 4.2.)

If there has been at least 1 FIFO underflow event since the last rising edge of LATCH_EVENT*,* then the FIFO, underflow, error event bit, **TX_POS_FIFOUNDER_ERR_x_SECE**, is set.

5.3 Transmit ATM Processing

5.3.1 Transmit Data HEC Check

In ATM mode (**TX_POS_x** = 0), RHINE calculates the HEC value across the ATM cell header (the first 4 octets) of the transmit data. The HEC calculation follows the procedure used in section 5.3.4, including the use of the **TX_ATM_HEC_ENH** register. HEC errors are reported to the management interface on a per-tributary basis by setting the **TX_ATM_HEC_ERR_x_E** event bit to 1. This HEC check can be inhibited by setting **TX_ATM_UTP_HEC_INH** = 1.

Register **TX_ATM_HEC_UDF_[1:0]** defines the location of the ATM HEC within the User Defined bytes in the Utopia ATM data structures. (See section 9.4.1.)

5.3.2 Transmit Valid Cell Count

In ATM mode (**TX_POS_x** = 0), RHINE contains, for every tributary, an ATM cell counter that counts every ATM cell (including idle cells) received from the ATM Layer that appears at the Utopia interface. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **TX_ATM_CELL_CNT_x_[22:0]** registers, and the ATM cell counters are cleared. (See section 4.2.)

5.3.3 SPE Payload Creation

5.3.3.1 ATM Cells into STS-3c/AU-4

The bit rate available for the ATM cells (user information cells, signalling cells, OAM cells, unassigned cells, and cells used for cell rate decoupling, excluding SONET overhead packets) is 149.760 Mb/s.

For STS-3c/AU-4 signals, RHINE maps the ATM cell stream to a C-4 container (VC-4 without the POH column i.e.,. the SPE minus the POH and stuff columns). The ATM cell boundaries are aligned with the STS-3c/STM-1 octet boundaries. Since the C-4 capacity (2340 octets) is not an integer multiple of the cell length (53 octets), a cell can cross a C-4 boundary.

5.3.3.2 ATM Cells into STS-12c/AU-4-4c

The bit rate available for the ATM cells (user information cells, signalling cells, OAM cells, unassigned cells, and cells used for cell rate decoupling, excluding SONET overhead) is 599.040 Mb/s.

For STS-12c/AU-4-4c signals, RHINE maps the local ATM cell streams to C-4-4c containers (VC-4-4c without the POH and fixed stuff columns i.e.,. the SPE minus the POH and stuff columns). The ATM cell boundaries are aligned with the STS-12c/STM-4 octet boundaries. Since the C-4-4c capacity (9360 octets) is not an integer multiple of the cell length (53 octets), a cell can cross a C-4-4c boundary.

5.3.3.3 ATM Cells into STS-48c/AU-4-16c

The bit rate available for the ATM cells (user information cells, signalling cells, OAM cells, unassigned cells, and cells used for cell rate decoupling, excluding SONET overhead) is 2396.16 Mb/s.

For STS-48c/AU-4-16c signals, RHINE maps the local ATM cell stream to a C-4-16c container (VC-4-16c without the POH and fixed stuff columns i.e.,. the SPE minus the POH and stuff columns). The ATM cell boundaries are aligned with the STS-48c/STM-16 octet boundaries. Since the C-4-16c capacity (37440 octets) is not an integer multiple of the cell length (53 octets), a cell can cross a C-4-16c boundary.

5.3.3.4 Idle Cell Stuffing

ATM, idle, cell stuffing is used when it is necessary to match the rate of ATM cell stream with the C-4, C-4-4c, or C-4-16c.

An ATM idle cell is defined as follows:

Table 14. Pattern for Default Idle Cell

	Octet 1	Octet 2	Octet 3	Octet 4	Octet 5
Header pattern	00000000	00000000	00000000	00000001	HEC = valid code 01010010
Note 1 - The default content of the information field is “01101010” repeated 48 times. Note 2 - There is no significance to any of these individual header fields from the point of view of the ATM layer, as idle cells are not passed to the ATM layer.					

RHINE provides user programmable fields within the idle cell. The format of an ATM cell header across the UNI is illustrated in the following table:

Table 15. ATM Cell Header Format

ATM Cell Header							
7	6	5	4	3	2	1	0
GFC (Generic Flow Control)				VPI (Virtual Path Identifier)			
VPI				VCI(Virtual Channel Identifier)			
VCI							
VCI				PTI (Payload Type Indicator)			CLP (Cell Loss Priority)
HEC (Header Error Control)							

RHINE allows the user to provision the GFC, PTI, and CLP fields of the idle cells via registers **TX_ATM_IDLE_GFC_[3:0]**, **TX_ATM_IDLE_PTI_[2:0],** and **TX_ATM_IDLE_CLP**. The default values for these fields are all 0s for GFC and PTI, and 1 for CLP. (See Table 14.) All other fields within the first 4 bytes of the header are set to 0. A valid HEC field is calculated for the idle cell, according to section 5.3.4 including the final stage XOR with 0x55.

RHINE also allows the user to provision the contents of the 48 information octets of the idle cell via register **TX_ATM_IDLE_DATA_[7:0]**. The default value of this register is 01101010.

5.3.4 Header Error Control (HEC) Sequence Generation

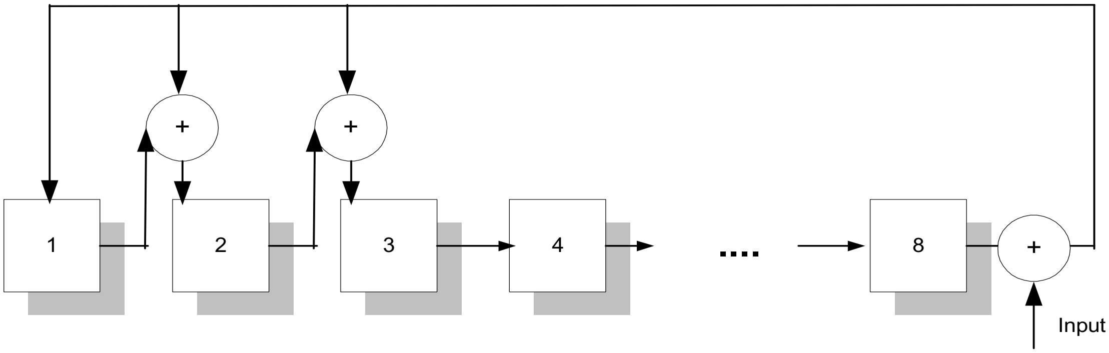
RHINE now optionally calculates the HEC value across the entire ATM cell header (the first 4 octets) and inserts the result in the appropriate header field. This option is controlled by register **TX_ATM_HEC_INH**. A value of **TX_ATM_HEC_INH** = 0 enables the HEC in all tributaries. A value of **TX_ATM_HEC_INH** = 1 disables it. When disabled, RHINE passes through the HEC byte received from the ATM Layer.

A CRC-8 calculation is used to produce the HEC octet, using the following generator polynomial:

$$X^8 + X^2 + X + 1$$

Figure 5 illustrates the CRC calculation.

Figure 5. HEC CRC Calculation



The CRC calculation begins by initializing the shift register with all 0s. The location of the addition operators between the shift registers and the feedback are determined by the coefficients of the CRC-8 polynomial. All addition operators are illustrated in the figure. The shift registers that are not shown all feed directly into the following register, with no feedback contribution. The data over which the CRC calculation is to be performed is loaded into the shift register from the bottom right. After the last data bit has entered the shift register, the contents of the shift register contain the 8-bit HEC, with the MSB in register 8.

As another configurable option to improve the cell delineation process, RHINE supports the addition (modulo 2) of the bit pattern 01010101 to the 8-bit HEC before being inserted into the last octet of the header. This option is controlled by register **TX_ATM_HEC_ENH**. A value of **TX_ATM_HEC_ENH** = 1 enables the modulo 2 addition of the alternating bit pattern (0x55) to the HEC. A value of **0** disables it. The default is 1.

5.4 Scrambling

After HDLC or ATM processing, the data can be scrambled using a self-synchronizing $X^{43} + 1$ scrambler. In all modes, register **TX_SCR_INH_x** controls the operation of the scrambler on a per-tributary basis. When **TX_SCR_INH_x** = 0 (the default), the scrambler is enabled. When **TX_SCR_INH_x** = 1, operation of the scrambler is inhibited. This payload scrambler cannot be enabled simultaneously with the pre-HDLC scrambler. If the payload scrambler is enabled, the pre-HDLC scrambler is disabled, regardless of the setting of the **TX_PREHDLC_SCR_INH_x** bit.

RHINE provides a self-synchronizing scrambler based on the following generator polynomial:

$$X^{43} + 1$$

This scrambler is illustrated in Figure 2.

5.4.1 ATM Scrambler Operation

In ATM mode (**TX_POS_x** = 0), the operation of the scrambler adheres to the following requirements:

- The scrambler randomizes the bits of the information fields, only.
- During the 5-octet header, the scrambler operation is suspended and the scrambler state retained.

5.4.2 HDLC and Direct Map Scrambler Operation

In POS mode (**TX_POS_x** = 1), the scrambler scrambles the whole SPE payload, including the FCS and the flags.

In POS and Direct Map modes (**TX_POS_x** = 1 or **TX_DIRECT_MAP_x** = 1), the scrambler operates on the C-4-Xc, and therefore does not scramble the POH and fixed stuff bytes. The result is scrambling of the entire SPE payload after HDLC processing, including the FCS and interframe fill flags.

5.5 POH Insert and SPE/VC Generation

At this point in the transmission flow, packets, ATM cells, or direct data have been encapsulated into 1 or more SONET/SDH SPE/VCs. From this point on, the 3 modes operate identically.

The Synchronous Payload Envelope/Virtual Container (SPE/VC) Generation block multiplexes bytes from the system interface with Path Overhead (POH) bytes that it generates to create the SPE for SONET or VC for SDH.

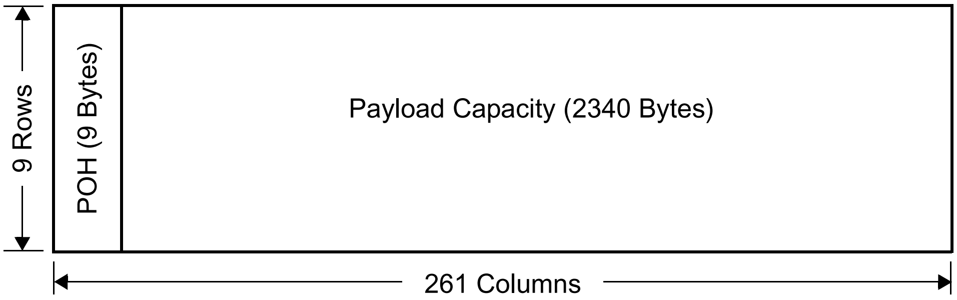
5.5.1 SPE/VC Structure

The first column of the SPE/VC is the POH. The ordering of these 9 bytes is shown below for SONET and SDH.

SONET POH:	SDH POH:
J1	J1
B3	B3
C2	C2
G1	G1
F2	F2
H4	H4
Z3	F3
Z4	K3
Z5	N1

The structure of the STS-3c SPE or VC-4 is shown in Figure 8.

Figure 6. STS-3c SPE or VC-4 Structure



There can be up to 16 SPE/VCs of this type, and each has its own set of POH bytes. The POH bytes, which are defined in section 5.5.2, are provisioned using the set of provisioning bytes indexed by [1] through [16]. The indexes are in ascending order: the first STS-3c/STM-1 is provisioned with the bytes indexed by [1].

The structure of the STS-12c SPE or VC-4-4c is shown in Figure 7.

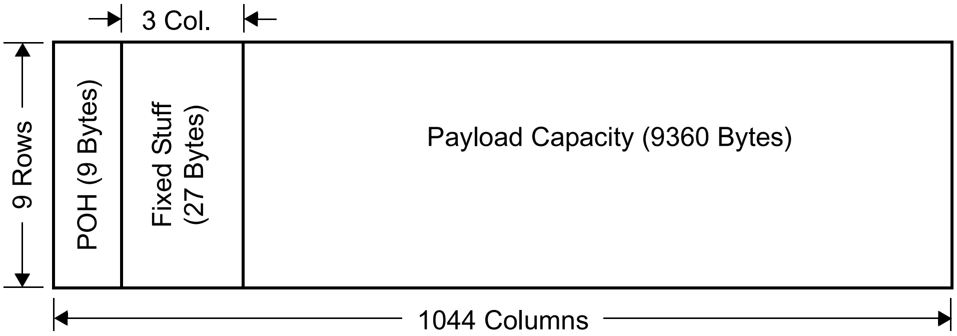


Figure 7. STS-12c SPE or VC-4-4c Structure

There can be up to 4 SPE/VCs of this type, and each has its own set of POH bytes. The POH bytes are provisioned using the set of provisioning bytes indexed by [1], [5], [9], and [13]. The indexes are in ascending order: the first STS-12c/STM-4 is provisioned with the bytes indexed by [1].

The structure of the STS-48c SPE or VC-4-16c is shown in Figure 8.

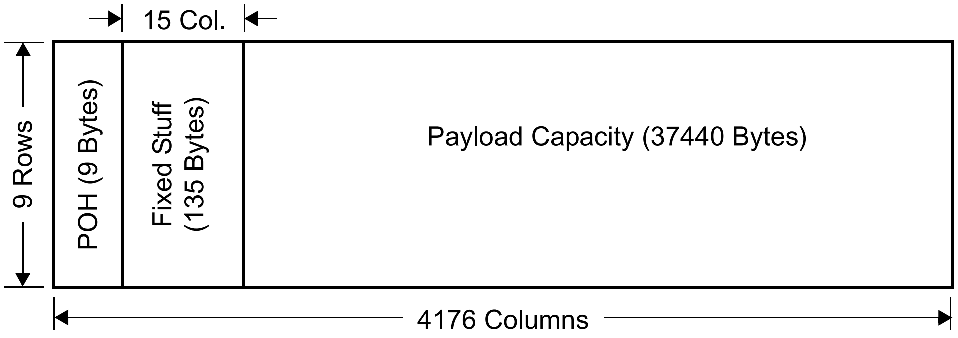


Figure 8. STS-48c SPE or VC-4-16c Structure

The single POH required for STS-48c/VC-4-16c operation is provisioned using index [1].

5.5.2 POH

There are 9 bytes of path overhead. The first byte of the path overhead is the path-trace byte, J1. Its location with respect to the SONET/SDH TOH/SOH is indicated by the associated STS/AU pointer. (See section 5.5.5.) The following sections define the transmitted values of the POH bytes. Where the byte names differ between SONET and SDH, the SONET name will be listed first.

5.5.2.1 Path Trace (J1)

RHINE provides registers through which the user can program either a 16-byte or a 64-byte, path-trace message in the J1 byte of each tributary. The path-trace message is provisioned slightly differently, depending on the mode (SONET vs. SDH) and the tributary type.

5.5.2.1.1 SDH Mode

If **TX_J1SEL_x** = 0 (SDH mode), the J1 byte is transmitted repetitively as the 16-byte sequence in **TX_J1_x_[15]_[7:0]** down to **TX_J1_x_[0]_[7:0]**. This is true for all tributary types (AU-4-16c to AU-4).

5.5.2.1.2 SONET Mode - STS-48c Operation

If **TX_J1SEL_[1]** = 1 and **TX_CONFIG_[20]** = 1, RHINE concatenates 4 path-trace messages to obtain a 64-byte path-trace. The 64-byte message is stored in registers **TX_J1_[1:4]_[15:0]_[7:0].** The transmission order is **TX_J1_[1]_[15]_[7:0]** first and **TX_J1_[4]_[0]_[7:0]** last.

5.5.2.1.3 SONET Mode - STS-12c Operation

If **TX_J1SEL_x** = 1 (for x = 1, 5, 9, and 13) and the corresponding **TX_CONFIG_[19:16]** = 1, RHINE concatenates four, 16-byte, path-trace messages to obtain a 64-byte path trace for the designated tributary. For STS-12c/AU-4-4c tributary x (according to the SONET convention, x can be [1], [5], [9], [13]), the 64-byte message is stored in registers **TX_J1_[x:x+3]_[15:0]_[7:0].** The transmission order is **TX_J1_[x+3]_[15]_[7:0]** first and **TX_J1_[x]_[0]_[7:0]** last.

5.5.2.1.4 SONET Mode - STS-3c Operation

If **TX_J1SEL_x** = 1 and **TX_CONFIG_[15:0]** = 1, RHINE provides a slightly different mechanism for storing the path-trace message. This method allows the user to provision a path-trace message for each of the up to 16 STS-3c tributaries, in which the majority of the path-trace message is a consistent value, common across all tributaries, and only a portion of the full 64-byte message is variable on a tributary-by-tributary basis. RHINE provides a set of registers to hold the common portion of the message, **TX_J1_COMMON_[47:0]_[7:0]**. The variable portion of the 64-byte trace message for each tributary is stored in **TX_J1_x_[15:0]_[7:0]**. The sequence of bytes transmitted for a specific STS-3c is (here, a + represents concatenation):

TX_J1_x_[15:0]_[7:0] + TX_J1_COMMON_[47:0]_[7:0]

5.5.2.2 Path BIP-8 (B3)

The Bit Interleaved Parity 8 (BIP-8) is transmitted as even parity (normal) if **B3_INV_x** = 0. Otherwise, odd parity (incorrect) is generated. The BIP-8 is calculated over all bits of the previous SPE/VC (including the POH) before scrambling and is then placed in the B3 byte from the current SPE/VC before scrambling.

By definition of BIP-8, the first B3 bit provides parity over the first bit from all bytes of the previous SPE/VC; the second B3 bit provides parity over the second bit of all bytes of the previous SPE/VC, and so on.

5.5.2.3 Signal Label (C2)

The signal label byte indicates the composition of the SPE/VC. The provisioned value, **TX_C2_x_[7:0]**, is inserted into the generated C2 bytes.

5.5.2.4 Path Status (G1)

Path REI. The Receive side monitors B3 bit errors in the received SPE/VC. (See section 6.11.2.) The number of B3 errors detected in each frame (0 to 8) is transferred from the Receive side to the Transmit side for insertion into the transmit path status byte, G1, as a Remote Error Indication.

If **FORCE_G1ERR_x** = 1, the 4 MSBs of G1 will continuously be transmitted as 1000 (for testing purposes). Else if **PREI_INH_x** = 0, they are set to the binary value (0000 through 1000, indicating between 0 and 8) equal to the number of B3 errors most recently detected by the Receive side POH monitoring block. Else if **PREI_INH** = 1, they are set to all 0s.

Path RDI. Bit 5 of G1 can be used as a Path/AU Remote Defect Indication (RDI-P), or bits 5, 6, and 7 of G1 can be used as an enhanced RDI-P indicator. The values transmitted in bits 5, 6, and 7 of G1 are taken from a variety of sources, according to the following tableIn certain modes, the Trace Identifier Mismatch (TIM) indication contributes to the enhanced PRDI indication. The TIM condition is detected externally to RHINE by the user software. RHINE provides a **RX_TIM_x** register for each tributary to allow the user to notify RHINE when a TIM condition has occurred. RHINE can then include this term in its automatic PRDI calculation. If TIM contribution is not desired, the user should set the **RX_TIM_x** register to 0.

Table 16. Path RDI Provisioning

PRDI_AUTO_x	PRDI_ENH_x	Source/Format of Bits 5, 6, and 7 of G1
0	X	TX_G1_x_[2:0].
1	0	One-bit RDI, generated from internal RX parameters.
1	1	Three-bit PRDI, generated from internal parameters.

The values transmitted in bits 5, 6, and 7 of G1 are shown in Table . The source of the RX side parameters used to drive the Path RDI bit values is the RX side of this RHINE . .

Table 17. Path RDI Bit Values

PRDI_ AUTO_x	PRDI_ ENH_ x	RX_PI_PAIS_ x || RX_PI_LOP_x	(RX_UNEQ_x && TX_UNEQ_PRDI_x) || RX_TIM_x	(RX_PLM_x && TX_PLM_PRDI_x) || RX_ATM_LCD_x && TX_ATM_LCD_PRDI_x	G1 Bits 5, 6, & 7 (for Tributary x)
0	x	x	x	x	TX_G1_x_[2:0]
1	0	1	x	x	**100**
		0	1	x	**100**
		0	0	x	000
	1	1	x	x	**101**
		0	1	x	110
		0	0	1	010
		0	0	0	001

In Table VII. 1/G.707, PLM-P is not a trigger condition for PRDI; however, in T1X1.5/99-036, a contribution to T1X1 that is attempting to align the definition of PRDI for SONET and SDH, PLM is a trigger condition for PRDI. Therefore, the **TX_PLM_PRDI_x** registers have been defined to support both options. If **TX_PLM_PRDI_x** = 0 (the default), PLM does not contribute to PRDI in tributary x. If **TX_PLM_PRDI_x** = 1, PLM is considered a Remote Payload Defect and contributes to PRDI. In an analogous manner, the **TX_UNEQ_PRDI_x** and **TX_ATM_LCD_PRDI_x** registers have been defined to support both options for the UNEQ and LCD contributions to PRDI in tributary x. If **TX_UNEQ_PRDI_x** = 1, UNEQ is considered a Remote Payload Defect, equivalent to a TIM condition and contributes to PRDI.

If **PRDI_AUTO_x** = 1, the values shown above are transmitted for a minimum of 20 frames. Once 20 frames have been transmitted with the same value, the value corresponding to the current state of the defect indication values listed in Table will be transmitted. If defect persists for less than 23 frames it is transmitted for a minimum of 20 frames. If it persists beyond 22 frames, it is transmitted in increments of 21 frames. That is, if the defect is received for less than 44 frames it would be transmitted for 20+21 = 41 frames, if received for less than 65 frames it would be transmitted for 20+21+21 = 62 frames. This 21 frame timer is located on the Transmit side of RHINE. See Table 16 for **PRD1_AUTO_x**=0.

Bit 8 of G1 (the LSB) is unused and is set to 0.

5.5.2.5 Other POH Bytes

The remaining POH bytes are not supported by RHINE and are transmitted as fixed all 0s bytes. These include the path user channel (F2), the position indicator (H4), the path growth/user channel (Z3/F3), the path growth/path APS channel (Z4/K3), and the tandem connection monitoring (Z5/N1) bytes.

5.5.3 Unequipped Generation

Unless AIS is active, unequipped SPE/VC for tributary x (all SPE/VC bytes are filled with all 0s) is generated if **TX_UNEQ_x** = 1.

5.5.4 PAIS Generation

Normal generation of SONET/SDH payload is suspended during transmission of the Path Administrative Unit (AU) AIS signals: PAIS. PAIS generation is controlled by the **TX_PAIS_x** register.

If **TX_PAIS_x** = 1, the entire payload of the applicable tributary (determined by **TX_LINE_CONFIG_[4:0]** and **TX_CONFIG_[20:0]**) is filled with all-1s bytes.

5.5.5 Pointer Bytes (H1, H2) and Pointer Action Byte (H3)

The H1 and H2 bytes contain 3 fields, as is shown in Figure 9.

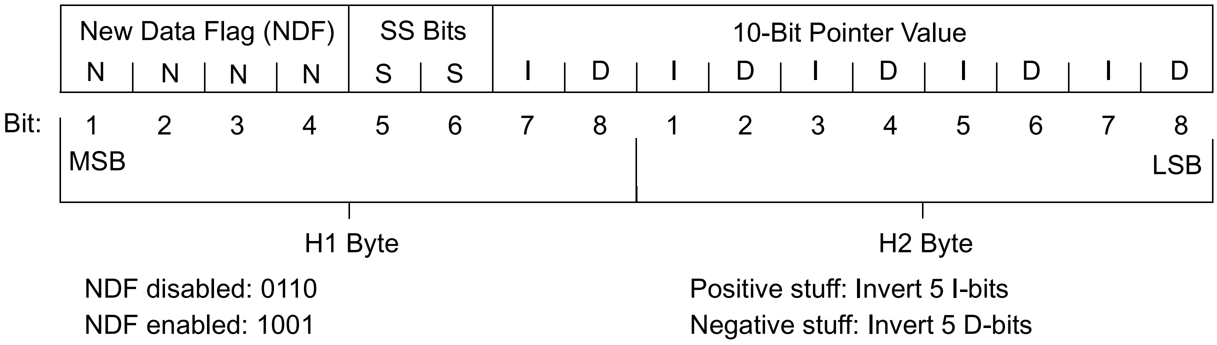


Figure 9: Pointer Byte Fields

Because the SPE/VC is generated synchronously with the TOH, variable pointer generation is not required. Instead, active H1 and H2 bytes are generated with the fixed pointer value of 522 (decimal) = 10_0000_1010 (binary), and the H3 bytes are fixed at all 0s. Thus, the J1 byte of the SPE/VC for tributary 1 is placed in column 145 of row 1 of the SONET/SDH frame for STS-48/STM-16 signals. It is placed in column 37 of row 1 of the SONET/SDH frame for quad STS-12/STM-4 operation; it is placed in column 10 of row 1 of the SONET/SDH frame for STS-3c/STM-1 operation.

5.5.5.1 AIS-P Generation

If **TX_PAIS_x** is active, the corresponding H1, H2, and H3 bytes are transmitted as all 1s. When **TX_PAIS_x** transitions to 0, RHINE transmits the first H1 byte in the next frame for the corresponding tributary with an enabled New Data Flag (NDF). Succeeding frames are generated with the NDF field disabled in the first H1 byte.

5.5.5.2 Non-AIS Generation

The first H1-H2 byte pair of the AU-4, AU-4-4c or AU-4-16c is transmitted as a normal pointer, with

- NDF = 0110
- SS = **TX_SDH_PG_x**, 0
- Pointer Value = 10_0000_1010

The remaining H1-H2 byte pairs in the AU are transmitted as concatenation indication bytes, with

- NDF =1001
- SS = **TX_SDH_PG_x**, 0
- Pointer Value = 11_1111_1111

5.6 Transmit Multiplexer

At this point, RHINE multiplexes the transmit signals into a single STS-48 format (see Figure 10), to facilitate the APS switching.

5.7 Transmit APS Port and Selector

5.7.1 Dual-Feed to APS Output Port

RHINE supports APS configurations through an APS interface. In the transmit processing path, the entire transmit SPE(s) is dual fed from this point to the APS Output Selector block (7.1), as well as to the Transmit APS Selector block.

5.7.2 Transmit APS Selector

To support APS switching between 2 RHINE devices, a selector is provided at this point in the transmit processing. This selector can select data to be input to the TOH/SOH Generation block from one of 2 possible sources; the Payload Generation block of this RHINE or the APS Input interface. The selection is controlled on a per-tributary basis by register **TX_APS_SEL_x**. If **TX_APS_SEL_x** = 0, the data coming from the Payload Generation block of this RHINE is selected (the default). If **TX_APS_SEL_x** = 1, the data coming from the APS Input interface is selected. Note that in order for APS to operate properly between 2 RHINE devices, they must have their frames aligned. This can be achieved by supplying each with a common *TX_FRAME_IN.*

For STS-48/STM-16 operation (**TX/RX_LINE_CONFIG[4]** = 1), all 16 **TX_APS_SEL_x** registers must be written to switch the entire signal. For STS-12/STM-4 configurations (**TX/RX_LINE_CONFIG[3:0]** = 1111), all 4 **TX_APS_SEL_x** registers corresponding to each STS-12/STM-4 tributary must be written.

5.7.3 Internal Hairpin Selector

To support a hairpin turn from the RX to TX direction within a single RHINE device, a second selector is provided just after the Transmit APS Selector in the transmit processing. This selector can select data to be input to the TOH/SOH Generation block from 1 of 2 possible sources: the output of the Transmit APS Selector of this RHINE or the output of the Receive Pointer Processor block. The selection is controlled on a per-tributary basis by register **TX_APS_DXC_SEL_x**. If **TX_APS_DXC_SEL_x** = 0, the data coming from the Transmit APS Selector block of this RHINE is selected (the default). If **TX_APS_DXC_SEL_x** = 1, the data coming from the Receive Pointer Processor block is selected.

5.8 Transmit Demultiplexer

At this point, RHINE demultiplexes the transmit signals into the formats defined by the **TX_LINE_CONFIG_[4:0]** registers.

5.9 SONET/SDH Frame Generation

If **TX_LINE_CONFIG_[4]** = 1, the SONET/SDH Frame Generation block creates an STS-48/STM-16 signal, by multiplexing the designated tributaries from the SPE/VC Generation blocks. If **TX_LINE_CONFIG_[4:0]** = 01111, the SONET/SDH Frame Generation block creates a STS-12/STM-4 signal for each quadrant. If **TX_LINE_CONFIG_[4:0]** = 00000, the SONET/SDH Frame Generation block creates 4 STS-3c/STM-1 signals for each quadrant.

The Frame Generation block then generates the Transport (Section) Overhead (TOH/SOH) bytes and scrambles all bytes of the SONET/SDH signal(s) except for the first row of TOH/SOH bytes. The structure of an STS-48/STM-16 is shown in Figure 10.

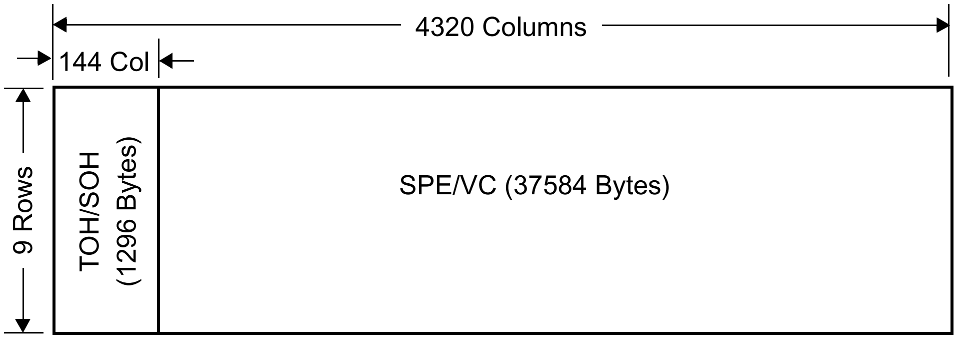


Figure 10: STS-48/STM-16 Structure

In Figure 10, the first 144 columns of each row are shown as the TOH or SOH. This is not strictly true for SDH, because the first 144 columns of the fourth row of SDH frames is not considered part of the SOH. Instead, the Administrative Unit (AU) pointer bytes in the fourth row of the TOH/SOH are grouped with the VC to form an AU-4, AU-4-4c, or AU-4-16c.

The structure of an STS-12/STM-4 is shown in Figure 11.

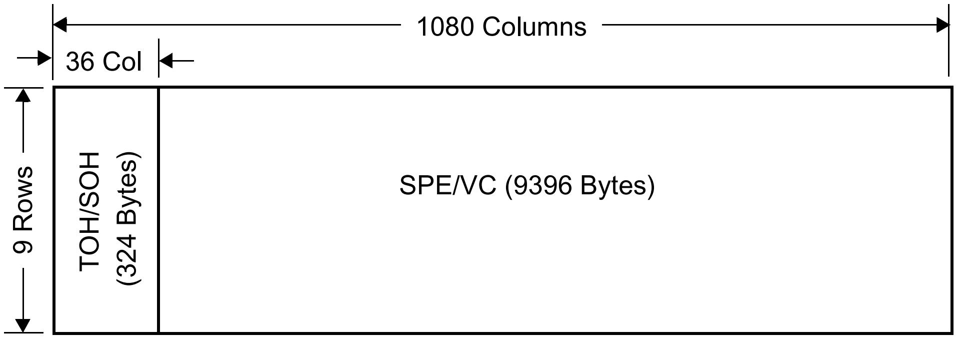


Figure 11: STS-12/STM-4 Structure

In Figure 11, the first 36 columns of each row are shown as the TOH or SOH.

The structure of an STS-3c/STM-1 is shown in Figure 12.

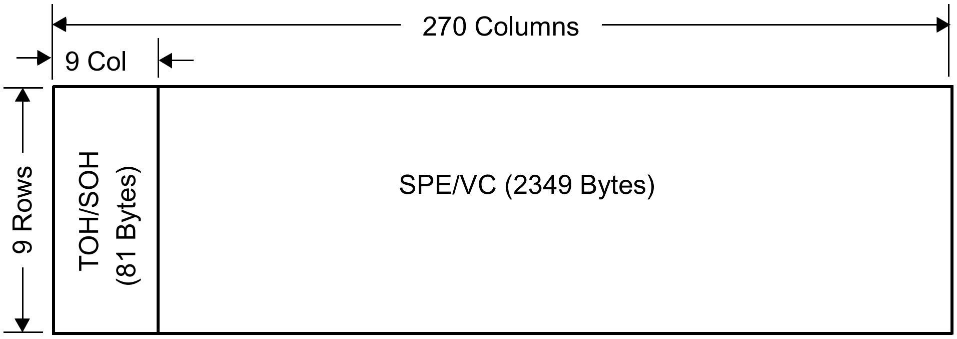


Figure 12: STS-3c/STM-1 Structure

In Figure 12, the first 9 columns of each row are shown as the TOH or SOH.

5.9.1 Frame Alignment

Rhine's output may be frame-synchronized with an external frame sync input provided on *TX_FRAME_IN*.

There is the potential for timing ambiguities associated with the exact position of the *TX_FRAME_IN* input as it is retimed to RHINE's internal clock. Therefore, the period of the rising edge of the *TX_FRAME_IN* input may not be precisely 125 µs each and every period. RHINE will not rely on the position of the *TX_FRAME_IN* input on a frame-by-frame basis, but will instead 'resync' its Frame Generation circuitry to a single rising edge of *TX_FRAME_IN* when instructed to by the user. This resynchronization to the *TX_FRAME_IN* will occur when register **SYS_SYNC_IN_RESYNC** is written from 0 to 1. When this bit is written from 0 to 1, RHINE will resynchronize its internal frame position generator to the next rising edge of *TX_FRAME_IN*; RHINE's frame generator will then flywheel on this frame position, regardless of the *TX_FRAME_IN* signal, until instructed once more by the user to resync again to *TX_FRAME_IN*. The window of uncertainty for the TX_DATA coming out is 4 bytes.

If using RHINE in APS configurations in which the APS interface is used to transfer APS data between 2 RHINEs, a common frame sync must be supplied to both RHINEs via **TX_FRAME_IN**.

When **SYS_SYNC_OUT_EN** = 1 and *GPIO0* is programmed as an output (see section 11.1.1.6), an output sync pulse is provided on *GPIO0*. This output sync pulse follows RHINE internal frame synchronization described above and may be used as a master frame sync to another slaved RHINE. In this mode, the frame sync output on *GPIO0* is aligned within one 78MHz clock cycle of *TX_FRAME_IN*.

5.9.2 Payload Generation

The SONET or SDH payload is normally filled with bytes from the SPE/VC. The J1 byte of the SPE/VC is placed in column 10 of row 1 of the SONET/SDH frame when RHINE is creating an STS-3c/STM-1 (**TX_LINE_CONFIG_[3:0]** = 0000).The J1 byte of the SPE/VC is placed in column 37 of row 1 of the SONET/SDH frame when RHINE is creating an STS-12/STM-4 (**TX_LINE_CONFIG_[3:0]** = 1111). The J1 byte of the SPE/VC is placed in column 145 of row 1 in STS-48/STM-16 mode (**TX_LINE_CONFIG_[4]** = 1).

5.9.2.1 AIS Generation

Normal generation of SONET/SDH payload is suspended during transmission of the Line Multiplex Section (MS) Line Alarm Indication Signal (LAIS). AIS generation is controlled by the **TX_LAIS_x** register.

If **TX_LAIS_x** = 1, the entire payload of signal x (37584, 9396 or 2349 bytes) is filled with all-1s bytes.

5.9.2.2 Unequipped Generation

Unless AIS is active, unequipped SPE/VC (all SPE/VC bytes are filled with all 0s) is generated if **TX_UNEQ_x** = 1.

5.9.3 TOH/SOH Generation

The SONET TOH bytes are generally the same as the SDH SOH bytes. The following sections define the values generated for all TOH/SOH bytes. Where the byte names differ between SONET and SDH, the SONET name will be listed first. Entries that are blank in Table are SONET undefined or SDH, non-standardized, reserved bytes. RHINE fills these bytes with all 0s.

Table 18. STS-48/STM-16 TOH/SOH

	Column					
Row	1	2-48	49	50-96	97	98-144
1	A1[1]	A1[2:48]	A2[1]	A2[2:48]	J0[1]	Z0[2:48]
2	B1		E1		F1	
3	D1		D2		D3	
4	H1[1]	H1[2:48]	H2[1]	H2[2:48]	H3[1]	H3[2:48]
5	B2[1]	B2[2:48]	K1		K2	
6	D4		D5		D6	
7	D7		D8		D9	
8	D10		D11		D12	
9	S1	Z1[2:48][b]	Z2[1][b]	Z2[2][b], M1, Z2[4:48][b]	E2	

[b] The Z1 and Z2 bytes are non-standardized reserved bytes for STM-16.

Table 19. STS-12/STM-4 TOH/SOH

	Column					
Row	1	2-12	13	14-24	25	26-36
1	A1[1]	A1[2:12]]	A2[1]	A2[2:12]	J0[1]	Z0[2:12]
2	B1		E1		F1	
3	D1		D2		D3	

Table 19. STS-12/STM-4 TOH/SOH

	Column					
Row	1	2-12	13	14-24	25	26-36
4	H1[1]	H1[2:12]	H2[1]	H2[2:12]	H3[1]	H3[2:12]
5	B2[1]	B2[2:12]	K1		K2	
6	D4		D5		D6	
7	D7		D8		D9	
8	D10		D11		D12	
9	S1	Z1[2:12][b]	Z2[1][b]	Z2[2][b], M1, Z2[4:12][b]	E2	

[b] The Z1 and Z2 bytes are non-standardized reserved bytes for STM-4.

Table 20. STS-3/STM-1 TOH/SOH

	Column					
Row	1	2-3	4	5-6	7	8-9
1	A1[1]	A1[2:3]]	A2[1]	A2[2:3]	J0[1]	Z0[2:3]
2	B1		E1		F1	
3	D1		D2		D3	
4	H1[1]	H1[2:3]	H2[1]	H2[2:3]	H3[1]	H3[2:3]
5	B2[1]	B2[2:3]	K1		K2	
6	D4		D5		D6	
7	D7		D8		D9	
8	D10		D11		D12	
9	S1	Z1[2:3][b]	Z2[1][b]	Z2[2][b], M1	E2	

[b] The Z1 and Z2 bytes are non-standardized reserved bytes for STM-1.

5.9.3.1 Frame Bytes (A1 and A2)

The frame bytes are normally generated with the following fixed patterns:

- A1: 1111_0110 = F6
- A2: 0010_1000 = 28

For testing purposes, A1 and A2 can be generated with errors. If **A1A2_ERR_x** = 0, no errors are inserted. When **A1A2_ERR_x** is 1, then m consecutive frames (where m is the binary equivalent of **A1A2_ERR_NUM_x_[2:0]**) in each group of 8 frames, is generated with A1 and A2 exclusive-ORed with the contents of **A1A2_ERR_PAT_x_[15:0]**. The MSB of A1 is XORed with **A1A2_ERR_PAT_x_[15]**, and the LSB of A2 is XORed with **A1A2_ERR_PAT_x_[0]**.

5.9.3.2 Section Trace/Regenerator Section Trace (J0) and Section Growth/Spare (Z0)

Section Trace. Over periods of 16-consecutive frames, RHINE continuously transmits the 16-byte pattern contained in **TX_J0_x_[15:0]_[7:0]**. The bytes are transmitted in descending order starting with **TX_J0_x_[15]_[7:0]**.

The SDH G.707 standard states that a 16-byte, section-trace frame containing the Section Access Point Identifier (SAPI) defined in clause 3/G.831 should be transmitted continuously in consecutive-J0 bytes. Note that only the frame start-marker byte should contain a 1 in its MSB.

The Section Trace function is not currently defined for SONET. Unless a similar section trace is defined for SONET, all of the **TX_J0_x** bytes should be filled with 0000_0001 so that a decimal 1 is transmitted continuously in J0.

Section Growth/Spare. If **TX_LINE_CONFIG_[4]** = 1, the Z0 bytes are transmitted in order as the binary equivalent of 2 to 48. Otherwise, if **TX_LINE_CONFIG_[3:0]** = 1111, the corresponding Z0 bytes are transmitted in order as the binary equivalent of 2 to 12. Else if **TX_LINE_CONFIG_[3:0]** = 0000, the Z0 bytes are transmitted in order as the binary equivalent of 2 to 3 (this is specified in GR-253 and permitted by G.707).

5.9.3.3 Section BIP-8 (B1)

The B1 Bit Interleaved Parity 8 (BIP-8) is transmitted as even parity (normal) if **B1_INV_x** = 0. Otherwise, odd parity (incorrect) is generated. The BIP-8 is calculated over all bits of the previous SONET/SDH frame after scrambling and placed in the B1 byte of the current frame before scrambling.

By definition of BIP-8, the first bit of B1 provides parity over the first bit of all bytes of the previous frame, the second bit of B1 provides parity over the second bit of all bytes of the previous frame, and so on.

5.9.3.4 Serial TOH/SOH Insert Channel

RHINE provides a single serial channel through which the user can insert certain TOH/SOH bytes for all 16 channels. The TOH/SOH bytes that are accessible via this serial channel are (in order of transmission over the serial channel): E1[1], E1[2]...E1[16], F1[1], F1[2]...F1[16], D1[1], D1[2]...D1[16], D2[1], D2[2]...D2[16], D3[1], D3[2]...D3[16], K1[1], K1[2]...K1[16], K2[1], K2[2]...K2[16], D4[1], D4[2]...D4[16], D5[1]...D12_[1], D12[2]...D12[16], E2[1], and E2[2]...E2[16]. The E1 and E2 orderwire bytes are defined for the purpose of carrying two 64kb/s of digitized voice signals. The F1 byte is available for use by the network provider. There are 2 DCCs defined in the TOH/SOH. The Section/Regenerator Section DCC uses the D1, D2, and D3 bytes to create a 192 kb/s channel. The Line/Multiplex Section DCC uses bytes D4 through D12 to create a 576 kb/s channel. The K1 and K2 bytes carry APS messages.

The transmit block for each line interface accepts a single serial input, *TX_TOH_DATA.* A single, 19.44-MHz clock (*TX_TOH_CLK*) is output from RHINE in order to provide a timing reference for the *TX_TOH_DATA* input. The serial channel carries 2430 bits-per-frame. The structure of this frame consists of first a stuff byte (SN_x) for each of the 16 tributaries, followed by the 17 TOH bytes identified in the previous paragraph. These 8*16 + 17*16*8 = 2304 bits are then followed by 126 stuff bits, which are set to 0. RHINE provides a *TX_TOH_FRAME* signal as a byte delineation indication for the TOH serial channel. RHINE will accept the TOH/SOH bytes in the following order (SN indicates a stuff byte) SN[1]-SN[16], E1[1]-E1[16], F1[1]-F1[16], D1[1]-D1[16], D2[1]-D2[16], D3[1]-D3[16], K1[1]-K1[16], K2[1]-K2[16], D4[1]-D4[16], D5[1]-D5[16], D6[1]-D6[16], D7[1]-D7[16], D8[1]-D8[16], D9[1]-D9[16], D10[1]-D10[16], D11[1]-D11[16], D12[1]-D12[16], E2[1]-E2[16], followed by 126 stuff bits, for a total of 2430 bits per frame. All bytes should be provided in MSB to LSB order. RHINE will accept the first bit (the MSB) of the SN byte for tributary [1] on *TX_TOH_DATA* the first clock cycle following a rising edge of *TX_TOH_FRAME*.

In addition to this method of inserting the K1 and K2 bytes, RHINE also provides register map access to these same bytes. The user can select the method of access for the K1K2 bytes via the **TX_K1K2_APS_[1:16]** and **TX_K2_3LSB_[1:16]** registers. If **TX_K1K2_APS_x** = 0 (the default), the **TX_K1_x_[7:0]** and **TX_K2_x_[7:3]** register are the source of the K1_[7:0] and K2_[7:3] bits in the outgoing TOH/SOH of tributary x. If **TX_K1K2_APS_x** = 1, *TX_TOH_DATA* is the source of these bits. Similarly, **TX_K2_3LSB_x** selects the source of the 3 LSBs of the K2 byte for tributary y; if = 0 (the default), the register map is the source of these 3 bits. (See section 5.9.3.6.) Else if = 1, *TX_TOH_DATA* is the source of these bits.

5.9.3.5 Line/MS BIP-384/96 (B2)

In the following B2 description, the numbers vary slightly, dependent on the mode of the device (STS-48, STS-12, or STS-3c mode). To describe the operation of all cases, the following convention will be used to identify the requirement that applies to each mode: STS-48 [STS-12 | STS-3c]. The STS-12/STM-4 and STS-3c/STM-1 requirements will follow the STS-48/STM-16 requirement in braces.

There are 48 [12 | 3] B2 bytes in the TOH/SOH, and together they provide a BIP-384 [BIP-96 | BIP-24] error detection capability. Each B2 byte provides BIP-8 parity over bytes in 1 of 48 [1 of 12 | 1 of 3] groups of bytes in the previous frame. The B2 byte in column j provides BIP-8 parity over bytes in the previous frame (except those in the first 3 rows of TOH/SOH) that appear in columns j + 48k [j + 12k | j + 3k], where k = 0 through 89. The BIP-8 is transmitted as even parity (normal) if **B2_INV_x** = 0. Otherwise, odd parity (incorrect) is generated. The BIP-8 values are calculated over bytes in the previous STS-48/STM-16 [STS-12/STM-4 | STS-3c/STM-1] frame before scrambling and placed in the B2 bytes of the current frame before scrambling.

5.9.3.6 APS Channel and Line/MS AIS/RDI (K1 and K2)

K1 and the 5 MSBs of K2 are used for automatic-protection switching (APS) signaling. The 3 LSBs of K2 are used as an AIS or Remote Defect Indication (RDI) at the line/MS level, and in SONET, they are also used for APS signalling.

The K1 byte and the 5 MSBs of the K2 byte are controlled from 3 sources. Note that, in the following description, for STS-48/STM-16 mode (**TX_LINE_CONFIG_[4]** = 1), **x** = 1; for STS-12/STM-4 modes (**TX_LINE_CONFIG_[3:0]** = 1111), **x** = 1, 5, 9, and 13; for STS-3/STM-1 modes (**TX_LINE_CONFIG_[3:0]** = 0000), **x** = 1 to 16. In order of priority:

- If **TX_K1K2_APS_x** = 1, the *TX_TOH_DATA* serial channel is the source of the K1_[7:0] and K2_[7:3] information.
- If **TX_K1K2_APS_x** = 0, RHINE inserts **TX_K1_x_[7:0]** in the transmitted K1 bytes and **TX_K2_x_[7:3]** in the transmitted 5 MSBs of K2 bytes.

The 3 LSBs of K2 are controlled from 4 sources. Note that, in the following description, for STS-48/STM-16 mode (**TX_LINE_CONFIG_[4]** = 1), **x** = 1; for STS-12/STM-4 modes (**TX_LINE_CONFIG_[3:0]** = 1111), **x** = 1, 5, 9, and 13; for STS-3/STM-1 modes (**TX_LINE_CONFIG_[3:0]** = 0000), **x** = 1 to 16. If **TX_K2_3LSB_x** = 0, the 3 LSBs of K2 are inserted according to the following logic, in order of priority:

- If **TX_LAIS_x** = 1, they are transmitted as all 1s (as are all line/MS overhead bytes).
- Else if (**LRDI_INH_x** = 0) and if any of (**RX_LOS_x** AND NOT **RX_LOS_RDI_INH_x)***,* **RX_LOF_x, RX_LOC_x** or **RX_LAIS_x** = 1, they are transmitted as 110. Any time this particular event is

active, the K2 is set to 110 for a minimum of 20 frames.

- If **TX_K2_3LSB_x** = 1, the source of the 3 LSBs of the K2 byte for tributary x is the *TX_TOH_DATA* serial channel. (See section 5.9.3.4.)
- Else **TX_K2_x_[2:0]** is transmitted.

Requirements R6-180 through R6-182 of GR-253 specify that RDI should be inserted and removed within 125 µs of detection and removal of received LOS, LOF, or LAIS, provided the 20 frame minimum criterion has been met. If **TX_K2_3LSB_x** = 1 or if the **TX_K2_x_[2:0]** register is used for insertion of the 3 LSBs of the K2 byte, the user is responsible for meeting the detection/removal timing and the 20 frame minimum via his/her external generation of this data. RHINE provides a transparent pass-through of this data in this mode of operation.

5.9.3.7 Synchronization Status (S1)

The 4 LSBs of this byte convey synchronization status messages. The transmitted S1 byte is set equal to **TX_S1_x_[7:0]**.

5.9.3.8 Line/MS REI (M1)

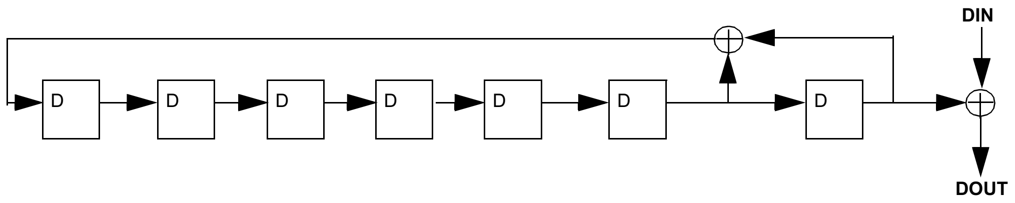
The Receive side monitors B2 bit errors in the received signal. The number of B2 errors detected each frame can range from 0 to 255 (0 to 96 for STS-12c/STM-4 operation, 0 to 24 for STS-3c/STM-1 operation; values outside of these ranges are interpreted as 0 errors) B2 bits per frame (values greater than 255 are truncated to 255). The line/MS Remote Error Indication (REI) byte, the M1 byte, normally conveys the count of B2 errors detected in the received signal.

The user can force the transmission of REI error indications by setting **TX_M1_ERR_x** = 1. If **TX_LINE_CONFIG_[4]** = 1 (STS-48/STM-16 mode), this causes a count of 255 to be transmitted in the M1 byte. If **TX_LINE_CONFIG_[3:0]** = 1111 (STS-12/STM-4 mode), this causes a count of 96 to be transmitted in the M1 byte. If **TX_LINE_CONFIG_[3:0]** = 0000 (STS-3c/STM-1 mode), this causes a count of 24 to be transmitted in the M1 byte. Else if (**LREI_INH_x** = 0), the M1 byte is set equal to the most recent B2 error count. Otherwise, the M1 byte is set to all 0s.

5.9.4 Scrambling

The input is scrambled with a frame synchronous scrambling sequence generated from the polynomial g(x) = $x^7 + x^6 + 1$. The scrambling is done in parallel but the result is equivalent to the serial scrambler shown in Figure 13. The scrambler is initialized to 1111111 at the beginning of the first SPE/VC byte (the byte in row 1, column 145 for STS-48/STM-16 operation, in column 37 for STS-12/STM-4 operation, or in column 10 for STS-3c/STM-1 operation), and it scrambles the entire SONET/SDH signal except for the first row of TOH/SOH. For testing purposes, the scrambler can be disabled by setting the **SCRINH_x** bit to 1.

Figure 13: SONET/SDH Scrambler



5.10 SONET R-to-T Loopback

There is no provisioning in Rhine for SONET receive to transmit loopback right at the line interface. However, SONET/SDH hairpin loopback can be provisioned on a per path basis as described in section 5.7.3. Note that this hairpin loopback occurs after section and line termination in the RX receiver and prior to line and section overhead generation in the transmitter. Consequently, received alarms and errors will be detected and reported, but removed in the loopback process (e.g. received B1 and B2 errors will not be contained in the retransmitted, loopback signal).

5.11 Transmit Output Disable

Primarily for power savings, but also to reduce external EMI and internal noise, all primary data I/O (including their associated clocks) have disable capability. Differential LVPECL outputs are disabled by tristating the outputs, as well as disabling them internally.

TX_TRIB_INH_x disables *TX_DATA_OUT_x*, *TX_SONETCLK_OUT_155,* and *TX_SONETCLK_OUT_155_622* (LVPECL). In STS-48/STM-16 operation, all the **TX_TRIB_INH_x** should be activated to disable all 16 output pins associated with the STS-48/STM-16 output signal.

All inhibits are active high and default to zero; all inputs and outputs are enabled.

6.0 Receive Direction

6.1 Configuration

RHINE supports the following registers for configuration of the Receive side: **RX_LINE_CONFIG, RX_POS_x, RX_DIRECT_MAP_x, RX_PP_CONFIG_[20:0],** and **RX_TRIB_INH_x**. These registers have the same definition as their Transmit counterparts.

6.2 Receive Input Disable

Primarily for power savings, but also to reduce external EMI and internal noise, all primary data I/O (including their associated clocks) have disable capability. Differential LVPECL inputs are disabled at the front end of the receiver.

RX_TRIB_INH_x disables *RX_DATA_x,* and *RX_SONETCLK_x* (LVPECL). For STS-48/STM-16 operation, all the **RX_TRIB_INH_x** bits should be activated to disable all pins associated with the STS-48/STM-16 signals.

All inhibits are active high and default to zero; all inputs and outputs are enabled.

6.3 T-to-R Loopback

RHINE receive section can be configured to loop back the generated transmit signal(s) if **SONET_T_TO_R_LOOP_x** = 1. Otherwise, the received signal from the SONET/SDH interface is selected. While in loopback, the *TX_SONETCLK_IN* input is used to clock the receiver framer and other receiver circuitry for tributary x. If loopback is not selected, the appropriate *RX_SONETCLK_x* input is used to clock this circuitry.

6.4 Clock Reference Output

RHINE outputs a reference clock signal through output signal *RX_REF_CLK_OUT.* The user can select any of the input clocks it receives to serve as the clock source for this reference output. The register **RX_REF_CLK_SEL_[4:0]** selects between the possible reference clocks for *RX_REF_CLK_OUT* and **RX_REF_CLK_FREQ_[1:0]** selects the frequency of this output clock. See Table 21 and Table 22.

Table 21. RX_REF_CLK_SEL_[4:0] Register Values

RX_REF_CLK_SEL_[4:0]	RX_LINE_CONFIG_[4]	Reference Clock
0 0001	1	RX_SONETCLK_[2]
0 0000	0	RX_SONETCLK_[1]
0 0001	0	RX_SONETCLK_[2]
0 0010	0	RX_SONETCLK_[3]
:	0	:
0 1111	0	RX_SONETCLK_[16]
1 1111	0	TX_SONETCLK_IN
(other values)	(other values)	Undefined

Table 22. RX_REF_CLK_FREQ_[1:0] Register Value

RX_REF_CLK_FREQ_[1:0]	Reference Clock Frequency
00	8 kHz (note that this is not a frame sync in that it has no fixed relationship to the framing of the input signal)
01	19.44 MHz
10	38.88 MHz
11	77.76 MHz

6.4.1 LOC

The active *RX_SONETCLK_x* inputs are monitored for loss-of-clock using the *UPCLK* input. If no transitions are detected on the receive line side clocks for 16 periods of *UPCLK*, the appropriate **RX_LOC_x** bit is set. It is cleared when transitions are detected.

The **RX_LOC_x_D** delta bit is set if **RX_LOC_x** transitions from either a 0 to a 1, or from a 1 to a 0.

6.4.2 LOS

RHINE provides 16 internal LOS monitors, as well as the ability to accept signals from external LOS monitors.

Sixteen input signals, *RX_LOSEXT_x*, are provided for an external Loss of Signal indication for each of the 16 possible input signals to RHINE. The state of this input is reported to the management interface via register **RX_LOSEXT_x**. *RX_LOSEXT_x* can be active high (**RX_LOSEXT_LEVEL_x** = 0, the default) or active low (**RX_LOSEXT_LEVEL_x** = 1). Internally, if **RX_LOSEXT_LEVEL_x** = 1, *RX_LOSEXT_x* is inverted to produce **RX_LOSEXT_x**.

RX_LOSEXT_x can contribute directly to the declaration of an LOS condition (which results in **RX_LOS_x** being set), or can first enter a delay block, where it must be set for a minimum of 3.29 µs (STS-48/STM-16 or STS-12/STM-4) or 10 µs (STS-3/VC-4) before **RX_LOS_x** is set. Inhibits **RX_LOSEXT_INH_x** and **RX_LOSEXT_DELAY_INH_x** are provided to control both the immediate and delayed contributions, respectively, of **RX_LOSEXT_x** to **RX_LOS_x**.

In addition, RHINE can itself detect an LOS condition, by monitoring the receive data (*RX_DATA_[15:0] or RX_DATA_[1:16])* for a continual stream of all 0s. If *RX_DATA_x* == 0 for a minimum of 3.29 µs (for STS-48/STM-16 or STS-12/STM-4; a minimum of 10 µs for STS-3/STM-1), RHINE declares Loss of Signal (**RX_LOS_x** = 1). A separate inhibit bit controls this all-zero detection feature, **RX_LOS_ALL_ZERO_INH_x**.

RX_LOS_x is cleared when all non-inhibited contributors are cleared: **RX_LOSEXT_x** = 0, and the incoming signal has 2-consecutive, valid-framing, alignment patterns, and during the intervening time (one frame), no all 0s pattern qualifying as an LOS defect was present.

RX_LOS_x also contributes to Line RDI insertion in the transmit direction. (See section 5.9.3.6.)

6.5 SONET/SDH Framers

The SONET/SDH framers locate the framing bytes in the selected data signal, and by doing so are able to find byte alignment and determine the position of all TOH/SOH bytes. After finding frame, the framers shift the data so that their output data is byte aligned. They also descramble the data, perform B1 monitoring, and provide frame counter outputs to the TOHMON and Pointer Interpreter blocks.

When the framer state machine is out-of-frame (**RX_OOF_x** = 1), it searches for the 32-bit A1-A1-A2-A2 framing byte sequence of 0xF6F6_2828. For STS-48/STM-16 applications (**RX_LINE_CONFIG_[4]** = 1), this pattern can start on any of the 16-input data lines and span up to 3-input words. In all modes, when the framer finds 2 successive sequences separated in time by 125 µs that exactly match the framing pattern, it goes into frame (**RX_OOF_x** = 0) and byte aligns its output data bus.

The framer remains in-frame until it receives 5 successive frames with at least 1-bit error in the A1-A1-A2-A2 framing pattern. When this occurs, **RX_OOF_x** is set to 1, and a new frame search is begun.

The framer also provides a loss-of-frame indication. If **RX_OOF_x** is active (1) continuously for 24-consecutive frames (3 ms), the **RX_LOF_x** bit is set to 1. Once **RX_LOF_x** is set, it remains high until **RX_OOF_x** is inactive (0) continuously for either 24 (if **RX_LOF_ALG_x** = 1) or 8 (if **RX_LOF_ALG_x** = 0) consecutive frames.

The **RX_OOF_x_D** and **RX_LOF_x_D** delta bits contribute to the summary interrupt, and the **RX_OOF_x_SECE** and **RX_LOF_x_SECE** second event bits are set at the end of each second that the **RX_OOF_x** and **RX_LOF_x** bits are in the active state at any time during the second.

6.5.1 Descrambling

Before the data is output from the Framer block, it can be descrambled using the same frame synchronous sequence that is used to scramble the transmit data. (See section 5.9.4.) The descrambler is reset to 1111111 at the beginning of the first SPE/VC byte (the byte in row 1, column 145 (STS-48/STM-16 modes) or column 37 (STS-12/STM-4 mode) or column 10 (STS-3c/STM-1 mode)), and it descrambles the entire SONET/SDH signal except for the first row of TOH/SOH. For testing purposes, the descrambler for SONET/SDH signal y can be disabled by setting **DSCRINH_x** to 1.

6.5.2 B1 Monitor

In all modes, RHINE checks the received B1 bytes for correct Bit Interleaved Parity 8 (BIP-8) values. Even parity BIP-8 is calculated over all bytes of each frame before descrambling. This value is then compared to the received B1 value in the following frame after descrambling. The comparison can result in from 0 to 8 mismatches (B1 bit errors).

RHINE contains 16 16-bit B1 error counters that either count every B1 bit error (if **BIT_BLKCNT** = 0) or every frame with at least 1 B1 bit error (if **BIT_BLKCNT** = 1). When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **B1_ERRCNT_x_[15:0]** registers, and the B1 error counters are cleared. (See section 4.2.)

If there has been at least 1 B1 error since the last rising edge of LATCH_EVENT*,* then the B1 error second event bit, **B1ERR_x_SECE**, is set.

6.6 Transport Overhead Monitoring

The TOH/SOH monitoring blocks consists of J0, B2, K1K2, S1, and M1 monitoring. These TOH/SOH bytes are monitored for errors or changes in states.

6.6.1 J0 Monitoring

There are 2 modes of operation for J0 monitoring, one is typically used in SONET applications; the other is used in SDH applications.

In the **RX_SDH_J0_x** = 0 mode (SONET), J0 monitoring consists of examining the received J0 bytes for values that match consistently for 3-consecutive frames. When a consistent J0 value is received, it is written to **RX_J0_x_[15]_[7:0]**.

In the **RX_SDH_J0_x** = 1 case (SDH), the J0 byte is expected to contain a repeating 16-byte section trace frame that includes the Section Access Point Identifier. J0 monitoring consists of locking on to the start of the 16-byte, section-trace frame and examining the received section trace frames for values that match consistently for 3-consecutive, section-trace frames. When a consistent frame value is received, it is written to **RX_J0_x_[15:0]_[7:0]**. The first byte of the section trace frame (which contains the frame start-marker) is written to **RX_J0_x_[15]_[7:0]**.

6.6.1.1 Framing

The MSBs of all section-trace frame bytes are 0, except for the MSB of the frame start-marker byte. The J0 monitor framer searches for a J0 byte with a 1 in its MSB, followed by 15-consecutive, J0 bytes that have a 0 in their MSB. When this pattern is found, the framer goes into frame, **J0_OOF_x** = 0. Once the J0 monitor framer is in-frame, it remains in frame until 3-consecutive, section-trace frames are received with at least 1 MSB bit error. If **RX_SDH_J0_x** = 0, the J0 frame indication is held in the In-frame state, **J0_OOF_x** = 0. The **J0_OOF_x_D** delta bit is set when **J0_OOF_x** changes state.

6.6.1.2 Pattern Acceptance and Comparison

Once in frame, the J0 monitor block looks for 3-consecutive, 16-byte (**RX_SDH_J0_x** = 1) or 1-byte (**RX_SDH_J0_x** = 0) section-trace frames. When 3-consecutive, identical frames are received, the accepted frame is stored in **RX_J0_x_[15:0]_[7:0]** (with the first byte of the frame stored in **RX_J0_x_[15]_[7:0]**) or **RX_J0_x_[15]_[7:0]** in the SONET mode.

Accepted frames are compared to the previous contents of these registers. When a new value is stored, the **RX_J0_x_D** delta bit is set. RHINE does not detect and report TIM-L directly. However, this feature enables users to rapidly detect TIM-L by flagging changes in the content of the captured J0 trace.

6.6.2 BIP-384 (B2) Checking

In the following B2 description, the numbers vary slightly dependent on the mode of the device (STS-48 mode vs. STS-12 vs. STS-3c). To describe the operation of all cases, the following convention will be used to identify the requirement that applies to each mode: STS-48 [STS-12 | STS-3c]. The STS-12/STM-4 and STS-3c/STM-1 requirements will follow the STS-48/STM-16 requirement in braces.

RHINE checks the received B2 bytes for correct BIP-8 values. Even parity BIP-384 [BIP-96 | BIP-24] is calculated over all groups of 48 [12 | 3] bytes of each frame, except the first 3 rows of TOH (SOH in SONET and RSOH in SDH). The calculation is done on the received data after descrambling. This value is then compared to the B2 values in the following frame after descrambling. The comparison can result in from 0 to 384 [0 to 96 | 0 to 24] mismatches (B2 bit errors). Values outside of these ranges are interpreted as 0 errors. The number of B2 bit errors detected each frame can be inserted into the transmitted M1 byte. (See section 5.9.3.8.)

6.6.2.1 B2 Error Counting

RHINE contains 16 22-bit B2 error counters that count every B2 bit error. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **B2_ERRCNT_x_[21:0]** registers, and the B2 error counters are cleared. (See section 4.2.)

If there has been at least 1 B2 error since the last rising edge of LATCH_EVENT, then the B2 error second event bit, **B2ERR_x_SECE**, is set.

6.6.2.2 B2 Error Rate Threshold Blocks

For the purpose of determining whether or not the bit error rate of a received signal is above or below 2 different provisionable thresholds (the B2 Signal Fail and the Signal Degrade conditions), RHINE provides B2 error rate threshold blocks. If a B2 SF block or a B2 SD block determines that the B2 error rate is above a threshold, it sets the corresponding **B2_ERR_SF_x** or **B2_ERR_SD_x**. The delta bits **B2_ERR_SF_x_D** or **B2_ERR_SD_x_D** are set if the corresponding error rate bit changes value.

For each B2 error rate threshold block, the user can provision a B2 BLOCK register and 2 pairs of B2 THRESH and B2 GROUP registers. In order to allow hysteresis in setting and clearing the state bits, each B2 error rate threshold block has 1 pair of B2 THRESH and B2 GROUP registers for setting the state and 1 pair of B2 THRESH and B2 GROUP registers for clearing the state. Thus, the registers used in the B2 error rate threshold blocks are

- While **B2_ERR_SF_x** = 0, determine if it should be set using **B2_BLOCK_SF_x_[7:0]**, **B2_THRESH_SET_SF_x_[7:0]**, and **B2_GROUP_SET_SF_x_[5:0]**.
- While **B2_ERR_SF_x** = 1, determine if it should be cleared using **B2_BLOCK_SF_x_[7:0]**, **B2_THRESH_CLR_SF_x_[7:0]**, and **B2_GROUP_CLR_SF_x_[5:0]**.
- While **B2_ERR_SD_x** = 0, determine if it should be set using **B2_BLOCK_SD_x_[15:0]**, **B2_THRESH_SET_SD_x_[5:0]**, and **B2_GROUP_SET_SD_x_[5:0]**.
- While **B2_ERR_SD_x** = 1, determine if it should be cleared using **B2_BLOCK_SD_x_[15:0]**, **B2_THRESH_CLR_SD_x_[5:0]**, and **B2_GROUP_CLR_SD_x_[5:0]**.

The values that should be provisioned in these registers as a function of the desired BER at which B2 Signal Fail and Signal Degrade should be declared and cleared are shown in Table 23, Table 24, and Table 25. For a given desired B2 BER threshold, the provisioned set and clear values for B2 Signal Fail should be taken from the same row in the appropriate table (i.e., the recommended SF values already take into account a clearing condition which is 10 times better than the setting condition). The same applies to selection of B2 Signal Degrade values. Signal Degrade threshold is typically 1000 times better than the Signal Fail threshold (e.g., if SF threshold is set at 1E-6, SD threshold is typically set to 1E-9).

NOTE in the above descriptions and the following tables, BLOCK, THRESH_SET, GROUP_SET, THRESH_CLR, and GROUP_CLR have the following meanings:

If >= THRESH_SET total number of B2 errors are detected in one BLOCK of consecutive frames for GROUP_SET consecutive BLOCKs, set the corresponding SF or SD condition. If >= THRESH_SET total number of B2 errors are detected in fewer than BLOCK consecutive frames, increment the event detected GROUP count, and reset the B2 error and consecutive frame detected counters to start monitoring for the next consecutive GROUP.

If < THRESH_CLR number of B2 errors are detected in one BLOCK of consecutive frames for GROUP_SET consecutive BLOCKs, clear the corresponding SF or SD condition. As soon as >= THRESH_CLR B2 errors are detected in a single BLOCK, reset the error, block, and group counters to begin a new clearing threshold search.

Table 23. Recommended Provisioning for STS-48/STM-16 B2 Signal Fail and Signal Degrade

BER	BLOCK	THRESH_SET	THRESH_CLR	GROUP_SET	GROUP_CLR
10^{-3}	1	154	30	4	5
10^{-4}	2	55	7	4	4
10^{-5}	10	24	4	4	4
10^{-6}	100	28	4	2	4
10^{-7}	1000	28	4	2	4
10^{-8}	5000	15	2	2	4
10^{-9}	65000	18	3	2	4

Table 24. Recommended Provisioning for STS-12/STM-4 B2 Signal Fail and Signal Degrade

BER	BLOCK	THRESH_SET	THRESH_CLR	GROUP_SET	GROUP_CLR
10^{-3}	1	39	8	4	5
10^{-4}	6	37	7	4	5
10^{-5}	40	28	4	2	5
10^{-6}	400	28	4	2	5
10^{-7}	4000	28	4	2	5
10^{-8}	20,000	15	2	2	4
10^{-9}	60,000	5	1	3	6

Table 25. Recommended Provisioning for STS-3/STM-1 B2 Signal Fail and Signal Degrade

BER	BLOCK	THRESH_SET	THRESH_CLR	GROUP_SET	GROUP_CLR
10^{-3}	3	28	7	4	5
10^{-4}	16	23	4	4	5
10^{-5}	160	28	4	2	5
10^{-6}	1600	28	4	2	5
10^{-7}	16,000	28	4	2	5
10^{-8}	40,000	7	1	3	5

Table 25. Recommended Provisioning for STS-3/STM-1 B2 Signal Fail and Signal Degrade

BER	BLOCK	THRESH_SET	THRESH_CLR	GROUP_SET	GROUP_CLR
10^{-9}	65,000	2	1	4	28

6.6.3 K1K2 Monitoring

The K1 and K2 bytes, which are used for sending Line/MS AIS or RDI and for APS signalling, are monitored for change in status.

6.6.3.1 Line/MS AIS Monitoring

The 3 LSBs of K2 can be used as an AIS or Remote Defect Indication (RDI) at the line/MS level.

If they are received as “111” for **K2_CONSEC_x_[3:0]** consecutive frames (**K2_CONSEC_x_[3:0]** = 000 operation is undefined), **RX_LAIS_x** is set. If for **K2_CONSEC_x_[3:0]** consecutive frames, they are not received as “111,” then **RX_LAIS_x** is cleared. The **RX_LAIS_x_D** delta bit is set when **RX_LAIS_x** changes state.

6.6.4 AIS-L Data Disable

During periods in which **RX_LAIS_x** is set, data is disabled from entering the corresponding RX Pointer Processor FIFOs. This is done to avoid loading the FIFOs with bad data during periods in which an alarm condition exists.

6.6.4.1 Line/MS RDI Monitoring

The 3 LSBs of K2 are also monitored for **K2_CONSEC_x_[3:0]** consecutive receptions or non-receptions of “110.” When this is received, **RX_LRDI_x** are set or cleared. **RX_LRDI_x_D** is set when **RX_LRDI_x** changes state.

6.6.4.2 APS Monitoring

If the K1 byte and the 5 MSBs of the K2 byte, which are used for sending APS requests and channel numbers, are received identically for 3-consecutive frames, their values are written to **RX_K1_x_[7:0]** and **RX_K2_x_[7:3]**. Accepted values are compared to the previous contents of these registers, and when a new 12-bit value is stored, the **RX_K1_x_D** delta bit is set.

The K1 byte is checked for instability. If, for 12-successive frames, no 3-consecutive frames are received with identical K1 bytes, the **K1_UNSTAB_x** bit is set. It is cleared when 3-consecutive, identical, K1 bytes are received. When **K1_UNSTAB_x** changes state, the **K1_UNSTAB_x_D** delta bit is set.

Bits 2 down to 0 of K2 can contain APS mode information. These bits are monitored for **K2_CONSEC_x_[3:0]** consecutive, identical values. **RX_K2_x_[2:0]** is written when this occurs, *unless the value of bits 2 and 1 of K2 is “11” (indicating Line/MS AIS or RDI, section 6.6.3.1).* The **RX_K2_x_D** delta bit is set when a new value is written to **RX_K2_x_[2:0]**.

The delta bits associated with APS monitors, **RX_K1_x_D, RX_K2_x_D** and **K1_UNSTAB_x_D** all contribute to an APS interrupt signal, *APS_INTB.* In addition, these deltas also contribute to the standard summary interrupt signal, *INTB. (*See section 10.1 and section 11.2.1.2.)

6.6.5 S1 Monitoring

The 4 LSBs of received S1 bytes are monitored for consistent values in either 8-consecutive frames in the SONET mode, **RX_SDH_S1_x** = 0, or 3-consecutive frames in the SDH (**RX_SDH_S1_x** = 1) mode. When these bits contain a consistent synchronization status message, the accepted value is written to **RX_S1_x_[3:0]**. Accepted values are compared to the previous contents of this register, and when a new value is stored, the **RX_S1_x_D** delta bit is set.

The S1 byte is also checked for message failure. If no message has met the above validation criterion (whether it is the same or different from the last accepted value) at any time since the last rising edge of LATCH_EVENT*,* then the S1 fail second event bit, **S1FAIL_x_SECE**, is set.

6.6.6 M1 Monitoring

The M1 byte indicates the number of B2 errors that were detected by the remote terminal in its received signal. RHINE contains 16 22-bit M1 error counters that count every error indicated by M1. When the performance-monitoring counters are latched, the value of these counters are latched to the **M1_ERRCNT_x_[21:0]** registers, and the M1 error counters are cleared. (See section 4.2.)

The number of errors detected each frame can range from 0 to 255 (0 to 96 for STS-12c/STM-4 operation, 0 to 24 for STS-3c/STM-1 operation, values outside of this range are interpreted as 0 errors) B2 bits per frame (values greater than 255 are truncated to 255). The line/MS Remote Error Indication (REI) byte, the M1 byte, normally conveys the count of B2 errors detected in the received signal. In certain ring configurations, the transmitter conveying this information back upstream can be sourced by a separate RHINE device.

If there has been at least 1 received M1 error indication since the last rising edge of LATCH_EVENT*,* then the M1 error second event bit, **M1ERR_x_SECE**, is set.

6.6.7 External Alarm Signal

RHINE provides 16 output signals, *RX_ALARM_OUT_[1:16]*, that report SONET Receive alarms for each of the possible x receive signals. *RX_ALARM_OUT_x* is set (= 1) when any of the unmasked alarm conditions exist, and is cleared (= 0) when all unmasked alarm conditions are cleared. The alarm set that contributes to *RX_ALARM_OUT_x* is provisionable, via individual mask registers.

RX_ALARM_OUT_x =
(**RX_OOF_x** && !**RX_OOF_ALARM_INH_x**) ||
(**RX_LOF_x** && !**RX_LOF_ALARM_INH_x**) ||
(**RX_LOC_x** && !**RX_LOC_ALARM_INH_x**) ||
(**RX_LAIS_x** && !**RX_LAIS_ALARM_INH_x**) ||
(**RX_LOS_x** && !**RX_LOS_ALARM_INH_x**)

6.6.7.1 Serial TOH/SOH Drop Channel

RHINE provides a single serial channel through which the user can extract certain TOH/SOH bytes for all 16 channels. The TOH/SOH bytes that are accessible via this serial channel are (in order of transmission over the serial channel): E1[1], E1[2]...E1[16], F1[1], F1[2]...F1[16], D1[1], D1[2]...D1[16], D2[1], D2[2]...D2[16], D3[1], D3[2]...D3[16], K1[1], K1[2]...K1[16], K2[1], K2[2]...K2[16], D4[1], D4[2]...D4[16], D5[1]...D12_[1], D12[2]...D12[16], E2[1], and E2[2]...E2[16]. The E1 and E2 orderwire bytes are defined for the purpose of carrying two 64kb/s of digitized voice signals. The F1 byte is available for use by the network provider. There are 2 DCCs defined in the TOH/SOH. The Section/Regenerator Section DCC uses the D1, D2, and D3 bytes to create a 192 kb/s channel. The Line/Multiplex Section DCC uses bytes D4 through D12 to create a 576 kb/s channel. The K1 and K2 bytes carry APS messages.

The receive block for each line interface provides a single serial output, *RX_TOH_DATA.* A single, 19.44-MHz clock (*RX_TOH_CLK*) is accepted by RHINE in order to provide a timing reference for the *RX_TOH_DATA* output. The serial channel carries 2430 bits-per-frame. The structure of this frame consists of first a stuff byte (SN_x) for each of the 16 tributaries, followed by the 17 TOH bytes identified in the previous paragraph. These 8*16 + 17*16*8 = 2304 bits are then followed by 126 stuff bits, which are set to 0. RHINE accepts a *RX_TOH_FRAME* signal as a byte delineation indication for the TOH serial channel. RHINE will provide the TOH/SOH bytes in the following order (SN indicates a stuff byte) SN[1]-SN[16], E1[1]-E1[16], F1[1]-F1[16], D1[1]-D1[16], D2[1]-D2[16], D3[1]-D3[16], K1[1]-K1[16], K2[1]-K2[16], D4[1]-D4[16], D5[1]-D5[16], D6[1]-D6[16], D7[1]-D7[16], D8[1]-D8[16], D9[1]-D9[16], D10[1]-D10[16], D11[1]-D11[16], D12[1]-D12[16], E2[1]-E2[16], followed by 126 stuff bits, for a total of 2430 bits per frame. All bytes will be provided in MSB to LSB order. RHINE will provide the first bit (the MSB) of the SN nibble for tributary [1] on *RX_TOH_DATA* the first clock cycle following a rising edge of *RX_TOH_FRAME*. The frequency of *RX_TOH_CLK* should either be derived from the RX clock, or slightly exceed 19.44 MHz to insure that Rhine will never drop TOH data. If the provided clock slightly exceeds 19.44 MHz, the TOH data channel may not always have valid data to send over the TOH interface. The SN_x stuff byte will contain a code to indicate if any of these error conditions have occurred; if SN_x is all 0s, the TOH data from tributary x in the previous TOH/SOH channel frame x is valid; if SN_x is all 1s, the TOH data in that previous frame from tributary x is invalid and should be discarded. SN_x = 10101010 indicates that the previous frame of TOH data from tributary x has been dropped. Any other code is undefined.

In addition to this method of reporting the received K1 and K2 bytes for each tributary, RHINE also provides register map access to these same bytes. (See section 6.6.3.1, section 6.6.4.1, and section 6.6.4.2.)

6.7 Pointer Processors

6.7.1 Concatenation Provisioning

The operation of the Pointer Processors is influenced by the concatenation state of the received signals, and whether they are SONET or SDH signals. The expected concatenation state of the received signals is provisioned through the **RX_PP_CONFIG_[20:0]** registers. The interpretation of these registers is the same as the **TX_CONFIG_[20:0]** registers given in Table 10, is also a function of the **RX_SDH_PI_x** (to the extent that it determines whether the payloads are SONET or SDH payloads) and the **RX_LINE_CONFIG_[4:0]** registers. Note that configurations with STS-48c/AU-4-16c payloads are only valid if **RX_LINE_CONFIG_[4]**=1, and configurations with STS-12c/AU-4-4c payloads are only valid if **RX_LINE_CONFIG_[4]**=1 or **RX_LINE_CONFIG_[3:0]**=1111. Also note that the pointer processor can be configured to handle STS-3 signals; these signals cannot be terminated by RHINE, but must be ‘hair-pinned’ at the SONET level.

Table 26. RX_PP_CONFIG_[20:0] Provisioning

RX_PP_CONFIG_[20:0][a]	Interpretation
1_xxxx_xxxx_xxxx_xxxx_xxxx	STS-48c/AU-4-16c payload (Tributary index = 1,1)
0_1xxx_xxxx_xxxx_xxxx_xxxx	STS-12c/AU-4-4c payload in first quadrant (Tributary index = 1,1)
0_x1xx_xxxx_xxxx_xxxx_xxxx	STS-12c/AU-4-4c payload in second quadrant (Tributary index = 5,1)
0_xx1x_xxxx_xxxx_xxxx_xxxx	STS-12c/AU-4-4c payload in third quadrant (Tributary index = 9,1)
0_xxx1_xxxx_xxxx_xxxx_xxxx	STS-12c/AU-4-4c payload in fourth quadrant (Tributary index = 13,1)
0_0xxx_1xxx_xxxx_xxxx_xxxx	STS-3c/AU-4 payload in first STS-3 tributary group of first quadrant (Tributary index = [1])
0_0xxx_0xxx_xxxx_xxxx_xxxx	STS-3/3xAU-3 payload in first STS-3 tributary group of first quadrant (Tributary index = [1])
0_0xxx_x1xx_xxxx_xxxx_xxxx	STS-3c/AU-4 payload in second STS-3 tributary group of first quadrant (Tributary index = [2])
0_0xxx_x0xx_xxxx_xxxx_xxxx	STS-3/3xAU-3 payload in first STS-3 tributary group of first quadrant (Tributary index = [2])

Table 26. RX_PP_CONFIG_[20:0] Provisioning

RX_PP_CONFIG_[20:0][a]	Interpretation
. . .	
0_xxx0_xxxx_xxxx_xxxx_xxx1	STS-3c/AU-4 payload in fourth STS-3 tributary group of fourth quadrant (Tributary index = 16,1)
0_xxx0_xxxx_xxxx_xxxx_xxx0	STS-3/3xAU-3 payload in fourth STS-3 tributary group of fourth quadrant (Tributary index = [16])

[a] **RX_PP_CONCAT[20:0]** has a similar interpretation.

The received H1 and H2 bytes contain concatenation information. RHINE can be provisioned to ignore the state of the H1-H2 byte (as regards using the received concatenation information) by setting **PP_AUTO_CONFIG** = 0. Alternatively, the **RX_PP_CONFIG_[20:0]** register is ignored if **PP_AUTO_CONFIG** = 1. (See section 6.9.3.)

6.7.2 Pointer State Determination

Pointer state determination involves examining the 48 pairs of H1-H2 bytes to establish the state of each pair and from these states, determining which of the SPE/VCs are indicated as being concatenated by the received pointers. For tributaries in the Normal state (or transitioning to Normal), the path overhead offset indicated by the pointer value (PTR) carried within the appropriate H1-H2 byte pair is determined.

6.7.2.1 State Descriptions

Each of the 48 pairs of H1-H2 bytes are monitored and are considered to be in 1 of 4 states. These are

- Normal (NORM = 00)
- Alarm Indication Signal (AIS = 01)
- Loss of Pointer (LOP = 10)
- Concatenated (CONC = 11)

The individual states are stored in **PP_PTR_STATE_[x]_[j]_[1:0]**. The states of individual pairs of H1-H2 bytes are then combined to determine which received SPE/VCs are concatenated and to determine the state of the concatenated pointers.

6.7.2.2 Concatenated Pointer Determination

The **RX_PP_CONCAT_[20:0]** register contains the received signal concatenation configuration as indicated by the H1-H2 bytes. The bits in this register are interpreted in a manner similar to those in the **RX_PP_CONFIG_[20:0]** register. Transitions of bits in this register are driven directly from the individual H1-H2 states, **PP_PTR_STATE_[x]_[j]**. When a transition occurs, the corresponding **RX_PP_CONCAT_[20:0]_D** delta bit is set. The operation of the remainder of the Pointer Processor is influenced either by the **RX_PP_CONCAT** register (if **PP_AUTO_CONFIG** = 1) or by the provisioned configuration values, **RX_PP_CONFIG** (if **PP_AUTO_CONFIG** = 0). Based on one of these registers, the Pointer Processor establishes which of the SPE/VCs are concatenated and thus determines which pointer states to set **RX_PP_PAIS_[x]_[j]** and **RX_PP_LOP_[x]_[j]** register bits to, which pointer bytes to interpret (see section 6.10), and the kind of pointers that are generated (see section 6.7.4).

6.7.2.3 State of Concatenated Pointers

The device sets **RX_PP_PAIS_[x]_[j]** and **RX_PP_LOP_[x]_[j]** register bits that indicate the pointer state for the configured or automatically determined tributaries. (See section 6.9.3.)

Changes in **RX_PP_PAIS_[x]_[j]** and **RX_PP_LOP_[x]_[j]** state values are indicated by the **RX_PP_PAIS_[x]_[j]_D** and **RX_PP_LOP_[x]_[j]_D** delta bits.

6.7.3 Pointer Interpretation

6.7.3.1 Interpretation Rules

The H1-H2 byte pairs (or the first H1-H2 byte pair of concatenated tributaries) are interpreted to locate the start of the SPE/VC for that tributary, using SDH rules if **RX_SDH_PP_x** =1 or SONET rules if **RX_SDH_PP_x** =0. The SS bits are considered only in SDH mode (**RX_SDH_PP_x** =1) and **RX_PP_SS_EN_x** =1.

6.7.3.2 Justification Counters

Using these pointer interpretation rules, the Pointer Processor blocks determine which bytes belong to the SPE/VC and locate the start of the POH for each of the possible 16 tributaries.

The Pointer Processors contain 5-bit, pointer-interpreter, justification counters that count every positive or negative justification. When the performance-monitoring counters are latched, the values of these counters are latched to the **RX_PP_POSCNT_[x]_[j]_[4:0]** and **RX_PP_NEGCNT_[x]_[j]_[4:0]** registers, and the justification counters are cleared. (See section 4.2.)

If for tributary **x**, there has been at least 1 positive or negative justification in the previous second, the **RX_PP_POSCNT_[x]_[j]_SECE** or **RX_PP_NEGCNT_[x]_[j]_SECE** bit is set.

6.7.4 Pointer Generation

Based on the interpreted received pointers, the Pointer Processors write all SPE/VC bytes to FIFOs, eliminating the received TOH/SOH bytes — except for the K1 K2 bytes, which are passed to the APS Output Selector block. The SPE/VC bytes are written to the FIFOs using the Receive input clock, *RX_SONETCLK_x*. They are read from the FIFOs using the system reference input clock, *TX_SONETCLK_IN*, as they are needed to fill the SPE/VC bytes of SONET/SDH frames created by the Pointer Processors. (The TOH/SOH bytes in these frames are “dummy” bytes (except for H1-H3), as they are either overwritten later by the APS Frame Generator block or discarded.) If a FIFO is near its empty or full level, a positive or negative justification is created in the generated pointer.

6.7.4.1 Received K1 K2 Byte Passthrough

It is desirable in APS configurations to convey the received APS values from the standby RHINE to the working RHINE device. In order to support this function, the K1 K2 bytes from each receive line interface are transparently passed through the Pointer Generation block, and inserted into the corresponding K1 K2 byte position in the newly generated frame. The signal after the pointer processor block will be in STS-48 format; for non-STS-48 line side interfaces, the K1K2 bytes will be inserted into the unused OH bytes that correspond to the K1K2 position for that tributary. Due to slight differences that can occur in the clock rates, a K1 K2 byte value can be dropped/repeated in an occasional frame.

6.7.4.2 Frame Boundary Alignment

If the quadrants contain STS-3 interfaces, all 4 of the Pointer Processors within the quadrant are active, and 4 separate SONET/SDH frames are created. So that these can be multiplexed together with outputs from the other quadrants into 1 STS-48/STM-16 stream, the frame boundaries of these frames are created in alignment. The start-of-frame is related to the *TX_FRAME_IN* and it could be in a window depending on the mode being used (see sections 5.10.1).

There is the potential for timing ambiguities associated with the exact position of the *TX_FRAME_IN* input as it is retimed to RHINE's internal clock. Therefore, the period of the rising edge of the *TX_FRAME_IN* input can not be precisely 125 µs each and every period. RHINE will not rely on the position of the *TX_FRAME_IN* input on a frame-by-frame basis, but will instead 'resync' its Frame Generation circuitry to a single rising edge of *TX_FRAME_IN* when instructed to by the user. This resynchronization to the *TX_FRAME_IN* will occur when register **SYS_SYNC_IN_RESYNC** is written from 0 to 1. When this bit is written from 0 to 1, RHINE will resynchronize its internal frame position generator to the next rising edge of *TX_FRAME_IN*; RHINE's frame generator will then flywheel on this frame position, regardless of the *TX_FRAME_IN* signal, until instructed once more by the user to resync to *TX_FRAME_IN*.

When **SYS_SYNC_OUT_EN** = 1 and *GPIO0* is programmed as an output (see section 11.1.1.6), an output sync pulse is provided on *GPIO0*. This output sync pulse follows RHINE internal frame synchronization described above.

6.7.4.3 Pointer Generation Rules

Pointers are generated on the output based on incoming pointer values and justifications, frequency variations between input and reference clocks, and the fill level of internal FIFOs.

Except in the case of AIS generation, the SS bits in H1 are generated as "10" when **RX_SDH_PG_x** = 1, else they are generated as "00" (**RX_SDH_PG_x** = 0, the default).

6.7.4.4 FIFO Overflow or Underflow

If a FIFO overflows or underflows, the **RX_PG_FIFO_[x]_[j]_E** event bit is set.

6.7.4.5 PAIS Generation

PAIS for tributary **x** is generated by setting the H-bytes and the entire SPE/VC for tributary **x** to all 1s.

If (**RX_LOF_x** = 1 and **LOF_INH** = 0), **RX_LAIS_x** = 1, (**RX_LOS_x** = 1 and **RX_LOS_RDI_INH_x** = 0), or (**B2_ERR_SF_x** = 1 and **RX_SF_PAIS_INH_x** = 0); then PAIS is generated for the affected tributaries. If for example (**RX_LOF_[1]** = 1 and **LOF_INH** = 0), PAIS is generated for all tributaries input through the associated mux low speed signal (*RX_DATA_x* for an STS-3 or STS-12 signal, or *RX_DATA_[15:0]* for an STS-48 signal).

If **RX_PP_PAIS_[x]_[j]** or **RX_PP_LOP_[x]_[j]** is active RHINE inserts PAIS for tributary **[x]_[j]**. If **RX_FAST_AIS_[x]_[j]** = 1 and the last frame received for tributary **[x]_[j]** contains all 1s in its H-bytes, RHINE inserts all ones in the H bytes for tributary **x**. If the condition (H bytes received are set to all ones) lasts for 3 frames, then in addition to the H bytes, the payload is also forced to an all ones pattern. If **RX_FAST_AIS_[x]_[j]** = 0, then 3 consecutive frames with all ones in the H bytes needs to be received before PAIS condition is declared.

The user can force PAIS generation for tributary **[x]_[j]** by setting **RX_PAIS_GEN_[x]_[j]** = 1.

6.7.4.6 Unequipped Generation

The Pointer Generators can also insert Unequipped signal. If **RX_PAIS_GEN_[x]_[j]** = 0, and **RX_UNEQ_GEN_[x]_[j]** = 1, the entire SPE/VC, for tributary **[x]_[j]**, is generated with all 0s. The pointer value used for unequipped insertion must be a valid pointer value; the specific value is not specified. An NDF does not need to be generated when the unequipped signal insertion is removed (**RX_UNEQ_GEN_[x]_[j]** is cleared).

6.7.4.7 Justification Counters

The Pointer Processors contain 5-bit pointer generator justification counters that count every positive or negative justification on a per STS-1 basis. When the performance-monitoring counters are latched, the values of these counters are latched to the **RX_PG_POSCNT_[x]_[j]_[4:0]** and **RX_PG_NEGCNT_[x]_[j]_[4:0]** registers, and the justification counters are cleared. (See section 4.2.)

If for tributary **[x]_[j]**, there has been at least 1 positive or negative justification in the previous second, the **RX_PG_POSCNT_[x]_[j]_SECE** or **RX_PG_NEGCNT_[x]_[j]_SECE** bit is set.

6.8 Receive APS Port and Selector

6.8.1 Dual-Feed to APS Output Port

RHINE supports APS configurations through an APS interface. In the receive processing path, the received frames, which have now been multiplexed into a single STS-48 format, are dual fed at this point to the APS Output Selector block (see section 7), as well as to the Receive APS Selector block.

6.8.2 Receive APS Selector

To support APS switching between 2 RHINE devices, a selector is provided at this point in the receive processing. This selector can select data to be input to the RX Pointer Interpreter block from one of 2 possible sources; the output of the Receive Pointer Processor block of this RHINE, or the APS Input interface. The selection is controlled on a per-STS-3 tributary basis by register **RX_APS_SEL_x**. If **RX_APS_SEL_x** = 0, the data coming from the Receive Pointer Processor block of this RHINE is selected (the default). If **RX_APS_SEL_x** = 1, the data coming from the APS Input interface is selected.

For STS-48/STM-16 operation (**TX/RX_LINE_CONFIG[4]** = 1), all 16 **RX_APS_SEL_x** registers must be written to switch the entire signal. For STS-12/STM-4 configurations (**TX/RX_LINE_CONFIG[3:0]** = 1111), all 4 **RX_APS_SEL_x** registers corresponding to each STS-12/STM-4 tributary must be written.

6.9 RX Pointer Interpreter Configuration

The expected type of SONET/SDH received signal is provisioned through the **RX_LINE_CONFIG_[4:0]** and **RX_PI_CONFIG_[4:0]** registers. The expected SPE/VC types within the received STS/STM signals are determined by the **RX_PI_CONFIG_[4:0]** register, which is interpreted as shown in Table 27.

Note that this configuration register only handles STS-48c, STS-12c, and STS-3c tributaries. RHINE does not terminate tributaries below an STS-3c; STS-3 tributaries must be 'hairpinned' at the SONET level. At this point in the transmission path, any STS-3 tributary will be detected as being in the Loss of Pointer state, and be replaced by PAIS.

Table 27. RX_PI_CONFIG_[4:0] Values

RX_LINE_CONFIG_[4:0]	RX_PI_CONFIG_[4:0]	Contents of Tributary x
1 XXXX	1 XXXX	STS-48c/AU-4-16c.
1 XXXX	0 1XXX	First quadrant of STS-48/STM-16 contains 1 STS-12c/AU-4-4c.
1 XXXX	0 0XXX	First quadrant of STS-48/STM-16 contains 4 STS-3c/AU-4s.
0 1XXX	X 1XXX	First STS-12/STM-4 contains 1 STS-12c/AU-4-4c.
0 1XXX	X 0XXX	First STS-12/STM-4 contains 4 STS-3c/AU-4s.
0 0XXX	XXXX	First quadrant of RHINE outputs 4 STS-3c/AU-4s.
1 XXXX	0 X1XX	Second quadrant of STS-48/STM-16 contains 1 STS-12c/AU-4-4c.
1 XXXX	0 X0XX	Second quadrant of STS-48/STM-16 contains 4 STS-3c/AU-4s.
0 X1XX	X X1XX	Second STS-12/STM-16 contains 1 STS-12c/AU-4-4c.
0 X1XX	X X0XX	Second STS-12/STM-4 contains 4 STS-3c/AU-4s.
0 X0XX	X XXXX	Second quadrant of RHINE outputs 4 STS-3c/AU-4s.
1 XXXX	0 XX1X	Third quadrant of STS-48/STM-16 contains a single STS-12c/AU-4-4c.
1 XXXX	0 XX0X	Third quadrant of STS-48/STM-16 contains 4 STS-3c/AU-4s.
0 XX1X	X XX1X	Third STS-12/STM-4 contains a single STS-12c/AU-4-4c.
0 XX1X	X XX0X	Third STS-12/STM-4 contains 4 STS-3c/AU-4s.
0 XX0X	X XXXX	Third quadrant of RHINE outputs 4 STS-3c/AU-4s.
1 XXXX	0 XXX1	Fourth quadrant of STS-48/STM-16 contains a single STS-12c/AU-4-4c.
1 XXXX	0 XXX0	Fourth quadrant of STS-48/STM-16 contains 4 STS-3c/AU-4s.
0 XXX1	X XXX1	Fourth STS-12/STM-4 contains a single STS-12c/AU-4-4c.
0 XXX1	X XXX0	Fourth STS-12/STM-4 contains 4 STS-3c/AU-4s.
0 XXX0	X XXXX	Fourth quadrant of RHINE outputs 4 STS-3c/AU-4s.

The received H1 and H2 bytes contain SPE/VC concatenation information. RHINE can be provisioned to ignore the state of the H1-H2 byte (as regards using the received concatenation information) by setting **PI_AUTO_CONFIG** = 0. Alternatively, the **RX_PI_CONFIG_[4:0]** register is ignored if **PI_AUTO_CONFIG** = 1. (See section 6.9.3.)

6.9.1 Pointer State Determination

Pointer state determination involves examining the 48 pairs of H1-H2 bytes to establish the state of each pair, and from these states, determining which of the SPE/VCs are indicated as being concatenated by the received pointers. For tributaries in the Normal state (or transitioning to Normal), the J1 byte offset indicated by the pointer value (PTR) carried within the appropriate H1-H2 byte pair is determined.

6.9.2 State Transition Rules

Each of the 48 pairs of H1-H2 bytes are monitored and are considered to be in 1 of 4 states. These are

- Normal (NORM = 00)
- Alarm Indication Signal (AIS = 01)
- Loss of Pointer (LOP = 10)
- Concatenated (CONC = 11)

The individual states are stored in **PI_PTR_STATE_[x]_[j]_[1:0]**. The states of individual pairs of H1-H2 bytes are then combined to determine which received SPE/VCs are concatenated and to determine the state of the concatenated pointers.

6.9.3 Concatenated Pointer Determination

The **RX_PI_CONCAT_[4:0]** register contains the received signal configuration as indicated by the H1-H2 bytes. The bits in this register are interpreted as the **RX_PI_CONFIG_[4:0]** registers. (See Table 27.)

Transitions of bits in this register are driven directly from the individual H1-H2 states, **PI_PTR_STATE_[x]_[j]**. When a transition occurs, the corresponding **RX_PI_CONCAT_[4:0]_D** delta bit is set.

6.9.4 PAIS and LOP Indicators

The device supplies the register state bits **RX_PI_PAIS_x** and **RX_PI_LOP_x** that indicate the pointer state for the configured or automatically determined tributaries. (See section 6.9.2.) The values of these registers are as follows:

For STS-48c/AU-4-16c operation:

For x = [1]:

RX_PI_PAIS_[1] = 1 if **PI_PTR_STATE_m_j_[0]** AND !**PI_PTR_STATE_m_j_[1]** for m = 1 to 16, j = 1 to 3.

RX_PI_LOP_[1] = 1 if **RX_PI_PAIS_[1]** = 0 AND (**PI_PTR_STATE_[1]_[1]_[1:0]** != 00 OR **PI_PTR_STATE_m_j_[1:0]** != 11) for all m, j != 1,1.

For x = [2] through [16], **RX_PI_PAIS_x** and **RX_PI_LOP_x** = are undefined.

For STS-12c/AU-4-4c operation:

For x = [1], [5], [9], and/or [13]:

RX_PI_PAIS_x = 1 if **PI_PTR_STATE_m_j_[0]** AND !**PI_PTR_STATE_m_j_[1]** for m = x to x+3, j = 1 to 3.

RX_PI_LOP_x = 1 if **RX_PI_PAIS_x** = 0 AND (**PI_PTR_STATE_x_[1]_[1:0]** != 00 OR

PTR_STATE_m_j_[1:0] != 11) for m = x, j = 2,3 and m = x+1 to x+3, j = 1 to 3.

For x = [2] to [4], [6] to [8], [10] to [12] and/or [14] to [16], **RX_PI_PAIS_x** and **RX_PI_LOP_x** = are undefined.

For STS-3c/AU-4 operation:

For x = [1] to [16]:

RX_PI_PAIS_x = **1** if **PI_PTR_STATE_[x]_[j]_[0]** AND !**PI_PTR_STATE_[x]_[j]_[1]** for j = 1 to 3.

RX_PI_LOP_x = **1** if **RX_PI_PAIS_x** = 0 AND (**PI_PTR_STATE_x_[1]_[1:0]** != 00 OR **PI_PTR_STATE_[x]_[j]_[1:0]** != 11) for j = 2,3.

The **RX_PI_PAIS_x** and **RX_PI_LOP_x** signals contribute to the Path Remote Defect Indication (PRDI). (See section 5.5.2.4.) Changes in these state values are indicated by the **RX_PI_PAIS_x_D** and **RX_PI_LOP_x_D** delta bits.

6.10 Pointer Interpretation

The first H1-H2 byte pair of concatenated tributaries (and AU-4s) is interpreted to locate the start of the SPE/VC for that tributary. The rules for pointer interpretation are:

1. During normal operation, the pointer locates the start of the SPE/VC.
2. Any variation from the current accepted pointer is ignored unless a consistent new value is received 3 times consecutively, or it is preceded by rules 3, 4, or 5. Any consistent new value received 3 times consecutively overrides rule 3 or 4.
3. For **RX_SDH_PI_x** = 0,

   if at least 3 out of 4 of the NDF bits match the disabled indication (0110) and at least 8 out of 10 of the pointer value bits match the current accepted pointer with its I-bits inverted, a positive justification is indicated. The byte following the H3 byte is considered a positive stuff byte, and the current accepted pointer value is incremented by 1 (mod 783).

   For **RX_SDH_PI_x** = 1,

   if at least 3 out of 4 of the NDF bits match the disabled indication (0110), 3 or more of the pointer value I-bits and 2 or fewer of the pointer value D-bits match the current accepted pointer with all its bits inverted, and either the received SS bits are 10 or **RX_PI_SS_EN_x** = 0, a positive justification is indicated. The byte following the H3 byte is considered a positive stuff byte, and the current accepted pointer value is incremented by 1 (mod 783).
4. For **RX_SDH_PI_x** = 0,

   if at least 3 out of 4 of the NDF bits match the disabled indication (0110) and at least 8 out of 10 of the pointer value bits match the current accepted pointer with its D-bits inverted, a negative justification is indicated. The H3 byte is considered a negative stuff byte (it is part of the SPE), and the current accepted pointer value is decremented by 1 (mod 783).

   For **RX_SDH_PI_x** = 1,

   if at least 3 out of 4 of the NDF bits match the disabled indication (0110), 3 or more of the pointer value D-bits and 2 or fewer of the pointer value I-bits match the current accepted pointer with all its bits inverted, and either the received SS bits are 10 or **RX_PI_SS_EN_x** = 0, a negative justification is indicated. The H3 byte is considered a negative stuff byte (it is part of the VC), and the current accepted pointer value is decremented by 1 (mod 783).
5. For **RX_SDH_PI_x** = 0,

   if at least 3 out of 4 of the NDF bits match the enabled indication (1001), and the pointer value is between 0 and 782, the received pointer replaces the current accepted pointer value.

For **RX_SDH_PI_x** = 1,

if at least 3 out of 4 of the NDF bits match the enabled indication (1001), the pointer value is between 0 and 782, and either the received SS bits are 10 or **RX_PI_SS_EN_x** = 0, the received pointer replaces the current accepted pointer value.

Using these pointer interpretation rules, the Pointer Interpreter block determines the POH location and provides indication signals for SPE/VC, POH, J1, B3, C2, and G1 bytes for each of the possible 16 tributaries. These internal signals are needed by the remaining blocks to complete the receive processing.

6.11 Path Overhead Monitoring

The POH monitoring block consists of J1, B3, C2, and G1 monitoring for each tributary. These POH bytes are monitored for errors or changes in state.

6.11.1 Path Trace (J1) Capture/Monitor

As with J1 insertion, RHINE supports 2 methods of Path Trace (J1) capture. The first, typically used in SONET applications, captures 64-consecutive J1 bytes in the SONET/SDH path overhead. The second, used in SDH applications, looks for a repeating 16-consecutive, J1-byte pattern. When it has detected a consistent 16-byte pattern for 3-consecutive instances, the J1 pattern is stored in designated registers.

6.11.1.1 SONET J1 Monitoring

When **RX_SDH_J1** = 0 (SONET mode), RHINE can be provisioned to capture a sample of the path-trace message for a specified tributary. The **J1_CAP_TRIB_[3:0]** register, in conjunction with **RX_PI_CONFIG_[4:0]** (see Table 28) specifies the capture tributary. The 4 bits of **J1_CAP_TRIB_[3:0]** are interpreted as described in Table 28. When **J1_CAP** transitions from 0 to 1, RHINE captures the next 64-consecutive J1 bytes from the specified tributary and writes them to **RX_J1_[63:0]_[7:0]**.

The user is responsible for ensuring that the setting of **J1_CAP_TRIB_[3:0]** is aligned with the configuration of the device (**RX_LINE_CONFIG_[4:0]** and **RX_PI_CONFIG_[4:0]**). The behavior of the J1 capture function is undefined when invalid values for the current configuration are used.

Table 28. J1_CAP_TRIB Usage

J1_CAP_TRIB_[3:0]	J1 Byte Captured from this Tributary
0000	STS-48c/AU-4-16c or STS-12c/AU-4-4c #1 or STS-3c/AU-4 #1
0001	STS-3c/AU-4 #2
0010	STS-3c/AU-4 #3
0011	STS-3c/AU-4 #4
0100	STS-12c/AU-4-4c #5 or STS-3c/AU-4 #5
0101	STS-3c/AU-4 #6
0110	STS-3c/AU-4 #7
0111	STS-3c/AU-4 #8
1000	STS-12c/AU-4-4c #9 or STS-3c/AU-4 #9
1001	STS-3c/AU-4 #10
1010	STS-3c/AU-4 #11
1011	STS-3c/AU-4 #12
1100	STS-12c/AU-4-4c #13 or STS-3c/AU-4 #13
1101	STS-3c/AU-4 #14
1110	STS-3c/AU-4 #15
1111	STS-3c/AU-4 #16

No path-trace frame structure is defined for SONET, but GR-253 does recommend that the 64-byte sequence consist of a string of ASCII characters padded out to 62 bytes with NULL characters (00) and terminated with <CR> (0D) and <LF> (0A) bytes. If the **J1_CRLF** bit is set, RHINE captures the first 64-byte string it receives in the J1 byte position that ends with {0D, 0A}. RHINE captures the next 64-J1 bytes. On completion of the capture, RHINE sets the **J1_CAP_E** event bit.

6.11.1.2 SDH J1 Monitoring

In the **RX_SDH_J1** = 1 case, the J1 bytes are expected to contain a repeating 16-byte, path-trace frame that includes the PAPI. J1 monitoring is done for the tributary specified in the **J1_CAP_TRIB_[3:0]** register, but the **J1_CAP**, **J1_CAP_E** and **J1_CRLF** bits are not used. J1 monitoring consists of locking on to the start of the 16-byte, path-trace frame and examining the received path-trace frames for values that match consistently for 3-consecutive, path-trace frames. When a consistent frame value is received, it is written to **RX_J1_[15:0]_[7:0]**. The first byte of the path-trace frame (which contains the frame start marker) is written to **RX_J1_[15]_[7:0]**.

Framing. The MSBs of all path-trace frame bytes are 0, except for the MSB of the frame start-marker byte. The J1 monitor framer searches for a J1 byte with a 1 in its MSB, followed by 15-consecutive, J1 bytes that have a 0 in their MSB. When this pattern is found, the framer goes into frame, **J1_OOF** = 0. Once the J1 monitor framer is in-frame, it remains in frame until 3-consecutive, path-trace frames are received with at least 1 MSB bit error. (In the SONET mode, the J1 frame indication is held in the out-of-frame state, **J1_OOF** = 1.) The **J1_OOF_D** delta bit is set when **J1_OOF** changes state.

Pattern Acceptance and Comparison. Once in frame, the J1 monitor block looks for 3-consecutive, 16-byte path-trace frames. When first monitoring a tributary, once the initial 3-consecutive, identical frames are received, the accepted frame is stored in **RX_J1_[15:0]_[7:0]**. Once this initial value has been accepted, subsequent accepted frames are compared to the previous one. When a subsequent value is stored, the **RX_J1_D** delta bit is set .

6.11.2 BIP-8 (B3) Checking

RHINE checks the received B3 bytes for correct BIP-8 values. Even parity BIP-8 is calculated over all bits in the SPE/VC (including the POH) each frame. These values are then compared to the B3 values received in the following frame. The comparison can result in from 0 to 8 mismatches (B3 bit errors). This value can be inserted into the Transmit side G1 byte (see section 5.5.2.4.).

For each of the possibly 16 received SPEs RHINE contains a 16-bit, B3 error counter that either counts every B3 bit error (if **BIT_BLKCNT** = 0) or every frame with at least 1 B3 bit error (if **BIT_BLKCNT** = 1). When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of this counter is latched to the **B3_ERRCNT_x_[15:0]** register, and the B3 error counter is cleared. (See section 4.2.)

If there has been at least 1 B3 error since the last rising edge of LATCH_EVENT*,* then the B3 error second event bit, **B3ERR_x_SECE**, is set.

6.11.3 Signal Label (C2) Monitoring

The received C2 bytes are monitored so that reception of the correct type of tributary can be verified. Each tributary is checked for 5-consecutive frames with identical C2 byte values. When a consistent C2 value is received, the accepted value is written to **RX_C2_x_[7:0]**. **RX_C2_x_D** delta bit is set when a new C2 value is accepted.

The expected value of the received C2 bytes is provisioned in **EXP_C2_x_[7:0]**. If the current accepted value does not match the expected value, and the accepted value is NOT

- the all 0s Unequipped label
- the 01(hex) Equipped - non-specific label
- the FC(hex) payload defect label
- the FF(hex) reserved label

then the Payload Label Mismatch register bit, **RX_PLM_x**, is set high.

If the current accepted value is the all 0s Unequipped label, and the provisioned **EXP_C2_x_[7:0]** != 00(hex), then the Unequipped register bit, **RX_UNEQ_x**, is set high.

The **RX_PLM_x** and **RX_UNEQ_x** signals contribute to the PRDI signal. (See section 6.11.4.2.) When **RX_PLM_x** or **RX_UNEQ_x** change state, the **RX_PLM_x_D** or the **RX_UNEQ_x_D** delta bit is set.

6.11.4 Path Status (G1) Monitoring

6.11.4.1 Path REI Monitoring

Bits 1 through 4 (the 4 MSBs) of the path status byte indicate the number of B3 errors that were detected by the remote terminal in its received SPE/VC signal. Only the binary values between 0 and 8 are legitimate. If a value greater than 8 is received, it is interpreted as 0 errors (as is specified in GR-253 and G.707). RHINE contains a 16-bit G1 error counter for each tributary that either counts every error indicated by G1 (if **BIT_BLKCNT** = 0) or every frame received with the first 4 bits of G1 that are not equal to 0 (if **BIT_BLKCNT** = 1). When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **G1_ERRCNT_x_[15:0]** registers, and the G1 error counter is cleared. (See section 4.2.)

If there has been at least 1 received G1 error indication since the last rising edge of LATCH_EVENT*,* then the G1 error second event bit, **G1ERR_x_SECE**, is set.

6.11.4.2 Path RDI Monitoring

RHINE can be provisioned to monitor bit 5 of G1 (RDI-P indicator), if **RX_PRDI5** = 1; or bits 5, 6 and 7 of G1 (enhanced RDI-P indicator), if **RX_PRDI5** = 0. Monitoring consists of checking for **G1_CONSEC_[3:0]** consecutive received values of the monitored bit(s) that are identical (**G1_CONSEC_[3:0]** = 0000 is interpreted as 16). When a consistent value is received, bits 5, 6 and 7 of G1 are written to **RX_G1_x_[2:0]**. Accepted values are compared to the previous contents of this register. (All 3 bits are written, but if **RX_PRDI5** = 1, only G1 bit 5 and **RX_G1_x_[2]** are involved in the comparisons.) When a new value is stored, the **RX_G1_x_D** delta bit is set.

6.11.5 Other POH Bytes

The remaining POH bytes are not monitored by RHINE. These include the path user channel (F2), the position indicator (H4), the path growth/user channel (Z3/F3), the path growth/path APS channel (Z4/K3), and the tandem connection monitoring (Z5/N1) bytes.

6.12 Receive Payload Descrambling

After the payload is extracted from the SONET/SDH signal, the payload data in tributary x is descrambled using a self-synchronizing $X^{43} + 1$ descrambler. In all modes, register **RX_DSCR_INH_x** controls the operation of the descrambler on a per-tributary basis. When **RX_DSCR_INH_x** = 0 (the default), the descrambler for tributary x is enabled. When **RX_DSCR_INH_x** = 1, operation of this descrambler is inhibited. The post-HDLC descrambler cannot be enabled if this descrambler is enabled; if the post-HDLC descrambler for tributary x is to be used, **RX_DSCR_INH_x** must be set to 1.

RHINE provides 16 self-synchronizing descramblers, one for each tributary, based on the following generator polynomial:

$$X^{43} + 1$$

The descrambler is illustrated in the following figure:

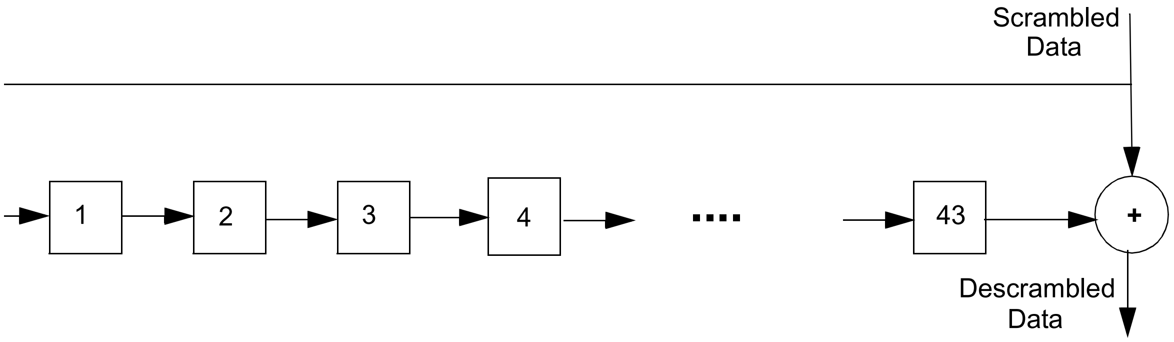


Figure 14. Payload Descrambler

6.12.1 ATM Descrambler Operation

In ATM operation (**RX_POS_x** = 0), the operation of the descrambler adheres to the following requirements:

- The descrambler operates on the bits of the information fields only.
- During the 5-octet header, the descrambler operation is suspended and the descrambler state retained.

In order to meet these requirements, the ATM Processing block provides cell delineation information to the descrambler.

6.12.2 POS and Direct Mapping Descrambler Operation

In POS and direct mapping mode (**RX_POS_x** = 1 or **RX_DIRECT_MAP_x** = 1), the descrambler operates on the C-4-Xc, and therefore does not descramble the POH and fixed stuff bytes. The result is descrambling of the entire SPE payload after HDLC processing, including interframe fill flags.

6.13 Receive HDLC Processing

At this point the SPE has been extracted from the SONET/SDH frame and is passed on to either the HDLC or ATM processor for further processing. In POS mode (**RX_POS_x** = 1), the HDLC processing provides the extraction of packets from the SPE.

6.13.1 Direct Mapping of Data from SPE

In the direct mapping mode (**RX_DIRECT_MAP_x** = 1), data passes through the HDLC processing block, before being output over RHINE Receive system interface. Most HDLC functions are disabled in this mode of operation, with the exception of the Transparency processing (optional) and RHINE system interface, which operates slightly differently in this mode (that is, *RX_SOC/P_y, RX_ERR_y, RX_LBYTE,* and *RX_EOP_y* are disabled).

6.13.2 HDLC Framer

In POS mode (**RX_POS_x** = 1), HDLC frames are extracted from the SPE payload by identifying the Flag Sequence (0x7e) that begins/ends a frame.

RHINE examines each octet of the payload. When an octet with bit pattern 0x7e is discovered, RHINE recognizes this as the start/end of a packet. The octets that follow this Flag Sequence are then examined. If these are also 0x7e, they are Flag Sequences used to fill the Inter-Packet gap, and are discarded. The first octet NOT equal to 0x7e that follows the initial Flag Sequence is considered the first octet of the HDLC frame.

After the start-of-frame flag, RHINE continues to examine each octet of the payload for the Flag Sequence. If it locates the bit pattern 0x7e and the immediately preceding octet is Control Escape (0x7d), the frame is aborted. (See section 6.13.4.) Otherwise, a normal end of the current frame is declared.

6.13.3 Removal of Transparency Byte Stuffing

6.13.3.1 POS Mode

In POS mode (**RX_POS_x** = 1, **RX_DIRECT_MAP_x** = 0) following the HDLC framer, RHINE reverses the transparency byte stuffing process specified in section 5.2.6 to recover the original packet stream. Figure 15 contains a flow chart describing the transparency reversal process. The FIFO underflow byte sequence, which can be inserted by the Transmit side during periods of FIFO underflow, will be detected and removed during the transparency processing if **RX_POS_FIFOUNDER_MODE_x** = 1. The default value is disabled, **RX_POS_FIFOUNDER_MODE_x** = 0. The special FIFO underflow byte code is programmed using register **RX_POS_FIFOUNDER_BYTE_[7:0]**.

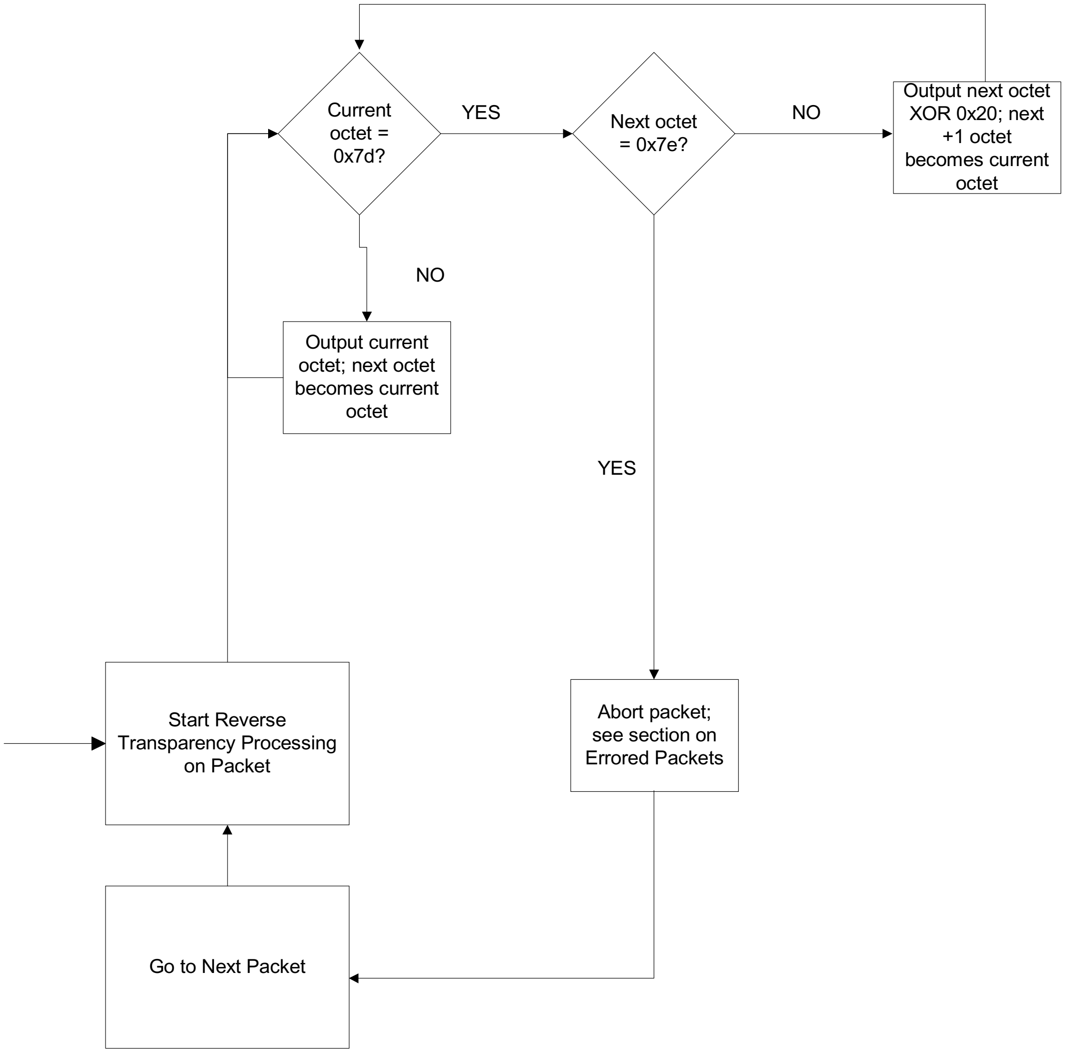
6.13.3.2 Direct Map Mode

In direct mapping mode (**RX_DIRECT_MAP_x** = 1), the transparency processing is controlled by **RX_POS_FIFOUNDER_MODE_x.** If **RX_POS_FIFOUNDER_MODE_x** = 1, transparency byte destuffing will be performed with the exception that 0x7e bytes that follow 0x7d bytes do not cause “aborts,” and they are treated as all other bytes (that is, they are output as 0x5e). If **RX_POS_FIFOUNDER_MODE_x** = 0, transparency byte destuffing is disabled.

6.13.3.3 Underflow Byte Removal

In either direct map or POS mode, if **RX_POS_FIFOUNDER_MODE_x** = 1, bytes that match the FIFO underflow byte code (**RX_POS_FIFOUNDER_BYTE_[7:0]**) are discarded if they are not immediately preceded by the Control Escape code (0x7d).

Figure 15: Removal of Transparency Byte Stuffing


Current octet = 0x7d?
YES
Next octet = 0x7e?
NO
Output next octet XOR 0x20; next +1 octet becomes current octet
NO
Output current octet; next octet becomes current octet
YES
Start Reverse Transparency Processing on Packet
Abort packet; see section on Errored Packets
Go to Next Packet

6.13.4 Errored Frames

In POS mode (**RX_POS_x** = 1), a special byte code (0x7d7e) is utilized to indicate that a frame has been aborted. If this byte code is received, the frame that contains it is aborted. If the packet is transmitted to the Link Layer device, it is marked as errored (that is, *RX_ERR_y* in section 9.0).

For each of the 16 possible tributaries, RHINE contains an 8-bit error counter that counts every packet in which the abort sequence is detected. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **RX_POS_PABORT_ERRCNT_x_[7:0]** registers, and the packet abort error counters are cleared. (See section 4.2.)

If there has been at least 1 packet abort error since the last rising edge of LATCH_EVENT, then the packet abort error second event bit, **RX_POS_PABORT_ERR_x_SECE**, is set.

As an alternative, a packet can also be aborted by inverting the FCS bytes. This will appear to RHINE Receive HDLC/ATM Processor as simply an FCS error, and is handled as described in the following section.

RHINE also, as an option, views a packet as being errored and marks it accordingly if it violates minimum or maximum packet sizes. The packet sizes refer to the size of the packets output from RHINE only, and do not include the dropped flag sequence, address, control, transparency, FIFO underflow and FCS bytes. These minimum and maximum sizes are programmable via the management interface. Register **RX_POS_PMIN_[3:0]** contains the minimum packet size. The default value of this register is 0.

Register **RX_POS_PMAX_[15:0]** contains the maximum packet size. The default value of this register is 0x05DE (RFC 1661, pg 4).

RHINE disables/enables minimum and maximum size-packet-checking when instructed to through the management interface. Registers **RX_POS_PMIN_ENB_x** and **RX_POS_PMAX_ENB_x** (both default to 0) control how violations of the minimum and maximum packet sizes are handled on a per-tributary basis. When either is set to 1, any violation of the corresponding packet size restriction within tributary x is marked as errored.

RHINE contains two 8-bit error counters for each tributary that count every violation of the maximum and minimum packet size limits. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **RX_POS_PMIN_ERRCNT_x_[7:0]** and **RX_POS_PMAX_ERRCNT_x_[7:0]** registers, and the packet size violation counters are cleared. (See section 4.2.)

If there has been at least 1, packet-size-violation error since the last rising edge of LATCH_EVENT, then the appropriate packet-size-violation second event-bit, **RX_POS_PMIN_ERR_x_SECE** or **RX_POS_PMAX_ERR_x_SECE**, is set.

6.13.5 Frame Check Sequence (FCS) Field

In POS mode (**RX_POS_x** = 1), an FCS is then calculated and checked against the FCS bytes at the end of each frame. This option is controlled by register **RX_POS_FCS_INH**. A value of **RX_POS_FCS_INH** = 0 enables the FCS in all tributaries. A value of **RX_POS_FCS_INH** = 1 disables it. Two types of FCS have been defined in RFC 1662, a 16-bit check sequence (FCS-16) and a 32-bit check sequence (FCS-32). The device supports both types, again on a per-tributary basis. **RX_POS_FCS_MODE_x** = 0 places the device in FCS-32 mode, and is the default. **RX_POS_FCS_MODE_x** = 1 places the device in FCS-16 mode.

RHINE provides FCS-16 functionality, using the following generator polynomial:

$$1 + X^5 + X^{12} + X^{16}$$

RHINE provides FCS-32 functionality, using the following generator polynomial:

$$1 + X + X^2 + X^4 + X^5 + X^7 + X^8 + X^{10} + X^{11} + X^{12} + X^{16} + X^{22} + X^{23} + X^{26} + X^{32}$$

The FCS field is calculated over all bits in the frame, except the Flag Sequence, any added FIFO underflow bytes, and the FCS field itself. The details of the FCS implementation are specified in section 5.2.5. The Receive side operates just as the Transmit side.

If **RX_POS_FCS_BIT_ORDR[1:4]** = 0x0000 (the default), the received data is read into the shift register in big-endian bit order (MSB first). If **RX_POS_FCS_BIT_ORDR[1:4]** = 0x1111, the received data is read into the shift register in little-endian bit order (LSB first). In either case, the data is restored to big-endian order for processing after the FCS calculation. The four bits of the this field control each of the respective 4 quadrants.

NOTE: For most applications, **RX_POS_FCS_BIT_ORDR[1:4]** should be set to 0x1111 for standards-compliant operation.

The resulting FCS is compared against the value in the received FCS field. If an error is detected, the management interface is notified, the appropriate counter incremented, and the last word of the packet is marked as errored in the FIFO. RHINE contains a 22-bit FCS error counter for each tributary that counts every FCS violation within the tributary. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **RX_POS_FCS_ERRCNT_X_[21:0]** registers, and the FCS error counters are cleared. (See section 4.2.)

If there has been at least 1 FCS error since the last rising edge of LATCH_EVENT*,* then the FCS error second event bit, **RX_POS_FCS_ERR_x_SECE**, is set.

Following FCS checking, the FCS bytes are terminated (they are not stored in the FIFO). If the FCS checking is disabled via the management interface, the last 2 or 4 bytes are sent on to the FIFO.

Should an FCS error be detected, the packet is marked as errored (*RX_ERR_y*) when transmitted to the Link Layer device. (See section 9.0)

6.13.6 HDLC Frame Termination

In POS mode (**RX_POS_x** = 1), after FCS calculation, the HDLC bytes (in the following sections) are monitored and optionally terminated.

6.13.6.1 Flag Sequence

All occurrences of the Flag Sequence, used for frame delineation and inter-frame fill purposes are deleted. The start-of frame and end-of-frame information is retained by RHINE and transmitted to the Link Layer via the *RX_SOC/P_y* and *RX_EOP_y* signals.

6.13.6.2 Address and Control Bytes

The address and control bytes (the first 2 bytes of the HDLC frame following the Flag Sequence) are monitored on a per-tributary basis by RHINE. Monitoring consists of checking for the user provisioned values of the Address and Control fields. These values are provisioned on a per-tributary basis via registers **RX_POS_ADDRESS_x_[7:0]** and **RX_POS_CONTROL_x_[7:0]**. If no match is detected, this field is assumed to be compressed and was not sent. If invalid values are detected, these 2 bytes are not dropped and are passed on the Link Layer via the system interface. The management interface is notified of the detection of invalid Address and Control fields by setting **RX_POS_ADRCTL_INVALID_x** = 1. Changes in the state of **RX_POS_ADRCTL_INVALID_x** are indicated by setting its corresponding delta bit **RX_POS_ADRCTL_INVALID_x_D** to 1.

If valid Address and Control fields are detected, RHINE terminates these 2 bytes, and does not pass them on to the RX FIFO. The deletion of valid address and control bytes can be inhibited on a per-tributary basis by setting **RX_POS_ADRCTL_DROP_INH_x** = 1. The default value of this register is 0 (automatic drop enabled).

6.13.6.3 FCS Bytes

As mentioned in the FCS section, the 2 or 4 FCS bytes are also terminated by RHINE. If the FCS checking is disabled via the management interface (**RX_POS_FCS_INH** = 1), this termination is also disabled, and the last 2 or 4 bytes in the HDLC frame are sent to the Link Layer.

6.13.7 Post-HDLC Descrambling

For both direct mapping and POS modes, following the HDLC processing, the data can be descrambled using a self-synchronizing $X^{43} + 1$ descrambler. This is provided to undo the optional pre-HDLC scrambler in the transmit direction. Register **RX_POSTHDLC_DSCR_INH_x** controls the operation of these descramblers on a per-tributary basis. When **RX_POSTHDLC_DSCR_INH_x** = 0, the descrambler for tributary x is enabled. When **RX_POSTHDLC_DSCR_INH_x** = 1 (the default), operation of the descrambler x is inhibited. Note that this descrambler cannot be enabled simultaneously with the enabling of the payload descrambler. If **RX_DSCR_INH** = 0, the post-HDLC scrambler is disabled, regardless of the value of **RX_POSTHDLC_DSCR_INH_x**.

RHINE provides 16 post-HDLC, self-synchronizing descramblers, one for each tributary, based on the following generator polynomial:

$$X^{43} + 1$$

This descrambler is described in section 6.12.

In POS mode (**TX_POS_x** = 1), this descrambles the entire packet stream of tributary x that will be output over the system interface. This descrambling therefore does not include any flag sequences, transparency processing bytes, FIFO underflow indication bytes, or any other bytes that are terminated within RHINE in the HDLC processing block. In direct data mode (**TX_DIRECT_MAP_x** = 1), this descrambles the entire direct data stream of tributary x, again, not including any FIFO underflow or transparency processing bytes terminated in RHINE.

6.13.8 Receive Valid Packet Count

For each tributary, RHINE contains a packet counter that counts every valid packet received from the SONET/SDH line. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **RX_POS_PKT_CNT_x_[22:0]** registers, and the packet counters are cleared. (See section 4.2.)

6.14 Receive ATM Processing

In ATM mode (**RX_POS_x** = 0), the ATM processor provides the extraction of ATM cells from the SPE.

6.14.1 SPE Payload Deconstruction

6.14.1.1 ATM Cells from an STS-48c/AU-4-16c

The bit rate available for the ATM cells (user information cells, signalling cells, OAM cells, unassigned cells and cells used for cell rate decoupling) is 2396.16 Mbit/s.

For STS-48c/STM-16 signals, RHINE removes the path overhead and fixed stuff bytes from the VC-4-16c, producing a C-4-16c container. The ATM cell boundaries are aligned with the STM-16 octet boundaries. Since the C-4-16c capacity (37440 octets) is not an integer multiple of the cell length (53 octets), a cell can cross a C-4-16c boundary.

6.14.1.2 ATM Cells from an STS-12c/AU-4-4c

The bit rate available for the ATM cells (user information cells, signalling cells, OAM cells, unassigned cells and cells used for cell rate decoupling) is 599 040 kbit/s.

For STS-12c/STM-4 signals, RHINE removes the path overhead and fixed stuff bytes from the VC-4-4c, producing a C-4-4c container. The ATM cell boundaries are aligned with the STM-4 octet boundaries. Since the C-4-4c capacity (9360 octets) is not an integer multiple of the cell length (53 octets), a cell can cross a C-4-4c boundary.

6.14.1.3 ATM Cells from an STS-3c/AU-4

The bit rate available for the ATM cells (user information cells, signalling cells, OAM cells, unassigned cells and cells used for cell rate decoupling) excluding SONET overhead packets is 149 760 kbit/s.

For STS-3c/STM-1 signals, RHINE removes the path overhead from the VC-4, producing a C-4 container (VC-4 without the POH row). The ATM cell boundaries are aligned with the STM-1 octet boundaries. Since the C-4 capacity (2340 octets) is not an integer multiple of the cell length (53 octets), a cell can cross a C-4 boundary.

6.14.2 ATM Cell Delineation

The cell delineation process is the procedure by which the ATM cell boundaries are identified. The HEC field is used by RHINE to achieve cell delineation, according to the algorithm defined in ITU-T I.432.1.

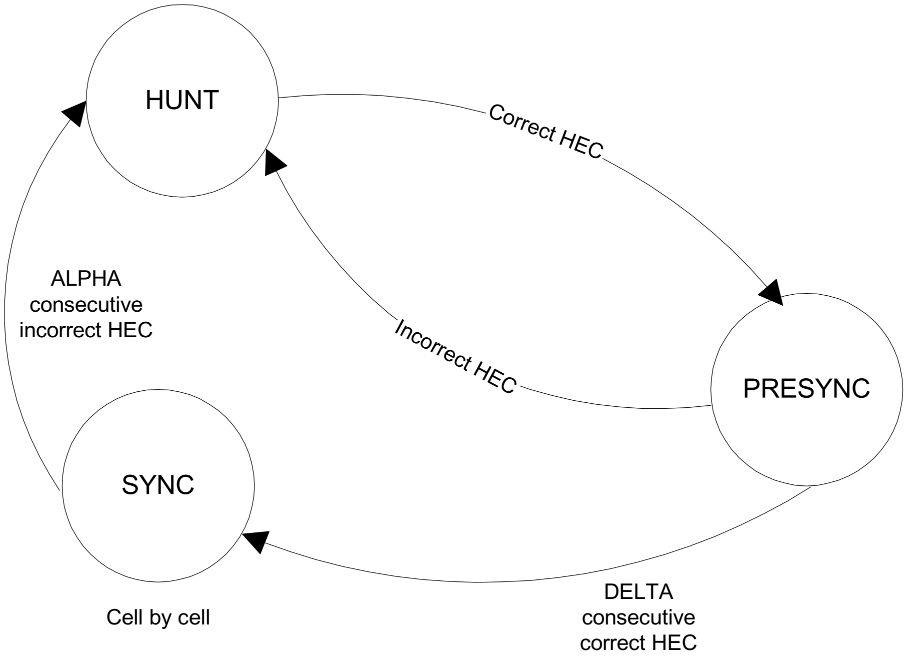
6.14.2.1 Cell Delineation Algorithm

Cell delineation is performed by using the correlation between the header bits to be protected (32 bits) and the relevant control bits (8 bits) introduced in the header by the HEC using a shortened cyclic code with generating polynomial $X^8 + X^2 + X + 1$. (See section 5.3.4 for detailed requirement on CRC calculation.)

RHINE takes the 4 ATM header bytes (or 4-consecutive octets, if it is in the HUNT state) and uses the generating polynomial to recalculate the HEC sequence. This recalculated sequence is then modulo 2 added to the received HEC byte. The result is referred to as the “syndrome”. RHINE also supports the optional modulo 2 addition of the byte pattern 01010101 to the received HEC byte, prior to the comparison (modulo 2 addition) to the recalculated sequence. The option is configurable via the management interface.

Figure 16 shows the state diagram of the HEC cell delineation method.

Figure 16. Cell Delineation State Diagram


HUNT
Correct HEC
ALPHA consecutive incorrect HEC
Incorrect HEC
PRESYNC
SYNC
Cell by cell
DELTA consecutive correct HEC

NOTE - The "correct HEC" means the header has no bit error (syndrome is zero) and has not been corrected.

The details of the state diagram are described as follows:

1) In the HUNT state, the delineation process is performed by checking octet by octet for the correct HEC (i.e. syndrome equals zero) for the assumed header field. When such an agreement is found, it is assumed that one header has been found, and the process enters the PRESYNC state.

2) In the PRESYNC state, the delineation process is performed by checking cell by cell for the correct HEC. The process repeats until the correct HEC has been confirmed DELTA times consecutively, at which point the process moves the SYNC state. If an incorrect HEC is found, the process returns to the HUNT state. The total number of consecutive correct HEC required to move from the HUNT state to the SYNC state is therefore DELTA + 1.

3) In the SYNC state, the cell delineation is assumed to be lost if an incorrect HEC is obtained ALPHA times consecutively.

4) Cell with correct HECs (or cell headers with 1-bit errors which are corrected) that are processed while in the SYNC state are passed to the ATM layer.

RHINE uses an ALPHA of 7 and a DELTA of 6, as defined in [12].

Register **RX_ATM_HEC_ENH** enables/disables mod2 addition of alternating 01010101 pattern to HEC calculation. A value of 1 (the default) enables; a value of 0 disables it.

6.14.2.2 Cell delineation signals

ITU Recommendation I.432.2 describes 2-cell delineation signals to be provided for performance-monitoring purposes. RHINE provides both of these performance indicators through the management interface.

Out of Cell Delineation (OCD). An OCD anomaly occurs when the cell delineation process changes from SYNC state to HUNT state while in a working state. (See Figure 16). An OCD anomaly terminates when the PRESYNC to SYNC state transition occurs. (See Figure 16.)

An OCD anomaly is indicated by setting **RX_ATM_OCD_x** = 1. This register is cleared when the OCD terminates (see the previous paragraph). The **RX_ATM_OCD_x_D** delta bit contributes to the summary interrupt, and the **RX_ATM_OCD_x_SECE** second event bits are set at the end of each second that the **RX_ATM_OCD_x** is in the active state at any time during the second.

Loss of Cell Delineation (LCD). An LCD defect occurs when an OCD anomaly (see above) has persisted for *y* ms., where *y* is in the range 0 to 4 (RHINE actually goes beyond this requirement, up to 7.875 ms.) and is provisioned via the register **RX_ATM_LCD_TIME_[5:0]**. An **RX_ATM_LCD_TIME** = 000 000 corresponds to 0 ms., **RX_ATM_LCD_TIME** = 100 000 corresponds to 4 ms. Increasing **RX_ATM_LCD_TIME** by 1 corresponds to a time increase of 125 µs. An LCD defect terminates when the cell delineation process enters and remains in the SYNC state for *y* continuous milliseconds.

An LCD defect is indicated by setting **RX_ATM_LCD_x** = 1. This register is cleared when the LCD terminates. The **RX_ATM_LCD_x_D** delta bit contributes to the summary interrupt, and the **RX_ATM_LCD_x_SECE** second event bits are set at the end of each second that the **RX_ATM_LCD_x** is in the active state at any time during the second.

6.14.3 Header Error Control Functions

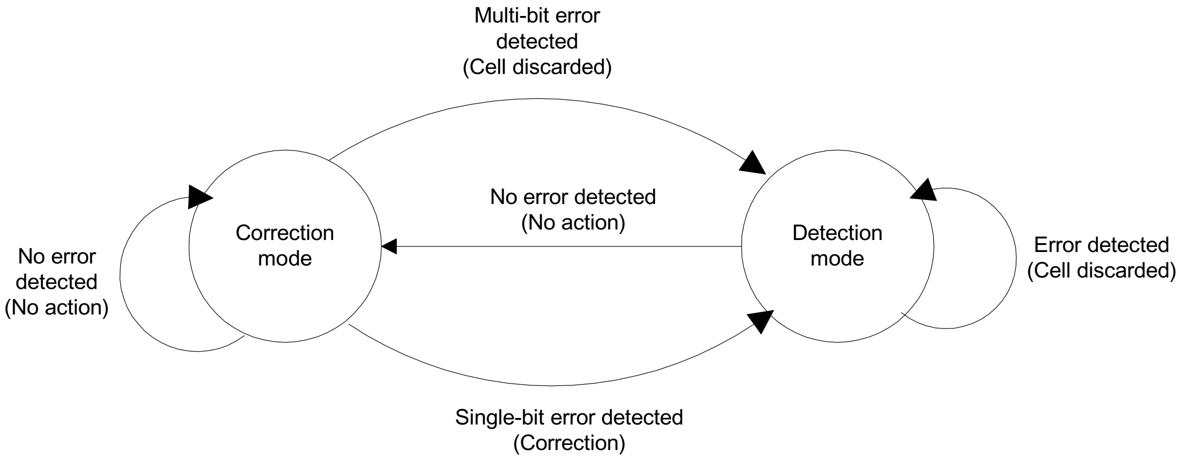
The following error processing procedure is performed by RHINE whenever it is in SYNC state.

The HEC byte covers the entire cell header. The code used for this function is capable of either:

- Single-bit error-correction
- Multiple-bit error-detection

The detailed description of the HEC procedure is given in section 5.3.4 and section 6.14.2. As described in section 5.3.4, the transmitting side computes the HEC field value. RHINE supports the 2 modes of operation shown in Figure 17. The default mode provides for single-bit error correction. Each cell header is examined, and if an error is detected, one of 2 actions takes place. The action taken depends on the state of the receiver. In “correction mode” only 1-bit errors can be corrected and the receiver switches to “detection mode.” In “detection mode,” all cells with detected header errors are discarded. When a header is examined and found not to be in error, the receiver switches to “correction mode.” The term “no action” in Figure 17 means no correction is performed and no cell is discarded.

Figure 17. HEC: Receiver Mode of Operation



RHINE also provides an HEC detection-only mode, where single-error correction is disabled. RHINE remains in the detection mode shown in Figure 17, and discards all cells with errored headers. This mode is enabled on a per-tributary basis by setting **RX_ATM_HEC_INH_x** = 1. The default value for this register is 0 (single-error correction, enabled).

RHINE furnishes performance-monitoring registers for each tributary to provide visibility into the Header Error Control Processing. The number of single-bit errors that are corrected by the HEC processing is provided in **RX_ATM_HEC_CORR_x_[17:0]**. The number of cells that are discarded due to errored Headers is provided in **RX_ATM_HEC_DROP_x_[17:0]**. These counters are large enough to count the expected number of error events received in a 1-second period during an error rate of 10^{-3}. The **RX_ATM_HEC_CORR_x_SECE** and **RX_ATM_HEC_DROP_x_SECE** second events are set whenever their corresponding counter registers are non-0 during a LATCH_EVENT transition from low to high.

6.14.4 ATM Idle Cell Removal

RHINE deletes all idle cells; they are not sent to the ATM Layer device. RHINE provides the user with the ability to specify the GFC, PTI, and CLP fields of the idle cell header. (See section 5.3.3 for the structure of the ATM cell header and the location of the user programmable fields.) The registers used to program these fields on the Receive side are **RX_ATM_IDLE_GFC_[3:0]**, **RX_ATM_IDLE_PTI_[2:0]**, and **RX_ATM_IDLE_CLP**. The default values for these registers are the same as the defaults for the corresponding registers on the Transmit side.

RHINE locates the user-defined, idle, cell header (all non-HEC header field bits 0 except those defined by **RX_ATM_IDLE_GFC_[3:0]**, **RX_ATM_IDLE_PTI_[2:0]**, and **RX_ATM_IDLE_CLP**) within the incoming data stream, and drops any cell with the idle cell header.

RHINE also provides the capability for the user to monitor the contents of the data bytes of the idle cells. A

4-bit tributary indicator, **RX_ATM_IDLE_TRIB_[3:0]** is provided to designate the tributary from which RHINE shall capture idle cell data. This register follows the same structure used for the receive J1 byte capture tributary selector. (See Table 28.) In addition, a byte indicator is provided (**RX_ATM_IDLE_BYTE_[5:0]**) to indicate which byte of the ATM idle cell payload is to be monitored. This register is interpreted as directly identifying the byte position in the idle cell payload; values 1 through 48 indicate the first through 48th byte of the idle payload, other values are undefined. When **RX_ATM_IDLE_CAP** is written from 0 to 1, RHINE captures the next ATM idle cell it detects in the indicated tributary and payload byte position. The value of the captured idle cell is written into **RX_ATM_IDLE_DATA[7:0]**.

6.14.5 Receive Valid Cell Count

RHINE contains 16 ATM cell counters that count every valid non-idle ATM cell received from the SONET/SDH line. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the appropriate **RX_ATM_CELL_CNT_x_[22:0]** register, and the ATM cell counters are cleared. (See section 4.2.) The counters have been sized to count a minimum of 1 full second of cells.

6.15 Receive FIFO Interface

6.15.1 System Side Cell/Packet Loopback

RHINE provides the capability for the user to loop back the cells/packets received via the system interface. When **SYS_T_TO_R_LOOP_x** = 1, the cells/packets received from the ATM or Link Layer device are routed from the transmit FIFO directly to the receive FIFO, and output back to the ATM or Link Layer device that originated the cell data. When **SYS_T_TO_R_LOOP_x** is set to 0, the cells/packets/direct data received within the SONET/SDH line signals are transmitted to the receive FIFOs and then out the system interface.

6.15.2 FIFO Processing

RHINE writes cells/packets/direct data into the FIFO in preparation for output via the Receive system interface to the ATM or Link Layer device. Receive FIFOs [1], [5], [9], and [13] handle the tributaries, which depending on the configuration, can carry STS-48c/AU-4-16c or STS-12c/AU-4-4c tributaries. When configured for STS-48c/AU-4-16c or STS-12c/AU-4-4c operation, these FIFOs are 1024-octets deep. FIFOs that are configured to handle STS-3c/AU-4 tributaries are 256-octets deep. Along with the cell/packet/data, the following indicators, when applicable, must accompany each word in the FIFO: start-of-cell/packet; end-of-packet; if end-of-packet; how many octets in word; and whether or not the packet is errored. When an error is detected in a cell/packet, no further bytes from that cell/packet are loaded into the FIFO. In the POS mode (**RX_POS_x** = 1), the last word of the packet is marked as errored (*RX_ERR_y*).

The state of the FIFOs are monitored by RHINE. FIFO overflow events are reported to the management interface by setting **RX_FIFOERR_x_E** = 1. The occurrence of a FIFO overflow also causes the appropriate performance-monitoring counter to be incremented.

For each of the 16 possible tributaries, RHINE contains an 8-bit FIFO overflow error counter that counts every cell/packet affected by a FIFO overflow event. When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **RX_FIFOERR_CNT_x_[7:0]** register, and the FIFO overflow error counters are cleared. (See section 4.2.)

If there has been at least 1 FIFO overflow event since the last rising edge of LATCH_EVENT, then the FIFO overflow error event bit, **RX_FIFOERR_x_SECE**, is set.

This FIFO immediately precedes the Receive System FlexBus-3^{TM} interface. Its purpose is to perform the rate matching function between the SONET clock domain and the Link Layer clock domain.

6.15.3 Errored Cell/Packet Handling

In ATM mode (**RX_POS_x** = 0), as with the transmit direction, RHINE will not forward ATM cells that have been corrupted by FIFO overflow events. These corrupted cells are deleted by RHINE.

In POS mode (**RX_POS_x** = 1), RHINE will mark as errored any packets that have been corrupted by FIFO overflow events, using *RX_ERR_y*. Other errored packet handling procedures in POS mode are described in section 6.13.4.

6.15.4 Receive Data Parity

As per the Utopia 3 specification and FlexBus-3^{TM} extensions, RHINE provides a parity check bit that accompanies each data word (*RX_SYS_DAT_[31:0]*) transmitted to the ATM or Link Layer device. This parity check bit is present on pin *RX_PRTY_y.* This bit provides an odd parity check as a default (**RX_PRTY_MODE_y** = 0). Even parity is provided if **RX_PRTY_MODE_y** = 1.

7.0 APS Output Processing

7.1 APS Output Selection

Both the transmit and receive data paths in RHINE are tapped off at certain points in their processing path and passed to the APS Output Processing block. The Transmit data path is tapped off before the Frame Generation block, passed through a mux stage to form an STS-48 format signal and then sent to the APS Output selection block. The receive data path is tapped off after the Pointer Processor block, just prior to the RX APS selection block. The APS Output selection block selects one of these 2 data paths, on a per-tributary basis to output from RHINE via the APS Output interface (*APS_DATA_OUT* and *APS_CLK_OUT*). This selection is controlled by registers **APS_OUT_SEL_x**. If **APS_OUT_SEL_x** = 0 (the default), the Transmit data path for tributary x is selected to be output via the APS Output interface. If **APS_OUT_SEL_x** = 1, the receive data path for tributary x is selected to be output via this interface. For STS-48/STM-16 operation (**TX/RX_LINE_CONFIG[4]** = 1), all 16 **APS_OUT_SEL_x** registers must be written to switch the entire signal. For STS-12/STM-4 configurations (**TX/RX_LINE_CONFIG[3:0]** = 1111), all 4 **APS_OUT_SEL_x** registers corresponding to each STS-12/STM-4 tributary must be written.

7.2 APS Frame Generation

The APS Output Processing block creates an STS-48/STM-16 signal for transporting all 16 tributaries between RHINE devices over the APS interface. Given that in the Transmit direction all SPEs have been created referenced to a common reference clock, and in the Receive direction the incoming tributaries have been pointer processed to reference them to a common reference clock, the APS interface can be an STS-48/STM-16 signal, regardless of the line configurations.

To provide for framing and basic performance monitoring over the APS interface, an APS Frame Generation block generates a subset of the Transport (Section) Overhead (TOH/SOH) bytes and scrambles all bytes of the APS signal except for the first row of TOH/SOH bytes. The structure of an STS-48/STM-16 is shown in Figure 10.

7.2.1 APS TOH/SOH Generation

The APS Output Processing block generates A1, A2, and B1 bytes for the APS Output interface signal. This signal is also scrambled, so it can be serialized and sent without a clock to a deserializer and clock recovery circuit on another board.

7.2.1.1 Frame Bytes (A1 and A2)

The frame bytes are normally generated with the following fixed patterns:

- A1: 1111_0110 = F6
- A2: 0010_1000 = 28

7.2.1.2 Section BIP-8 (B1)

The B1 Bit Interleaved Parity 8 (BIP-8) is transmitted as even parity (normal) if **APS_B1_INV** = 0. Otherwise, odd parity (incorrect) is generated. The BIP-8 is calculated over all bits of the previous SONET/SDH frame after scrambling and placed in the B1 byte of the current frame before scrambling.

By definition of BIP-8, the first bit of B1 provides parity over the first bit of all bytes of the previous frame, the second bit of B1 provides parity over the second bit of all bytes of the previous frame, and so on.

7.2.2 K1K2 Insert

Certain 1+1 APS configurations have a single RHINE dual feeding its transmit data through itself as well as a second RHINE device. In order to expedite K1K2 insertion in this type of configuration, it is useful to source the K1K2 APS byte information for both signals through a single RHINE. In order to facilitate this, the APS message is inserted into the APS Output interface signal of the Working RHINE device, and carried in-band to the mate Stand-by RHINE device.

If **APS_OUT_SEL_x** = 0 (transmit data passes out the APS interface), the K1K2 byte information from line x is sourced from the register map (**APS_K1_GEN_x_[7:0]** and **APS_K2_GEN_x_[7:0]**). The APS output signal is in STS-48 format; for non-STS-48 line side interfaces, the K1K2 bytes will be inserted in the unused OH bytes that correspond to the K1K2 position for that tributary. Due to slight differences that can occur in the clock rates, a K1 K2 byte value can be dropped/repeated in an occasional frame.

If **APS_OUT_SEL_x** = 1 (receive data passes out the APS interface), the K1K2 byte information for line x arriving at the APS Output processing block/selector is not modified, but passed through transparently. This allows the working RHINE mate device to monitor the APS messages from the standby RHINE.

7.2.3 Scrambling

The APS Output signal is then scrambled with a frame synchronous scrambling sequence generated from the polynomial g(x) = x7 + x6 + 1. The scrambling is done in parallel, but the result is equivalent to the serial scrambler shown in Figure 13. The scrambler is initialized to 1111111 at the beginning of the first SPE/VC byte (the byte in row 1, column 145), and it scrambles the entire SONET/SDH signal except for the first row of TOH/SOH. For testing purposes, the scrambler can be disabled by setting the **APS_SCRINH** bit to 1.

7.3 APS Output Format

The APS Output signal is then output through the APS Output interface. The format of the signal at this point is a 4 byte-wide STS-48/STM-16 signal, with a single reference clock, *APS_CLK_OUT*.

8.0 APS Input Processing

8.1 APS Input Format

The format of the APS Input signal is a 4-byte wide, STS-48/STM-16 signal, with four clock signals, one for each byte. If **APS_IF_MODE**=0, a single clock signal is used for the entire 4 bytes of the APS Input Signal. If **APS_IF_MODE**=1, four clock signals are used, for each byte of the APS Input Signal. The four clock signals will be the same frequency, as they were sourced from a single clock reference in the transmitting RHINE device. The **APS_IF_MODE**=1 configuration is provided to allow the user to transfer the STS-48 APS signal between boards as four, 622.08 MHz, pseudo-STS-12 signals. As a result, there can be some skew between the start of the frames on each of the four input bytes. In order to realign the data, RHINE provides four separate framers to frame on the each of the four pseudo-STS-12s. These framers are followed by a small buffer that is used to realign the four pseudo-STS-12s into a single STS-48 frame prior to the remainder of the overhead processing.

8.2 SONET/SDH Framers

The SONET/SDH framers locate the framing bytes in the selected data signals, and by doing so are able to find byte alignment and determine the position of all TOH/SOH bytes. After finding frame, the framers shift the data so that their output data is byte aligned. They then align the frames between the four pseudo-STS-12 inputs, descramble the data and perform B1 monitoring on the reconstructed STS-48.

For both values of **APS_IF_MODE**, all 4 framers described in this section are active.

When the framer state machine is out-of-frame (**APS_OOF_y** = 1), it searches for the 32-bit, A1-A1-A2-A2 framing-byte sequence of 0xF6F6_2828. This pattern can start on any of the 8-input data lines and span up to 5-input bytes. When the framer finds 2-successive sequences separated in time by 125 µs that exactly match the framing pattern, it goes into frame (**APS_OOF_y** = 0) and byte aligns its output data bus.

The framer remains in-frame, until it receives 5-successive frames with at least 1-bit error in the A1-A1-A2-A2 framing pattern. When this occurs, **APS_OOF_y** is set to 1, and a new frame search is begun.

The **APS_OOF_y_D** delta bits contribute to the summary interrupt.

8.2.1 APS Input Deskew and Start of Frame Alignment

The APS channel can interface to an external SERDES device to transmit an STS-48 data stream as four separate pseudo-STS-12 streams. Due to path length variations as well as variable delay in the external SERDES devices used to serialize the data stream, the frame start positions of the four pseudo-STS-12 tributaries may no longer be aligned. RHINE deskews these four data streams and reconstructs the original STS-48 signal. RHINE provides a 15 byte buffer for deskew.

When **APS_OOF_[1:4]** deasserts for all four streams, the alignment buffer (15 bytes deep) measures the time between the first and last arrival of frame boundaries. If it exceeds 13 clock cycles, **APS_MISALIGN_ERR_E** asserts. Otherwise, the alignment buffer compensates for the different arrival times and outputs all the frame boundaries simultaneously. If any **APS_OOF_y** asserts, the alignment buffer will output zeros until alignment can be reestablished.

In some protection configurations, some primary and some protection (APS) inputs may be routed to the path termination processing or transmit output. These inputs are selected in transmit and receive direction as described in section 5.7.2 and section 6.8.2, respectively. In applications where a mix of primary and protection inputs are selected, Rhine will dynamically accommodate either -4 to 6 or 3 to 13 78MHz clock cycles difference between the start of frame of the APS and primary RX inputs. When register **APS_DELAY_EN** = 1, the difference may be 3 to 13 clock cycles (38 to 167 ns); when **APS_DELAY_EN**

= 0 (the default), the difference may be -4 to 6 clock cycles. A difference of -4 clock cycles implies the start of frame of the APS input occurs 4 78MHz clock cycles prior to the start of frame of the primary RX input.

8.2.2 Descrambling

Before the data is output from the Framer block, it can be descrambled using the same frame synchronous sequence that is used to scramble the transmit data. (See section 5.9.4.) The descrambler is reset to 1111111 at the beginning of the first SPE/VC byte (the byte in row 1, column 145), and it descrambles the entire SONET/SDH signal except for the first row of TOH/SOH. For testing purposes, the descrambler for the APS Input signal can be disabled by setting **APS_DSCRINH** to 1.

8.2.3 B1 Monitor

RHINE checks the received B1 byte for correct Bit Interleaved Parity 8 (BIP-8) values. Even parity BIP-8 is calculated over all bytes of each frame before descrambling. This value is then compared to the received B1 value in the following frame after descrambling. The comparison can result in from 0 to 8 mismatches (B1 bit errors).

RHINE contains a single, 16-bit B1 error counter that either counts every B1 bit error (if **BIT_BLKCNT** = 0) or every frame with at least 1 B1 bit error (if **BIT_BLKCNT** = 1). When the performance-monitoring counters are latched (LATCH_EVENT transitions high), the value of these counters are latched to the **APS_B1_ERRCNT_[15:0]** register, and the B1 error counter is cleared. (See section 4.2.)

If there has been at least 1 B1 error since the last rising edge of LATCH_EVENT, then the APS B1 error second event bit, **APS_B1ERR_SECE**, is set.

8.2.4 K1 K2 APS Monitor

If the K1 byte and the 5 MSBs of the K2 byte, which are used for sending APS requests and channel numbers, are received identically for 3-consecutive frames, their values are written to **APS_K1_x_[7:0]** and **APS_K2_x_[7:3]**. Accepted values are compared to the previous contents of these registers, and when a new 13-bit value is stored, the **APS_K1_x_D** delta bit is set.

The APS bytes (K1 and the 5 MSBs of the K2 byte) are checked for instability. If, for 12 successive frames, no 3-consecutive frames are received with identical APS bytes, the **APS_UNSTAB_x** bit is set. It is cleared when 3-consecutive, identical, K1 bytes are received. When **APS_UNSTAB_x** changes state, the **APS_UNSTAB_x_D** delta bit is set.

Bits 2 down to 0 of K2 can contain APS mode information. These bits are monitored for **APS_CONSEC_[3:0]** consecutive, identical values. **APS_K2_x_[2:0]** is written when this occurs, unless the value of bits 2 and 1 of K2 is “11” (indicating Line/MS AIS or RDI, section 6.6.3.1). The **APS_K2_x_D** delta bit is set when a new value is written to **APS_K2_x_[2:0]**.

The delta bits associated with APS monitors, **APS_K1_x_D, APS_K2_x_D,** and **APS_UNSTAB_x_D** all contribute to an APS interrupt signal, APS_INTB. In addition, these deltas also contribute to the standard summary interrupt signal, INTB. (See section 10.1 and section 11.2.1.2.)

8.3 APS Input Data

After framing, descrambling and B1 verification, the APS Input data path is then sent to 2 locations in RHINE, where a selector is placed which can, when properly set, replace RHINE internal data path with the APS Input data path. One of these locations is the TX APS Selector block, which lies immediately after the SPE Generation block, just prior to the TOH/SOH Generation block. The second location is the RX APS Selector block, which lies after the RX Pointer Processor, just prior to the RX Pointer Interpreter.

8.4 APS I/O Disable

Primarily for power savings, but also to reduce external EMI and internal noise, all primary data I/O (including their associated clocks) have disable capability. Outputs are gated (prior to retiming) to keep at a static level. LVTTL inputs will have internal pulldowns; for maximum power savings, the accompanying clock inputs should also be disabled.

APS_OUT_INH disables *APS_DATA_OUT_[31:0]* and *APS_CLK_OUT* (LVTTL).

APS_IN_INH disables *APS_DATA_IN_[31:0]* and *APS_CLK_IN[1:4]* (LVTTL).

All inhibits are active high, and default to zero (all inputs and outputs enabled).

9.0 RHINE System Interface Requirements

RHINE interface provides 3 slightly different system interfaces, to accommodate the demands of various applications. A Utopia Level 3 compliant interface is provided when the device is performing ATM cell extraction from the SONET SPE.

For packet applications, the Utopia Level 3 interface is extended to provide the additional functionality necessary to transfer packets to/from a PHY layer device. This later interface is based closely on the Utopia Level 3 interface, with modifications required to support variable length packets.

A subset of the FlexBus-3^{TM} interface is provided for applications where raw data is mapped directly to the SONET/SDH SPE. In this last case, the packet interface remains primarily intact, with the exception that the *SOC*/*P_y*, *EOP_y*, *LBYTE,* and *ERR_y* signals are disabled.

The FlexBus-3^{TM} interface supports Multi-PHY operation, as defined by Utopia Level 3, in which multiple PHY ports are accessed by the ATM or Link Layer device over a common Multi-PHY bus. Communication over this bus is point-to-point. The FlexBus-3^{TM} interface also supports an extension to Utopia Level 3, which provides four, 8-bit, buses operating in parallel. The quad, 8-bit, bus operation has been proposed to the ATM Forum as part of the Utopia Level 3 standard; see [4]. These 4 individual buses are point-to-point and communicate with 4 separate ATM or Link Layer devices. The FlexBus-3^{TM} interface provides 3-bit address buses in both the TX and RX direction for quad, 8-bit, bus operation, to allow the use of the quad bus for channelized OC-12 applications.

The Utopia Level 2 and Level 3 specifications describe single-PHY and multi-PHY operation. Single-PHY is defined as when a single ATM layer device transmits data to/from a single PHY layer device containing 1 or more ports. Multi-PHY operation is defined as when a single ATM layer device transfers data to/from 1 or more PHY layer devices, containing 1 or more ports. This particular distinction operates well for Utopia Level 2, but needs some modification to be precisely applicable for Utopia Level 3. We are electing to replace these definitions with new ones. These new definitions do not alter the actual performance of the FlexBus-3^{TM} interface; this change is being made only for clarification purposes. For the purposes of this specification, we will operate using the following new definitions: first, 'single-port' bus operation, in which 1 ATM layer-device transmits data to/from a single PHY layer port, this port can be the only port on a PHY layer device; or a PHY layer device can support multiple ports, each using 'single-port' buses to communicate to its own ATM layer device. No PHY layer addressing is required in single-port operation, as each bus only serves 1 port. Additionally, we define 'multi-port' bus mode, wherein a single ATM Layer device communicates to a single PHY layer device containing multiple ports over a single bus. Multi-port operation requires PHY layer addressing in order to differentiate the ports.

9.1 Conventions

RHINE system interface supports several operational modes. In certain applications, RHINE provides a single bus between 1 ATM/Link Layer device and 1 or more PHY-Layer ports. In other applications, RHINE provides 4 separate parallel buses between 4 ATM/Link Layer devices and 4 or more PHY-Layer ports (which are located in the same PHY Layer device). In the former case, a single set of control signals is all that is needed. In the latter case, a total of 4 sets of control signals, 1 for each individual bus, is required. In the following descriptions, a **'y'** is used in the names of these control signals to indicate, in a generic way, that there are in fact multiple instantiations of this signal. The value of **'y'** ranges from [1] to [4].

RHINE system interface where data flows from the ATM or Link Layer device to RHINE is labeled the Transmit interface. RHINE system interface where data flows from RHINE to the ATM or Link Layer is labeled the Receive interface.

The byte numbering convention used with this specification is [31:0] or [7:0], where [31] or [7] refers to the MSB.

All signals are active high, except for the *TX_ENB_y* and *RX_ENB_y* signals, which are active low.

9.2 RHINE System Interface - Basic Concepts

9.2.1 FlexBus-3 Modes of Operation

These specifications define several different modes of operation

:

Table 29. Operational Modes

Name	Functionality	Design Implications
Single-port or Multi-port	Single: A single, ATM layer device transmits data to/from a single, PHY layer port.	No PHY addressing is required.
	Multi: A single, ATM layer device transfers data to/from multiple, PHY ports on a single, PHY device via a single bus.	PHY addressing is required to distinguish PHY ports.
Direct status or multiplexed status (polled mode)	Direct: Each port provides both TX and RX status signals to the ATM/Link Layer, which it uses to indicate the fill status of its FIFO. The fill level of the port is always available to the ATM/Link Layer.	1. Each PHY port provides its own FIFO status signal to ATM or Link Layer. 2. Port FIFO status signals driven continuously.
	Multiplexed: Multiple PHY ports have their TX and RX status indications multiplexed to 1 status signal. The ATM/Link Layer device requests that a specific port place its status indication on the multiplexed status signal.	PHY ports share FIFO status signal to ATM/Link Layer - ATM/Link Layer device polls to arbitrate access to status signal.
Single, 32-bit, bus mode or quad, 8-bit, bus mode	Single, data bus, 32-bits wide.	Single set of control signals, except possibly for CLAV.
	Quad buses: 1 or more PHY ports, 1 ATM/Link Layer device for each 8-bit data bus.	Multiple SOC/P, ENB, and PRTY signals needed.

RHINE supports single-port, direct status polling for both the single, 32-bit, bus operation as well as a quad, 8-bit, bus mode. RHINE interface supports multi-port bus operation with direct status polling for the 32-bit bus, as well as multiplexed status polling for both single, 32-bit, and quad, 8-bit, bus operation. Table 30 illustrates RHINE modes supported by RHINE.

Table 30. FlexBus-3^{TM} Operational Modes Supported by RHINE

Single-port		Multi-port			
Direct		Direct		Multiplexed	
Single, 32-bit bus	Quad, 8-bit buses	Single, 32-bit bus	Quad, 8-bit buses	Single, 32-bit bus	Quad, 8-bit buses
Yes	Yes	Yes	No	Yes	Yes

The following figures illustrate the various FlexBus-3^{TM} bus modes supported by RHINE.

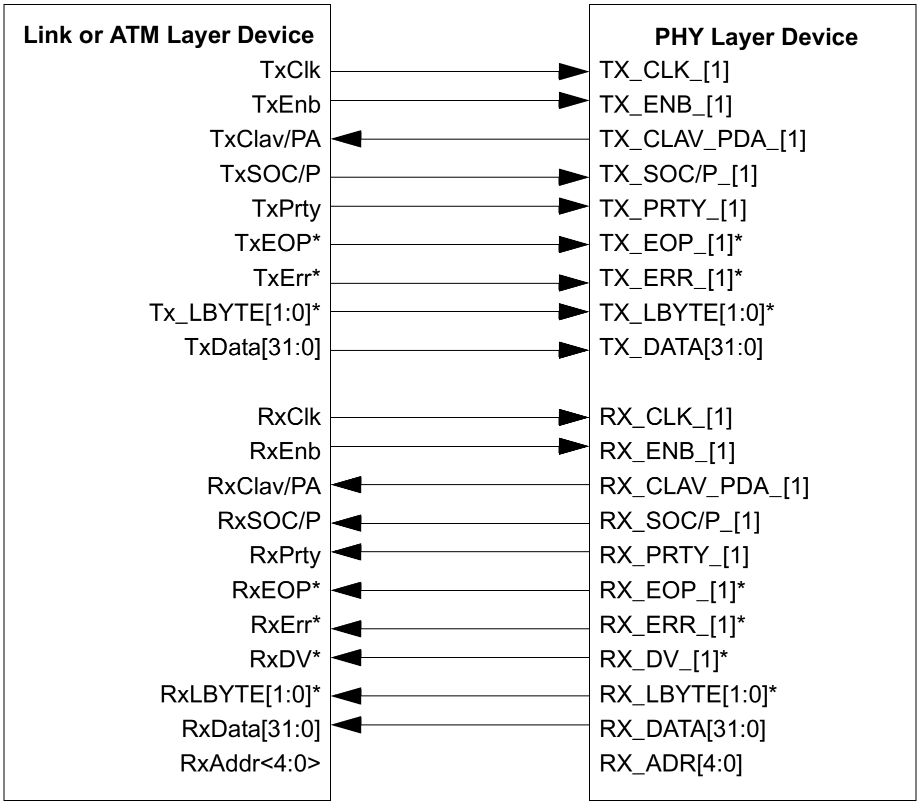


Figure 18: Single-Port, 32-Bit, Bus Operation

* Designates those signals that are extensions to standard Utopia operation to accommodate packet transfers.

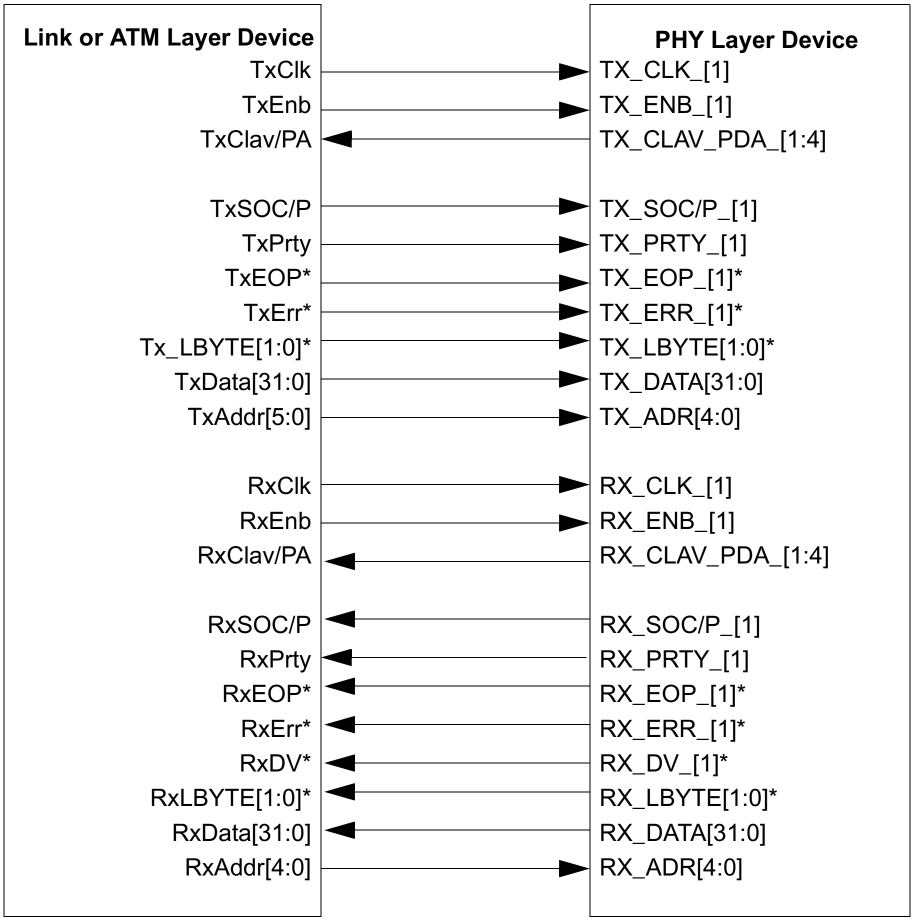


Figure 19: Multi-Port, 32-Bit Bus Operation, Direct Status

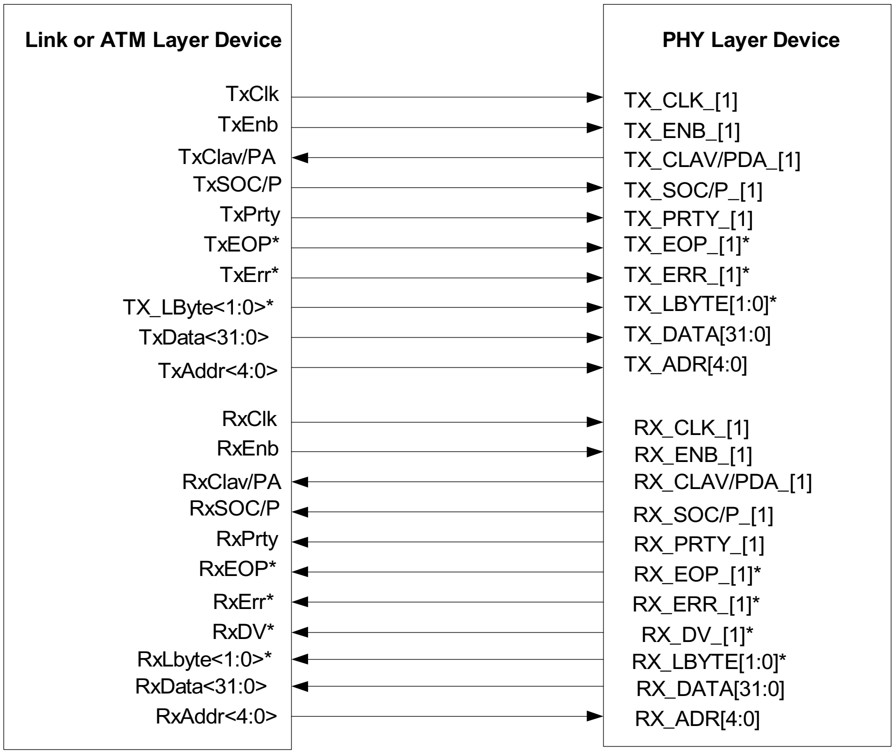


Figure 20: Multi-Port, 32-Bit Bus Operation, Multiplexed-Polling Status

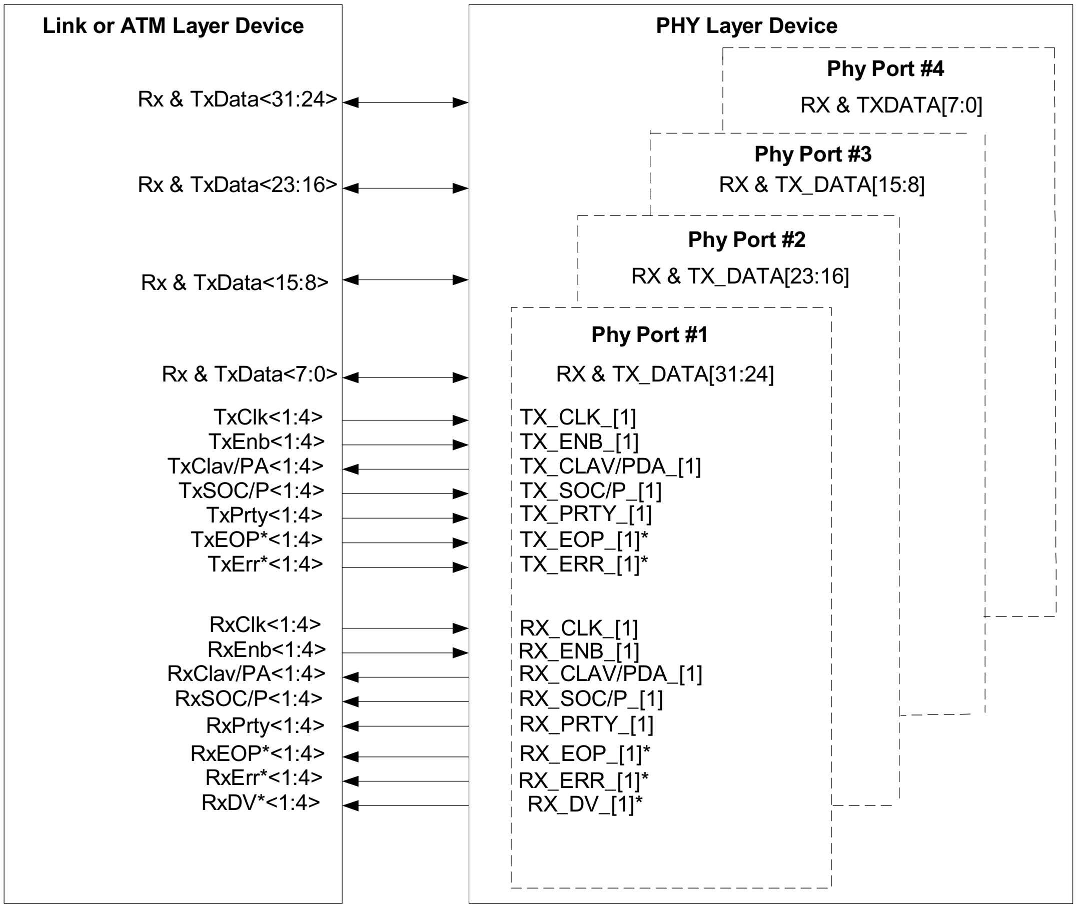


Figure 21: Four by 8-Bit Mode, Single-port Operation

* Designates those signals that are extensions to standard Utopia operation to accommodate packet transfers.

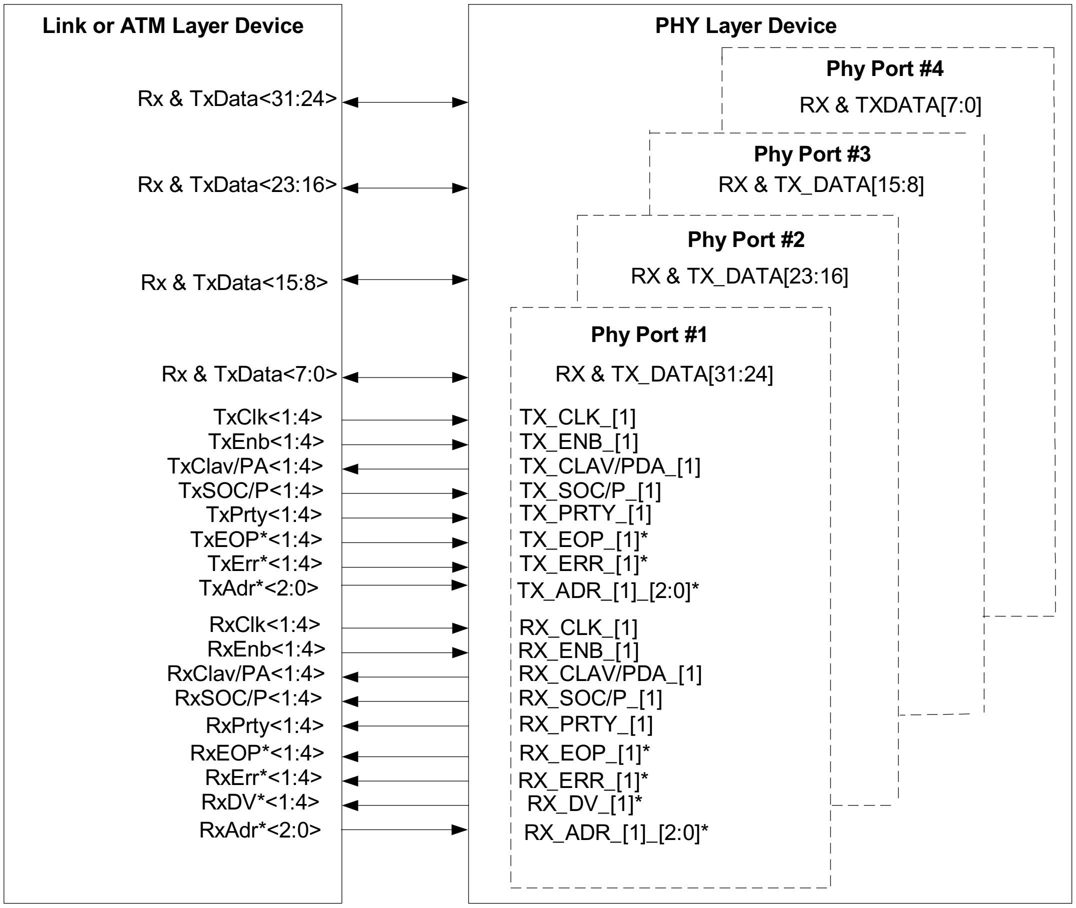


Figure 22: Four by 8-Bit Mode, Multi-port Operation

* Designates those signals that are extensions to standard Utopia operation.

9.2.2 RHINE System Interface Mode Selection

The selection of the desired RHINE, system-interface operating mode is controlled via the **TX_SYSINT_WIDTH** register in the Transmit direction, and the analogous registers in the Receive direction. If **TX** or **RX_SYSINT_WIDTH** = 0, the quad, 8-bit, bus mode is selected; if **TX** or **RX_SYSINT_WIDTH** = 1 (the default), the single, 32-bit bus mode is selected.

When configured for 4 port STS-12c/STM-4 operation (**TX/RX_PP_CONFIG_[19:16]** = 1111), RHINE supports both direct and multiplexed status-polling for a single, 32-bit, bus operation; the selection between these 2 modes is made using registers **TX** and **RX_SYSINT_POLL.** If **TX** or **RX_SYSINT_POLL** = 0 (the default), direct status mode is used; if = 1, multiplexed-polling status is used. All other modes operate strictly in either direct status or multiplexed-polling status mode, only. The state of **TX/RX_SYSINT_POLL** is ignored in these other modes.

9.2.3 RHINE Utopia Operation

9.2.3.1 Basic Signals

The ATM Forum is in the process of specifying the Utopia Level 3 interface for transferring ATM cells between ATM Layer devices and a PHY Layer device/port. RHINE system interface supports Utopia Level 3 as currently defined in [3] and [4]. Separate interfaces are defined for the transmit and receive directions. Utopia specifies that transmit and receive data transfers can occur at clock rates independent of the line bit rate. As a result, the PHY device must support cell rate decoupling using FIFOs. Control signals are provided to and from the ATM Layer device to support flow control in either direction.

In both directions, the ATM Layer device controls the flow of data by either sending data to RHINE, or pulling data from RHINE. RHINE has the ability to indicate to the ATM Layer when it has room for cell data (in the transmit direction) or when it has a cell to transfer (in the receive direction). As specified in Utopia Level 3, in each direction the following signals exist for multi-port bus operation:

SYS_DATA, CLK and *ADR:* Traditional Data/Clock/Address bus signals. A 32-bit data bus, its associated clock, and a 5 bit address bus.

PRTY: Single parity bit over entire 32-bit data bus.

SOC: Start of Cell signal. Provides ATM cell delineation on the bus. Since ATM has fixed length cells, only a start-of-cell indication is necessary.

ENB: Provides flow control for ATM Device. When deasserted, the PHY device either ignores the data on the bus (transmit direction) or does not transfer data (receive direction).

CLAV_y: Provides flow control for each port of the PHY device. When deasserted, the PHY port either ignores the incoming data (transmit direction) or does not have data to send (receive direction).

In addition to the ‘standard’ Utopia Level 3 interface, the FlexBus-3^{TM} interface provides four, 8-bit buses. This extension is described in [4]. Support of 4-parallel, single-port buses requires the following signals:

SYS_DATA, CLK_y and *ADR_y:* Traditional Data/Clock/Address bus signals. Four 8-bit data buses, associated clocks and 3-bit address buses.

PRTY_y: Single parity bit over the associated 8-bit data bus.

SOC_y: Start of Cell signal. Provides ATM cell delineation on each bus. Since ATM has fixed length cells, only a start-of-cell indication is necessary.

ENB_y: Provides flow control for ATM Device. When deasserted, the PHY device either ignores the data on the bus (transmit direction) or does not transfer data (receive direction).

CLAV_y: Provides flow control for each port of the PHY device. When deasserted, the PHY port either ignores the incoming data (transmit direction) or does not have data to send (receive direction).

EOP_y: Although the *EOP* signal only applies to packet transfers, it is also asserted during the last word transfer of an ATM cell if **RX_SYS_ATM_EOP** = 1 for easier implementation of mixed ATM/POS systems. Otherwise, when **RX_SYS_ATM_EOP** = 0 (the default), RX_EOP is always held low. In ATM transfers in the transmit direction, *EOP* is ignored. In 32-bit mode, 52- or 56-byte ATM cell transfers are always expected (depending on the provisioning of **TX(RX)_ATM_UDF**), and *TX_LBYTE[1:0]* are ignored in the TX direction, and *RX_LBYTE_[1:0]* are always output as 00 in the RX direction.

9.2.4 RHINE Packet Mode

9.2.4.1 Basic Signals

RHINE system interface is specified here for transferring packets between 1 or more Link Layer devices and 1 or more ports on a PHY layer device. This interface is based on the Utopia Level 3 interface, with extensions provided to support variable length packets. As in Utopia, separate interfaces are defined for the transmit and receive directions. Transmit and receive data transfers can occur at clock rates independent of the line bit rate. As a result, the PHY device must support data rate decoupling using FIFOs. Control signals, very similar to those defined for Utopia, are provided to and from the Link Layer device to support flow control in either direction.

In both directions, the Link Layer device controls the flow of data by either sending data to RHINE, or pulling data from RHINE. RHINE has the ability to indicate to the Link Layer when it has room for packet data (in the transmit direction) or when it has packet data to transfer (in the receive direction). In each direction, the following signals exists for multi-port bus operation:

SYS_DATA, CLK, and *ADR:* Traditional Data/Clock/Address bus signals. A 32-bit data bus, its associated clock, and a 5-bit address bus.

PRTY: Single parity bit over entire 32-bit data bus.

SOC/P: Start of Packet signal. Indicates the first bytes of a packet are being transferred over the system interface.

EOP: End of Packet signal. Packet extension. Indicates when the last word of a packet is being transferred over the system interface. Necessary to support the variable length of packets.

LBYTE: Size of Last Word in Packet. Packet extension. Indicates the position of the last byte of the packet within the last word of the packet. Decoding values given in Table 40.

ENB: Provides flow control for Link Layer device. When deasserted, the PHY device either ignores the data on the bus (transmit direction) or does not transfer data (receive direction).

CLAV_PDA_y: Packet Data Available. Provides flow control for the PHY device. When deasserted, the PHY port either ignores the incoming data (transmit direction) or does not have data to send (receive direction).

DV: Receive Data Valid. Packet extension. *DV* provides a continuous report of the status of the port from which data is currently being read. Due to the variability of the packet length, the user does not know when to expect a packet transfer to complete - this signal provides the user with an indication of when the data being read from the receive FIFO is valid.

In addition, the FlexBus-3^{TM} interface provides four, 8-bit buses. Support of 4 single-port buses requires the following signals:

SYS_DATA, CLK_y and ADR_y: Traditional Data/Clock/Address bus signals. Four 8-bit data buses and associated clocks and 3-bit address buses.

PRTY_y: Single parity bit over associated 8-bit data bus.

SOC/P_y: Start of Packet signal. Indicates the first bytes of a packet are being transferred over the system interface.

EOP_y: End of Packet signal. Packet extension. Indicates when the last word of a packet is being transferred over the system interface. Necessary to support the variable length nature of packets.

ENB_y: Provides flow control for Link Layer device. When deasserted, the PHY device either ignores the data on the bus (transmit direction) or does not transfer data (receive direction).

CLAV_PDA_y: Packet Data Available. Provides flow control for the PHY device. When deasserted, the

PHY device either cannot accept data (transmit direction) or does not have data to send (receive direction).

DV_y: Receive Data Valid. Packet extension. *DV* provides a continuous report of the status of the port from which data is currently being read. Due to the variability of the packet length, the user does not know when to expect a packet transfer to complete - this signal provides the user with an indication of when the data being read from the receive FIFO is valid.

9.3 Common System Interface Specifications

9.3.1 Bus Widths

RHINE supports 4 x 8 and 32-bit data bus widths.

In packet or Direct Map modes, signals are defined to allow word transfers of less than a full size word, but only during the last word of a packet transfer.

Data bus operation is selected by setting the **TX/RX_SYSINT_WIDTH** register. (See section 9.2.2.)

9.3.2 Clock Rates

The system interface supports nominal clock rates from 25 MHz to 100 MHz. In the direct mapping mode, the Link Layer is responsible for insuring the data rate provides enough data to fill the SPE. If this cannot be guaranteed, RHINE can be programmed to insert a particular byte sequence to fill the SPE during periods of FIFO underflow (**TX_POS_FIFOUNDER_BYTE_[7:0]**. (See section 5.2.7.3.)

Table 31. System Interface Clock Rates

Line Signal	Bit Rate (Mbit/s)	System Bus Width	Nominal System Bus Rate
STS-48/STM-16	2488.32	4 x 8 bits	100 MHz
STS-48/STM-16	2488.32	32 bits	100 MHz
STS-12/STM-4	622.08	4 x 8 bits	100 MHz
STS-12/STM-4	622.08	32 bits	100 MHz
STS-3c/STM-1	155.52	32 bits	100 MHz

In STS-48c POS mode, the system interface must run at a frequency of at least 78Mhz or TX FIFO underflows can result.

9.4 Utopia Level 3 System Interface Requirements

9.4.1 Interface Data Structures

In ATM mode, 52-, 53- or 56-octet cells are transferred between the ATM Layer device and RHINE. User-defined fields are optionally provided for backward compatibility or to fill out an integral number of words. The byte arrangement is Big-Endian. Table 32 through Table 37 show the cell formats for all bus widths.

Table 32. 8-Bit Data Structure Configuration via TX/RX_ATM_UDF and TX/RX_ATM_UDF53

TX/RX_ATM_UDF	TX/RX_ATM_UDF53	8-Bit Data Structure Mode	32-Bit Data Structure Mode
0	X	52-Byte	52-Byte
1	0	56-Byte	56-Byte
1	1	53-Byte	56-Byte

Table 33. 52-Byte Data Structure for 8-Bit ATM Mode

Bits 7-0	8-Bit Mode
Header 1	Header bytes
:	|
Header 4	Time
Payload 1	|
:	|
Payload 48	V

Table 34. 53-Byte Data Structure for 8-Bit ATM Mode

Bits 7-0	8-Bit Mode
Header 1	Header bytes
:	
Header 4	
UDF	User Defined Field
Payload 1	Time
:	|
Payload 48	V

Table 35. 56-Byte Data Structure for 8-Bit ATM Mode

Bits 7-0	8-Bit Mode
Header 1	Header bytes
:	
Header 4	
UDF1	User Defined Field
UDF2	User Defined Field

Table 35. 56-Byte Data Structure for 8-Bit ATM Mode

Bits 7-0	8-Bit Mode
UDF3	User Defined Field
UDF4	User Defined Field
Payload 1	Time
:	|
Payload 48	V

Table 36. 52-Byte Data Structure for 32-Bit ATM Mode

Bits 31-24	Bits 23-16	Bits 15-8	Bits 7-0	32-Bit Mode
Header 1	Header 2	Header 3	Header 4	Header bytes
Payload 1	Payload 2	Payload 3	Payload 4	
Payload 5	Payload 6	Payload 7	Payload 8	|
:	:	:	:	Time
:	:	:	:	|
Payload 45	Payload 46	Payload 47	Payload 48	V

Table 37. 56-Byte Data Structure for 32-Bit ATM Mode, TX/RX_ATM_UDF = 1

Bits 31-24	Bits 23-16	Bits 15-8	Bits 7-0	32-Bit Mode
Header 1	Header 2	Header 3	Header 4	Header bytes
UDF1	UDF2	UDF3	UDF4	User Defined Fields (HEC in UDF1)
Payload 1	Payload 2	Payload 3	Payload 4	
Payload 5	Payload 6	Payload 7	Payload 8	|
:	:	:	:	Time
:	:	:	:	|
Payload 45	Payload 46	Payload 47	Payload 48	V

Register **TX_ATM_HEC_UDF_[1:0]** defines the location of the ATM HEC within the 4 User Defined bytes in the UDF 'capable' 32-bit Utopia ATM transmit data structures. In a single, 32-bit bus operation, if **TX_ATM_HEC_UDF_[1:0]** = 0, the HEC is located in UDF1; if = 01, the HEC is located in UDF2, and so on. In all modes, the HEC is placed in UDF1 in the receive direction. Contents of UDF2 - UDF4 are undefined in RX direction.

9.4.2 ATM Utopia Transmit Operation

RHINE transmit system interface has data flowing from the ATM or link layer device to Rhine for insertion into an output SONET/SDH signal. Signals *TX_SYS_DAT*, *TX_SOC/P_y* and *TX_PRTY_y* are sampled on the rising edge of *TX_CLK_y*.

9.4.2.1 Utopia Transmit Signals

The Utopia Transmit interface signals are described in the Table 2.

9.4.2.2 Direct Status Operation

Each RHINE device provides 4 signals that indicate the status its TX FIFOs to the ATM Layer. These signals are called *TX_CLAV_PDA_[1:4].*

In the transmit direction, *TX_CLAV_PDA_y* is asserted when there is sufficient empty space available in the corresponding TX FIFO to accommodate a provisionable number of ATM cells.

CLAV_PDA timing depends on the setting of **TX_CLAV_MODE** and **TX_CLAV/PDA_DSST_y**. When **TX_CLAV_MODE** = 0 (the default), Rhine CLAV_PDA timing is controlled by **TX_CLAV/PDA_DSST_[1:4]**, for backwards compatibility with earlier versions of RHINE. In this mode, *TX_CLAV_PDA_y* is deasserted four clock cycles before (if **TX_CLAV/PDA_DSST_y** = 0, the default) or 1 clock-cycle after (**TX_CLAV/PDA_DSST_y** = 1) the end of the cell transfer, if the TX FIFO is filled to a provisionable number of ATM cells (provisioned in **TX_FIFOFULL_TPA_y_[3:0]**). For ATM operation, the **TX_FIFOFULL_TPA_y_[3:0]** and **TX_FIFOEMPTY_TPA_y_[3:0]** registers have the interpretation in Table 38.

When **TX_CLAV_MODE** = 1, Rhine transmit CLAV_PDA timing is compliant with the current version of the ATM Forum's Utopia-3 specification (af_phy_0136.000). In this transmit direct status mode, *TX_CLAV_PDA_y* will deassert within 1 *TX_CLK_y* clock cycle after the start of cell (*TX_SOC/P_y*) is detected on the rising edge of *TX_CLK_y* while *TX_ENB_y* is asserted. Once *TX_CLAV_PDA_y* is asserted, it will remain asserted until the next cell transfer is initiated by receipt of *TX_SOC/P_y*.

Table 38. TX/RX_FIFOFULL_T/RPA_y_[3:0] and TX/RX_FIFOEMPTY_T/RPA_y_[3:0] Values in ATM mode

Port Capacity	TX/RX_FIFOFULL_T/RPA_y_[3:0] or TX/RX_FIFOEMPTY_T/RPA_y_[3:0]	Number of cells in FIFO (< for FIFOEMPTY, >= for FIFOFULL
STS-12c/AU-4-4c or STS-48c/AU-4-16c	0000	1
	0001	2
	: :	: :
	1110	15
	1111	undefined
STS-3c/AU-4	XX00	1
	XX01	2
	XX10	3
	XX11	undefined

The reset values of **TX_FIFOFULL_TPA_y_[3:0]** and **TX_FIFOEMPTY_TPA_y_[3:0]** are 0000.

9.4.2.3 Multiplexed-Polling Operation

For certain configurations, RHINE supports multiplexed polling as defined by Utopia Level 3. During multiplexed-polling operation, the *TX_CLAV_PDA_[1]* signal reports the FIFO status for all active ports.

Each PHY port is assigned a PHY address. During multiplexed-polling operation, the *TX_CLAV_PDA_[1]* signal is held low whenever an address not matching any of its port addresses appears on *TX_ADR_y.* Whenever an assigned PHY port address appears on *TX_ADR_y*, *TX_CLAV_PDA_[1]* reports the current status of this port's FIFO, with a two clock cycle delay.

TX_CLAV_PDA_[1] is asserted when there is sufficient space available in the corresponding TX FIFO to accommodate a provisionable number of ATM cells. *TX_CLAV_PDA_[1]* is deasserted if the TX FIFO cannot accommodate a group of ATM cells. The size of this group of cells is provisionable via **TX_FIFOFULL_TPA_y_[3:0]** and **TX_FIFOEMPTY_TPA_y_[3:0]**. The assignment of **TX/RX_FIFOFULL_T/RPA_y** and **TX/RX_FIFOEMPTY_T/RPA_y** registers to RHINE ports is defined in the following table:

Table 39. FIFOFULL/FIFOEMPTY Register to Tributary Assignment

Tributaries	Value of y
STS-48c/AU-4-16c	[1]
STS-12c/AU-4-4c Tributary [1]	[1]
STS-12c/AU-4-4c Tributary [5]	[2]
STS-12c/AU-4-4c Tributary [9]	[3]
STS-12c/AU-4-4c Tributary [13]	[4]
STS-3c/AU-4 Tributaries [1] - [4]	[1]
STS-3c/AU-4 Tributaries [5] - [8]	[2]
STS-3c/AU-4 Tributaries [9] - [12]	[3]
STS-3c/AU-4 Tributaries [13] - [16]	[4]

When **TX_CLAV_MODE** = 1, Rhine transmit CLAV_PDA timing is compliant with the current version of the ATM Forum's Utopia-3 specification (af_phy_0136.000). In this transmit multiplexed status mode, once *TX_CLAV_PDA_y* is asserted for a particular port, responses to subsequent polls of that port will continue to assert *TX_CLAV_PDA_y* for that port until after the second clock cycle of a transfer to that port.

9.4.2.4 Data Transfer - Single-Port Bus Operation

Transmit data is transferred from 1 ATM layer device to a RHINE port via the following procedure. RHINE port p indicates when it can accept an ATM cell into its TX FIFO, using its *TX_CLAV_PDA_y* signal. The ATM Layer device for port p then asserts *TX_ENB_y* to initiate the data transfer. RHINE controls the flow of data via the *TX_CLAV_PDA_y* signal, as described in section 9.4.2.2. The ATM Layer device controls the flow of data via the *TX_ENB_y* signal.

When *TX_ENB_y* is asserted, RHINE port stores data from *TX_SYS_DAT* on the low-to-high transition of *TX_CLK_y.* When *TX_ENB_y* is deasserted, data on *TX_SYS_DAT* is ignored.

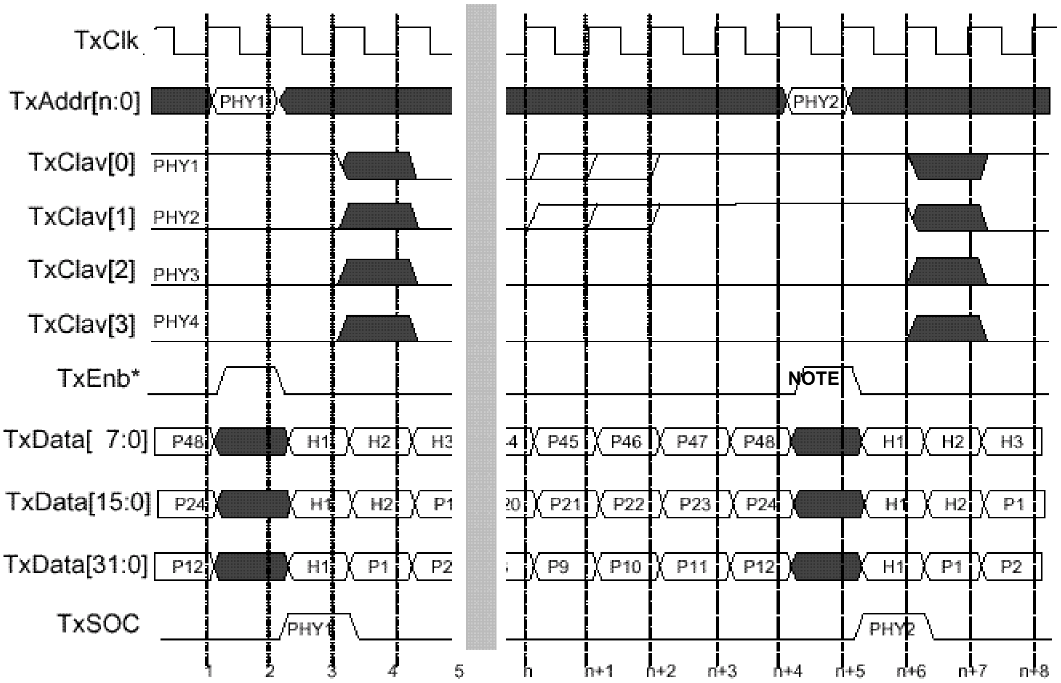
9.4.2.5 Data Transfer - Multi-Port Bus Operation

Transmit data is transferred from the ATM Layer device to a RHINE device provisioned for multiple PHY ports via the following procedure. RHINE indicates when it can accept data into 1 or more of its TX ports, using the appropriate *TX_CLAV_PDA_y* signal(s) (as defined in section 9.4.2.2 or section 9.4.2.3). The ATM Layer then selects the transmit port targeted for data transfer by placing this port's address on *TX_ADR_y* when *TX_ENB_y* is deasserted, and then immediately (the next clock cycle) asserting *TX_ENB_y* to initiate the data transfer. RHINE controls the flow of data via the *TX_CLAV_PDA_y* signals, as described in section 9.4.2.2 or section 9.4.2.3. The ATM Layer controls the flow of data via the

TX_ENB_y signals.

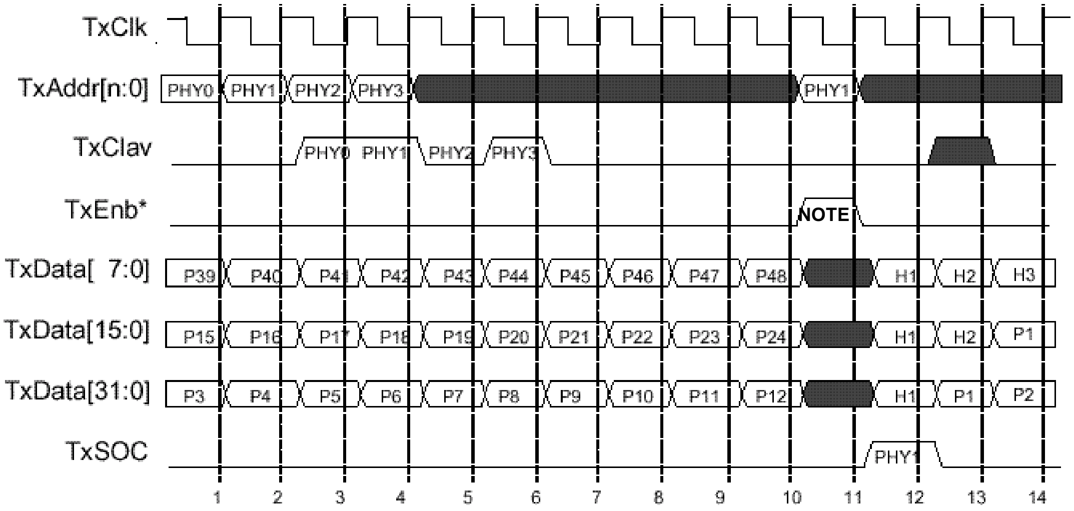
When *TX_ENB_y* is asserted, RHINE stores data from *TX_SYS_DAT* on the low-to-high transition of *TX_CLK_y*. When *TX_ENB_y* is deasserted, data on *TX_SYS_DAT* is ignored.

The following figures illustrate transmit timing for Direct Status multi-PHY operation, Multiplexed Status multi-PHY with a Single CLAV, and Multiplexed Status multi-PHY with Multiple CLAVs.



Note: If TxEnb not deasserted and TxClav[0] still asserted, implicit re-selection of PHY1 occurs on clock edge n+4 for back-to-back transfer.

Figure 23. Direct Status, Multi-PHY Transfer from Link Layer (TX_CLAV_MODE = 1)



Note: If TxEnb not deasserted and TxClav still asserted, implicit re-selection of PHY1 occurs on clock edge 10 for back-to-back transfer.

Figure 24. Multiplexed Status, Multi-PHY Transfer from Link Layer with Single CLAV (TX_CLAV_MODE = 1)

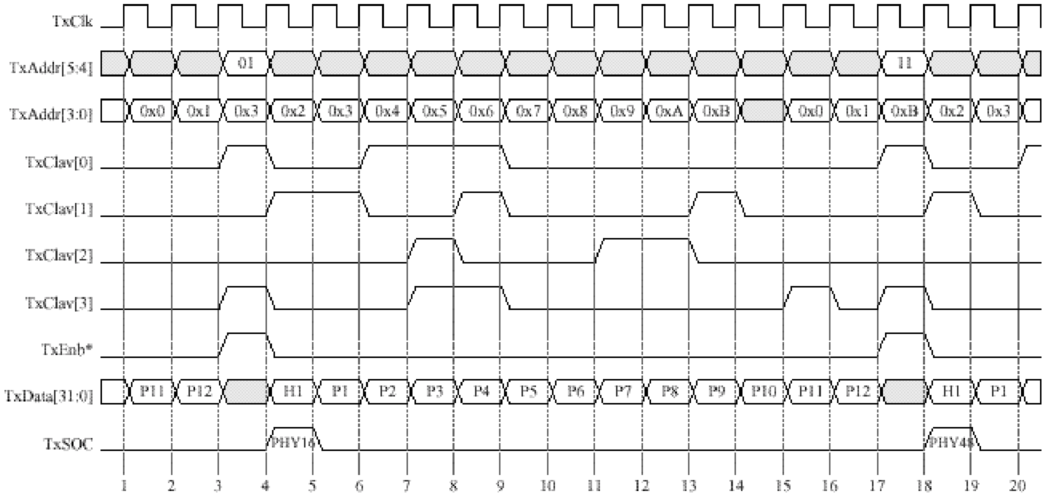


Figure 25. Multiplexed Status, Multi-PHY Transfer from Link Layer with Multiple CLAVs (TX_CLAV_MODE = 1)

9.4.2.6 Error Handling

RHINE does not store data from *TX_SYS_DAT* if a new cell begins on the data bus, but the destination port cannot accept another full cell's worth of data. This will not occur if the ATM Layer responds to the *TX_CLAV_PDA_y* signals correctly. RHINE also does not store data when *TX_ENB_y* is deasserted. RHINE does not store data from the ATM Layer if it is expecting the data to begin a new cell, but

TX_SOC/P_y has not been asserted to indicate that the first octets of a cell are being transferred. If *TX_SOC/P_y* is asserted in the 'middle' of a cell transfer (before an entire cell has been transferred), the interrupted cell is dropped. Only complete cells are transmitted over the SONET line.

9.4.3 ATM Utopia Receive Operation

RHINE receive system interface has data flowing in the opposite direction to the ATM Layer enable. RHINE generates all output signals on the rising edge of the *RX_CLK_y*.

9.4.3.1 Utopia Receive Signals

The Utopia receive signals are defined in Table 2.

9.4.3.2 Direct Status Operation

Each RHINE device provides 4 signals that indicate the status of its RX FIFOs to the ATM Layer. These signals are called *RX_CLAV_PDA_y. RX_CLAV_PDA_y* is asserted when the corresponding receive FIFO has stored a provisionable number of ATM cells, and is ready to transfer them to the ATM Layer. This threshold is provisioned via **RX_FIFOFULL_RPA_y_[3:0]**.

In the receive direction, CLAV_PDA timing depends on the setting of **RX_CLAV_MODE** and **RX_CLAV/PDA_DSST_y**. When **RX_CLAV_MODE** = 0 (the default), Rhine CLAV_PDA timing is controlled by **RX_CLAV/PDA_DSST_[1:4]**, for backwards compatibility with earlier versions of RHINE. In this mode, *RX_CLAV_PDA_y* is deasserted four clock cycles before (if **RX_CLAV/PDA_DSST_y** = 0, the default) or 1 clock-cycle after (**RX_CLAV/PDA_DSST_y** = 1) the end of the cell transfer, if the RX FIFO does not contain a specified number of cells. This threshold is provisionable via **RX_FIFOEMPTY_RPA_y_[3:0]**. For Utopia operation, the **RX_FIFOFULL_RPA_y_[3:0]** and **RX_FIFOEMPTY_RPA_y_[3:0]** registers have the interpretation defined in Table 38.

The reset value of **RX_FIFOFULL_RPA_y_[3:0]** and **RX_FIFOEMPTY_RPA_y_[3:0]** is 0000. This is the best setting (for ATM only) and will guarantee no cells can be stuck in the receive FIFO.

When **RX_CLAV_MODE** = 1, Rhine receive CLAV_PDA timing is compliant with the current version of the ATM Forum's Utopia-3 specification (af_phy_0136.000). In this receive direct status mode, *RX_CLAV_PDA_y* will deassert coincident with asserting start of cell (*RX_SOC/P_y*) on the rising edge of *RX_CLK_y* when *RX_ENB_y* is asserted (assuming Rhine currently has no additional received cells to transfer to the link layer device). Once *RX_CLAV_PDA_y* is asserted, it will remain asserted until the next cell transfer is initiated.

9.4.3.3 Multiplexed-Polling Operation

In certain configurations, RHINE supports multiplexed polling as defined by Utopia Level 3.

Each PHY port is assigned a PHY address. The same PHY address is used in both the TX and RX directions. During multiplexed-polling operation, the *RX_CLAV_PDA_[1]* signal is held low whenever an address not matching any of its port addresses appears on *RX_ADR_y.* Whenever an assigned PHY port address appears on *RX_ADR_y*, *RX_CLAV_PDA_[1]* reports the current status of this port's FIFO, *with a two clock cycle delay*.

RX_CLAV_PDA_[1] is asserted when the FIFO for the RX port currently being polled has a preprovisioned number of ATM cells ready to transfer to the ATM Layer device.

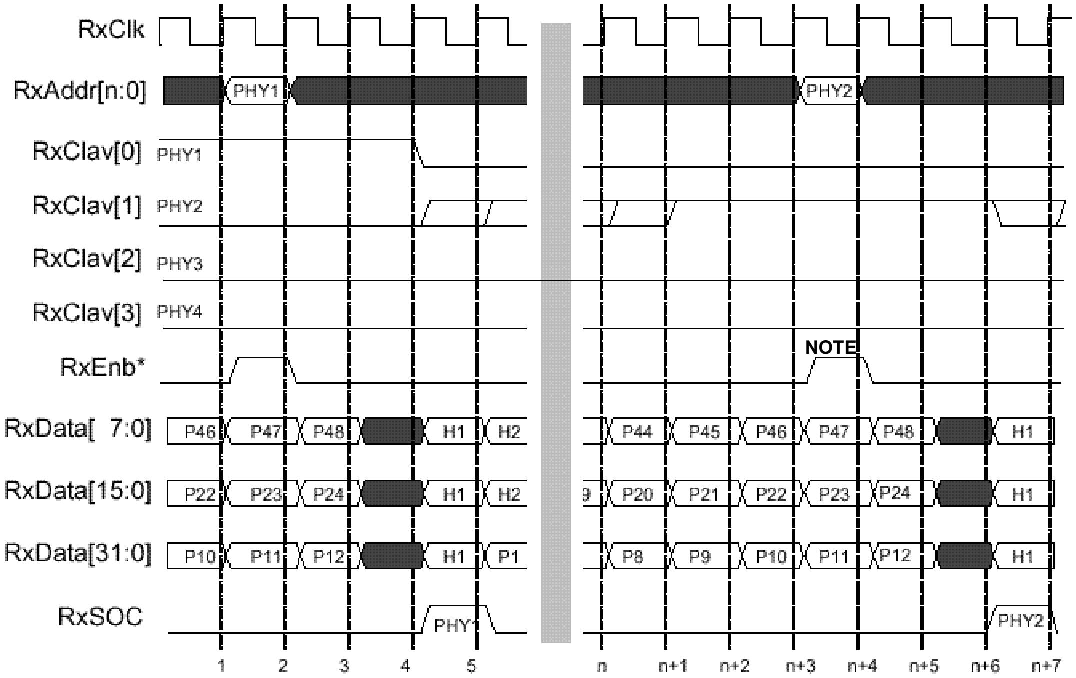
When **RX_CLAV_MODE** = 1, once *RX_CLAV_PDA_y* is asserted for a particular port, responses to subsequent polls of that port will continue to assert *RX_CLAV_PDA_y* for that port until the next cell transfer for that port is initiated (*RX_SOC/P_y* asserted).

9.4.3.4 Data Transfer - Single-Port Bus Operation

Receive data is transferred from a RHINE port to a Link Layer device via the following procedure. A RHINE RX port indicates it has valid data available for transfer from its FIFO, using its *RX_CLAV_PDA_y* signal. The Link Layer device for this port then asserts *RX_ENB_y* to initiate the data transfer, *with a two clock cycle delay*. RHINE port controls the flow of data via the *RX_CLAV_PDA_y* signal, as described in section 9.5.3.2. The Link Layer device controls the flow of data via the *RX_ENB_y* signal. For back-to-back transfers when *RX_CLAV_PDA_y* indicates another cell is available, *RX_ENB_y* should remain asserted; another 0 to 1 transition of *RX_ENB_y* is not required.

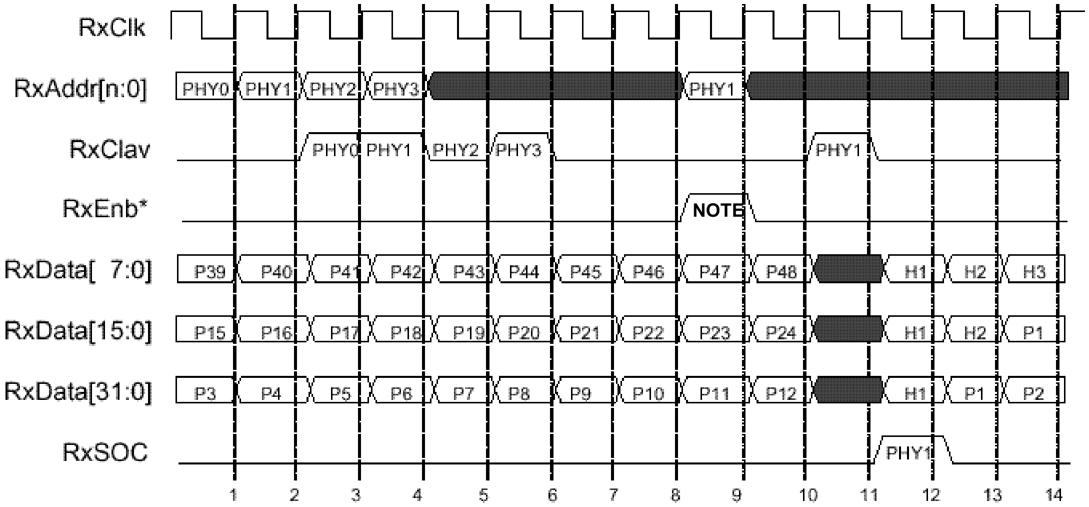
9.4.3.5 Data Transfer - Multi-Port Bus Operation

Receive data is transferred from a RHINE device provisioned for multiple ports to an ATM Layer device via the following procedure. RHINE indicates it has valid data in some/all of its RX ports, using its *RX_CLAV_PDA_y* signals. The ATM Layer then selects the receive port targeted for data transfer by placing this port's address on *RX_ADR_y* when *RX_ENB_y* is deasserted, and then immediately (the next clock cycle) asserting *RX_ENB_y* to initiate the data transfer, *with a two clock cycle delay*. RHINE controls the flow of data via the *RX_CLAV_PDA_y* signals, as described in section 9.4.3.2 or section 9.4.3.3. The ATM Layer controls the flow of data via the *RX_ENB_y* signal(s).



Note: If RxEnb remains asserted, implicit re-selection of PHY1 occurs on clock edge n+3, enabling back-to-back transfer.

Figure 26. Direct Status, Multi-PHY Transfer to Link Layer



Note: If RxEnb remains asserted, implicit re-selection of previously selected PHY occurs on clock edge 8, enabling back-to-back transfer.

Figure 27. Multiplexed Status, Multi-PHY Transfer to Link Layer with Single CLAV

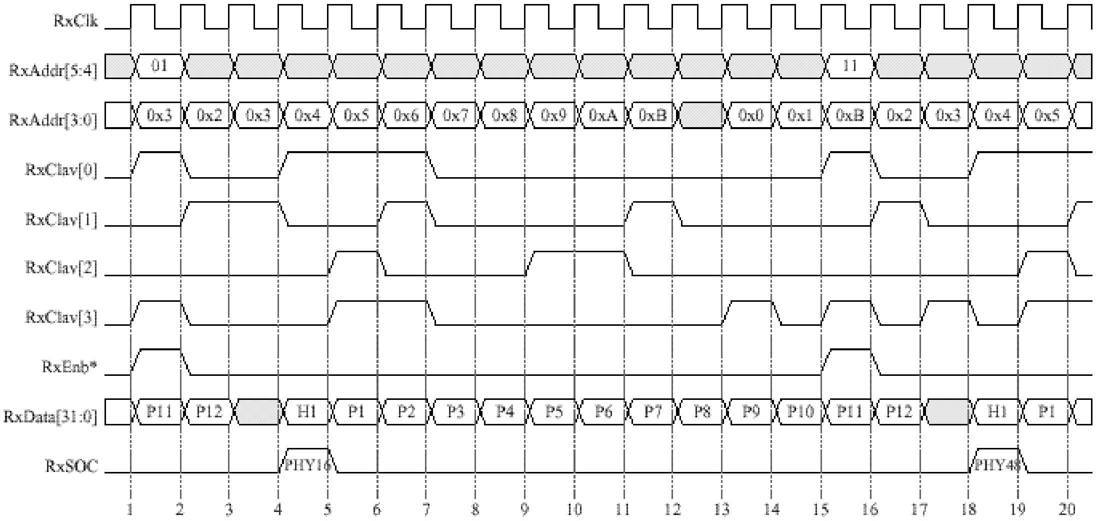


Figure 28. Multiplexed Status, Multi-PHY Transfer to Link Layer with Multiple CLAVs

9.4.3.6 Error Handling

RHINE does not place data on *RX_SYS_DAT* if it does not have another cell’s worth of data. RHINE also does not transfer data when *RX_ENB_y* is deasserted (with a two clock cycle delay).

9.5 RHINE Packet Interface Requirements

9.5.1 POS Packet Format

Packets are written to the transmit FIFOs and read from the receive FIFOs using the following defined data structures.

For 8-bit bus operation, the data is sent across RHINE system interface in byte increments.

For 32-bit operation, all words are composed of 4 bytes, except the last word of a packet transfer which can have 1 to 4 bytes. The signals *TX_LBYTE_[1:0]* and *RX_LBYTE_[1:0]* convey the position of the last data byte within the final data word to the receiving device. The coding for these signals depends on the setting of **TX(RX)_SYS_SIZE_MODE**. When **TX(RX)_SYS_SIZE_MODE** = 1 (the default), *TX(RX)_LBYTE_[1:0]* are interpreted as shown in Table 40. When **TX(RX)_SYS_SIZE_MODE** = 0, *TX(RX)_LBYTE_[1:0]* indicate the number of valid bytes modulo-4.

Table 40. *TX* and *RX_LBYTE_[1:0]* Values

TX(RX)_LBYTE_[1:0]	**Valid Data Bytes on Data Signals for TX(RX)_SYS_SIZE_MODE=1 (default)**	**Valid Data Bytes on Data Signals for TX(RX)_SYS_SIZE_MODE=0**
00	*TX/RX_SYS_DAT_[31:0]*, 4 bytes valid	*TX/RX_SYS_DAT_[31:0]*, 4 bytes valid
01	*TX/RX_SYS_DAT_[31:8]*, 3 bytes valid	*TX/RX_SYS_DAT_[31:24]*, 1 byte valid
10	*TX/RX_SYS_DAT_[31:16]*, 2 bytes valid	*TX/RX_SYS_DAT_[31:16]*, 2 bytes valid
11	*TX/RX_SYS_DAT_[31:24],* 1 byte valid	*TX/RX_SYS_DAT_[31:8]*, 3 bytes valid

These data structures do not preclude the transfer of packets less than 4 bytes. In this case, both the start-of-packet packet and end-of-packet signals are asserted simultaneously. The 32-bit POS packet structures are illustrated in Table .

Table 41. Data Structure for 32-Bit POS Mode

Bits 31-24	**Bits 23-16**	**Bits 15-8**	**Bits 7-0**	**32-Bit Mode**
Data 1	Data 2	Data 3	Data4	
Data 5	Data 6	Data 7	Data 8	|
				Time
				|
Data 21	Data 22	Data 23		V
23 Byte Packet; TX/RX_LBYTE_[1:0] = 01				

9.5.2 RHINE Packet Transmit Operation

RHINE transmit system interface has data flowing from the Link Layer device to Rhine. All system interface input signals are sampled on the rising edge of the appropriate *TX_CLK_y*.

9.5.2.1 RHINE Packet Transmit Signals

RHINE Packet Transmit interface signals are described in Table 2.

9.5.2.2 Direct Status Operation

RHINE provides 4 signals that indicate the status of the transmit FIFOs to the Link Layer. These signals are called *TX_CLAV_PDA_y*.

In the transmit direction, RHINE indicates which of its FIFOs are not full by asserting the corresponding transmit packet available signals, *TX_CLAV_PDA_y*. *TX_CLAV_PDA_y* is asserted when there are less than a specified number of bytes (provisionable through **TX_FIFOEMPTY_TPA_y_[3:0]**) available in the FIFO. *TX_CLAV_PDA_y* remains asserted until the transmit FIFO is almost full. Almost full implies that RHINE FIFO contains at least a predefined number of bytes (provisioned via **TX_FIFOFULL_TPA_y_[3:0]**). The **TX_FIFOFULL_TPA_y_[3:0]** and **TX_FIFOEMPTY_TPA_y_[3:0]** registers are interpreted as shown in Table 42:

Table 42. TX/RX_FIFOFULL_T/RPA_y_[3:0] and TX/RX_FIFOEMPTY_T/RPA_y_[3:0] Values in Direct Map and POS mode

Port Capacity	TX/RX_FIFOFULL_T/RPA_y_[3:0] or TX/RX_FIFOEMPTY_T/RPA_y_[3:0]	Number of bytes in FIFO (< for FIFOEMPTY, >= for FIFOFULL
STS-12c/AU-4-4c or STS-48c/AU-4-16c	0000	0 = Undefined
	0001	64
	: :	: :
	1110	896
	1111	960
STS-3c/AU-4	X000	0 = Undefined
	X001	32
	X010	64
	: :	: :
	X110	192
	X111	224

9.5.2.3 Multiplexed-Polling Operation

For certain packet applications, RHINE supports multiplexed polling identical to that defined by Utopia Level 3 for ATM applications.

Each RHINE port is assigned a PHY address. During multiplexed-polling operation, the *TX_CLAV_PDA_[1]* signal is held low (*with a two clock cycle delay*) whenever an address not matching any of its port addresses appears on *TX_ADR_y.* Whenever an assigned PHY port address appears on *TX_ADR_y*, *TX_CLAV_PDA_[1]* reports the current status of this port's FIFO, *with a two clock cycle delay*.

TX_CLAV_PDA_[1] is asserted when the FIFO for the TX port currently being polled has room to accept a predefined amount of packet data, set via **TX_FIFOEMPTY_TPA_y_[3:0]**. *TX_CLAV_PDA_[1]* is deasserted if the fill of the FIFO for the TX port currently being polled exceeds a predefined level, which is set via **TX_FIFOFULL_TPA_y_[3:0]**. **TX_FIFOEMPTY_TPA_y_[3:0]** and **TX_FIFOFULL_TPA_y_[3:0]** should be set > 0.

9.5.2.4 Data Transfer - Single-Port Bus Operation

Transmit data is transferred to a single port on RHINE from a Link Layer device via the following procedure. A RHINE port indicates it has room available for data in its TX FIFO, using its *TX_CLAV_PDA_y* signal. The Link Layer device for this port then asserts *TX_ENB_y* to initiate the data transfer. RHINE port controls the flow of data via the *TX_CLAV_PDA_y* signal, as described in section 9.5.2.2. The Link Layer device controls the flow of data via the *TX_ENB_y* signal.

9.5.2.5 Data Transfer - Multi-Port Bus Operation

Transmit data is transferred to a RHINE provisioned for multi-port bus operation from a single Link Layer device via the following procedure. RHINE indicates it has room available for data in 1 or more of its FIFOs, using the *TX_CLAV_PDA_y* signals. (See section 9.5.2.2 or section 9.5.2.3.) The Link Layer then selects the transmit port targeted for data transfer by placing this port's address on *TX_ADR_y* when *TX_ENB_y* is deasserted, and then immediately (the next clock cycle) asserting *TX_ENB_y* to initiate the data transfer. RHINE controls the flow of data via the *TX_CLAV_PDA_y* signals, as described in section 9.5.2.2 or section 9.5.2.3. The Link Layer controls the flow of data via the *TX_ENB_y* signals.

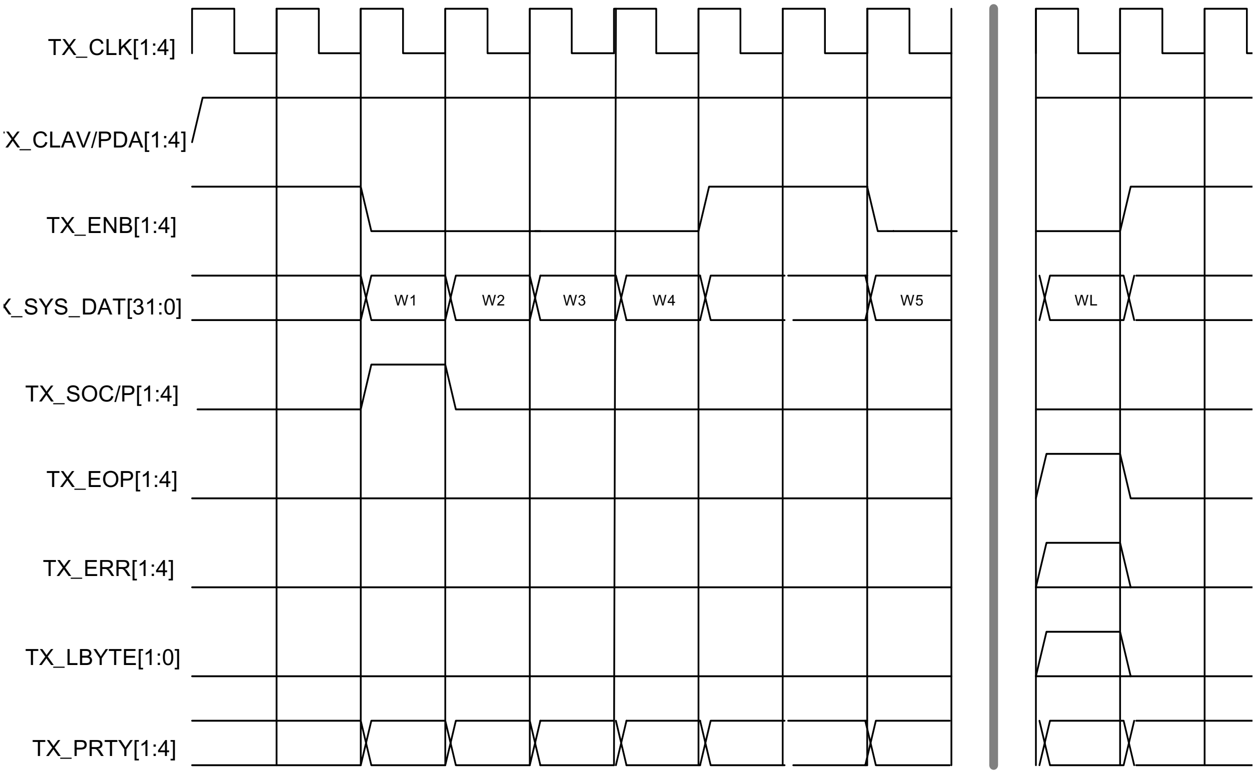


Figure 29. Packet Transmit Transfer stalled by Link Layer Deasserting *TX_ENB*

9.5.2.6 Error Handling

RHINE does not store data when *TX_ENB_y* is deasserted. RHINE does not store data from the Link Layer if it is expecting the beginning of a new packet, but *TX_SOC/P_y* is not asserted to indicate that the first bytes of a packet are being transferred. If *TX_SOC/P_y* is asserted in the 'middle' of a packet transfer, before *TX_EOP_y* is asserted, the interrupted packet is aborted.

9.5.3 RHINE Packet Receive Operation

The receive RHINE Packet system interface has data flowing in the opposite direction to the Link Layer enable. The Link receive block samples its inputs and generates all output signals on the rising edge of the *RX_CLK_y*.

9.5.3.1 RHINE Packet Receive Signals

RHINE packet receive signals are defined in Table 2.

9.5.3.2 Direct Status Operation

Each RHINE device provides up to 4 signals that indicate the status of its receive FIFOs to the Link Layer. These signals are called *RX_CLAV_PDA_y.*

In the receive direction, RHINE indicates that 1 or more of its FIFOs contain an end of packet that is ready to transfer by asserting the corresponding receive packet available signal, *RX_CLAV_PDA_y*. *RX_CLAV_PDA_y* is also asserted when there are greater than or equal to a specified number of bytes (provisionable through **RX_FIFOFULL_RPA_y_[3:0]**) available in the FIFO. *RX_CLAV_PDA_*y remains asserted until the receive FIFO contains no end-of-packet and is almost empty. Almost empty implies that RHINE contains at most a predefined number of reads after the current read (provisioned via **RX_FIFOEMPTY_RPA_y_[3:0]**). The interpretation of the **RX_FIFOFULL_RPA_y_[3:0]** and **RX_FIFOEMPTY_RPA_y_[3:0]** registers are given in Table 42. **RX_FIFOEMPTY_RPA_y_[3:0]** and **RX_FIFOFULL_RPA_y_[3:0]** should be set > 0.

9.5.3.3 Multiplexed-Polling Operation

For certain packet applications, RHINE supports multiplexed polling identical to that defined by Utopia Level 3 for ATM applications.

Each RHINE port is assigned a PHY address. The same address is used for both TX and RX sides of a single port. During multiplexed-polling operation, the *RX_CLAV_PDA_[1]* signal is held low (*with a two clock cycle delay*) whenever an address not matching any of RHINE active port addresses appears on *RX_ADR_y*. Whenever an assigned PHY port address appears on *RX_ADR_y*, *RX_CLAV_PDA_[1]* reports the current status of this port's FIFO, with a two clock cycle delay.

RX_CLAV_PDA_[1] is asserted when the FIFO for the RX port currently being polled has stored and is ready to transfer a predefined amount of packet data (set via **RX_FIFOFULL_RPA_y_[3:0]**), or an end-of-packet. *RX_CLAV_PDA_[1]* is deasserted if the fill of the FIFO for the RX port currently being polled falls below a predefined level (set via **RX_FIFOEMPTY_RPA_y_[3:0]**) and does not contain an end-of-packet.

9.5.3.4 Data Transfer - Single-Port Bus Operation

Receive data is transferred from a RHINE port to a Link Layer device via the following procedure. A RHINE RX port indicates it has valid data available for transfer from its FIFO, using its *RX_CLAV_PDA_y* signal. If **RX_SYS_B2B_ALLOW** = 0 (the default), the Link Layer device for this port must then assert *RX_ENB_y* to initiate the data transfer, *with a two clock cycle delay*. When **RX_SYS_B2B_ALLOW** = 1, RHINE will initiate data transfer, as long as *RX_ENB_y* is asserted; it does not require a 0 to 1 transition to activate data transfer, and thus back-to-back transfers are supported. RHINE port controls the flow of data via the *RX_CLAV_PDA_y* signal, as described in section 9.4.3.2. The Link Layer device controls the flow of data via the *RX_ENB_y* signal.

RX_ENB_y must undergo at least one high-to-low transition after a hard reset in order for Rhine to properly begin its first data transfer. This means *RX_ENB_y* cannot simply be tied low even in single port operation with **RX_SYS_B2B_ALLOW** = 1. One option if the link layer device does not need to use this *RX_ENB_y* pin for flow control would be to invert and delay the reset pulse one FlexBus3 clock cycle and use this signal to drive *RX_ENB_y.*

9.5.3.5 Data Transfer - Multi-Port Bus Operation

Receive data is transferred from a RHINE configured for multi-port operation to a Link Layer device via the following procedure. RHINE indicates it has valid data available for transfer from 1 or more of its ports, using the *RX_CLAV_PDA_y* signals. The Link Layer device then selects the receive port targeted for data transfer by placing this port's address on *RX_ADR_y* when *RX_ENB_y* is deasserted, and then immediately (the next clock cycle) asserting *RX_ENB_y* to initiate a data transfer, *with a 2 clock cycle delay*. RHINE controls the flow of data via the *RX_CLAV_PDA_y* signals, as described in section 9.5.3.2 or section 9.5.3.3. The Link Layer controls the flow of data via the *RX_ENB_[1]* signal, *with a 2 clock cycle delay*.

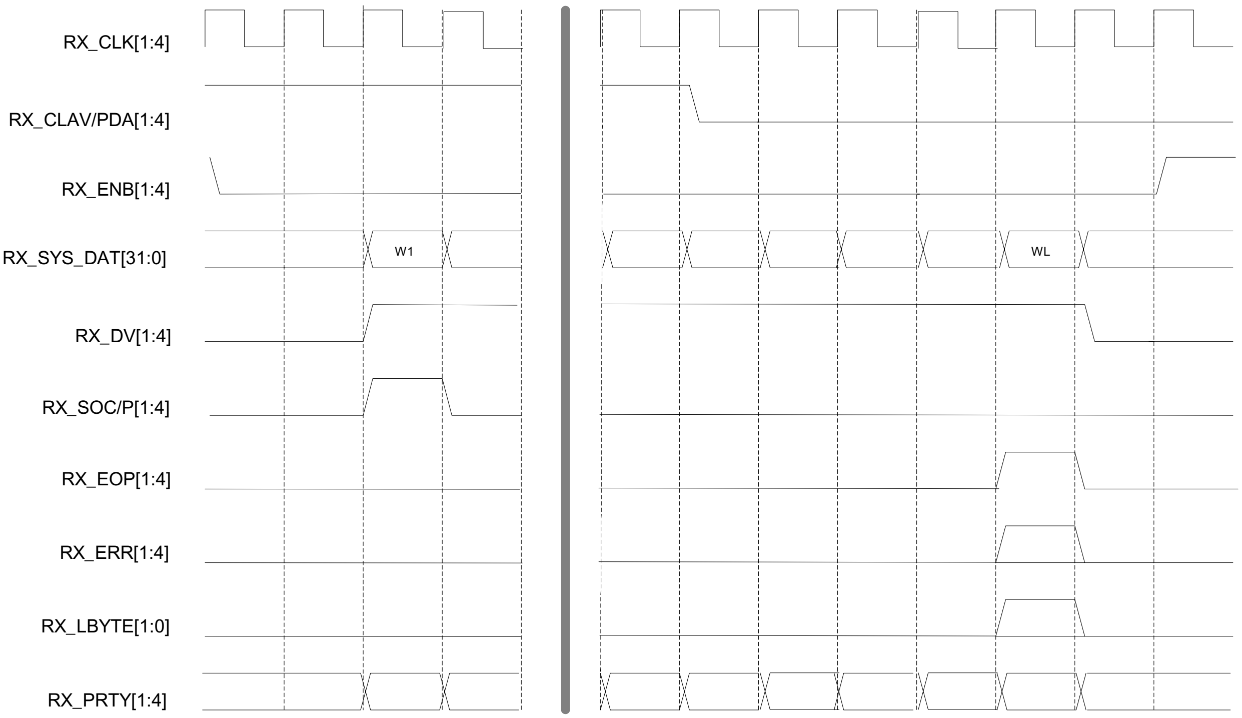


Figure 30. Receive Packet Transfer

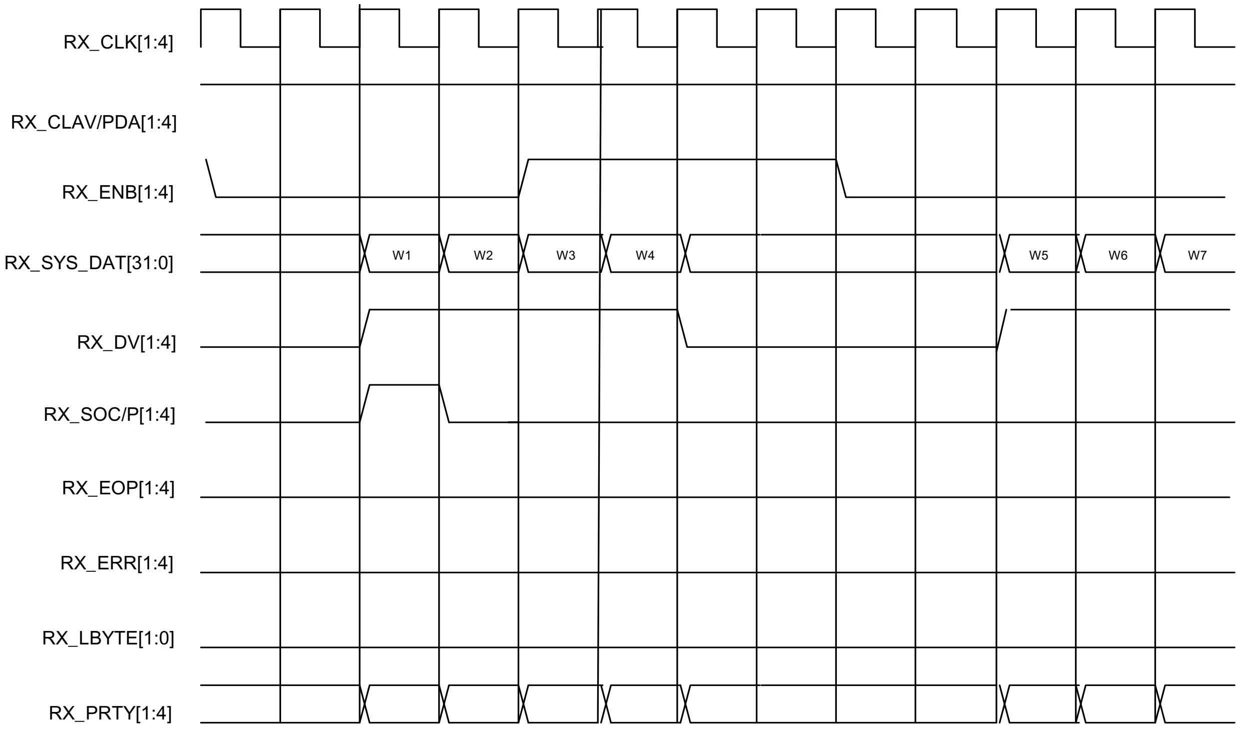


Figure 31. Receive Packet Transfer Stalled by Link Layer Deasserting RX_ENB

9.5.4 Direct Mode Operation

Direct mode operates as a simpler subset of the packet interface. RHINE system interface operates as in the packet mode, except that the *TX_SOC/P_y, TX_EOP_y, TX_LBYTE, TX_ERR_y, RX_SOC/P_y, RX_EOP_y, RX_LBYTE,* and *RX_ERR_y* are not used.

The microprocessor interface is capable of operating in either an interrupt driven or a polled mode. In the interrupt mode, RHINE is capable of supporting multiple Interrupt Sources. RHINE is capable of masking out any of the interrupts in either interrupt mode.

9.6 RHINE FlexBus-3 Operation with Mixed Payloads

Rhine supports mixed payload operation. That is, a mix of ATM, POS, and/or Direct Map payloads can be simultaneously provisioned in Rhine in both transmit and receive directions. With mixed payloads, Rhine can still operate in either 32-bit mode or quad 8-bit mode. **RX_SYS_ATM_EOP** is provided so that ATM cells can be transferred over the FlexBus with the same operation as packets (e.g. EOP will be asserted coincident with the last 8- or 32-bit transfer of a cell). Please take note of the following when operating with a mix of payloads:

- If you wish to transfer packets and cells with common interface operation, set **RX_SYS_ATM_EOP** = 1.
- If you wish ATM cell transfers to be Utopia-3 compliant, while packet transfers are handled with FlexBus-3 packet extensions, set **RX_SYS_ATM_EOP** = 0.
- If possible, it is recommended to segregate ATM and POS or DIrect Map operation by quadrant

(e.g. on a per STS-12c or 4xSTS-3c basis).

- For POS packet transfers in quadrant y, **TX/RX_FIFOFULL_TPA_y_[3:0]** and **TX/RX_FIFOEMPTY_RPA_y_[3:0]** must be > 0.
- For segregated ATM cell transfers in quadrant y, **RX_FIFOFULL_RPA_y_[3:0]** are recommended to be set to 0000 to insure a cell does not get stuck in the RX FIFO indicated by *RX_CLAV_y.*
- If ATM cell payloads are segregated in separate STS-12c / 4xSTS-3c quadrants from POS or Direct Map payloads, it is recommended to set **TX_FIFOFULL_TPA_y_[3:0]** and **RX_FIFOFULL_RPA_y_[3:0]** > 0 so that larger burst transfers can be accommodated.

10.0 Management Interface

This section describes the management interface to RHINE and defines the address of all registers that are available for reading or writing by an external microprocessor. Table through Table are the management interface register maps for the Transmit side, and Table through Table are the management interface register maps for the Receive side.

The MSB of the microprocessor bus address, *ADDR_[13]*, designates whether the map is associated with the Transmit (*ADDR_[13]* = 0) or Receive (*ADDR_[13]* = 1) direction. *ADDR_[12:0]* indicate the specific register and these values are identified with the following detailed descriptions of each map. For 8-bit microprocessor operation, all 14 address bits must be used. For 16-bit operation, the *ADDR_[0]* signal is not needed; the two 8-bit registers that correspond to *ADDR_[13:1]* are read from/written to the same cycle; bit 7 of the register at address location *ADDR_[0]* = 1 is the MSB of this 16-bit data transfer; bit 0 of the register at address *ADDR_[0]* = 0 is the LSB.

To facilitate accessing specific tributaries within these maps, the following addressing convention is utilized whenever possible: within a particular map, Tributary [1] registers begin in address 0xXY00, Tributary [2] registers begin in address 0xXY10, and so on. When referencing register addresses in the following text, the convention “0xXXXX **-** 0xYYYY” or “0xXXXX through 0xYYYY” enumerates all consecutive registers within the specified range. The phrase “[0xXXXX, 0xYYYX”] stipulates incrementing by byte increments through the specified range, for example [0x0040, 0x0060] refers to addresses 0x040, 0x050, and 0x060.

10.1 Interrupt or Polled Operation

The management interface can be operated in either an interrupt driven or a polled mode. In both modes, RHINE register bits **TX_SUM_INT** in address 0x0002 and **RX_SUM_INT** in address 0x2002 can be used to determine whether or not changes have occurred in the state of monitoring registers in RHINE.

10.1.1 Interrupt Sources

10.1.1.1 Transmit Side

The Transmit side register maps are almost entirely provisioning parameters that determine the composition of the SONET/SDH signal and provide the HDLC POS, ATM, SONET/SDH POH, and SONET/SDH TOH/SOH values.

In addition to these provisioning parameters, the Transmit side register map includes system interface and General Purpose I/O monitors. If any of these indications are active, the **TX_SUM_INT** bit in register 0x0002 will be high (logic 1). If **TX_SUM_INT_MASK** = 0, the interrupt output for the microprocessor interface, *INTB,* becomes active (logic 0).

10.1.1.2 Receive Side

Table 54 contains summary status bits for the Receive side in registers 0x2004-200B. These bits contribute to the **RX_SUM_INT** bit in register 0x2002. If any of the summary status bits is “1” and the corresponding mask bit is “0,” then the **RX_SUM_INT** bit will be set to “1.”

The summary status bits in registers 0x2003-B of Table 54 are “1” if 1 or more of the corresponding group of bits for that particular status is “1.” Individual TOH/SOH delta and second event bits can be masked (Table 55, addresses [0x2204:0x2206] to [0x25C4:0x25C6].

10.1.2 Interrupt Driven

In an interrupt driven mode, the **TX_SUM_INT_MASK** bit in register 0x0006 and the **RX_SUM_INT_MASK** bit in register 0x200C should be cleared (to logic 0). This allows the *INTB* output to become active (logic 0). This output is

INTB = !((**TX_SUM_INT** && !**TX_SUM_INT_MASK**) OR (**RX_SUM_INT** && !**RX_SUM_INT_MASK**))

In addition, the **RX_APS_INT_MASK** bits of the Receive side should be cleared (to logic 0). This allows the *APS_INTB* output to become active (logic 0). This output is

APS_INTB = !(!**RX_APS_INT_MASK** && **RX_APS_INT**)

If an interrupt occurs, the microprocessor can first read the summary status registers to determine the class(es) of interrupt source(s) that is active, and then read the specific registers in that class(es) to determine the exact cause of the interrupt.

10.1.3 Polled mode

The **TX_SUM_INT_MASK, RX_SUM_INT_MASK** and **RX_APS_INT_MASK** bits should be set (to logic 1) to suppress all hardware interrupts and operate in a polled mode. In this mode, RHINE outputs *INTB* and *APS_INTB* are held in the inactive (logic 1) state.

Note that the **TX_SUM_INT_MASK, RX_SUM_INT_MASK** and **RX_APS_INT_MASK** bits do not affect the state of the register bits **TX_SUM_INT, RX_SUM_INT** and **RX_APS_INT**. These bits can be polled to determine if further register interrogation is needed.

11.0 Register Map Descriptions

The following sections describe the register bits accessible through the management interface. The addresses of these bits are shown in Table through Table . Register addresses that contain at least 1 but fewer than 8 bits (shown as blank bit fields in the tables) return 0 in the undefined bit fields when read. RHINE *D_[7:0]* or *D_[15:0]* response to read requests of register addresses that are not shown in the tables is not defined.

11.1 Transmit Side Registers

The following sections describe the individual pages that constitute the transmit side register maps.

11.1.1 Transmit Configuration and Summary Status - Addresses 0x0000 through 0x01FF (Table)

11.1.1.1 Software Resets - Addr 0x0000

When the **TX_PROV_RESET** bit is written to a 1, all transmit provisioning (read/write) registers (except the register containing the **TX_PROV_RESET** bit itself) are reset. These transmit registers are held in their default state until **TX_PROV_RESET** is written to 0 or a hardware reset occurs. These registers include all transmit provisioning and transmit mask registers, but do not include delta, event, or second event bits.

When the **TX_STATE_RESET** bit is written to a 1, all transmit status, monitoring, delta, event, second event and counter registers (the read-only registers), and all transmit state machines are reset to their default states (except again the register containing the **TX_PROV** and **TX_STATE_RESET** bits itself). This group should consist of all transmit registers NOT reset by **TX_PROV_RESET**. The Transmit side of the device remains inactive until the **TX_STATE_RESET** bit is written to 0 (or a hardware reset occurs).

The hardware reset input, *RSTB*, produces the same result as writing a 1 to **TX_PROV_RESET** and **TX_STATE_RESET**, except that the hardware reset forces **TX_PROV_RESET** and **TX_STATE_RESET** to 0, and the device begins operation when *RSTB* goes high (logic 1).

When the registers in the following tables are reset, they are forced low (logic 0), unless stated otherwise.

11.1.1.2 Summary Interrupt and Mask - Addr 0x0002 and 0x0006

The **TX_SUM_INT** bit is high (logic 1) if any bit in registers 0x0004-0x0005 is high and the corresponding mask is low. On reset, the **TX_SUM_INT** bit is low, because the bits in registers 0x0004-0x0005 are all forced low and all mask bits are forced high on reset.

The logic equation for **TX_SUM_INT** is:

TX_SUM_INT = (**TX_ATM_HEC_ERR_SUME** && !**TX_ATM_HEC_ERR_SUME_MASK**)**||**

(**TX_FIFOERR_SECE_SUM** && !**TX_FIFOERR_SECE_SUM_MASK**) **||**

(**TX_PRTY_ERR_SUME** && !**TX_PRTY_ERR_SUME_MASK**) ||

(**TX_POS_LLPKT_ERR_SECE_SUM** &&
!**TX_POS_LLPKT_ERR_SECE_SUM_MASK**) ||

(**TX_POS_PMAX_ERR_SECE_SUM** &&
!**TX_POS_PMAX_ERR_SECE_SUM_MASK**) ||

(**TX_POS_PMIN_ERR_SECE_SUM** &&
!**TX_POS_PMIN_ERR_SECE_SUM_MASK**) ||

(**TX_POS_FIFOUNDER_ERR_SECE_SUM** &&

!**TX_POS_FIFOUNDER_ERR_SECE_SUM_MASK**) ||
(**GPIO_SUMD** && !**GPIO_SUMD_MASK**)

The **TX_SUM_INT_MASK** bit is used to disable RHINE Transmit side contribution to the interrupt output, *INTB*. (See section 10.1.1.)

11.1.1.3 Device Version Number - Addr 0x0001

The device version number, **DEV_VER_[7:0]** = 0x00.

11.1.1.4 Transmit Summary Deltas, Summary Events, Summary Second Events and Masks - Addr 0x0004-0x0005 and 0x0008-0x0009

The summary delta and summary event bits contribute to the **TX_SUM_INT** bit. On reset, all mask bits are set high, and all summary delta and summary event bits are set low.

TX_PRTY_ERR_SUME is a function of the POS parity error event bits in register 0x0B01 and the transmit parity error masks in register 0x0B03. The logic equation for this bit is:

TX_PRTY_ERR_SUME = (**TX_PRTY_ERR_[1]_E** && !**TX_PRTY_ERR_[1]_E_MASK**) ||
(**TX_PRTY_ERR_[2]_E** && !**TX_PRTY_ERR_[2]_E_MASK**) ||
(**TX_PRTY_ERR_[3]_E** && !**TX_PRTY_ERR_[3]_E_MASK**) ||
(**TX_PRTY_ERR_[4]_E** && !**TX_PRTY_ERR_[4]_E_MASK**)

TX_ATM_HEC_ERR_SUME is a function of the ATM HEC error event bits in registers [0x0A00,0x0AF0] and the transmit ATM tributary masks in registers [0x0100,0x01F0]. The logic equation for this bit is:

TX_ATM_HEC_ERR_SUME = (**TX_ATM_HEC_ERR_[1]_E** && !**TX_ATM_TRIB_[1]_MASK**) ||
(**TX_ATM_HEC_ERR_[2]_E** && !**TX_ATM_TRIB_[2]_MASK**) ||
(**TX_ATM_HEC_ERR_[3]_E** && !**TX_ATM_TRIB_[3]_MASK**) ||...||
(**TX_ATM_HEC_ERR_[16]_E** && !**TX_ATM_TRIB_[16]_MASK**)

TX_FIFOERR_SECE_SUM is a function of the FIFO error second event bits in registers [0x0C00,0x0CF00] and the transmit tributary masks in registers [0x0100,0x01F0]. The logic equation for this bit is:

TX_FIFOERR_SECE_SUM = (**TX_FIFOERR_[1]_SECE** && !**TX_TRIB_[1]_MASK**) ||
(**TX_FIFOERR_[2]_SECE** && !**TX_TRIB_[2]_MASK**) ||
(**TX_FIFOERR_[3]_SECE** && !**TX_TRIB_[3]_MASK**) ||...||
(**TX_FIFOERR_[16]_SECE** && !**TX_TRIB_[16]_MASK**)

TX_POS_PMAX_ERR_SECE_SUM is a function of the POS maximum packet size violation second event bits in registers [0x0E00,0x0EF0] and the transmit POS tributary masks in registers [0x0100,0x01F0]. The logic equation for this bit is:

TX_POS_PMAX_ERR_SECE_SUM = (**TX_POS_PMAX_ERR_[1]_SECE** &&
!**TX_POS_TRIB_[1]_MASK**) ||
(**TX_POS_PMAX_ERR_[2]_SECE** && !**TX_POS_TRIB_[2]_MASK**) ||
(**TX_POS_PMAX_ERR_[3]_SECE** && !**TX_POS_TRIB_[3]_MASK**) ||...||
(**TX_POS_PMAX_ERR_[16]_SECE** && !**TX_POS_TRIB_[16]_MASK**)

TX_POS_PMIN_ERR_SECE_SUM is a function of the POS Minimum packet size violation second event bits in registers [0x0E00,0x0EF0] and the transmit POS tributary masks in registers [0x0100,0x01F0]. The logic equation for this bit is:

TX_POS_PMIN_ERR_SECE_SUM = (**TX_POS_PMIN_ERR_[1]_SECE** &&
!**TX_POS_TRIB_[1]_MASK**) ||

(**TX_POS_PMIN_ERR_[2]_SECE** && !**TX_POS_TRIB_[2]_MASK**) ||

(**TX_POS_PMIN_ERR_[3]_SECE** && !**TX_POS_TRIB_[3]_MASK**) ||...||

(**TX_POS_PMIN_ERR_[16]_SECE** && !**TX_POS_TRIB_[16]_MASK**)

TX_POS_LLPKT_ERR_SECE_SUM is a function of the POS packet error second event counters in [0x0E00,0x0EF0] and the transmit POS tributary masks in registers [0x0100,0x01F0]. The logic equation for this bit is:

TX_POS_LLPKT_ERR_SECE_SUM = (**TX_POS_LLPKT_ERR_[1]_SECE** && !**TX_POS_TRIB_[1]_MASK**)

|| (**TX_POS_LLPKT_ERR_[2]_SECE** && !**TX_POS_TRIB_[2]_MASK**) ||

(**TX_POS_LLPKT_ERR_[3]_SECE** && !**TX_POS_TRIB_[3]_MASK**) ||...||

(**TX_POS_LLPKT_ERR_[16]_SECE** && !**TX_POS_TRIB_[16]_MASK**)

TX_POS_FIFOUNDER_ERR_SECE_SUM is a function of the POS FIFO underflow second event counters in [0x0E00,0x0EF0] and the transmit tributary masks in registers [0x0100,0x01F0]. The logic equation for this bit is:

TX_POS_FIFOUNDER_ERR_SECE_SUM =

(**TX_POS_FIFOUNDER_ERR_[1]_SECE** && !**TX_POS_TRIB_[1]_MASK**) ||

(**TX_POS_FIFOUNDER_ERR_[2]_SECE** && !**TX_POS_TRIB_[2]_MASK**) ||

(**TX_POS_FIFOUNDER_ERR_[3]_SECE** && !**TX_POS_TRIB_[3]_MASK**) ||...||

(**TX_POS_FIFOUNDER_ERR_[16]_SECE** && !**TX_POS_TRIB_[16]_MASK**)

GPIO_SUMD is a function of the General Purpose I/O delta bits in register 0x000E and their masks in register 0x0010. The logic equation for this bit is:

GPIO_SUMD = (**GPIO0_D** && !**GPIO0_MASK**) ||

(GPIO1_D && **!GPIO1_MASK**) || ...||

(**GPIO7_D** && !**GPIO7_MASK**)

11.1.1.5 Signal Composition Provisioning - Addr 0x000A-0x000D

The **TX_CONFIG_[20:0]** and **TX_LINE_CONFIG_[4:0]** registers determine the structure of the generated signal(s). (See section 4.3.)

The **TX_APS_SEL_x** and **TX_APS_DXC_SEL_x** registers control the APS selectors.

11.1.1.6 General Purpose I/O Deltas, Masks, Status, Control, Parity Error Masks - Addr 0x000B through 0x0010

GPIO0 through **GPIO7** contain General Purpose Input/Output registers that are tied directly to RHINE pins *GPIO0* through *GPIO7*. These are grouped in blocks of 2 in the register maps, each group of 2 is controlled by a **GPIOCTLx** register. **GPIOCTL1** = 1 designates that **GPIO0** and **GPIO1** are to be General Purpose Inputs, else they are outputs. **GPIOCTL2** = 1 designates that **GPIO2** and **GPIO3** are to be General Purpose Inputs, else they are outputs. **GPIOCTL3** = 1 designates that **GPIO4** and **GPIO5** are to be General Purpose Inputs, else they are outputs. **GPIOCTL4** = 1 designates that **GPIO6** and **GPIO7** are to be General Purpose Inputs, else they are outputs. All **GPIOCTLx** registers default to 1 (*GPIOn* default to inputs). If **GPIOx** is provisioned as an input, the **GPIOx_D** delta bit is written to 1 whenever there is a change in **GPIOx**.

There are a number of restrictions on GPIO operation, as summarized below:

- No restrictions when all 8 GPIO are either all inputs or all outputs
- No restrictions when lower 4 bits are outputs and upper 4 bits are inputs
- If the lower 4 bits are inputs and the upper 4 bits are outputs, a read returns the previous content of registers; an additional write is required before reading the current register content
- If bits 7/6 and 3/2 are outputs and bits 5/4 and 3/2 are inputs, a read returns the previous content of registers; an additional write is required before reading the current register content
- GPIO registers should not be programmed for bits 7/6 and 3/2 as inputs and 5/4 and 1/0 as outputs.
- GPIO0 is reserved for frame sync output pulse when programmed as an output and **SYS_SYNC_OUT_EN** = 1, as described in section 5.9.1 and section 6.7.4.2.

11.1.1.7 Tributary Summary Delta/Event Bits - Addr 0x0016-7

The tributary summary delta/event bits do not directly contribute to the **TX_SUM_INT** bit. These bits reflect the contributors to the functional summary deltas and events on a per-tributary basis. These are provided to allow the user to quickly determine the tributary that is the source of an interrupt, minimizing register read cycles. On reset, all mask bits are set high, which forces all tributary summary delta/event bits low.

The logic equation for **TX_TRIB_x_SUMD** is:

TX_TRIB_x_SUMD = (**TX_ATM_HEC_ERR_x_E** && !**TX_ATM_TRIB_x_MASK**)||
(**TX_FIFOERR_x_SECE** && !**TX_TRIB_x_MASK**) ||
(**TX_POS_PMAX_ERR_x_SECE** && !**TX_POS_TRIB_x_MASK**) ||
(**TX_POS_PMIN_ERR_x_SECE** && !**TX_POS_TRIB_x_MASK**) ||
(**TX_POS_LLPKT_ERR_x_SECE** & **TX_POS_TRIB_x_MASK**) ||
(**TX_POS_FIFOUNDER_ERR_x_SECE** && !**TX_POS_TRIB_x_MASK**)

11.1.1.8 Tributary Configuration Provisioning - Addr [0x0100,0x01F0]

If **TX_UNEQ_x** is active (logic 1), tributary **x** is generated with all 0s in its SPE/VC bytes. (See section 5.5.3.)

TX_TRIB_x_MASK should be written to 1 if the user desires to disable transmit monitoring of tributary x. **TX_ATM_TRIB_x_MASK** should be written to 1 if the user desires to disable ATM transmit monitoring of tributary x; **TX_POS_TRIB_x_MASK** should be written to 1 if the user desires to disable POS transmit monitoring for tributary x.

SYS_R_TO_T_LOOP_x controls the loopback of tributary **x** from the receive to transmit direction. (See section 5.1.4.)

The operation of the payload scrambler circuitry for tributary x is controlled by **TX_SCR_INH_x**. (See section 5.4.)

The **TX_DIRECT_MAP_x** and **TX_POS_x** registers determine the structures of the generated signal(s).

Table 43. Transmit Side Configuration and Summary Status Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Reset Control (Read/Write)									
0000	0x00							TX_STATE_RESET	TX_PROV_RESET
Device Version Number (Read Only)									
0001	0x00	DEV_VER_[7:0]							
Transmit Side Summary Interrupt and Summary Delta/Event (Read Only)									
0002	0x00								TX_SUM_INT
0003		Unused							
0004	0x20			Reserved	TX_PRTY_ERR_SUME	TX_ATM_HEC_ERR_SUME	GPIO_SUMD	TX_FIFOERR_SECE_SUM	Reserved
0005	0x00				TX_POS_PMAX_ERR_SECE_SUM	TX_POS_PMIN_ERR_SECE_SUM	TX_POS_LLPKT_ERR_SECE_SUM	TX_POS_FIFOUNDER_ERR_SECE_SUM	Reserved
Transmit Side Summary Interrupt and GPIO Summary Masks (Read/Write)									
0006	0x01								TX_SUM_INT_MASK
0007		Unused							
0008	0x3F			Reserved	TX_PRTY_ERR_SUME_MASK	TX_ATM_HEC_ERR_SUME_MASK	GPIO_SUMD_MASK	TX_FIFOERR_SECE_SUM_MASK	Reserved
0009	0x1F				TX_POS_PMAX_ERR_SECE_SUM_MASK	TX_POS_PMIN_ERR_SECE_SUM_MASK	TX_POS_LLPKT_ERR_SECE_SUM_MASK	TX_POS_FIFOUNDER_ERR_SECE_SUM_MASK	Reserved
Configuration, PRBS Provisioning, GPIO Provisioning (Read/Write)									
000A	0x00	Reserved; Must be left set to 00 for proper operation							
000B	0x0F				SYS_SYNC_OUT_EN	GPIOCTL1	GPIOCTL2	GPIOCTL3	GPIOCTL4
000C	0x00	TX_CONFIG_[7:0]							
000D	0x00	TX_CONFIG_[15:8]							
000E	0x00	APS_IF_MODE	Reserved	Reserved	TX_CONFIG_[20:16]				
000F	0x40	APS_OUT_INH	Reserved	APS_IN_INH	TX_LINE_CONFIG_[4:0]				

Table 43. Transmit Side Configuration and Summary Status Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0010	0x00	TX_APS_SEL_[1:8]							
0011	0x00	TX_APS_SEL_[9:16]							
0012	0x00	TX_APS_DXC_SEL_[1:8]							
0013	0x00	TX_APS_DXC_SEL_[9:16]							
0014	0x00	APS_OUT_SEL_[1:8]							
0015	0x00	APS_OUT_SEL_[9:16]							
0016-001F	0x00	Unused							
GPIO Delta Bits (Read / cleared by uP)									
0020	0x00	GPIO7_D	GPIO6_D	GPIO5_D	GPIO4_D	GPIO3_D	GPIO2_D	GPIO1_D	GPIO0_D
0021		Unused							
Mask Bits (Read/Write)									
0022	0xFF	GPIO7_MASK	GPIO6_MASK	GPIO5_MASK	GPIO4_MASK	GPIO3_MASK	GPIO2_MASK	GPIO1_MASK	GPIO0_MASK
0023									
General Purpose Inputs/Outputs (Read/Write)									
0024	0x00	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0
0025		Unused							
Tributary Summary Delta/Events (Read Only)									
0026	0x00	TX_TRIB_[1:8]_SUMD							
0027	0x00	TX_TRIB_[9:16]_SUMD							
0028-00FF		Unused							
Tributary Configuration Provisioning (Read/Write)									
0100	0x68	Reserved	TX_POS_TRIB_[1]_MASK	TX_ATM_TRIB_[1]_MASK	SYS_R_TO_T_LOOP_[1]	TX_TRIB_[1]_MASK	TX_UNEQ_[1]	TX_SCR_INH_[1]	
0101	0x00	Reserved; must be set to 0000				Reserved	TX_POS_[1]	TX_DIRECT_MAP_[1]	TX_TRIB_INH_[1]
0102-010F		Unused							
0110	0x68	Reserved	TX_POS_TRIB_[2]_MASK	TX_ATM_TRIB_[2]_MASK	SYS_R_TO_T_LOOP_[2]	TX_TRIB__[2]_MASK	TX_UNEQ_[2]	TX_SCR_INH_[2]	
0111	0x10	Reserved; must be set to 0001				Reserved	TX_POS_[2]	TX_DIRECT_MAP_[2]	TX_TRIB_INH_[2]
0112-011F		Unused							

Table 43. Transmit Side Configuration and Summary Status Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0120	0x68	Reserved	TX_POS_TRIB_MASK_[3]	TX_ATM_TRIB_MASK_[3]	SYS_R_TO_T_LOOP_[3]	TX_TRIB_MASK_[3]	TX_UNEQ_[3]	TX_SCR_INH_[3]	
0121	0x20	Reserved; must be set to 0010				Reserved	TX_POS_[3]	TX_DIRECT_MAP_[3]	TX_TRIB_INH_[3]
0122-012F		Unused							
0130-01EF		: :							
01F0	0x68	Reserved	TX_POS_TRIB_MASK_[16]	TX_ATM_TRIB_MASK_[16]	SYS_R_TO_T_LOOP_[16]	TX_TRIB_MASK_[16]	TX_UNEQ_[16]	TX_SCR_INH_[16]	
01F1	0xF0	Reserved; must be set to 1111				Reserved	TX_POS_[16]	TX_DIRECT_MAP_[16]	TX_TRIB_INH_[16]
01F2-01FF		Unused							

11.1.2 Transmit Side Addresses 0x0200 through 0x03FF (Table)

A1A2_ERR_x, **A1A2_ERR_PAT_x_[15:0]**, **A1A2_ERR_NUM_x_[2:0]**, **B1_INV_x**, **B2_INV_x**, and **TX_M1_ERR_x** control the generation of errors in the A1, A2, B1, B2, and M1 bytes in SONET/SDH signal x.(See section 5.9.3.1, section 5.9.3.3, section 5.9.3.5, and section 5.9.3.8.)

Automatic generation of LRDI and LREI are inhibited if **LRDI_INH_x** and **LREI_INH_x** are high. Scrambling of the SONET/SDH signal is disabled if **SCRINH_x** is high. (See section 5.9.3.6 and section 5.9.4.)

If **TX_LAIS_x** is high, the entire SONET/SDH signal y is generated as all 1s, except for the first 3 rows of Section Overhead. (See section 5.9.3.6.)

TX_SDH_PG_x is used to select between SDH (=1) and SONET (=0) modes for pointer generation. (See Section 5.5.5.)

TX_K1K2_APS_x and **TX_K2_3LSB_x** registers are used to select the source of the K1 and K2 bytes; either the register map (if =0) or the TOH serial channel (if = 1). (See section 5.9.3.4.)

The **TX_J0_x_[15:0]_[7:0]**, **TX_K1_x_[7:0]**, **TX_K2_x_[7:0]**, and **TX_S1_x_[7:0]** registers are used in the generation of J0, K1, K2, and S1 bytes. (See section 5.9.3.2, section 5.9.3.6, and section 5.9.3.7.)

The SONET/SDH TOH/SOH Provisioning and Scrambler Inhibit register map for SONET/SDH signal [1] is illustrated in Table . Identical maps exist for Signals [2] through [16]; these are located starting at addresses 0x0220, 0x0240, 0x0260 through 0x03E0.

Table 44. Transmit Side Register Address Map for SONET/SDH TOH/SOH Provisioning and Scrambler Inhibit

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Signal 1 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0200	0x00	A1A2_ERR_[1]	A1A2_ERR_NUM_[1]_[2:0]			B1_INV_[1]	B2_INV_[1]	TX_M1_ERR_[1]	SCRINH_[1]

Table 44. Transmit Side Register Address Map for SONET/SDH TOH/SOH Provisioning and Scrambler Inhibit

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0201	0x00	TX_LAIS_[1]	TX_SDH_PG_[1]	TX_K1K2_APS_[1]	TX_K2_3LSB_[1]			LRDI_INH_[1]	LREI_INH_[1]
0202	0x00	A1A2_ERR_PAT_[1]_[7:0]							
0203	0x00	A1A2_ERR_PAT_[1]_[15:8]							
0204	0x01	TX_J0_[1]_[0]_[7:0]							
0205	0x01	TX_J0_[1]_[1]_[7:0]							
0206	0x01	TX_J0_[1]_[2]_[7:0]							
0207	0x01	TX_J0_[1]_[3]_[7:0]							
0208	0x01	TX_J0_[1]_[4]_[7:0]							
0209	0x01	TX_J0_[1]_[5]_[7:0]							
020A	0x01	TX_J0_[1]_[6]_[7:0]							
020B	0x01	TX_J0_[1]_[7]_[7:0]							
020C	0x01	TX_J0_[1]_[8]_[7:0]							
020D	0x01	TX_J0_[1]_[9]_[7:0]							
020E	0x01	TX_J0_[1]_[10]_[7:0]							
020F	0x01	TX_J0_[1]_[11]_[7:0]							
0210	0x01	TX_J0_[1]_[12]_[7:0]							
0211	0x01	TX_J0_[1]_[13]_[7:0]							
0212	0x01	TX_J0_[1]_[14]_[7:0]							
0213	0x01	TX_J0_[1]_[15]_[7:0]							
0214	0x00	TX_K2_[1]_[7:0]							
0215	0x00	TX_K1_[1]_[7:0]							
0216	0x00	TX_S1_[1]_[7:0]							
0217	0x00	Reserved		Reserved			Reserved		
0218-021F		Unused							
Signal 2 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0220-023F		Same as 0x0200 through 0x21F							
Signal 3 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0240-025F		Same as 0x0200 through 0x21F							
Signal 4 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0260-027F		Same as 0x0200 through 0x21F							
Signal 5 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									

Table 44. Transmit Side Register Address Map for SONET/SDH TOH/SOH Provisioning and Scrambler Inhibit

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0280-029F		Same as 0x0200 through 0x21F							
Signal 6 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
02A0-02BF		Same as 0x0200 through 0x21F							
Signal 7 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
02C0-02DF		Same as 0x0200 through 0x21F							
Signal 8 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
02E0-02FF		Same as 0x0200 through 0x21F							
Signal 9 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0300-031F		Same as 0x0200 through 0x21F							
Signal 10 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0320-033F		Same as 0x0200 through 0x21F							
Signal 11 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0340-035F		Same as 0x0200 through 0x21F							
Signal 12 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0360-037F		Same as 0x0200 through 0x21F							
Signal 13 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
0380-039F		Same as 0x0200 through 0x21F							
Signal 14 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
03A0-03BF		Same as 0x0200 through 0x21F							
Signal 15 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
03C0-03DF		Same as 0x0200 through 0x21F							
Signal 16 SONET/SDH Line/MS AIS Generation Control, Scrambler Inhibit, and TOH/SOH Provisioning (Read/Write)									
03E0-03FF		Same as 0x0200 through 0x21F							

11.1.3 Transmit Side Addresses 0x0400 through 0x05FF

The **PRDI_ENH_x**, **PRDI_AUTO_x**, **FORCE_G1ERR_x**, and **TX_G1_x_[2:0]** registers are used in the generation of G1 bytes. (See section 5.5.2.4.)

The **TX_J1SEL_x** and **TX_J1_COMMON_[47:0]_[7:0]** are used for generation of J1 bytes. (See section

5.5.2.1.)

If **TX_PAIS_x** is high, Path/AU AIS is generated in path x. (See section 5.5.5.1.)

B3 bytes for are generated with errors if **B3_INV_x** is high. (See section 5.5.2.2.) The C2 bytes are transmitted with the value provisioned in **TX_C2_x_[7:0]**, where the MSB of C2 is bit 7. (See section 5.5.2.3.)Transmit Side Register Address Map for Path/AU AIS Generation and Path Overhead Provisioning

Table 45. Path Overhead Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Path Overhead Provisioning (Read/Write)									
0400	0x00	Reserved							
0401-040F		Unused							
0410	0x00	TX_J1_COMMON_[47]_[7:0]							
0411	0x00	TX_J1_COMMON_[46]_[7:0]							
0412-043E	0x00	: :							
043F	0x00	TX_J1_COMMON_[0]_[7:0]							
0440-04FF		Unused							
SONET/SDH Path/AU AIS Generation Control and Path Overhead Provisioning - Tributary [1] (Read/Write)									
0500	0x00	FORCE_G1ERR_[1]	TX_G1_[1]_[2:0]					B3_INV_[1]	TX_PAIS_[1]
0501	0x01	TX_C2_[1]_[7:0]							
0502	0x00		TX_J1SEL_[1]	TX_ATM_LCD_PRDI_[1]	TX_UNEQ_PRDI_[1]	TX_PLM_PRDI_[1]	PREI_INH_[1]	PRDI_ENH_[1]	PRDI_AUTO_[1]
0503-050F		Unused							
SONET/SDH Path/AU AIS Generation Control and Path Overhead Provisioning - Tributary [2] (Read/Write)									
0510	0x00	FORCE_G1ERR_[2]	TX_G1_[2]_[2:0]					B3_INV_[2]	TX_PAIS_[2]
0511	0x01	TX_C2_[2]_[7:0]							
0512	0x00		TX_J1SEL_[2]	TX_ATM_LCD_PRDI_[2]	TX_UNEQ_PRDI_[2]	TX_PLM_PRDI_[2]	PREI_INH_[2]	PRDI_ENH_[2]	PRDI_AUTO_[2]
0513-051F		Unused							
SONET/SDH Path/AU AIS Generation Control and Path Overhead Provisioning - Tributary [3] (Read/Write)									
0520	0x00	FORCE_G1ERR_[3]	TX_G1_[3]_[2:0]					B3_INV_[3]	TX_PAIS_[3]
0521	0x01	TX_C2_[3]_[7:0]							
0522	0x00		TX_J1SEL_[3]	TX_ATM_LCD_PRDI_[3]	TX_UNEQ_PRDI_[3]	TX_PLM_PRDI_[3]	PREI_INH_[3]	PRDI_ENH_[3]	PRDI_AUTO_[3]

Table 45. Path Overhead Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0523-052F		Unused							
0530-05EF		: :							
SONET/SDH Path/AU AIS Generation Control and Path Overhead Provisioning - Tributary [16] (Read/Write)									
05F0	0x00	FORCE_G1ERR_[16]	TX_G1_[16]_[2:0]					B3_INV_[16]	TX_PAIS_[16]
05F1	0x01	TX_C2_[16]_[7:0]							
05F2	0x00		TX_J1SEL_[16]	TX_ATM_LCD_PRDI_[16]	TX_UNEQ_PRDI_[16]	TX_PLM_PRDI_[16]	PREI_INH_[16]	PRDI_ENH_[16]	PRDI_AUTO_[16]
05F3-05FF		Unused							

11.1.4 Transmit Side Addresses 0x0600 through 0x08FF (Table 46)

The **TX_J1_x_[15:0]_[7:0]** are used for generation of J1 bytes. (See section 5.5.2.1.)

Table 46. Transmit J1 Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
J1 Path Trace Messages (Read/Write)									
0600	0x00	TX_J1_[1]_[0]_[7:0]							
0601	0x00	TX_J1_[1]_[1]_[7:0]							
0602 to 060D	0x00	: : :							
060E	0x00	TX_J1_[1]_[14]_[7:0]							
060F	0x00	TX_J1_[1]_[15]_[7:0]							
0610	0x00	TX_J1_[2]_[0]_[7:0]							
0611	0x00	TX_J1_[2]_[1]_[7:0]							
0612 to 061D	0x00	: : :							
061E	0x00	TX_J1_[2]_[14]_[7:0]							
061F	0x00	TX_J1_[2]_[15]_[7:0]							
0620	0x00	TX_J1_[3]_[0]_[7:0]							
0621	0x00	TX_J1_[3]_[1]_[7:0]							
0622 to 062D	0x00	: : :							
062E	0x00	TX_J1_[3]_[14]_[7:0]							

Table 46. Transmit J1 Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
062F	0x00	TX_J1_[3]_[15]_[7:0]							
0630	0x00	TX_J1_[4]_[0]_[7:0]							
0631	0x00	TX_J1_[4]_[1]_[7:0]							
0632 to 063D	0x00	: : :							
063E	0x00	TX_J1_[4]_[14]_[7:0]							
063F	0x00	TX_J1_[4]_[15]_[7:0]							
0640-06EF		: :							
06F0	0x00	TX_J1_[16]_[0]_[7:0]							
06F1	0x00	TX_J1_[16]_[1]_[7:0]							
06F2 to 06FD	0x00	: : :							
06FE	0x00	TX_J1_[16]_[14]_[7:0]							
06FF	0x00	TX_J1_[16]_[15]_[7:0]							
0700-08FF	0x00	Unused							

11.1.5 Transmit Side Address 0x0900 through 0x0AFF (Table)

11.1.5.1 ATM Provisioning - Addr 0x0902 through 0x0906

The modulo 2 addition of an alternating bit pattern to the ATM HEC is controlled by **TX_ATM_HEC_ENH**. (See section 5.3.4.)

The generation of the ATM HEC is controlled by **TX_ATM_HEC_INH**. (See section 5.3.4.)

TX_ATM_UTP_HEC_INH controls the HEC check at the transmit Utopia interface. (See section 5.3.1.)

TX_ATM_HEC_UDF_[1:0] and **TX_ATM_UDF** control the presence of UDF bytes, as well as the location of the HEC byte within the Utopia UDF bytes. (See section 9.4.2.)

The content of the ATM idle cells is provisioned via **TX_ATM_IDLE_GFC**, **TX_ATM_IDLE_PTI**, **TX_ATM_IDLE_CLP** (reset value of 1), and **TX_ATM_IDLE_DATA** (reset value of 0x6A). (See section 5.3.3.)

11.1.5.2 ATM Event Bits - Addr [0x0A00,0x0AF0]

These register bits are set by RHINE when it detects a HEC defect in the incoming data of tributary x arriving from the ATM Layer device. (See section 5.3.1.) These event bits contribute to the **TX_ATM_HEC_ERR_SUME** bit in register 0x0004. On reset, these event bits are cleared.

11.1.5.3 ATM Valid Cell Counter - Addr [0x0A04-6,0x0AF4-6]

The **TX_ATM_CELL_CNT_x_[22:0]** registers hold the count of the number of ATM cells that traverse the transmit system interface. (See section 5.3.2.)

Table 47. ATM Transmit Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0900-0901		Unused							
Utopia Level 3 and ATM Provisioning(Read/Write)									
0902	0x00						TX_ATM_ UDF53	TX_ATM_HEC_UDF_[1:0]	
0903	0x00					TX_ATM_ HEC_INH	TX_ATM_ UTP_HEC_ INH	TX_ATM_ UDF	Reserved
0904	0x80	TX_ATM_ IDLE_ CLP	TX_ATM_IDLE_PTI_[2:0]			TX_ATM_IDLE_GFC_[3:0]			
0905	0x6A	TX_ATM_IDLE_DATA_[7:0]							
0906	0x01								TX_ATM_HEC_ENH
0907-09FF		Unused							
ATM Event Bits - Tributary [1] (Read / cleared by uP)									
0A00	0x00								TX_ATM_HEC_ERR_[1]_E
0A01		Unused							
ATM Provisioning - Tributary [1] (Read/Write)									
0A02-0A03		Unused							
ATM Valid Cell and Idle Cell Counters - Tributary [1] (Read Only)									
0A04	0x00	TX_ATM_CELL_CNT_[1]_[7:0]							
0A05	0x00	TX_ATM_CELL_CNT_[1]_[15:8]							
0A06	0x00		TX_ATM_CELL_CNT_[1]_[22:16]						
0A07-0A0F		Unused							
ATM Event Bits - Tributary [2] (Read / cleared by uP)									
0A10	0x00								TX_ATM_HEC_ERR_[2]_E
0A11		Unused							
ATM Provisioning - Tributary [2] (Read/Write)									
0A12-0A13		Unused							
ATM Valid Cell and Idle Cell Counters - Tributary [2] (Read Only)									
0A14	0x00	TX_ATM_CELL_CNT_[2]_[7:0]							
0A15	0x00	TX_ATM_CELL_CNT_[2]_[15:8]							

Table 47. ATM Transmit Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0A16	0x00		TX_ATM_CELL_CNT_[2]_[22:16]						
0A17-0A1F		Unused							
ATM Event Bits - Tributary [3] (Read / cleared by uP)									
0A20	0x00								TX_ATM_HEC_ERR_[3]_E
0A21		Unused							
ATM Provisioning - Tributary [3] (Read/Write)									
0A22-0A23		Unused							
ATM Valid Cell and Idle Cell Counters - Tributary [3] (Read Only)									
0A24	0x00	TX_ATM_CELL_CNT_[3]_[7:0]							
0A25	0x00	TX_ATM_CELL_CNT_[3]_[15:8]							
0A26	0x00		TX_ATM_CELL_CNT_[3]_[22:16]						
0A27-0A2F		Unused							
0A30-0AEF		: :							
ATM Event Bits - Tributary [16] (Read / cleared by uP)									
0AF0	0x00								TX_ATM_HEC_ERR_[16]_E
0AF1		Unused							
ATM Provisioning - Tributary [16] (Read/Write)									
0AF2-0AF3		Unused							
ATM Valid Cell and Idle Cell Counters - Tributary [16] (Read Only)									
0AF4	0x00	TX_ATM_CELL_CNT_[16]_[7:0]							
0AF5	0x00	TX_ATM_CELL_CNT_[16]_[15:8]							
0AF6	0x00		TX_ATM_CELL_CNT_[16]_[22:16]						
0AF7--AFF		Unused							

11.1.6 Transmit Side Address 0x0B00 through 0x0CFF (Table 48)

11.1.6.1 Parity Error Event and Mask Bits- Addr 0x0B01 and 0x0B03

TX_PRTY_ERR_y_E bits are set to 1 whenever a parity error event occurs on system interface bus y. (See section 5.1.2.1.)

TX_PRTY_ERR_y_E_MASK should be written to 1 if the user desires to disable the system-interface parity monitoring for bus y. Note that the parity error event bit is given their own masks, since the parity error is dependent on the system interface configuration, rather than the number of active tributaries.

11.1.6.2 System Interface Provisioning - Addr 0x0B02 through 0x0B0D

The bits in these registers control the provisionable parameters of the system interface.

TX_PRTY_MODE_y controls the parity (even or odd) of the transmit data parity check. (See section 5.1.1.)

TX_SING_CLAV controls the single cell available signal mode, in which a single cell available signal, RX_CLAV_0, provides the multiplexed status of all “active” ports. This is enabled by setting **TX_SING_CLAV** = 1. When **TX_SING_CLAV** = 0, four cell available signals are provided, TX_CLAV_y for y = 0 to 3, which display the multiplexed status of their associated tributaries.

TX_CLAV/PDA_DSST_y controls the assert/deassert timing of the 4 *TX_CLAV_PDA_y* signals. (See sections 9.4.2 and 9.5.2.)

TX_SYSINT_WIDTH controls the width of the system interface, **TX_SYSINT_POLL** places the system interface in direct or multiplexed-polling modes for certain configurations. (See section 9.2.2.)

TX_FIFOFULL_TPA_y_[3:0] and **TX_FIFOEMPTY_TPA_y_[3:0]** allow the user to provision the FIFO fill thresholds at which *TX_CLAV_PDA_y* is to be asserted/deasserted. Operates for both Utopia/ATM and packet modes. (See section 9.5.2.)

11.1.6.3 System Interface Data Integrity Event Bits- Addr [0x0C00,0x0CF0]

These register bits are set by RHINE when it detects a defect in the incoming data arriving from the ATM or Link Layer device. (See section 5.1.1 and section 5.1.2.) These registers are active for both the POS and ATM modes of operation.

The event and second event bits contribute to the FIFO summary status bits in register 0x0004. On reset, these event and second event bits are cleared.

The **TX_FIFOERR_x_SECE** second event indicates that 1 or more cells/packets have been lost due to FIFO error since the last LATCH_EVENT. (See section 5.1.2.)

11.1.6.4 System Interface Provisioning - Addr [0x0C02,0x0CF2]

PHYADDR_x_[4:0] is RHINE FlexBus-3^{TM} address associated with tributary x of this RHINE device.

11.1.6.5 System Interface Error Counters - Addr [0x0C04,0x0CF4]

The **TX_FIFOERR_CNT_x** count the number of cells/packets lost in tributary x due to FIFO error events. (See section 5.1.2.1.)

Table 48. Transmit System Interface Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0B00		Unused							
		(Read / cleared by uP)							
0B01	0x00					TX_PRTY_ERR_[1:4]_E			
System Interface Provisioning(Read/Write)									
0B02	0x00	Unused				TX_CLK_OUT_INH_[1:4]			
0B03	0x0F					TX_PRTY_ERR_[1:4]_E_MASK			
0B04	0x00	TX_PRTY_MODE_[1:4]				TX_CLAV/PDA_DSST_[4]	TX_CLAV/PDA_DSST_[3]	TX_CLAV/PDA_DSST_[2]	TX_CLAV/PDA_DSST_[1]

Table 48. Transmit System Interface Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0B05	0x13	TX_CLAV_MODE			TX_SING_CLAV		TX_SYSINT_POLL	TX_SYS_SIZE MODE	TX_SYSINT_WIDTH
0B06	0x00					TX_FIFOEMPTY_TPA_[1]_[3:0]			
0B07	0x00					TX_FIFOFULL_TPA_[1]_[3:0]			
0B08	0x00					TX_FIFOEMPTY_TPA_[2]_[3:0]			
0B09	0x00					TX_FIFOFULL_TPA_[2]_[3:0]			
0B0A	0x00					TX_FIFOEMPTY_TPA_[3]_[3:0]			
0B0B	0x00					TX_FIFOFULL_TPA_[3]_[3:0]			
0B0C	0x00					TX_FIFOEMPTY_TPA_[4]_[3:0]			
0B0D	0x00					TX_FIFOFULL_TPA_[4]_[3:0]			
0B0E-0BFF		Unused							
System Interface Event Bits - Tributary [1] (Read / cleared by uP)									
0C00	0x00					TX_FIFOERR_[1]_SECE	Reserved		
0C01		Unused							
System Interface PHY Address Provisioning - Tributary [1] (Read/Write)									
0C02	0x1F				PHYADDR_[1]_[4:0]				
0C03		Unused							
System Interface FIFO Error Counter - Tributary [1] (Read Only)									
0C04	0x00	TX_FIFOERR_CNT_[1]_[7:0]							
0C05-0C0F		Unused							
System Interface Event Bits - Tributary [2] (Read / cleared by uP)									
0C10	0x00					TX_FIFOERR_[2]_SECE	Reserved		
0C11		Unused							
System Interface PHY Address Provisioning - Tributary [2] (Read/Write)									
0C12	0x1F				PHYADDR_[2]_[4:0]				
0C13		Unused							
System Interface FIFO Error Counter - Tributary [2] (Read Only)									
0C14	0x00	TX_FIFOERR_CNT_[2]_[7:0]							
0C15-0C1F		Unused							
System Interface Event Bits - Tributary [3] (Read / cleared by uP)									

Table 48. Transmit System Interface Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0C20	0x00					TX_FIFOERR_[3]_SECE	Reserved		
0C21		Unused							
System Interface PHY Address Provisioning - Tributary [3] (Read/Write)									
0C22	0x1F				PHYADDR_[3]_[4:0]				
0C23		Unused							
System Interface FIFO Error Counter - Tributary [3] (Read Only)									
0C24	0x00	TX_FIFOERR_CNT_[3]_[7:0]							
0C25-0C2F		Unused							
0C30-0CEF		: :							
System Interface Event Bits - Tributary [16] (Read / cleared by uP)									
0CF0	0x00					TX_FIFOERR_[16]_SECE	Reserved		
0CF1		Unused							
System Interface PHY Address Provisioning - Tributary [16] (Read/Write)									
0CF2	0x1F				PHYADDR_[16]_[4:0]				
0CF3		Unused							
System Interface FIFO Error Counter - Tributary [16] (Read Only)									
0CF4	0x00	TX_FIFOERR_CNT_[16]_[7:0]							
0CF5-0CFF		Unused							

11.1.7 Transmit Side Address 0x0D00 through 0x0EFF (Table)

11.1.7.1 POS Provisioning - Addr 0x0D02 through 0x0D06

The bits in this register control provisionable parameters of the POS processing.

TX_POS_FCS_INH controls the insertion of the POS FCS. (See section 5.2.5.)

TX_POS_FCS_BIT_ORDR controls the bit order of the FCS calculation. (See section 5.2.5.)

TX_POS_PMIN_[3:0] and **TX_POS_PMAX_[15:0]** provision the minimum and maximum length packet thresholds. (See section 5.1.3.)

TX_POS_FIFOUNDER_BYTE_[7:0] provisions the fixed byte pattern that can be inserted during periods of FIFO underflow. (See section 5.1.2.1.)

11.1.7.2 POS Second Event Bits - Addr [0x0E00,0x0EF0]

The second event bits contribute to the POS summary status bits in register 0x0005. On reset, the event and second event bits are cleared.

11.1.7.3 POS Provisioning - Addr [0x0E02-4, 0x0EF2-4]

The bits in these registers control provisionable parameters of the POS processing on a per-tributary basis.

TX_POS_ADRCTL_INS_x control the insertion of the address and control fields in the HDLC frame. (See section 5.2.4.)

TX_POS_EOP_FLAG_x control the insertion of an end of packet flag sequence. (See section 5.2.3.)

TX_POS_FCS_MODE_x controls the mode of the POS FCS. (See section 5.2.5.)

TX_POS_PMAX_ENB_x and **TX_POS_PMIN_ENB_x** control the minimum and maximum length packet check. (See section 5.1.3.)

TX_POS_FIFOUNDER_MODE_x control the insertion of a fixed byte pattern during periods of FIFO underflow. (See section 5.1.2.1.)

TX_PREHDLC_SCR_INH_x control the operation of the pre-HDLC scramblers in the POS and direct map modes. (See section 5.2.2.)

TX_POS_ADDRESS_x_[7:0] and **TX_POS_CONTROL_x_[7:0]** contain the address and control fields to be inserted into the HDLC frame. (See section 5.2.4.)

11.1.7.4 POS Error Counters - Addr [0x0E06-C,0x0EF6-C]

The POS error counters provide the capability to monitor packet errors on a per-tributary basis.

The **TX_POS_LLPKT_ERRCNT_x_[7:0]** count the number of packets received from the Link Layer device that are marked as errored. (See section 5.1.3.)

TX_POS_FIFOUNDER_ERRCNT_x_[7:0] count the number of packets corrupted due to FIFO underflow events. (See section 5.2.7.)

The **TX_POS_PMAX_ERRCNT_x_[7:0]** and **TX_POS_PMIN_ERRCNT_x_[7:0]** count the number of packets received from the Link Layer device that violate minimum and maximum packet lengths. (See section 5.1.3.)

TX_POS_PKT_CNT_x_[22:0] count the number of valid packets received from the Link Layer device. (See section 5.2.1.)

Table 49. POS Transmit Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0D00-0D01		Unused							
POS Provisioning(Read/Write)									
0D02	0x01				TX_POS_FCS_INH				RESERVED Must be set to 0
0D03	0x10				TX_POS_FCS_BIT_ORDR	TX_POS_PMIN_[3:0]			
0D04	0xDE	TX_POS_PMAX_[7:0]							
0D05	0x05	TX_POS_PMAX_[15:8]							
0D06	0x50	TX_POS_FIFOUNDER_BYTE_[7:0]							

Table 49. POS Transmit Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0D07-0DFF		Unused							
POS Event and Second Event Bits - Tributary [1] (Read / cleared by uP)									
0E00	0x00				TX_POS_FIFOUNDER_ERR_[1]_SECE	TX_POS_PMIN_ERR_[1]_SECE	TX_POS_PMAX_ERR_[1]_SECE	TX_POS_LLPKT_ERR_[1]_SECE	Reserved
0E01		Unused							
POS Provisioning - Tributary [1] (Read/Write)									
0E02	0x80	TX_PREHDLC_SCR_INH_[1]	TX_POS_FIFOUNDER_MODE_[1]	TX_POS_PMIN_ENB_[1]	TX_POS_PMAX_ENB_[1]	TX_POS_FCS_MODE_[1]	TX_POS_FCSABRT_ENB_[1]	TX_POS_EOP_FLAG_[1]	TX_POS_ADRCTL_INS_[1]
0E03	0xFF	TX_POS_ADDRESS_[1]_[7:0]							
0E04	0x03	TX_POS_CONTROL_[1]_[7:0]							
0E05		Unused							
POS Errored Packet Counters - Tributary [1] (Read Only)									
0E06	0x00	TX_POS_FIFOUNDER_ERRCNT_[1]_[7:0]							
0E07	0x00	TX_POS_LLPKT_ERRCNT_[1]_[7:0]							
0E08	0x00	TX_POS_PMAX_ERRCNT_[1]_[7:0]							
0E09	0x00	TX_POS_PMIN_ERRCNT_[1]_[7:0]							
0E0A	0x00	TX_POS_PKT_CNT_[1]_[7:0]							
0E0B	0x00	TX_POS_PKT_CNT_[1]_[8:15]							
0E0C	0x00		TX_POS_PKT_CNT_[1]_[22:16]						
0E0D-0E0F		Unused							
POS Event and Second Event Bits - Tributary [2] (Read / cleared by uP)									
0E10	0x00				TX_POS_FIFOUNDER_ERR_[2]_SECE	TX_POS_PMIN_ERR_[2]_SECE	TX_POS_PMAX_ERR_[2]_SECE	TX_POS_LLPKT_ERR_[2]_SECE	Reserved
0E11		Unused							
POS Provisioning - Tributary [2] (Read/Write)									
0E12	0x80	TX_PREHDLC_SCR_INH_[2]	TX_POS_FIFOUNDER_MODE_[2]	TX_POS_PMIN_ENB_[2]	TX_POS_PMAX_ENB_[2]	TX_POS_FCS_MODE_[2]	TX_POS_FCSABRT_ENB_[2]	TX_POS_EOP_FLAG_[2]	TX_POS_ADRCTL_INS_[2]
0E13	0xFF	TX_POS_ADDRESS_[2]_[7:0]							
0E14	0x03	TX_POS_CONTROL_[2]_[7:0]							
0E15		Unused							

Table 49. POS Transmit Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POS Errored Packet Counters - Tributary [2] (Read Only)									
0E16	0x00	TX_POS_FIFOUNDER_ERRCNT_[2]_[7:0]							
0E17	0x00	TX_POS_LLPKT_ERRCNT_[2]_[7:0]							
0E18	0x00	TX_POS_PMAX_ERRCNT_[2]_[7:0]							
0E19	0x00	TX_POS_PMIN_ERRCNT_[2]_[7:0]							
0E1A	0x00	TX_POS_PKT_CNT_[2]_[7:0]							
0E1B	0x00	TX_POS_PKT_CNT_[2]_[8:15]							
0E1C	0x00		TX_POS_PKT_CNT_[2]_[22:16]						
0E1D-0E1F		Unused							
POS Event and Second Event Bits - Tributary [3] (Read / cleared by uP)									
0E20	0x00				TX_POS_FIFOUNDER_ERR_[3]_SECE	TX_POS_PMIN_ERR_[3]_SECE	TX_POS_PMAX_ERR_[3]_SECE	TX_POS_LLPKT_ERR_[3]_SECE	Reserved
0E21		Unused							
POS Provisioning - Tributary [3] (Read/Write)									
0E22	0x80	TX_PREHDLC_SCR_INH_[3]	TX_POS_FIFOUNDER_MODE_[3]	TX_POS_PMIN_ENB_[3]	TX_POS_PMAX_ENB_[3]	TX_POS_FCS_MODE_[3]	TX_POS_FCSABRT_ENB_[3]	TX_POS_EOP_FLAG_[3]	TX_POS_ADRCTL_INS_[3]
0E23	0xFF	TX_POS_ADDRESS_[3]_[7:0]							
0E24	0x03	TX_POS_CONTROL_[3]_[7:0]							
0E25	0x00	Unused							
POS Errored Packet Counters - Tributary [3] (Read Only)									
0E26	0x00	TX_POS_FIFOUNDER_ERRCNT_[3]_[7:0]							
0E27	0x00	TX_POS_LLPKT_ERRCNT_[3]_[7:0]							
0E28	0x00	TX_POS_PMAX_ERRCNT_[3]_[7:0]							
0E29	0x00	TX_POS_PMIN_ERRCNT_[3]_[7:0]							
0E2A	0x00	TX_POS_PKT_CNT_[3]_[7:0]							
0E2B	0x00	TX_POS_PKT_CNT_[3]_[8:15]							
0E2C	0x00		TX_POS_PKT_CNT_[3]_[22:16]						
0E2D-0E2F		Unused							
0E30-0EEF		: :							
POS Event and Second Event Bits - Tributary [16] (Read / cleared by uP)									

Table 49. POS Transmit Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0EF0	0x00				TX_POS_FIFOUNDER_ERR_[16]_SECE	TX_POS_PMIN_ERR_[16]_SECE	TX_POS_PMAX_ERR_[16]_SECE	TX_POS_LLPKT_ERR_[16]_SECE	Reserved
0EF1		Unused							
POS Provisioning - Tributary [16] (Read/Write)									
0EF2	0x80	TX_PREHDLC_SCR_INH_[16]	TX_POS_FIFOUNDER_MODE_[16]	TX_POS_PMIN_ENB_[16]	TX_POS_PMAX_ENB_[16]	TX_POS_FCS_MODE_[16]	TX_POS_FCSABRT_ENB_[16]	TX_POS_EOP_FLAG_[16]	TX_POS_ADRCTL_INS_[16]
0EF3	0xFF	TX_POS_ADDRESS_[16]_[7:0]							
0EF4	0x03	TX_POS_CONTROL_[16]_[7:0]							
0EF5	0x00	Unused							
POS Errored Packet Counters - Tributary [16] (Read Only)									
0EF6	0x00	TX_POS_FIFOUNDER_ERRCNT_[16]_[7:0]							
0EF7	0x00	TX_POS_LLPKT_ERRCNT_[16]_[7:0]							
0EF8	0x00	TX_POS_PMAX_ERRCNT_[16]_[7:0]							
0EF9	0x00	TX_POS_PMIN_ERRCNT_[16]_[7:0]							
0EFA	0x00	TX_POS_PKT_CNT_[16]_[7:0]							
0EFB	0x00	TX_POS_PKT_CNT_[16]_[8:15]							
0EFC	0x00		TX_POS_PKT_CNT_[16]_[22:16]						
0EFD-1FFF		Unused							

11.2 Receive Side Registers

The following sections describe the individual pages that constitute the receive side register maps.

NOTE that many read only registers exist on the receive side that indicate the status of receiver-monitored input signals. During initialization configuration, numerous status changes can be expected, resulting in many delta and event bits being set. It is recommended to read (thereby clearing) all delta and event registers once receiver configuration has been completed and input signals are stable.

11.2.1 Receive Side Addresses 0x2000 through 0x21FF (Table)

11.2.1.1 Software Reset - Addr 0x2000

When the **RX_PROV_RESET** bit is written to a 1, all receive provisioning (read/write) registers (except the register containing the **RX_PROV_RESET** bit itself) are reset. These receive registers are held in their default state until **RX_PROV_RESET** is written to 0 or a hardware reset occurs. These registers include all receive provisioning and receive mask registers, but do not include delta, event, or second event bits.

When the **RX_STATE_RESET** bit is written to a 1, all receive status, monitoring, delta, event, second event and counter registers (the read-only registers), and all receive state machines are reset to their default states (except again for the register containing the **RX_PROV** and **RX_STATE_RESET** bits). This group should consist of all receive registers NOT reset by **RX_PROV_RESET**. The Receive side of the device remains inactive until the **RX_STATE_RESET** bit is written to 0 (or a hardware reset occurs).

The hardware reset input, *RSTB*, produces the same result as writing a 1 to both **RX_PROV_RESET** and **RX_STATE_RESET**, except that the hardware reset forces both **RX_PROV_RESET** and **RX_STATE_RESET** to 0, and the device begins operation when *RSTB* goes high (logic 1).

The **MASTER_RESET** register produces the same result as writing a 1 to **TX_** and **RX_PROV_RESET** as well as **TX_** and **RX_STATE_RESET**. RHINE begins operation when **MASTER_RESET** is written to 0 (or a hardware reset occurs).

The **LATCH_CNT** register bit can be used to transfer performance-monitoring counter values to holding registers so that they can be read. (See section 4.2.) It is cleared on reset.

The **SHORT_FRAME** register bit is used during verification. Setting this bit places the device in SONET/SDH short-frame mode.

11.2.1.2 Summary Interrupt and Mask - Addr 0x2002 and 0x200C

The **RX_SUM_INT** bit is high (logic 1) if any bit in *ADDR_[12:0]* = 0x2003 through 0x200B is high and the corresponding mask bit is low. On reset, the **RX_SUM_INT** bit is low, because all mask bits are all forced high and all delta and event bits are forced low on reset.

The logic equation for **RX_SUM_INT** is:

```
RX_SUM_INT =
    (TOH_SUMD_x && !TOH_SUMD_MASK_x) ||
    (PTR_INT_PAIS_SUMD && !PTR_INT_PAIS_SUMD_MASK) ||
    (PTR_INT_LOP_SUMD && !PTR_INT_LOP_SUMD_MASK) ||
    (PTR_INT_CONCAT_SUMD && !PTR_INT_CONCAT_SUMD_MASK) ||
    (PATH_PLM_SUMD && !PATH_PLM_SUMD_MASK) ||
    (PATH_UNEQ_SUMD && !PATH_UNEQ_SUMD_MASK) ||
    (PATH_G1_SUMD && !PATH_G1_SUMD_MASK) ||
    (PATH_J1_SUMD && !PATH_J1_SUMD_MASK) ||
    (PATH_C2_SUMD && !PATH_C2_SUMD_MASK) ||
    (RX_ATM_OCD_SUMD && !RX_ATM_OCD_SUMD_MASK) ||
    (RX_ATM_LCD_SUMD && !RX_ATM_LCD_SUMD_MASK) ||
    (RX_POS_ADRCTL_INVALID_SUMD && !RX_POS_ADRCTL_INVALID_SUMD_MASK) ||
    (RX_FIFOERR_SUME && !RX_FIFOERR_SUME_MASK) ||
    (TOH_SECE_SUM_x && !TOH_SECE_SUM_MASK_x) ||
    (PATH_B3ERR_SECE_SUM && !PATH_B3ERR_SECE_SUM_MASK) ||
    (PATH_G1ERR_SECE_SUM && !PATH_G1ERR_SECE_SUM_MASK) ||
    (RX_ATM_HEC_DROP_SECE_SUM && !RX_ATM_HEC_DROP_SECE_SUM_MASK) ||
    (RX_ATM_HEC_CORR_SECE_SUM && !RX_ATM_HEC_CORR_SECE_SUM_MASK) ||
    (RX_ATM_OCD_SECE_SUM && !RX_ATM_OCD_SECE_SUM_MASK) ||
    (RX_ATM_LCD_SECE_SUM && !RX_ATM_LCD_SECE_SUM_MASK) ||
    (RX_FIFOERR_SECE_SUM && !RX_FIFOERR_SECE_SUM_MASK) ||
    (RX_POS_PMAX_ERR_SECE_SUM && !RX_POS_PMAX_ERR_SECE_SUM_MASK) ||
    (RX_POS_PMIN_ERR_SECE_SUM && !RX_POS_PMIN_ERR_SECE_SUM_MASK) ||
    (RX_POS_PABORT_ERR_SECE_SUM && !RX_POS_PABORT_ERR_SECE_SUM_MASK) ||
    (RX_POS_FCS_ERR_SECE_SUM && !RX_POS_FCS_ERR_SECE_SUM_MASK) ||
    (LATCH_E && !LATCH_E_MASK) ||
    (PTR_PROC_CONCAT_SUMD && !PTR_PROC_CONCAT_SUMD_MASK) ||
    (PTR_PROC_PAIS_SUMD && !PTR_PROC_PAIS_SUMD_MASK) ||
    (PTR_PROC_LOP_SUMD && !PTR_PROC_LOP_SUMD_MASK) ||
    (PTR_GEN_FIFO_SUME && !PTR_GEN_FIFO_SUME_MASK) ||
```

(**PTR_GEN_JUSTCNT_SECE_SUM** && !**PTR_GEN_JUSTCNT_SECE_SUM_MASK**) ||
(**PTR_PROC_JUSTCNT_SECE_SUM** && !**PTR_PROC_JUSTCNT_SECE_SUM_MASK**) ||
(**APS_ERR_SUMD** && !**APS_ERR_SUMD_MASK**) ||
(**APS_K1K2_SUMD** && !**APS_K1K2_SUMD_MASK**)

The **RX_SUM_INT_MASK** bit is used to disable the **RX_SUM_INT** contribution to RHINE interrupt output, *INTB*. The **RX_SUM_INT** bit contributes to the *INTB* output, unless the **RX_SUM_INT_MASK** bit in *ADDR_[13:0]* = 0x200C is set.

The **RX_APS_INT** bit is high (logic 1) if any of its contributing delta bits is high and the corresponding mask bit is low. On reset, the **RX_APS_INT** bit is low, because all mask bits are all forced high and all delta bits are forced low on reset.

The logic equation for **RX_APS_INT** is:

RX_APS_INT =
(**RX_K1_[1]_D** && !**RX_K1_[1]_D_MASK**) ||
(**RX_K2_[1]_D** && !**RX_K2_[1]_D_MASK**) ||
(**K1_UNSTAB_[1]_D** && !**K1_UNSTAB_[1]_D_MASK**) ||
(**RX_K1_[2]_D** && !**RX_K1_[2]_D_MASK**) ||
(**RX_K2_[2]_D** && !**RX_K2_[2]_D_MASK**) ||
(**K1_UNSTAB_[2]_D** && !**K1_UNSTAB_[2]_D_MASK**) ||
(**RX_K1_[3]_D** && !**RX_K1_[3]_D_MASK**) ||
(**RX_K2_[3]_D** && !**RX_K2_[3]_D_MASK**) ||
(**K1_UNSTAB_[3]_D** && !**K1_UNSTAB_[3]_D_MASK**) ||...||
(**RX_K1_[16]_D** && !**RX_K1_[16]_D_MASK**) ||
(**RX_K2_[16]_D** && !**RX_K2_[16]_D_MASK**) ||
(**K1_UNSTAB_[16]_D** && !**K1_UNSTAB_[16]_D_MASK**) ||
(**APS_K1_[1]_D** && !**APS_K1_[1]_D_MASK**) ||
(**APS_K2_[1]_D** && !**APS_K2_[1]_D_MASK**) ||
(**APS_UNSTAB_[1]_D** && !**APS_UNSTAB_[1]_D_MASK**) ||
(**APS_K1_[2]_D** && !**APS_K1_[2]_D_MASK**) ||
(**APS_K2_[2]_D** && !**APS_K2_[2]_D_MASK**) ||
(**APS_UNSTAB_[2]_D** && !**APS_UNSTAB_[2]_D_MASK**) ||
(**APS_K1_[3]_D** && !**APS_K1_[3]_D_MASK**) ||
(**APS_K2_[3]_D** && !**APS_K2_[3]_D_MASK**) ||
(**APS_UNSTAB_[3]_D** && !**APS_UNSTAB_[3]_D_MASK**) ||...||
(**APS_K1_[16]_D** && !**APS_K1_[16]_D_MASK**) ||
(**APS_K2_[16]_D** && !**APS_K2_[16]_D_MASK**) ||
(**APS_UNSTAB_[16]_D** && !**APS_UNSTAB_[16]_D_MASK**)

The **RX_APS_INT_MASK** bit is used to disable RHINE APS interrupt output, *APS_INTB*. The **RX_APS_INT** bit drives to the *APS_INTB* output, unless the **RX_APS_INT_MASK** bit in *ADDR_[13:0]* = 0x200C is set.

11.2.1.3 Functional Deltas, Events, Summary Status, and Masks - Addr 0x2003 through 0x2013

The summary delta and summary event bits contribute to the **RX_SUM_INT** bit. On reset, all mask bits are set high, which forces all summary delta and summary event bits low.

The **LATCH_E** is set to 1 whenever RHINE updates the second event bits and the counters in all register maps, due to either the expiration of SEC_EVENT or the setting of **LATCH_CNT.**

TOH_SUMD_x is a function of the TOH/SOH delta bits in 0x2200-2201, 0x2240-2241, 0x2280-2281 through 0x25C0-25C1 and their masks in 0x2204-2205, 0x2244-2245, and 0x2284-2285 through 0x25C4-0x25C5. The logic equation for this bit is:

TOH_SUMD_x =
(**RX_LOS_x_D** && !**RX_LOS_x_D_MASK**) ||
(**RX_LOSEXT_x_D** && !**RX_LOSEXT_x_D_MASK**) ||
(**RX_LOC_x_D** && !**RX_LOC_x_D_MASK**) ||
(**RX_OOF_x_D** && !**RX_OOF_x_D_MASK**) ||
(**RX_LOF_x_D** && !**RX_LOF_x_D_MASK**) ||
(**J0_OOF_x_D** && !**J0_OOF_x_D_MASK**) ||
(**RX_J0_x_D** && !**RX_J0_x_D_MASK**) ||
(**B2_ERR_SF_x_D** && !**B2_ERR_SF_x_D_MASK**) ||
(**B2_ERR_SD_x_D** && !**B2_ERR_SD_x_D_MASK**) ||
(**RX_LAIS_x_D** && !**RX_LAIS_x_D_MASK**) ||
(**RX_LRDI_x_D** && !**RX_LRDI_x_D_MASK**) ||
(**RX_K1_x_D** && !**RX_K1_x_D_MASK**) ||
(**K1_UNSTAB_x_D** && !**K1_UNSTAB_x_D_MASK**) ||
(**RX_K2_x_D** && !**RX_K2_x_D_MASK**) ||
(**RX_S1_x_D** && !**RX_S1_x_D_MASK**)

TOH_SECE_SUM_x is a function of the TOH/SOH second event bits in 0x2202, 0x2242, 0x2282 through 0x25C2 and their masks in 0x2206, 0x2246, and 0x2286 through 0x25C6. The logic equation for this bit is:

TOH_SECE_SUM_x =
(**RX_OOF_x_SECE** && !**RX_OOF_x_SECE_MASK**) ||
(**RX_LOF_x_SECE** && !**RX_LOF_x_SECE_MASK**) ||
(**B1ERR_x_SECE** && !**B1ERR_x_SECE_MASK**) ||
(**B2ERR_x_SECE** && !**B2ERR_x_SECE_MASK**) ||
(**S1FAIL_x_SECE** && !**S1FAIL_x_SECE_MASK**) ||
(**M1ERR_x_SECE** && !**M1ERR_x_SECE_MASK**)

PTR_INT_PAIS_SUMD is a function of the PAIS Pointer Interpreter delta bits in [0x2B00,0x2BF0] and the receive tributary masks in [0x2100,0x21F0]. The logic equation for this bit is:

PTR_INT_PAIS_SUMD =
(**RX_PI_PAIS_[1]_D** && !**RX_TRIB_[1]_MASK**) ||
(**RX_PI_PAIS_[2]_D** && !**RX_TRIB_[2]_MASK**) ||
(**RX_PI_PAIS_[3]_D** && !**RX_TRIB_[3]_MASK**) ||...||
(**RX_PI_PAIS_[15]_D** && !**RX_TRIB_[15]_MASK**) ||
(**RX_PI_PAIS_[16]_D** && !**RX_TRIB_[16]_MASK**)|

PTR_INT_LOP_SUMD is a function of the LOP Pointer Interpreter delta bits in [0x2B00,0x2BF0] and the receive tributary masks in [0x2100,0x21F0]. The logic equation for this bit is:

PTR_INT_LOP_SUMD =
(**RX_PI_LOP_[1]_D** && !**RX_TRIB_[1]_MASK**) ||
(**RX_PI_LOP_[2]_D** && !**RX_TRIB_[2]_MASK**) ||
(**RX_PI_LOP_[3]_D** && !**RX_TRIB_[3]_MASK**) ||...||
(**RX_PI_LOP_[15]_D** && !**RX_TRIB_[15]_MASK**) ||
(**RX_PI_LOP_[16]_D** && !**RX_TRIB_[16]_MASK**)

PTR_INT_CONCAT_SUMD is a function of the Pointer Interpreter delta bits in register 0x2A02 and the concatenation indication masks in 0x2A04. The logic equation for this bit is:

PTR_INT_CONCAT_SUMD =
(**RX_PI_CONCAT_[4]_D** && !**RX_PI_CONCAT_[4]_MASK**) ||
(**RX_PI_CONCAT_[3]_D** && !**RX_PI_CONCAT_[3]_MASK**) ||
(**RX_PI_CONCAT_[2]_D** && !**RX_PI_CONCAT_[2]_MASK**) ||
(**RX_PI_CONCAT_[1]_D** && !**RX_PI_CONCAT_[1]_MASK**) ||
(**RX_PI_CONCAT_[0]_D** && !**RX_PI_CONCAT_[0]_MASK**)

PATH_PLM_SUMD is a function of the Path Monitoring delta bits in [0x2D00,0x2DF0] and the receive tributary masks. The logic equation for this bit is:

```
PATH_PLM_SUMD =
        (RX_PLM_[1]_D && !RX_TRIB_[1]_MASK) ||
        (RX_PLM_[2]_D && !RX_TRIB_[2]_MASK) ||
        (RX_PLM_[3]_D && !RX_TRIB_[3]_MASK) ||...||
        (RX_PLM_[16]_D && !RX_TRIB_[16]_MASK)
```

PATH_C2_SUMD is a function of the Path Monitoring delta bits in [0x2D00,0x2DF0] and the receive tributary masks. The logic equation for this bit is:

```
PATH_C2_SUMD =
        (RX_C2_[1]_D && !RX_TRIB_[1]_MASK) ||
        (RX_C2_[2]_D && !RX_TRIB_[2]_MASK) ||
        (RX_C2_[3]_D && !RX_TRIB_[3]_MASK) ||...||
        (RX_C2_[16]_D && !RX_TRIB_[16]_MASK)
```

PATH_UNEQ_SUMD is a function of the Path Monitoring delta bits in [0x2D00,0x2DF0] and the receive tributary masks. The logic equation for this bit is:

```
PATH_UNEQ_SUMD =
        (RX_UNEQ_[1]_D && !RX_TRIB_[1]_MASK) ||
        (RX_UNEQ_[2]_D && !RX_TRIB_[2]_MASK) ||
        (RX_UNEQ_[3]_D && !RX_TRIB_[3]_MASK) ||...||
        (RX_UNEQ_[16]_D && !RX_TRIB_[16]_MASK)
```

PATH_G1_SUMD is a function of the Path Monitoring delta bits in [0x2D00,0x2DF0] and the receive tributary masks. The logic equation for this bit is:

```
PATH_G1_SUMD =
        (RX_G1_[1]_D && !RX_TRIB_[1]_MASK) ||
        (RX_G1_[2]_D && !RX_TRIB_[2]_MASK) ||
        (RX_G1_[3]_D && !RX_TRIB_[3]_MASK) ||...||
        (RX_G1_[16]_D && !RX_TRIB_[16]_MASK)
```

PATH_J1_SUMD is a function of the Path J1 Monitoring interrupt and delta bits in 0x2C02. The logic equation for this bit is:

```
PATH_J1_SUMD = J1_CAP_E || RX_J1_D || J1_OOF_D
```

PATH_B3ERR_SECE_SUM is a function of the Path B3 error second event bits in [0x2D00,0x2DF0] and the receive tributary masks. The logic equation for this bit is:

```
PATH_B3ERR_SECE_SUM =
        (B3ERR_[1]_SECE && !RX_TRIB_[1]_MASK) ||
        (B3ERR_[2]_SECE && !RX_TRIB_[2]_MASK) ||
        (B3ERR_[3]_SECE && !RX_TRIB_[3]_MASK) ||...||
        (B3ERR_[16]_SECE && !RX_TRIB_[16]_MASK)
```

PATH_G1ERR_SECE_SUM is a function of the Path G1 error second event bits in [0x2D00,0x2DF0] and the receive tributary masks. The logic equation for this bit is:

```
PATH_G1ERR_SECE_SUM =
        (G1ERR_[1]_SECE && !RX_TRIB_[1]_MASK) ||
        (G1ERR_[2]_SECE && !RX_TRIB_[2]_MASK) ||
        (G1ERR_[3]_SECE && !RX_TRIB_[3]_MASK) ||...||
        (G1ERR_[16]_SECE && !RX_TRIB_[16]_MASK)
```

RX_ATM_OCD_SUMD is a function of the ATM OCD delta bits in [0x2F00,0x2FF0] and the receive ATM tributary masks. The logic equation for this bit is:

RX_ATM_OCD_SUMD =
(**RX_ATM_OCD_[1]_D** && !**RX_ATM_TRIB_[1]_MASK**) ||
(**RX_ATM_OCD_[2]_D** && !**RX_ATM_TRIB_[2]_MASK**) ||
(**RX_ATM_OCD_[3]_D** && !**RX_ATM_TRIB_[3]_MASK**) ||...||
(**RX_ATM_OCD_[16]_D** && !**RX_ATM_TRIB_[16]_MASK**)

RX_ATM_LCD_SUMD is a function of the ATM LCD delta bits in [0x2F00,0x2FF0] and the receive ATM tributary masks. The logic equation for this bit is:

RX_ATM_LCD_SUMD =
(**RX_ATM_LCD_[1]_D** && !**RX_ATM_TRIB_[1]_MASK**) ||
(**RX_ATM_LCD_[2]_D** && !**RX_ATM_TRIB_[2]_MASK**) ||
(**RX_ATM_LCD_[3]_D** && !**RX_ATM_TRIB_[3]_MASK**) ||...||
(**RX_ATM_LCD_[16]_D** && !**RX_ATM_TRIB_[16]_MASK**)

RX_FIFOERR_SUME is a function of the FIFO overflow indication event bits in [0x3100,0x31F0] and the receive tributary masks. The logic equation for this bit is:

RX_FIFOERR_SUME =
(**RX_FIFOERR_[1]_E** && !**RX_TRIB_[1]_MASK**) ||
(**RX_FIFOERR_[2]_E** && !**RX_TRIB_[2]_MASK**) ||
(**RX_FIFOERR_[3]_E** && !**RX_TRIB_[3]_MASK**) ||...||
(**RX_FIFOERR_[16]_E** && !**RX_TRIB_[16]_MASK**)

RX_ATM_OCD_SECE_SUM is a function of the ATM OCD second event bits in [0x2F00,0x2FF0] and the receive ATM tributary masks. The logic equation for this bit is:

RX_ATM_OCD_SECE_SUM =
(**RX_ATM_OCD_[1]_SECE** && !**RX_ATM_TRIB_[1]_MASK**) ||
(**RX_ATM_OCD_[2]_SECE** && !**RX_ATM_TRIB_[2]_MASK**) ||
(**RX_ATM_OCD_[3]_SECE** && !**RX_ATM_TRIB_[3]_MASK**) ||...||
(**RX_ATM_OCD_[16]_SECE** && !**RX_ATM_TRIB_[16]_MASK**)

RX_ATM_LCD_SECE_SUM is a function of the ATM LCD second event bits in [0x2F00,0x2FF0] and the receive ATM tributary masks. The logic equation for this bit is:

RX_ATM_LCD_SECE_SUM =
(**RX_ATM_LCD_[1]_SECE** && !**RX_ATM_TRIB_[1]_MASK**) ||
(**RX_ATM_LCD_[2]_SECE** && !**RX_ATM_TRIB_[2]_MASK**) ||
(**RX_ATM_LCD_[3]_SECE** && !**RX_ATM_TRIB_[3]_MASK**) ||...||
(**RX_ATM_LCD_[16]_SECE** && !**RX_ATM_TRIB_[16]_MASK**)

RX_ATM_HEC_CORR_SECE_SUM is a function of the ATM HEC errored second event bits in [0x2F00,0x2FF0] and the receive ATM tributary masks. The logic equation for this bit is:

RX_ATM_HEC_CORR_SECE_SUM =
(**RX_ATM_HEC_CORR_[1]_SECE** && !**RX_ATM_TRIB_[1]_MASK**) ||
(**RX_ATM_HEC_CORR_[2]_SECE** && !**RX_ATM_TRIB_[2]_MASK**) ||
(**RX_ATM_HEC_CORR_[3]_SECE** && !**RX_ATM_TRIB_[3]_MASK**) ||...||
(**RX_ATM_HEC_CORR_[16]_SECE** && !**RX_ATM_TRIB_[16]_MASK**)

RX_ATM_HEC_DROP_SECE_SUM is a function of the ATM HEC errored second event bits in [0x2F00,0x2FF0] and the receive ATM tributary masks. The logic equation for this bit is:

RX_ATM_HEC_DROP_SECE_SUM =
(**RX_ATM_HEC_DROP_[1]_SECE** && !**RX_ATM_TRIB_[1]_MASK**) ||
(**RX_ATM_HEC_DROP_[2]_SECE** && !**RX_ATM_TRIB_[2]_MASK**) ||
(**RX_ATM_HEC_DROP_[3]_SECE** && !**RX_ATM_TRIB_[3]_MASK**) ||...||
(**RX_ATM_HEC_DROP_[16]_SECE** && !**RX_ATM_TRIB_[16]_MASK**)

RX_FIFOERR_SECE_SUM is a function of the FIFO overflow indication second event bits in [0x3100,0x31F0] and the receive tributary masks. The logic equation for this bit is:

RX_FIFOERR_SECE_SUM =
(**RX_FIFOERR_[1]_SECE** && !**RX_TRIB_[1]_MASK**) ||
(**RX_FIFOERR_[2]_SECE** && !**RX_TRIB_[2]_MASK**) ||
(**RX_FIFOERR_[3]_SECE** && !**RX_TRIB_[3]_MASK**) ||...||
(**RX_FIFOERR_[16]_SECE** && !**RX_TRIB_[16]_MASK**)

RX_POS_FCS_ERR_SECE_SUM is a function of the POS FCS errored second event bits in [0x3300,0x33F0] and the receive POS tributary masks. The logic equation for this bit is:

RX_POS_FCS_ERR_SECE_SUM =
(**RX_POS_FCS_ERR_[1]_SECE** && !**RX_POS_TRIB_[1]_MASK**) ||
(**RX_POS_FCS_ERR_[2]_SECE** && !**RX_POS_TRIB_[2]_MASK**) ||
(**RX_POS_FCS_ERR_[3]_SECE** && !**RX_POS_TRIB_[3]_MASK**) ||...||
(**RX_POS_FCS_ERR_[16]_SECE** && !**RX_POS_TRIB_[16]_MASK**)

RX_POS_PMAX_ERR_SECE_SUM is a function of the POS maximum packet length errored second event bits in [0x3300,0x33F0] and the receive tributary masks. The logic equation for this bit is:

RX_POS_PMAX_ERR_SECE_SUM =
(**RX_POS_PMAX_ERR_[1]_SECE** && !**RX_POS_TRIB_[1]_MASK**) ||
(**RX_POS_PMAX_ERR_[2]_SECE** && !**RX_POS_TRIB_[2]_MASK**) ||
(**RX_POS_PMAX_ERR_[3]_SECE** && !**RX_POS_TRIB_[3]_MASK**) ||...||
(**RX_POS_PMAX_ERR_[16]_SECE** && !**RX_POS_TRIB_[16]_MASK**)

RX_POS_PMIN_ERR_SECE_SUM is a function of the POS minimum packet length errored second event bits in [0x3300,0x33F0] and the receive tributary masks. The logic equation for this bit is:

RX_POS_PMIN_ERR_SECE_SUM =
(**RX_POS_PMIN_ERR_[1]_SECE** && !**RX_POS_TRIB_[1]_MASK**) ||
(**RX_POS_PMIN_ERR_[2]_SECE** && !**RX_POS_TRIB_[2]_MASK**) ||
(**RX_POS_PMIN_ERR_[3]_SECE** && !**RX_POS_TRIB_[3]_MASK**)||...||
(**RX_POS_PMIN_ERR_[16]_SECE** && !**RX_POS_TRIB_[16]_MASK**)

RX_POS_PABORT_ERR_SECE_SUM is a function of the POS aborted packet errored second event bits in [0x3300,0x33F0] and the receive tributary masks. The logic equation for this bit is:

RX_POS_PABORT_ERR_SECE_SUM =
(**RX_POS_PABORT_ERR_[1]_SECE** && !**RX_POS_TRIB_[1]_MASK**) ||
(**RX_POS_PABORT_ERR_[2]_SECE** && !**RX_POS_TRIB_[2]_MASK**) ||
(**RX_POS_PABORT_ERR_[3]_SECE** && !**RX_POS_TRIB_[3]_MASK**) ||...||
(**RX_POS_PABORT_ERR_[16]_SECE** && !**RX_POS_TRIB_[16]_MASK**)

RX_POS_ADRCTL_INVALID_SUMD is a function of the POS invalid address and control field event bits in [0x3300,0x33F0] and the receive tributary masks. The logic equation for this bit is:

RX_POS_ADRCTL_INVALID_SUMD =
(**RX_POS_ADRCTL_INVALID_[1]_D** && !**RX_POS_TRIB_[1]_MASK**) ||
(**RX_POS_ADRCTL_INVALID_[2]_D** && !**RX_POS_TRIB_[2]_MASK**) ||
(**RX_POS_ADRCTL_INVALID_[3]_D** && !**RX_POS_TRIB_[3]_MASK**) ||...||
(**RX_POS_ADRCTL_INVALID_[16]_D** && !**RX_POS_TRIB_[16]_MASK**)

PTR_PROC_CONCAT_SUMD is a function of the Pointer Processor delta bits in register 0x2600-2602 and the concatenation indication masks in 0x2604-2606. The logic equation for this bit is:

PTR_PROC_CONCAT_SUMD =
(**RX_PP_CONCAT_[20]_D** && !**RX_PP_CONCAT_[20]_MASK**) ||
(**RX_PP_CONCAT_[19]_D** && !**RX_PP_CONCAT_[9]_MASK**) ||

(**RX_PP_CONCAT_[18]_D** && !**RX_PP_CONCAT_[18]_MASK**) ||...||
(**RX_PP_CONCAT_[1]_D** && !**RX_PP_CONCAT_[1]_MASK**) ||
(**RX_PP_CONCAT_[0]_D** && !**RX_PP_CONCAT_[0]_MASK**)

PTR_PROC_PAIS_SUMD is a function of the PAIS Pointer Processor delta bits in [0x2700,0x27F0] and the receive tributary masks in [0x2100,0x21F0]. The logic equation for this bit is:

PTR_PROC_PAIS_SUMD =
((**RX_PP_PAIS_[1]_[1]_D || RX_PP_PAIS_[1]_[2]_D || RX_PP_PAIS_[1]_[3]_D**)&&
!**RX_TRIB_[1]_MASK**) ||
((**RX_PP_PAIS_[2]_[1]_D || RX_PP_PAIS_[2]_[2]_D || RX_PP_PAIS_[2]_[3]_D**) &&
!**RX_TRIB_[2]_MASK**) ||
((**RX_PP_PAIS_[3]_[1]_D || RX_PP_PAIS_[3]_[2]_D || RX_PP_PAIS_[3]_[3]_D**) &&
!**RX_TRIB_[3]_MASK**) ||...||
((**RX_PP_PAIS_[15]_[1]_D || RX_PP_PAIS_[15]_[2]_D || RX_PP_PAIS_[15]_[3]_D**) &&
!**RX_TRIB_[15]_MASK**) ||
((**RX_PP_PAIS_[16]_[1]_D || RX_PP_PAIS_[16]_[2]_D || RX_PP_PAIS_[16]_[3]_D**) &&
!**RX_TRIB_[16]_MASK**)

PTR_PROC_LOP_SUMD is a function of the LOP Pointer Processor delta bits in [0x2700,0x27F0] and the receive tributary masks in [0x2100,0x21F0]. The logic equation for this bit is:

PTR_PROC_LOP_SUMD =
((**RX_PP_LOP_[1]_[1]_D** || **RX_PP_LOP_[1]_[2]_D** || **RX_PP_LOP_[1]_[3]_D**)&&
!**RX_TRIB_[1]_MASK**) ||
((**RX_PP_LOP_[2]_[1]_D** || **RX_PP_LOP_[2]_[2]_D** || **RX_PP_LOP_[2]_[3]_D**) &&
!**RX_TRIB_[2]_MASK**) ||
((**RX_PP_LOP_[3]_[1]_D** || **RX_PP_LOP_[3]_[2]_D** || **RX_PP_LOP_[3]_[3]_D**) &&
!**RX_TRIB_[3]_MASK**) ||...||
((**RX_PP_LOP_[15]_[1]_D** || **RX_PP_LOP_[15]_[2]_D** || **RX_PP_LOP_[15]_[3]_D**) &&
!**RX_TRIB_[15]_MASK**) ||
((**RX_PP_LOP_[16]_[1]_D** || **RX_PP_LOP_[16]_[2]_D** || **RX_PP_LOP_[16]_[3]_D**) &&
!**RX_TRIB_[16]_MASK**)

PTR_PROC_JUSTCNT_SECE_SUM is a function of the pointer processor positive and negative justification event bits in [0x2701,0x27F1] and the receive tributary masks in [0x2100,0x21F0]. The logic equation for this bit is:

PTR_PROC_JUSCNT_SECE_SUM =
((**RX_PP_POSCNT_SECE_[1]_[1]** || **RX_PP_POSCNT_SECE_[1]_[2] ||
RX_PP_POSCNT_SECE_[1]_[3] || RX_PP_NEGCNT_SECE_[1]_[1] ||
RX_PP_NEGCNT_SECE_[1]_[2]** || **RX_PP_NEGCNT_SECE_[1]_[3]**) &&
!**RX_TRIB_[1]_MASK**) ||
((**RX_PP_POSCNT_SECE_[2]_[1]** || **RX_PP_POSCNT_SECE_[2]_[2] ||
RX_PP_POSCNT_SECE_[2]_[3] || RX_PP_NEGCNT_SECE_[2]_[1] ||
RX_PP_NEGCNT_SECE_[2]_[2]** || **RX_PP_NEGCNT_SECE_[2]_[3]**) || &&
!**RX_TRIB_[2]_MASK**) ||...||
((**RX_PP_POSCNT_SECE_[16]_[1]** || **RX_PP_POSCNT_SECE_[16]_[2] ||
RX_PP_POSCNT_SECE_[16]_[3] || RX_PP_NEGCNT_SECE_[16]_[1] ||
RX_PP_NEGCNT_SECE_[16]_[2]** || **RX_PP_NEGCNT_SECE_[16]_[3]**) &&
!**RX_TRIB_[16]_MASK**)

PTR_GEN_FIFO_SUME is a function of pointer generator FIFO event bits in [0x2902,29F2] and the receive tributary masks in [0x2100, 0x21F0]. The logic equation for this bit is:

PTR_GEN_FIFO_SUME =
((**RX_PG_FIFO_[1]_[1]_E || RX_PG_FIFO_[1]_[2]_E || RX_PG_FIFO_[1]_[3]_E**) &&

!**RX_TRIB_[1]_MASK**) ||
((**RX_PG_FIFO_[2]_[1]_E || RX_PG_FIFO_[2]_[2]_E || RX_PG_FIFO_[2]_[3]_E**) &&
!**RX_TRIB_[2]_MASK**) ||...||
((**RX_PG_FIFO_[16]_[1]_E || RX_PG_FIFO_[16]_[2]_E || RX_PG_FIFO_[16]_[3]_E**) &&
!**RX_TRIB_[16]_MASK**)

PTR_GEN_JUSTCNT_SECE_SUM is a function of the pointer generator positive and negative justification event bits in [0x2900,0x29F0] and the receive tributary masks in [0x2100,0x21F0]. The logic equation for this bit is:

PTR_GEN_JUSCNT_SECE_SUM =
((**RX_PG_POSCNT_SECE_[1]_[1] || RX_PG_POSCNT_SECE_[1]_[2] ||
RX_PG_POSCNT_SECE_[1]_[3] || RX_PG_NEGCNT_SECE_[1]_[1] ||
RX_PG_NEGCNT_SECE_[1]_[2] || RX_PG_NEGCNT_SECE_[1]_[3]**) &&
!**RX_TRIB_[1]_MASK**) ||
((**RX_PG_POSCNT_SECE_[2]_[1] || RX_PG_POSCNT_SECE_[2]_[2] ||
RX_PG_POSCNT_SECE_[2]_[3] || RX_PG_NEGCNT_SECE_[2]_[1] ||
RX_PG_NEGCNT_SECE_[2]_[2] || RX_PG_NEGCNT_SECE_[2]_[3]**) || &&
!**RX_TRIB_[2]_MASK**)||...||
((**RX_PG_POSCNT_SECE_[16]_[1] || RX_PG_POSCNT_SECE_[16]_[2] ||
RX_PG_POSCNT_SECE_[16]_[3] || RX_PG_NEGCNT_SECE_[16]_[1] ||
RX_PG_NEGCNT_SECE_[16]_[2] || RX_PG_NEGCNT_SECE_[16]_[3]**) &&
!**RX_TRIB_[16]_MASK**)

APS_ERR_SUMD is a function of the APS interface error event bits and their respective masks. The logic equation for this bit is:

APS_ERR_SUMD =
(**APS_OOF_[1]_D** && !**APS_OOF_[1]_D_MASK**) ||
(**APS_OOF_[2]_D** && !**APS_OOF_[2]_D_MASK**) ||
(**APS_OOF_[3]_D** && !**APS_OOF_[3]_D_MASK**) ||
(**APS_OOF_[4]_D** && !**APS_OOF_[4]_D_MASK**) ||
(**APS_B1ERR_SECE** && !**APS_B1ERR_SECE_MASK**) ||
(**APS_MISALIGN_ERR_E** && !**APS_MISALIGN_ERR_E_MASK**)

APS_K1K2_SUMD is a function of the APS K1K2 monitors and their respective masks. The logic equation for this bit is:

APS_K1K2_SUMD =
(**APS_K1_[1]_D** && !**APS_K1_[1]_D_MASK**) ||
(**APS_K2_[1]_D** && !**APS_K2_[1]_D_MASK**) ||
(**APS_UNSTAB_[1]_D** && !**APS_UNSTAB_[1]_D_MASK**) ||
(**APS_K1_[2]_D** && !**APS_K1_[2]_D_MASK**) ||
(**APS_K2_[2]_D** && !**APS_K2_[2]_D_MASK**) ||
(**APS_UNSTAB_[2]_D** && !**APS_UNSTAB_[2]_D_MASK**) ||
(**APS_K1_[3]_D** && !**APS_K1_[3]_D_MASK**) ||
(**APS_K2_[3]_D** && !**APS_K2_[3]_D_MASK**) ||
(**APS_UNSTAB_[3]_D** && !**APS_UNSTAB_[3]_D_MASK**) ||...||
(**APS_K1_[16]_D** && !**APS_K1_[16]_D_MASK**) ||
(**APS_K2_[16]_D** && !**APS_K2_[16]_D_MASK**) ||
(**APS_UNSTAB_[16]_D** && !**APS_UNSTAB_[16]_D_MASK**)

11.2.1.4 Configuration, Loopback, and PM Counter Latch Enable Provisioning

If the **SONET_T_TO_R_LOOP_x** bit is high, RHINE loops the transmitted signal(s) back to the Receive side input. The loopback is provided on a per tributary basis and the enable bits are located at bit position 7 at address locations 0x2100, 0x2110, 0x2120...etc.

If the **CNT_SEC_EN** bit is high, the performance-monitoring counters are latched on the rising edge of the SEC_EVENT internal signal. (See section 4.2.). This bit is located at 0x2020 bit position 4.

The **BIT_BLKCNT** bit provisions performance-monitoring counters to count bit errors or block errors. (See section 6.5.2 and section 6.6.6.) This bit is located at 0x2020 bit position 6.

The **RX_PP_CONFIG_[20:0]** Address 0x2024-0x2026, **RX_LINE_CONFIG_[4:0]** at Address 0x2028, determine the structure of the generated signal.

11.2.1.5 Tributary Summary Delta Bits - Addr 0x2030/31

The tributary summary delta/event bits do not contribute to the **RX_SUM_INT** bit. These bits reflect the contributors to the functional summary deltas and events on a per-tributary basis. These are provided to allow the user to quickly determine the tributary that is the source of an interrupt, minimizing register read cycles. On reset, all mask bits are set high, which forces all tributary summary delta/event bits low.

The logic equation for **RX_TRIB_x_SUMD** is:

11.2.1.6 RX_TRIB_x_SUMD =

(**RX_PP_PAIS_[x]_[j]_D** && !**RX_TRIB_x_MASK**) ||
(RX_PP_LOP_[x]_[j]_D && !**RX_TRIB_x_MASK)** ||
(RX_PI_PAIS_[x]_[j]_D && !**RX_TRIB_x_MASK)** ||
(RX_PI_LOP_[x]_[j]_D && !**RX_TRIB_x_MASK)** ||
(RX_PLM_x_D && !**RX_TRIB_x_MASK)** ||
(RX_C2_x_D && !**RX_TRIB_x_MASK)** ||
(RX_UNEQ_x_D && !**RX_TRIB_x_MASK)** ||
(RX_G1_x_D && !**RX_TRIB_x_MASK)** ||
(B3ERR_x_SECE && !**RX_TRIB_x_MASK)** ||
(**G1ERR_x_SECE** && !**RX_TRIB_x_MASK)** ||
(RX_ATM_OCD_x_D && !**RX_ATM_TRIB_x_MASK)** ||
(RX_ATM_LCD_x_D && !**RX_ATM_TRIB_x_MASK)** ||
(RX_FIFOERR_x_E && !**RX_TRIB_x_MASK)** ||
(RX_ATM_OCD_x_SECE && !**RX_ATM_TRIB_x_MASK)** ||
(RX_ATM_LCD_x_SECE && !**RX_ATM_TRIB_x_MASK)** ||
(RX_ATM_HEC_CORR_x_SECE && !**RX_ATM_TRIB_x_MASK)** ||
(**RX_ATM_HEC_DROP_x_SECE** && !**RX_ATM_TRIB_x_MASK)** ||
(RX_FIFOERR_x_SECE && !**RX_TRIB_x_MASK**) ||
(RX_POS_FCS_ERR_x_SECE && !**RX_POS_TRIB_x_MASK**) ||
(RX_POS_PMAX_ERR_x_SECE && !**RX_POS_TRIB_x_MASK**) ||
(**RX_POS_PMIN_ERR_x_SECE** && !**RX_POS_TRIB_x_MASK)** ||
(RX_POS_PABORT_ERR_x_SECE && !**RX_POS_TRIB_x_MASK**) ||
(**RX_POS_ADRCTL_INVALID_x_D** && !**RX_POS_TRIB_x_MASK**) ||

Tributary Configuration Provisioning and Masks

SYS_T_TO_R_LOOP_x controls the loopback of tributary **x** from the transmit to receive direction. (See section 6.15.1.)

LOF_INH When set to 1, this bit prevents the LOF signal from causing an interrupt condition. It also prevents LOF from contributing to PAIS generation. This bit is strictly meant for test purposes.

RX_TRIB_x_MASK should be written to 1 if the user desires to disable receive monitoring of tributary x. **RX_POS_TRIB_x_MASK** should be written to 1 if the user desires to disable receive POS monitoring of tributary x. **RX_ATM_TRIB_x_MASK** should be written to 1 if the user desires to disable receive ATM monitoring of tributary x.

The operation of the payload descrambler circuitry for tributary x is controlled by **RX_DSCR_INH_x**.

The **RX_DIRECT_MAP_x** and **RX_POS_x** registers determine the configuration of the generated signal.

Table 50. Receive Side Base Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PM Latch Count and Reset Control (Read/Write)									
2000	0x00			MASTER_RESET	SHORT_FRAME	Reserved	LATCH_CNT	RX_STATE_RESET	RX_PROV_RESET
2001		Unused							
FIFO Receive Side Summary Interrupt (Read Only)									
2002	0x00							RX_APS_INT	RX_SUM_INT
Receive Side Delta/Event Bits (Read / Cleared by uP)									
2003	0x08	LATCH_E				Reserved			
Receive Side Summary Status (Read Only)									
2004	0x00	TOH_SUMD_[1:8]							
2005	0x00	TOH_SUMD_[9:16]							
2006	0x00	TOH_SECE_SUM_[1:8]							
2007	0x00	TOH_SECE_SUM_[9:16]							
2008	0x00	PTR_INT_CONCAT_SUMD	PTR_INT_PAIS_SUMD	PTR_INT_LOP_SUMD	PATH_C2_SUMD	PATH_PLM_SUMD	PATH_UNEQ_SUMD	PATH_G1_SUMD	PATH_J1_SUMD
2009	0x00	PTR_PROC_CONCAT_SUMD	PTR_PROC_PAIS_SUMD	PTR_PROC_LOP_SUMD	RX_ATM_OCD_SUMD	RX_ATM_LCD_SUMD	RX_SW_STATUS_SUMD	RX_POS_ADRCTL_INVALID_SUMD	RX_FIFOERR_SUME
200A	0x00	PATH_B3_SF_SUMD	PATH_B3_SD_SUMD			PATH_B3ERR_SECE_SUM	PATH_G1ERR_SECE_SUM		RX_FIFOERR_SECE_SUM
200B	0x00	RX_ATM_HEC_DROP_SECE_SUM	RX_ATM_HEC_CORR_SECE_SUM	RX_ATM_OCD_SECE_SUM	RX_ATM_LCD_SECE_SUM	RX_POS_FCS_ERR_SECE_SUM	RX_POS_PMAX_ERR_SECE_SUM	RX_POS_PMIN_ERR_SECE_SUM	RX_POS_PABORT_ERR_SECE_SUM
200C			PTR_GEN_FIFO_SUME		PTR_GEN_JUSTCNT_SECE_SUM		APS_ERR_SUMD	APS_K1K2_SUMD	PTR_PROC_JUSTCNT_SECE_SUM
Receive Side Interrupt and Summary Status Masks (Read/Write)									
2010	0x03							RX_APS_INT_MASK	RX_SUM_INT_MASK
2011	0x88	LATCH_E_MASK				Reserved			

Table 50. Receive Side Base Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2012	0xFF	TOH_SUMD_MASK_[1:8]							
2013	0xFF	TOH_SUMD_MASK_[9:16]							
2014	0xFF	TOH_SECE_SUM_MASK_[1:8]							
2015	0xFF	TOH_SECE_SUM_MASK_[9:16]							
2016	0xFF	PTR_INT_CONCAT_SUMD_MASK	PTR_INT_PAIS_SUMD_MASK	PTR_INT_LOP_SUMD_MASK	PATH_C2_SUMD_MASK	PATH_PLM_SUMD_MASK	PATH_UNEQ_SUMD_MASK	PATH_G1_SUMD_MASK	PATH_J1_SUMD_MASK
2017	0xFF	PTR_PROC_CONCAT_SUMD_MASK	PTR_PROC_PAIS_SUMD_MASK	PTR_PROC_LOP_SUMD_MASK	RX_ATM_OCD_SUMD_MASK	RX_ATM_LCD_SUMD_MASK	RX_SW_STATUS_SUMD_MASK	RX_POS_ADRCTL_INVALID_SUMD_MASK	RX_FIFOERR_SUME_MASK
2018	0xCD	PATH_B3_SF_SUMD_MASK	PATH_B3_SD_SUMD_MASK			PATH_B3ERR_SECE_SUM_MASK	PATH_G1ERR_SECE_SUM_MASK		RX_FIFOERR_SECE_SUM_MASK
2019	0xFF	RX_ATM_HEC_DROP_SECE_SUM_MASK	RX_ATM_HEC_CORR_SECE_SUM_MASK	RX_ATM_OCD_SECE_SUM_MASK	RX_ATM_LCD_SECE_SUM_MASK	RX_POS_FCS_ERR_SECE_SUM_MASK	RX_POS_PMAX_ERR_SECE_SUM_MASK	RX_POS_PMIN_ERR_SECE_SUM_MASK	RX_POS_PABORT_ERR_SECE_SUM_MASK
201A	0x57		PTR_GEN_FIFO_SUME_MASK		PTR_GEN_JUSTCNT_SECE_SUM_MASK		APS_ERR_SUMD_MASK	APS_K1K2_SUMD_MASK	PTR_PROC_JUSTCNT_SECE_SUM_MASK
201B-201F	0x00	Unused							
Configuration, Loopback, Selector and PM Counter Latch Enable Provisioning (Read/Write)									
2020	0x00		CNT_SEC_EN		BIT_BLKCNT				
2021	0x00	Reserved							
2022	0x00	Reserved							SYS_SYNC_IN_RESYNC
2023		Unused							
2024	0x00	RX_PP_CONFIG_[7:0]							
2025	0x00	RX_PP_CONFIG_[15:8:]							
2026	0x00			PP_AUTO_CONFIG	RX_PP_CONFIG_[20:16]				
2027	0x00			PI_AUTO_CONFIG	RX_PI_CONFIG_[4:0]				
2028	0x40	LOF_INH	Reserved		RX_LINE_CONFIG_[4:0]				

Table 50. Receive Side Base Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2029	0x00		RX_REF_CLK_FREQ_[1:0]		RX_REF_CLK_SEL_[4:0]				
202A	0x00	RX_APS_SEL_[1:8]							
202B	0x00	RX_APS_SEL_[9:16]							
202C-202F	0x00	Unused							
Tributary Summary Delta/Events (Read Only)									
2030	0x00	RX_TRIB_[1:8]_SUMD							
2031	0x00	RX_TRIB_[9:16]_SUMD							
2032-20FF	0x00	Unused							
Tributary [1] Configuration Provisioning and Masks (Read/Write)									
2100	0x70	SONET_T_TO_R_LOOP_[1]	RX_POS_TRIB_[1]_MASK	RX_ATM_TRIB_[1]_MASK	RX_TRIB_[1]_MASK	RX_DSCR_INH_[1]	SYS_T_TO_R_LOOP_[1]		
2101	0x00					Reserved	RX_POS_[1]	RX_DIRECT_MAP_[1]	RX_TRIB_INH_[1]
2102-210F		Unused							
Tributary [2] Configuration Provisioning and Masks (Read/Write)									
2110	0x70	SONET_T_TO_R_LOOP_[2]	RX_POS_TRIB_[2]_MASK	RX_ATM_TRIB_[2]_MASK	RX_TRIB_[2]_MASK	RX_DSCR_INH_[2]	SYS_T_TO_R_LOOP_[2]		
2111	0x00					Reserved	RX_POS_[2]	RX_DIRECT_MAP_[2]	RX_TRIB_INH_[2]
2112-211F		Unused							
Tributary [3] Configuration Provisioning and Masks (Read/Write)									
2120	0x70	SONET_T_TO_R_LOOP_[3]	RX_POS_TRIB_[3]_MAFSK	RX_ATM_TRIB_[3]_MASK	RX_TRIB_[3]_MASK	RX_DSCR_INH_[3]	SYS_T_TO_R_LOOP_[3]		
2121	0x00					Reserved	RX_POS_[3]	RX_DIRECT_MAP_[3]	RX_TRIB_INH_[3]
2122-212F		Unused							
2130-21EF		::							
Tributary [16] Configuration Provisioning and Masks (Read/Write)									
21F0	0x70	SONET_T_TO_R_LOOP_[16]	RX_POS_TRIB_[16]_MASK	RX_ATM_TRIB_[16]_MASK	RX_TRIB_[16] MASK	RX_DSCR_INH_[16]	SYS_T_TO_R_LOOP_[16]		

Table 50. Receive Side Base Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
21F1	0x00					Reserved	RX_POS_[16]	RX_DIRECT_MAP_[16]	RX_TRIB_INH_[16]
21F2-21FF		Unused							

11.2.2 Receive Side Addresses 0x2200 through 0x25FF (Table)

11.2.2.1 TOH/SOH Deltas, Second Events, and Masks - Addr 0x2200 through 0x2206

The delta and second event bits contribute to the **TOH_SUMD_x** and **TOH_SECE_SUM_x** summary status bits in register 0x2004-0x2007. The contribution of any of the bits can be removed by setting the corresponding mask bit. On reset, the delta and second event bits are cleared, and the mask bits are set high.

11.2.2.2 TOH/SOH Provisioning - Addr 0x2208 through 0x220B

The **K2_CONSEC_x** register is used in monitoring the received K2 byte. (See section 6.6.3.) On reset this register is set to 0101 (decimal 5).

The **RX_LOF_ALG_x** bit determines which of 2 algorithms is used to clear the **RX_LOF_x** status bit. Descrambling is inhibited if **DSCRINH_x** is high. (See section 6.5.1.) On reset, these bits are cleared.

RX_LOS_RDI_INH_x controls the LOS contribution to LRDI and PAIS.

RX_LOSEXT_LEVEL_x controls the polarity of its external input signal, *RX_LOSEXT_x.* (See section 5.9.3.6.) **RX_LOSEXT_INH_x, RX_LOSEXT_DELAY_INH_x,** and **RX_LOS_ALL_ZERO_INH_x** control the contributors to the LOS monitor. (See section 6.4.2.)

RX_SDH_J0_x and **RX_SDH_S1_x** determine whether the Receive side performs J0 and S1 processing in SONET or SDH mode. (See section 6.6.)

RX_OOF/LOF/LOC/LAIS/LOS_ALARM_INH_x registers determine which alarms (OOF, LOF, LAIS, LOC or LOS) contribute to the *RX_ALARM_OUT_x* pins. (See section 6.6.7.)

11.2.2.3 TOH/SOH Status - Addr 0x220C through 0x2223

The bits in these registers hold the current state of the TOH/SOH Receiver Monitor. (See sections 6.5 through 6.6.) On reset, all bits in these registers are cleared, except for **RX_OOF_x**, **RX_LOF_x**, **B2_ERR_SF_x**, and **B2_ERR_SD_x**.

11.2.2.4 B2 Signal Fail and Signal Degrade Parameters - Addr 0x2224 through 0x222F

These registers contain the algorithm parameters for the B2 based Signal Fail and Signal Degrade Monitors. (See section 6.6.2.) On reset, all of the parameter values are set to decimal 1.

11.2.2.5 B1, B2, and M1 Error Counters - Addr 0x2230 through 0x223A

These registers contain the latched results of error counters. (See section 6.5.2, section 6.6.2, and section 6.6.6). On reset, these registers are cleared.

The SONET/SDH TOH/SOH Monitoring register map for SONET/SDH signal [1] is illustrated in Table . Identical maps exist for Signals [2] through [16]; these are located starting at addresses 0x2240, 0x2280 through 0x25C0, respectively.

Table 51. Receive TOH Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Signal 1 TOH/SOH Delta and Second Event Bits (Read / cleared by uP)									
2200	0x00	B2_ERR_SF_[1]_D	B2_ERR_SD_[1]_D	RX_LAIS_[1]_D	RX_LRDI_[1]_D	RX_K1_[1]_D	K1_UNSTAB_[1]_D	RX_K2_[1]_D	RX_S1_[1]_D
2201	0x00		RX_LOSEXT_[1]_D	RX_LOS_[1]_D	RX_LOC_[1]_D	RX_OOF_[1]_D	RX_LOF_[1]_D	J0_OOF_[1]_D	RX_J0_[1]_D
2202	0x00			RX_OOF_[1]_SECE	RX_LOF_[1]_SECE	B1ERR_[1]_SECE	B2ERR_[1]_SECE	S1FAIL_[1]_SECE	M1ERR_[1]_SECE
2203		Unused							
TOH/SOH Masks and Provisioning Bits (Read/Write)									
2204	0xFF	B2_ERR_SF_[1]_D_MASK	B2_ERR_SD_[1]_D_MASK	RX_LAIS_[1]_D_MASK	RX_LRDI_[1]_D_MASK	RX_K1_[1]_D_MASK	K1_UNSTAB_[1]_D_MASK	RX_K2_[1]_D_MASK	RX_S1_[1]_D_MASK
2205	0x7F		RX_LOSEXT_[1]_D_MASK	RX_LOS_[1]_D_MASK	RX_LOC_[1]_D_MASK	RX_OOF_[1]_D_MASK	RX_LOF_[1]_D_MASK	J0_OOF_[1]_D_MASK	RX_J0_[1]_D_MASK
2206	0x3F			RX_OOF_[1]_SECE_MASK	RX_LOF_[1]_SECE_MASK	B1ERR_[1]_SECE_MASK	B2ERR_[1]_SECE_MASK	S1FAIL_[1]_SECE_MASK	M1ERR_[1]_SECE_MASK
2207		Unused							
2208	0x50	K2_CONSEC_[1]_[3:0]						RX_LOF_ALG_[1]	DSCRINH_[1]
2209	0x88	RX_LOSEXT_INH_[1]	RX_LOSEXT_LEVEL_[1]	RX_LOS_RDI_INH_[1]	Reserved	RX_LOSEXT_DELAY_INH_[1]	RX_LOS_ALL_ZERO_INH_[1]	RX_SDH_S1_[1]	RX_SDH_J0_[1]
220A	0x00						Reserved		
220B	0x00				RX_LOS_ALARM_INH_[1]	RX_LAIS_ALARM_INH_[1]	RX_LOC_ALARM_INH_[1]	RX_LOF_ALARM_INH_[1]	RX_OOF_ALARM_INH_[1]
TOH/SOH Status (Read Only)									
220C	0x00	B2_ERR_SF_[1]	B2_ERR_SD_[1]	RX_LAIS_[1]	RX_LRDI_[1]		K1_UNSTAB_[1]		
220D	0x00		RX_LOSEXT_[1]	RX_LOS_[1]	RX_LOC_[1]	RX_OOF_[1]	RX_LOF_[1]	J0_OOF_[1]	
220E-220F		Unused							
2210	0x00	RX_J0_[1]_[0]_[7:0]							
2211	0x00	RX_J0_[1]_[1]_[7:0]							

Table 51. Receive TOH Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2212	0x00	RX_J0_[1]_[2]_[7:0]							
2213 to 221D	0x00	: : :							
221E	0x00	RX_J0_[1]_[14]_[7:0]							
221F	0x00	RX_J0_[1]_[15]_[7:0]							
2220		Unused							
2221	0x00					RX_S1_[1]_[3:0]			
2222	0x00	RX_K2_[1]_[7:0]							
2223	0x00	RX_K1_[1]_[7:0]							
Signal Fail & Signal Degrade Parameters (Read/Write)									
2224	0x01	B2_BLOCK_SF_[1]_[7:0]							
2225		Unused							
2226	0x01	B2_THRESH_SET_SF_[1]_[7:0]							
2227	0x01			B2_GROUP_SET_SF_[1]_[5:0]					
2228	0x01	B2_THRESH_CLR_SF_[1]_[7:0]							
2229	0x01			B2_GROUP_CLR_SF_[1]_[5:0]					
222A	0x01	B2_BLOCK_SD_[1]_[7:0]							
222B	0x00	B2_BLOCK_SD_[1]_[15:8]							
222C	0x01			B2_THRESH_SET_SD_[1]_[5:0]					
222D	0x01			B2_GROUP_SET_SD_[1]_[5:0]					
222E	0x01			B2_THRESH_CLR_SD_[1]_[5:0]					
222F	0x01			B2_GROUP_CLR_SD_[1]_[5:0]					
Performance-Monitoring Counters (Read Only)									
2230	0x00	B1_ERRCNT_[1]_[7:0]							
2231	0x00	B1_ERRCNT_[1]_[15:8]							
2232-2233		Unused							
2234	0x00	B2_ERRCNT_[1]_[7:0]							
2235	0x00	B2_ERRCNT_[1]_[15:8]							
2236	0x00			B2_ERRCNT_[1]_[21:16]					
2237		Unused							
2238	0x00	M1_ERRCNT_[1]_[7:0]							
2239	0x00	M1_ERRCNT_[1]_[15:8]							
223A	0x00			M1_ERRCNT_[1]_[21:16]					

Table 51. Receive TOH Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
223B-223F		Unused							
Signal 2 SONET/SDH TOH/SOH Monitoring Map									
2240-227F		Same as 0x2200 through 0x223F							
Signal 3 SONET/SDH TOH/SOH Monitoring Map									
2280-22BF		Same as 0x2200 through 0x223F							
Signal 4 SONET/SDH TOH/SOH Monitoring Map									
22C0-22FF		Same as 0x2200 through 0x223F							
Signal 5 SONET/SDH TOH/SOH Monitoring Map									
2300-233F		Same as 0x2200 through 0x223F							
Signal 6 SONET/SDH TOH/SOH Monitoring Map									
2340-237F		Same as 0x2200 through 0x223F							
Signal 7 SONET/SDH TOH/SOH Monitoring Map									
2380-23BF		Same as 0x2200 through 0x223F							
Signal 8 SONET/SDH TOH/SOH Monitoring Map									
23C0-23FF		Same as 0x2200 through 0x223F							
Signal 9 SONET/SDH TOH/SOH Monitoring Map									
2400-243F		Same as 0x2200 through 0x223F							
Signal 10 SONET/SDH TOH/SOH Monitoring Map									
2440-247F		Same as 0x2200 through 0x223F							
Signal 11 SONET/SDH TOH/SOH Monitoring Map									
2480-24BF		Same as 0x2200 through 0x223F							
Signal 12 SONET/SDH TOH/SOH Monitoring Map									
24C0-24FF		Same as 0x2200 through 0x223F							
Signal 13 SONET/SDH TOH/SOH Monitoring Map									
2500-253F		Same as 0x2200 through 0x223F							
Signal 14 SONET/SDH TOH/SOH Monitoring Map									
2540-257F		Same as 0x2200 through 0x223F							
Signal 15 SONET/SDH TOH/SOH Monitoring Map									

Table 51. Receive TOH Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2580-25BF		Same as 0x2200 through 0x223F							
Signal 16 SONET/SDH TOH/SOH Monitoring Map									
25C0-25FF		Same as 0x2200 through 0x223F							

11.2.3 Receive Side Addresses 0x2600 through 0x27FF (Table 52)

11.2.3.1 Pointer Interpreter Provisioning - Addr 0x2602 and 0x2606

The **RX_PP_CONCAT_[20:0]_D** bits are set to 1 whenever there is a change of their associated status bits. The **RX_PP_CONCAT_[20:0]_D_MASK** bits are set to 1 to disable the contribution of the associated delta bit to the **PTR_PROC_CONCAT_SUMD** interrupt.

11.2.3.2 Pointer Interpreter Status - Addr 0x2608 to 0x260A

The **RX_PP_CONCAT_[20:0]** bits report the received signal configuration as indicated by the H1H2 bytes. (See section 6.9.3.)

11.2.3.3 Pointer Interpreter Deltas - Addr [0x2700,0x2702]

The delta bits contribute to the **PTR_PROC_PAIS_SUMD** and **PTR_PROC_LOP_SUMD** summary status bits in receive register 0x1009. (See section 6.9.1.) On reset, the delta bits are cleared.

11.2.3.4 Pointer Interpreter Provisioning - Addr [0x2704,0x27F4]

The **RX_PP_SS_EN_x** bits determine whether or not the SS bits are used in the Pointer Interpreter algorithms. (See sections 6.9.1 and 6.10.) **RX_SDH_PP_x** determine whether the Receive side performs SONET or SDH pointer interpretation. (See section 6.10.)

11.2.3.5 Pointer Interpreter Status - Addr [0x2700 - 0x27F8]

These registers hold the current state of the Pointer Interpreter. (See section 6.9.1.) On reset, **RX_PP_LOP_x**, **RX_PP_PAIS_x**, and **PP_PTR_STATE_[x]_[j]_[1:0]** for x = 1 to 16 and j = 1 to 3 are cleared indicating normal STS-1/AU-3 pointers.

Table 52. RX Pointer Processor Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Pointer Interpreter Delta Bits (Read / cleared by uP)									
2600	0x00	RX_PP_CONCAT_[7:0]_D							
2601	0x00	RX_PP_CONCAT_[15:0]_D							
2602	0x00				RX_PP_CONCAT_[20:16]_D				
2603		Unused							
Pointer Interpreter Mask Bits (Read/Write)									
2604	0xFF	RX_PP_CONCAT_[7:0]_D_MASK							
2605	0xFF	RX_PP_CONCAT_[15:0]_D_MASK							

Table 52. RX Pointer Processor Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2606	0x1F				RX_PP_CONCAT_[20:16]_D_MASK				
2607		Unused							
Pointer Interpreter Status Bits (Read Only)									
2608	0x00	RX_PP_CONCAT_[7:0]							
2609	0x00	RX_PP_CONCAT_[15:8]							
260A	0x00				RX_PP_CONCAT_[20:16]				
260B-26FF		Unused							
Tributary [1] Pointer Interpreter Delta Bits (Read / cleared by uP)									
2700	0x00			RX_PP_LOP_[1]_[1]_D	RX_PP_PAIS_[1]_[1]_D	RX_PP_LOP_[1]_[2]_D	RX_PP_PAIS_[1]_[2]_D	RX_PP_LOP_[1]_[3]_D	RX_PP_PAIS_[1]_[3]_D
2701	0x00			RX_PP_NEGCNT_SECE_[1]_[1]	RX_PP_POSCNT_SECE_[1]_[1]	RX_PP_NEGCNT_SECE_[1]_[2]	RX_PP_POSCNT_SECE_[1]_[2]	RX_PP_NEGCNT_SECE_[1]_[3]	RX_PP_POSCNT_SECE_[1]_[3]
2702-2703		Unused							
Pointer Interpreter Configuration Bits (Read/Write)									
2704	0x00		RX_SDH_PP_[1]	RX_PP_SS_EN_[1]					
2705		Unused							
Tributary [1] Pointer Interpreter Status Bits (Read Only)									
2706	0x00			PP_PTR_STATE_[1]_[1]_[1:0]		PP_PTR_STATE_[1]_[2]_[1:0]		PP_PTR_STATE_[1]_[3]_[1:0]	
2707	0x00			RX_PP_LOP_[1]_[1]	RX_PP_PAIS_[1]_[1]	RX_PP_LOP_[1]_[2]	RX_PP_PAIS_[1]_[2]	RX_PP_LOP_[1]_[3]	RX_PP_PAIS_[1]_[3]
Tributary [1] Pointer Interpreter Counter Bits (Read Only)									
2708	0x00				RX_PP_POSCNT_[1]_[1]_[4:0]				
2709	0x00				RX_PP_NEGCNT_[1]_[1]_[4:0]				
270A	0x00				RX_PP_POSCNT_[1]_[2]_[4:0]				
270B	0x00				RX_PP_NEGCNT_[1]_[2]_[4:0]				
270C	0x00				RX_PP_POSCNT_[1]_[3]_[4:0]				
270D	0x00				RX_PP_NEGCNT_[1]_[3]_[4:0]				
270E-270F		Unused							
Tributary [2] Pointer Interpreter Delta Bits (Read / cleared by uP)									

Table 52. RX Pointer Processor Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2710	0x00			RX_PP_LOP_[2]_[1]_D	RX_PP_PAIS_[2]_[1]_D	RX_PP_LOP_[2]_[2]_D	RX_PP_PAIS_[2]_[2]_D	RX_PP_LOP_[2]_[3]_D	RX_PP_PAIS_[2]_[3]_D
2711	0x00			RX_PP_NEGCNT_SECE_[2]_[1]	RX_PP_POSCNT_SECE_[2]_[1]	RX_PP_NEGCNT_SECE_[2]_[2]	RX_PP_POSCNT_SECE_[2]_[2]	RX_PP_NEGCNT_SECE_[2]_[3]	RX_PP_POSCNT_SECE_[2]_[3]
2712-2713		Unused							
Pointer Interpreter Configuration Bits (Read/Write)									
2714	0x00		RX_SDH_PP_[2]	RX_PP_SS_EN_[2]					
2715		Unused							
Tributary [2] Pointer Interpreter Status Bits (Read Only)									
2716	0x00			PP_PTR_STATE_[2]_[1]_[1:0]		PP_PTR_STATE_[2]_[2]_[1:0]		PP_PTR_STATE_[2]_[3]_[1:0]	
2717	0x00			RX_PP_LOP_[2]_[1]	RX_PP_PAIS_[2]_[1]	RX_PP_LOP_[2]_[2]	RX_PP_PAIS_[2]_[2]	RX_PP_LOP_[2]_[3]	RX_PP_PAIS_[2]_[3]
Tributary [2] Pointer Interpreter Counter Bits (Read Only)									
2718	0x00				RX_PP_POSCNT_[2]_[1]_[4:0]				
2719	0x00				RX_PP_NEGCNT_[2]_[1]_[4:0]				
271A	0x00				RX_PP_POSCNT_[2]_[2]_[4:0]				
271B	0x00				RX_PP_NEGCNT_[2]_[2]_[4:0]				
271C	0x00				RX_PP_POSCNT_[2]_[3]_[4:0]				
271D	0x00				RX_PP_NEGCNT_[2]_[3]_[4:0]				
271E-271F		Unused							
2720-27EF	. . .								
Tributary [16] Pointer Interpreter Delta Bits (Read / cleared by uP)									
27F0	0x00			RX_PP_LOP_[16]_[1]_D	RX_PP_PAIS_[16]_[1]_D	RX_PP_LOP_[16]_[2]_D	RX_PP_PAIS_[16]_[2]_D	RX_PP_LOP_[16]_[3]_D	RX_PP_PAIS_[16]_[3]_D
27F1	0x00			RX_PP_NEGCNT_SECE_[16]_[1]	RX_PP_POSCNT_SECE_[16]_[1]	RX_PP_NEGCNT_SECE_[16]_[2]	RX_PP_POSCNT_SECE_[16]_[2]	RX_PP_NEGCNT_SECE_[16]_[3]	RX_PP_POSCNT_SECE_[16]_[3]

Table 52. RX Pointer Processor Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
27F2-27F3		Unused							
Pointer Interpreter Configuration Bits (Read/Write)									
27F4	0x00		RX_SDH_PP_[16]	RX_PP_SS_EN_[16]					
27F5		Unused							
Tributary [16] Pointer Interpreter Status Bits (Read Only)									
27F6	0x00			PP_PTR_STATE_[16]_[1]_[1:0]		PP_PTR_STATE_[16]_[2]_[1:0]		PP_PTR_STATE_[16]_[3]_[1:0]	
27F7	0x00			RX_PP_LOP_[16]_[1]	RX_PP_PAIS_[16]_[1]	RX_PP_LOP_[16]_[2]	RX_PP_PAIS_[16]_[2]	RX_PP_LOP_[16]_[3]	RX_PP_PAIS_[16]_[3]
Tributary [16] Pointer Interpreter Counter Bits (Read Only)									
27F8	0x00				RX_PP_POSCNT_[16]_[1]_[4:0]				
27F9	0x00				RX_PP_NEGCNT_[16]_[1]_[4:0]				
27FA	0x00				RX_PP_POSCNT_[16]_[2]_[4:0]				
27FB	0x00				RX_PP_NEGCNT_[16]_[2]_[4:0]				
27FC	0x00				RX_PP_POSCNT_[16]_[3]_[4:0]				
27FD	0x00				RX_PP_NEGCNT_[16]_[3]_[4:0]				
27FE-27FF	0x00	Unused							

* Designates those registers that use only a single control bit, the bit associated with tributary x = 1. This bit controls the corresponding function for all tributaries.

** Reset values for register 0x2706 is 0x0C. Remainder of 0x27n6 registers reset to 0x3C. This results in a reset state of 1 STS-48c/AU-4-16c pointer.

11.2.4 Receive Side Addresses 0x2800 through 0x29FF (Table)

11.2.4.1 Pointer Generator Event Bits - Addr [0x2900,0x29F0] and [0x2902,0x29F2]

If for tributary x, there has been at least 1 positive or negative justification in the previous second, the **RX_PG_POSCNT_x_SECE** or **RX_PG_NEGCNT_x_SECE** bit is set.

If a RX Pointer Generator FIFO overflows or underflows, the **RX_PG_FIFO_[x]_[j]_E** event bit is set. The FIFO is then recentered, and the Pointer Generator continues normal operation.

11.2.4.2 Pointer Generator Provisioning - Addr [0x2904-5,0x29F4-5]

If **RX_FAST_AIS_x** = 1 and the last frame received for tributary **x** contains all 1s in its H-bytes, RHINE inserts all ones in the pointer bytes for tributary **x**. **RX_SF_PAIS_INH_x** controls the SF contribution to PAIS insertion. **RX_LOS_RDI_INH_x** controls the LOS contribution to PAIS insertion.**RX_SDH_PG_x** is used to select between SDH (=1) and SONET (=0) modes for pointer generation.

The user can force PAIS generation for tributary **x** by setting **RX_PAIS_GEN_x** = 1 (if tributary **x** is in operation based on the value of **RX_PP_CONCAT** or **RX_PP_CONFIG**. (See section 6.9.3.)

The Pointer Generators can also insert Unequipped. If **RX_PAIS_GEN_x** = 0 and **RX_UNEQ_GEN_x** = 1, the entire SPE/VC is generated with all 0s. The pointer value used for unequipped insertion must be a valid pointer value; the specific value is not specified. An NDF does not need to be generated when the unequipped signal insertion is removed (**RX_UNEQ_GEN_x** is cleared).

11.2.4.3 Pointer Generation Justification Counters - Addr [0x2908-D, 0x29F8-D]

The RX Pointer Generator contains 5-bit pointer generator justification counters that count every positive or negative justification. When the performance-monitoring counters are latched, the values of these counters are latched to the **RX_PG_POSCNT_x_[4:0]** and **RX_PG_NEGCNT_x_[4:0]** registers, and the justification counters are cleared.

Table 53. Pointer Generator Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2800-28FF		Unused							
Tributary [1] Pointer Generator Event Bits (Read / cleared by uP)									
2900	0x00			RX_PG_NEGCNT_SECE_[1]_[1]	RX_PG_POSCNT_SECE_[1]_[1]	RX_PG_NEGCNT_SECE_[1]_[2]	RX_PG_POSCNT_SECE_[1]_[2]	RX_PG_NEGCNT_SECE_[1]_[3]	RX_PG_POSCNT_SECE_[1]_[3]
2901		Unused							
2902	0x00				RX_PG_FIFO_[1]_[1]_E		RX_PG_FIFO_[1]_[2]_E		RX_PG_FIFO_[1]_[3]_E
2903		Unused							
Tributary [1] Pointer Generator Configuration Bits (Read/Write)									
2904	0x00	Reserved	RX_SF_PAIS_INH_[1]	RX_UNEQ_GEN_[1]_[1]	RX_PAIS_GEN_[1]_[1]	RX_UNEQ_GEN_[1]_[2]	RX_PAIS_GEN_[1]_[2]	RX_UNEQ_GEN_[1]_[3]	RX_PAIS_GEN_[1]_[3]
2905	0x00		RX_SDH_PG_[1]	RX_FAST_AIS_[1]_[1]		RX_FAST_AIS_[1]_[2]		RX_FAST_AIS_[1]_[3]	
2906-2907		Unused							
Tributary [1] Pointer Generator Counter Bits (Read Only)									
2908	0x00				RX_PG_POSCNT_[1]_[1]_[4:0]				
2909	0x00				RX_PG_NEGCNT_[1]_[1]_[4:0]				
290A	0x00				RX_PG_POSCNT_[1]_[2]_[4:0]				
290B	0x00				RX_PG_NEGCNT_[1]_[2]_[4:0]				
290C	0x00				RX_PG_POSCNT_[1]_[3]_[4:0]				
290D	0x00				RX_PG_NEGCNT_[1]_[3]_[4:0]				
290E-290F		Unused							
Tributary [2] Pointer Generator Event Bits (Read / cleared by uP)									

Table 53. Pointer Generator Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2910	0x00			RX_PG_NEGCNT_SECE_[2]_[1]	RX_PG_POSCNT_SECE_[2]_[1]	RX_PG_NEGCNT_SECE_[2]_[2]	RX_PG_POSCNT_SECE_[2]_[2]	RX_PG_NEGCNT_SECE_[2]_[3]	RX_PG_POSCNT_SECE_[2]_[3]
2911		Unused							
2912	0x00				RX_PG_FIFO_[2]_[1]_E		RX_PG_FIFO_[2]_[2]_E		RX_PG_FIFO_[2]_[3]_E
2913		Unused							
Tributary [2] Pointer Generator Configuration Bits (Read/Write)									
2914	0x00	Reserved	RX_SF_PAIS_INH_[2]	RX_UNEQ_GEN_[2]_[1]	RX_PAIS_GEN_[2]_[1]	RX_UNEQ_GEN_[2]_[2]	RX_PAIS_GEN_[2]_[2]	RX_UNEQ_GEN_[2]_[3]	RX_PAIS_GEN_[2]_[3]
2915	0x00		RX_SDH_PG_[2]	RX_FAST_AIS_[2]_[1]		RX_FAST_AIS_[2]_[2]		RX_FAST_AIS_[2]_[3]	
2916-2917		Unused							
Tributary [2] Pointer Generator Counter Bits (Read Only)									
2918	0x00				RX_PG_POSCNT_[2]_[1]_[4:0]				
2919	0x00				RX_PG_NEGCNT_[2]_[1]_[4:0]				
291A	0x00				RX_PG_POSCNT_[2]_[2]_[4:0]				
291B	0x00				RX_PG_NEGCNT_[2]_[2]_[4:0]				
291C	0x00				RX_PG_POSCNT_[2]_[3]_[4:0]				
291D	0x00				RX_PG_NEGCNT_[2]_[3]_[4:0]				
291E-291F		Unused							
2920-29EF	0x00	. . .							
Tributary [16] Pointer Generator Event Bits (Read / cleared by uP)									
29F0	0x00			RX_PG_NEGCNT_SECE_[16]_[1]	RX_PG_POSCNT_SECE_[16]_[1]	RX_PG_NEGCNT_SECE_[16]_[2]	RX_PG_POSCNT_SECE_[16]_[2]	RX_PG_NEGCNT_SECE_[16]_[3]	RX_PG_POSCNT_SECE_[16]_[3]
29F1		Unused							
29F2	0x00				RX_PG_FIFO_[16]_[1]_E		RX_PG_FIFO_[16]_[2]_E		RX_PG_FIFO_[16]_[3]_E
29F3		Unused							
Tributary [16] Pointer Generator Configuration Bits (Read/Write)									
29F4	0x00	Reserved	RX_SF_PAIS_INH_[16]	RX_UNEQ_GEN_[16]_[1]	RX_PAIS_GEN_[16]_[1]	RX_UNEQ_GEN_[16]_[2]	RX_PAIS_GEN_[16]_[2]	RX_UNEQ_GEN_[16]_[3]	RX_PAIS_GEN_[16]_[3]

Table 53. Pointer Generator Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
29F5	0x00		RX_SDH_PG_ [16]	RX_FAST_AIS_[16]_[1]		RX_FAST_AIS_[16]_[2]		RX_FAST_AIS_[16]_[3]	
29F6-29F7		Unused							
Tributary [16] Pointer Generator Counter Bits (Read Only)									
29F8	0x00				RX_PG_POSCNT_[16]_[1]_[4:0]				
29F9	0x00				RX_PG_NEGCNT_[16]_[1]_[4:0]				
29FA	0x00				RX_PG_POSCNT_[16]_[2]_[4:0]				
29FB	0x00				RX_PG_NEGCNT_[16]_[2]_[4:0]				
29FC	0x00				RX_PG_POSCNT_[16]_[3]_[4:0]				
29FD	0x00				RX_PG_NEGCNT_[16]_[3]_[4:0]				
29FE-29FF		Unused							

* Designates those registers that use only a single control bit, the bit associated with tributary x = 1. This bit controls the corresponding function for all tributaries.

11.2.5 Receive Side Addresses 0x2A00 through 0x2BFF (Table)

11.2.5.1 Pointer Interpreter Provisioning - Addr 0x2A02 and 0x2A04

The **RX_PI_CONCAT_[4:0]_D** bits are set to 1 whenever there is a change of their associated status bits. The **RX_PI_CONCAT_[4:0]_D_MASK** bits are set to 1 to disable the contribution of the associated delta bit to the **PTR_INT_CONCAT_SUMD** interrupt.

11.2.5.2 Pointer Interpreter Provisioning - Addr 0x2A08

The **RX_PI_CONCAT_[4:0]** bits report the received signal configuration as indicated by the H1H2 bytes. (See section 6.9.)

11.2.5.3 Pointer Interpreter Deltas - Addr [0x2B00,0x2BF0]

The delta bits contribute to the **PTR_INT_PAIS_SUMD** and **PTR_INT_LOP_SUMD** summary status bits in transmit register 0x1008. (See section 6.9.1.) On reset, the delta bits are cleared.

11.2.5.4 Pointer Interpreter Provisioning - Addr [0x2B02, 0x2BF2]

The **RX_PG_SS_EN_x** bits determine whether or not the SS bits are used in the Pointer Interpreter algorithms. (See sections 6.9.1 and 6.10.) **RX_SDH_PG_x** determine whether the Receive side performs SONET or SDH pointer interpretation. (See section 6.10.)

11.2.5.5 Pointer Interpreter Status - Addr [0x2B04-8,0x2BF4-8]

These registers hold the current state of the Pointer Interpreter. (See section 6.9.1.) On reset, **RX_PI_LOP_[x]_[j]**, **RX_PI_PAIS_[x]_[j]**, both bits of **PI_PTR_STATE_[1]_[1]_[1:0]** are cleared; all bits of the 47 remaining **PI_PTR_STATE_[i]_[j]_[1:0]** are set, indicating an STS-48c or AU-4-16c normal pointer.

Table 54. Pointer Interpreter Provisioning and Monitoring Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Pointer Interpreter Delta Bits (Read / cleared by uP)									
2A00-2A01		Unused							
2A02	0x00				RX_PI_CONCAT_[4:0]_D				
Pointer Interpreter Mask Bits (Read / Write)									
2A03		Unused							
2A04	0x1F				RX_PI_CONCAT_[4:0]_D_MASK				
2A05-7		Unused							
Pointer Interpreter Status Bits (Ready Only)									
2A08	0x00				RX_PI_CONCAT_[4:0]				
2A09-2AFF		Unused							
Tributary [1] Pointer Interpreter Delta Bits (Read / cleared by uP)									
2B00	0x01							RX_PI_LOP_[1]_D	RX_PI_PAIS_[1]_D
2B01		Unused							
2B02	0x00		RX_SDH_PI_[1]	RX_PI_SS_EN_[1]					
2B03		Unused							
Tributary [1] Pointer Interpreter Status Bits (Read Only)									
2B04	0x01							RX_PI_LOP_[1]	RX_PI_PAIS_[1]
2B05-2B07	0x00	Unused							
2B08	0x15			PI_PTR_STATE_[1]_[1]_[1:0]		PI_PTR_STATE_[1]_[2]_[1:0]		PI_PTR_STATE_[1]_[3]_[1:0]	
2B09-2B0F		Unused							
Tributary [2] Pointer Interpreter Delta Bits (Read / cleared by uP)									
2B10	0x01							RX_PI_LOP_[2]_D	RX_PI_PAIS_[2]_D
2B11		Unused							
2B12	0x00		RX_SDH_PI_[2]	RX_PI_SS_EN_[2]					
2B13		Unused							
Tributary [2] Pointer Interpreter Status Bits (Read Only)									
2B14	0x01							RX_PI_LOP_[2]	RX_PI_PAIS_[2]

Table 54. Pointer Interpreter Provisioning and Monitoring Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2B15-2B17	0x00	Unused							
2B18	0x15			PI_PTR_STATE_[2]_[1]_[1:0]		PI_PTR_STATE_[2]_[2]_[1:0]		PI_PTR_STATE_[2]_[3]_[1:0]	
2B19-2B1F		Unused							
Tributary [3] Pointer Interpreter Delta Bits (Read / cleared by uP)									
2B20	0x01							RX_PI_LOP _[3]_D	RX_PI_PAIS _[3]_D
2B21		Unused							
2B22	0x00		RX_SDH_ PI_[3]	RX_PI_SS_ EN_[3]					
2B23		Unused							
Tributary [3] Pointer Interpreter Status Bits (Read Only)									
2B24	0x01							RX_PI_LOP _[3]	RX_PI_PAIS _[3]
2B25-2B27	0x00	Unused							
2B28	0x15			PI_PTR_STATE_[3]_[1]_[1:0]		PI_PTR_STATE_[3]_[2]_[1:0]		PI_PTR_STATE_[3]_[3]_[1:0]	
2B29-2B2F		Unused							
2B30-2BEF		: :							
Tributary [16] Pointer Interpreter Delta Bits (Read / cleared by uP)									
2BF0	0x01							RX_PI_LOP _[16]_D	RX_PI_PAIS _[16]_D
2BF1	0x00	Unused							
2BF2	0x00		RX_SDH_ PI_[16]	RX_PI_SS_ EN_[16]					
2BF3		Unused							
Tributary [16] Pointer Interpreter Status Bits (Read Only)									
2BF4	0x01							RX_PI_LOP _[16]	RX_PI_PAIS _[16]
2BF5-2BF7	0x00	Unused							
2BF8	0x15			PI_PTR_STATE_[16]_[1]_[1:0]		PI_PTR_STATE_[16]_[2]_[1:0]		PI_PTR_STATE_[16]_[3]_[1:0]	
2BF9-2BFF		Unused							

11.2.6 Receive Side Addresses 0x2C00 through 0x2DFF (Table)

11.2.6.1 Path Provisioning - Addr 0x2C00 and 0x2D01

These bits affect the operation of J1, C2, and G1 path monitoring. (See sections 6.11.1 and 6.11.4.) On reset, these bits are cleared, except for the **G1_CONSEC** register, which is set to 0101 (decimal 5), and the **EXP_C2_x** registers that are set to 01 (hex).

11.2.6.2 J1 Path Monitor Delta and Event Bits - Addr 0x2C02

These delta and event bits contribute to the summary status bits in register 0x2008. (See section 6.11.) On reset, the delta and event bits are cleared.

11.2.6.3 J1 Monitoring - Addr 0x2C05 through 0x2C45

The **RX_J1_[63:0]_[7:0]** registers hold the last captured, path-trace frame of the specified tributary, in the SONET mode. In the SDH mode, the last accepted, path-trace frame of the specified tributary is held in the **RX_J1_[15:0]_[7:0]** registers. (See section 6.11.1.) On reset, these registers are cleared.

11.2.6.4 Path Monitor Delta and Second Event Bits - Addr [0x2D00,0x2DF0]

These delta and second event bits contribute to summary status bits in receive registers 0x2008 and 0x200A. (See section 6.11.) On reset, the delta and second event bits are cleared.

11.2.6.5 Path Provisioning - Addr [0x2D02,0x2DF2]

The **EXP_C2_x** registers, which are set to 01 (hex), holds the expected C2 values. (See section 5.5.2.3.)

The **RX_TIM_x** registers, reset value 0, should be written to 1 by the user whenever a Trace Identifier Mismatch (TIM) is detected for tributary x. The status of this bit is incorporated into the enhanced RDI-P message. (See section 5.5.2.4.)

11.2.6.6 Path Status - Addr [0x2D04-5,0x2DF4-5]

The bits in these registers hold the current state of the POH Receiver Monitors. (See sections 6.11.4 and 6.11.1.) On reset, **RX_C2_x** are set to 0x01 and all other bits in these registers are cleared.

11.2.6.7 B3 and G1 Error Counters - Addr [0x2D06-9,0x2DF6-9]

These registers contain the latched results of B3 and G1 error counters. (See section 4.2, section 6.11.2, and section 6.11.4.1.) On reset, these registers are cleared.

Table 55. Receive POH Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Path Provisioning (Read/Write)									
2C00	0x00	J1_CRLF	J1_CAP	Reserved	RX_SDH_ J1	J1_CAP_TRIB_[3:0]			
2C01	0x50	G1_CONSEC_[3:0]							RX_PRDI5
Path J1 Monitoring Event and Delta Bits (Read / cleared by uP)									
2C02	0x00						J1_CAP_E	RX_J1_D	J1_OOF_D
2C03-2C04		Unused							
J1 Monitoring (Read Only)									
2C05	0x01								J1_OOF
2C06	0x00	RX_J1_[0]_[7:0]							
2C07	0x00	RX_J1_[1]_[7:0]							

Table 55. Receive POH Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2C08 to 2C43	0x00	: : :							
2C44	0x00	RX_J1_[62]_[7:0]							
2C45	0x00	RX_J1_[63]_[7:0]							
2C46-2CFF		Unused							
Tributary [1] Path Delta and Second Event Bits (Read / cleared by uP)									
2D00	0x00	G1ERR_[1]_SECE	B3ERR_[1]_SECE		RX_C2_[1]_D	RX_G1_[1]_D	RX_UNEQ_[1]_D	RX_PLM_[1]_D	
2D01		Unused							
Tributary [1] Path Provisioning (Read/Write)									
2D02	0x01	EXP_C2_[1]_[7:0]							
2D03	0x00								RX_TIM_[1]
Tributary [1] Path Status (Read Only)									
2D04	0x00			RX_G1_[1]_[2:0]			RX_UNEQ_[1]	RX_PLM_[1]	
2D05	0x01	RX_C2_[1]_[7:0]							
2D06	0x00	B3_ERRCNT_[1]_[7:0]							
2D07	0x00	B3_ERRCNT_[1]_[15:8]							
2D08	0x00	G1_ERRCNT_[1]_[7:0]							
2D09	0x00	G1_ERRCNT_[1]_[15:8]							
2D0A-2D0F		Unused							
Tributary [2] Path Delta and Second Event Bits (Read / cleared by uP)									
2D10	0x00	G1ERR_[2]_SECE	B3ERR_[2]_SECE		RX_C2_[2]_D	RX_G1_[2]_D	RX_UNEQ_[2]_D	RX_PLM_[2]_D	
2D11		Unused							
Tributary [2] Path Provisioning (Read/Write)									
2D12	0x01	EXP_C2_[2]_[7:0]							
2D13	0x00								RX_TIM_[2]
Tributary [2] Path Status (Read Only)									
2D14	0x00			RX_G1_[2]_[2:0]			RX_UNEQ_[2]	RX_PLM_[2]	
2D15	0x01	RX_C2_[2]_[7:0]							
2D2D	0x00	B3_ERRCNT_[2]_[7:0]							
2D17	0x00	B3_ERRCNT_[2]_[15:8]							
2D18	0x00	G1_ERRCNT_[2]_[7:0]							

Table 55. Receive POH Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2D19	0x00	G1_ERRCNT_[2]_[15:8]							
2D1A-2D1F		Unused							
Tributary [3] Path Delta and Second Event Bits (Read / cleared by uP)									
2D20	0x00	G1ERR_[3]_SECE	B3ERR_[3]_SECE		RX_C2_[3]_D	RX_G1_[3]_D	RX_UNEQ_[3]_D	RX_PLM_[3]_D	
2D21		Unused							
Tributary [3] Path Provisioning (Read/Write)									
2D22	0x01	EXP_C2_[3]_[7:0]							
2D23	0x00								RX_TIM_[3]
Tributary [3] Path Status (Read Only)									
2D24	0x00			RX_G1_[3]_[2:0]			RX_UNEQ_[3]	RX_PLM_[3]	
2D25	0x01	RX_C2_[3]_[7:0]							
2D26	0x00	B3_ERRCNT_[3]_[7:0]							
2D27	0x00	B3_ERRCNT_[3]_[15:8]							
2D28	0x00	G1_ERRCNT_[3]_[7:0]							
2D29	0x00	G1_ERRCNT_[3]_[15:8]							
2D2A-2D2F		Unused							
2D30-2DEF		: :							
Tributary [16] Path Delta and Second Event Bits (Read / cleared by uP)									
2DF0	0x00	G1ERR_[16]_SECE	B3ERR_[16]_SECE		RX_C2_[16]_D	RX_G1_[16]_D	RX_UNEQ_[16]_D	RX_PLM_[16]_D	
2DF1		Unused							
Tributary [16] Path Provisioning (Read/Write)									
2DF2	0x01	EXP_C2_[16]_[7:0]							
2DF3	0x00								RX_TIM_[16]
Tributary [16] Path Status (Read Only)									
2DF4	0x00			RX_G1_[16]_[2:0]			RX_UNEQ_[16]	RX_PLM_[16]	
2DF5	0x01	RX_C2_[16]_[7:0]							
2DF6	0x00	B3_ERRCNT_[16]_[7:0]							
2DF7	0x00	B3_ERRCNT_[16]_[15:8]							
2DF8	0x00	G1_ERRCNT_[16]_[7:0]							
2DF9	0x00	G1_ERRCNT_[16]_[15:8]							

Table 55. Receive POH Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2DFA-2DFF		Unused							

11.2.7 Receive Side Addresses 0x2E00-0x2FFF (Table 61)

11.2.7.1 ATM Provisioning Bits - Addr 0x2E03 through 0x2E06

The bits in this register control several parameters that define the operation of the Receive ATM processor.

RX_ATM_UDF controls the presence/absence of the Utopia UDF bytes. (See section 9.4.2.)

RX_ATM_HEC_ENH enables the mod 2 addition of the alternating bit pattern to the HEC calculation, an enhancement to the HEC described in section 6.14.2.

RX_ATM_LCD_TIME specifies the time interval used in the determination of ATM Loss of Cell Delineation. (See section 6.14.2.2.)

RX_ATM_IDLE_BYTE, **RX_ATM_IDLE_CAP**, **RX_ATM_IDLE_GFC**, **RX_ATM_IDLE_PTI**, **RX_ATM_IDLE_TRIB_[3:0]**, and **RX_ATM_IDLE_CLP** (reset value of 1) control the definition and payload byte capture of the ATM idle cells. (See section 6.14.4.)

11.2.7.2 ATM Idle Cell Data Status - Addr 0x2E08

The **RX_ATM_IDLE_DATA** register holds the captured idle cell payload data. (See section 6.14.4.)

11.2.7.3 ATM Delta, Event, and Second Event Bits - Addr [0x2F00,0x2FF0]

The delta and event bits contribute to the summary status bits in registers 0x1009 and 0x100B. (See sections 6.14 and 6.14.2.2.) On reset, the delta and second event bits are cleared.

11.2.7.4 ATM Provisioning Bits - Addr [0x2F02,0x2FF2]

RX_ATM_HEC_INH_x enable/disable the ATM HEC header correction processing. (See section 6.14.3.)

11.2.7.5 ATM Status and Cell Counters - Addr [0x2F03-C,0x2FF3-C]

The **RX_ATM_OCD_x** and **RX_ATM_LCD_x** registers are used to report the performance of the ATM Cell Delineation process. (See section 6.14.2.)

The **RX_ATM_HEC_CORR_x** and **RX_ATM_HEC_DROP_x** registers contain the latched results of ATM HEC error counters. (See section 6.14.3.) On reset, these registers are cleared.

The **RX_ATM_CELL_CNT_x** registers contain the latched results of the ATM non-idle cell-error counters. (See section 6.14.4.) On reset, these registers are cleared.

Table 56. Receive ATM Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2E00-2E01		Unused							
ATM and Utopia Level 3 Provisioning (Read/Write)									
2E02	0x00						RX_ATM_ UDF53		
2E03	0x81	RX_ATM_LCD_TIME_[5:0]						RX_ATM_ UDF	RX_ATM_ HEC_ENH

Table 56. Receive ATM Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2E04	0x80	RX_ATM_ IDLE_CLP	RX_ATM_IDLE_PTI_[2:0]			RX_ATM_IDLE_GFC_[3:0]			
2E05	0x01	RX_ATM_ IDLE_CAP		RX_ATM_IDLE_BYTE_[5:0]					
2E06	0x00					RX_ATM_IDLE_TRIB_[3:0]			
2E07		Unused							
Idle Cell Data Status (Read Only)									
2E08	0x00	RX_ATM_IDLE_DATA_[7:0]							
2E09-2EFF		Unused							
ATM Delta, Event and Second Events - Tributary [1] (Read / cleared by uP)									
2F00	0x00		RX_ATM_HEC_DROP_[1]_SECE	RX_ATM_HEC_CORR_[1]_SECE	RX_ATM_LCD_[1]_SECE	RX_ATM_OCD_[1]_SECE	RX_ATM_ LCD_[1]_D	RX_ATM_ OCD_[1]_D	
2F01	Unused								
ATM Provisioning - Tributary [1] (Read/Write)									
2F02	0x00								RX_ATM_HEC_INH_[1]
ATM Status Bits - Tributary [1] (Read Only)									
2F03	0x06						RX_ATM_LCD_[1]	RX_ATM_OCD_[1]	
ATM Counters - Tributary [1] (Read Only)									
2F04	0x00	RX_ATM_HEC_CORR_[1]_[7:0]							
2F05	0x00	RX_ATM_HEC_CORR_[1]_[15:8]							
2F06	0x00							RX_ATM_HEC_CORR_ [1]_[17:16]	
2F07		Unused							
2F08	0x00	RX_ATM_HEC_DROP_[1]_[7:0]							
2F09	0x00	RX_ATM_HEC_DROP_[1]_[15:8]							
2F0A	0x00							RX_ATM_HEC_DROP_ [1]_[17:16]	
2F0B		Unused							
2F0C	0x00	RX_ATM_CELL_CNT_[1]_[7:0]							
2F0D	0x00	RX_ATM_CELL_CNT_[1]_[15:8]							
2F0E	0x00		RX_ATM_CELL_CNT_[1]_[22:16]						
2F0F		Unused							
ATM Delta, Event and Second Events - Tributary [2] (Read / cleared by uP)									

Table 56. Receive ATM Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2F10	0x00		RX_ATM_HEC_DROP_[2]_SECE	RX_ATM_HEC_CORR_[2]_SECE	RX_ATM_LCD_[2]_SECE	RX_ATM_OCD_[2]_SECE	RX_ATM_LCD_[2]_D	RX_ATM_OCD_[2]_D	
2F11		Unused							
ATM Provisioning - Tributary [2] (Read/Write)									
2F12	0x00								RX_ATM_HEC_INH_[2]
ATM Status Bits - Tributary [2] (Read Only)									
2F13	0x06						RX_ATM_LCD_[2]	RX_ATM_OCD_[2]	
ATM Counters - Tributary [2] (Read Only)									
2F14	0x00	RX_ATM_HEC_CORR_[2]_[7:0]							
2F15	0x00	RX_ATM_HEC_CORR_[2]_[15:8]							
2F16	0x00							RX_ATM_HEC_CORR_[2]_[17:16]	
2F17		Unused							
2F18	0x00	RX_ATM_HEC_DROP_[2]_[7:0]							
2F19	0x00	RX_ATM_HEC_DROP_[2]_[15:8]							
2F1A	0x00							RX_ATM_HEC_DROP_[2]_[17:16]	
2F1B		Unused							
2F1C	0x00	RX_ATM_CELL_CNT_[2]_[7:0]							
2F1D	0x00	RX_ATM_CELL_CNT_[2]_[15:8]							
2F1E	0x00		RX_ATM_CELL_CNT_[2]_[22:16]						
2F1F		Unused							
ATM Delta, Event and Second Events - Tributary [3] (Read / cleared by uP)									
2F20	0x00		RX_ATM_HEC_DROP_[3]_SECE	RX_ATM_HEC_CORR_[3]_SECE	RX_ATM_LCD_[3]_SECE	RX_ATM_OCD_[3]_SECE	RX_ATM_LCD_[3]_D	RX_ATM_OCD_[3]_D	
2F21		Unused							
ATM Provisioning - Tributary [3] (Read/Write)									
2F22	0x00								RX_ATM_HEC_INH_[3]
ATM Status Bits - Tributary [3] (Read Only)									
2F23	0x06						RX_ATM_LCD_[3]	RX_ATM_OCD_[3]	

Table 56. Receive ATM Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ATM Counters - Tributary [3] (Read Only)									
2F24	0x00	RX_ATM_HEC_CORR_[3]_[7:0]							
2F25	0x00	RX_ATM_HEC_CORR_[3]_[15:8]							
2F26	0x00							RX_ATM_HEC_CORR_ [3]_[17:16]	
2F27		Unused							
2F28	0x00	RX_ATM_HEC_DROP_[2]_[7:0]							
2F29	0x00	RX_ATM_HEC_DROP_[2]_[15:8]							
2F2A	0x00							RX_ATM_HEC_DROP_ [3]_[17:16]	
2F2B		Unused							
2F2C	0x00	RX_ATM_CELL_CNT_[3]_[7:0]							
2F2D	0x00	RX_ATM_CELL_CNT_[3]_[15:8]							
2F2E	0x00		RX_ATM_CELL_CNT_[3]_[22:16]						
2F2F		Unused							
2F30-2FEF		: :							
ATM Delta, Event and Second Events - Tributary [16] (Read / cleared by uP)									
2FF0	0x00		RX_ATM_HEC _DROP_[16]_ SECE	RX_ATM_HEC _CORR_[16]_ SECE	RX_ATM_LCD _[16]_SECE	RX_ATM_OCD _[16]_SECE	RX_ATM_ LCD_[16]_D	RX_ATM_ OCD_[16]_ D	
2FF1		Unused							
ATM Provisioning - Tributary [16] (Read/Write)									
2FF2	0x00								RX_ATM_HEC _INH_[16]
ATM Status Bits - Tributary [16] (Read Only)									
2FF3	0x06						RX_ATM_LCD_[16]	RX_ATM_OCD_[16]	
ATM Counters - Tributary [16] (Read Only)									
2FF4	0x00	RX_ATM_HEC_CORR_[16]_[7:0]							
2FF5	0x00	RX_ATM_HEC_CORR_[16]_[15:8]							
2FF6	0x00							RX_ATM_HEC_CORR_ [16]_[17:16]	
2FF7		Unused							
2FF8	0x00	RX_ATM_HEC_DROP_[16]_[7:0]							
2FF9	0x00	RX_ATM_HEC_DROP_[16]_[15:8]							

Table 56. Receive ATM Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
2FFA	0x00							RX_ATM_HEC_DROP_ [16]_[17:16]	
2FFB		Unused							
2FFC	0x00	RX_ATM_CELL_CNT_[16]_[7:0]							
2FFD	0x00	RX_ATM_CELL_CNT_[16]_[15:8]							
2FFE	0x00		RX_ATM_CELL_CNT_[16]_[22:16]						
2FFF		Unused							

11.2.8 Receive Side Addresses 0x3000 through 0x31FF (Table)

11.2.8.1 System Interface Provisioning Bits - Addr 0x3002 through 0x300B

RX_CLAV/PDA_DSST_y controls the assert/deassert of the 4 *RX_CLAV_PDA_y* signals. (See section 9.5.3.)

RX_SING_CLAV controls the single cell available signal mode, in which a single cell available signal, RX_CLAV_0, provides the multiplexed status of all “active” ports. This is enabled by setting **RX_SING_CLAV** = 1. When **RX_SING_CLAV** = 0, four cell available signals are provided, RX_CLAV_y for y = 0 to 3, which display the multiplexed status of their associated tributaries. **RX_SYSINT_WIDTH** controls the width of the system interface, **RX_SYSINT_POLL** places the system interface in direct or multiplexed-polling mode, for certain configurations. (See section 9.3.1.)

RX_PRTY_MODE_y controls the parity mode of bus y of the Receive system interface. (See section 6.15.4.)

RX_CLK_OUT_INH_[1:4] enables/inhibits the *RX_CLK_OUT_[1:4]* outputs from the Receive system interface.

RX_FIFOFULL_RPA_y_[3:0] and **RX_FIFOEMPTY_RPA_y_[3:0]** registers allow the user to program the RX FIFO data levels that cause *RX_CLAV_PDA_y* to assert/deassert. These thresholds operate for both the Utopia ATM as well as the packet modes. (See section 12.0 and section 9.5.3.)

11.2.8.2 System Interface Event and Second Event Bits - Addr [0x3100,0x31F0]

The event and second event bits contribute to the summary status bits in transmit registers 0x1009 and 0x100A. (See sections 6.13 and 6.13.4.) On reset, these event and second event bits are cleared.

11.2.8.3 System Interface Errored Packet Counters - Addr [0x3102,0x31F2]

The **RX_FIFOERR_CNT_x** registers are used to report the number of receive packets that are lost due to RX FIFO overflow in tributary x at the receive system interface. (See section 6.15.2.)

Table 57. Receive System Interface Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
3000-3001		Unused							
System Interface Provisioning(Read/Write)									
3002	0x00	RX_PRTY_MODE_[1:4]				RX_CLAV/PDA_DSST_[1:4]			

Table 57. Receive System Interface Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
3003	0x13	RX_CLAV_MODE		RX_SYS_ATM_EOP	RX_SING_CLAV	RX_SYS_B2B_ALLOW	RX_SYSINT_POLL	RX_SYS_SIZE_MODE	RX_SYSINT_WIDTH
3004	0x00					RX_CLK_OUT_INH_[1:4]			
3005		Unused							
3006	0x00					RX_FIFOEMPTY_RPA_[1]_[3:0]			
3007	0x00					RX_FIFOFULL_RPA_[1]_[3:0]			
3008	0x00					RX_FIFOEMPTY_RPA_[2]_[3:0]			
3009	0x00					RX_FIFOFULL_RPA_[2]_[3:0]			
300A	0x00					RX_FIFOEMPTY_RPA_[3]_[3:0]			
300B	0x00					RX_FIFOFULL_RPA_[3]_[3:0]			
300C	0x00					RX_FIFOEMPTY_RPA_[4]_[3:0]			
300D	0x00					RX_FIFOFULL_RPA_[4]_[3:0]			
300E-30FF		Unused							
System Interface Event Bits - Tributary [1] (Read / cleared by uP)									
3100	0x00					RX_FIFOERR_[1]_SECE	RX_FIFOERR_[1]_E		
3101		Unused							
System Interface FIFO Error Counter - Tributary [1] (Read Only)									
3102	0x00	RX_FIFOERR_CNT_[1]_[7:0]							
3103-310F		Unused							
System Interface Event Bits - Tributary [2] (Read / cleared by uP)									
3110	0x00					RX_FIFOERR_[2]_SECE	RX_FIFOERR_[2]_E		
3111		Unused							
System Interface FIFO Error Counter - Tributary [2] (Read Only)									
3112	0x00	RX_FIFOERR_CNT_[2]_[7:0]							
3113-311F		Unused							
System Interface Event Bits - Tributary [3] (Read / cleared by uP)									
3120	0x00					RX_FIFOERR_[3]_SECE	RX_FIFOERR_[3]_E		
3121		Unused							
System Interface FIFO Error Counter - Tributary [3] (Read Only)									
3122	0x00	RX_FIFOERR_CNT_[3]_[7:0]							
3123-312F		Unused							

Table 57. Receive System Interface Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
3130-31EF		: :							
System Interface Event Bits - Tributary [16] (Read / cleared by uP)									
31F0	0x00					RX_FIFOERR_[16]_SECE	RX_FIFOERR_[16]_E		
31F1		Unused							
System Interface FIFO Error Counter - Tributary [16] (Read Only)									
31F2	0x00	RX_FIFOERR_CNT_[16]_[7:0]							
31F3-31FF	Unused								

11.2.9 Receive Side Addresses 0x3200 through 0x33FF (Table)

11.2.9.1 POS Provisioning Bits- Addr 0x3201 through 0x3207

The **RX_POS_FCS_INH** bit controls the termination of the POS FCS. (See section 6.13.5.)

The **RX_POS_PMAX_[15:0]** and **RX_POS_PMIN_[3:0]** hold the minimum and maximum packet length thresholds. (See section 6.13.4.)

RX_POS_FIFOUNDER_BYTE_[7:0] holds the special FIFO underflow byte code. (See section 6.13.4.)

RX_POS_FCS_BIT_ORDR controls the bit order of the FCS calculation. (See section 6.13.5.)

11.2.9.2 POS Delta and Second Event Bits - Addr [0x3300,0x33F0]

The delta and second event bits contribute to the summary status bits in registers 0x1009 and 0x100B. (See sections 6.13 and 6.13.4.) On reset, the delta and second event bits are cleared.

11.2.9.3 POS Provisioning Bits - Addr [0x3301-3,0x33F1-3]

The **RX_POS_FCS_MODE_x** bits control the mode of the POS FCS. (See section 6.13.5.)

The **RX_POS_PMAX_ENB_x** and **RX_POS_PMIN_ENB_x** control the handling of packets that exceed minimum and maximum packet length restrictions. (See section 6.13.4.)

The **RX_POS_ADRCTL_DROP_INH_x** registers control the termination of the HDLC address and control bytes. (See section 6.13.6.)

RX_POS_FIFOUNDER_MODE_x control the identification and removal of a special FIFO underflow byte code. (See section 6.13.4.)

RX_POS_ADDRESS_x_[7:0] and **RX_POS_CONTROL_x_[7:0]** contain the expected address and control fields to be monitored in the HDLC frame. (See section 6.13.6.2.)

11.2.9.4 POS Status and Packet Counters - Addr [0x3304-F,0x33F4-F]

The **RX_POS_ADRCTL_INVALID_x** registers provide an indication as to whether the received address and control bytes are valid. (See section 6.13.6.)

The **RX_POS_FCS_ERRCNT_x** registers are used to report the number of FCS errors detected. (See section 6.13.5.)

The **RX_POS_PABORT_ERRCNT_x** registers are used to report the number of aborted receive packets. (See section 6.13.4.)

The **RX_POS_PMAX_ERRCNT_x** and **RX_POS_PMIN_ERRCNT_x** registers are used to report the number of receive packets that violate minimum and maximum packet length restrictions. (See section 6.13.4.)

RX_POS_PKT_CNT_x_[22:0] count the number of valid packets received over the SONET/SDH line. (See section 6.13.8.)

Table 58. Receive POS Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
3200-3201		Unused							
POS Provisioning(Read/Write)									
3202	0x00				RX_POS_FCS_INH				
3203	0x00	RX_POS_FCS_BIT_ORDR[1:4]				RX_POS_PMIN_[3:0]			
3204	0xDE	RX_POS_PMAX_[7:0]							
3205	0x05	RX_POS_PMAX_[15:8]							
3206	0x50	RX_POS_FIFOUNDER_BYTE_[7:0]							
3207-32FF		Unused							
POS Event and Second Event Bits - Tributary [1] (Read / cleared by uP)									
3300	0x00				RX_POS_FCS_ERR_[1]_SECE	RX_POS_PMIN_ERR_[1]_SECE	RX_POS_PMAX_ERR_[1]_SECE	RX_POS_PABORT_ERR_[1]_SECE	RX_POS_ADRCTL_INVALID_[1]_D
POS Provisioning - Tributary [1] (Read/Write)									
3301	0x80	RX_POSTHDLC_DSCR_INH_[1]	RX_POS_FIFOUNDER_MODE_[1]	RX_POS_PMIN_ENB_[1]	RX_POS_PMAX_ENB_[1]			RX_POS_FCS_MODE_[1]	RX_POS_ADRCTL_DROP_INH_[1]
3302	0xFF	RX_POS_ADDRESS_[1]_[7:0]							
3303	0x03	RX_POS_CONTROL_[1]_[7:0]							
POS Status - Tributary [1] (Read Only)									
3304	0x00								RX_POS_ADRCTL_INVALID_[1]
POS Packet Counters - Tributary [1] (Read Only)									
3305	0x00	RX_POS_PABORT_ERRCNT_[1]_[7:0]							
3306	0x00	RX_POS_PMAX_ERRCNT_[1]_[7:0]							
3307	0x00	RX_POS_PMIN_ERRCNT_[1]_[7:0]							
3308	0x00	RX_POS_FCS_ERRCNT_[1]_[7:0]							
3309	0x00	RX_POS_FCS_ERRCNT_[1]_[15:8]							
330A	0x00			RX_POS_FCS_ERRCNT_[1]_[21:16]					

Table 58. Receive POS Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
330B		Unused							
330C	0x00	RX_POS_PKT_CNT_[1]_[7:0]							
330D	0x00	RX_POS_PKT_CNT_[1]_[15:8]							
330E	0x00		RX_POS_PKT_CNT_[1]_[22:16]						
330F		Unused							
POS Event and Second Event Bits - Tributary [2] (Read / cleared by uP)									
3310	0x00				RX_POS_FCS_ERR_[2]_SECE	RX_POS_PMIN_ERR_[2]_SECE	RX_POS_PMAX_ERR_[2]_SECE	RX_POS_PABORT_ERR_[2]_SECE	RX_POS_ADRCTL_INVALID_[2]_D
POS Provisioning - Tributary [2] (Read/Write)									
3311	0x80	RX_POSTHDLC_DSCR_INH_[2]	RX_POS_FIFOUNDER_MODE_[2]	RX_POS_PMIN_ENB_[2]	RX_POS_PMAX_ENB_[2]			RX_POS_FCS_MODE_[2]	RX_POS_ADRCTL_DROP_INH_[2]
3312	0xFF	RX_POS_ADDRESS_[2]_[7:0]							
3313	0x03	RX_POS_CONTROL_[2]_[7:0]							
POS Status - Tributary [2] (Read Only)									
3314	0x00								RX_POS_ADRCTL_INVALID_[2]
POS Packet Counters - Tributary [2] (Read Only)									
3315	0x00	RX_POS_PABORT_ERRCNT_[2]_[7:0]							
3316	0x00	RX_POS_PMAX_ERRCNT_[2]_[7:0]							
3317	0x00	RX_POS_PMIN_ERRCNT_[2]_[7:0]							
3318	0x00	RX_POS_FCS_ERRCNT_[2]_[7:0]							
3319	0x00	RX_POS_FCS_ERRCNT_[2]_[15:8]							
331A	0x00			RX_POS_FCS_ERRCNT_[2]_[21:16]					
331B		Unused							
331C	0x00	RX_POS_PKT_CNT_[2]_[7:0]							
331D	0x00	RX_POS_PKT_CNT_[2]_[15:8]							
331E	0x00		RX_POS_PKT_CNT_[2]_[22:16]						
331F		Unused							
POS Event and Second Event Bits - Tributary [3] (Read / cleared by uP)									
3320	0x00				RX_POS_FCS_ERR_[3]_SECE	RX_POS_PMIN_ERR_[3]_SECE	RX_POS_PMAX_ERR_[3]_SECE	RX_POS_PABORT_ERR_[3]_SECE	RX_POS_ADRCTL_INVALID_[3]_D

Table 58. Receive POS Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
POS Provisioning - Tributary [3] (Read/Write)									
3321	0x80	RX_POSTHDLC_DSCR_INH_[3]	RX_POS_FIFOUNDER_MODE_[3]	RX_POS_PMIN_ENB_[3]	RX_POS_PMAX_ENB_[3]			RX_POS_FCS_MODE_[3]	RX_POS_ADRCTL_DROP_INH_[3]
3322	0xFF	RX_POS_ADDRESS_[3]_[7:0]							
3323	0x03	RX_POS_CONTROL_[3]_[7:0]							
POS Status - Tributary [3] (Read Only)									
3324	0x00								RX_POS_ADRCTL_INVALID_[3]
POS Packet Counters - Tributary [3] (Read Only)									
3325	0x00	RX_POS_PABORT_ERRCNT_[3]_[7:0]							
3326	0x00	RX_POS_PMAX_ERRCNT_[3]_[7:0]							
3327	0x00	RX_POS_PMIN_ERRCNT_[3]_[7:0]							
3328	0x00	RX_POS_FCS_ERRCNT_[3]_[7:0]							
3329	0x00	RX_POS_FCS_ERRCNT_[3]_[15:8]							
332A	0x00			RX_POS_FCS_ERRCNT_[3]_[21:16]					
332B		Unused							
332C	0x00	RX_POS_PKT_CNT_[3]_[7:0]							
332D	0x00	RX_POS_PKT_CNT_[3]_[15:8]							
332E	0x00		RX_POS_PKT_CNT_[3]_[22:16]						
332F		Unused							
3330-33EF		: :							
POS Event and Second Event Bits - Tributary [16] (Read / cleared by uP)									
33F0	0x00				RX_POS_FCS_ERR_[16]_SECE	RX_POS_PMIN_ERR_[16]_SECE	RX_POS_PMAX_ERR_[16]_SECE	RX_POS_PABORT_ERR_[16]_SECE	RX_POS_ADRCTL_INVALID_[16]_D
POS Provisioning - Tributary [16] (Read/Write)									
33F1	0800	RX_POSTHDLC_DSCR_INH_[16]	RX_POS_FIFOUNDER_MODE_[16]	RX_POS_PMIN_ENB_[16]	RX_POS_PMAX_ENB_[16]			RX_POS_FCS_MODE_[16]	RX_POS_ADRCTL_DROP_CRI_NH_[16]
33F2	0xFF	RX_POS_ADDRESS_[16]_[7:0]							
33F3	0x03	RX_POS_CONTROL_[16]_[7:0]							
POS Status - Tributary [16] (Read Only)									

Table 58. Receive POS Monitoring and Provisioning Register Map

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
33F4	0x00								RX_POS_ ADRCTL_ INVALID_[16]
POS Packet Counters - Tributary [16] (Read Only)									
33F5	0x00	RX_POS_PABORT_ERRCNT_[16]_[7:0]							
33F6	0x00	RX_POS_PMAX_ERRCNT_[16]_[7:0]							
33F7	0x00	RX_POS_PMIN_ERRCNT_[16]_[7:0]							
33F8	0x00	RX_POS_FCS_ERRCNT_[16]_[7:0]							
33F9	0x00	RX_POS_FCS_ERRCNT_[16]_[15:8]							
33FA	0x00			RX_POS_FCS_ERRCNT_[16]_[21:16]					
33FB		Unused							
33FC	0x00	RX_POS_PKT_CNT_[16]_[7:0]							
33FD	0x00	RX_POS_PKT_CNT_[16]_[15:8]							
33FE	0x00		RX_POS_PKT_CNT_[16]_[22:16]						
33FF		Unused							

11.2.10 APS Interface Addresses 0x3600 through 0x3FFE (Table)

11.2.10.1 APS Input Delta and Event Bits, and Masks - Addr 0x3600

These delta and event bits contribute to the summary status bit in receive register 0x200C. On reset, these event bits are cleared. Their mask bits are used to enable/disable the contribution of the **APS_B1ERR_SECE** and **APS_OOF_y_D** to the summary interrupt.

11.2.10.2 APS Input Provisioning Bits - Addr 0x3604

APS_DSCRINH controls descrambler for the APS Input interface. (See section 8.2.2.)

APS_DELAY_EN enables / disables additional delay to frame-align APS input with primary line inputs.

11.2.10.3 APS Output Provisioning Bits - Addr 0x3406

APS_SCRINH controls scrambler for the APS Output interface. (See section 7.2.3.) **APS_B1_INV** controls the parity of the B1 byte over the APS Output interface. (See section 7.2.1.)

11.2.10.4 APS Status Bits - Addr 0x3408

The **APS_OOF_y** bits are used to report the current state of the corresponding APS interface framer. (See section 8.2.)

11.2.10.5 B1 Error Counter - Addr 0x360A and 0x360B

The B1 parity error count from the APS Input Processing block is stored in the **APS_B1_ERRCNT_[15:0]** register. (See section 8.2.3.)

Table 59. APS Interface Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
APS Input Delta and Event Bits (Read / cleared by uP)									
3600	0x00	APS_B1ERR_SECE	APS_MISALIGN_ERR_E			APS_OOF_[1]_D	APS_OOF_[2]_D	APS_OOF_[3]_D	APS_OOF_[4]_D
3601		Unused							
APS Input Mask Bits (Read/Write)									
3602	0xCF	APS_B1ERR_SECE_MASK	APS_MISALIGN_ERR_E_MASK			APS_OOF_[1]_D_MASK	APS_OOF_[2]_D_MASK	APS_OOF_[3]_D_MASK	APS_OOF_[4]_D_MASK
3603	0x00	Unused							
APS Input Provisioning Bits (Read/Write)									
3604	0x50	APS_CONSEC_[3:0]				APS_DSCRINH			APS_DELAY_EN
3605	0x00	Unused							
APS Output Provisioning Bits (Read/Write)									
3606	0x00	APS_B1_INV				APS_SCRINH			
3607		Unused							
APS Input Status Bits (Read Only)									
3608	0x00					APS_OOF_[1]	APS_OOF_[2]	APS_OOF_[3]	APS_OOF_[4]
3609	0x00	Unused							
APS B1 Error Counters (Read Only)									
360A	0x00	APS_B1_ERRCNT_[7:0]							
360B	0x00	APS_B1_ERRCNT_[15:8]							
360C-36FF		Unused							
Tributary [1] APS Input Event Bits (Read / cleared by uP)									
3700	0x00						APS_K1_[1]_D	APS_K2_[1]_D	APS_UNSTAB_[1]_D
3701		Unused							
Tributary [1] APS Input Mask Bits (Read/Write)									
3702	0x07						APS_K1_[1]_D_MASK	APS_K2_[1]_D_MASK	APS_UNSTAB_[1]_D_MASK
3703	0x00	Unused							

Table 59. APS Interface Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Tributary [1] APS Input Status Bits (Read Only)									
3704	0x00	APS_K1_[1]_[7:0]							
3705	0x00	APS_K2_[1]_[7:0]							
3706	0x00	APS_K1_GEN_[1]_[7:0]							
3707	0x00	APS_K2_GEN_[1]_[7:0]							
3708	0x00								APS_UNSTAB_[1]
3709-370F	0x00	Unused							
Tributary [2] APS Input Event Bits (Read / cleared by uP)									
3710	0x00						APS_K1_[2]_D	APS_K2_[2]_D	APS_UNSTAB_[2]_D
3711		Unused							
Tributary [2] APS Input Mask Bits (Read/Write)									
3712	0x07						APS_K1_[2]_D_MASK	APS_K2_[2]_D_MASK	APS_UNSTAB_[2]_D_MASK
3713	0x00	Unused							
Tributary [2] APS Input Status Bits (Read Only)									
3714	0x00	APS_K1_[2]_[7:0]							
3715	0x00	APS_K2_[2]_[7:0]							
3716	0x00	APS_K1_GEN_[2]_[7:0]							
3717	0x00	APS_K2_GEN_[2]_[7:0]							
3718	0x00								APS_UNSTAB_[1]
3719-371F	0x00	Unused							
3720-37EF	0x00	: :							
Tributary [16] APS Input Event Bits (Read / cleared by uP)									
37F0	0x00						APS_K1_[16]_D	APS_K2_[16]_D	APS_UNSTAB_[16]_D
37F1		Unused							
Tributary [1] APS Input Mask Bits (Read/Write)									

Table 59. APS Interface Provisioning

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
37F2	0x07						APS_K1_[16]_D_MASK	APS_K2_[16]_D_MASK	APS_UNSTAB_ [16]_D_MASK
37F3	0x00	Unused							
Tributary [16] APS Input Status Bits (Read Only)									
37F4	0x00	APS_K1_[16]_[7:0]							
37F5	0x00	APS_K2_[16]_[7:0]							
37F6	0x00	APS_K1_GEN_[16]_[7:0]							
37F7	0x00	APS_K2_GEN_[16]_[7:0]							
37F8	0x00								APS_UNSTAB_ [16]
37F9-3FFE	0x00	Unused							

11.2.11 Address 0x3FFF

11.2.11.1 Device Identification Code - Addr 0x3FFF

This register holds a device identification number for RHINE: **DEV_ID_[7:0]** = 0x03.

Table 60. Device ID Register

Addr	Reset Value	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Device Identification Code (Read Only)									
3FFF	0x03	DEV_ID_[7:0]							

12.0 AC Electrical Characteristics

12.1 FlexBus3™ System Interface Transmit Timing

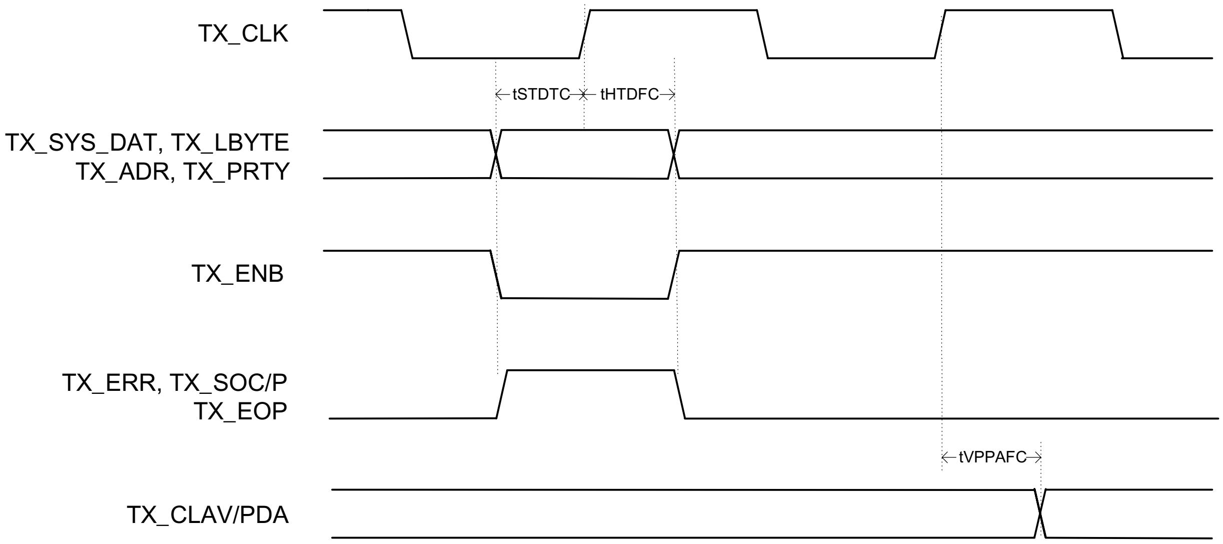


Table 61. FlexBus-3 System Interface Transmit Timing

Label	Parameter	Min	Nom	Max	Units
TX_CLK	TX_CLK frequency		100	104	MHz
tSTDTC	TX_SYS_DAT, TX_LBYTE, TX_ADR, TX_PRTY, TX_ENB, TX_ERR, TX_SOC/P, TX_EOP Setup time to TX_CLK high	2			ns
t_{HTDFC}	TX_SYS_DAT, TX_LBYTE, TX_ADR, TX_PRTY, TX_ENB, TX_ERR, TX_SOC/P, TX_EOP Hold time from TX_CLK high	1			ns
t_{VPPAFC}	TX_CLAV/PDA valid from TX_CLK high	1		5.3	ns

12.2 FlexBus3^{TM} System Interface Receive Timing

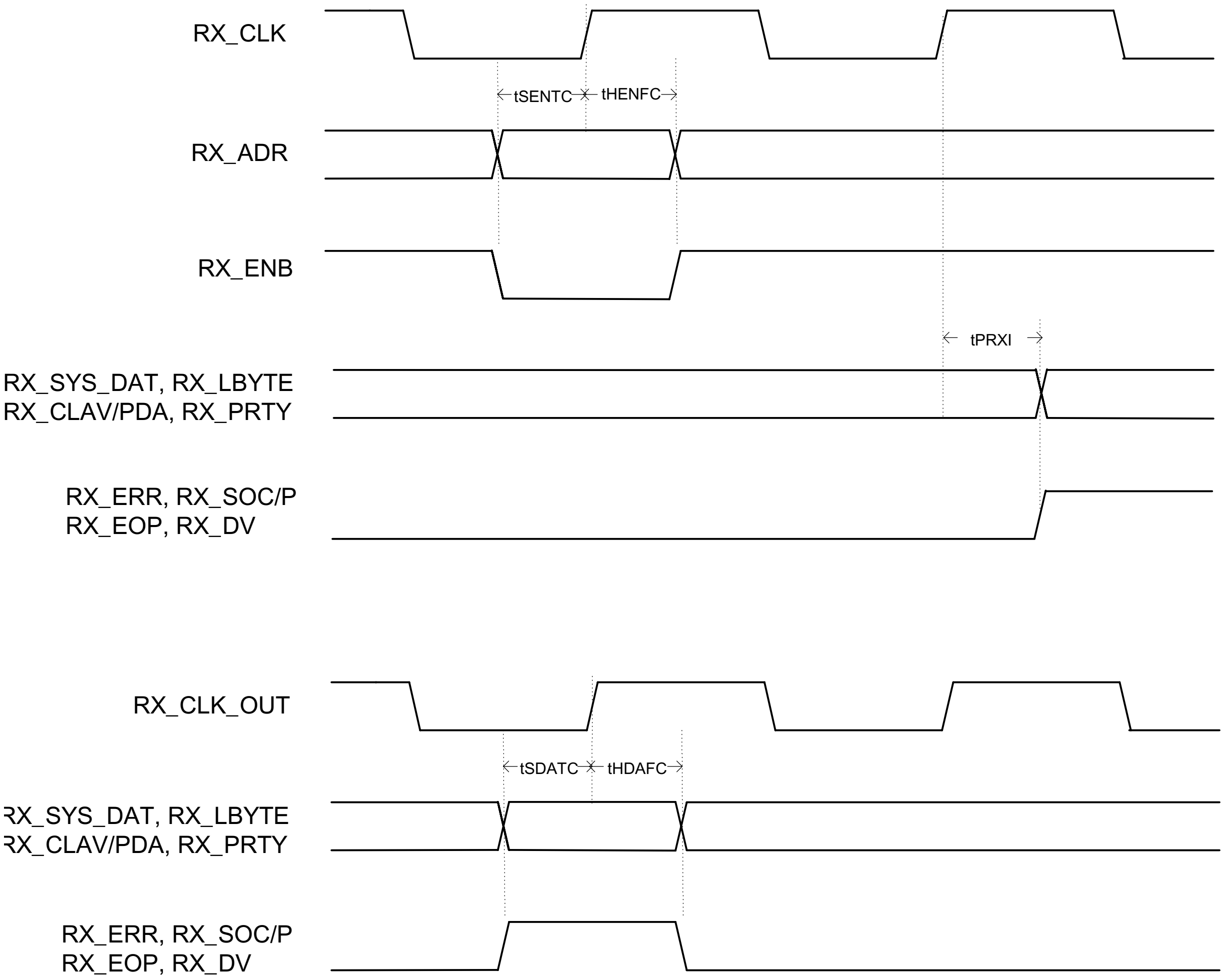


Table 62. FlexBus-3 System Interface Receive Timing

Label	Parameter	Min	Nom	Max	Units
RX_CLK	RX_CLK frequency		100	104	MHz
tSENTC	Setup RX_ENB, RX_ADR to RX_CLK high	2			ns
tHENFC	Hold RX_ENB, RX_ADR from RX_CLK high	1			ns
tPRXI	RX_SYS_DAT, RX_LBYTE, RX_CLAV/PDA, RX_PRTY, RX_ERR, RX_SOC/P RX_EOP, RX_DV valid from RX_CLK high (25 pF load)	2		6	ns
tSDATC	RX_SYS_DAT, RX_LBYTE, RX_CLAV/PDA, RX_PRTY, RX_ERR, RX_SOC/P RX_EOP, RX_DV valid set-up time to RX_CLK_OUT	5			ns

Table 62. FlexBus-3 System Interface Receive Timing

Label	Parameter	Min	Nom	Max	Units
tXDAHC	RX_SYS_DAT, RX_LBYTE, RX_CLAV/PDA, RX_PRTY, RX_ERR, RX_SOC/P RX_EOP, RX_DV valid hold time from RX_CLK_OUT	1			ns

12.3 UTOPIA-3 System Interface Transmit Timing

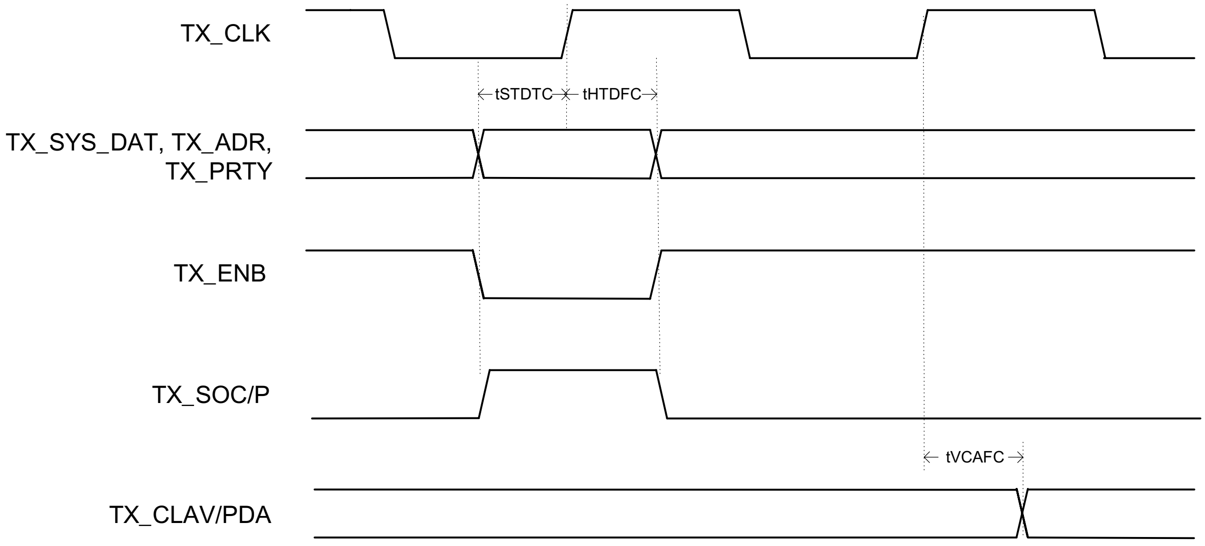


Table 63. Utopia-3 System Inteface Transmit Timing

Label	Parameter	Min	Nom	Max	Units
TX_CLK	TX_CLK frequency		100	104	MHz
tSTDTC	TX_SYS_DAT, TX_ADR, TX_PRTY, TX_ENB, TX_SOC/P Setup time to TX_CLK high	2			ns
t_{HTDFC}	TX_SYS_DAT, TX_ADR, TX_PRTY, TX_ENB, TX_SOC/P Hold from TX_CLK high	1			ns
t_{VCAFC}	TX_CLAV/PDA valid from TX_CLK high	1		5.3	ns

12.4 UTOPIA-3 System Interface Receive Timing

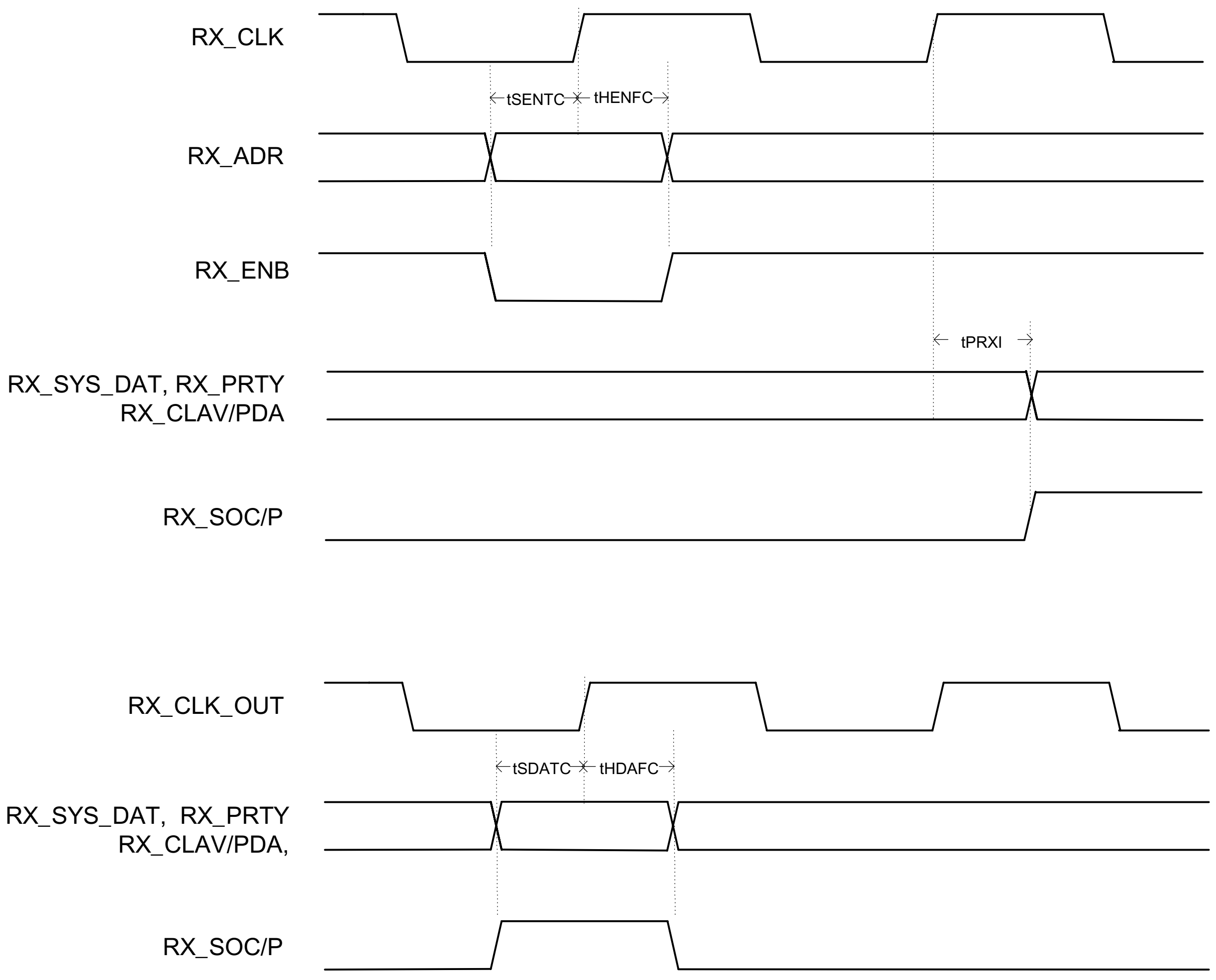


Table 64. UTOPIA-3 System Interface Receive Timing

Label	Parameter	Min	Nom	Max	Units
RX_CLK	RX_CLK frequency		100	104	MHz
tSENTC	Setup RX_ENB, RX_ADR to RX_CLK high	2			ns
tHENFC	Hold RX_ENB, RX_ADR from RX_CLK high	1			ns
tPRXI	RX_SYS_DAT, RX_CLAV/PDA, RX_PRTY, RX_SOC/P valid from RX_CLK high (25 pF load)	2		6	ns
tSDATC	RX_SYS_DAT, RX_CLAV/PDA, RX_PRTY, RX_SOC/P valid set-up time to RX_CLK_OUT	3			ns

Table 64. UTOPIA-3 System Interface Receive Timing

Label	Parameter	Min	Nom	Max	Units
tXDAHC	RX_SYS_DAT, RX_CLAV/PDA, RX_PRTY, RX_SOC/P valid hold time from RX_CLK_OUT	1			ns

12.5 SONET Line Interface Transmit Timing

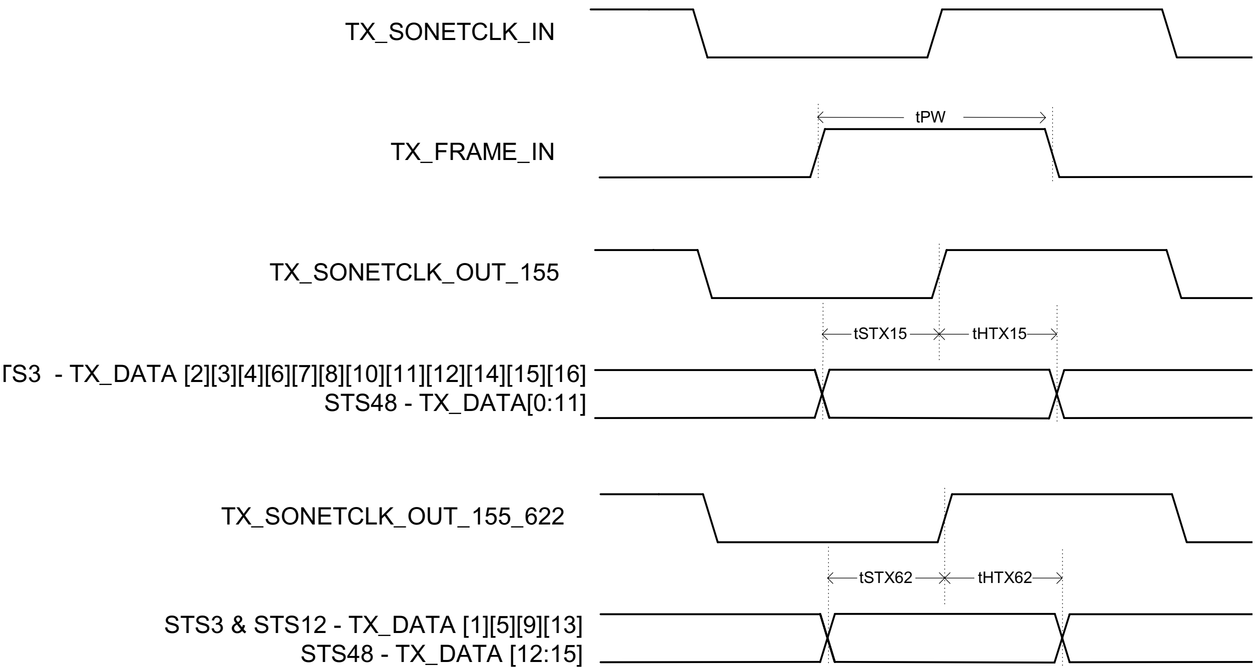


Table 65. SONET Line Interface Transmit Timing

Label	Parameter	Min	Nom	Max	Units
t_{PW}	TX_FRAME_IN Pulse Width	20			ns
STS12 Line Interface 622 Mb/s Operation					
	TX_SONETCLK_IN frequency		622.08		MHz
	TX_SONETCLK_IN duty cycle	45		55	%

Table 65. SONET Line Interface Transmit Timing

Label	Parameter	Min	Nom	Max	Units
	TX_SONETCLK_OUT_155_622 frequency		622.08		MHz
	TX_SONETCLK_OUT_155_622 duty cycle	45		55	%
t_{STX62}	TX_DATA[1][5][9][13] setup time w.r.t. TX_SONETCLK_OUT_155_622	400			ps
t_{HTX62}	TX_DATA[1][5][9][13] hold time w.r.t. TX_SONETCLK_OUT_155_622	400			ps
STS48 Line Interface 155 Mb/s Operation					
	TX_SONETCLK_IN frequency		155.52		MHz
	TX_SONETCLK_IN duty cycle	45		55	%
	TX_SONETCLK_OUT_155 frequency		155.52		MHz
	TX_SONETCLK_OUT_155 duty cycle	45		55	%
t_{STX15}	TX_DATA [0:15] setup time w.r.t. TX_SONETCLK_OUT_155	2			ns
t_{HTX15}	TX_DATA[0:15] hold time w.r.t. TX_SONETCLK_OUT_155	.5			ns
STS3 Line Interface 155 Mb/s Operation					
	TX_SONETCLK_IN frequency		155.52		MHz
	TX_SONETCLK_IN duty cycle	45		55	%
	TX_SONETCLK_OUT_155 frequency		155.52		MHz
	TX_SONETCLK_OUT_155 duty cycle	45		55	%
	TX_SONETCLK_OUT_155_622 frequency		155.52		MHz
	TX_SONETCLK_OUT_155_622 duty cycle	45		55	%
t_{STX62}	TX_DATA [1][5][9][13] setup time w.r.t. TX_SONETCLK_OUT_155_622	2			ns
t_{HTX62}	TX_DATA[1][5][9][13] hold time w.r.t. TX_SONETCLK_OUT_155_622	.5			ns
t_{STX15}	TX_DATA [2][3][4][6][7][8][10][11][12][14][15][16] setup time w.r.t. TX_SONETCLK_OUT_155	2			ns
t_{HTX15}	TX_DATA [2][3][4][6][7][8][10][11][12][14][15][16] hold time w.r.t. TX_SONETCLK_OUT_155	.5			ns

* **Important:** Please see the Application Note called "System Design Consideration When Using Rhine S4804 Rhine 3.0 in STS48 Line Interface Mode," to ensure proper operation. This application note is not necessary when designing with Rhine 4.0 or Rhine 4.1.

12.6 SONET Line Interface Receive Timing

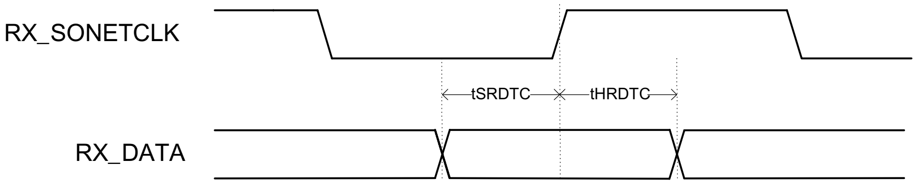


Table 66. SONET Line Interface Receive Timing

Label	Parameter	Min	Nom	Max	Units
STS12 Line Interface 622 Mb/s Operation					
	RX_SONETCLK frequency		622.0 8		MHz
	RX_SONETCLK duty cycle	45		55	%
t_{SRDTC}	Setup RX_DATA w.r.t. RX_SONETCLK	450			ps
t_{HRDTC}	Hold RX_DATA w.r.t. RX_SONETCLK	450			ps
STS48 and STS3 Line Interface 155 Mb/s Operation					
	RX_SONETCLK frequency		155.5 2		MHz
	RX_SONETCLK duty cycle	45		55	%
t_{SRDTC}	Setup RX_DATA w.r.t. RX_SONETCLK	1.5			ns
t_{HRDTC}	Hold RX_DATA w.r.t. RX_SONETCLK	.5			ns

12.7 Receive TOH Drop Timing

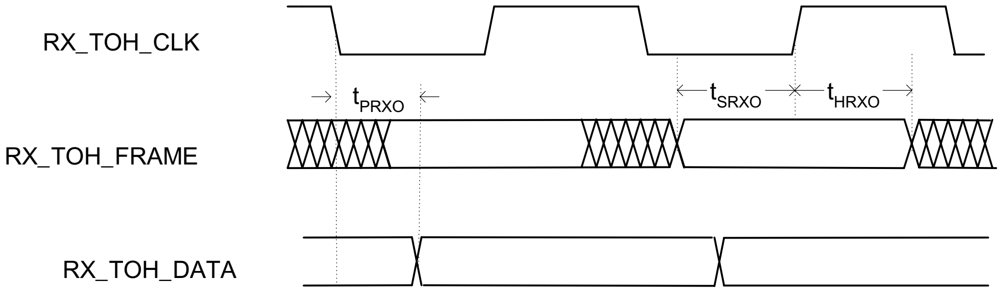


Table 67. Receive TOH Drop Timing

Label	Parameter	Min	Nom	Max	Units
	RX_TOH_CLK frequency		19.44		MHz
t_{PRXO}	Propagation delay w.r.t. RX_TOH_CLK	-10		10	ns
tSRXO	Setup time w.r.t. to RX_TOH_CLK	10			ns
tHRXO	Hold time w.r.t. to RX_TOH_CLK	60			ns

12.8 Transmit TOH Insert Timing

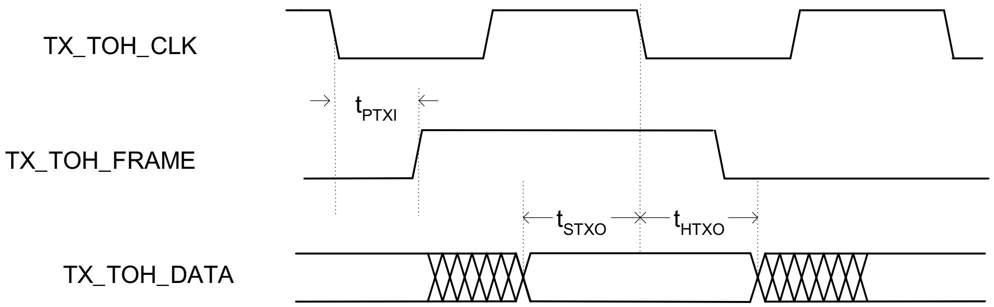


Table 68. Transmit TOH Insert Timing

Label	Parameter	Min	Nom	Max	Units
	TX_TOH_CLK frequency		19.44		MHz
t_{PTXI}	Propagation delay w.r.t. TX_TOH_CLK	-10		10	ns

Table 68. Transmit TOH Insert Timing

Label	Parameter	Min	Nom	Max	Units
tSTXO	Setup time w.r.t. to TX_TOH_CLK	10			ns
tHTXO	Hold time w.r.t. to TX_TOH_CLK	60			ns

12.9 Automatic Protection Switching Output Interface Timing

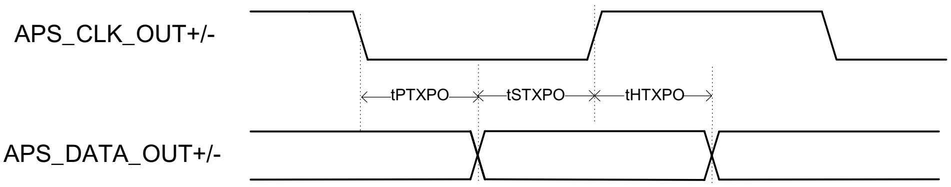


Table 69. Automatic Protection Switching Output Interface Timing

Label	Parameter	Min	Nom	Max	Units
	APS_CLK_OUT frequency		77.76		MHz
	APS_CLK_OUT duty cycle	45		55	%
t_{PTXPO}	Propagation delay w.r.t. APS_CLK_OUT	-1		1	ns
t_{STXPO}	Setup APS_DATA_OUT w.r.t. APS_CLK_OUT	2			ns
t_{HTXPO}	Hold APS_DATA_OUT w.r.t. APS_CLK_OUT	2			ns

12.10 Automatic Protection Switching Input InterfaceTiming

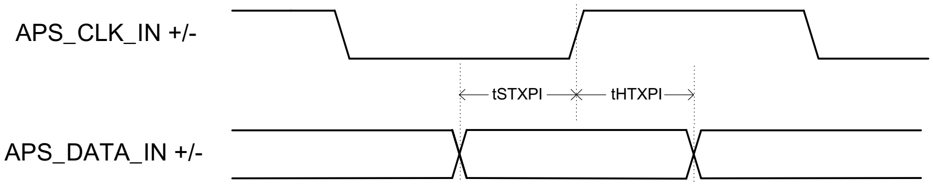


Table 70. Automatic Protection Switching Input InterfaceTiming

Label	Parameter	Min	Nom	Max	Units
	APS_CLK_IN frequency		77.76		MHz
	APS_CLK_IN duty cycle	45		55	%
tSTXPI	Setup time w.r.t. to APS_CLK_IN	1.5			ns
t_{HTXPI}	Hold time w.r.t. to APS_CLK_IN	0.5			ns

12.11 Microprocessor Interface Write Timing (Asynchronous Mode)

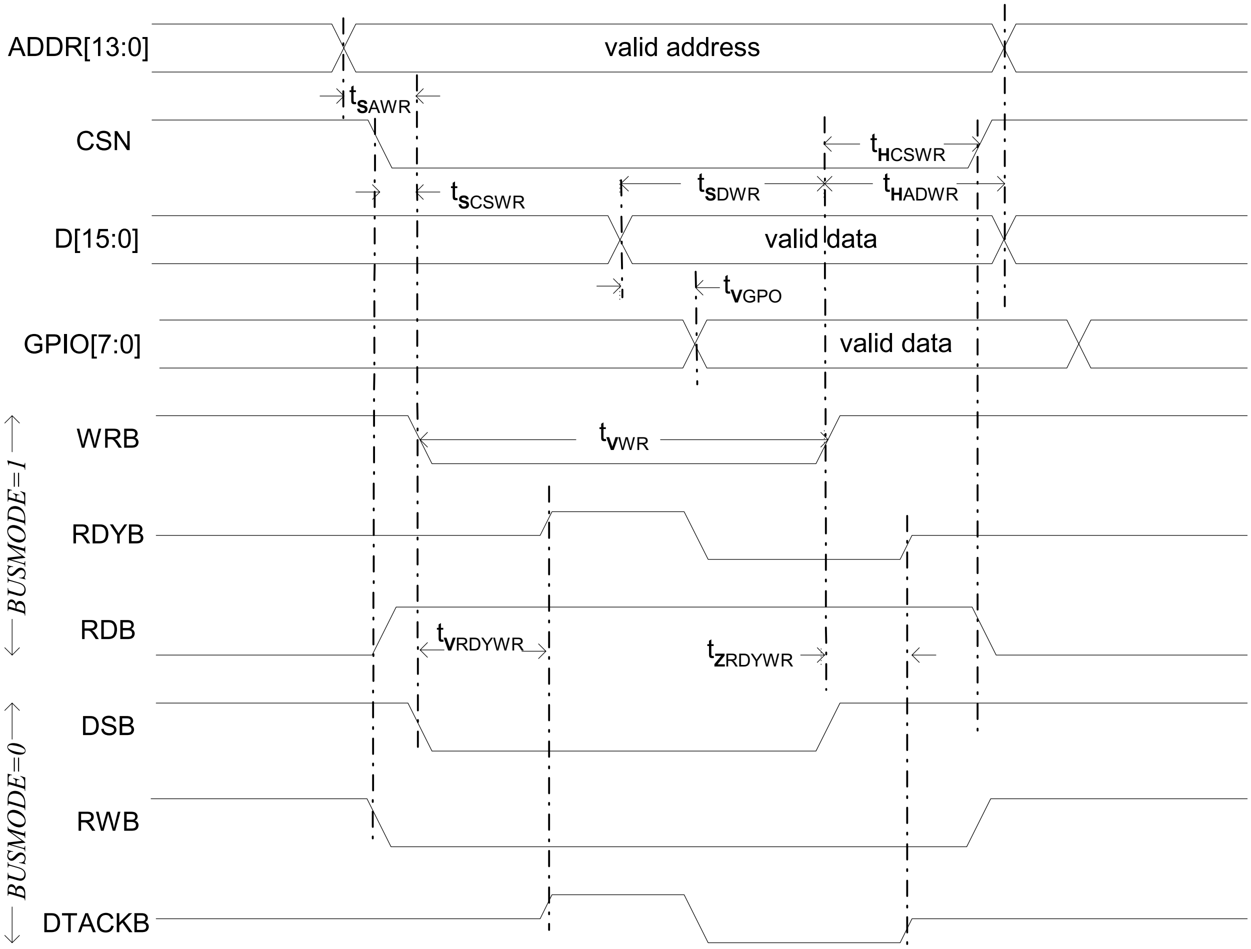


Table 71. Microprocessor Interface Write Timing (Asynchronous Mode)

Label	Parameter	Min	Max	Units
t_{SAWR}	Setup addr to write (WRB,DSB) assert	10		ns
t_{SCSWR}	Setup chip select (CSN,RWB) to write (WRB,DSB) assert	5		ns
t_{VWR}	Valid write (WRB,DSB) pulse width	50		ns
t_{SDWR}	Setup data to write (WRB,DSB) deassert	15		ns
t_{HADWR}	Hold addr/data from write (WRB,DSB) deassert	4		ns

Table 71. Microprocessor Interface Write Timing (Asynchronous Mode)

Label	Parameter	Min	Max	Units
t_{VRDYWR}	Valid RDYB,DTACKB from write (WRB,DSB) assert	0	15	ns
t_{ZRDYWR}	Tri-state RDYB,DTACKB from write (WRB,DSB) deassert	0	10	ns
t_{HCSWR}	Hold chip select (CSB,RWB) from write (WRB,DSB) deassert	0		ns
t_{VGPO}	GPIO output valid after D[7:0] written	0	8	ns

12.12 Microprocessor Interface Read Timing (Asynchronous Mode)

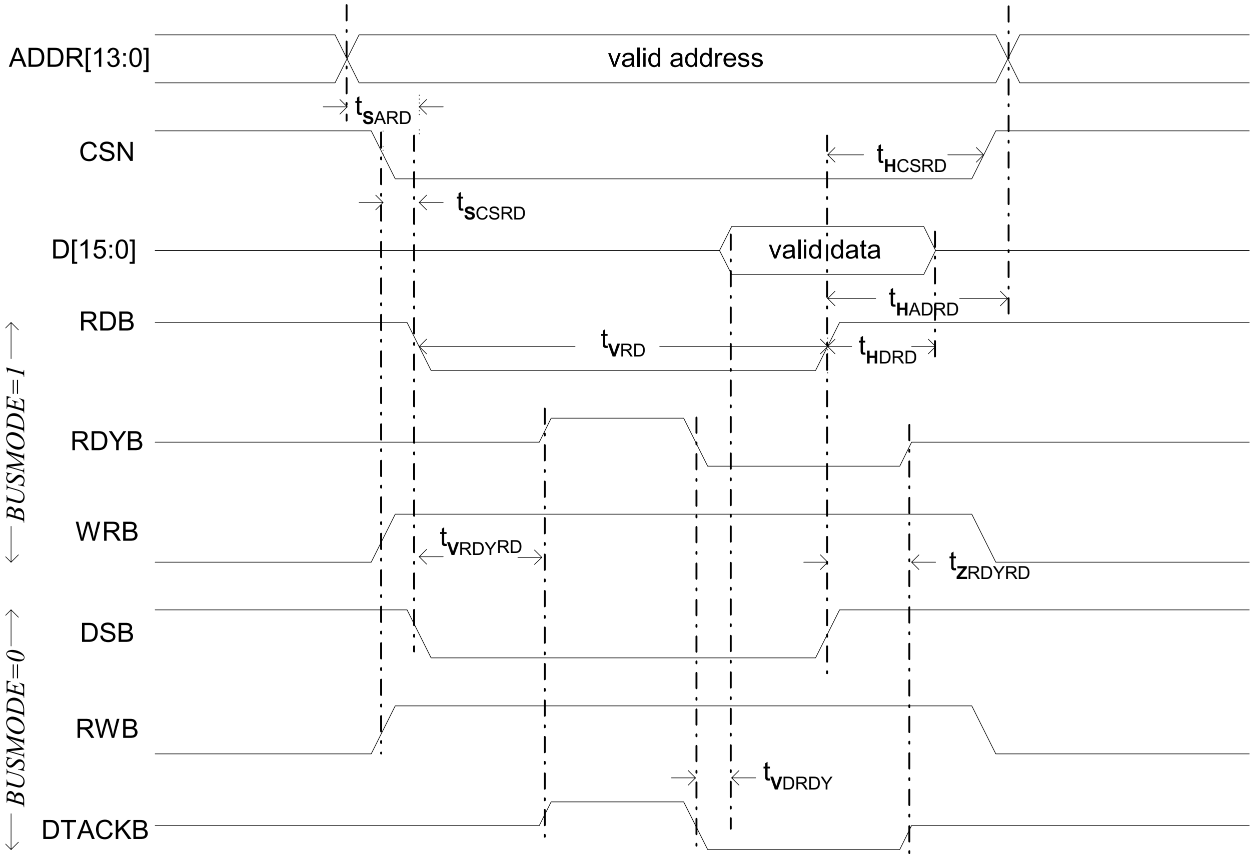


Table 72. Microprocessor Interface Read Timing (Asynchronous Mode)

Label	Parameter	Min	Max	Units
t_{SARD}	Setup addr to read (RDB,DSB) assert	10		ns
t_{SCSRD}	Setup chip select (CSN,RWB) to read (RDB,DSB) assert	5		ns
t_{HCSRD}	Hold chip select (CSN,RWB) from read (RDB,DSB) deassert	5		ns
t_{VRD}	Valid read (RDB,DSB) pulse width	50		ns
t_{VDRDY}	Valid data from RDYB/DTACKB assert		10	ns
t_{HADRD}	Hold addr from read (RDB,DSB) deassert	4		ns
t_{VRDYRD}	Valid RDYB,DTACKB from read (RDB,DSB) assert		15	ns
t_{ZRDYRD}	Tri-state RDYB,DTACKB from read (RDB,DSB) deassert	10		ns
t_{HDRD}	Hold data from read (RDB,DSB) deassert	15		ns

12.13 Microprocessor Interface Write Timing (Synchronous Mode)

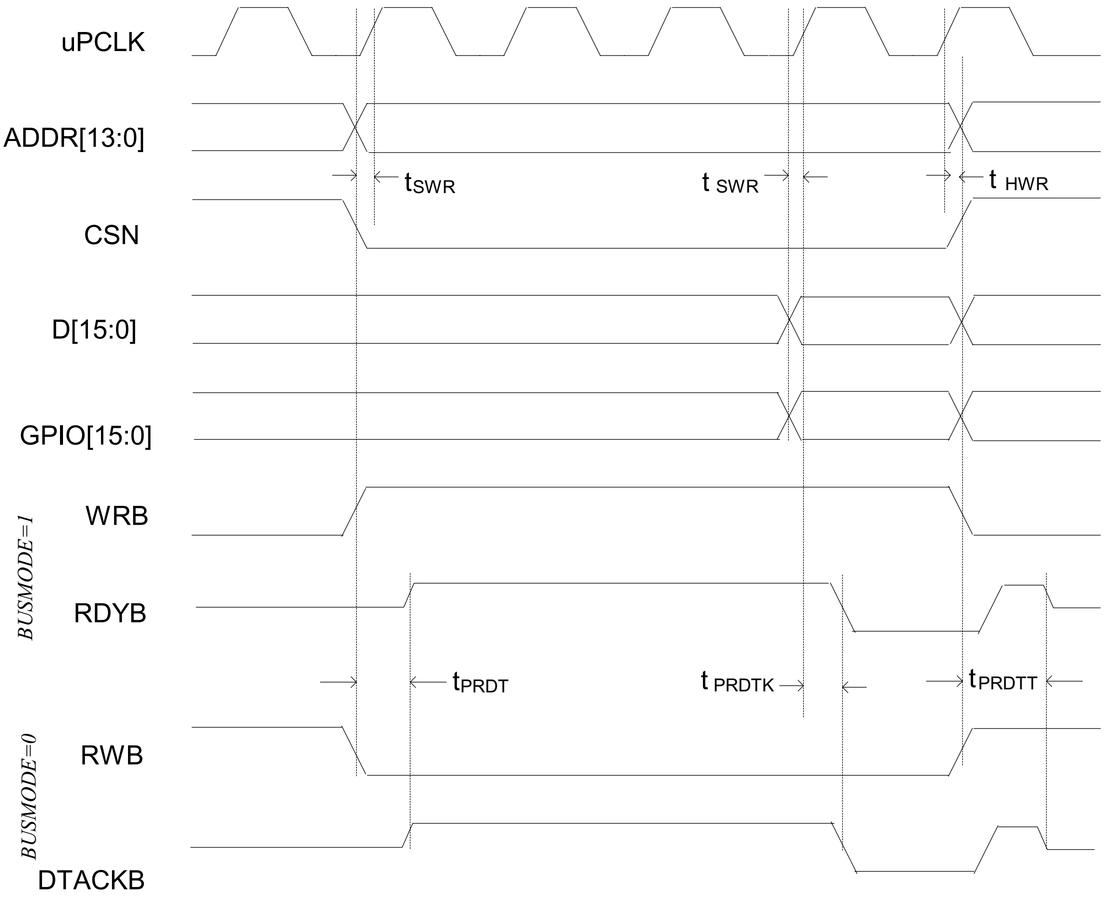


Table 73. Microprocessor Interface Write Timing (Synchronous Mode

Label	Parameter	Min	Nom	Max	Units
	uPCLK frequency	25		50	MHz
t_{SWR}	Setup time w.r.t. uPCLK	6			ns
t_{HWR}	Hold time w.r.t. uPCLK	2			ns
t_{PRDTK}	DTACKB/RDYB propagation delay w.r.t. uPCLK	5		11	ns

Table 73. Microprocessor Interface Write Timing (Synchronous Mode

Label	Parameter	Min	Nom	Max	Units
t_{PRDT}	Propagation delay w.r.t. CSN asserted			20	ns
t_{PRDTT}	Propagation delay w.r.t. CSN deasserted			20	ns

12.14 Microprocessor Interface Read Timing (Synchronous Mode)

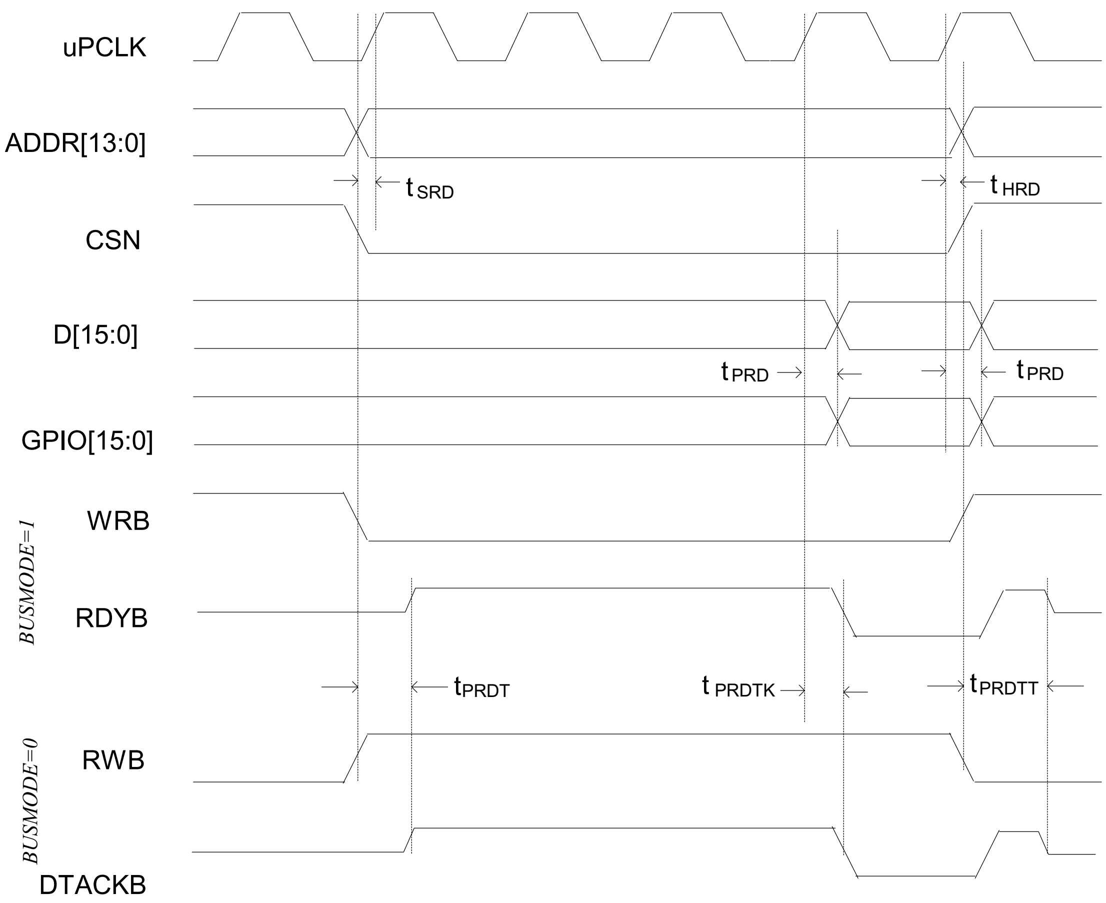


Table 74. Microprocessor Interface Read Timing (Synchronous Mode)

Label	Parameter	Min	Nom	Max	Units
	uPCLK frequency	25		50	MHz
t_{SRD}	Setup time w.r.t. uPCLK	6			ns
t_{HRD}	Hold time w.r.t. uPCLK	2			ns
t_{PRD}	valid data from uPCLK	5		11	ns
t_{PRDTK}	DTACKB/RDYB propagation delay w.r.t. uPCLK	5		11	ns
t_{PRDT}	Propagation delay w.r.t. CSN asserted			20	ns
t_{PRDTT}	Propagation delay w.r.t. CSN deasserted			20	ns

12.15 JTAG Interface Timing

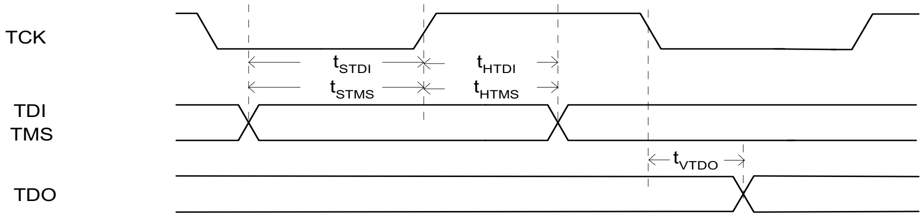


Table 75. JTAG Interface Timing

Label	Parameter	Min	Max	Units
TCK	TCK frequency		1	MHz
tSTDI	Setup TDI to TCK high	50		ns
t_{HTDI}	Hold TDI from TCK high	50		ns
tSTMS	Setup TMS to TCK high	50		ns
t_{HTMS}	Hold TMS from TCK high	50		ns
t_{HTDO}	TDO valid from TCK low	2	50	ns

13.0 DC Electrical Characteristics

13.1 Absolute Maximum Ratings

Table 76.

Parameter	Level	Unit s
Power Dissipation	8.55	W
Operating Temperature	-40 to 85	°C
Storage Temperature	-65 to 150	°C
Junction Temperature	125	°C
Voltage on any pin (3.3V LVTTL)	-0.6 to VDD_3.3 + 0.3[1]	V
Voltage on any pin (3.3V - tolerant 2.5V CMOS)	-0.6 to 3.9[1]	V
Voltage on any pin (LVPECL)	-0.2 to VDD_2.5 + 0.20	V
Static Discharge Protection	3000	V

Notes: 1 - The Minimum and Maximum voltage that can be applied without affecting device reliability. Minimum and Maximum values applies to under and overshoot only

13.2 Power Supply

Table 77.

Label	Parameter	Min	Nom	Max	Units
VDD_3.3	3.3 V Power Supply Voltage	3.0	3.3	3.6	V
ICC_3.3*	3.3 V Power Supply Current STS48C mode		130	160	mA
	3.3 V Power Supply Current 4xSTS12C mode		185	215	mA
	3.3 V Power Supply Current 16xSTS3C mode		420	480	mA
VDD_2.5	2.5 V Power Supply Voltage	2.3	2.5	2.7	V
ICC_2.5*	2.5 V Power Supply Current STS48C mode		1610	1870	mA
	2.5 V Power Supply Current 4xSTS12C mode		1650	1920	mA
	2.5 V Power Supply Current 16xSTS3C mode		1770	2120	mA

* Note: The ICC_3.3 and ICC_2.5 specifications are based on measured values with the maximum number of channels actively passing packets/cells per configuration mode. For STS48C and 4xSTS12C measurements, the TX/RX_TRIB_INH_x bits were set for unused tributaries to minimize power consumption (see section 4.4)

13.3 3.3V LVTTL I/O Specifications

Table 78.

Label	Parameter	Min	Max	Unit s	Comments
V_{OH}	High Level Output Voltage	2.4	VDD_3.3	V	VDD_3.3 = 3.0V I_{OH}= - 9 mA - 65 Ohm Driver I_{OH}= -12 mA - 50 Ohm Driver I_{OH}= - 40mA - 20 Ohm Driver
V_{OL}	Low Level Output Voltage		0.4	V	I_{OL}= 6 mA - 65 Ohm Driver I_{OL}= 8 mA - 50 Ohm Driver I_{OL}= 25 mA - 20 Ohm Driver
V_{IH}	High Level Input Voltage	2.0	VDD_3.3	V	
V_{IL}	Low Level Input Voltage	0.0	0.8	V	
I_{IH}	Input High Leakage Current		200	uA	V_{in} = $V_{IH\ (max)}$
I_{IL}	Input Low Leakage Current		0.0	uA	V_{in} = 0.0V

13.4 2.5V CMOS I/O Specifications

Table 79.

Label	Parameter	Min	Max	Unit s	Conditions
V_{OH}	High Level Output Voltage	2.2	VDD_2 .5	V	VDD_2.5 = 2.3V I_{OH}= - 1.5 mA - 65 Ohm Driver I_{OH}= - 2.0 mA - 50 Ohm Driver I_{OH}= - 5.0 mA - 20 Ohm Driver
V_{OL}	Low Level Output Voltage		0.4	V	I_{OL}= 7 mA - 65 Ohm Driver I_{OL}= 9 mA - 50 Ohm Driver I_{OL}= 26 mA - 20 Ohm Driver
V_{IH}	High Level Input Voltage	1.7	VDD_2 .5	V	
V_{IH}	High Level Input Voltage (3.3V - tolerant)	2.0	3.6	V	
V_{IL}	Low Level Input Voltage	0.0	0.7	V	

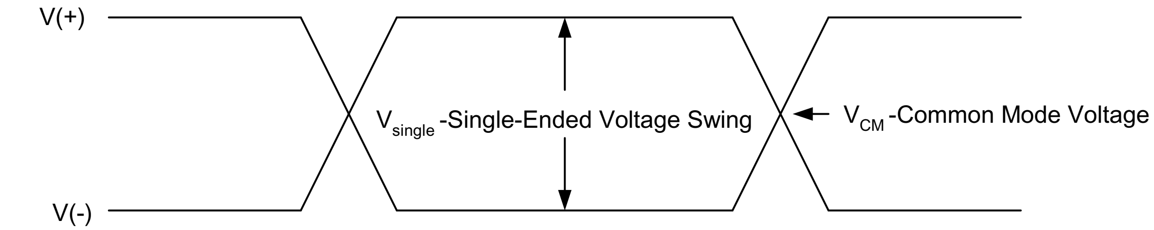
Table 79.

Label	Parameter	Min	Max	Unit s	Conditions
V_{IL}	Low Level Input Voltage (3.3V - tolerant)	0.0	0.8	V	
I_{IH}	Input High Leakage Current		200	uA	$V_{in} = V_{IH\ (max)}$
I_{IL}	Input Low Leakage Current		0.0	uA	V_{in} = 0.0V

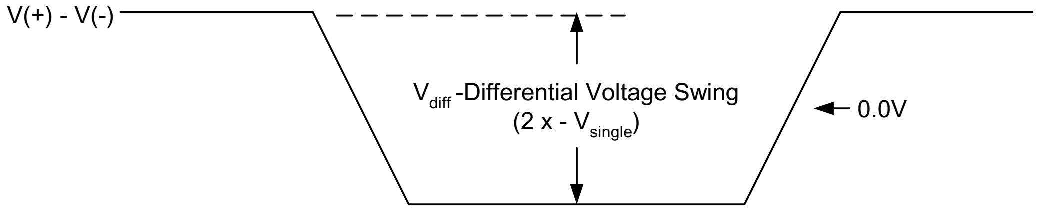
13.5 LVPECL I/O Specifications

Table 80.

Label	Parameter	Min	Nom	Max	Units	Comments
V_{IH}	Input Voltage High			VDD_2.5+0.4	V	
V_{IL}	Input Voltage Low	-0.4			V	
V_{ICM}	Input Common Mode Voltage	0		VDD_2.5	V	
Vindiff	Diff. Ended Delta input voltage	0.2			V	Peak-to-Peak Differential Voltage Swing. (See figure 1)
Vinsingle	Single Ended Delta input voltage	0.1			V	Peak-to-Peak Single-Ended Voltage Swing. (See figure 2)
R_I	Input Impedance	80	100	120	ohm	Differential Input Impedance. (Internal 100 Ohm resistor.)
V_{OH}	Output Voltage High	VDD_3.3-1.15		VDD_3.3-.70	V	Internally Biased LVPECL Output (Not open emitter)
V_{OL}	Output Voltage Low	VDD_3.3-1.950		VDD_3.3-1.465	V	Internally Biased LVPECL Output (Not open emitter)
ΔVoutdiff	Diff. Delta output voltage	1.18	1.62	2.12	V	Peak-to-Peak Differential Voltage Swing. (See figure 1)
ΔVoutsingle	Single Ended Delta output voltage	.59	0.81	1.06	V	Peak-to-Peak Single-Ended Voltage Swing. (See figure 2)
R_O	Output Impedance	50	60	70	ohm	Differential Output Impedance.


V(+)
V(-)
V_{single} -Single-Ended Voltage Swing
V_{CM} -Common Mode Voltage

Single-ended Voltage Swing



Differential Voltage Swing

Appendix A: User Notes

Chapter 1: User Notes Introduction

The purpose of providing ***Rhine Device User Notes*** is to communicate to device users the following information:

1. Device operation that varies from the device product specification.
2. Network and/or system application related issues that the user should be aware of when using the device.
3. Device operation that varies from applicable standards.

Rhine Device User Notes lists user notes by the following groups (by chapter)*:

- SONET/SDH
- ATM
- POS
- APS
- Miscellaneous

* Omission of one or more groups denotes no user note applies at this time.

Each group or chapter consists of the following sections.

- **Title** - The title of the user note as listed in the Table of Contents.
- **Issue** - A short statement regarding subject of description to follow.
- **Description** - A detailed description of issue.
- **Possible System Implication** - How the issue could affect or behave in a given network environment.
- **Solution or User Workaround** - Suggestions as to how the issue could be mitigated.

Chapter 2: SONET/SDH

2.1 RX_LOS_RDI_INH bit must be set when operating in SONET T/R Loopback mode

Issue:

SONET T/R loopback is functional only when RX_LOS_RDI_INH_[x] is set.

Description:

When operating the Rhine device in SONET T/R loopback, an LOS condition on the RX Line interface will result in a RX_PAIS condition. This RX_PI_PAIS condition will prevent flow of data in the RX path of the Rhine.

Possible System Implication:

None.

Solution or User Workaround:

This issue can be resolved by setting the RX_LOS_RDI_INH_[x] bit.

2.2 RX_LAIS error reported during LOS condition in OC3 mode

Issue:

When the Rhine’s line side interface is configured in STS-3/3C mode, an RX LOS condition will inadvertently cause the RX_LAIS and RX_LAIS_D status bits to be set.

Description:

The Rhine includes a status register to report an RX LAIS condition (via the RX_LAIS[x] bits) whenever the 3 LSBs of the K2 byte are detected as all ones. When operating the Rhine’s line side interface in STS-3C mode, it has been determined that in addition to a valid RX LAIS condition causing RX_LAIS[x]/ RX_LAIS_[x]_D being set, an RX LOS condition also results in the RX_LAIS[x]/RX_LAIS_[x]_D bits being set.

This issue does not occur in OC12 or OC48 line side modes.

Possible System Implication:

With the below workaround, the system level impact from this issue should be minimal. The only limitation would be the ability to independently detect an RX_LAIS event which occurs before an RX LOS condition (when there is not enough time for the RX_LAIS_[x]_D and RX_LOS_[x]_D registers to be read before the RX LOS condition occurs).

Solution or User Workaround:

When detecting an RX_LAIS[x]/RX_LAIS_[x]_D error, sample RX_LOS[x]/ RX_LOS_[x]_D to determine if the RX_LAIS[x]/RX_LAIS_[x]_D condition is valid or caused by an RX LOS condition.

2.3 RX LOC does not contribute to an LRDI in OC3 mode

Issue:

An RX LOC alarm condition on specific tributaries in OC3 mode is not correctly reported via LRDI.

Description:

Per the SONET/SDH specification, an LRDI is transmitted via the 3 LSBs of the TX K2 byte whenever a LOS, LOF, LAIS or LOC condition is detected on the RX Line interface. In the Rhine 4.0, it has been determined when the line side interface is configured in STS3/3C mode, an RX LOC condition on channels 6,7,8,10,11,12,14,15 and 16 does not result in an LRDI event being reported with the K2 byte of the outgoing TX frames.

This issue does not impact the other channels in OC3 mode and does not impact OC12 or OC48 modes.

Possible System Implication:

This issue will have minimal system level impact due to the fact that an LOC condition will result in an LOF condition which does correctly contribute to an LRDI event. Also, for many CRU devices (like the AMCC S3024) used to recover the line side RX clock in OC3 mode, an LOC condition from the line will result in the reference clock on the CRU still providing a valid clock to the Rhine. In this case, a line LOC condition is never detected in the first place.

Solution or User Workaround:

No workaround is required for this issue since an RX LOF condition (which will occur when a LOC occurs) can be used to contribute to an LRDI. However, if desired, software could monitor RX_LOC[x] or RX_LOC_[x]_D and manually set the 3 LSBs of the TX_K2 byte to 110b (via the TX_K2_[x]_[7:0] register) when an RX LOC condition occurs.

2.4 A change in RX_PI_PAIS causes a pulse on RX_PI_LOP

Issue:

A high-to-low transition of RX_PI_PAIS_x causes a pulse in RX_PI_LOP_x and RX_PI_LOP_x_D to be set.

Description:

When the RX_PI_PAIS real time status bit transitions from a high to low, a pulse occurs on the RX_PI_PAIS real time status bit that results in the RX_PI_LOP_D bit getting set. This issue only occurs in STS48C mode and on channels 13-16 in STS3C mode.

Possible System Implication:

This issue is not seen to impact a system application since it only occurs after an AIS condition, and is limited to a single pulse status bit error. In addition, the PP_PAIS and PP_LOP bits operate normally, and these bits are used to monitor the line for a AIS or LOP condition ahead of the PI block in Rhine (see Rhine product spec for more details on the RX PP and PI blocks).

Solution or User Workaround:

If a RX_PI_PAIS event is reported in Rhine, the RX_PI_LOP_D bit should be cleared after the PI_PAIS has been eliminated.

2.5 Channel 5 (in STS-12c mode) and Channel 8 (in STS-3c mode) do not properly respond to a NDF

Issue:

The Rhine’s RX pointer interpreter block does not correctly recognize a NDF on channel 5 in STS12C mode or channel 8 in STS3C mode. This results in two extra frames where data is not passed to the RX FIFO after an LAIS event.

Description:

Rhine is designed to not pass SPE data into its RX FIFO whenever its pointer interpreter is reporting an LAIS or LOP event. Per the SONET/SDH standards, when an AIS event is resolved, the transmitting node will send a NDF along with a new pointer to allow the receiving node to immediately establish a pointer (to clear the LOP condition caused by the AIS) and begin demapping data from the SPE.

In the Rhine device, the pointer interpreters on channel 5 in STS12C mode and channel 8 in STS3C mode do not recognize the NDF. Therefore, these channels must wait for three frames with the same pointer values before the pointer processor will clear the LOP condition. This results in two frames of data not entering the RX FIFO.

Possible System Implication:

The impact of this issue is seen as minimal because the issue only occurs after a LAIS condition. Since an LAIS condition results from a major line problem, the extra two frames without data entering the RX FIFO should have minor impact compared to the lost data during the LAIS event. It is expected that this issue would likely not be visible to most users.

Solution or User Workaround:

None required.

2.6 RX_SDH_PP[x] & RX_PP_SS_EN[x] bits are miswired

Issue:

The RX_SDH_PP[x] & RX_PP_SS_EN[x] bits (used to select whether the RX pointer processor will operate in SDH or SONET mode) are miswired for the 16 pointer processors in Rhine. As a result, in STS48, STS12/12C, and STS3C modes, the Rhine can not support mixed SDH and SONET processing independently on each channel.

Description:

The Rhine was designed to support the option of processing SDH or SONET frames independently for each of its 16 channels. As part of this support, the RX pointer processor blocks each were assigned separate provision bits (RX_SDH_PP[x] & RX_PP_SS_EN[x]), which configured each pointer processor for SONET mode when set to 0, or for SDH mode when set to 1.

It has been determined that the RX_SDH_PP[1] & RX_PP_SS_EN[1] provisioning bits on Rhine 4.0 are wired to all 16 RX pointer processors. With this mis-wiring error, RX_SDH_PP[1] & RX_PP_SS_EN[1] configures all 16 RX pointer processors to either SDH or SONET modes and RX_SDH_PP[2..16] and RX_PP_SS_EN[2..16] have no effect on RX PP operation.

Possible System Implication:

This issue prevents the Rhine from supporting mixed SDH and SONET processing on a per-channel basis. Instead, the entire chip is configured to process either SONET or SDH frames.

Solution or User Workaround:

None required.

2.7 Channels 7 and 15 (in 16xSTS-3c mode) received M1 error counters and SECE bits do not report errors when BIT_BLKCNT = 1

Issue:

When BIT_BLKCNT is set high, M1 errors received by the Rhine are not counted and the second event bit is not set on channels 7 and 15. When BIT_BLKCNT is set low (default), the M1 error counters and second events operate correctly for all channels.

Description:

Many of the updated SONET/SDH standards (including T1.231 and G.829) specify that B1 and B3 errors should be counted as block errors, while the B2 and M1 errors should be counted as bit errors. In previous versions of the Rhine device, a single BIT_BLKCNT provisioning register bit was implemented that resulted in all parity errors being counted and reported based on either block errors (i.e., number of frames having at least one parity error) or bit errors.

To support the updated standards, a change was made in Rhine 4.1 to hard-wire the B2/M1 counters to count/report bit level errors. With this change, the mature standards (such as GR253) could be met by forcing the B1/B3/G1 counters to be based on bit errors (setting BIT_BLKCNT =0) or to meet the updated standards by setting the BIT_BLKCNT =1 (for provisioning the B1/B3/G1 counters to count block errors).

It has been determined that the above mentioned change in Rhine 4.1 was only partially implemented. Specifically, the hard-wiring of the B2/M1 logic resulted in correct bit count monitoring of B2 on all channels and generation/insertion of the M1 byte on outgoing TX frames on all channels. However, for the RX M1 error counters and SECE error status bits, the change was not implemented correctly for all channels. In particular, tributaries 7 and 15 were not correctly accounted for with the change. As a result, channels 7 and 15 do not operate correctly (i.e., no received M1 errors are counted or reported) unless the BIT_BLKCNT = 0 and all parity monitoring/reporting is based on bit errors.

Since only channels 7 and 15 are impacted, this issue is limited to OC3c line configuration mode on Rhine. OC12 and OC48 line interface modes can be configured to be fully compliant with all applicable standards.

Possible System Implication:

Since protection switching, signal degrade, and other operational decisions are not typically made based on received M1 error counts, the impact to the system is only in terms of compliance and performance monitoring. With BIT_BLKCNT = 0, Rhine 4.1 is fully compliant with the GR253 SONET standard. With BIT_BLKCNT=1, Rhine 4.1 complies with the updated ANSI and ITU standards for B1, B2, B3 monitoring and G1 reporting. However, it does not comply with updated standards in terms of with the M1 error-counting on frames received on channels 7 and 15 in OC3C mode.

Solution or User Workaround:

There is no workaround for this issue.

2.8 TX P-RDI is not removed immediately following clearing of received P_AIS condition

Issue:

If PRDI_AUTO is set, Rhine will transmit P_RDI for a minimum of 20 frames after receipt of a P_AIS condition. After this condition is cleared, it can take up to 20 additional frames for P_RDI to stop transmitting.

Description:

This behavior does not exactly match the datasheet text.

Possible System Implication:

The number of P_RDI frames is as follows:

- 20 frames of P_RDI if P-AIS is received for less than 20 frames.
- 41 frames of P_RDI if P-AIS is received for between 21 and 41 frames.
- 62 frames of P_RDI if P-AIS is received for between 42 and 62 frames.
- and so on....

Solution or User Workaround:

None. This user note is for informational purposes only.

Chapter 3: ATM

3.1 RX_ATM_IDLE_CAP_E is not reliable

Issue:

RX_ATM_IDLE_CAP_E is not always set when an IDLE cell has been captured.

Description:

For test purposes, the RHINE provides the capability for the user to monitor the contents of the data bytes of the idle cells. When RX_ATM_IDLE_CAP is written from 0 to 1, the RHINE captures the next ATM idle cell it detects in the indicated tributary and payload byte position. The value of the captured idle cell is written into RX_ATM_IDLE_DATA[7:0]; RX_ATM_IDLE_CAP_E is written to 1 to indicate the capture is completed.

In Rhine, the 0 to 1 transition of the RX_ATM_IDLE_CAP register bit does result in the Rhine capturing the next ATM idle cell it detects. However, when the capture occurs, the RX_ATM_IDLE_CAP_E event bit is not always set.

Possible System Implication:

As a test mode feature, there should not be any system-level impact from this issue in normal operation.

Solution or User Workaround:

When using the ATM IDLE Cell capture mode in Rhine, do not rely on the RX_ATM_IDLE_CAP_E event bit to indicate that the IDLE cell has been captured. Instead, change the value of the IDLE cell payload contents prior to changing RX_ATM_IDLE_CAP from a 0 to 1 and poll the RX_ATM_IDLE_DATA[7:0] to determine when a IDLE cell has been captured.

3.2 In STS48C mode, RX ATM cell counter may not be incremented if RX FIFO is almost full

Issue:

In STS48C mode, RX ATM cell counter may not be incremented if RX FIFO is almost full.

Description:

In Rhine, cells are counted properly until the RX FIFO comes within one cell of overflowing the RX FIFO in STS48c mode. When the 16 cell RX FIFO contains 15 cells, the next received cell is not always counted, although it is not dropped.

Possible System Implication:

None, since with the correct FIFO full/emtpy levels configured (i.e. RX_FIFOFULL_TPA and RX_FIFOEMPTY_TPA both = 0), the RX FIFO should never be close to full in ATM mode.

Solution or User Workaround:

Set RX_FIFOFULL_TPA and RX_FIFOEMPTY_TPA both = 0 per Rhine product specification.

3.3 TX_ATM_CELL_CNT counter for channel 16 is only 16 bits wide

Issue:

TX_ATM_CELL_CNT_16[22:16] miswired for channel 16. Therefore only the [15:0] bits are usable in this counter. Only channel 16 in 16xOC-3c mode is affected.

Description:

Each of the 16 channels of Rhine have a transmitted ATM cell counter. Some bits of the register that reports the count for channel number 16 were miswired to a copy of the same bits from the channel 15 register. The lower bits of the register (TX_ATM_CELL_CNT_16[15:0]) work fine. Also note that TX_ATM_CELL_CNT_15[22:0] operation is not affected in any way by this bug.

Possible System Implication:

The 23 bit size was chosen to support STS-48c on channel 1 and was reused in slower tributaries for simplicity. A monitoring interval of about 1 second was assumed. Even though channel 16 only runs at STS-3c rate, 16 bits is not sufficient to count a full second of full rate cell traffic.

Solution or User Workaround:

User should either monitor this counter on a faster interval, (5 times per second will suffice) or if this is not acceptable, use the transmit cell counter in the nP or SAR device.

Chapter 4: POS

4.1 An LOS or LOF condition in the Rhine can result in an infinitely large packet transmission over the RX Flexbus^{TM} interface in POS and direct map modes

Issue:

When Rhine is configured in POS or direct mapped modes, an LOS or LOF condition can result in a continuous stream of bytes entering the RX FIFO. This may lead to the transmission of an infinitely large packet over the system interface until the LOS/LOF condition is removed.

Description:

During a LOS or LOF condition, there is no mechanism in the Rhine to disable the demapping of data into the RX FIFO. As a result, when a LOS or LOF condition occurs, the Rhine may transmit multiple invalid packets or an infinitely large packet over the RX Flexbus^{TM} interface until the Rhine has once again recovered SONET/SDH frame.

Possible System Implication:

This issue may result in a continuous transmission of invalid data across the system interface (until the LOF condition is removed).

The issue can be prevented using the following workarounds.

Solution or User Workaround:

There are two possible options for preventing an infinitely large packet from being transferred from the Rhine during an LOS/LOF condition.

1. In POS mode, the PMAX packet size abort feature can be used to abort packets above a certain size. This packet abort will result in the assertion of RXERR and RXEOP on the RX Flexbus^{TM} interface while dropping all bytes arriving in the RX FIFO after the packet has exceeded the max packet size set in RX_POS_PMAX_[x] register.
2. In direct map mode (where PMAX is not applicable), an abort timer or byte counter can be implemented in the link layer device to abort a transfer after a specific time period or after counting a specific number of bytes. This mechanism would effectively delete (via a pointer adjustment) the invalid bytes after the timer/counter limit has been reached. The link layer device would then drop all additional bytes it receives from the Rhine port address that sent the problem packet until a valid start of frame sequence is received.

4.2 TX_POS_FIFOUNDER_E bit is not reliable

Issue:

The TX_POS_FIFOUNDER_E status bit is not set every time a TX FIFO underflow event occurs.

Description:

The Rhine uses two registers to report a TX FIFO underflow condition. These registers are labeled in the register map as TX_POS_FIFOUNDER_E and TX_POS_FIFOUNDER_SECE. It has been determined that the TX_POS_FIFOUNDER_E is not always set when the TX FIFO underflows. However, the TX_POS_FIFOUNDER_SECE (and its associated TX_POS_FIFOUNDER_CNT count register) are always set when a underflow occurs.

Possible System Implication:

None. The TX_POS_FIFOUNDER_SECE can be used to correctly detect a TX FIFO underflow event.

Solution or User Workaround:

Use the TX_POS_FIFOUNDER_SECE to report a TX FIFO underflow event. It should be noted that the LATCH_CNT bit must be used to latch the underflow bit into the _SECE register (see Rhine Product Specification for more information on LATCH_CNT and the _SECE bits).

4.3 RX_EOP may be asserted for 2 RX_CLK cycles

Issue:

The RX_EOP signal may pulse for 2 RX_CLK cycles instead of 1 cycle for some RX packets.

Description:

The Rhine asserts a two clk cycles RX_EOP for every RX packet of a specific size (including 36, 44, 50, 58, 66, 68, 74, 82, etc).   The packets transmitted with the two cycle RX_EOP are not marked as errored, and the other RX output signals (including RX_DV and RX_CLAV) are asserted/deasserted at the correct times.

Possible System Implication:

SInce RX_DV is only asserted during the first cycle that RX_EOP is asserted and the timing on the other RX system interface outputs is correct, there should be no system impact if the link layer device uses RX_DV to qualify the RX_EOP signal.

Solution or User Workaround:

Use RX_DV to qualify all RX system interface outputs (including RX_EOP). The outputs should only be considered valid when RX_DV is asserted.

4.4 Rhine maps corrupted data after a TX FIFO underflow in STS-48c mode

Issue:

When the Rhine TX FIFO underflows, the Rhine may map interpacket 7E flags, 7D7E abort sequences, or underflow byte mode character until the next packet arrives in the TX FIFO.

Description:

In STS-48c mode, after the TX side has experienced an FIFO underflow, the Rhine HDLC processor correctly aborts the packet. However, instead of correctly inserting 7E flags until a subsequent packet is available to be mapped from the TX FIFO, the HDLC processor outputs:

1. A mix of 7E interpacket fill bytes, 7D 7E abort sequence bytes and the underflow mode byte (50h default) if TX_POS_FCSABORT_ENB=0 (default).
2. A series of very short corrupt packets with bad FCS if TX_POS_FCSABORT_ENB=0

A transmission of a good packet into the TX FIFO or a TX state reset eliminates this condition.

This behavior is not seen in STS-3c and STS-12c mode

Possible System Implication:

This issue is not seen as a problem in an end application for three reasons:

1. In both cases above the corrupted packets will fail FCS at the receiving node and will be dropped.
2. TX FIFO underflows are not expected to occur within a properly designed system.
3. If an underflow occurs, the above issue will be resolved when the next packet arrives in the TX FIFO. No other user intervention is required.

Solution or User Workaround:

None required. The TX system interface should be designed to use TX_CLAV/PDA so that underflow conditions will not occur.

4.5 RX_POS_PABORT_ERRCNT does not increment short packets

Issue:

When the RX HDLC processor receives an aborted packet (packet ending with the 7d7e abort sequence) that is less than six bytes in length, the RX_POS_PABORT_ERRCNT will not be incremented and the associated _SECE bit will not get set.

Description:

When a packet is aborted by the transmitting node using the 7d7e abort sequence, the Rhine’s RX HDLC processor will correctly abort the packet. However, it does not correctly increment the RX_POS_PABORT_ERRCNT and does not set the RX_POS_PABORT_SECE bit to reflect that an aborted packet was received.

Possible System Implication:

Based on the size of the packet and the fact that the packet is correctly aborted, this issue should have no system impact.

Solution or User Workaround:

None. This user note is for informational purposes only.

4.6 Packets aborted by inverting FCS are counted in TX packet count

Issue:

When a TX packet is aborted using the inverted FCS option, it is counted in the TX packet counter.

Description:

The Rhine provides two methods of aborting a TX packet. With TX_POS_FCSABORT_ENB=0 (default), the packet will be terminated and a 7d7e abort sequence will be appended to the packet. With TX_POS_FCSABORT_ENB=1, the packet will be terminated and an inverted FCS byte field will be appended to the packet. In either cause the RX will see the packet as errored.

When the packet in aborted in Rhine using the FCS abort method, the FCS errored packets are still counted by as good TX packets in the TX_POS_PKT_CNT register.

Possible System Implication:

This issue should have not system impact for three reasons:

1. The packet counter is typically only used in debug.
2. A packet abort should not be a common event in a well-designed system.
3. The packet abort contribution (LLPKT error or TX FIFO error) is still

reported via counters and SECE bits.

4. With a fix to user note 4.9, the default 7e7d abort sequence can be used instead of the FCS invert method to abort packets and eliminate the impact of this issue.

Solution or User Workaround:

If using the FCS invert method to abort packets, the LLPKT and TX FIFO counters can be used to determine if any of the TX POS packets were aborted. Alternatively, the default 7d7e abort method can be used to eliminate this issue entirely.

4.7 RX PMAX abort may operate incorrectly in STS3C/STS12C modes

Issue:

If an RX packet exceeds the RX PMAX byte count value coincident with a data byte that requires transparency processing, the HDLC processor will not correctly abort drop all remaining bytes from the aborted packet. Instead, it will treat the remaining bytes from the packet that enter the TX HDLC processor as a new packet.

Description:

The Rhine includes a maximum packet size abort feature which, when enabled, is designed to abort a packet which is more than the number of bytes provisioned in the RX_PMAX_POS_[15:0] register. When the abort occurs, the remaining bytes in the packet (up to the next 7E byte demapped from the SPE) are supposed to be dropped before they enter the RX FIFO.

In Rhine 3.0 and Rhine 4.0 it has been determined that if the byte that causes a packet to exceed the PMAX register value requires transparency processing, then instead of dropping the remaining bytes in the packet, the HDLC processor will treat the remaining bytes as a new packet. The result will be either a packet entering the RX FIFO that will fail FCS or, if the remaining part of the FIFO exceeds the PMAX register value again, then the packet will be aborted as a PMAX violation.

Possible System Implication:

Since the PMAX value should be set at a level that is not typically exceeded by expected RX packets and the issue only occurs if the byte causing the PMAX error requires transparency processing, this issue does not pose a problem for most applications. Also, for other user note items that rely on PMAX (e.g. User Note 4.1), this issue does not pose a problem since the PMAX will correctly ensure that packets larger than the PMAX register value do not get passed across the system interface.

Solution or User Workaround:

None required.

4.8 Packets failing PMIN are counted as FCS error

Issue:

When the RX PMIN packet abort feature is enabled, RX packets that are less than the PMIN register value are correctly aborted. However, they are counted as both PMIN errors and FCS errors.

Description:

The Rhine includes an RX PMIN abort feature which, when enabled, aborts received packets less than a value specified in the RX_POS_PMIN_[3:0] register. When a packet is aborted due to a RX PMIN violation, it is reported using the RX_POS_PMIN_SECE status register and counted in the RX_POS_PMIN_CNT register.

In all versions of Rhine, it has been determined that a packet failing PMIN is correctly aborted and reported in the PMIN status register/counter. However, it is also reported as a RX FCS error via the RX_POS_FCS_ERR_SECE status bit and RX_POS_FCS_ERR_CNT register.

Possible System Implication:

There should be very little impact due to this problem since a PMIN abort does occur correctly and a PMIN event occurs at a very low rate.

Solution or User Workaround:

None required.

4.9 When addr/cntrl drop is active, packets not matching addr/cntrl field are dropped due to FCS mismatch

Issue:

When RX addr/cntrl drop is active, packets not matching addr/cntrl field are dropped due to FCS mismatch.

Description:

The Rhine includes an RX addr/cntrl drop feature which, when enabled, results in the add/cntrl bytes being dropped if they match the values programmed in the RX_POS_ADDRESS[7:0] and RX_POS_CONTROL[7:0] registers. This feature is used only when the addr/cntrl bytes are fixed and are expected on every received packet.

In all versions of Rhine, it has been determined that if the addr/cntrl drop feature is enabled, a packet that is received with addr/cntrl bytes that do not match the addr/cntrl register will be dropped (not just the addr/control bytes, but the entire packet) due to an RX FCS mismatch error.

Possible System Implication:

This issue should be insignificant in an end application since an addr/cntrl mismatch would represent a corrupted packet anyway (if addr/cntrl drop feature as used, addr/cntrl would be assumed as fixed and should match). If addr/cntrl is expected to change dynamically, then user would not use addr/cntrl drop, but would always pass the addr/cntrl and check it at the link layer.

Solution or User Workaround:

None Required.

4.10 If RX_POS_ADRCTL_DROP_INH_x = 1, then RX_POS_ADRCTL_INVALID_x will always be set to 1

Issue:

If RX Address/Control byte drop is inhibited (bytes are passed through to the network processor) for a STS-12c or STS-3c tributary, then the RX_POS_ADRCTL_INVALID_x bit for that tributary will always be set.

Description:

Rhine includes a special mode in which the received HDLC Address and Control bytes may be passed over the Flexbus3^{TM} interface to be processed in the link layer/network processor device. In this mode we did not intend for the bytes to also be checked for validity by Rhine as well as in the nP. This user note was created to supplement the datasheet and prevent any user confusion.

Possible System Implication:

Addr/Ctrl validity checking works as expected when Rhine is configured to drop these bytes. This is the normal mode of operation. Users need to be aware that this error bit will be set if they chose not to drop the bytes.

Solution or User Workaround:

None required. Users wishing to passthrough the bytes and check them at higher layers should set the RX_POS_ADRCTL_INVALID_SUMD_MASK bit to avoid any interrupts being generated.

4.11 RX_POS_PMAX feature is fully functional but RX_POS_PMAX_ERRCNT counting is not completely accurate in STS-48c mode

Issue:

RX_POS_PMAX feature is operational and useful for its intended purpose in Rhine 4.1. It does however have some idiosyncrasies that the user should be made aware of.

Description:

Rhine includes a RX_POS_PMAX register to enable aborting packets above a certain size. This feature was broken in 4.0 and a metal change was done to fix it. This was successful but had some side effects relating to the PMAX error reporting:

1. RX_POS_PMAX must be set to at least 8 bytes.
2. In STS-48c mode only, packets greater that RX_POS_PMAX + 10 bytes will be counted both in RX_POS_PMAX_ERRCNT_x and in RX_POS_PABORT_ERRCNT_x.

   In STS-48c mode only, RX_POS_PMAX_ERRCNT and RX_POS_PMAX_ERR_SECE may trigger at a higher byte count than is set in RX_POS_PMAX, according to the following formula:

   (n = sequence of 4, 8, 12, 16, ..)

   For RX_POS_PMAX = 10 + n , the counters work fine in all cases.

   For RX_POS_PMAX = 11 + n and 13 + n, the counters only update on packet byte count greater than RX_POS_PMAX + 1.

   For RX_POS_PMAX = 12 + n, the counters only update on packet byte count greater than RX_POS_PMAX + 2.

   Despite these counter behaviors, packets are always aborted on the Flexbus3^{TM} interface when their size is >= RX_POS_PMAX.

Possible System Implication:

There is obviously some potential for inaccurate counting of PMAX errors for certain PMAX settings in STS-48c mode. However packets can still be aborted reliably. The primary application of this PMAX feature is to block infinite or very large packets from reaching higher layers under SONET error conditions during which packet counts are not applicable or reliable anyway.

Solution or User Workaround:

When choosing a value for PMAX, user should consider the above limitations if exact counts of PMAX errors are a priority. Otherwise this user note is for informational purposes only.

4.12 Extra Flags may be inserted on transmit if Address/Control bytes or FCS bytes are not inserted in POS OC-48c mode

Issue:

Under certain conditions, the STS-48c TX HDLC processor may insert extra 7e flags between packets in the SONET Payload Envelope, if both Address/ Control and FCS bytes are not inserted by the Rhine. No data corruption will occur but the extra bytes between packets limit the ability to run TX at full SONET payload rate for long periods.

Description:

“Line rate” or “Full bandwidth” in a Packet over SONET application is defined as the ability to receive and transmit packets of any size with only a single 7e flag between them.

The Rhine TX HDLC processor can run continuously with only one 7e flag between packets for all valid packet sizes if TX_POS_ADRCTL_INS = 1 and TX_FCS_INH = 0. This is the desired mode of operation for most systems as it removes the responsibility for processing and transmitting Address/Control, and bytes from the higher (nP) layers.

However, we have provided the option to accept these bytes from the higher layers for certain very specialized applications. When this is done, packets of certain sizes and alignments can cause the HDLC processor to add flag bytes before and after packets.

As an example, for random packet sizes between 40 and 1518 bytes, with TX_POS_ADRCTL_INS = 1 and TX_FCS_INH = 1 (FCS not inserted), and with sustained single flag gaps, the throughput was found to be 99.8% of full line rate.

There are no bandwidth restrictions in channelized OC-48 or in any slower line or payload mode for any combination of HDLC processor options. There is also no bandwidth restriction on any RX mode, including OC-48c.

Possible System Implication:

Short peaks of “line rate” traffic are a corner case for normal system operation and can easily be handled by Rhine. Eventually, the extra flags will cause the TX FIFO to fill up and the Rhine will assert the TX_CLAV_PDA pin to backpressure the nP device. This is only likely to occur in a benchmarking scenario, and only in the special HDLC modes mentioned.

Solution or User Workaround:

None. For benchmarking in OC-48c POS mode, it is strongly recommended that the user allow Rhine to insert AC and FCS bytes on TX, even if the user does not intend to do so for normal operation. If this is done sustained bandwidth will be 100%.

Chapter 5: APS

5.1 APS_MISALIGN_ERR_E is always set

Issue:

APS_MISALIGN_ERR_E is always set despite good data passing through the APS interface.

Description:

The APS_MISALIGN_ERR_E error bit is always set, even when good APS traffic is being passed over the APS interface.

Possible System Implication:

Users cannot rely on APS_MISALIGN_ERR_E bit to detect APS errors. However, the APS_B1ERR_SECE error status bit can be used to monitor for RX APS data-alignment errors.

Solution or User Workaround:

Monitor the APS_B1ERR_SECE error status bit (reg addr 3600h, bit 7) as a means of detecting APS data-misalignment errors.

Chapter 6: Miscellaneous

6.1 *_SECE register bits values and count registers values are not reliable in OC3 mode when CNT_SEC_EN is set

Issue:

When the Line side interface of the Rhine 4.0 is configured for STS3/3C mode and the CNT_SEC_EN bit is set, the SECE register and count register bits values are not reliable.

Description:

For performance monitoring purposes, the Rhine provides “second event” bits (designated with a *_SECE suffix), and performance counters that can be monitored for occurrence each second. This allows a controller to accumulate the number of seconds that contain a particular event; for example, the number of seconds that at least 1 error was detected in a received signal by monitoring the B1 bytes. For this purpose, Rhine can internally create a 1 Hz signal, SEC_EVENT (enabled using the CNT_SEC_EN bit). Alternatively, the timing for “second events” can be controlled by the LATCH_CNT register.

When RHINE detects a rising edge of SEC_EVENT (and CNT_SEC_EN = 1) or when LATCH_CNT, in register 0x2000, is written from a 0 to a 1 (and CNT_SEC_EN = 0), it produces a pulse on an internal signal, LATCH_EVENT. When a pulse occurs on LATCH_EVENT, the *_SECE register bits are set if their corresponding internal-current, second-event, monitoring-bits are active. Like delta and event bits, the *_SECE bits are not cleared until they are written to 1 by the controller. In addition to updating the *_SECE bits, the internal LATCH_EVENT pulse also latches the current value of the various performance counters in Rhine into their associated count registers so that they can be read.

It has been determined when channel 1 of the Rhine 4.0 Line side interface is configured in STS3/3C mode, all of the *_SECE register bits and count register values are not reliable.

This issue does not impact STS12, ST48 or mixed STS12/STS3 Line side modes if quadrant 1 is configured for STS12 operation. In addition, the issue does not occur in STS3 mode when using LATCH_CNT to update the SECE bits (with CNT_SEC_EN=0).

Possible System Implication:

This issue prevents the use of the internal 1hz timing pulse for monitoring whether a SECE condition has occurred within a 1 second period and as a mechanism for updating SECE bits and count register values. Therefore, when channel 1 of the Rhine is configured for STS3 operation, the only option for correct error detection within the Rhine is to set the CNT_SEC_EN =0 and use the LATCH_CNT bit to update the SECE bits and count registers.

Solution or User Workaround:

When using the Rhine in OC3 mode, use the LATCH_CNT register bit (written from a 0 to a 1) to periodically check for an error condition and update the count registers.

6.2 TX_CLK_OUT_INH[1:4] register is writable but not readable

Issue:

The TX_CLK_OUT_INH register can not be read by the microprocessor.

Description:

This register can be written and operates as documented when values are written to it. However, uP reads do not return the actual contents of the register.

Possible System Implication:

None, provided software is aware of the limitation on read back.

Solution or User Workaround:

None.

15.0 Index of Registers

KEY:

In this index, primary entries (outdented) are register descriptions that appear in the main body of the product specification. Secondary (indented) registers in this index refer to register locations (hexidecimal addresses) in the Memory Map tables.

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

Index of Registers

16.0 Ordering and Packaging Information

Ordering information for Rhine: S4804CBI41.

Table 81. Part Number decoding of S4804CBI41

RHINE: S4804CBI41	Definition
S	S = Standard (Integrated Circuit)
4804	Rhine's model number

CBI	Packaging information:
	C = Ceramic
	B = Ball (Grid Array)
	I = Industrial (Temperature Grade)

41	Revision number of the Rhine device: 4.1

The information contained in this document is ***Production Release Information*** and is about a product in its fully tested and characterized phase. All features described herein are supported. Contact AMCC for updates to this document and the latest product status. All trademarks contained in this specification are the property of their respective owners.

AMCC
200 Minuteman Road
Andover, MA 01810
Phone (978) 247-8000, Fax (978) 623-0024
Internet: http://www.amcc.com