Advance Data Sheet
August 1998

microelectronics group

Lucent Technologies
Bell Labs Innovations

LU5M31
Gigabit Ethernet Media Access Controller (MAC)

Overview

The LU5M31 is a single-port 1 Gbit/s MAC that incorporates physical coding sublayer (PCS) functionality. The LU5M31 is intended to enhance 10/100 Mbits/s Ethernet frame switching, multiple port bridging, and routing applications by providing a 1 Gbit/s uplink port to a high-speed Ethernet backbone. The LU5M31 operates in full-duplex mode and is intended to be used with a physical layer device (PHY) that supports a 10b encoded data interface at 125 MHz.

PCS support is provided including an 8b/10b encoder/decoder for direct connection to a PHY. The PCS encodes 8-bit transmit data into 10-bit data codes required by the transceiver, and decodes 10-bit data codes received by the transceiver into 8-bit data. In addition, the PCS manages the autonegotiation and link synchronization functions required by *IEEE* 802.3z.

Full-duplex GMII support is provided for an interconnection between the LU5M31 and the plug. It provides an independent eight-bit-wide transmit and receive data paths, and has stations management to and from the PHY device. This GMII interface is based on the *IEEE* 802.32 clause 35 standard.

The LU5M31 allows single-clock cycle data transfers directly to and from its internal transmit and receive FIFOs. The receive FIFO uses programmable watermarks to initiate flow control in order to prevent FIFO overflow.

The LU5M31 provides configuration and management support through CPU interface accessing on-chip registers and event counters. The event counters maintained by the LU5M31 are derived from the RMON MIB and provide enough raw data to implement the Ethernet statistics group. Other groups within RMON can also use the data collected by the event counters.

The LU5M31 comprises four main functional blocks:

- The PCS block provides support for autonegotiation, internal loopback, random jitter testing, and encoding and decoding of data transferred to or from a GMAC block across an internal GMII compliant interface.
- The GMAC block operates between the host and the PCS and implements all the full-duplex functions of the media access protocol. The GMAC block is subdivided into a MAC layer (managing data transfer), and a full-duplex MAC control sublayer which handles all control functions required by the MAC sublayer.
- The host interface provides a high-speed interface between the LU5M31 and the host to allow buffering for receive and transmit data.
- The configuration and management block provides the host with synchronous CPU access to both registers and event counters, allowing the host system to configure and monitor the device independent of the data path.

The LU5M31 is a full-scan device that is *IEEE* 1149.1 (JTAG) compliant providing support for built-in self-test (BIST) for automated memory testing.

The device is designed in 0.35 µm CMOS and is packaged in a 208-pin SQFP.

Figure 1 illustrates the LU5M31 architecture.

Features

- 1 Gbit/s full-duplex Ethernet MAC integrated with separate transmit and receive port FIFOs:
  - — Adds a 1 Gbit/s uplink to existing 10/100 Mbits/s switches and routers.
  - — Supports ISO 8802.3 standard and *IEEE** 802.3z MAC parameters, repeater and management parameters for 1000 Mbits/s operation.
  - — *IEEE* 802.3x frame-based flow control.

Features (continued)

- Internal PCS layer including 8b/10b encoding/decoding provides 10b interface to an external transceiver including the ability to enable lock to reference and comma detection.
- GMII interface provides connectivity to 1000 base media access controllers and various PHYs.
- Autonegotiation used to advertise speed and duplex capability and to determine flow control symmetry.
- 64-bit bidirectional data bus.
- Host clock operates from 33 MHz to 66 MHz.
  - — Byte lane enables on each write to the LU5M31.
  - — Transmit and receive operations are under hardware control.
  - — CRC generation can be enabled on a per-frame basis while transmitting frames.
  - — CRC can be removed from receive frames before being read from the LU5M31.
  - — Programmable word count threshold determines burst size for FIFO read and write.
- Network management support through on-chip receive and transmit event counters.
  - — 33 MHz synchronous CPU interface with burst support for event counter reads.
  - — Event counters are based on the remote monitor (RMON) management information base (MIB) and Chapter 30 of *IEEE* 802.3u/*IEEE* 802.3z.
  - — Frame statistics are appended to every receive frame.
  - — Interrupt signal alerts the host of transmit and receive bus errors or CPU interface errors as well as event counter overflows.
- Separate 2048-byte transmit and 8192-byte receive FIFOs allow on-chip buffering of entire frame:
  - — Transmit and receive FIFOs can hold multiple frames.
  - — Independent thresholds allow filtering of undersized frames and fragments.
  - — Independent transmit thresholds allow host to set the number of words that must be written to the transmit FIFO before the transmission on the Ethernet starts.
  - — User-selectable watermarks control when flow control frames are transmitted by the LU5M31.
- Internal and external loopback support provides debug capability to the host.
- Internally generated random jitter test pattern capability
- Testability functions:
  - — JTAG boundary scan.
  - — Full internal scan.
  - — Built-in self test (BIST) for internal memories.
- 0.35 µm CMOS technology.
- 3.3 V power supply.
- 5 V input tolerance capability (uses separate supply).
- 208-pin SQFP package.

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

Table of Contents

Contents

Page

Table of Contents (continued)

Tables

Page

Table of Contents (continued)

Tables

Figures

Description

Architecture

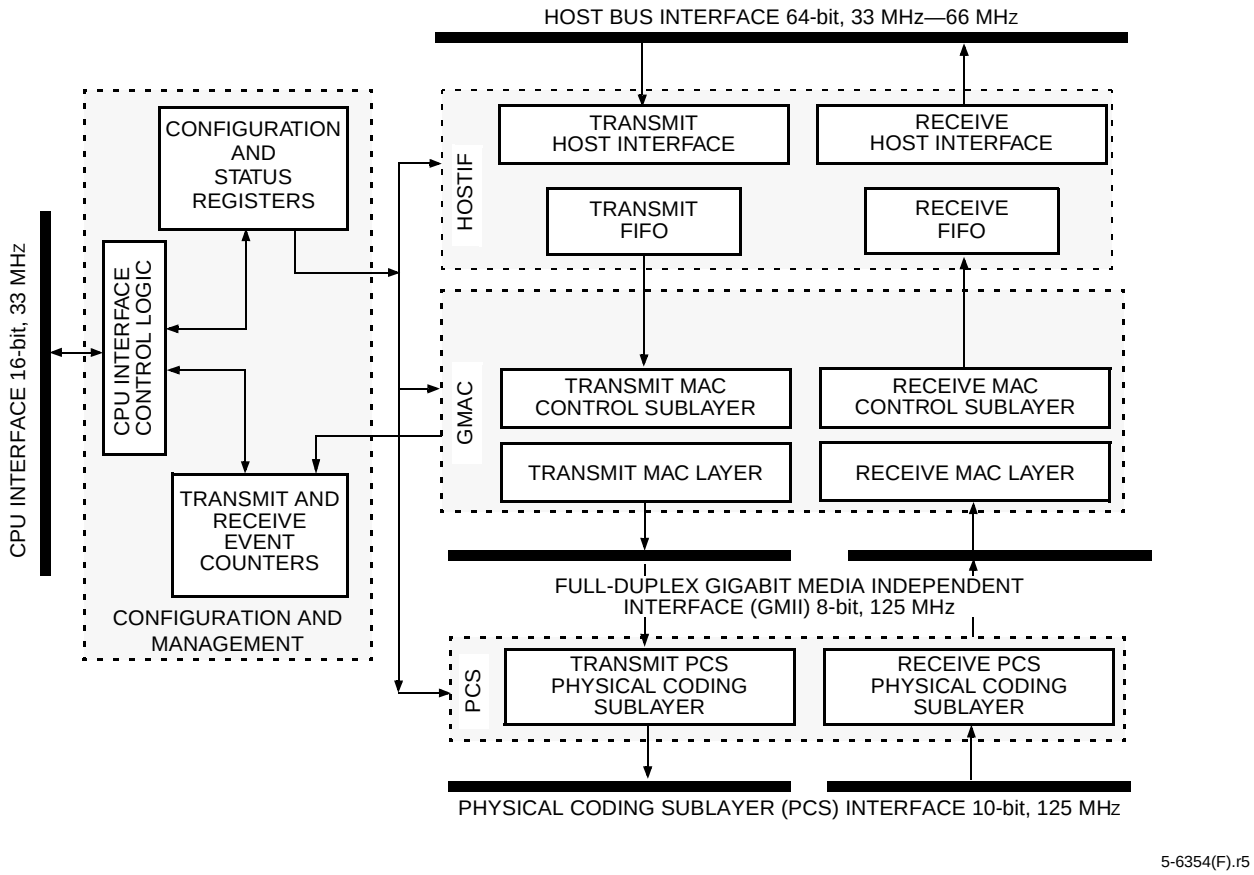


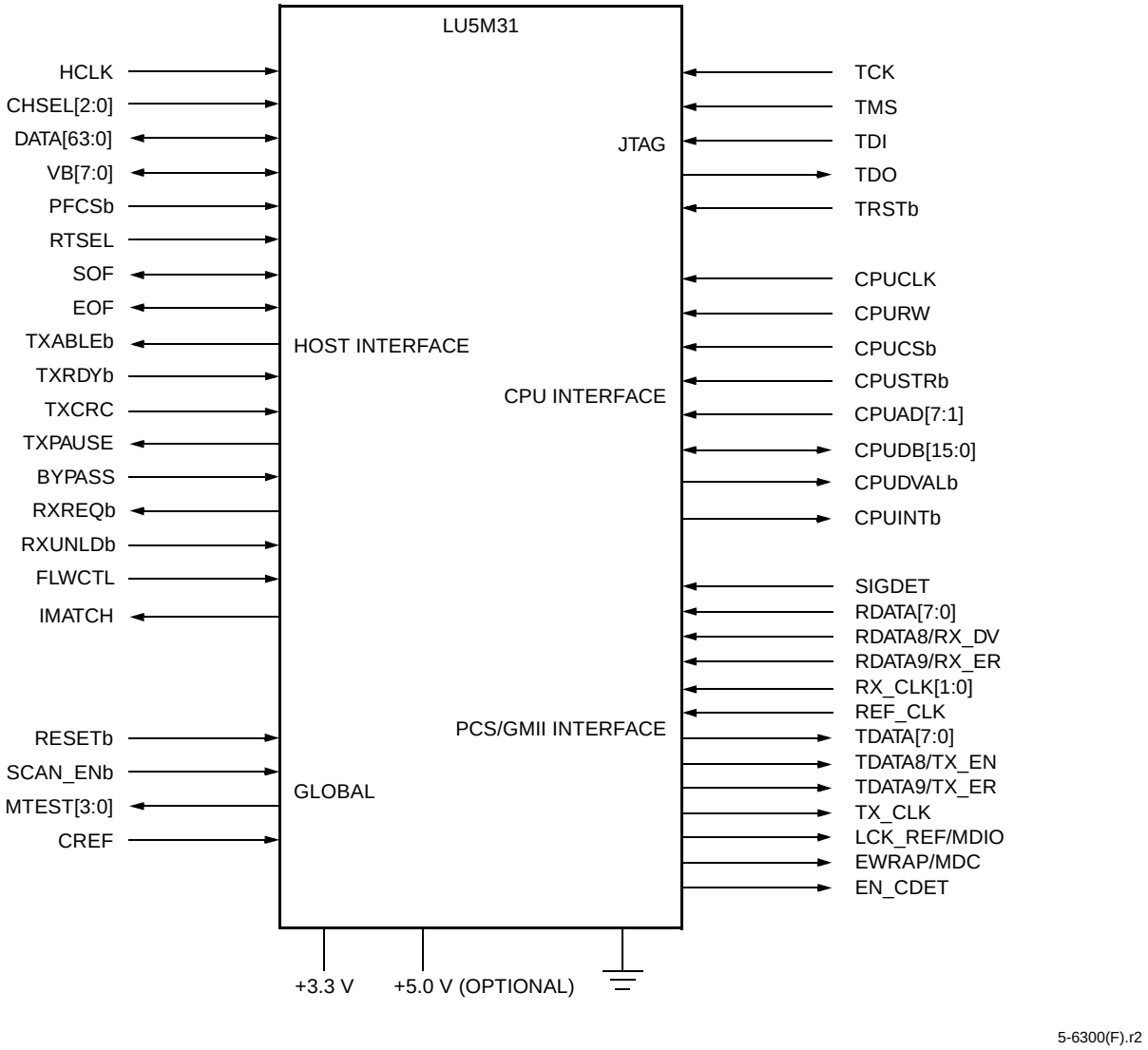
Figure 1. LU5M31 Architecture

Pin Information

Signal Configuration

This section describes each LU5M31 signal and its function. Further information on these signals and how they interact can be found in the appropriate functional section of this data sheet.

Figure 2 shows the LU5M31 signal configuration. The signals are grouped into functional sections representing the primary interfaces on the device. The arrows shown indicate the signal's direction (input and/or output).

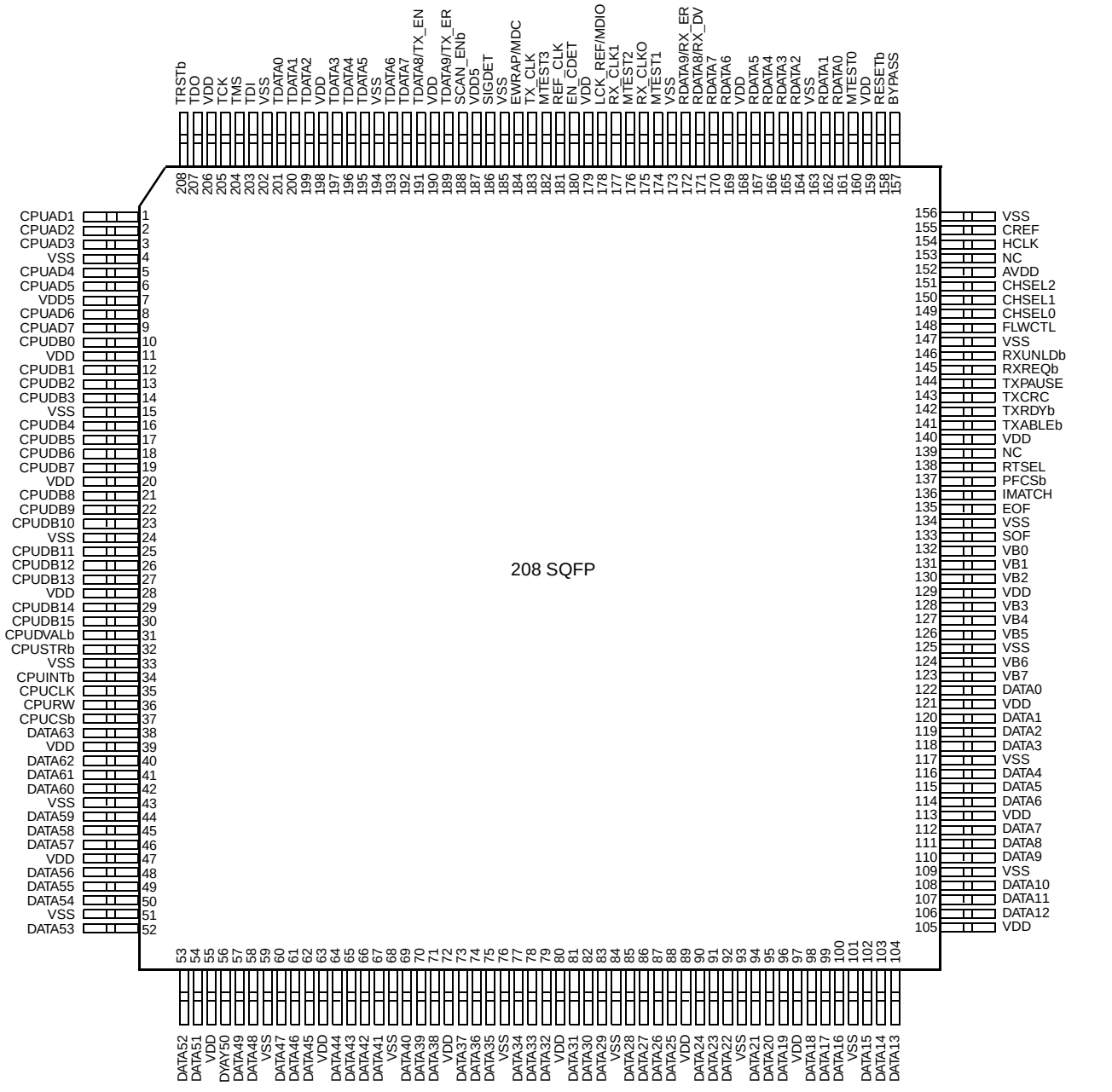


5-6300(F).r2

Figure 2. LU5M31 Signal Group Configuration

Pin Information (continued)

Preliminary Pin Diagram



5-6349(F).r3

Figure 3. LU5M31 Pinout—Top View

Pin Information (continued)

Table 1. LU5M31 Signals According to Pin Number in Numeric Sequence

Number	Name
1	CPUAD1
2	CPUAD2
3	CPUAD3
4	VSS
5	CPUAD4
6	CPUAD5
7	VDD5
8	CPUAD6
9	CPUAD7
10	CPUDB0
11	VDD
12	CPUDB1
13	CPUDB2
14	CPUDB3
15	VSS
16	CPUDB4
17	CPUDB5
18	CPUDB6
19	CPUDB7
20	VDD
21	CPUDB8
22	CPUDB9
23	CPUDB10
24	VSS
25	CPUDB11
26	CPUDB12
27	CPUDB13
28	VDD
29	CPUDB14
30	CPUDB15
31	CPUDVALb
32	CPUSTRb
33	VSS
34	CPUINTb
35	CPUCLK
36	CPURW
37	CPUCSb
38	DATA63
39	VDD
40	DATA62
41	DATA61
42	DATA60
43	VSS
44	DATA59
45	DATA58
46	DATA57
47	VDD
48	DATA56
49	DATA55
50	DATA54
51	VSS
52	DATA53
53	DATA52
54	DATA51
55	VDD
56	DATA50
57	DATA49
58	DATA48
59	VSS
60	DATA47
61	DATA46
62	DATA45
63	VDD
64	DATA44
65	DATA43
66	DATA42
67	DATA41
68	VSS
69	DATA40
70	DATA39
71	DATA38
72	VDD
73	DATA37
74	DATA36
75	DATA35
76	VSS
77	DATA34
78	DATA33
79	DATA32
80	VDD
81	DATA31
82	DATA30
83	DATA29
84	VSS
85	DATA28
86	DATA27
87	DATA26
88	DATA25
89	VDD
90	DATA24
91	DATA23
92	DATA22
93	VSS
94	DATA21
95	DATA20
96	DATA19
97	VDD
98	DATA18
99	DATA17
100	DATA16
101	VSS
102	DATA15
103	DATA14
104	DATA13
105	VDD
106	DATA12
107	DATA11
108	DATA10
109	VSS
110	DATA9
111	DATA8
112	DATA7
113	VDD
114	DATA6
115	DATA5
116	DATA4
117	VSS
118	DATA3
119	DATA2
120	DATA1
121	VDD
122	DATA0
123	VB7
124	VB6
125	VSS
126	VB5
127	VB4
128	VB3
129	VDD
130	VB2
131	VB1
132	VB0
133	SOF
134	VSS
135	EOF
136	IMATCH
137	PFCSb
138	RTSEL
139	NC
140	VDD
141	TXABLEb
142	TXRDYb
143	TXCRC
144	TXPAUSE
145	RXREQb
146	RXUNLDb
147	VSS
148	FLWCTL
149	CHSEL0
150	CHSEL1
151	CHSEL2
152	AVDD
153	NC
154	HCLK
155	CREF
156	VSS
157	BYPASS
158	RESETb
159	VDD
160	MTEST0
161	RDATA0
162	RDATA1
163	VSS
164	RDATA2
165	RDATA3
166	RDATA4
167	RDATA5
168	VDD
169	RDATA6
170	RDATA7
171	RDATA8/ RX_DV
172	RDATA9/ RX_ER
173	VSS
174	MTEST1
175	RX_CLK0
176	MTEST2
177	RX_CLK1
178	LCK_REF/ MDIO
179	VDD
180	EN_CDET
181	REF_CLK
182	MTEST3
183	TX_CLK
184	EWRAP/ MDC
185	VSS
186	SIGDET
187	VDD5
188	SCAN_ENb
189	TDATA9/ TX_ER
190	VDD
191	TDATA8/ TX_EN
192	TDATA7
193	TDATA6
194	VSS
195	TDATA5
196	TDATA4
197	TDATA3
198	VDD
199	TDATA2
200	TDATA1
201	TDATA0
202	VSS
203	TDI
204	TMS
205	TCK
206	VDD
207	TDO
208	TRSTb

Pin Information (continued)

Table 2. LU5M31 Signals According to Pin Number in Alphanumeric Sequence

Number	Name
152	AVDD
157	BYPASS
149	CHSEL0
150	CHSEL1
151	CHSEL2
1	CPUAD1
2	CPUAD2
3	CPUAD3
5	CPUAD4
6	CPUAD5
8	CPUAD6
9	CPUAD7
35	CPUCLK
37	CPUCSb
10	CPUDB0
12	CPUDB1
23	CPUDB10
25	CPUDB11
26	CPUDB12
27	CPUDB13
29	CPUDB14
30	CPUDB15
13	CPUDB2
14	CPUDB3
16	CPUDB4
17	CPUDB5
18	CPUDB6
19	CPUDB7
21	CPUDB8
22	CPUDB9
31	CPUDVALb
34	CPUINTb
36	CPURW
32	CPUSTRb
155	CREF
122	DATA0
120	DATA1
108	DATA10
107	DATA11
106	DATA12
104	DATA13
103	DATA14
102	DATA15
100	DATA16
99	DATA17
98	DATA18
96	DATA19
119	DATA2
95	DATA20
94	DATA21
92	DATA22
91	DATA23
90	DATA24
88	DATA25
87	DATA26
86	DATA27
85	DATA28
83	DATA29
118	DATA3
82	DATA30
81	DATA31
79	DATA32
78	DATA33
77	DATA34
75	DATA35
74	DATA36
73	DATA37
71	DATA38
70	DATA39
116	DATA4
69	DATA40
67	DATA41
66	DATA42
65	DATA43
64	DATA44
62	DATA45
61	DATA46
60	DATA47
58	DATA48
57	DATA49
115	DATA5
56	DATA50
54	DATA51
53	DATA52
52	DATA53
50	DATA54
49	DATA55
48	DATA56
46	DATA57
45	DATA58
44	DATA59
114	DATA6
42	DATA60
41	DATA61
40	DATA62
38	DATA63
112	DATA7
111	DATA8
110	DATA9
180	EN_CDET
135	EOF
184	EWRAP/ MDC
148	FLWCTL
154	HCLK
136	IMATCH
178	LCK_REF/ MDIO
160	MTEST0
174	MTEST1
176	MTEST2
182	MTEST3
139	NC
153	NC
137	PFCSb
161	RDATA0
162	RDATA1
164	RDATA2
165	RDATA3
166	RDATA4
167	RDATA5
169	RDATA6
170	RDATA7
171	RDATA8/ RX_DV
172	RDATA9/ RX_ER
181	REF_CLK
158	RESETb
138	RTSEL
175	RX_CLK0
177	RX_CLK1
145	RXREQb
146	RXUNLDb
188	SCAN_ENb
186	SIGDET
133	SOF
205	TCK
201	TDATA0
200	TDATA1
199	TDATA2
197	TDATA3
196	TDATA4
195	TDATA5
193	TDATA6
192	TDATA7
191	TDATA8/ TX_EN
189	TDATA9/ TX_ER
203	TDI
207	TDO
204	TMS
208	TRSTb
183	TX_CLK
141	TXABLEb
143	TXCRC
144	TXPAUSE
142	TXRDYb
132	VB0
131	VB1
130	VB2
128	VB3
127	VB4
126	VB5
124	VB6
123	VB7
11	VDD
20	VDD
28	VDD
39	VDD
47	VDD
55	VDD
63	VDD
72	VDD
80	VDD
89	VDD
97	VDD
105	VDD
113	VDD
121	VDD
129	VDD
140	VDD
159	VDD
168	VDD
179	VDD
190	VDD
198	VDD
206	VDD
7	VDD5
187	VDD5
4	VSS
15	VSS
24	VSS
33	VSS
43	VSS
51	VSS
59	VSS
68	VSS
76	VSS
84	VSS
93	VSS
101	VSS
109	VSS
117	VSS
125	VSS
134	VSS
147	VSS
156	VSS
163	VSS
173	VSS
185	VSS
194	VSS
202	VSS

Pin Information (continued)

Signal Pins

The LU5M31 uses the signals listed in the following tables. Signal names ending in a lower case <b> are active-low signals. Signal type is designated as input (I), output (O), 3-state bidirectional (B), or power (PWR).

Table 3. Physical Coding Sublayer (PCS) Signal Pins (3.3 V Only)

Pin	Type	Name/Description
TX_CLK	O	**Transmit Clock**. 125 MHz transmit clock. TDATA[9:0] is driven by the LU5M31 on the rising edge of this clock.
TDATA[7:0] TDATA8/TX_EN TDATA9/TX_ER	O	**In PCS Mode:** **Transmit Data**. The TDATA[9:0] outputs provide 10b code groups to the external Serdes. TDATA[9:0] is driven by the LU5M31 on the rising edge of TX_CLK. **Bypass PCS Mode:** **Transmit Data.** The TDATA[7:0] outputs provide 8-bit wide data on each rising edge of TX_CLK to the external Gigabit PHY. **Transmit Enable**. Transmit enable is an output of the LU5M31 that indicates a transmission is starting. This signal is asserted by the LU5M31 synchronously with the first byte of preamble, and it remains asserted while all bytes are transmited. TX_CLK following the transmission of the final byte of the frame. TX_EN is synchronous to TX_CLK. **Transmit Error.** TX_ER indicates that LU5M31 is requesting that the PHY transmit a coding error, as per Section 35.2.2.8
LCK_REF/MDIO	O	**Lock PHY Clock**. In PCS mode this signal mirrors **lckref** (bit 15 in the global configuration register) and allows the LU5M31 to drive the LCK_REF input to a serde's device. This signal will be high after reset. **Management Data Input/Output.** In bypass PCS mode MDIO is a bidirectional data signal used for reading status data and writing control data between the LU5M31 and a physical layer device. As an input or an output, MDIO is synchronous to MDC. An external 2 kΩ ± 5% pull-down resistor is required on this signal.
EWRAP/MDC	O	**PHY Loopback**. In PCS mode this signal is asserted when external **Loopback** (bit 14 in the GMII control register) is set allowing the LU5M31 to drive the EWRAP input to a PHY. This signal will be low after reset. **Management Data Clock.** In bypass PCS mode MDC is sourced by the LU5M31 when the LU5M31 needs to read management status or write management control to the PHY. Data is read by the LU5M31 on the rising edge, and is driven by the LU5M31 on the falling edge of MDC when management registers are not being accessed. MDC is held low and MDIO is 3-stated. MDC is aperiodic, with a minimum low and high time of 160 ns. When SpeedSel=1. MDC frequency is HCLK/32. When SpeedSel=0, MDC frequency is HCLK/16. When Fast_MDC is set, MDC frequency is HCLK/8, regardless of SpeedSel.
EN_CDET	O	**Enable Comma Detect**. This signal mirrors **encomdet** (bit 14 in the global configuration register) and allows the LU5M31 to drive the EN_CDET input to a PHY. This signal will be high after reset.
RX_CLK[1:0]	I	**Receive Clock**. These recovered clocks are sourced by the PHY. RX_CLK[1:0] are 62.5 MHz clocks, 180 degrees out of phase. RX_CLK1 is used to clock even-code groups on RDATA[9:0], and RX_CLK0 is used to clock odd code groups on RDATA[9:0]. In addition, RX_CLK[1] is the 125 MHz clock in the bypass PCS mode and is used to clock in RDATA[7:0].

Pin Information (continued)

Table 3. Physical Coding Sublayer (PCS) Signal Pins (3.3 V Only) (continued)

Pin	Type	Name/Description
RDATA[7:0] RDATA8/RX_DV RDATA9/RX_ER	I	**In PCS Mode:** **Receive Data.** The LU5M31 samples 10b code groups from the external PHY on the RDATA[9:0] pins. RDATA[9:0] are sampled by the LU5M31 on the rising edges of RX_CLK[1:0]. **Bypass Mode:** Receive Data. RDATA[7:0] is a bundle of eight data signals. **Receive Data Valid.** RX_DV is asserted by the PHY to indicate that recovered and decoded data bytes are valid on RXD. RX_DV remains asserted high from the first byte of receive data through to the final byte received. RX_DV is asserted synchronously to RX_CLK, and is deasserted prior to the first RX_CLK after the final byte is read. **Receive Error.** RX_ER is asserted by the PHY when it has detected an error that the LU5M31 may not be able to detect in the frame currently being received. RX_ER is asserted synchronously to RX_CLK.
REF_CLK	I	**Reference Clock.** This 125 MHz reference clock input is used by the LU5M31 to generate TX_CLK.
SIGDET	I	**Signal Detect**. This signal provides an indication from the PHY that the PMD is detecting light when asserted. This signal should be pulled high on the board if not used. External pull-up recommended.

Pin Information (continued)

Table 4. Host Interface Signal Pins (3.3 V Only)

Pin	Type	Name/Description
BYPASS	I	**Phase-Locked Loop (PLL) Bypass.** This signal is used to bypass the PLL. BYPASS should be low during normal operation.
HCLK	I	**Host Clock.** 33 MHz—66 MHz host clock. This clock is used to control the transfer of data between the host and the LU5M31, as well as providing a local clock within the LU5M31 to manage the host interface logic. All transfers on the host interface are synchronous to the rising edge of HCLK.
CHSEL[2:0]	I	**Channel Select**. These signals provide the host with a method to address a specific LU5M31 in a system via a port address. The value on this bus is compared to the value stored in the global configuration register to determine if the device is selected. This comparison can be disabled on receive or transmit by clearing **rxchsel** or **txchsel** (bits 8 and 9, respectively, in the global configuration register).
PFCSb	I	**LU5M31 Chip Select.** The LU5M31 chip select, when asserted low, is used to select the appropriate LU5M31 device.
RTSEL	I	**Receive-Transmit Select.** The receive-transmit select input determines the direction of the LU5M31 DATA[63:0] bus. When high, the LU5M31 DATA[63:0] bus is an output. When low, the DATA[63:0] bus is an input.
DATA[63:0]	B 3-state	**Data Bus.** The data bus is a 64-bit bidirectional bus. Data is transferred to and from the LU5M31 via the DATA[63:0] bus on the rising edge of HCLK. RTSEL qualifies the data bus as an input or output. The bus is in 3-state when a transfer is not in progress. External 10 kΩ pull-down resistors are recommended.
VB[7:0]	B 3-state	**Valid Bytes.** VB[7:0] is a bidirectional, 3-state bus used to indicate which byte lanes in a data transfer are valid. RTSEL qualifies VB[7:0] as an input or an output. VB[7:0] is in 3-state when the LU5M31 host bus is not enabled. External 10 kΩ pull-down resistors are recommended.
SOF	B 3-state	**Start of Frame.** The start-of-frame signal indicates the start of a receive or transmit frame. While RTSEL is high (receive), SOF is an output and indicates that the current data transfer contains the first byte of a received frame. While RTSEL is low (transmit), SOF is an input and should be asserted high when the first word of a transmit frame is valid during the current data access. An external 10 kΩ pull-down resistor is recommended.
EOF	B 3-state	**End of Frame.** The end-of-frame signal indicates the end of a receive or transmit frame. While RTSEL is high (receive), EOF is an output and indicates that the current data transfer contains the last byte of a received frame. While RTSEL is low (transmit), EOF is an input and should be asserted high when the last word of a transmit frame is written to the LU5M31. An external 10 kΩ pull-down resistor is recommended.
FLWCTL	I	**Flow Control**. Asserting this signal high will force the LU5M31 to transmit a MAC control frame that contains a valid PAUSE opcode and a preprogrammed number that represents the number of pause quantums the receiving station should pause for. This signal must be asserted for at least two HCLK cycles.
TXABLEb	O	**Host Transmit Request.** The LU5M31 asserts this signal low when the transmit FIFO is capable of accepting data. When TXABLEb is asserted low, the available space in the transmit FIFO will be equal to or greater than the transmit word count threshold set in the global FIFO configuration register.

Pin Information (continued)

Table 4. Host Interface Signal Pins (3.3 V Only) (continued)

Pin	Type	Name/Description
TXRDYb	I	**Host Transmit Ready.** The LU5M31 clocks data into the transmit FIFO from the DATA[63:0] bus on the rising edge of HCLK while TXRDYb and RTSEL are low and the LU5M31 host bus is enabled.
TXCRC	I	**Transmit CRC Checksum.** When asserted high, in conjunction with **hwcrc** (bit 2 of the transmit frame configuration register), this signal is used to inform the LU5M31 that a CRC checksum is to be calculated for the current frame and appended to the end of the frame prior to transmission. TXCRC must be asserted with EOF during a write to the LU5M31.
TXPAUSE	O	**Transmission Paused.** This signal is asserted high by the LU5M31 while the LU5M31 transmitter is paused due to reception of a valid MAC control frame with a PAUSE opcode.
RXREQb	O	**Receive Request**. RXREQb is asserted low when the receive threshold conditions (as defined in the global FIFO configuration register) have been met. The assertion of this signal indicates that data in the receive FIFO is available to be read by the host.
RXUNLDb	I	**Host Receive Unload.** The host asserts this signal low to acknowledge a receive request from the LU5M31. When the host bus is enabled and this signal is asserted low, a burst read from the LU5M31 will start. The burst length will be equal to the burst size defined by the receive word count threshold.
IMATCH	O 3-state	**Address Match**. The address match signal will be asserted high by the LU5M31 when the frame, currently being read from the receive FIFO, has a destination address that matches the Ethernet address loaded in the source address registers of the LU5M31. This signal will be asserted for one HCLK cycle with SOF if a match is made. This signal will not be asserted if the address in the source address register is 0x0000. An external 10 kΩ pull-down resistor is recommended.

Pin Information (continued)

Table 5. CPU Interface Pins (5.0 V Tolerant Inputs)

Pin	Type	Name/Description
CPUCLK	I	**CPU Clock.** 33 MHz CPU clock. This clock is used to control transfer of data between the LU5M31 and the CPU, as well as providing an internal clock for the CPU interface logic. All transfers on the CPU interface are synchronous to the rising edge of CPUCLK.
CPUCSb	I	**CPU Interface Chip Select**. This signal must be asserted low in order to access a register or counter in the LU5M31. All CPU accesses are qualified with CPUCSb. When CPUCSb is sampled high, the CPU data bus (CPUDB) will be in 3-state one CPU clock cycle later.
CPUSTRb	I	**CPU Strobe**. When asserted low, this signal initiates a CPU transaction. When deasserted high, this signal indicates the end of a CPU transaction.
CPUAD[7:1]	I	**CPU Address Bus.** The address for the register or event counter being accessed is provided on this bus. All LU5M31 registers and event counters are 16-bit word aligned, irrespective of their size.
CPUDB[15:0]	B 3-state	**CPU Data Bus**. Data is transferred to or from the LU5M31 registers or event counters via CPUDB[15:0]. The state of CPURW determines whether this bus is an input or an output. This bus is qualified with both CPUSTRb and CPUCSb. External 10 kΩ pull-down resistors are recommended.
CPURW	I	**CPU Read or Write**. This signal must be high for a read and low for a write and determines the direction of the CPU data bus.
CPUDVALb	O 3-state	**CPU Data Valid**. For a read, CPUDVALb is asserted low while there is valid data being presented on the CPU data bus. For a write, CPUDVALb is asserted low for one CPUCLK cycle to indicate data has been registered by the LU5M31. An external 1 kΩ pull-up resistor is recommended.
CPUINTb	O Open Drain	**CPU Interrupt.** The CPU interrupt is asserted low when any interrupt bit is asserted and not masked. This signal will remain asserted low until the interrupt register causing the interrupt is read. An external 1 kΩ pull-up resistor is recommended.

Pin Information (continued)

Table 6. Global Signal Pins (5.0 V Tolerant Inputs)

Pin	Type	Name/Description
RESETb	I	**Hardware Reset.** This signal, when asserted low, performs a complete asynchronous reset of the LU5M31, including the PLL. The LU5M31 will be functional 250 ms after the completion of reset (RESETb deasserted high).
SCAN_ENb	I Internal Pull-up	**Scan Enable.** While asserted low, this signal enables the shifting of data through scan chains when the associated scan clock is pulsed. While deasserted high, scan chain shifting is disabled and the device is in functional mode. This signal should be deasserted high for normal operation.
MTEST[3:0]	O	**Factory Test**. These pins are used for factory test and should not be connected on a production board.
CREF	I	**PLL Reference Voltage.** This pin provides the dc reference voltage for the PLL. This pin must be connected to ground through a 1.0 nF ± 10% capacitor.
V_{DD}	PWR	**3.3 V Power.**
AV_{DD}	PWR	**3.3 V Analog Power.**
$V_{DD}5$	PWR	**5.0 V Power (Optional).**
V_{SS}	PWR	**Digital Ground.**
NC	—	**No Connect.** These pins should be left unconnected.

Pin Information (continued)

Table 7. JTAG Test Signal Pins (5.0 V Tolerant Inputs)

Pin	Type	Name/Description
Refer to *IEEE* 1149.1		
TCK	I	**Test Clock**. This signal provides the clock for the test logic defined by the *IEEE* 1149.1 standard.
TMS	I Internal Pull-up	**Test Mode Select**. This signal is decoded by the TAP controller to control test operations.
TDI	I Internal Pull-up	**Test Data Input**. Serial test instructions and data are received by the test logic through this signal.
TDO	O	**Test Data Output**. Serial data and instructions from the test logic are provided by this signal.
TRSTb	I	**Test Reset**. When asserted low, this signal provides an asynchronous reset of the TAP controller. This signal needs to be asserted low for normal operation.

Register Information

Register Address Map

The LU5M31 has several configuration registers. These registers all reset to known default values that enable the LU5M31 to transmit and receive frames. These registers should be updated if the default configuration does not suit the needs of the host system. The LU5M31 resets to a default configuration. The registers are addressed in memory space through the LU5M31 CPU interface.

Configuration registers should only be written at device initialization or after a hardware reset (RESETb) or a global software reset. The **gsrst** bit (bit 3 set in the global configuration register) resets all registers, except the global configuration register, to their default values as shown in parenthesis in the Register Descriptions section. In general, they should not be updated while the LU5M31 is transmitting or receiving frames. Table 8 lists the address map for the LU5M31 registers.

Table 8. Address Map

Address[7:0]	Register/Event Counter	Reset Value	Read/Write (R/W)
Configuration/Status Registers			
0x00	Identification Register	0x0800	R
0x02	Global Configuration Register, Low Word	0xEB00	R/W
0x04	Global Configuration Register, High Word	0x0000	R/W
0x06	Global FIFO Configuration Register, Low Word	0x5558	R/W
0x08	Global FIFO Configuration Register, High Word	0x3BE8	R/W
0x0A	Transmit Frame Configuration Register	0x005C	R/W
0x0C	Receive Frame Configuration Register	0x0020	R/W
0x0E	One-level VLAN Tag Register	0x0000	R/W
0x10	Two-level VLAN Tag Register	0x0000	R/W
0x12	Source Address Register, Low Word	0x0000	R/W
0x14	Source Address Register, Middle Word	0x0000	R/W
0x16	Source Address Register, High Word	0x0000	R/W
0x18	Transmit Pause Frame Timer Register	0x0000	R/W
0x1A	Interrupt Identification Register	0x0000	R
0x1C	Interrupt Mask Register	0xFFE7	R/W
0x1E	Interrupt Register	0x0000	R
0x20	Receive Counter Interrupt Mask Register, Low Word	0xFFFF	R/W
0x22	Receive Counter Interrupt Mask Register, High Word	0xFFFF	R/W
0x24	Receive Counter Interrupt Register, Low Word	0x0000	R
0x26	Receive Counter Interrupt Register, High Word	0x0000	R
0x28	Transmit Counter Interrupt Mask Register, Low Word	0xFFFF	R/W
0x2A	Transmit Counter Interrupt Mask Register, High Word	0xFFFF	R/W
0x2C	Transmit Counter Interrupt Register, Low Word	0x0000	R
0x2E	Transmit Counter Interrupt Register, High Word	0x0000	R
0x30	Receive Port Diagnostic Register	0x0000	R
0x32	GMII Control Register	0x0120	R/W
0x34	GMII Status Register	0x0009	R
0x36	Autonegotiation Advertisement Register	0x01A0	R/W
0x38	Autonegotiation Link Partner Ability Base Page Register	0x0000	R
0x3A	Autonegotiation Expansion Register	0x0004	R
0x3C	Autonegotiation Next Page Transmit Register	0x0000	R/W
0x3E	Autonegotiation Link Partner Ability Next Page Register	0x0000	R
0x40—0x42	Reserved	—	—
0x44	MII Data Register	0x0000	R/W

Register Information (continued)

Table 8. Address Map (continued)

Address[7:0]	Register/Event Counter	Reset Value	Read/Write (R/W)
0x46	MII Control Register	0x0000	R/W
Configuration/Status Registers (continued)			
0x48	PCS Data Register	0x0000	R
0x4A—0x4E	Reserved	—	—
Transmit Event Counters			
0x50	Frames Transmitted (64 Octets), Low Word	0x0000	R/W
0x52	Frames Transmitted (64 Octets), Middle Word	0x0000	R/W
0x54	Frames Transmitted (64 Octets), High Word	0x0000	R/W
0x56	Frames Transmitted (65 to 127 Octets), Low Word	0x0000	R/W
0x58	Frames Transmitted (65 to 127 Octets), Middle Word	0x0000	R/W
0x5A	Frames Transmitted (65 to 127 Octets), High Word	0x0000	R/W
0x5C	Frames Transmitted (128 to 255 Octets), Low Word	0x0000	R/W
0x5E	Frames Transmitted (128 to 255 Octets), High Word	0x0000	R/W
0x60	Frames Transmitted (256 to 511 Octets), Low Word	0x0000	R/W
0x62	Frames Transmitted (256 to 511 Octets), High Word	0x0000	R/W
0x64	Frames Transmitted (512 to 1023 Octets), Low Word	0x0000	R/W
0x66	Frames Transmitted (512 to 1023 Octets), High Word	0x0000	R/W
0x68	Frames Transmitted (1024 to maximum size octets), Low Word	0x0000	R/W
0x6A	Frames Transmitted (1024 to maximum size octets), High Word	0x0000	R/W
0x6C	Long Frames Transmitted, Low Word	0x0000	R/W
0x6E	Long Frames Transmitted, High Word	0x0000	R/W
0x70	Total Good Frames Transmitted, Low Word	0x0000	R/W
0x72	Total Good Frames Transmitted, Middle Word	0x0000	R/W
0x74	Total Good Frames Transmitted, High Word	0x0000	R/W
0x76	MAC Pause Frames Transmitted	0x0000	R/W
0x78	Multicast Frames Transmitted, Low Word	0x0000	R/W
0x7A	Multicast Frames Transmitted, High Word	0x0000	R/W
0x7C	Broadcast Frames Transmitted, Low Word	0x0000	R/W
0x7E	Broadcast Frames Transmitted, High Word	0x0000	R/W
0x80	Total Octets Transmitted, Low Word	0x0000	R/W
0x82	Total Octets Transmitted, Middle Word	0x0000	R/W
0x84	Total Octets Transmitted, High Word	0x0000	R/W
0x86	Transmit Underruns	0x0000	R/W
Receive Event Counters			
0x88	Frames Received (64 Octets), Low Word	0x0000	R/W
0x8A	Frames Received (64 Octets), Middle Word	0x0000	R/W
0x8C	Frames Received (64 Octets), High Word	0x0000	R/W
0x8E	Frames Received (65 to 127 Octets), Low Word	0x0000	R/W
0x90	Frames Received (65 to 127 Octets), Middle Word	0x0000	R/W
0x92	Frames Received (65 to 127 Octets), High Word	0x0000	R/W
0x94	Frames Received (128 to 255 Octets), Low Word	0x0000	R/W
0x96	Frames Received (128 to 255 Octets), High Word	0x0000	R/W
0x98	Frames Received (256 to 511 Octets), Low Word	0x0000	R/W
0x9A	Frames Received (256 to 511 Octets), High Word	0x0000	R/W
0x9C	Frames Received (512 to 1023 Octets), Low Word	0x0000	R/W

Register Information (continued)

Table 8. Address Map (continued)

Address[7:0]	Register/Event Counter	Reset Value	Read/Write (R/W)
0x9E	Frames Received (512 to 1023 Octets), High Word	0x0000	R/W
0xA0	Frames Received (1024 to maximum size octets), Low Word	0x0000	R/W
Receive Event Counters (continued)			
0xA2	Frames Received (1024 to maximum size octets), High Word	0x0000	R/W
0xA4	Long Frames Received, Low Word	0x0000	R/W
0xA6	Long Frames Received, High Word	0x0000	R/W
0xA8	Receive Errors	0x0000	R/W
0xAA	Total Readable Frames Received, Low Word	0x0000	R/W
0xAC	Total Readable Frames Received, Middle Word	0x0000	R/W
0xAE	Total Readable Frames Received, High Word	0x0000	R/W
0xB0	Jabber Frames Received	0x0000	R/W
0xB2	MAC Pause Frames Received	0x0000	R/W
0xB4	Frames Received with Bad CRC, Low Word	0x0000	R/W
0xB6	Frames Received with Bad CRC, High Word	0x0000	R/W
0xB8	Multicast Frames Received, Low Word	0x0000	R/W
0xBA	Multicast Frames Received, High Word	0x0000	R/W
0xBC	Broadcast Frames Received, Low Word	0x0000	R/W
0xBE	Broadcast Frames Received, High Word	0x0000	R/W
0xC0	Total Octets Received, Low Word	0x0000	R/W
0xC2	Total Octets Received, Middle Word	0x0000	R/W
0xC4	Total Octets Received, High Word	0x0000	R/W
0xC6	Total Readable Octets Received, Low Word	0x0000	R/W
0xC8	Total Readable Octets Received, Middle Word	0x0000	R/W
0xCA	Total Readable Octets Received, High Word	0x0000	R/W
0xCC	Missed Frames	0x0000	R/W
0xCE	Fragments Received, Low Word	0x0000	R/W
0xD0	Fragments Received, High Word	0x0000	R/W
0xD2	Undersized Frames Received, Low Word	0x0000	R/W
0xD4	Undersized Frames Received, High Word	0x0000	R/W

Register Descriptions

Identification Register

Address = 0x00

This register is used to identify the revision of the LU5M31. This is a read-only register and is valid at the end of reset. The static value of this register is 0x0801. Writes to this register will be ignored.

Table 9. Identification Register Bit Map

15	14	13	12	11	10	9	8
(0)	(0)	(0)	(0)	(1)	(0)	(0)	(0)

7	6	5	4	3	2	1	0
(0)	(0)	(0)	(0)	(0)	(0)	(0)	(1)

Register Descriptions (continued)

Global Configuration Register

Address = 0x02 (Low Word), 0x04 (High Word)

The global configuration register is used to configure the LU5M31. Upon the completion of reset, this register defaults to the values shown in parentheses. The reset bits in this register are self-clearing, polling this register allows the host to determine when the software reset is complete.

Table 10. Global Configuration Register Bit Map, High Word

31	30	29	28	27	26	25	24
Reserved (0)	bypass_pcs (0)	cont_MDC (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved (0)	Reserved (0)	burst (0)	extbin (0)	enjitter (0)	jitter[1] (0)	jitter[0] (0)	jlength (0)

Table 11. Global Configuration Register Bit Map, Low Word

15	14	13	12	11	10	9	8
lckref (1)	encomdet (1)	timdur[1] (1)	timdur[0] (0)	eninc (1)	loopint (0)	txchsel (1)	rxchsel (1)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
chsel[2] (0)	chsel[1] (0)	chsel[0] (0)	enimatch (0)	gsrst (0)	txres (0)	rxres (0)	cntres (0)

Table 12. Global Configuration Register Bit Definitions

Bit	Name	Description
		High Word
31	Reserved	**Reserved.** This bit is reserved and must be set to zero for normal operation.
30	bypass_pcs	**Bypass PCS Logic.** This bit, when set, enables the bypassing of the PCS logic (useful for debugging). This bit should be set to zero for normal operation.
29	cont_MDC	**Continuous MDC.** When this bit is set, the LU5M31 will provide a continuous management data clock to the GMII interface. When 0, MDC will be running only 64 clocks during the transfer of data over the MDIO pin.
28:22	Reserved	**Reserved.** These bits are reserved and must be set to zero for normal operation.
21	burst	**Burst CPU Read**. This bit, when set, enables the LU5M31 to burst reads of the event counters. When this bit is cleared, a CPU read transaction will return only the addressed location.
20	extbin	**Extend Frame Binning.** This bit, when set, enables the LU5M31 to extend the maximum frame size from 1518 bytes to 1538 bytes, irrespective of VLAN tagging. When this bit is cleared, the maximum frame size for non-VLAN frames is set to 1518 bytes. The maximum size of VLAN1 frames is set to 1522 bytes. The maximum size of VLAN2 frames is set to 1538 bytes.
19	enjitter	**Enable Jitter Test.** This bit, when set, enables the LU5M31 to automatically transmit the jitter test patterns in accordance with Annex 36A in *IEEE* 802.3z. Also, when this bit is set, **txenable** (bit 0 of transmit frame configuration register) will automatically be cleared.

Register Descriptions (continued)

Table 12. Global Configuration Register Bit Definitions (continued)

Bit	Name	Description
18:17	jitter[1:0]	**Jitter Test Encoding.** When the **enjitter** bit is set, the following encoding defines which jitter tests will be performed: 00 = High-frequency test pattern. 01 = Low-frequency test pattern. 10 = Mixed-frequency test pattern. 11 = Continuous random pattern. (In this mode, the **pream** bit will be ignored and 7 bytes preamble will always be prepended to the random pattern.)
16	jlength	**Jitter length.** When this bit is set and the continuous random pattern jitter test has been programmed into bits[19:17], then the continuous random pattern will be 1512 octets long plus CRC. When this bit is cleared and the continuous random pattern jitters test has been programmed, then the continuous random pattern will be 348 octets long plus CRC.
Low Word		
15	lckref	**Lock to Reference**. The value written to this bit will be applied to the LCK_REF output of the LU5M31.
14	encomdet	**Enable Comma Detect.** The value written to this bit will be applied to the EN_CDET output of the LU5M31.
13:12	timdur[1:0]	**Timer Duration**. These bits provide the encoding for the duration of the link timer used in the autonegotiation process. The encoding is 00 = 1 µs, 01 = 1 ms, 10 = 10 ms, 11 = 16 ms.
11	eninc	**Enable Counter**. This bit, when set, enables the event counter to continue to increment while the LU5M31 is in loopback mode. When cleared, the LU5M31 will not increment the counters while in loopback mode.
10	loopint	**Loopback**. This bit, when set, puts the LU5M31 in loopback mode. Data to be transmitted will be looped back into the receive path and presented to the host. When this bit is cleared, loopback is disabled and normal functional mode is selected.
9	txchsel	**Enable Channel Select for Transmit**. Clearing this bit forces the LU5M31 to ignore the CHSEL[2:0] inputs on the device during a transmit transaction.
8	rxchsel	**Enable Channel Select for Receive**. Clearing this bit forces the LU5M31 to ignore the CHSEL[2:0] inputs on the device during a receive transaction.
7:5	chsel[2:0]	**Device Address**. These 3 bits define the LU5M31 device address. These 3 bits are compared against the CHSEL[2:0] inputs to determine if the device is selected by the host.
4	enimatch	**Enable IMATCH.** This bit, when set, enables the address match capability within the GMAC. When cleared, the address match capability will be disabled.
3	gsrst	**Global Software Reset.** Setting this bit forces a reset of the entire LU5M31, except the global configuration register and the PLL. This bit is self-clearing after three CPUCLK cycles.
2	txres	**Transmit Port Reset.** This bit provides a transmit port reset. Setting the **txres** bit forces a reset of the transmit port with the exception of the counters. This bit is self-clearing after three CPUCLK cycles.
1	rxres	**Receive Port Reset**. This bit provides a receive port reset. Setting the **rxres** bit forces a reset of the receive port with the exception of the counters. This bit is self-clearing after three CPUCLK cycles.
0	cntres	**Counter Port Reset.** This bit provides a counter reset. Setting the **cntres** bit forces a reset of all the counters. Counters are reset to zero. This bit is self-clearing after three CPUCLK cycles.

Register Descriptions (continued)

Global FIFO Configuration Register

Address = 0x06 (Low Word), 0x08 (High Word)

The global FIFO configuration register allows the user to configure the transmit and receive FIFOs to match their system requirements. Word count thresholds must be configured prior to enabling **rxenable** (bit 7 in the receive frame configuration register) and **txenable** (bit 0 in the transmit frame configuration register).

Table 13. Global FIFO Configuration Register Bit Map, High Word

31:27	26:22	21:16
rxwcth (00111)	txwcth (01111)	rxwmh (101000)

Table 14. Global FIFO Configuration Register Bit Map, Low Word

15:10	9:5	4:0
rxwml (010101)	rxsfth (01010)	txsfth (11000)

Table 15. Global FIFO Configuration Register Bit Definitions

Bit	Name	Description
		High Word
31:27	rxwcth	**Receive Word Count Threshold.** This represents the number of 8-byte words in the FIFO when RXREQb is low: size = [**rxwcth** (dec) + 1]. Minimum size = 1 word, maximum = 32 words.
26:22	txwcth	**Transmit Word Count Threshold.** This represents the number of 8-byte words in the FIFO when TXABLEb is low: size = [**txwcth** (dec) + 1]. Minimum burst size = 1 word, maximum = 32 words.
21:16	rxwmh	**Receive FIFO High Watermark.** When the occupancy value of the receive FIFO is greater than or equal to this watermark value, a pause frame will be transmitted, if **txfcen** (bit 6 of the transmit frame configuration register) is set. This level is calculated as follows: number of 8-byte words = [(**rxwmh** (dec) + 1) * 16].
		Low Word
15:10	rxwml	**Receive FIFO Low Watermark.** When the occupancy value of the receive FIFO is less than or equal to this watermark value, a pause frame with a zero pause quanta will be transmitted if **rxfcen** (bit 5 of the receive frame configuration register) is set. This level is calculated as follows: number of 8-byte words = [(**rxwml** (dec) + 1) * 16].
9:5	rxsfth	**Receive Start of Frame Threshold.** This represents the number of 8-byte words that must be received before RXREQb is asserted at the start of a frame. The size of **rxsfth** must be greater than or equal to the size of **rxwcth**. The RXREQb signal is asserted once **rxwcth** has been met on all subsequent bursts of a frame: number of 8-byte words in RXFIFO = [**rxsfth** (dec) + 1].
4:0	txsfth	**Transmit Start of Frame Threshold.** This represents the number of words that must be written to the transmit FIFO, before the MAC is requested to transmit data at the start of a frame: number of words in TXFIFO = [(**txsfth** (dec) + 1) * 8].

Register Descriptions (continued)

Transmit Frame Configuration Register

Address = 0x0A

The transmit frame configuration register configures the transmit port of the LU5M31. This register defaults to an ISO 8802.3 compliant mode after reset. These register bits can be changed to accommodate different options.

Table 16. Transmit Frame Configuration Register Bit Map

15	14	13	12	11	10	9	8
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	flwent (0)	flwent (0)	flwent (0)	flwent (0)

7	6	5	4	3	2	1	0
flow (0)	txfcen (1)	txhalt (0)	pream (1)	pad (1)	hwcrc (1)	crc (0)	txenable (0)

Table 17. Transmit Frame Configuration Register Bit Definitions

Bit	Name	Description
15:12	Reserved	**Reserved**. These bits are reserved and must be zero.
11:8	flwent[3:0]	**Flow Count.** The value programmed into these bits will determine how many pause_quanta (512 bit times) early to send the next pause frame when sending multiple pause frames. This pause – quantum = transmit pause frame timer – flow count.
7	flow	**Flow Control Mode**. When this bit is set, the LU5M31 will transmit multiple flow control frames while the FLWCTL signal is asserted high. When this bit is cleared, the LU5M31 will transmit a single-flow control frame with the FLWCTL signal asserted.
6	txfcen	**Transmit Flow Control Enable**. When this bit is set, the LU5M31 will respond to received PAUSE frames by pausing the transmitter for the number of pause quanta designated by the received PAUSE frame. If cleared, the transmitter will not PAUSE when PAUSE frames are received.
5	txhalt	**Halt on Error Condition.** When this bit is set, the transmit FIFO will halt after a FIFO underrun. The FIFO can be restarted by resetting the **txenable** bit in this register. When **txhalt** is cleared, the transmit FIFO continues to operate as normal, even after a transmit FIFO underrun.
4	pream	**Preamble Enable.** If this bit is set, 7 bytes of preamble will be prepended to the transmitted frame. If this bit is cleared, 1 byte of preamble will be prepended to the transmitted frame. The start-of-frame delimiter (SFD) will always be transmitted after preamble.
3	pad	**Pad Transmit Data.** While this bit is set, the transmit MAC logic of the LU5M31 will automatically append data to a frame less than 60 bytes in length. Frames are padded with data and a new CRC such that the frame length is equal to 64 bytes. Refer to Table 68.
2	hwcrc	**Hardware Select CRC.** When this bit is set, the option to calculate and append CRC to each transmitted frame is determined by the TXCRC hardware signal. When **hwcrc** is cleared, the **crc** bit in the transmit frame configuration register determines whether or not the CRC is calculated and appended by the LU5M31.
1	crc	**Append CRC.** This bit is valid when **hwcrc** is cleared. While **crc** is cleared, the CRC checksum is not calculated and appended to each transmitted frame. While **crc** is set, the CRC checksum is calculated and transmitted with each frame as the last four octets. This bit should only be set or reset during initialization.
0	txenable	**Transmit Port Enable**. Setting this bit enables the LU5M31 to transmit frames. This bit will automatically be cleared when **enjitter** (bit 19 of the global configuration register) is set and writes to this bit will be ignored. The transmit start of frame and word count thresholds must be set prior to setting this bit.

Register Descriptions (continued)

Receive Frame Configuration Register

Address = 0x0C

The receive frame configuration register configures the receive port of the LU5M31. This register resets to a default ISO 8802.3-1993 compliant mode.

Table 18. Receive Frame Configuration Register Bit Map

15	14	13	12	11	10	9	8
Reserved (0)	Reserved (0)	Reserved (0)	an_clear (0)	uni (0)	id[2] (0)	id[1] (0)	id[0] (0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
rxenable (0)	stripcrc (0)	rxfcen (1)	jabber (0)	crc (0)	und (0)	frag (0)	rxerr (0)

Table 19. Receive Frame Configuration Register Bit Definitions

Bit	Name	Description
15:13	Reserved	**Reserved**. These bits are reserved and must be set to zero.
12	an_clear	**Autonegotiation Clear.** When this bit is set and autonegotiation is started, **rxenable** will be automatically clear; when this bit is cleared, rxenable will hold its current value when autonegotiation is started.
11	uni	**Unicast PAUSE Frame Detection**. This bit, when set, enables the LU5M31 to detect PAUSE frames with a unicast address that is equal to the source address defined in the source address register. This capability is in addition to detection of PAUSE frames with the standard multicast address. When this bit is cleared, only PAUSE frames with the unique multicast address will be recognized.
10:8	id[2:0]	**Port ID.** These three bits are a programmable port identification for the LU5M31 and should be set in conjunction with CHSEL[2:0]. The port ID bits are included in the receive port diagnostic register which is part of the end of frame statistics. These bits reset to 0 and are writable by the host.
7	rxenable	**Receive Port Enable.** Setting this bit enables the LU5M31 to receive frames. If this bit is set while a frame is being received by the LU5M31, the frame is not accepted by the LU5M31. If this bit is cleared during the reception of a frame, the frame is completely received before the receive port is disabled. The receive event counters continue to update, regardless of the state of this bit. The receive start of frame and word count thresholds must be set prior to setting this bit.
6	stripcrc	**Strip CRC.** When this bit is set, the LU5M31 strips the last four bytes (CRC) from all frames being received.
5	rxfcen	**Flow Control Enable**. Setting this bit enables the LU5M31 to initiate and terminate flow control based on the receive FIFO high watermarks (**rxwmh** and **rxwml** in the global FIFO configuration register). An *IEEE* 802.3x Chapter 31 MAC control frame is transmitted with a PAUSE opcode. When this bit is cleared, the LU5M31 does not initiate flow control. This bit has no effect on the FLWCTL signal.
4	jabber	**Reject Jabber Frames**. When this bit is set, the LU5M31 will reject jabber frames. Transfer of the jabber frame to the host will be terminated when the error is detected.
3	crc	**Reject CRC Errored Frames.** When this bit is set, the LU5M31 will reject CRC errored frames. Transfer of the CRC errored frame to the host will be terminated when the error is detected.

Register Descriptions (continued)

Table 19. Receive Frame Configuration Register Bit Definitions (continued)

Bit	Name	Description
2	und	**Reject Undersized Frames.** When this bit is set, the LU5M31 will reject undersized frames. Transfer of the undersized frame to the host will be terminated when the error is detected.
1	frag	**Reject Fragments**. When this bit is set, the LU5M31 will reject fragments. Transfer of the fragment to the host will be terminated when the error is detected.
0	rxerr	**Reject Errored Frames**. When this bit is set, the LU5M31 will reject frames with receive errors. Transfer of a receive errored frame to the host will be terminated when the error is detected.

One-Level VLAN Tag Register

Address = 0x0E

This 16-bit register should be programmed with the 2-byte, one-level VLAN tag ID. When frames are transmitted or received, the 13th and 14th octets in the frame are compared to this register to determine if the frame is tagged with a one-level VLAN ID. If this register is 0x0000, then no match can be made.

Table 20. One-Level VLAN Tag Register Bit Map

15	14	13	12	11	10	9	8
D15 (0)	D14 (0)	D13 (0)	D12 (0)	D11 (0)	D10 (0)	D9 (0)	D8 (0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
D7 (0)	D6 (0)	D5 (0)	D4 (0)	D3 (0)	D2 (0)	D1 (0)	D0 (0)

Two-Level VLAN Tag Register

Address = 0x10

This 16-bit register should be programmed with the 2-byte, two-level VLAN tag ID. When frames are transmitted or received, the 13th and 14th octets in the frame are compared to this register to determine if the frame is tagged with a two-level VLAN ID. If this register is 0x0000, then no match can be made.

Table 21. Two-Level VLAN Tag Register Bit Map

15	14	13	12	11	10	9	8
D15 (0)	D14 (0)	D13 (0)	D12 (0)	D11 (0)	D10 (0)	D9 (0)	D8 (0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
D7 (0)	D6 (0)	D5 (0)	D4 (0)	D3 (0)	D2 (0)	D1 (0)	D0 (0)

Register Descriptions (continued)

Source Address (SA) Register

Address = 0x12, 0x14, 0x16

The source address register is used by the MAC when transmitting MAC control, unicast, pause, frame recognition, and IMATCH recognition frames. The data in this register should be written by the CPU during initialization and should contain the *IEEE* node address for the MAC port.

This *IEEE* address is transmitted with MAC control frames as the source address (SA). SA5 is the most significant byte and the least significant bit, SA5(0), is transmitted first from the PHY. SA0 is the least significant byte. Upon the completion of reset, this register defaults to 0x000000.

Table 22. SA Bit Map, Most Significant Word

Address = 0x16	
47:40	**39:32**
SA5 [7:0]	SA4 [7:0]

Table 23. SA Bit Map, Middle Significant Word

Address = 0x14	
31:24	**23:16**
SA3 [7:0]	SA2 [7:0]

Table 24. SA Bit Map, Least Significant Word

Address = 0x12	
15:8	**7:0**
SA1 [7:0]	SA0 [7:0]

Register Descriptions (continued)

Transmit Pause Frame Timer Register

Address = 0x18

The transmit pause frame timer register is used by the MAC control sublayer when a flow control is transmitted. The contents of the transmit pause frame timer register is included in the MAC control frame as the pause-time variable, where pause time represent the number of pause quantums (512 bit times/pause quantums) for which the receiving station shuts off its transmitter. The range of valid pause times can be from 0 to 65535. Therefore, all bits in the transmit pause frame timer register are valid. Upon the completion of reset (RESETb deasserted high), this register defaults to 0x0000.

Table 25. Transmit Pause Frame Timer Register Bit Map

15	14	13	12	11	10	9	8
D15 (0)	D14 (0)	D13 (0)	D12 (0)	D11 (0)	D10 (0)	D9 (0)	D8 (0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
D7 (0)	D6 (0)	D5 (0)	D4 (0)	D3 (0)	D2 (0)	D1 (0)	D0 (0)

Table 26. Transmit Pause Frame Timer Register Bit Definitions

Bit	Name	Description
15:0	DATA	**Data Bits**. The value of this register is a binary number which represents an integer between 0 and 65535 (dec).

Register Descriptions (continued)

Interrupt Identification Register

Address = 0x1A

The interrupt identification register is used to identify the cause of an interrupt (CPUINTb asserted low). When the CPUINTb signal is asserted low, the CPU must read this register first to determine which type of interrupt is present. The interrupt signal will not be cleared by reading this register. The appropriate interrupt register itself must be read to clear the interrupt.

Table 27. Interrupt Identification Register Bit Map

15	14	13	12	11	10	9	8
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)

7	6	5	4	3	2	1	0
Reserved (0)	Reserved (0)	Reserved (0)	rxcounth (0)	rxcountl (0)	txcounth (0)	txcountl (0)	general (0)

Table 28. Interrupt Identification Register Bit Definitions

Bit	Name	Description
15:5	Reserved	**Reserved.** These bits are reserved and set to 0 for normal operation.
4	rxcounth	**Receive Counter Interrupt High**. When this bit is set, an interrupt in the high word of the receive counter interrupt register was caused by a receive counter overflow. The receive counter interrupt register high word must be read to determine which counter overflowed.
3	rxcountl	**Receive Counter Interrupt Low.** When this bit is set, an interrupt in the low word of the receive counter interrupt register was caused by a receive counter overflow. The receive counter interrupt register low word must be read to determine which counter overflowed.
2	txcounth	**Transmit Counter Interrupt High**. When this bit is set, an interrupt in the high word of the transmit counter interrupt register was caused by a transmit counter overflow. The transmit counter interrupt register high word must be read to determine which counter overflowed.
1	txcountl	**Transmit Counter Interrupt Low**. When this bit is set, an interrupt in the low word of the transmit counter interrupt register was caused by a transmit counter overflow. The transmit counter interrupt register low word must be read to determine which counter overflowed.
0	general	**General Interrupt**. When this bit is set, an interrupt was caused by either a frame event or an invalid CPU interface transaction. The interrupt register must be read to determine what caused the interrupt.

Register Descriptions (continued)

Interrupt Mask Register

Address = 0x1C

This register is used to control which bits in the interrupt register drive the CPUINTb signal when they are set. Clearing any of these bits allows the corresponding interrupt bit in the interrupt register to drive the CPUINTb signal when set.

Table 29. Interrupt Mask Register Bit Map

15	14	13	12	11	10	9	8
Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
link_status (1)	dlpbkm (1)	waterm (1)	nextm (0)	basem (0)	macvm (1)	tfifom (1)	rfifom (1)

Table 30. Interrupt Mask Register Bit Definitions

Bit	Name	Description
15:8	Reserved	**Reserved**. These bits are reserved and set to 1 for normal operation.
7	link_status	**Link Status Mask.** This bit, when set, masks the link status bit in the interupt register.
6	dlpbkm	**Double Loopback Setting Mask.** This bit, when set, masks the double loop-back interrupt bit in the interrupt register.
5	waterm	**Invalid Watermark Setting Mask.** This bit, when set, masks the invalid water-mark interrupt bit in the interrupt register.
4	nextm	**Next Page Received Mask**. This bit, when set, masks the next page received interrupt bit in the interrupt register. This mask bit is cleared by default.
3	basem	**Base Page Received Mask**. This bit, when set, masks the base page received interrupt bit in the interrupt register. This mask bit is cleared by default.
2	macvm	**Memory Access Violation Mask**. This bit, when set, masks the memory access violation bit in the interrupt register
1	tfifom	**Transmit FIFO Underrun Error Mask**. This bit, when set, masks the transmit FIFO underrun error bit in the interrupt register.
0	rfifom	**Receive FIFO Overrun Mask**. This bit, when set, masks the receive FIFO over-run bit in the interrupt register.

Register Descriptions (continued)

Interrupt Register

Address = 0x1E

The events in this register cause an interrupt event to occur on CPUINTb if the interrupt is enabled by clearing the appropriate mask bit in the interrupt mask register. Multiple interrupt bits may be set if multiple events occur before this register is read. The bits in this register are cleared when this register is read. CPUINTb is also deasserted high when this register is read.

Table 31. Interrupt Register Bit Map

15	14	13	12	11	10	9	8
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
link_status (0)	dlpbk (0)	water (0)	next (0)	base (0)	macv (0)	tfifo (0)	rfifo (0)

Table 32. Interrupt Register Bit Definitions

Bit	Name	Description
15:8	Reserved	**Reserved**. These bits are reserved and set to 0 for normal operation.
7	link_status	**Link Status.** This bit is set when the link is going low.
6	dlpbk	**Double Loopback.** This bit is set when internal loopback and external loopback are selected.
5	water	**Invalid Watermark**. This bit is set when a pair of invalid watermark values are written to the global FIFO configuration register.
4	next	**Next Page Received**. This bit is set when a next page has been received by the LU5M31 during autonegotiation. The setting of this bit is used to inform the host that a new next page register should be written if required.
3	base	**Base Page Received**. This bit is set when a base page has been received by the LU5M31 during autonegotiation. The setting of this bit is used to inform the host that a next page register should be written if required.
2	macv	**Memory Access Violation.** This bit is set if an invalid address is presented on the CPUAD[7:1] bus, or if a CPU burst increments out of valid address space.
1	tfifo	**Transmit FIFO Underrun Error**. This bit is set if a transmit FIFO underrun occurs.
0	rfifo	**Receive FIFO Overrun**. This bit is set when the receive FIFO overflows.

Register Descriptions (continued)

Receive Counter Interrupt Mask Register

Address = 0x20 (Low Word), 0x22 (High Word)

This register is used to control which counters drive the CPUINTb when they roll over. Clearing these bits allows the corresponding counter to drive the CPUINTb signal when the counter overflows.

Table 33. Receive Counter Interrupt Mask Register Bit Map, High Word

31	30	29	28	27	26	25	24
Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	rxfm (1)	rxem (1)	rxdm (1)	rxcm (1)

Table 34. Receive Counter Interrupt Mask Register Bit Map, Low Word

15	14	13	12	11	10	9	8
rxbm (1)	rxam (1)	rxlongm (1)	rxtotoctm (1)	rxjabm (1)	rxfragm (1)	rxundm (1)	Reserved (1)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
rxbcm (1)	rxmcm (1)	rxoctm (1)	rxcrcm (1)	rxerrm (1)	rxmfm (1)	rxfrmm (1)	rxpausem (1)

Table 35. Receive Counter Interrupt Mask Register Bit Definitions

Bit	Name	Description
		High Word
31:20	Reserved	**Reserved.** These bits are reserved and must be set to 1 for normal operation.
19	RxFM	Frames Received (1024 to maximum size octets) Counter Mask.
18	RxEM	Frames Received (512 to 1023 Octets) Counter Mask.
17	RxDM	Frames Received (256 to 511 Octets) Counter Mask.
16	RxCM	Frames Received (128 to 255 Octets) Counter Mask.
		Low Word
15	RxBM	Frames Received (65 to 127 Octets) Counter Mask.
14	RxAM	Frames Received (64 Octets) Counter Mask.
13	RxLongM	Long Frames Received Counter Mask.
12	RxTotOctM	Total Octets Received Counter Mask.
11	RxJabM	Jabber Frames Received Counter Mask.
10	RxFragM	Fragments Received Counter Mask.
9	RxUndM	Undersized Frames Received Counter Mask.
8	Reserved	**Reserved.** This bit is reserved and must be set.
7	RxBCM	Broadcast Frames Received Counter Mask.
6	RxMCM	Multicast Frames Received Counter Mask.
5	RxOctM	Readable Octets Received Counter Mask.
4	RxCRCM	Frames Received with Bad CRC Counter Mask.
3	RxErrM	Receive Errors Counter Mask.
2	RxMFM	Missed Frames Counter Mask.
1	RxFrmM	Readable Frames Received Counter Mask.
0	RxPauseM	Pause Frames Received Counter Mask.

Register Descriptions (continued)

Receive Counter Interrupt Register

Address = 0x24 (Low Word), 0x26 (High Word)

This register can be used to identify any receive counters that have reached the overflow threshold and need to be read. When read after an interrupt, this register indicates which counter(s) caused the interrupt. This register is a clear-on-read register. Upon the completion of reset, this register defaults to the values shown in parentheses.

Table 36. Receive Counter Interrupt Register Bit Map, High Word

31	30	29	28	27	26	25	24
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	rxf (0)	rxe (0)	rxd (0)	rxc (0)

Table 37. Receive Counter Interrupt Register Bit Map, Low Word

15	14	13	12	11	10	9	8
rxb (0)	rxa (0)	rxlong (0)	rxtotoct (0)	rxjab (0)	rxfrag (0)	rxund (0)	Reserved (0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
rxbc (0)	rxmc (0)	rxoct (0)	rxcrc (0)	rxerr (0)	rxmf (0)	rxfrm (0)	rxpause (0)

Table 38. Receive Counter Interrupt Register Bit Definitions

Bit	Name	Description
		High Word
31:20	Reserved	**Reserved.** These bits are reserved and must be set to 0 for normal operation.
19	RxF	Frames Received (1024 to maximum size octets) Counter Overflow.
18	RxE	Frames Received (512 to 1023 Octets) Counter Overflow.
17	RxD	Frames Received (256 to 511 Octets) Counter Overflow.
16	RxC	Frames Received (128 to 255 Octets) Counter Overflow.
		Low Word
15	RxB	Frames Received (65 to 127 Octets) Counter Overflow.
14	RxA	Frames Received (64 Octets) Counter Overflow.
13	RxLong	Long Frames Received Counter Overflow.
12	RxTotOct	Total Octets Received Counter Overflow.
11	RxJab	Jabber Frames Received Counter Overflow.
10	RxFrag	Fragments Received Counter Overflow.
9	RxUnd	Undersized Frames Received Counter Overflow.
8	Reserved	**Reserved.** This bit is reserved and must be set to 0.
7	RxBC	Broadcast Frames Received Counter Overflow.
6	RxMC	Multicast Frames Received Counter Overflow.
5	RxOct	Readable Octets Received Counter Overflow.
4	RxCRC	Frames Received with Bad CRC Counter Overflow.
3	RxErr	Receive Errors Counter Overflow.
2	RxMF	Missed Frames Counter Overflow.
1	RxFrm	Readable Frames Received Counter Overflow.
0	RxPause	Pause Frames Received Counter Overflow.

Register Descriptions (continued)

Transmit Counter Interrupt Mask Register

Address = 0x28 (Low Word), 0x2A (High Word)

This register is used to control which transmit counters drive the CPUINTb when they roll over. Clearing these bits allows the corresponding counter to drive the CPUINTb signal when the counter overflows.

Table 39. Transmit Counter Interrupt Register Bit Map, High Word

31	30	29	28	27	26	25	24
Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	Reserved (1)	txunm (1)	txlongm (1)	txfm (1)

Table 40. Transmit Counter Interrupt Register Bit Map, Low Word

15	14	13	12	11	10	9	8
txem (1)	txdm (1)	txcm (1)	txbm (1)	txam (1)	txmcm (1)	txbcm (1)	Reserved (1)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved (1)	Reserved (1)	txoctm (1)	Reserved (1)	Reserved (1)	txpausem (1)	Reserved (1)	txfrmm (1)

Table 41. Transmit Counter Interrupt Register Bit Definitions

Bit	Name	Description
		High Word
31:19	Reserved	**Reserved.** These bits are reserved and must be set to 1 for normal operation.
18	TxUnm	Transmit Underruns Counter Overflow.
17	TxLongm	Long Frames Transmitted Counter Overflow.
16	TxFm	Frames Transmitted (1024 to maximum size octets) Counter Overflow.
		Low Word
15	TxEm	Frames Transmitted (512 to 1023 Octets) Counter Overflow.
14	TxDm	Frames Transmitted (256 to 511 Octets) Counter Overflow.
13	TxCm	Frames Transmitted (128 to 255 Octets) Counter Overflow.
12	TxBm	Frames Transmitted (65 to 127 Octets) Counter Overflow.
11	TxAm	Frames Transmitted (64 Octets) Counter Overflow.
10	TxMCm	Multicast Frames Transmitted Counter Overflow.
9	TxBCm	Broadcast Frames Transmitted Counter Overflow.
8:6	Reserved	**Reserved.** These bits are reserved and must be set to 1 for normal operation.
5	TxOctm	Total Octets Transmitted Counter Overflow.
4:3	Reserved	**Reserved.** These bits are reserved and must be set to 1 for normal operation.
2	TxPausem	MAC Pause Frames Transmitted Counter Overflow.
1	Reserved	**Reserved.** This bit is reserved and must be set to 1 for normal operation.
0	TxFrmm	Total Good Frames Transmitted Counter Overflow.

Register Descriptions (continued)

Transmit Counter Interrupt Register

Address = 0x2C (Low Word), 0x2E (High Word)

This register can be used to identify any transmit counters that have reached the overflow threshold and need to be read. When read after an interrupt, this register indicates which counter(s) caused the interrupt.

Table 42. Transmit Counter Interrupt Register Bit Map, High Word

31	30	29	28	27	26	25	24
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)
23	**22**	**21**	**20**	**19**	**18**	**17**	**16**
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	txun (0)	txlong (0)	txf (0)

Table 43. Transmit Counter Interrupt Register Bit Map, Low Word

15	14	13	12	11	10	9	8
txe (0)	txd (0)	txc (0)	txb (0)	txa (0)	txmc (0)	txbc (0)	Reserved (0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Reserved (0)	Reserved (0)	txoct (0)	Reserved (0)	Reserved (0)	txpause (0)	Reserved (0)	txfrm (0)

Table 44. Transmit Counter Interrupt Register Bit Definitions

Bit	Name	Description
		High Word
31:19	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.
18	TxUn	Transmit Underruns Counter Overflow.
17	TxLong	Long Frames Transmitted Counter Overflow.
16	TxF	Frames Transmitted (1024 to maximum size octets) Counter Overflow.
		Low Word
15	TxE	Frames Transmitted (512 to 1023 Octets) Counter Overflow.
14	TxD	Frames Transmitted (256 to 511 Octets) Counter Overflow.
13	TxC	Frames Transmitted (128 to 255 Octets) Counter Overflow.
12	TxB	Frames Transmitted (65 to 127 Octets) Counter Overflow.
11	TxA	Frames Transmitted (64 Octets) Counter Overflow.
10	TxMC	Multicast Frames Transmitted Counter Overflow.
9	TxBC	Broadcast Frames Transmitted Counter Overflow.
8:6	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.
5	TxOct	Total Octets Transmitted Counter Overflow.
4:3	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.
2	TxPause	MAC Pause Frames Transmitted Counter Overflow.
1	Reserved	**Reserved**. This bit is reserved and must be set to 0 for normal operation.
0	TxFrm	Total Good Frames Transmitted Counter Overflow.

Register Descriptions (continued)

Receive Port Diagnostic Register

Address = 0x30

This register is intended for diagnostic purposes and for end of frame statistics. It is updated every time a frame is received, and is part of the end of frame statistics that are read from the FIFO at the end of each frame. Note that multiple frames can be stored in the receive FIFO of each port. Error bits are set regardless of the state of the receive frame configuration register. Therefore, reading this register in real time does not guarantee that the status read represents the current frame being read on the data bus.

Table 45. Receive Port Diagnostic Register Bit Map

15	14	13	12	11	10	9	8
id[2] (0)	id[1] (0)	id[0] (0)	vlan2 (0)	vlan1 (0)	rxer (0)	rxjab (0)	Reserved (0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
crc (0)	und (0)	frag (0)	long (0)	phys (0)	mult (0)	brd (0)	fifo (0)

Table 46. Receive Port Diagnostic Register Bit Definitions

Bit	Name	Description
15:13	ID[2:0]	**Port ID Nibble.** The 3-bit port ID nibble bits are copied from the receive frame configuration register which is programmed by the host. These bits are not the same as the **chsel[2:0]** bits in the global configuration register.
12	VLAN2	**Two-Level VLAN Frame.** When this bit is set, the frame currently being received is tagged with a two-level VLAN ID. The 13th and 14th bytes at the frame are compared to the two-level VLAN tag register. This bit is set if there is a nonzero match.
11	VLAN1	**One-Level VLAN Frame.** When this bit is set, the frame currently being received is tagged with a one-level VLAN ID. The 13th and 14th bytes at the frame are compared to the one-level VLAN tag register. This bit is set if there is a nonzero match.
10	RxEr	**Receive Error.** This bit is set when receive errored code groups are received by the LU5M31.
9	RxJab	**Jabber on Reception.** This bit is set when the current receive frame is greater than the maximum size octets and the VLAN2 and VLAN1 bits are not set; or greater than 1538 octets and the VLAN2 bit is set; or is greater than 1522 octets and the VLAN1 bit is set and has an incorrect CRC.
8	Reserved	**Reserved.** This bit is reserved and must be set to zero for normal operation.
7	CRC	**CRC Error.** This bit is set if the frame being received has an incorrect CRC, and the octet count is greater than or equal to 64 octets and less than or equal to the maximum size octets.
6	UND	**Undersized Frame.** This bit is set if the frame being received is less than 64 octets, not including preamble, and is otherwise well formed (correct CRC).
5	FRAG	**Fragment.** This bit is set if the frame being received is less than 64 octets, not including preamble, had a valid SFD, and an invalid CRC.
4	LONG	**Frame Long Error.** This bit is set if the frame received is greater than the maximum size octets and the VLAN2 and VLAN1 bits are not set; or greater than 1538 octets and the VLAN2 bit is set; or is greater than 1522 octets and the VLAN1 bit is set and is otherwise well formed.
3	PHYS	**Physical Address.** This bit is set if the destination address of the frame received was a physical address. (A physical address may also be referred to as a unicast address.)
2	MULT	**Multicast Address.** This bit is set if the destination address of the frame received was a multicast (group) address. This does not include the broadcast address.
1	BRD	**Broadcast Address.** This bit is set if the destination address of the frame received was the broadcast address. This does not include multicast addresses.
0	FIFO	**FIFO Overrun.** This bit is set if the receive FIFO is not serviced in time, and the frame is not able to be stored in the receive FIFO.

Register Descriptions (continued)

GMII Control Register

Address = 0x32

The GMII control register provides the mechanisms to enable or disable autonegotiation, restart autonegotiation and allow for manual configuration when autonegotiation is not enabled.

Table 47. GMII Control Register Bit Map

15	14	13	12	11	10	9	8
Reserved (0)	Loopback (0)	Speed Selection (0)	AN Enable (0)	Power Down (0)	Isolate (0)	Restart AN (0)	Duplex Mode (1)

7	6	5	4	3	2	1	0
Collision Test (0)	Gig Speed Selection (0)	Negotiation Selection (1)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)

Table 48. GMII Control Register Bit Definitions

Bit	Name	Description
15	Reserved	**Reserved.**
14	Loopback	**External Loopback Enable**. This bit, when set, will cause the PHY to loopback the transmit data into the receive path for presentation to the LU5M31. When this bit is set, the internal loopback bit (**loopint** bit in the global configuration register) will be cleared. The EWRAP signal will be asserted high when this bit is set.
13	Speed Selection	**10/100 Mbits/s Speed Selection**. This bit is not supported in the LU5M31.
12	AN Enable	**Enable Autonegotiation**. This bit, when set, will enable the LU5M31 to perform autonegotiation with a link partner. Setting this bit causes autonegotiation to restart. Clearing this bit will disable autonegotiation.
11	Power Down	**Power Down PHY**. This bit is not supported in the LU5M31.
10	Isolate	**Isolate PHY**. This bit is not supported in the LU5M31.
9	Restart AN	**Restart Autonegotiation**. This bit, when set, will cause autonegotiation to restart if the an **AN Enable** bit is set. This bit is self-clearing after autonegotiation starts. This bit should be cleared for normal operation.
8	Duplex Mode	**Duplex Mode Selection**. This bit is not supported in the LU5M31.
7	Collision Test	**Collision Test**. This bit is not supported in the LU5M31.
6	Gig Speed Selection	**Gigabit Speed Selection**. This bit is not supported in the LU5M31.
5	Negotiation Selection	**Negotiation Selection**. This bit is not supported in the LU5M31.
4:0	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.

Register Descriptions (continued)

GMII Status Register

Address = 0x34

The GMII status register includes all information about modes of operation supported by the LU5M31 and the status of autonegotiation.

Table 49. GMII Status Register Bit Map

15	14	13	12	11	10	9	8
100BASET4 (0)	100BASE-X Full Duplex (0)	100BASE-X Half Duplex (0)	10 Mbits/s Full Duplex (0)	10 Mbits/s Half Duplex (0)	Reserved (0)	Reserved (0)	Reserved (0)

7	6	5	4	3	2	1	0
Reserved (0)	MF Preamble Suppress (0)	AN Complete (0)	Remote Fault (0)	AN Ability (1)	Link Status (0)	Jabber Detect (0)	Extended Capability (1)

Table 50. GMII Status Register Bit Definitions

Bit	Name	Description
15	100BASE-T4	**100Base-T4**. This function is not supported in the LU5M31.
14	100BASE-X Full Duplex	**100Base-X Full Duplex**. This function is not supported in the LU5M31.
13	100BASE-X Half Duplex	**100Base-X Half Duplex**. This function is not supported in the LU5M31.
12	10 Mbits/s Full Duplex	**10 Mbits/s Full Duplex**. This function is not supported in the LU5M31.
11	10 Mbits/s Half Duplex	**10 Mbits/s Half Duplex**. This function is not supported in the LU5M31.
10:7	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.
6	MF Preamble Suppression	**MF Preamble Suppression**. This bit is not supported in the LU5M31.
5	AN Complete	**Autonegotiation Complete**. This bit, when set, indicates that the autonegotiation process has been completed.
4	Remote Fault	**Remote Fault Condition Detected**. This bit, when set, indicates that a remote fault has been detected.
3	AN Ability	**Autonegotiation Ability**. This bit will be tied high (set) since the LU5M31 will perform autonegotiation.
2	Link Status	**Link Status**. This bit, when set, will indicate that the link is up. When cleared, this bit will indicate that the link is down.
1	Jabber Detect	**Jabber Detected**. This bit is not supported in the current version.
0	Extended Capability	**Extended Capability**. This bit is not supported in the current version.

Register Descriptions (continued)

Autonegotiation Advertisement Register

Address = 0x36

The autonegotiation advertisement register contains the advertised ability of the LU5M31. Prior to the start of autonegotiation, this register defaults to the advertised ability of the LU5M31.

Table 51. Autonegotiation Advertisement Register Bit Map

15	14	13	12	11	10	9	8
Next Page (0)	Reserved (0)	Remote Fault RF2 (0)	Remote Fault RF1 (0)	Reserved (0)	Reserved (0)	Reserved (0)	Pause PS2 (1)

7	6	5	4	3	2	1	0
Pause PS1 (1)	Half Duplex (0)	Full Duplex (1)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)

Table 52. Autonegotiation Advertisement Register Bit Definitions

Bit	Name	Description
15	Next Page	**Next Page**. This bit, when set, indicates that additional next page information is available to be transmitted. When cleared, this bit indicates that next page exchange is not desired.
14	Reserved	**Reserved**. This bit is reserved and must be set to 0 for normal operation.
13:12	Remote Fault RF2, RF1	**Remote Fault**. These 2 bits provide a remote fault encoding, indicating to a link partner a fault or error condition has occurred. The encoding of these 2 bits is defined in *IEEE* 802.3z section 37.2.1.4.
11:9	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.
8:7	Pause PS2, PS1	**Pause Encoding**. These 2 bits provide an encoding for the PAUSE bits indicating that the LU5M31 is capable of configuring the PAUSE function as defined in *IEEE* 802.3x. The encoding of these 2 bits is defined in *IEEE* 802.3z section 37.2.1.3.
6	Half Duplex	**Half Duplex**. This bit is tied low indicating that the LU5M31 cannot operate in half-duplex mode.
5	Full Duplex	**Full Duplex**. This bit is tied high indicating that the LU5M31 operates in a full-duplex mode.
4:0	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.

Register Descriptions (continued)

Autonegotiation Link Partner Ability Base Page Register

Address = 0x38

The autonegotiation link partner ability base page register contains the advertised ability of the link partner. These bits will be a direct representation of the link partner's base page register. When autonegotiation has been completed successfully, the **AN Complete** (bit 5 of the GMII status register) will be set. This is a read-only register.

Table 53. Autonegotiation Link Partner Ability Base Page Register Bit Map

15	14	13	12	11	10	9	8
Next Page (0)	Acknowledge (0)	Remote Fault RF2 (0)	Remote Fault RF1 (0)	Reserved (0)	Reserved (0)	Reserved (0)	Pause PS2(0)
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Pause PS1 (0)	Half Duplex (0)	Full Duplex (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)

Table 54. Autonegotiation Link Partner Ability Base Page Register Bit Definitions

Bit	Name	Description
15	Next Page	**Next Page**. This bit, when set, indicates that more next page information is available. When cleared, this bit indicates that next page exchange is not desired.
14	Acknowledge	**Acknowledge**. This bit, when set, is used by the autonegotiation function to indicate that the link partner has successfully received the LU5M31's base page. When cleared, it indicates that a successful receipt of the base has not been achieved.
13:12	Remote Fault RF2, RF1	**Remote Fault Encoding**. These 2 bits provide a remote fault encoding indicating to a link partner that a fault or error condition has occurred. The encoding of these 2 bits is defined in *IEEE* 802.3z Section 37.2.1.4.
11:9	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.
8:7	Pause PS2, PS1	**Pause Encoding**. These 2 bits provide an encoding for the PAUSE bits indicating the link partner's capability of configuring the PAUSE function as defined in *IEEE* 802.3x. The encoding of these 2 bits is defined in *IEEE* 802.3z section 37.2.1.3.
6	Half Duplex	**Half Duplex**. This bit, when set, indicates that the link partner has the ability to operate in half-duplex mode. When cleared, the link partner does not have the ability to operate in half-duplex mode.
5	Full Duplex	**Full Duplex**. This bit, when set, indicates that the link partner has the ability to operate in full-duplex mode. When cleared, the link partner does not have the ability to operate in full-duplex mode.
4:0	Reserved	**Reserved**. These bits are reserved and must be set to 0.

Register Descriptions (continued)

Autonegotiation Expansion Register

Address = 0x3A

This is a read-only expansion register for autonegotiation containing only the **Page Received** bit, which indicates a new page has been received, and the **Next Page Able** bit. A write to this register has no effect.

Table 55. Autonegotiation Expansion Register Bit Map

15	14	13	12	11	10	9	8
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)

7	6	5	4	3	2	1	0
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Next Page Able (1)	Page Received (0)	Reserved (0)

Table 56. Autonegotiation Expansion Register Bit Definitions

Bit	Name	Description
15:3	Reserved	**Reserved**. These bits are reserved and must be set to 0 for normal operation.
2	Next Page Able	**Next Page Ability.** This bit, when set, indicates that the LU5M31 has next page exchange ability.
1	Page Received	**Page Received**. This bit, when set, indicates that a new page has been received by the LU5M31. This bit will clear when read.
0	Reserved	**Reserved**. This bit is reserved and must be set to 0.

Register Descriptions (continued)

Autonegotiation Next Page Transmit Register

Address = 0x3C

The autonegotiation next page transmit register contains the next page to be transmitted, if required. The definition for this register is provided in *IEEE* 802.3u 28.2.4.1.6.

Table 57. Autonegotiation Next Page Transmit Register Bit Map

15	14	13	12	11	10	9	8
Next Page (0)	Reserved (0)	Message Page (0)	Acknowledge 2 (0)	Toggle (0)	Message/ Unformat-ted Code (0)	Message/ Unformat-ted Code (0)	Message/ Unformat-ted Code (0)

7	6	5	4	3	2	1	0
Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformat-ted Code (0)	Message/ Unformat-ted Code (0)	Message/ Unformat-ted Code (0)

Table 58. Autonegotiation Next Page Transmit Register Bit Definitions

Bit	Name	Description
15	Next Page	**Next Page**. This bit when set, indicates that more next page information is available. If cleared, this bit indicates that the current next page is the last next page.
14	Reserved	**Reserved**. This bit is reserved and must be set to 0.
13	Message Page	**Message Page**. This bit is used to differentiate a message page from an unformatted page. When set, it indicates that a message page has been received. When cleared, it indicates that an unformatted page has been received.
12	Acknowledge 2	**Acknowledge 2**. This bit is used by the next page function to indicate that the LU5M31 has the ability to comply with the message. When set, it indicates that the LU5M31 will comply with the message. When cleared, it indicates that the LU5M31 cannot comply with the message.
11	Toggle	**Toggle**. This bit will be ignored and the **toggle** bit will be controlled by hardware for next page synchronization.
10:0	Message/Unformatted Code	**Message/Unformatted Code Field**. These bits provide an 11-bit field that encodes 2048 possible messages. The current support message code field definitions are defined in *IEEE* 802.3u Annex 28C.

Register Descriptions (continued)

Autonegotiation Link Partner Ability Next Page Register

Address = 0x3E

The autonegotiation link partner ability next page register contains the advertised ability of the link partner's next page. The definition for this register is provided in *IEEE* 802.3u 28.2.4.1.4.

Table 59. Autonegotiation Link Partner Ability Next Page Register Bit Map

15	14	13	12	11	10	9	8
Next Page (0)	Acknowledge (0)	Message Page (0)	Acknowledge 2 (0)	Toggle (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)

7	6	5	4	3	2	1	0
Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)	Message/ Unformatted Code (0)

Table 60. Autonegotiation Link Partner Ability Next Page Register Bit Definitions

Bit	Name	Description
15	Next Page	**Next Page**. This bit, when set, indicates that more next page information is available. If cleared, this bit indicates that the current next page is the last next page.
14	Acknowledge	**Acknowledge**. This bit, when set, is used by the autonegotiation function to indicate that the link partner has successfully received the LU5M31's link code word. When cleared, it indicates that a successful receipt of the link code word has not been achieved.
13	Message Page	**Message Page**. This bit is used to differentiate a message page from an unformatted page. When set, it indicates that a message page has been received. When cleared, it indicates that an unformatted page has been received.
12	Acknowledge 2	**Acknowledge 2.** This bit is used by the next page function to indicate that the link partner has the ability to comply with the message. When set, it indicates that the link partner will comply with the message. When cleared, it indicates that the link partner cannot comply with the message.
11	Toggle	**Toggle**. This bit is used to ensure synchronization with the link partner during next page exchange. When set, this bit indicates that the previous value of the transmitted link code word was a logic 0. When cleared, it indicates that the previous value of the transmitted link code word was a logic 1.
10:0	Message/Unformatted Code	**Message/Unformatted Code Field**. These bits provide an 11-bit field that encodes 2048 possible messages. The current support message code field definitions are defined in *IEEE* 802.3u Annex 28C.

Register Descriptions (continued)

MII Data Register

MII Data Register (Address = 0x44)

The MII data register is used in conjunction with the MII control register. When reading a PHY register, data is written to this register by the PHY. When writing to a PHY register, data from this register is written to the PHY. Upon the completion of reset, RESET# deasserted, this register defaults to the values shown in parentheses.

Table 61. MII Data Register

15	14	13	12	11	10	9	8
DATA15 (0)	DATA14 (0)	DATA13 (0)	DATA12 (0)	DATA11 (0)	DATA10 (0)	DATA9 (0)	DATA8 (1)

7	6	5	4	3	2	1	0
DATA7 (0)	DATA6 (0)	DATA5 (1)	DATA4 (0)	DATA3 (0)	DATA2 (0)	DATA1 (0)	DATA0 (0)

Table 62. MII Data Register Bit Descriptions (Address = 0x44)

Bit	Name	Description
15:0	DATA	**PHY Data.** The 16-bit data value read after an MII read. The 16-bit data value to be written to the PHY before an MII write. This register should not be written if the **Busy** bit in the MII control register is 1 (if **Busy** = 1, writes are acknowledged with CPUDVal# but internally ignored).

Register Descriptions (continued)

MII Control Register

MII Control Register (Address = 0x46)

The MII control register allows the host CPU to read and write any one of the PHYs connected to the LU5M31. This register provides bits to address a particular PHY, to address a register, to set the read or write direction, and to indicate that the read or write is still in progress. Upon the completion of reset, RESET# deasserted, this register defaults to the values shown in parentheses.

Table 63. MII Control Register Bit Map

15	14	13	12	11	10	9	8
PHY4 (0)	PHY3 (0)	PHY2 (0)	PHY1 (0)	PHY0 (0)	REG4 (0)	REG3 (0)	REG2 (0)

7	6	5	4	3	2	1	0
REG1 (0)	REG0 (0)	TA1 (0)	Clk_Speed (0)	Clk_Speed (1)	Supp_Pream (0)	Write (0)	Busy (0)

Table 64. MII Control Register Bit Descriptions (Address = 0x46)

<table>
<tr><th>Bit</th><th>Name</th><th colspan="2">Description</th></tr>
<tr><td>15:11</td><td>PHY[4:0]</td><td colspan="2">PHY Address. The 5-bit address of the PHY being read or written.</td></tr>
<tr><td>10:6</td><td>REG[4:0]</td><td colspan="2">Register Address. The 5-bit address of the MII register being read or written.</td></tr>
<tr><td>5</td><td>TA1</td><td colspan="2">Indication that PHY device is not actually driving the bus when high this bit is valid. When Busy bit is sampled low after read access.</td></tr>
<tr><td>4:3</td><td>Clk_Spd[1:0]</td><td colspan="2">Clock Speed. Clock speed for management logic. Indicates the number of times to divide the CPU_CLK to generate the MDC output at a maximum frequency of 2.5 MHz. The following table should be used to program these bits:
<table>
<tr><th>Clk_Spd</th><th>CPU_CLK Range</th></tr>
<tr><td>0 0</td><td>0 MHz to but not including 20 MHz</td></tr>
<tr><td>0 1</td><td>20 MHz to but not including 40 MHz</td></tr>
<tr><td>1 0</td><td>40 MHz to but not including 80 MHz</td></tr>
<tr><td>1 1</td><td>80 MHz to but not including 160 MHz</td></tr>
</table></td></tr>
<tr><td>2</td><td>Supp_Pream</td><td colspan="2">Suppress Preamble. When high suppresses preamble generation, before every MDIO access.</td></tr>
<tr><td>1</td><td>Write</td><td colspan="2">MII Write. Setting this bit tells the PHY that this is a write operation using the MII data register. If this bit is not set, the operation is a read, placing the data in the MII data register.</td></tr>
<tr><td>0</td><td>Busy (read only)</td><td colspan="2">Busy. This bit is set by the LU5M31 when the MII control register is written, and remains set for the duration of the MII management read or write operation. (Note the MII data register must contain the valid transmit data prior to writing this MII control register.) If the Busy bit is set, writes to the MII control or data register are ignored although CPUDVal# will be asserted. The MII control register should not be written while the bit is set. When writing this register, Busy should be zero.</td></tr>
</table>

Register Descriptions (continued)

PCS Status Register

PCS Status Register (Address = 0x48)

The PCS status register contains the recorded events in the PCS of the LU5M31.

Table 65. PCS Status Register Bit Map (Address = 0x48)

15	14	13	12	11	10	9	8
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)

7	6	5	4	3	2	1	0
Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	Reserved (0)	SIG_DET (1)	Sync_Status (0)	AN_IN_Prog (0)

Table 66. MII Control Register Bit Descriptions

Bit	Name	Description
15:3	Reserved	**Reserved.** These bits are reserved and must be zero.
2	SIG_DET	**Signal Detect.** This bit provides an indication from the signal detect pin that the PMD is detecting light.
1	Sync_Status	**Synchronization Status.** This bit, when set, indicates that three consecutive valid COMMA patterns have been received by the PHY and syncronization has been acquired.
0	AN_IN_Prog	**Autonegotiation in Process.** This bit, when set, indicates that autonegotiation is in progress.

Functional Description

Event Counters

The LU5M31 provides an extensive list of network event counters which perform the following:

- All counters clear to zero upon a hardware reset (RESETb asserted low) or a software counter reset (**cntres** bit 0 set in the global configuration register).
- The counters count all events even when the receive port is disabled by **rxenable** (bit 7 of the receive frame configuration register) or the receive FIFO is in an overflow condition.
- When a counter reaches its maximum value and rolls over, an interrupt signal (CPUINTb) is generated and the counter continues to count from 1.
- Internal arbitration allows counters to be read efficiently by the host and updated by the LU5M31 concurrently.
- The event counters in the LU5M31 are derived from the RMON MIB and from *IEEE* 802.3z, Chapter 30.

64 Octet Frames Transmitted

This 48-bit counter counts the frames transmitted by the LU5M31 (including frames with errors) that are 64 octets in length, including CRC but not including preamble bits.

65—127 Octet Frames Transmitted

This 48-bit counter counts the frames transmitted by the LU5M31 (including frames with errors) that are greater than or equal to 65 octets and less than or equal to 127 octets, including CRC but not including preamble bits.

128—255 Octet Frames Transmitted

This 32-bit counter counts the frames transmitted by the LU5M31 (including frames with errors) that are greater than or equal to 128 octets and less than or equal to 255 octets, including CRC but not including preamble bits.

256—511 Octet Frames Transmitted

This 32-bit counter counts the frames transmitted by the LU5M31 (including frames with errors) that are greater than or equal to 256 octets and less than or equal to 511 octets, including CRC but not including preamble bits.

512—1023 Octet Frames Transmitted

This 32-bit counter counts the frames transmitted by the LU5M31 (including frames with errors) that are greater than or equal to 512 octets and less than or equal to 1023 octets, including CRC but not including preamble bits.

1024—Maximum Size Octet Frames Transmitted

This 32-bit counter counts the frames transmitted by the LU5M31 (including frames with errors) that are greater than or equal to 1024 octets and less than or equal to maximum size octets, including CRC but not including preamble bits. The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

Long Frames Transmitted

This 32-bit counter counts the frames that are transmitted by the LU5M31 that are greater than maximum size octets, including the frame check sequence (CRC) and do not have any errors. The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when the **extbin** bit is set.

Total Good Frames Transmitted

This 48-bit counter counts the good frames that are transmitted out of the LU5M31 and does not include those frames that are transmitted with errors (i.e., due to FIFO underruns).

MAC Pause Frames Transmitted

This 16-bit counter counts the pause frames transmitted successfully.

Multicast Frames Transmitted

This 32-bit counter counts the good multicast frames that are transmitted from the LU5M31. A multicast frame is identified by having a 1 in the least significant bit of the most significant byte of the destination address (the first bit transmitted on the media is a 1). Frames transmitted to the broadcast address are not included in this count. This count does not include multicast frames that are transmitted with errors (i.e., partial frames due to FIFO underruns).

Functional Description (continued)

Broadcast Frames Transmitted

This 32-bit counter counts the good broadcast frames that are transmitted from the LU5M31. A broadcast frame is a special type of multicast frame where all the bits of the destination address are 1. This count does not include broadcast frames that are transmitted with errors (i.e., partial frames due to FIFO underruns).

Total Octets Transmitted

This 48-bit counter counts the encoded octets that are transmitted out of the LU5M31 including the four octets of the frame check sequence (CRC). An octet is defined as 8 bits of transmitted data and is counted after SFD is transmitted. This count does not include framing octets (preamble). This does include octets that are transmitted before a FIFO underrun occurs. This count is the exact number of encoded octets transmitted on the media by the LU5M31.

Transmit Underruns

This 16-bit counter counts the number of times the transmit FIFO was underrun.

64 Octet Frames Received

This 48-bit counter counts the frames received by the LU5M31 (including frames with errors) that have exactly 64 octets, including CRC but not including preamble bits.

65—127 Octet Frames Received

This 48-bit counter counts the frames received by the LU5M31 (including frames with errors) that are greater than or equal to 65 octets and less than or equal to 127 octets, including CRC but not including preamble bits.

128—255 Octet Frames Received

This 32-bit counter counts the frames received by the LU5M31 (including frames with errors) that are greater than or equal to 128 octets and less than or equal to 255 octets, including CRC but not including preamble bits.

256—511 Octet Frames Received

This 32-bit counter counts the frames received by the LU5M31 (including frames with errors) that are greater than or equal to 256 octets and less than or equal to 511 octets, including CRC but not including preamble bits.

512—1023 Octet Frames Received

This 32-bit counter counts the frames received by the LU5M31 (including frames with errors) that are greater than or equal to 512 octets and less than or equal to 1023 octets, including CRC but not including preamble bits.

1024—Maximum Size Octet Frames Received

This 32-bit counter counts the frames received by the LU5M31 (including frames with errors) that are greater than or equal to 1024 octets, and less than or equal to maximum size octets, including CRC but not including preamble bits. The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

Long Frames Received

This 32-bit counter counts the frames that are received by the LU5M31 which are longer than maximum size octets, including the frame check sequence (FCS), and without errors (i.e., good CRC and valid code group while the frame is being received). The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

Receive Errors

This 16-bit counter counts the frames that are received with an errored code group detected.

Functional Description (continued)

Total Readable Frames Received

This 48-bit counter counts the good frames that are received by the LU5M31. This count includes broadcast and multicast frames. Frames with errors are not included in this count, but are counted individually based on the error.

Frames that are greater than or equal to 64 octets and less than or equal to maximum size octets, and are otherwise well-formed (i.e., correct CRC and no errored code group during reception of frame) are counted. The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

Jabber Frames Received

This 16-bit counter counts the no receive-errored frames that are received by the LU5M31 which are greater than maximum size octets, including the frame check sequence (FCS), and have an invalid CRC. The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

MAC Pause Frames Received

This 16-bit counter counts the good MAC pause frames that are received. This count does not include pause frames with errors.

Frames Received with Bad CRC

This 32-bit counter counts the frames that are received by the LU5M31 where the frame check sequence (FCS) is invalid. The frame must be greater than or equal to 64 octets and less than or equal to maximum size octets, including FCS but not including preamble bits.

If an errored code group is received from the PHY during the reception of a frame, this counter will be incremented as well as the receive errors counters. The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

Multicast Frames Received

This 32-bit counter counts the good (greater than or equal to 64 octets and less than or equal to maximum size octets, no CRC errors, no errored code group during frame reception) multicast frames that are received by the LU5M31.

Multicast frames that have errors are not included in this count. Broadcast frames received are not included in this count.

The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

Broadcast Frames Received

This 32-bit counter counts the good (greater than or equal to 64 octets and less than or equal to maximum size octets, no CRC errors, no errored code group during frame reception) broadcast frames that are received by the LU5M31. This counter does not include broadcast frames that have errors.

The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged framed and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

Total Octets Received

This 48-bit counter counts the octets that are received by the LU5M31. The octet count includes the frame check sequence (FCS) as well as octets from erroneous frames. Octets are 8-bits in length of receive data received after the SFD.

Functional Description (continued)

Total Readable Octets Received

This 48-bit counter counts the octets that are received by the LU5M31. The octet count includes the frame checksum (FCS) but does not include octets from erroneous frames.

Octets are 8-bit bytes of receive data received after the SFD. Octets are counted from frames that are greater than or equal to 64 octets and less than or equal to maximum size octets and are otherwise well-formed (i.e., correct CRC, no errored code group during frame reception).

The upper valid frame limit of maximum size octets is adjusted for VLAN frames (limit = 1538 for two-level VLAN tagged frames and 1522 for one-level VLAN tagged frames) and is adjusted to 1538 for all frames when **extbin** (bit 20 of the global configuration register) is set.

Missed Frames

This 16-bit counter counts the frames that would otherwise be received by the LU5M31 but are not able to be stored in the receive FIFO because it has overflowed. If the LU5M31 is programmed to accept undersized or long frames they will be included in this count.

Fragments Received

This 32-bit counter counts the frames that are received by the LU5M31 which are shorter than 64 octets and have an invalid CRC. SFD must be received.

Undersized Frames Received

This 32-bit counter counts the frames that are received by the LU5M31 which are shorter than 64 octets including the frame check sequence (FCS), and do not have any errors, and no errored code group during reception of the frame. SFD must be received for the CRC calculation.

Functional Description (continued)

Host Interface

The LU5M31 provides a high-speed host bus interface capable of transferring data at up to 4.2 Gbits/s. The LU5M31 uses request signals from the receive and transmit FIFOs to initiate a transaction. These request signals are controlled by threshold settings in the global FIFO configuration register. All other control is managed by the host.

64-Bit Bidirectional Bus Operation

The host bus interface of the LU5M31 provides a means to transfer data between one or more LU5M31s, multiple 10/100 Mbits/s MACs, and a host ASIC. The host would be designed to provide all master control, whereas the LU5M31 operates as a slave device.

The LU5M31 will request either transmit or receive service based on the current status of the transmit or receive FIFO. The host must respond by asserting the proper signals and by either reading or writing data. The read function cannot be prempted.

The exact read and write functions are described separately in the proceeding sections.

The signals that request read or write service are the RXREQb and the TXABLEb. These signals will transition from an inactive high state to an active-low state in order to request service. They will be deasserted once the host responds.

When responding to a transmit write or receive read, the host must drive the following common signals:

- PFCSb must be driven low to select the device.
- CHSEL[2:0] can be driven to a binary value that matches the device address set by the **chsel[2:0]** bits (bits 7:5 in the global configuration register).
- RTSEL must be driven either high for a read transaction or low for a write transaction.

Channel Assignment

There are two ways that the LU5M31 can be selected by the system. The global configuration register provides two bits, bit 9 **txchsel** and bit 8 **rxchsel**, which allow the system to assign a channel number to the LU5M31 for receive and transmit operations.

When both of these bits are cleared to zero, the LU5M31 will be selected when PFCSb is asserted low. With PFCSb asserted high and both of these bits set high, the LU5M31 will compare the value on the CHSEL[2:0] signals to the value stored in **chsel[2:0]** (bits 7:5 in the global configuration register). If a match is made, the device will be selected. Otherwise, the proceeding transaction will not be completed by the LU5M31.

This function allows multiple LU5M31 devices to be uniquely selected while sharing the same chip select.

The **txchsel** and **rxchsel** bits can be enabled independently of each other.

The LU5M31 also provides a 3-bit programmable port ID **id[2:0]** (bits 10:8 in the receive frame configuration Register). This ID is unrelated to the **chsel[2:0]** bits and is provided as part of the end of frame statistics.

Figure 4 shows two LU5M31 devices used together in a system along with a 10/100 Mbits/s MAC. In this case, the LU5M31 devices share a chip select (PFCSb) and are distinguished as channel 0 and channel 1.

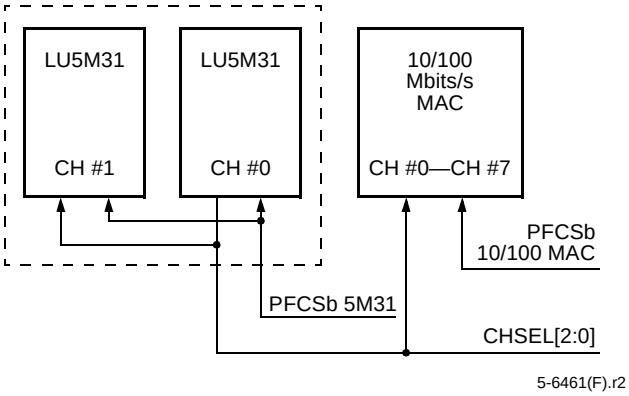


Figure 4. Channel Assignment

Functional Description (continued)

Host Write Operation

The LU5M31 uses a 2 Kbyte transmit FIFO to queue Ethernet frames for transmission through the MAC and PCS interface on to the media. The interface between the host and the transmit FIFO is the 64-bit bidirectional bus.

In order to start and complete a write to the transmit FIFO, the host must adhere to the following process:

1. Configure the device through the configuration register interface:
   a. Set **txwcth** and **txsfth** thresholds (in the global FIFO configuration register) and match system requirements.
   b. Set enable bit **txenable** (bit 0 in transmit configuration register).
   c. Configure/enable transmit port through the transmit configuration register.
2. Wait for TXABLEb to be asserted by the LU5M31.
3. Assert the following signals for the duration of the bus cycle:
   - PFCSb is low.
   - CHSEL[2:0] is the valid channel. (if **txchsel** = 1).
   - RTSEL is low.
   - TXREQb is low.
   - SOF is high for first word, low for others.
   - EOF is high for last word, low for others.
   - DATA[63:0] is the valid data.
   - VB[7:0] is nonzero for valid data transfer.
   - TXCRC is high while EOF is high, if CRC calculation is required.
4. Repeat burst writes (steps 2 and 3) until entire frame is written to the FIFO.

Figure 5 shows the functional timing diagram for the write procedure. As shown in the figure, the host writes transmit frame data to the LU5M31 in burst fashion when TXABLEb is asserted. All signals must be set up to the rising edge of HCLK.

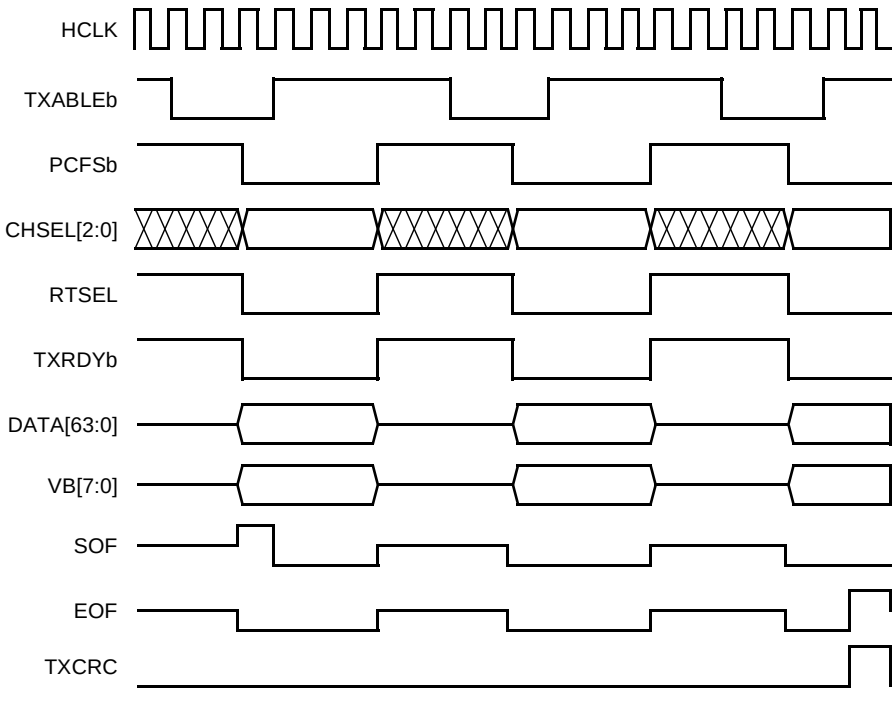


Figure 5. TXWRITE: Timing Diagram for Write Procedure

Functional Description (continued)

Host Write Process

Valid data can be presented on the bus during the same cycle that the host selects the LU5M31 by asserting the PFCSb and CHSEL[2:0]. For a transmit, the host must set the direction of the data bus by driving the RTSEL low for the duration of the transaction.

The host needs to validate frame data in two ways:

- The first word of the frame must be written to the FIFO while SOF is asserted.
- The last word must be written with EOF asserted.

Transmit FIFO Thresholds

The LU5M31 uses two thresholds to control transmit operations. Both thresholds can be set using 5 bits in the global FIFO configuration register:

- Transmit Word Count Threshold, **trwcth** (bits 26:22). The assertion of TXABLEb is controlled by the transmit word count threshold **trwcth** (bits 26:22). As the FIFO empties during a transmission, TXABLEb will be asserted as the FIFO empties through this threshold.

  After reset, when the FIFO is empty, this threshold is met by default and TXABLEb will be asserted when **txenable** (bit 0 of the transmit frame configuration register) is set. When writing to the FIFO the host can write any number of words up to the threshold value.

- Transmit Start of Frame Threshold, **txsfth** (bits 4:0). This threshold can be set between 8 and 256 words in increments of eight. Hence, the minimum threshold value is 64 bytes and the maximum is 2048 bytes.

  When writing a new frame to the transmit FIFO, the transmission of the frame will start when this threshold is met or an EOF is written to the FIFO. In the event that multiple frames are present in the FIFO, **txsfth** needs to be met for each frame.

Data written to the FIFO before SOF is written will be ignored. If EOF is not written at the end of the frame, an underrun will occur. All valid bytes (VB) within the frame must be written to the FIFO with the appropriate VB[7:0] bits set.

Any invalid bytes written to the FIFO will be discarded by the LU5M31. VB[7:0] must be nonzero while SOF and EOF are written, otherwise, the transaction will be ignored.

Table 67 details the legal valid byte (VB) combinations.

Table 67. Valid Byte Reference

VB [7:0]	DATA [63:56]	DATA [55:48]	DATA [47:40]	DATA [39:32]	DATA [31:24]	DATA [23:16]	DATA [15:8]	DATA [7:0]
00000001	—	—	—	—	—	—	—	valid
00000011	—	—	—	—	—	—	valid	valid
00000111	—	—	—	—	—	valid	valid	valid
00001111	—	—	—	—	valid	valid	valid	valid
00011111	—	—	—	valid	valid	valid	valid	valid
00111111	—	—	valid	valid	valid	valid	valid	valid
01111111	—	valid	valid	valid	valid	valid	valid	valid
11111111	valid	valid	valid	valid	valid	valid	valid	valid
11111110	valid	valid	valid	valid	valid	valid	valid	—
11111100	valid	valid	valid	valid	valid	valid	—	—
11111000	valid	valid	valid	valid	valid	—	—	—
11110000	valid	valid	valid	valid	—	—	—	—
11100000	valid	valid	valid	—	—	—	—	—
11000000	valid	valid	—	—	—	—	—	—
10000000	valid	—	—	—	—	—	—	—
00000000	—	—	—	—	—	—	—	—

Functional Description (continued)

Enable and Reset

The transmit port can be disabled and reset through two register bits. By default, the transmit port is disabled after reset, and can be enabled by setting **txenable** (bit 0 in the transmit frame configuration register). When this bit is cleared, the transmit port will be disabled and TXABLEb will be held high, preventing host access. If enable is cleared while a transmit is in progress, the transmission will be allowed to finish before the port is disabled. If necessary, the transmit port can also be reset. When **txres** (bit 2 in the global configuration register) is set, the transmit FIFO will be cleared.

All state machines will return to IDLE and operation will cease immediately.

Transmit Halt

The LU5M31 will transmit frames of any size. If **pad** (bit 3 in the transmit frame configuration register) is set. Frames having less than 64 bytes are automatically padded with 0xAA whereby making the frame length at least 64 bytes, including CRC. The LU5M31 will calculate and append CRC to frames less than 64 bytes when the **pad** bit is set, regardless of the values of the **hwcrc** and **crc** bits or the TXCRC signal.

Table 68 shows the new frame lengths when padding is enabled or disabled.

Table 68. Padding Conditions

Pad	CRC or TXCRC	4-Byte Value Appended	New Frame Length
Frame Lengths <= 60 Bytes			
0	0	None	Frame Length
0	1	New CRC	Frame Length + 4 Bytes
1	X	New CRC	64 Bytes
Frame Lengths of 61, 62, 63 Bytes			
0	0	None	Frame Length
0	1	New CRC	Frame Length + 4 Bytes
1	X	New CRC	Frame Length + 4 Bytes

CRC Generation

The transmit FIFO has the ability to transmit a frame with or without the CRC frame-check sequence (FCS) calculated and appended by the LU5M31. This option can be turned on or off in one of two ways based on the status of **hwcrc**, bit 2 of the transmit frame configuration register:

- If **hwcrc** is cleared, **crc** (bit 1 of the transmit frame configuration register) determines whether the transmitter will calculate and append CRC. The **crc** bit should not be changed dynamically.
- If the **hwcrc** bit is set, TXCRC is qualified when the host bus is enabled. The LU5M31 will calculate and append CRC if TXCRC is high while EOF is asserted. This method allows the host to calculate and append CRC dynamically.

When a transmit underrun occurs, the LU5M31 transmits a frame with a CRC error regardless of the state of the **crc** bit or the TXCRC signal and no padding will be added.

Loopback

The LU5M31 supports loopback operation for diagnostic purposes. Loopback should not be used under normal operating conditions. Loopback through the PHY is programmable by setting **Loopback** (bit 14 in the GMII control register). Loopback within the LU5M31 is programmable via setting **loopint** (bit 10 in the global configuration register). If both these bits are set, **dlpbk** (bit 6 in the interrupt register) will be set, indicating a loopback conflict, and if this interrupt is not masked by **dlpbkm** (bit 6 in the interrupt mask register), CPUINTb will be asserted.

For internal loopback, the loopback data is not transmitted across the PCS interface. Rather, IDLE (/I2/) patterns will be transmitted by the LU5M31, thus internal loopback should only be enabled during interframe transmission. As the loopback data is received by the transmit portion of the PCS logic, it is looped back into the receive side and transferred to the host.

In loopback, all configuration is manual. The CRC can be generated based on the state of the TXCRC signal or **crc** (bit 1 in the transmit frame configuration register). The CRC is also checked when the loopback frame is received.

All other statistics are monitored by the receiver and statistics are appended to the end of the frame. Event counters are only updated when **eninc** (bit 11 in the global configuration register) is set. Loopback allows testing of the host by writing and reading the same frame to and from the LU5M31. During loopback, flow control packets initiated by the FLWCTL signal or a receive FIFO overflow will not be transmitted. When the **Loopback** bit is set, the EWRAP signal will be asserted high automatically, indicating that the PHY should not transmit data onto the network but should loop transmit data back onto the receive path.

Functional Description (continued)

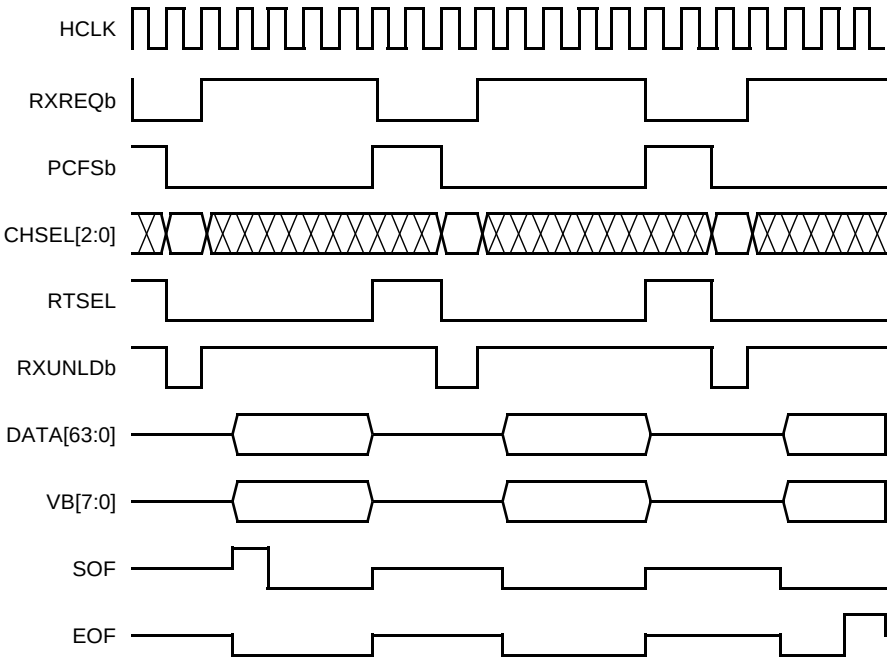
Host Read Operation

The LU5M31 uses an 8 Kbyte receive FIFO to store frames as they are received through the PCS interface. The interface between the receive FIFO and the host is the 64-bit bidirectional bus. In order to start and complete a read of the receive FIFO, the host must follow the following process:

1. Configure the device through the CPU register interface:
   a. Set **rxwcth** and **rxsfth** thresholds (in the global FIFO configuration register) and match system requirements.
   b. Set **rxenable** (bit 7 in the receive frame configuration register).
2. Wait for RXREQb to be asserted by the LU5M31.
3. Assert the following signals for the duration of the burst:
   - PFCSb is low.
   - CHSEL[2:0]: valid channel (if **rxchsel** = 1).
   - RTSEL is high.
   - RXUNLD is low.

   The following signals are driven by the LU5M31:
   - SOF is high for first word, low for others.
   - EOF is high for last word, low for others.
   - DATA[63:0] is valid data.
   - VB[7:0] is nonzero for valid data transfer.
4. Repeat burst reads (steps 2 and 3) until entire frame is read from the FIFO.

Figure 6 shows the functional timing diagram for the read procedure.



5-6462(F).r2

Figure 6. RXREAD: Timing Diagram for Read Procedure

Functional Description (continued)

Host Read Process

As shown in Figure 6, the LU5M31 will provide receive data in burst fashion. The host selects the device and channel and then asserts the RXUNLDb low to initiate the read. All the signals on the host are synchronous to the rising edge of HCLK.

After RXREQb is asserted by the LU5M31, the host can initiate a read by asserting PFCSb, CHSEL[2:0], RXUNLDb, and by driving RTSEL high. An internal pipeline will drive data out of the FIFO to the host after a two cycle HCLK delay from when RXUNLDb is asserted. The LU5M31 will keep the bus in 3-state during this start-up delay.

The LU5M31 will present a burst of data to the host. The burst length is determined by the number of words in the receive word count threshold (**rxwcth**). This burst can not be pre-empted.

On the first word of each frame, the LU5M31 will assert SOF. Data will always be aligned to the upper byte lane of the data bus, DATA[63:56], and will only contain invalid bytes when EOF is asserted while the last word of the frame is presented.

Received Statistics

The LU5M31 will present a status word on the rising edge of HCLK after EOF is driven, provided EOF does not occur on the last word of a burst. Statistics will always be word aligned such that VB = F0. The upper 2 bytes of statistics will be the byte count of the frame while the lower two bytes of statistics are defined by the port diagnostic register.

If the receive frame is greater than 64 Kbytes, the byte count will not be valid. If EOF is present on the last word of a burst, the host must reassert RXUNLDb in order to read the statistics. RXREQb will not be indicated solely to remove statistics and data from the proceeding frame will not be provided in the same burst as statistics.

If **enimatch** (bit 4 in the global configuration register) is set, the LU5M31 will compare the destination address (DA) of every received frame with the source address (SA) stored in the SA registers. If a nonzero comparison is made, the LU5M31 will assert the IMATCH signal for one HCLK pulse while SOF is high.

Receive FIFO Thresholds

The LU5M31 uses two thresholds to control receive FIFO host interface operations. Both thresholds can be set from 1 to 32 words in the global FIFO configuration register. This allows for programmable thresholds from 8 to 256 bytes in 8-byte increments:

- The first threshold **rxsfth** controls when RXREQb will be asserted relative to the beginning of each frame. Each time a new frame is received, the first assertion of RXREQb will occur when **rxsfth** is met.
- The second threshold **rxwcth** controls each successive assertion of RXREQb and also determines the burst size. Therefore, receive word count threshold is less than the receive start of frame threshold.

Enable and Reset

The receive port can be disabled and reset through two register bits. By default, the receive port is disabled after reset, and can be enabled by setting the enable bit in the receive frame configuration register. When this bit is cleared, the receive port will be disabled and RXREQb will remain deasserted high, preventing host access.

If enable is cleared while a receive is in progress the LU5M31 will complete the reception before the port is disabled. If necessary, the receive port can also be reset. When **rxres** (bit 1 in the global configuration register) is set the receive FIFO will be cleared, all state machines will return to IDLE and the operation will cease immediately.

Receive Multiple Frames

The receive FIFO can buffer multiple frames. The LU5M31 is capable of receiving frames with a minimum IFG time. If the IFG is less than 64 bit times, the LU5M31 will buffer the frames in the FIFO as normal however, the accuracy of the event counters cannot be guaranteed.

Functional Description (continued)

Receive FIFO Overflow

In the event that the receive FIFO becomes full and frames cannot be stored, a FIFO overflow error will occur and the missed frames counter is incremented for each frame that is received until the overflow condition is cleared. Receive event counters are maintained while the receive FIFO is in an overflow state, so that statistics are correctly maintained. Frames will be received again immediately after the overflow condition is cleared.

The LU5M31 provides a maskable interrupt bit for a receive FIFO overflow in the interrupt register. When the receive FIFO overflows, the **rfifo** bit in the interrupt register is set, and if not masked, the CPUINTb signal will be asserted low. The receiver will ignore all subsequent data from that frame and the LU5M31 will initiate flow control.

Receive Frame Rejection

The LU5M31 has the capability to reject certain types of frames from being buffered and forwarded to the host. These frames include fragments, jabber frames, undersized frames, frames with a bad CRC and frames with receive errors (RX_ER asserted across the internal GMII interface).

The receive frame configuration register uses the following bits to enable the rejection function: **rxerr, frag, und, crc,** and **jabber** (bits 0 through 4, respectively). If any of these bits are set and the LU5M31 receives a frame with one of the corresponding errored conditions, the frame will be discarded.

If a host transaction is in progress for that frame, it will be ended prematurely and the corresponding error bit will be set in the status word. If a host read has not yet been initiated, the frame will be rejected and the LU5M31 will not assert RXREQb for that frame. In both cases, the receive event counters will update properly.

Stripping CRC from Receive Frames

The LU5M31 provides users with the capability of removing the CRC frame check sequence (FCS) from frames that are being received, before being passed to the host.

The **stripcrc** (bit 6 in the receive frame configuration register) determines whether or not the CRC is removed from every receive frame. If **stripcrc** is low, the CRC is not removed. The **stripcrc** bit is evaluated at the start of a frame and any changes in **stripcrc** value during frame reception will be ignored by the CRC stripping function. CRC will not be stripped from received frames less than 6 bytes in length.

Functional Description (continued)

Link Interface

Physical Coding Sublayer

The LU5M31 provides an *IEEE* 802.3z D5 compliant 10-bit physical coding sublayer (PCS) interface. This interface will present 10-bit code groups together with a 125 MHz transmit clock (TX_CLK), which is derived (buffered) from the REF_CLK input. In addition, it can receive 10-bit code groups, together with two 62.5 MHz receive clocks which are 180 degrees out of phase in accordance with *IEEE* 802.3z D5.

Link Synchronization

The LU5M31 implements synchronization of received code groups in compliance with *IEEE* 802.3z. Once bit synchronization has been achieved by the PHY, the LU5M31 will acquire synchronization after detecting three consecutive valid COMMA patterns sent by the PHY in the even code group position. With synchronization acquired, it is possible to start autonegotiation.

After acquiring synchronization, if four consecutive invalid code groups are received, the synchronization process will restart automatically. Hysteresis is built-in to prevent restarting of the synchronization process due to receiving invalid code groups and to reduce the occurrence of random bit errors restarting the synchronization process.

Autonegotiation (AN)

The autonegotiation process is implemented as depicted in *IEEE* 802.3z D5 Figure 37-6. Autonegotiation provides an automatic process for two devices to share configuration information and resolve to operate at the highest of their common abilities. Manual configuration can also be accommodated to force a certain level of operation via the management control register and advertisement register, and autonegotiation can be disabled via the management control register.

A subset of the 32 management registers is supported in the LU5M31. The seven management registers supported are registers 0, 1, 4, 5, 6, 7, and 8. These registers are defined in the *IEEE* 802.3z standard within clauses 22, 28, and 37. The management registers are defined within the processor register map and can be read written via the processor interface for configuration and base and next page exchange. The LU5M31 supports base page exchange and multiple next page exchanges.

Upon reset, AN will be disabled; **AN Enable** (bit 12 of the GMII control register) will default to 0 on reset. AN will be enabled and will start once **AN Enable** is set to a 1. AN can be restarted at any time manually once **AN Enable** is set by setting **Restart AN** (bit 9 in the GMII control register). **Restart AN** is self-clearing once the AN process is restarted. AN can be disabled at any time by clearing the **AN Enable** bit.

Once **AN Enable** is set, there are many different ways to start AN. AN will be started or restarted if requested by another device transmitting configuration code groups. AN can also be restarted if synchronization is lost or SIGDET goes low for more than 10 ms. If **rxres** (bit 1 in the global configuration register) is set while **AN Enable** is set, AN will also restart.

Once AN has been started, the LU5M31 will initially send config words containing all 0s for a duration of time that can be programmed by **timdur**, bits [13:12] in the global configuration register (this time duration is referred to as the link timer duration). The link timer duration defaults to 10 ms, but can be set higher to 16 ms which is still within the standard's limits. The link timer duration can also be set to much smaller values for diagnostic purposes. After the link timer duration has expired, the contents of the AN advertisement register will be transmitted within the config codes as the LU5M31's base page. The AN advertisement register should be loaded prior to AN being enabled.

When the LU5M31 recognizes that it has received three matching base pages from its link partner, ignoring **acknowledge** (bit 14 in the acknowledge link partner ability base page register), it will automatically send its base page with the **acknowledge** bit set, that is, the **acknowledge** bit is controlled by the AN logic. After the LU5M31 recognizes that three matching base pages with the **acknowledge** bit set have been received, and the base page matches the three matching base pages it received prior, the base page value received will be loaded into the AN link partner ability base page register.

Also at this point, **Page Received** (bit 1 in the AN expansion register) will be set and **base** (bit 3 in the interrupt register) will be set. If the **base** bit is not masked, CPUINTb will also go low. **Page Received** in the AN expansion register is cleared on read. The interrupt register may be more useful since there are separate interrupts for **base** and **next** (bits 3 and 4, respectively, in the interrupt register) being received. If AN was restarted during next page exchange, management would know by the **base** interrupt bit being set, not the **next** interrupt bit.

Functional Description (continued)

If both devices are capable of exchanging next pages and next page exchange is desired by one or both of the devices, then after the base or a next page is received (use interrupt or poll **Page Received**), the AN next page transmit register must be loaded. The value written to the **toggle** (bit 11 in the AN next page transmit register) is not used, but is automatically set or cleared by the AN logic. The **Acknowledge 2** (bit 12 in the autonegotiation next page transmit register) bit must be controlled by management. If the link partner desires to continue sending next page information when the LU5M31's management does not, then management must write null message coded pages to the AN next page transmit register.

After the AN next page transmit register has been written, next page exchange will occur similarly as described for base page exchange. The link partner's next page information will be loaded into the AN link partner ability next page register. Once loaded, **Page Received** in the AN expansion register will be set and the **next** bit in the interrupt register will be set. If the **nextm** (bit 4 in the interrupt mask register) was not set, CPUINTb will also go low.

If next page exchange is not desired or supported by both devices, then AN will be completed after base page exchange, else AN will continue until neither device desires to send next pages. Upon completion of page exchange, **AN Complete**, bit 5 in the GMII status register, will be set. Management has the responsibility of performing priority resolution and reconfiguring the **rxfcen** and **txfcen** bits if necessary after AN is complete. AN is complete after final base or next page exchange, and after the LU5M31 has detected valid IDLE patterns after a link timer duration. Once AN is complete, the LU5M31 is capable of transmitting packets.

Jitter Tests

The LU5M31 supports five different automatic jitter testing patterns as specified in Annex 36A of 802.3z D5. The jitter patterns can be transmitted by setting **enjitter** (bit 19 in the global configuration register, and setting **jitter**, bits [18:17], and setting jlength, bit 16 in the global configuration register) to the proper encodes. The jitter test patterns are for diagnostic use and should be enabled only during interframe transmission.

Functional Description (continued)

Embedded Gigabit MAC

The LU5M31 contains an embedded full-duplex gigabit MAC (GMAC). The GMAC functionality is split between a MAC control sublayer and a MAC layer. The MAC control sublayer responds to data and control requests from the host interface (considered the MAC client) and provides the MAC layer with transmit frame and receive frame function calls. Transmit and receive functions operate concurrently and for the most part independently. The only exception being interaction between the receive and transmit functions at the MAC control sublayer in support of flow control.

The GMAC and PCS functional blocks are intended to be complete in terms of implementation of the *IEEE* 802.3z D5 standard. The PCS includes all autonegotiation functions defined in 802.3z for full-duplex operation.

Flow Control

The LU5M31 supports *IEEE* 802.3x frame-based flow control in both the transmit and receive direction. It can initiate flow control in one of two ways. The host can request flow control by asserting the FLWCTL signal or by responding to a receive FIFO fill situation.

In order to reduce the risk of receive FIFO overflow, two programmable watermarks are provided which initiate flow control through an XON/XOFF protocol:

- The **rxwmh** Watermark: Once the level in the receive FIFO reaches a predetermined watermark, **rxwmh** (bits 21:16 set in the global FIFO configuration register) and the **rxfcen** bit (bit 5 in the receive frame configuration register) is set, flow control is initiated by the LU5M31 (XOFF).
- The **rxwml** Watermark: When the level in the receive FIFO drops below the second watermark **rxwml** (bits 15:10 in the global FIFO configuration register), then flow control is terminated (XON). Flow control is terminated by sending a PAUSE frame with the number of pause quantums set to zero.

These two watermarks are separated to prevent oscillation of the flow control function. Care should be taken in setting these fields. The LU5M31 will not initiate flow control if the high watermark is set to a smaller value than the low watermark.

If these two watermarks are set to the same value, flow control will be initiated as described above but oscillation of the XON/XOFF algorithm is likely.

The duration of the pause quantum in the frame transmitted is determined by the value in the transmit PAUSE frame timer register. After an XOFF PAUSE frame is transmitted by the LU5M31, a blind timer (equal in length to the value in the PAUSE frame timer register minus 3 pause quantums) starts decrementing. If, after this timer has expired, the receive FIFO level remains above the low watermark (**rxwml**), then a second PAUSE frame will be transmitted by the LU5M31 with a pause-quantum equal to the value in the PAUSE frame timer register.

If the host has requested flow control by asserting, the FLWCTL signal high and the **flow** bit (bit 7 in the transmit frame configuration register) is cleared, the LU5M31 will generate and transmit a single PAUSE frame. The duration of the PAUSE will be equal in length to the value contained in the transmit pause frame timer register.

When FLWCTL is asserted high and the **flow** bit is set, the LU5M31 will generate and transmit a PAUSE frame after the current frame transmission (if any) has completed. The LU5M31 will then sample the FLWCTL signal after a blind time equal to the transmit pause frame timer register minus 3 pause quantums. If the FLWCTL signal remains asserted, a second PAUSE frame will be transmitted by the LU5M31. This process will be repeated for as long as the FLWCTL signal remains high. If the FLWCTL signal is deasserted prior to the blind time being reached, a second PAUSE frame will be transmitted with a zero value pause quantum indicating an XON to the paused sending station.

When a request is made to send a PAUSE frame, the GMAC transmits preamble and SFD, and builds a valid GMAC control frame with the 48-bit multicast destination address, specified in *IEEE* 802.3x, chapter 31B.1 (address of 0180C2000001). It also transmits the address loaded in the source address registers of the LU5M31, the type/length code (8808), the PAUSE opcode (opcode of 0x0001), the number of pause-quantum indicated by the transmit pause frame timer register, the pad data, and the correct CRC. The GMAC will then transmit this frame according to normal transmission rules. If the request to transmit a new frame is received simultaneous to a request to transmit a pause frame, the PAUSE frame is transmitted first.

Functional Description (continued)

The LU5M31 also responds to received pause frames. When the LU5M31 receives a pause MAC control frame, indicated by the multicast destination address of 0x0180C2000001, or a unicast address equal to the source address register with **uni** (bit 11 of the receive frame configuration register) is set, a type/length field = 0x8808 and an opcode of 0x0001, and the frame is 64 bytes with a good CRC, the LU5M31 will load its internal pause timer with the pause-quantum in the MAC control frame just received. The LU5M31 does not transmit any frames for a duration of time equal to pause-quantums * 512 bit times.

If a transmission is in progress when the PAUSE frame is received, the transmission is allowed to complete. The LU5M31 pause timer will commence at the end of the transmission. If a frame is not being transmitted when the PAUSE frame is received, then the pause timer will commence immediately. If a PAUSE frame is received while the transmitter is already in the PAUSE state, the new pause time indicated by pause-quantum is loaded into the pause timer, even if pause-quantum is zero.

MAC control frames received by the LU5M31 are not written to the receive FIFO if the receive start frame threshold (**rxsfth***)* is set to 24 bytes or greater. These frames are used by the MAC sublayer only and are not passed above the MAC to the logical link control (LLC) layer. MAC control frames received are however included in the receive event counters. Note the MAC control frame must be valid (good CRC, and equal to 64 bytes) before any action is taken by the LU5M31. The LU5M31 is capable of transmitting PAUSE frames while in the PAUSE state.

VLAN Support

The LU5M31 recognizes transmit and receive frames which are tagged with either one-level or two-level VLAN IDs. For both levels of tagging, the LU5M31 compares the 13th and 14th bytes of transmit and receive frames to the contents of both the one-level VLAN tag register and the two-level VLAN tag register.

If a nonzero match is made, the LU5M31 identifies the frame as either a one-level or two-level VLAN frame. If both a VLAN1 tagged frame and a VLAN2 tagged frame are identified (VLAN1 and VLAN2 registers have the same value), the frame will be identified as both a VLAN1 and a VLAN2 frame in the receive port diagnostic register, and the end of frame statistics, however the frame will be treated as a VLAN2 frame by the event counters.

Upon recognizing that a frame has a VLAN tag, counter thresholds are adjusted to account for the extra bytes that the VLAN tag adds to the frame. If the frame is a one-level VLAN frame, the upper threshold on receive and transmit counters is changed from maximum size octets to 1522 octets. If the frame is a two-level VLAN frame, the upper threshold on transmit and receive event counters is changed from maximum size octets to 1538 octets. In both cases, status bits are set in the receive port diagnostic register when a VLAN frame is received.

If **extbin** (bit 20 in the global configuration register) is set, the upper threshold on transmit and receive event counters is changed from maximum size octets to 1538 octets independent of the VLAN status of the frame. When the **extbin** bit is set, a frame recognized as having a VLAN tag will still be marked as a VLAN frame in receive port diagnostic register and the end of frame statistics.

Functional Description (continued)

CPU Interface

The LU5M31 provides a synchronous 16-bit CPU interface to support an extensive configuration and management function, including access to configuration and status registers and receive and transmit event counters. The CPU interface, which runs synchronously to a CPU clock (CPUCLK), allows configuration and status registers to be accessed for both read and write operations. It also provides real-time access of all event counters, independent of any ongoing CPU bus transactions.

The registers and counters vary in size from 16 bits up to 48 bits. The 16-bit wide registers or event counters can be read and written directly in one CPU transaction. The 32-bit and 48-bit registers and event counters must be written using two and three consecutive CPU transactions, respectively.

This protocol must be followed to successfully update the register or counter. If a 32-bit counter has a value of 0x01ABCDEF, data would be read in two transactions while the **burst** bit (bit 21 in the global configuration register) is not set. The high word is 0x01AB, and the low word is 0xCDEF as shown in Table 69.

Table 69. Big Endian Data Format

Bus Width	Address	Big Endian
16	0x0002	0x01AB
	0x0000	0xCDEF

Bus Protocol

Two signals CPUCSb and CPUSTRb are used to select the LU5M31 for a CPU transaction. The CPU chip select signal (CPUCSb) must be asserted low to qualify all other signals on the CPU interface. With CPUCSb asserted low, a strobe signal (CPUSTRb) is asserted low to initiate a transaction. CPUSTRb must remain asserted low until the CPU transaction has been completed.

Deasserting CPUSTRb high at any point during a CPU transaction will terminate the transaction. The CPURW signal is used to inform the LU5M31 of the nature of the transaction. When CPURW is high, a read transaction is requested. When low, a write transaction is requested.

Read Transaction

For a read, CPURW is asserted high, and CPUCSb and CPUSTRb are both asserted low, setup to the rising edge of CPUCLK. The next rising edge of CPUCLK will initiate the transaction. Data will be presented to the CPU on CPUDB[15:0] when CPUDV is asserted low and on every rising edge of CPUCLK after a predetermined latency.

The LU5M31 provides an autoincrement burst capability for counter read operations when the **burst** bit (bit 21 in the global configuration register) is set. The allows the CPU to burst read data from up to six 16-bit locations. A single read transaction is supported for registers. It is not possible to burst read registers.

The CPU must provide a valid address on CPUAD[7:1] where the address presented is at least six locations from the top of the address space. This prevents bursting into invalid memory locations. A burst into invalid memory space will result in a maskable interrupt being generated (**macv**, bit 2 of the interrupt register) and data will be returned for all valid memory locations.

When an increment of the address pointer would put it into invalid address space, data from the last valid location is presented on the CPU data bus (CPUDB[15:0]) for the remaining number of burst cycles.

The CPU can terminate a burst read at any time by deasserting CPUSTRb high. This will terminate the read and no more data will be presented. CPUDVALb will be deasserted high after CPUSTRb is deasserted high. An attempted read of a reserved memory location will result in a **macv** interrupt being generated.

Write Transaction

If a write transaction is presented, then the data to be written to the register or counter is provided on CPUDB[15:0] in a big endian mode, together with a valid address. Note that burst capability is not supported for a write transaction. CPURW would be asserted low, CPUCSb and CPUSTRb would also be asserted low. These signals, which are set up to a rising edge of the CPU clock (CPUCLK) would initiate a CPU transaction on that rising edge of CPUCLK. When the write transaction is complete, the LU5M31 will assert CPUDVALb low on the rising edge of CPUCLK for one CPUCLK clock cycle.

If a write to a 32-bit or 48-bit location is taking place, all 16-bit words must be presented consecutively to the LU5M31 to successfully complete the write transaction. The order in which these words are written to the LU5M31 is not important, however all words must be written consecutively.

Functional Description (continued)

Gigabit Management Interface

The LU5M31 GMII management interface is used to control PHYs and obtain their status. The interface is a good two-wire serial interface as described in *IEEE* 802.3, Section 22. The LU5M31 initiates the transfer of the configuration and status information through the GMII management interface to the PHYs by addressing them with their 5-bit address and the 5-bit register address of the target configuration or status register. The frame format follows the format described in *IEEE* 802.3 Section 22.24 The PHY address and register address are sent MSB first.

MDIO, management data input/output, is a bidirectional pin that carries data and is clocked with MDC, the management data clock. When IDLE, the MDIO pin is in a high-impedence state. The LU5M31 will always initiate a transfer. The LU5M31 always sends a preamble of 32 contiguous 1s before it will transfer data with the PHY. After 32 1s are sent, the read or write opcode will be sent depending on the MII_Write bit of the MII control register followed by the address and register fields. During a read, the LU5M31 will turn around the MDIO by setting MDIO in the high-impedance state after the REGAD has been sent. The LU5M31 expects the PHY to also be in the high impedence state for one bit, followed by the PHY driving low for the next bit, followed by the register data, and then followed by the high-impedence IDLE state. During a write operation, the LU5M31 will drive the MDIO during the turnaround time with a 1 then a 0 followed by the data field.

Table 70. MII Management Interface Frame Format

Read/ Write	MDIO Frame Fields							
	Pre	**st**	**op**	**PHYAD (4 . . . 0)**	**REGAD (4 . . . 0)**	**TA**	**DATA (15 . . . 0)**	**IDLE**
Read	1 . . . 1	01	10	AAAAA	RRRRR	Z0	dddddddddddddddd	Z
Write	1 . . . 1	01	01	AAAAA	RRRRR	10	dddddddddddddddd	Z

Note: Bits are transmitted left to right.

Reset Operation

Hardware Reset

The LU5M31 has a single hardware reset pin, RESETb, which when asserted low, will reset the entire LU5M31 including the phase-locked loop circuitry. When RESETb is asserted low, registers will return to default values, state machines will return to IDLE, counters are reset to zero, and the FIFOs are completely flushed.

Any frames being transmitted or received at the time of a hardware reset are lost. RESETb does require that HCLK be free running. However, any time the HCLK frequency changes (including during a hardware reset), the internal PLL needs 500 µs to stabilize it.

Software Reset

The LU5M31 offers software reset bits in the global configuration register. There are four different reset bits: a global reset, counter reset, transmit reset, and receive reset. The receive and transmit reset bits reset the receive and transmit paths, respectively. These resets do not have any effect on the counter values, although during the transmit or receive port reset, associated counters are not updated. The counter reset bit **cntres** (bit 0 in the global configuration register) will reset the event counters to zero.

Register values are not affected by **cntres**. The global software reset **gsrst** (bit 3 in the global configuration register) resets the transmit and receive data paths as well event counters, management logic and all registers except the global configuration register. A software reset will not affect the PLL. All software resets in the LU5M31 are self-clearing after three CPUCLK cycles.

Circuit Board Manufacturing Testability

The LU5M31 provides a standard *IEEE* P1149.1 test access port (TAP) for board-level test capability (boundary scan). The boundary-scan (B-S) cells for the pins of a chip are interconnected to form a shift register around the border of a device. There exists one B-S cell for each I/O signal. This path is provided with serial input and output connections and appropriate clock and control signals. The serial path can be used for many purposes as described below.

- EXTEST mode allows the board interconnections between the various chips to be tested. Test data can be shifted into the B-S output cell associated with chip output pins and then loaded in parallel through the chip interconnections into those B-S input cells associated with input pins.
- SAMPLE mode captures the inputs and outputs of the chip into the B-S register while it's in normal mode, and then shifts the data out through TDO. This test enables B-S to take a snapshot of the data flow through a chip without interfering with its normal operation. This option is useful for design debugging and fault diagnosis.
- BYPASS mode provides a means to shorten the movement of data from TDI to TDO. This mode is selected when no other on-chip data register is needed during board-level testing and is used to speed test access to other devices.
- Four pins plus a reset are used to control and communicate with the B-S logic. A reset must be provided at power up to ensure the B-S logic is in the normal mode.
- TCK—Test Clock (maximum 10 MHz).
- TMS—Test Mode Select. Controls B-S state machine, pulled high on-chip.
- TDI—Test Data Input. Serial data input, pulled high on-chip.
- TDO—Test Data Output. Serial data output, 3-state.
- TRSTb—Test Reset (active-low).

Figure 7 demonstrates the use of B-S on a board for manufacturing tests. If this test capability is desired, please contact your Lucent Technologies Sales Representative and a BSDL (boundary-scan description language) file will be provided.

Circuit Board Manufacturing Testability (continued)

A schematic overview of boundary scan (B-S) on a board is shown in Figure 7.

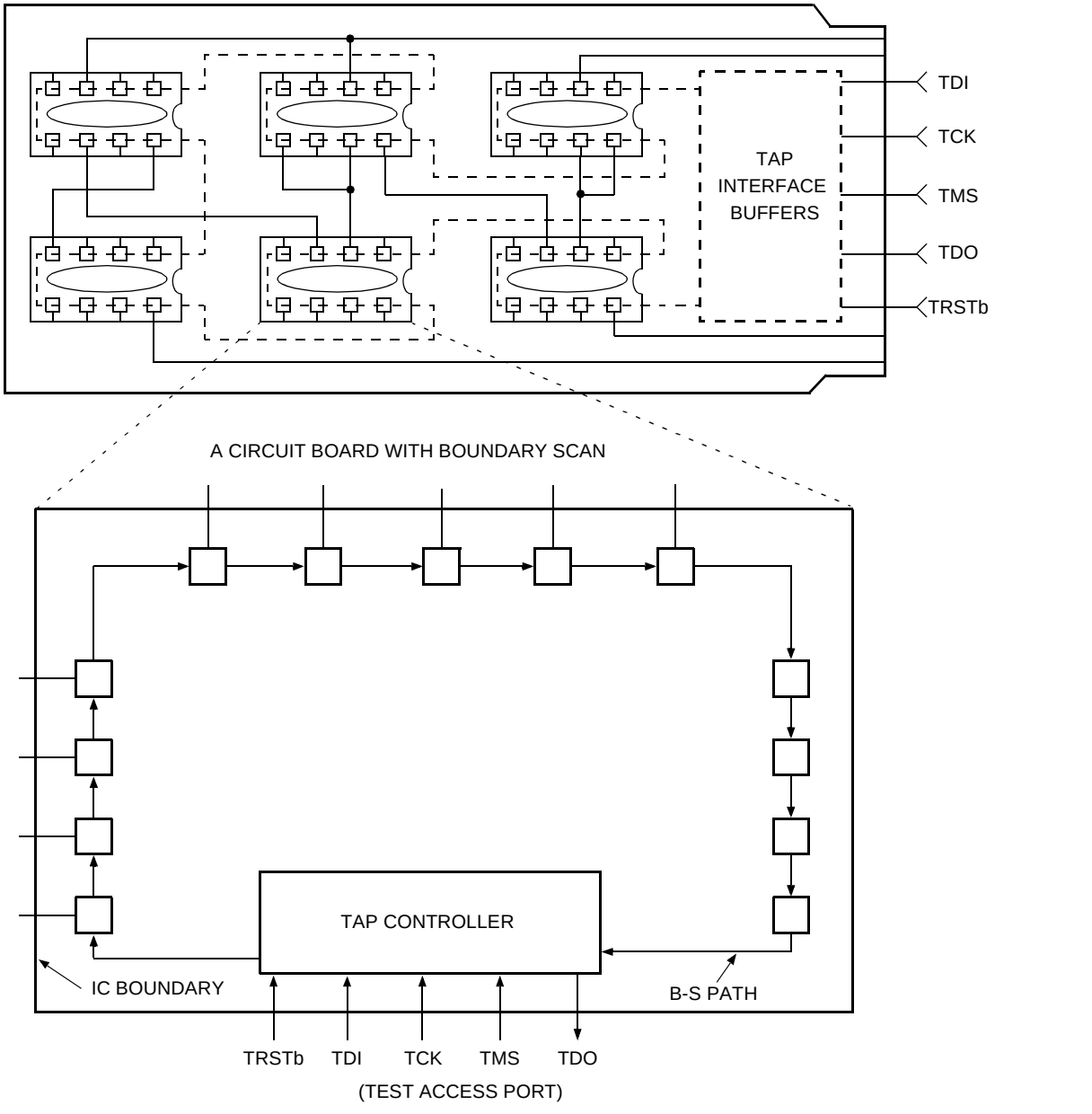


Figure 7. Boundary-Scan Interaction on a Chip

Power Supply Considerations

The LU5M31 operates at 3.3 V for low power consumption and can interface to TTL, LVTTL, or LVCMOS signal levels with no reliability problems.

This operation is made possible by the use of special, patented protection circuits. For pins defined as 5 V tolerant, these circuits operate in such a way that the voltage across the gate oxide never exceeds 3.6 V, even when the pad voltage is as high as 5.5 V.

These buffers use two different protection circuits. The one used to protect output and bidirectional buffers dissipates a small (35 μW max, 16 μW nominal) amount of dc power. The circuit used to protect input buffers does not consume any dc power, but does allow a small but finite input current to flow into the input stage during switching. At V_{IN} = 2 V, this can be as high as 60 μA, but at $V_{IN} \leq 1.4$ V or $V_{IN} \geq 2.7$ V, this current is zero. For comparison, the displacement current during switching, C x (dV/dT) is several mA for any fast input signal.

In order to interface with 5 V signals, these buffers require a 5 V power supply in addition to the normal (3.3 V) power supply. It is required that this 5 V supply should be equal to (within ± 0.3 V) or greater than the maximum input signal that will be applied to the buffers.

Even though the 5 V power supply is normally at a higher voltage than the 3.3 V power supply, during powerup and powerdown sequencing, the 3.3 V power supply is allowed to be at a higher voltage than the 5 V supply for as long as 100 ms. Longer times under this condition may result in damage to the circuit due to excessive heating.

If the 3.3 V power supply fails during normal operation, no current will be drawn from any signal applied to the circuit as long as the 5 V supply is still greater than or equal to this signal voltage.

The output buffer will drive TTL, LVTTL, or LVCMOS levels on 5 V chips, but should not be used to drive CMOS levels on 5 V chips, since its output only reaches V_{DD} (3.3 V nominal).

The buffers can be used without the 5 V supply on straight 3.3 V systems. In this case, the 5 V power supply pin should be connected to V_{DD} (3.3 V).

Absolute Maximum Ratings

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

Table 71. Absolute Maximum Ratings

Parameter	Symbol	Supply	Min	Max	Unit
Maximum Supply Voltage with Respect to V_{SS}	V_{DD}	3.3 V	—	4.6	V
	V_{DD5}	5.0 V	—	7.0	V
Maximum Pin Voltage on Any Pin	V_{IN}	3.3 V	V_{SS} – 0.3	V_{DD} + 0.3	V
	V_{IN}	5.0 V	V_{SS} – 0.3	V_{DD5} + 0.3	V
Ambient Operating Temperature with 200 FPM Airflow	T_A	—	0	70	°C
Ambient Storage Temperature	T_{stg}	—	–40	125	°C

Handling Precautions

Although protection circuitry has been designed into this device, proper design precautions should be taken to avoid exposure to electrostatic discharge (ESD) during handling and mounting. Lucent Technologies employs a human-body model (HBM) and charged-device model (CDM) for ESD-susceptibility testing and protection design evaluation. ESD voltage thresholds are dependent on the circuit parameters used in the defined model. No industry-wide standard has been adopted for the CDM. However, a standard HBM (resistance = 1500 Ω, capacitance = 100 pF) is widely used and, therefore, can be used for comparison. The HBM ESD threshold presented here was obtained by using these circuit parameters:

Device	Voltage
LU5M31	1000 V

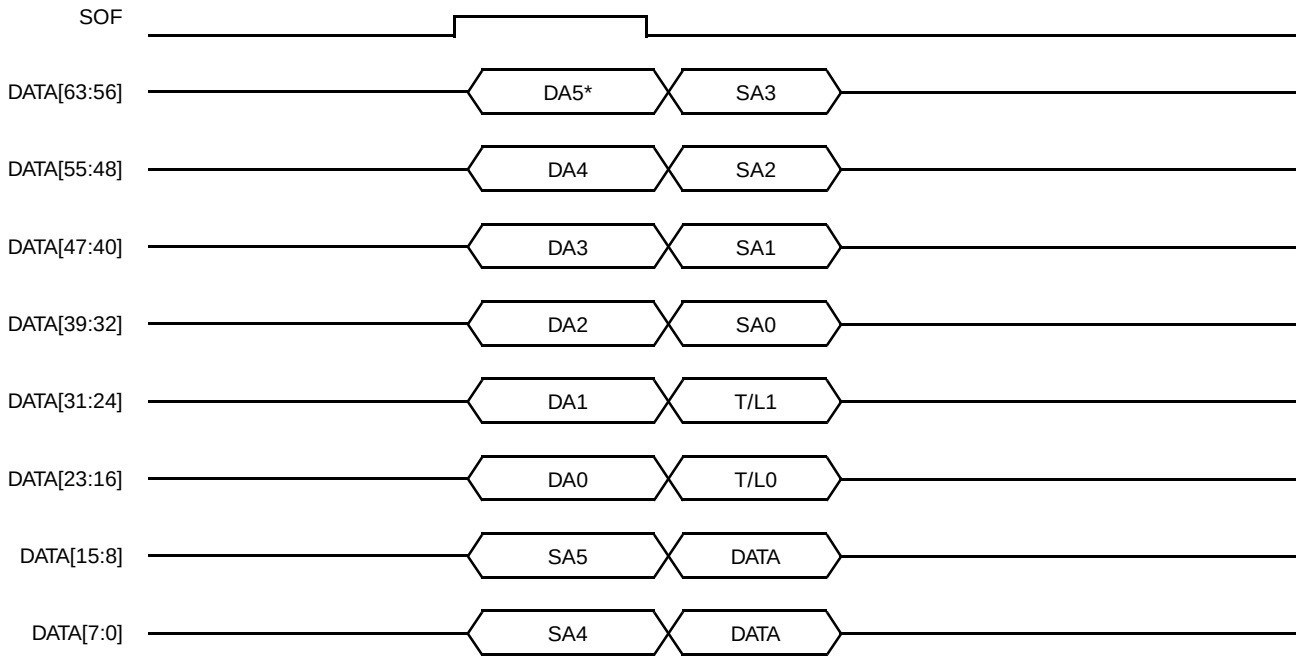
Electrical Characteristics

Ambient temperature 0 °C to 70 °C, V = 3.3 V ± 5%, V_{SS} = 0.0 V.

Parameter	Symbol	Test Conditions	Min	Max	Unit
Input Voltage:					
Low	V_{IL}	—	–0.3	0.8	V
High	V_{IH}	—	2.0	V_{DD} + 0.3	V
Output Voltage:					
Low	V_{OL}	—	—	0.5	V
High	V_{OH}	—	2.4	—	V
Input Leakage Current	I_{ILH}, I_{ILL}	V = 5.5 V	—	2	µA
Power Supply Current	I_{DDS}	0 °C, V_{DD} = 3.3 V	—	1.0	A
Power Dissipation	P_D	0 °C, V_{DD} = 3.3 V	—	1.6	W
Pin Capacitance:					
CMOS Input	—	—	—	5	pF
CMOS Output	—	—	—	5	pF
CMOS 3-state	—	—	—	5	pF

Timing Characteristics (Preliminary)

Figure 8 and Figure 9 show the relationship between the contents of the Ethernet frame to data on the host bus. However, data bytes are transmitted and received most significant byte (MSB) first. The hexadecimal value of 1234 would be transmitted or received as 2-1-4-3.



* DA5 is the most significant byte of the destination address. Bit 56 is the individual group address bit and is the first bit received or transmitted on the Ethernet.

Figure 8. Ethernet Frame Data First Byte Received or Transmitted

Timing Characteristics (Preliminary) (continued)

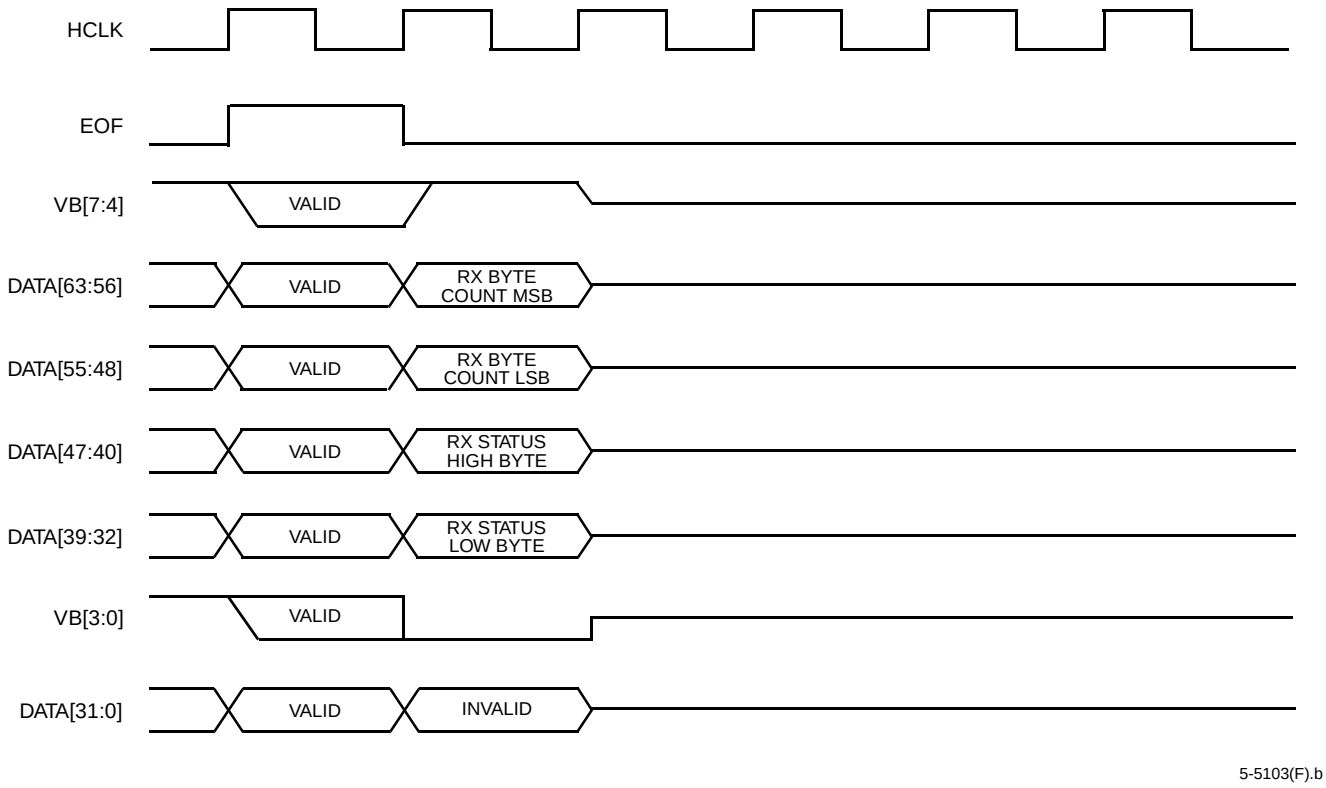


Figure 9. Byte Order on Host Interface Data Bus During Receive Statistics

Timing Characteristics (Preliminary) (continued)

PCS Receive Interface

Table 72. Receive Bus ac Specifications

Name	Parameter	Min	Max	Unit
t1	RX_CLK[1:0] Frequency.	62.5	—	MHz
t2	RX_CLK[1:0] Skew.	—	1.0	ns
t3	RDATA[9:0] Setup Before RX_CLK1.	2.5	—	ns
t4	RDATA[9:0] Setup Before RX_CLK0.	2.5	—	ns
t5	RDATA[9:0] Hold After RX_CLK1.	1.5	—	ns
t6	RDATA[9:0] Hold After RX_CLK0.	1.5	—	ns
t_{DUTY}	RX_CLK[1:0] Duty Cycle.	40	60	%

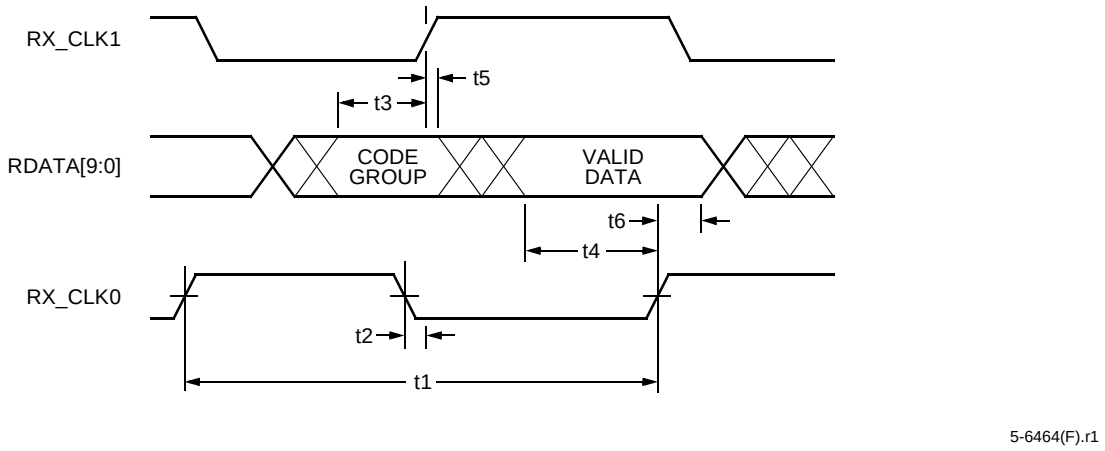


Figure 10. PCS Receive Interface Timing

Timing Characteristics (Preliminary) (continued)

PCS Transmit Interface

Table 73. Transmit Bus ac Specifications

Name	Parameter	Min	Max	Unit
t1	TX_CLK Period (nominal).	8	—	ns
t2	TDATA[9:0] Setup to TX_CLK.	2.0	—	ns
t3	TDATA[9:0] Hold from TX_CLK.	1.0	—	ns
t_{DUTY}	TX_CLK Duty Cycle.	40	60	%

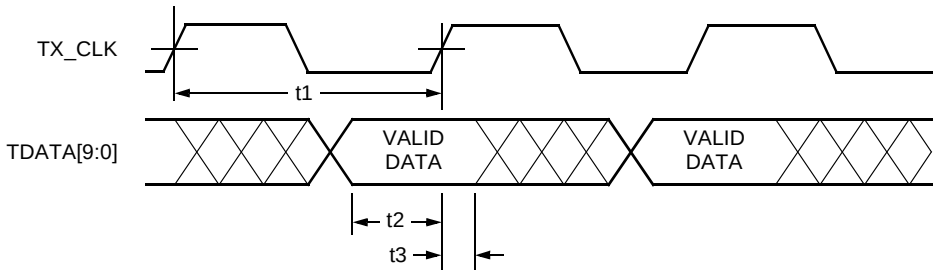


Figure 11. PCS Transmit Interface Timing

Timing Characteristics (Preliminary) (continued)

The following ac timing specifications require that HCLK be in the range of 33 MHz—66 MHz and that SpeedSel is driven high. When SpeedSel is driven low, these timing specifications are not valid.

Table 74. DMA Receive Timing

Name	Parameter	Min	Max	Unit
t1	RXREQb Delay from HCLK Rising Edge.	—	7	ns
t2	PFCSb, RTSEL, CHSEL[2:0], RXUNLDb Setup to HCLK Rising Edge.	5	—	ns
t3	SOF, EOF, DATA[63:0], VB[7:0], IMATCH Delay 3-State to Active from HCLK Rising Edge.	—	7	ns
t4	SOF, EOF, DATA[63:0], VB[7:0], IMATCH Delay Active to Active States from HCLK Rising Edge.	—	7	ns
t5	PFCSb, RTSEL, CHSEL[2:0], RXUNLDb Hold from HCLK Rising Edge.	0	—	ns
t6	SOF, EOF, DATA[63:0], VB[7:0], IMATCH Delay Active to 3-state from HCLK Rising Edge.	—	7	ns
t7	Clock Frequency.	33	66	MHz

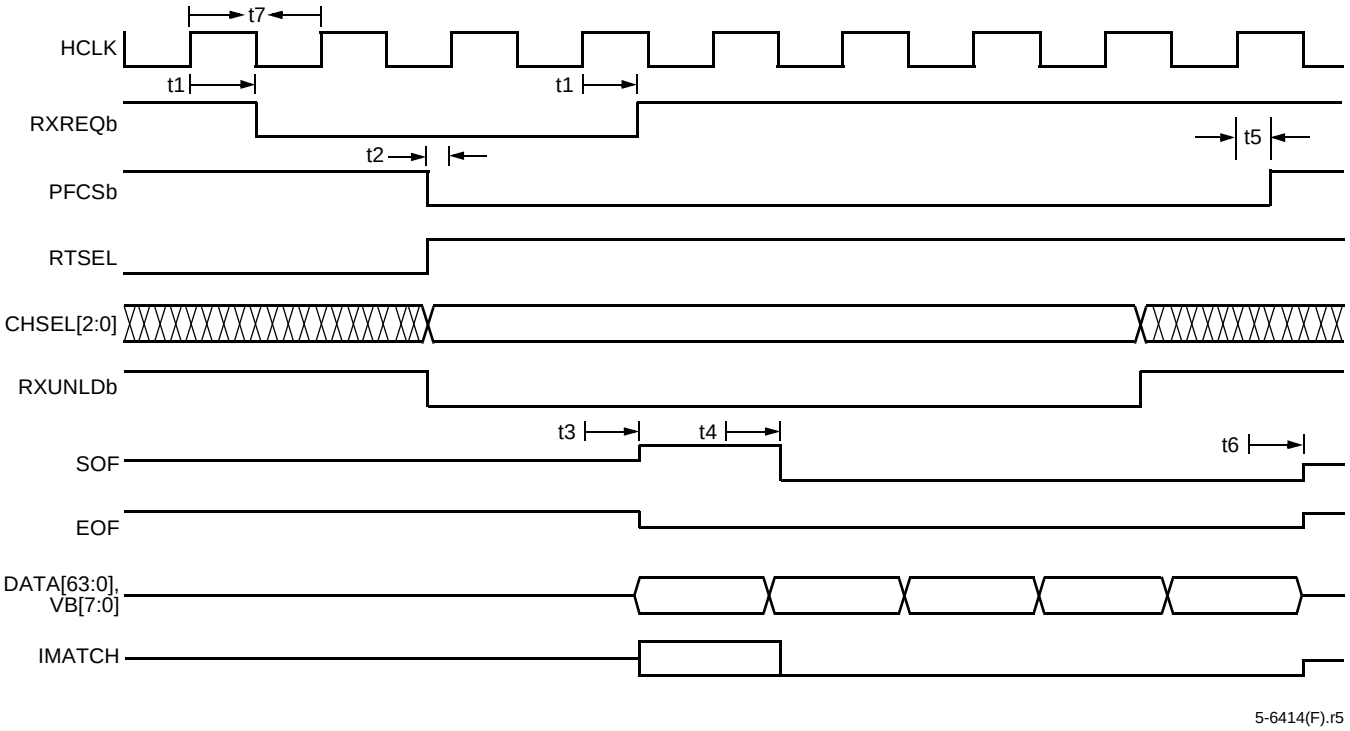
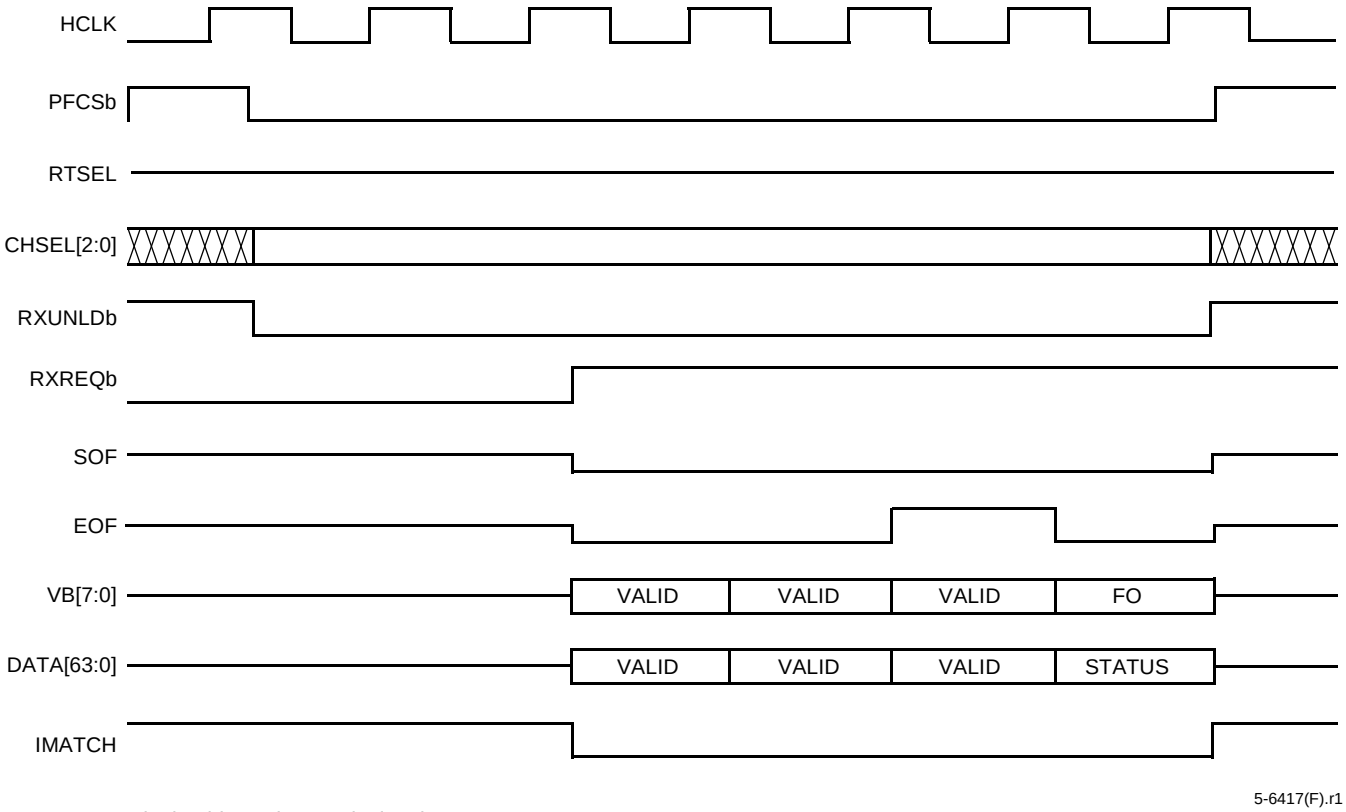


Figure 12. DMA Receive Timing

Timing Characteristics (Preliminary) (continued)

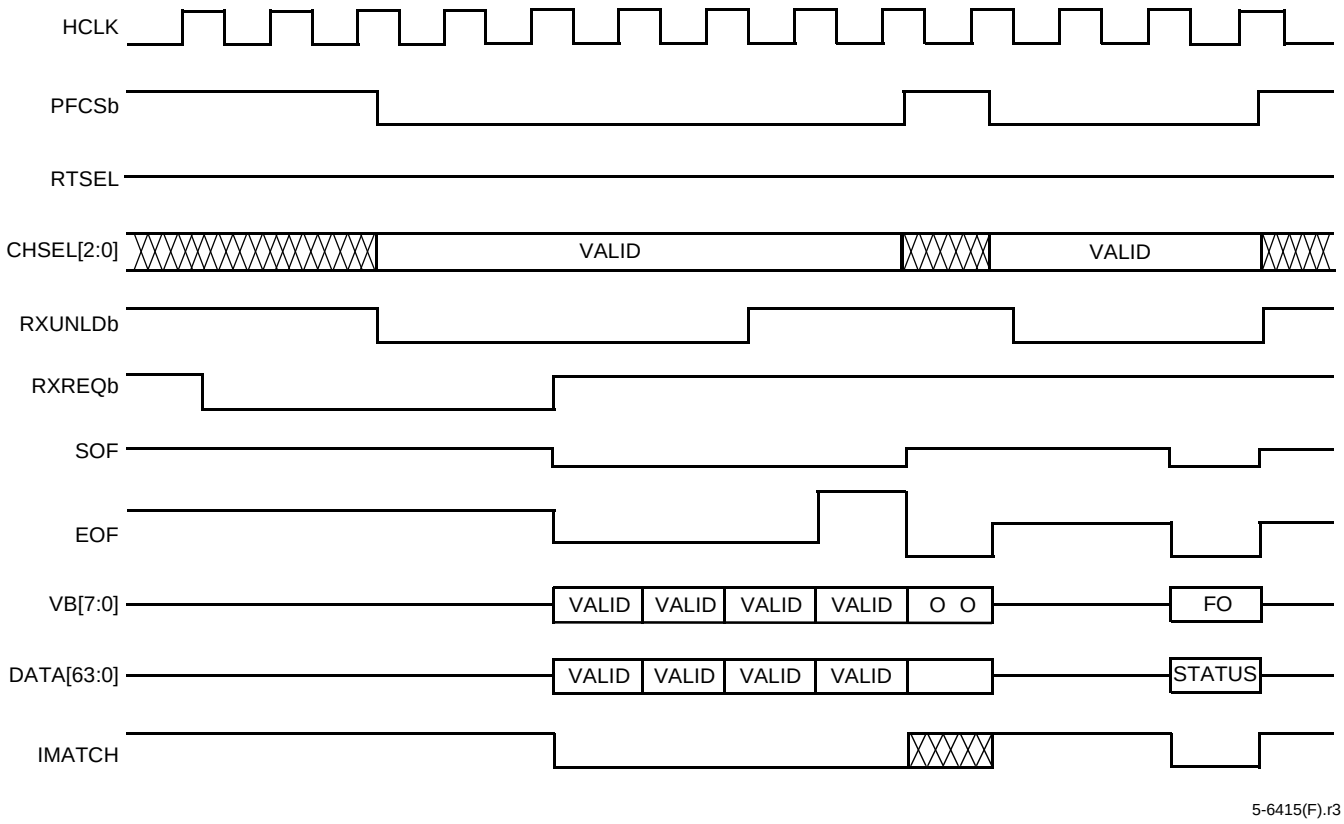
Host Interface



Note: RXUNLDb should match PFCSb signal.

Figure 13. DMA Receive with EOF in Middle of Burst

Timing Characteristics (Preliminary) (continued)



Note: RXUNLDb should match PFCSb signal.

Figure 14. DMA Receive with EOF on Last Word of Burst

Timing Characteristics (Preliminary) (continued)

Table 75. DMA Transmit Timing

Name	Parameter	Min	Max	Unit
t1	TXABLEb Delay from Rising Edge of HCLK.	—	7	ns
t2	PFCSb, RTSEL, CHSEL[2:0], TXRDYb, SOF, EOF, DATA[63:0], VB[7:0], TXCRC Setup to Rising Edge of HCLK.	5	—	ns
t3	PFCSb, RTSEL, CHSEL[2:0], TXRDYb, SOF, EOF, DATA[63:0], VB[7:0], TXCRC Hold from Rising Edge of HCLK.	0	—	ns

Note: Once a transmit burst operation has started, TXABLEb does not deassert for 2 clock cycles.

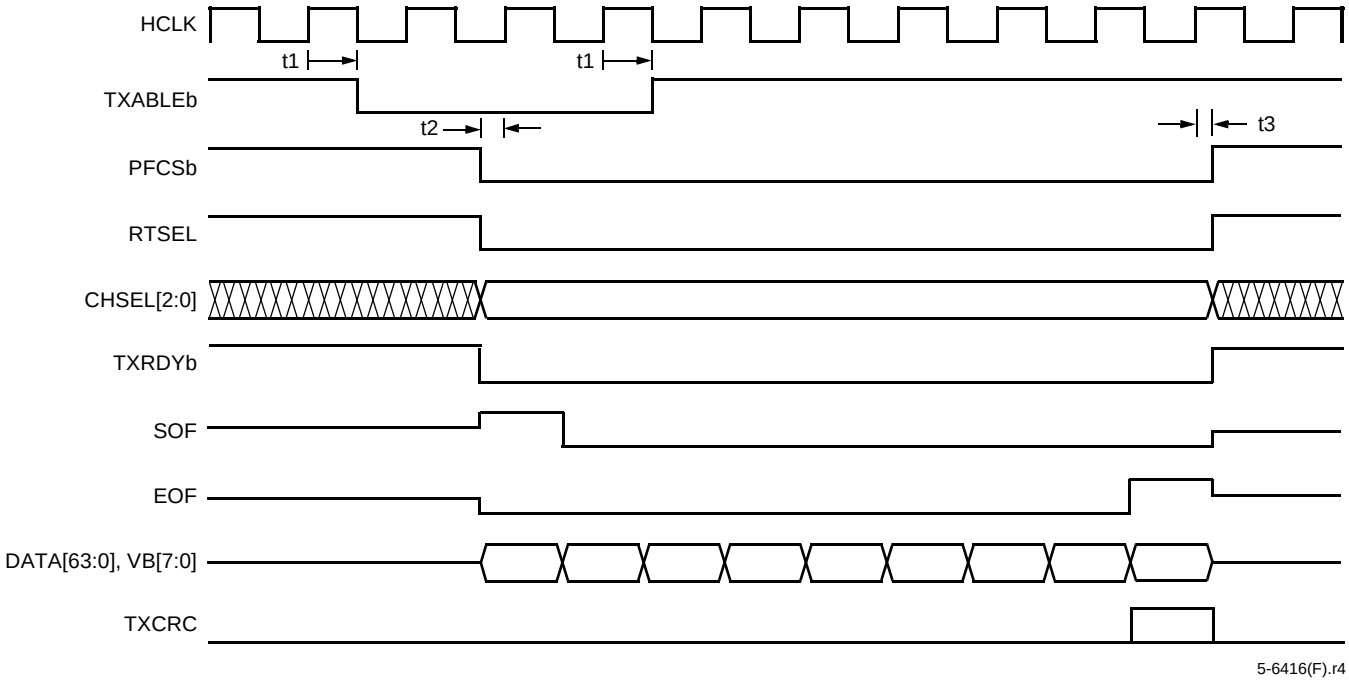


Figure 15. DMA Transmit Timing

Timing Characteristics (Preliminary) (continued)

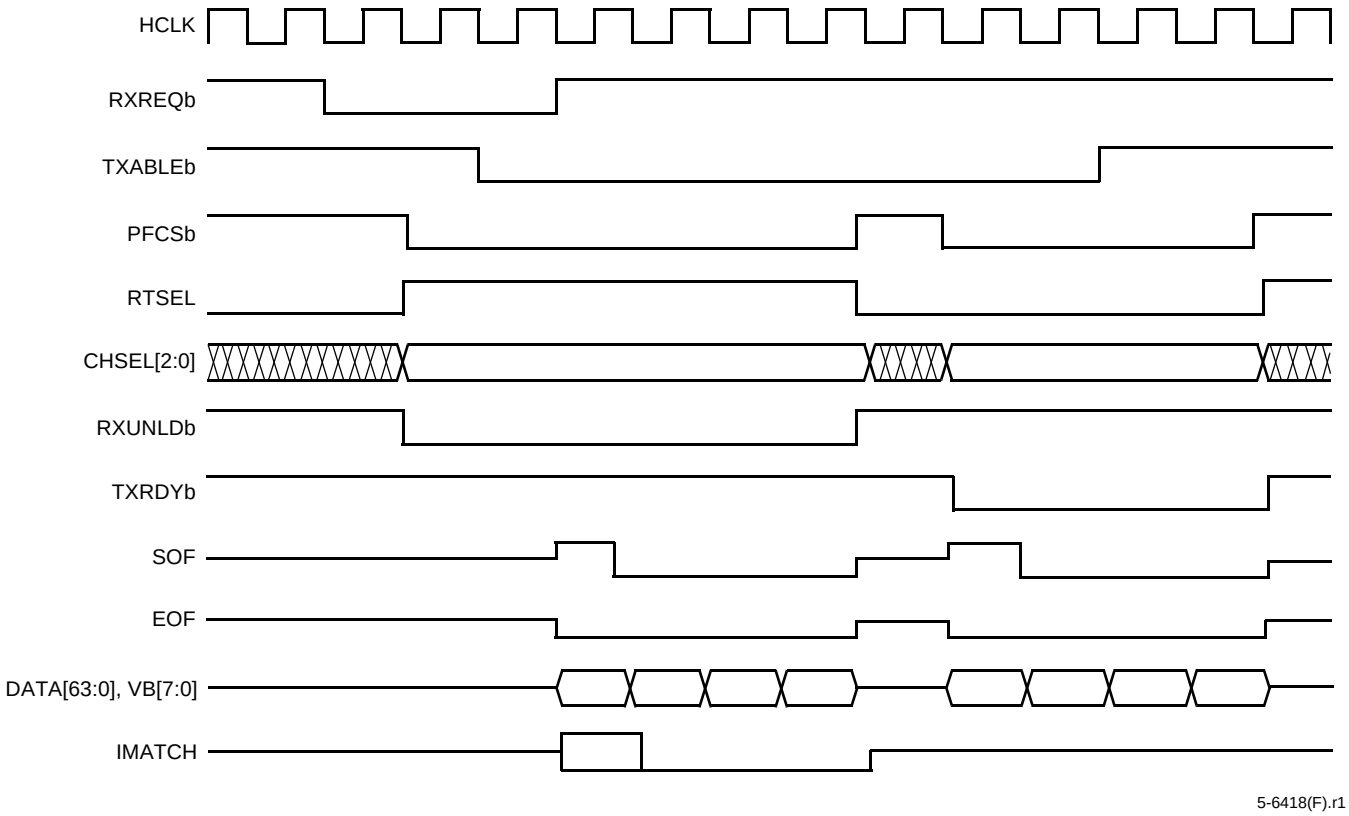


Figure 16. DMA Receive to Transmit 1 Dead Cycle for Bus Turnaround

Timing Characteristics (Preliminary) (continued)

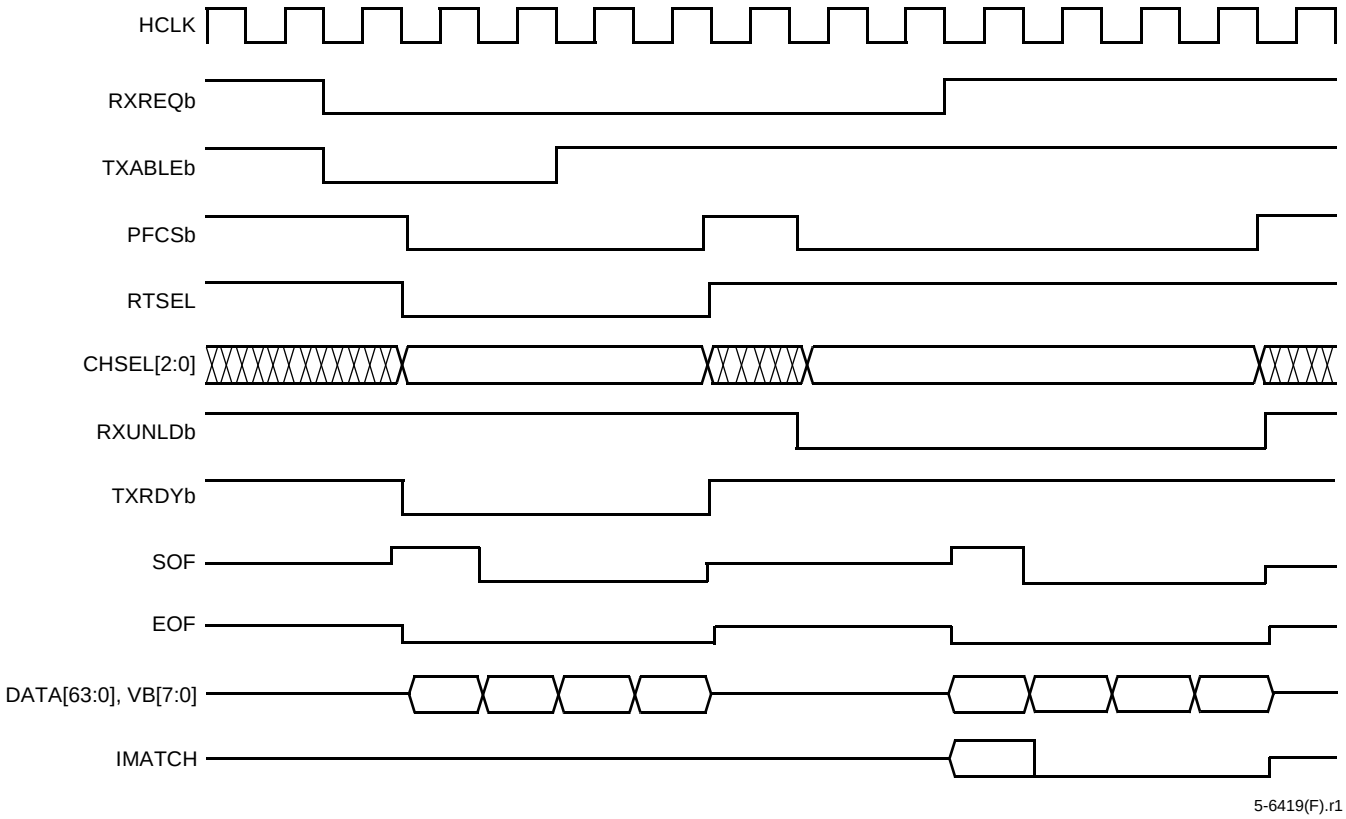


Figure 17. DMA Transmit to Receive 1 Dead Cycle for Bus Turnaround

Timing Characteristics (Preliminary) (continued)

Table 76. Hardware Reset Timing Parameters

Name	Parameter	Min	Typ	Max	Unit
t1	Reset Assertion Time for RESETb.	5	—	—	µs
t2	From RESETb Deasserted to Chip Activity.	250	—	—	µs

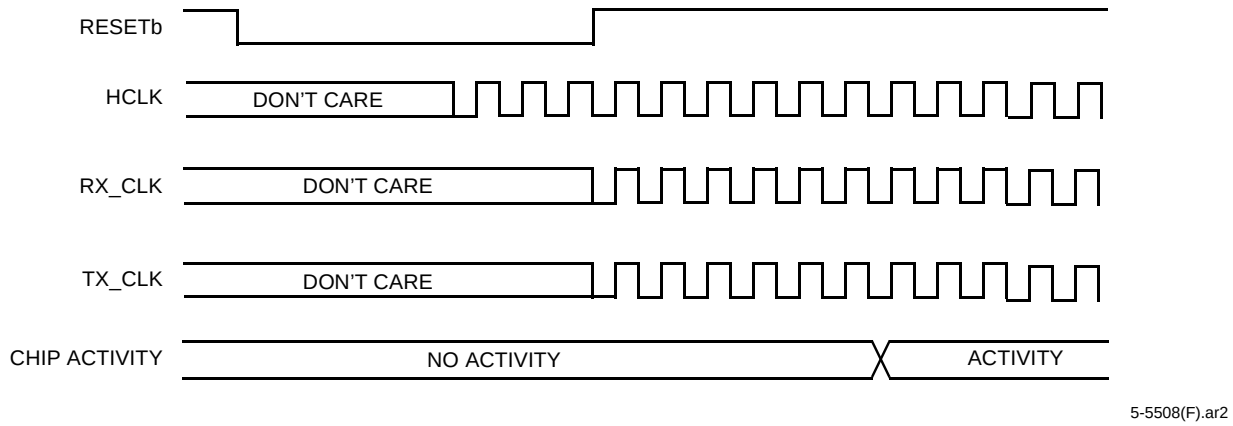


Figure 18. DMA Hardware Reset Timing

Timing Characteristics (Preliminary) (continued)

CPU Timing

Table 77. CPU Timing Parameters

Name	Parameter	Min	Max	Unit
t1	CPUCSb Setup to CPUCLK Rising Edge.	7	—	ns
t2	CPUCSb Hold from CPUCLK Rising Edge.	0	—	ns
t3	CPUSTRb Setup to CPUCLK Rising Edge.	7	—	ns
t4	CPUSTRb Hold from CPUCLK Rising Edge.	0	—	ns
t5	CPURW Setup to CPUCLK Rising Edge.	7	—	ns
t6	CPURW Hold from CPUCLK Rising Edge.	0	—	ns
t7	CPUAD Setup to CPUCLK Rising Edge.	7	—	ns
t8	CPUAD Hold from CPUCLK Rising Edge.	0	—	ns
t9	CPUDB Setup to CPUCLK Rising Edge.	7	—	ns
t10	CPUDB Hold from CPUCLK Rising Edge.	0	—	ns
t11	CPUDB Delay from CPUCLK Rising Edge.	—	15	ns
t12	CPUDB 3-State from CPUCLK Rising Edge.	—	15	ns
t13	CPUDVALb Delay from CPUCLK Rising Edge.	—	15	ns
t14	CPUDVALb 3-state from CPUCLK Rising Edge.	—	15	ns
t15	CPUINTb Delay from CPUCLK Rising Edge.	—	15	ns

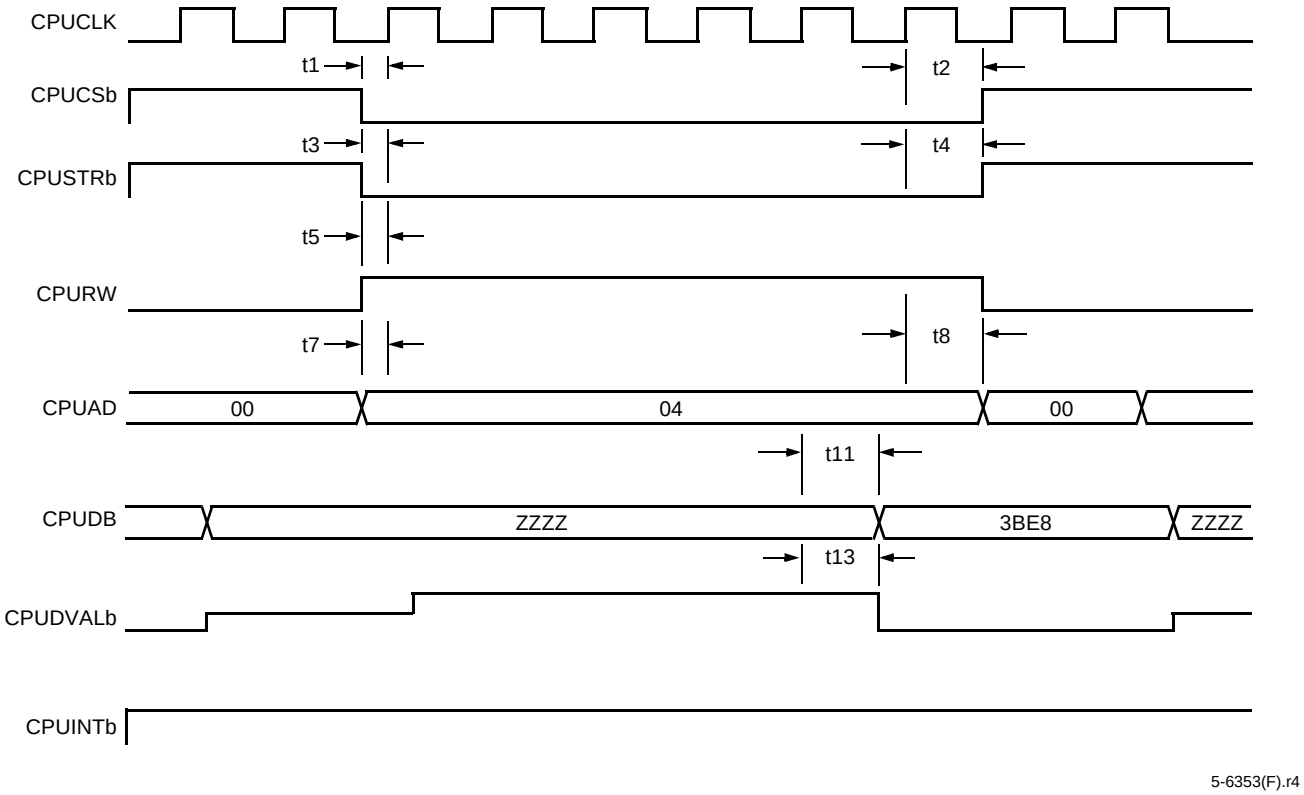


Figure 19. Single Read

Timing Characteristics (Preliminary) (continued)

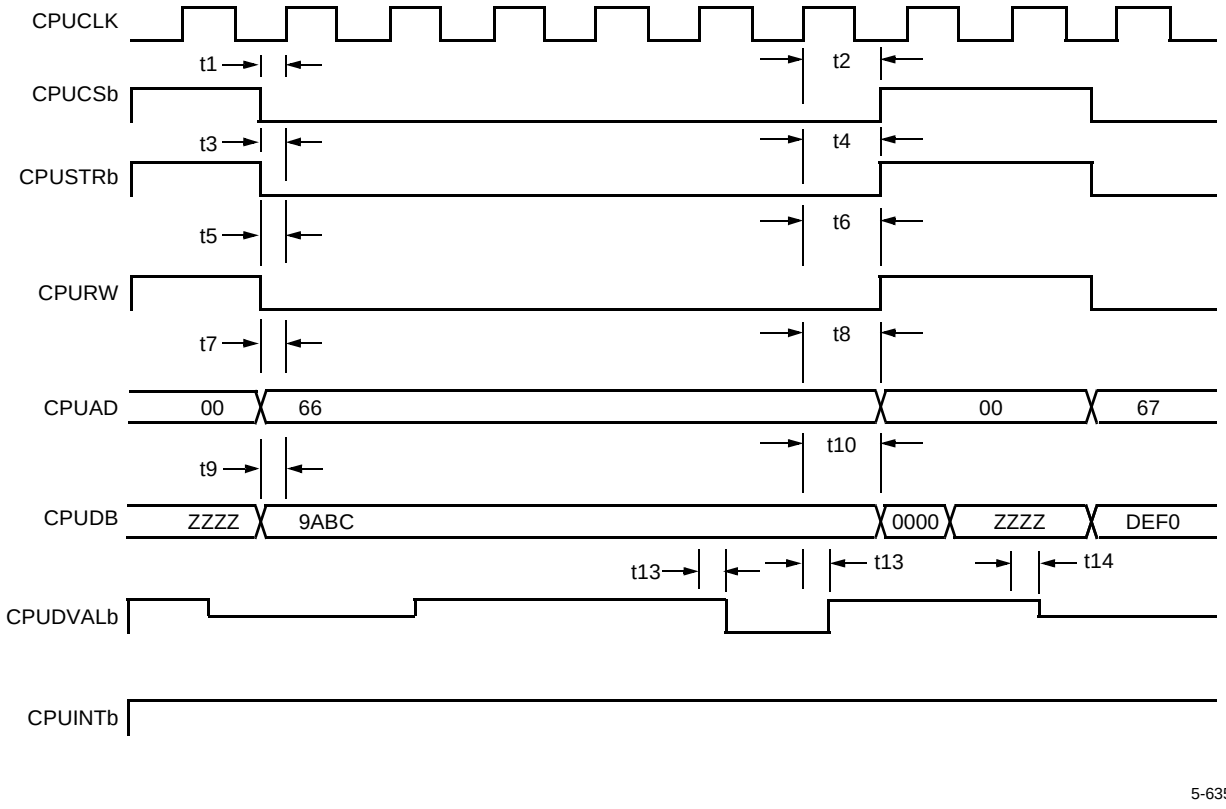


Figure 20. 16-Bit Register/Counter Write

Timing Characteristics (Preliminary) (continued)

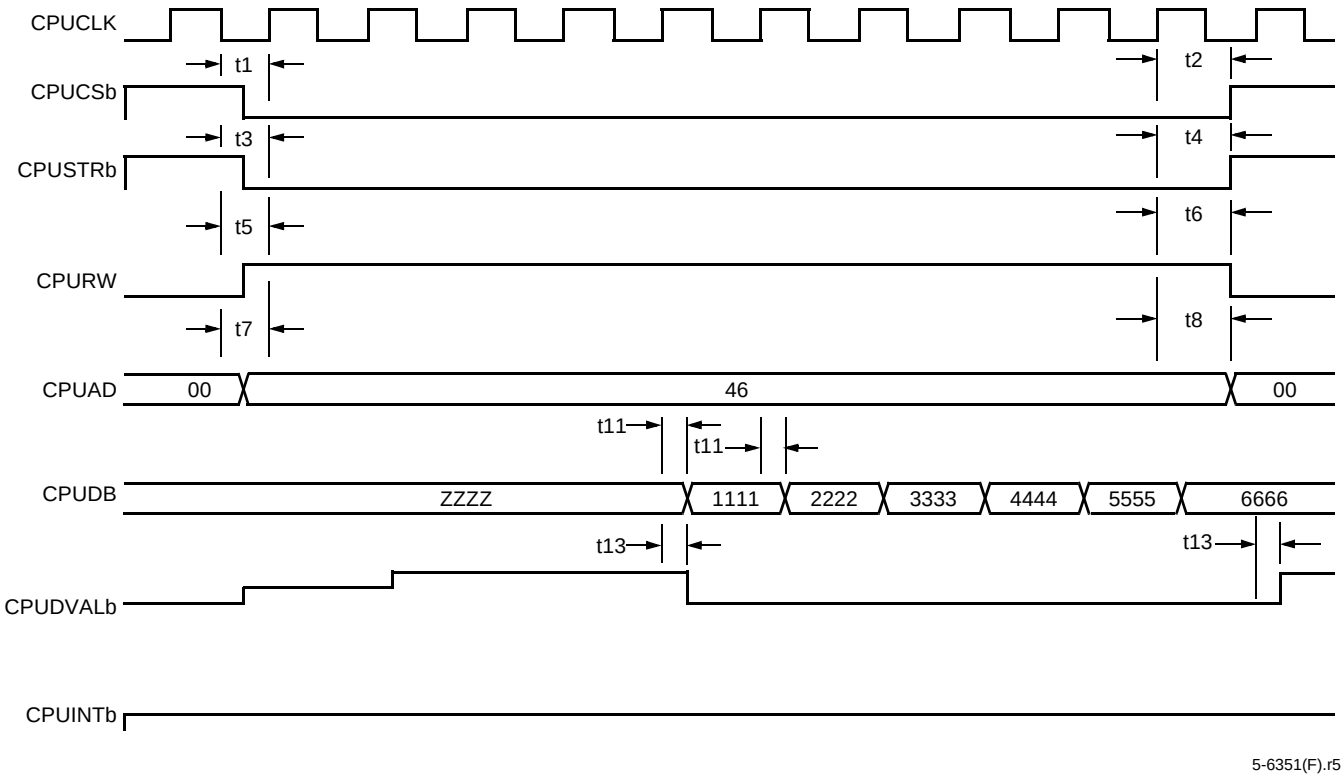


Figure 21. Burst Read

Timing Characteristics (Preliminary) (continued)

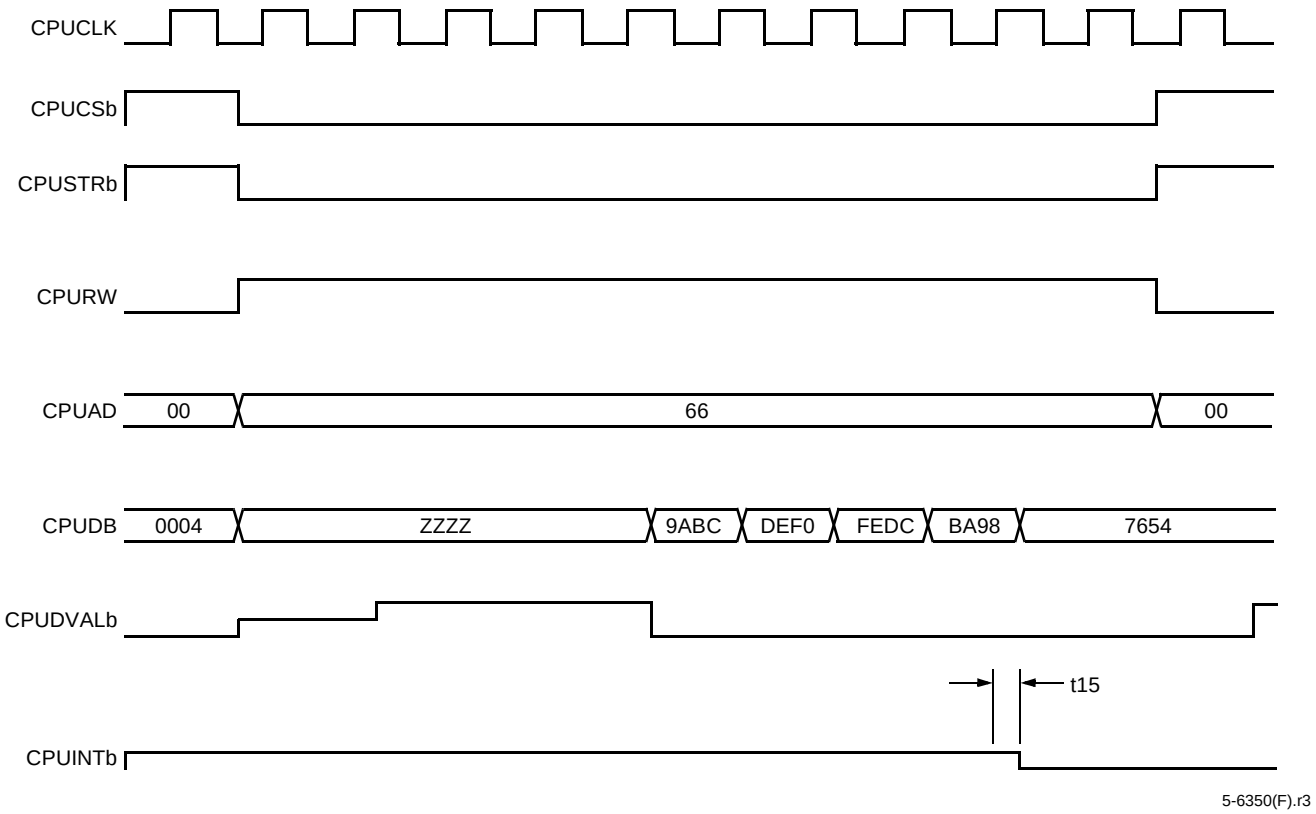
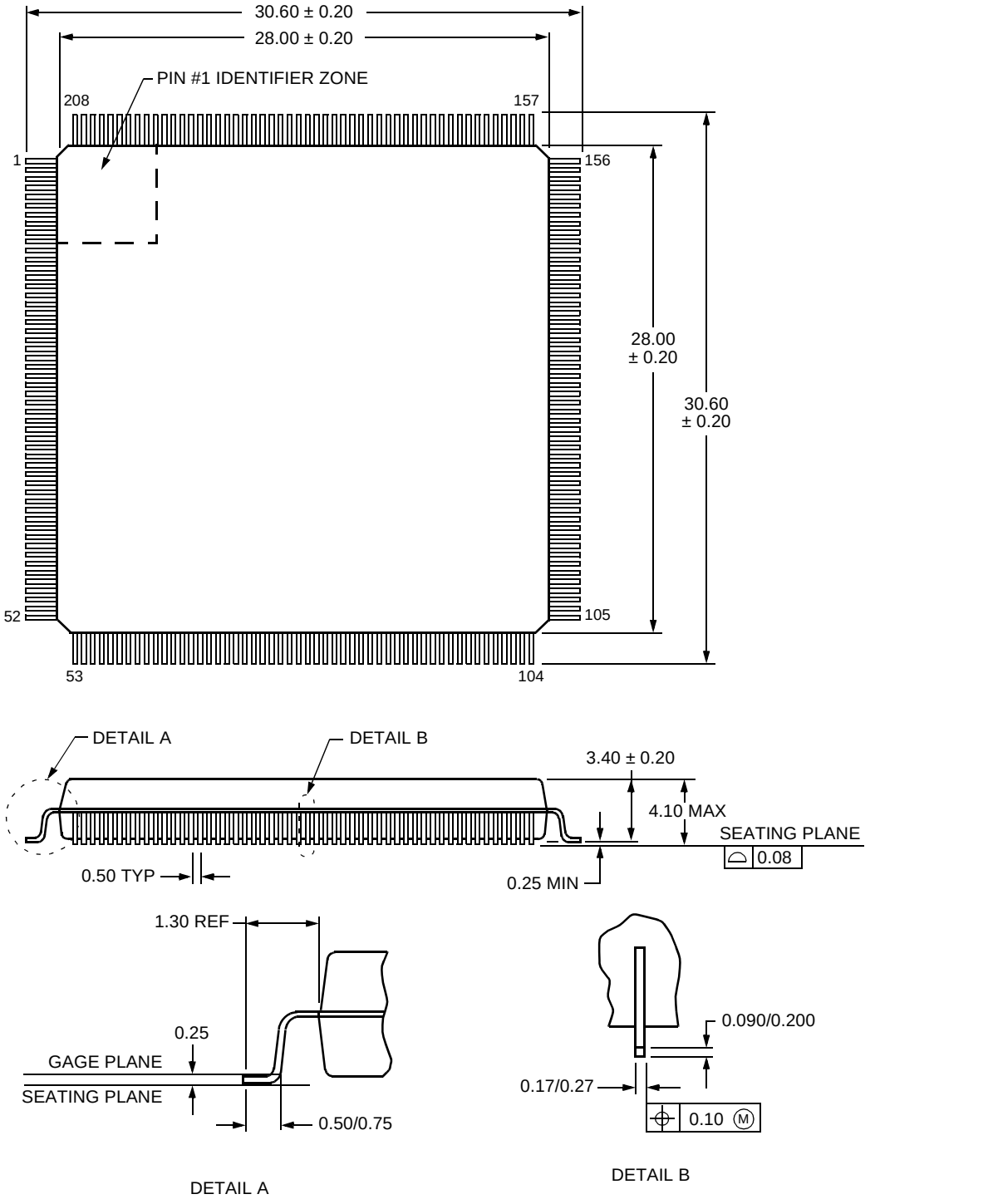


Figure 22. Burst Read into Invalid Address Space

Outline Diagram

208-Pin SQFP Package Outline

Dimensions are in millimeters.



5-5296(F).r1

FASTCAT ™ Ethernet IC Naming Convention

A new naming scheme is being implemented for all *FASTCAT* Ethernet integrated circuits (ICs). Figure 23 illustrates the codes and symbols that will be used to identify the basic characteristics of these IC devices. Note that the codes which apply to the example shown below are highlighted.

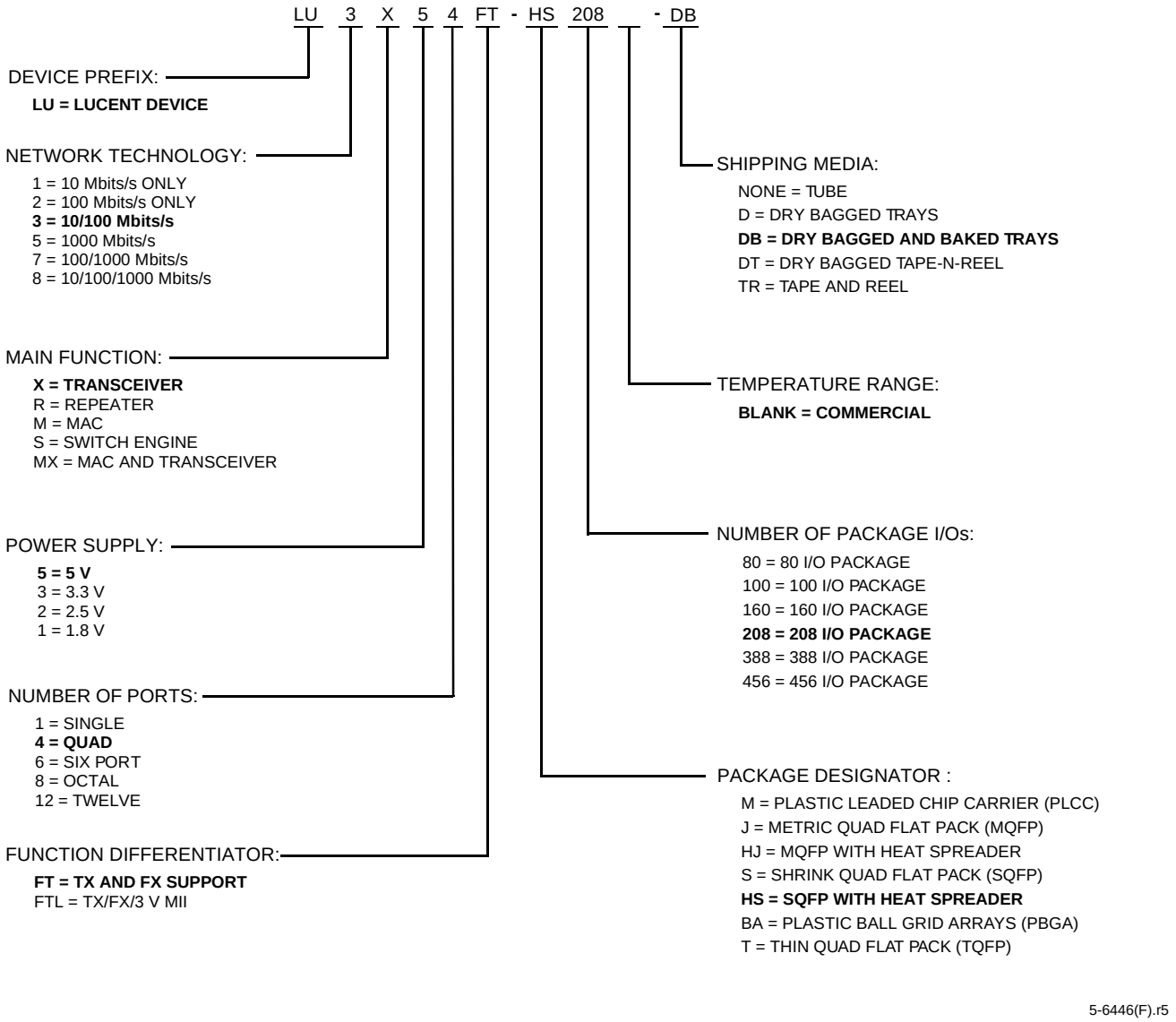


Figure 23. Device Nomenclature: Codes and Definitions

FASTCAT Ethernet IC Naming Convention (continued)

Figure 24 is an illustrated example of an Ethernet IC device I/O package showing Lucent's *FASTCAT* trademark.

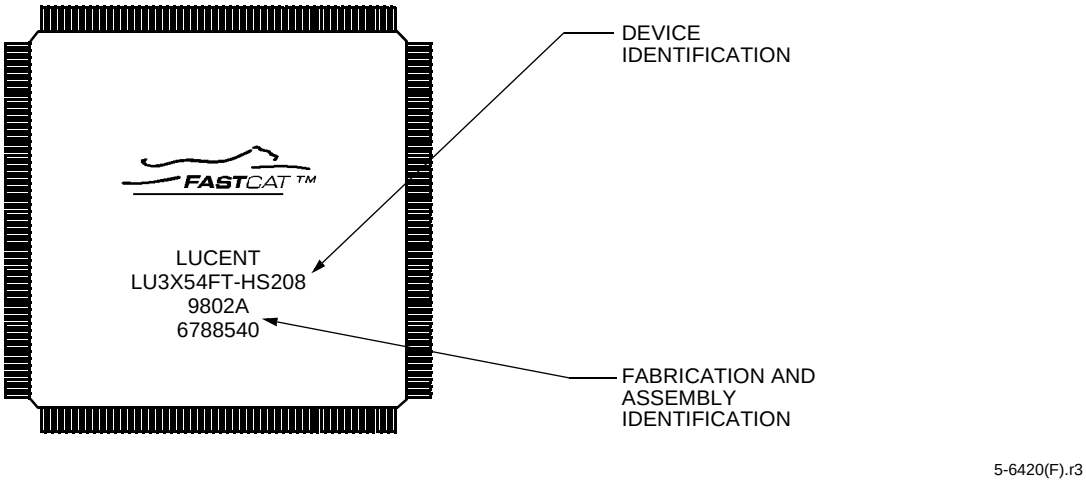


Figure 24. Top View of I/O Package

Technical Document Types

The following descriptions pertain to the types of individual product data sheets.

Data sheets provide a definition of the particular integrated circuit device by detailing its full electrical and physical specifications. They are intended to be the basic source of information for designers of new systems and to provide data for users requiring information on equipment troubleshooting, training, incoming inspection, equipment testing, and system design modification.

A data sheet is classified according to the following criteria:

Advance Data Sheet: An advance data sheet presents the device's proposed design architecture. It lists target specifications but may not have complete parameter values and is subject to change.

Preliminary Data Sheet: Preliminary data sheets describe the characteristics of initial prototypes.

Data Sheet: When a data sheet has the specifications of a product in full production and has complete parameter values, it is considered final and is classified as a data sheet.

Ordering Information

Device Code	Comcode	Package	Temperature
LU5M31-S208-DB	108193269	208-Pin SQFP	0 °C to 70 °C

For additional information, contact your Microelectronics Group Account Manager or the following:

INTERNET: **http://www.lucent.com/micro**
E-MAIL: **docmaster@micro.lucent.com**
N. AMERICA: Microelectronics Group, Lucent Technologies Inc., 555 Union Boulevard, Room 30L-15P-BA, Allentown, PA 18103
1-800-372-2447, FAX 610-712-4106 (In CANADA: **1-800-553-2448**, FAX 610-712-4106)
ASIA PACIFIC: Microelectronics Group, Lucent Technologies Singapore Pte. Ltd., 77 Science Park Drive, #03-18 Cintech III, Singapore 118256
Tel. (65) 778 8833, FAX (65) 777 7495
CHINA: Microelectronics Group, Lucent Technologies (China) Co., Ltd., A-F2, 23/F, Zao Fong Universe Building, 1800 Zhong Shan Xi Road, Shanghai 200233 P. R. China **Tel. (86) 21 6440 0468**, **ext. 316**, FAX (86) 21 6440 0652
JAPAN: Microelectronics Group, Lucent Technologies Japan Ltd., 7-18, Higashi-Gotanda 2-chome, Shinagawa-ku, Tokyo 141, Japan
Tel. (81) 3 5421 1600, FAX (81) 3 5421 1700
EUROPE: Data Requests: MICROELECTRONICS GROUP DATALINE: **Tel. (44) 1189 324 299**, FAX (44) 1189 328 148
Technical Inquiries: GERMANY: **(49) 89 95086 0** (Munich), UNITED KINGDOM: **(44) 1344 865 900** (Bracknell), FRANCE: **(33) 1 48 83 68 00** (Paris), SWEDEN: **(46) 8 600 7070** (Stockholm), FINLAND: **(358) 9 4354 2800** (Helsinki), ITALY: **(39) 2 6608131** (Milan), SPAIN: **(34) 1 807 1441** (Madrid)

Lucent Technologies Inc. reserves the right to make changes to the product(s) or information contained herein without notice. No liability is assumed as a result of their use or application. No rights under any patent accompany the sale of any such product(s) or information. *FASTCAT* is a trademark of Lucent Technologies Inc.

Copyright © 1998 Lucent Technologies Inc.
All Rights Reserved
Printed in U.S.A.

August 1998
DS98-351LAN (Replaces DS97-447LAN)

microelectronics group

Lucent Technologies
Bell Labs Innovations