agere systems

Advisory
April 2001

Ambassador ® T8110 PCI-Based H.100/H.110 Switch and Packet Payload Engine

Introduction

The purpose of this advisory is to provide information on two device issues found after testing models of the T8110 *Ambassador* device.

T8110 Model Testing

Testing of models of the T8110 has been recently completed with two device issues to report.

The two device issues only affect the microprocessor interface and packet switching capabilities. The T8110 can function as a 4096 connection standard telephony switch when using the PCI interface.

Issue 1:	Microprocessor interface: The RDY(DTACKn) signal can oscillate if the microprocessor device driving the microprocessor interface does not relinquish its RDn (or WRn) signal within one 65 MHz clock cycle after the reassertion of RDY (*Intel** mode) or deassertion of DTACKn (*Motorola*† mode).
Workaround:	The processor or board-level component driving the microprocessor port must deassert RDn or WRn immediately (within 15 ns) upon reassertion of RDY.
Issue 2:	Packet switch malfunction. The T8110 does not disable its upper byte lanes on the descriptor table update, resulting in an over-write of descriptor table data. The descriptor table update occurs as the last phase of a PCI Master PUSH & PULL cycle. This results in virtual channel connection malfunctions. TDM switching is unaffected.
Workaround:	A systemic workaround for the user is to keep a shadow table for the UOR portion of the descriptor table.

The T8110 silicon is currently being redesigned to correct these issues. For additional information, contact your local FAE (field application engineer), or call 1-800-372-2447.

* *Intel* is a registered trademark of Intel Corporation.
† *Motorola* is a registered trademark of Motorola Inc.

For additional information, contact your Agere Systems Account Manager or the following:
INTERNET: **http://www.agere.com**
E-MAIL: **docmaster@micro.lucent.com**
N. AMERICA: Agere Systems Inc., 555 Union Boulevard, Room 30L-15P-BA, Allentown, PA 18109-3286
1-800-372-2447, FAX 610-712-4106 (In CANADA: **1-800-553-2448**, FAX 610-712-4106)
ASIA PACIFIC: Agere Systems Singapore Pte. Ltd., 77 Science Park Drive, #03-18 Cintech III, Singapore 118256
Tel. (65) 778 8833, FAX (65) 777 7495
CHINA: Agere Systems (Shanghai) Co., Ltd., 33/F Jin Mao Tower, 88 Century Boulevard Pudong, Shanghai 200121 PRC
Tel. (86) 21 50471212, FAX (86) 21 50472266
JAPAN: Agere Systems 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: DATALINE: **Tel. (44) 7000 582 368**, FAX (44) 1189 328 148
Technical Inquiries: GERMANY: **(49) 89 95086 0** (Munich), UNITED KINGDOM: **(44) 1344 865 900** (Ascot),
FRANCE: **(33) 1 40 83 68 00** (Paris), SWEDEN: **(46) 8 594 607 00** (Stockholm), FINLAND: **(358) 9 3507670** (Helsinki),
ITALY: **(39) 02 6608131** (Milan), SPAIN: **(34) 1 807 1441** (Madrid)

April 2001
AY01-021CTI (must accompany DS00-012CTI)

Ambassador® T8110 PCI-Based H.100/H.110 Switch and Packet Payload Engine

Introduction

This advisory describes T8110 pin changes to the June 2000 advance data sheet. In addition, there are clarifications to minibridge circuit operation as described in the data sheet.

Pin Changes

Table 8 of the June 2000 Advance Data Sheet was printed incorrectly, the corrected material is shown below:

1. Page 15 MB_A10/uP_A10 is actually on ball K3, not J3 as the data sheet indicates.
   MB_A11/uP_A11 is actually on ball J3, not K3 as the data sheet indicates.

2 Page 16 CT_D2 is actually on ball C10, not C11 as the data sheet indicates.
CT_D3 is actually on ball C11, not C10 as the data sheet indicates.

3. Page 19 GP5 is actually on ball F3, not E3 as the data sheet indicates.
   GP6 is actually on ball E3, not F3 as the data sheet indicates.

Minibridge Clarification Summary

Microprocessor interface mode: nonissue.

PCI interface mode: minibridge function.

Section 10 Minibridge (page 106)

The Advance data sheet, June 2000, Section 10, requires clarification because it does not clearly state that there is a direct mapping of the PCI address bits to the minibridge address bits. Customers must be aware that MB_A[1:0] of a minibridge transaction is a direct pass-thru from the PCI_AD[1:0] of the address phase, and that, for PCI MEMORY transactions (which is the only type of transaction T8110 responds to), the value is always "00." From the June 2000 data sheet, the first paragraph of Section 10 states:

> "PCI access requests are converted to a simple interface to external devices hanging on the minibridge port. There are 8 chip select outputs, a read strobe, a write strobe, 16-bit address, and 16-bit data bus."

This section should also have stated:

> PCI_AD[15:0], during the address phase, is DIRECTLY MAPPED as the MB_A[15:0] address. Customers could possibly assume this, OR may assume that byte lane enables determine the state of MB_A[1:0].

In addition, be aware that the minibridge only operates as 16-bit-only transfers, with the data positioned only on bits [15:0] of PCI_AD during the data phase.

Replace Section 10.3 Minibridge Circuit Operation With the Following Text

The minibridge circuit accepts PCI memory read and memory write transactions and translates them into simple asynchronous* control strobes for external devices hanging off the minibridge port.

The PCI address is passed straight through to the minibridge port, so MB_A[15:0] is always DWORD-aligned (MB_A[1:0] = "00"). Transactions are always 16-bit data, with the data on the PCI side always positioned at bits PCI_AD[15:0]. Byte lane enables, PCI_CBEn[3:0], are ignored for minibridge transactions.

Refer to Section 4.1.1.6 on page 34 for more detail on the PCI side of the transactions. Refer to Figure 33 for the access cycle descriptions of the minibridge side of the transactions.

* Asynchronous relative to the PCI clock. Strobes are generated relative to the internal chip clock, in multiples of 65.536 MHz clock periods.

Add Section 10.4 Minibridge Operational Addressing

The operating space (in PCI) is from 0x70000— 0x7FFFC. The address presented on the minibridge side, MB_A[15:0], is a straight pass-thru of the PCI_AD[15:0] bits during the address phase of the PCI transaction. The eight chip selects decode address bits [15:13], so that the chip selects are active in the eight spaces shown in the table below.

Table 1. Minibridge Operational Addressing

A[15:13]	PCI Space	MB Space	Chip Select
000	0x70000—71FFC	0x0000—1FFC	0
001	0x72000—73FFC	0x2000—3FFC	1
010	0x74000—75FFC	0x4000—5FFC	2
011	0x76000—77FFC	0x6000—7FFC	3
100	0x78000—79FFC	0x8000—9FFC	4
101	0x7A000—7BFFC	0xA000—BFFC	5
110	0x7C000—7DFFC	0xC000—DFFC	6
111	0x7E000—7FFFC	0xE000—FFFC	7

Since the upper address lines are made available on the minibridge side in addition to the chip selects, it is possible to create variations of the selected spaces using a minimal amount of external logic.

Example: A (Posted) Write to or (Delayed) Read from PCI address 0x77A10 would occur at minibridge address 0x7A10 where CS3 was active. The user could choose to use the full 16-bit address (0x7A10), or use CS3 with a 13-bit address 0x1A10. (Both are simultaneously available.) The timing of the CS signal relative to the read, write, address, and data is set by the parameters for CS3 in PCI registers 0x00730— 00737.

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 18109-3286
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. 325**, 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) 7000 582 368**, FAX (44) 1189 328 148
Technical Inquiries: GERMANY: **(49) 89 95086 0** (Munich), UNITED KINGDOM: **(44) 1344 865 900** (Ascot),
FRANCE: **(33) 1 40 83 68 00** (Paris), SWEDEN: **(46) 8 594 607 00** (Stockholm), FINLAND: **(358) 9 3507670** (Helsinki),
ITALY: **(39) 02 6608131** (Milan), SPAIN: **(34) 1 807 1441** (Madrid)

December 2000
AY01-009CTI (Must accompany DS00-012CTI)

microelectronics group
Lucent Technologies
Bell Labs Innovations

Advisory
September 2000

microelectronics group

Lucent Technologies
Bell Labs Innovations

Ambassador® T8110 PCI-Based H.100/H.110 Switch and Packet Payload Engine

Introduction

This advisory describes T8110 pin changes, and a package type change, to the published June 2000 Ambassador® *T8110 PCI-Based H.100/H.110 Switch and Packet Payload Engine* Advance Data Sheet.

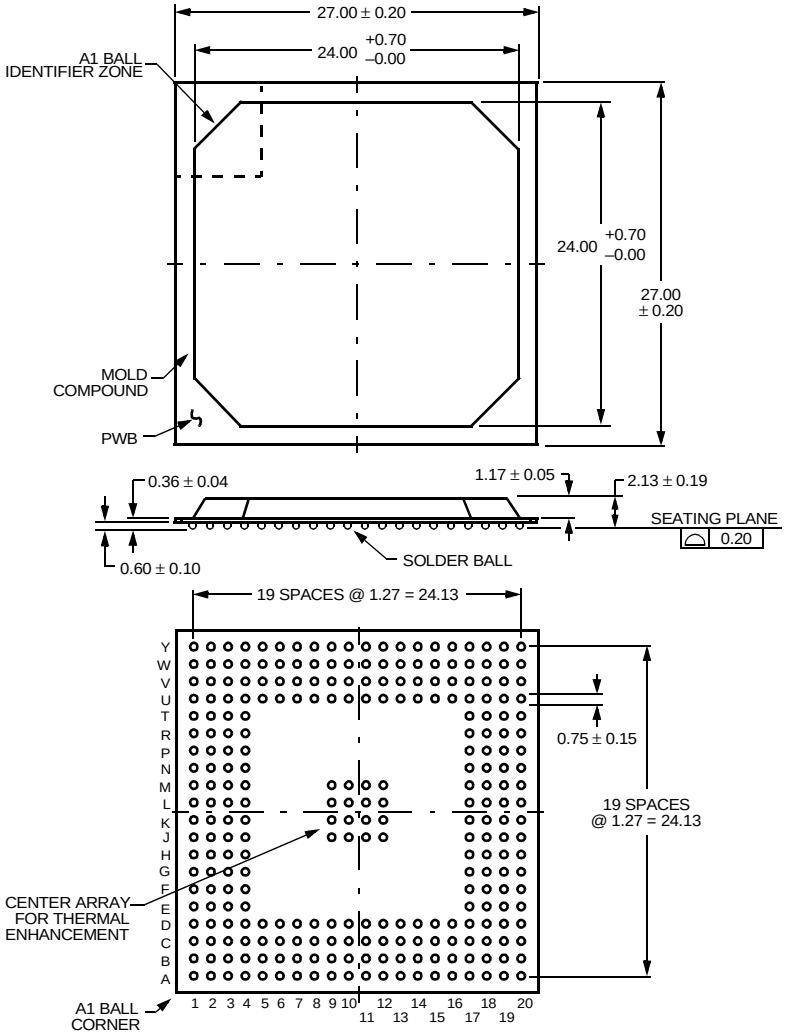
Pin Changes

SCAN_EN has been moved to be MUXed with PCI_IDSEL, and the former SCAN_EN is replaced with a new required signal for scan called TESTMODE. Since TESTMODE is active-high, an internal pull-down of 20 kΩ is now provided.

1. p. 15 PCI_IDSEL, W10, is replaced by PCI_IDSEL/SCAN_EN#. No pull-up is assigned to this signal. When TESTMODE = 1, this input will be assumed to be SCAN_EN# which is the signal used to shift in the scan chain.
2. p. 17 CT_C8A/B, A13/B13, and /CT_FRAME_A/B, A12/B12, should only have 50 kΩ pull-ups enabled by H100_ENABLE. Currently, the data sheet specifies an internal 20 kΩ pull-down for H.110. This has been removed. The correct pull-down of 10 kΩ needs to be externally supplied.
3. p. 19 SCAN_EN, F20, is replaced by TESTMODE (active-high). A pull-down of 20 kΩ is required. Other references in the data sheet will be corrected that describe the function of SCAN_EN. Basically, TESTMODE = 1 enables the scan operation and SCAN_EN# = 0 enables the shifting of data through the chain.
4. p. 19 EE_CS, G4, is an output that has an internal 20 kΩ pull-down specified. This resistor has been removed and is now specified as not applicable (NA).
5. p. 20 JTAG signals, TCK (E18), TMS (D18), TDI (F18) are incorrectly shown as internal 20 kΩ pull-downs, and each has been changed to internal 50 kΩ pull-up.

Package Type Change

The 272 PBGAM has been changed to a 272 PBGA. The footprint has not changed. The package thickness has been reduced.



5-4406.b

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 18109-3286
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. 325**, 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) 7000 582 368**, FAX (44) 1189 328 148
Technical Inquiries: GERMANY: **(49) 89 95086 0** (Munich), UNITED KINGDOM: **(44) 1344 865 900** (Ascot),
FRANCE: **(33) 1 40 83 68 00** (Paris), SWEDEN: **(46) 8 594 607 00** (Stockholm), FINLAND: **(358) 9 3507670** (Helsinki),
ITALY: **(39) 02 6608131** (Milan), SPAIN: **(34) 1 807 1441** (Madrid)

September 2000
AY00-030CTI (Must accompany DS00-012CTI)

microelectronics group

Lucent Technologies
Bell Labs Innovations

Advance Data Sheet
June 2000

microelectronics group

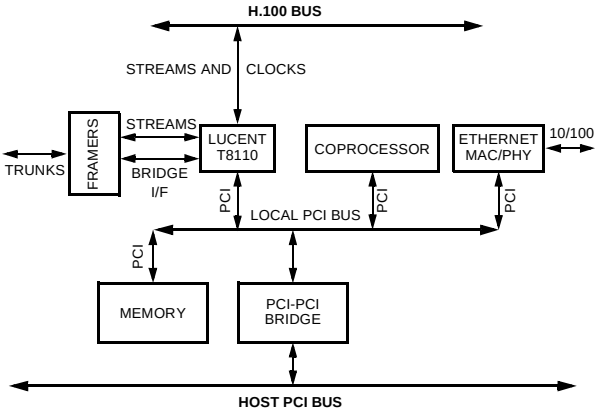
Lucent Technologies
Bell Labs Innovations

Ambassador ® T8110 PCI-Based H.100/H.110 Switch and Packet Payload Engine

1 Introduction

The T8110 is Lucent Technologies' newest addition to the *Ambassador* series of computer telephony integrated circuits. This device not only has the capabilities of previous members of the *Ambassador* family but also extends them by providing a flexible interface for switching packet payloads between a local PCI bus and the H.100/H.110 buses. This part is intended to work with a coprocessor for providing header, framing, and checksum generation. Since the T8110 operates purely on payloads, multiple protocols such as IP, ATM, and A-Bis can therefore be supported simultaneously. To reduce system integration costs support for non-PCI devices is provided through a minibridge.

A typical application, which provides connectivity for framers, IP over ethernet, and the H.100 bus is shown in Figure 1.



5-8921 (F)

Figure 1. Basic Application of the T8110 as a CT Switch and CT-IP Payload Processor

1.1 Features

- Combined master/slave PCI interface with burst
- Full H.100/H.110 support (32 data lines, all clock modes)
- 4,096-connection unified switch
- 32 local I/O lines (2, 4, 8 or 16 Mbits/s)
- Packet payload engine supports up to 512 virtual channels
- Interrupt controller with external inputs
- Minibridge with programmable chip selects
- 8 independent general-purpose I/O lines
- 8 independently programmed framing signals
- 4 local clocks
- T1/E1 rate adaptation
- Two clock fallback modes
- Hot swap friendly
- Single 3.3 V supply
- 272 PBGAM package

Table of Contents

Contents **Page**

Table of Contents (continued)

Contents **Page**

Table of Contents (continued)

Contents **Page**

Table of Contents (continued)

Contents **Page**

Table of Contents (continued)

Contents

Figure

Table of Contents (continued)

Figure **Page**

Table of Contents (continued)

Table **Page**

Table of Contents (continued)

Table **Page**

Table of Contents (continued)

Table **Page**

2 I/O Description

2.1 Interface Signals

Table 1. Interface Signals

Signal	I/O	Width	Function
PCI_AD	I/O	32	PCI Bus Address/Data
PCI_CBE#	I/O	4	PCI Bus Command/Byte Enable
PCI_CLK	I	1	PCI Bus Clock (33 MHz)
PCI_DEVSEL#	I/O	1	PCI Bus Device Select
PCI_FRAME#	I/O	1	PCI Bus Cycle Frame
PCI_GNT#	I	1	PCI Bus Grant
PCI_IDSEL	I	1	PCI Bus Initialization Device Select
PCI_INTA#	O	1	PCI Bus Interrupt
PCI_IRDY#	I/O	1	PCI Bus Initiator Ready
PCI_LOCK#	I	1	PCI Bus Lock
PCI_PAR	I/O	1	PCI Bus Parity
PCI_PERR#	I/O	1	PCI Bus Parity Error
PCI_REQ#	O	1	PCI Bus Request
PCI_RST#	I	1	PCI Bus Reset
PCI_SERR#	O	1	PCI Bus System Error
PCI_STOP#	I/O	1	PCI Bus Stop
PCI_TRDY#	I/O	1	PCI Bus Target Ready

2.1.1 Minibridge Interface Signals

Table 2. Minibridge Interface Signals

Signal	I/O	Width	Minibridge Function	ISA Interface Function
MB_A	I/O	16	Address[15:0] Out **Note:** Special Power-on Function for PCI CORE EEPROM. MB_A[3] = EE_SK_OUT MB_A[2] = EE_DI_OUT MB_A[1] = EE_DO_IN	Address[15:0] In
MB_D	I/O	16	Data Bus I/O	Data Bus In/Out
MB_RD	I/O	1	Read Strobe Out	RDn(DSn) In
MB_WR	I/O	1	Write Strobe Out	WRn(R/Wn) In
MB_CS0	I/O	1	Chip Select 0 Out	Address[16] In
MB_CS1	I/O	1	Chip Select 1 Out	Address[17] In
MB_CS2	I/O	1	Chip Select 2 Out	Address[18] In
MB_CS3	I/O	1	Chip Select 3 Out	Address[19] In
MB_CS4	I/O	1	Chip Select 4 Out	CSn In
MB_CS5	I/O	1	Chip Select 5 Out	Word/Byte Select In
MB_CS6	O	1	Chip Select 6 Out	RDY(DTACKn) Out
MB_CS7	I/O	1	Chip Select 7 Out	*Intel*/**Motorola*† Select In

* *Intel* is a registered trademark of Intel Corporation.
† *Motorola* is a registered trademark of Motorola, Inc.

2 I/O Description (continued)

2.1.2 H-Bus (H.100/H.110) Interface Signals

Table 3. H-Bus (H.100/H.110 Interface) Signals

Signal	I/O	Width	Function
VPRECHARGE	I	1	Precharge voltage for pull-downs, H.110 bus signals, CT_D, CT_NETREF1, CT_NETREF2
H110_ENABLE	I	1	Pull-down enable for H.110 bus signals, CT_D, CT_NETREF1, CT_NETREF2, CT_C8_A, CT_C8_B, /CT_FRAME_A, /CT_FRAME_B
H100_ENABLE	I	1	Pull-up enable for H.100 bus signals, CT_D, CT_NETREF1, CT_NETREF2, CT_C8_A, CT_C8_B, /CT_FRAME_A, /CT_FRAME_B
CT_D	I/O	32	H.100/H.110 bus data
CT_C8_A	I/O	1	H.100/H.110 bit clock A
/CT_FRAME_A	I/O	1	H.100/H.110 frame reference A
CT_C8_B	I/O	1	H.100/H.110 bit clock B
/CT_FRAME_B	I/O	1	H.100/H.110 frame reference B
CT_NETREF1	I/O	1	H.100/H.110 network reference 1
CT_NETREF2	I/O	1	H.100/H.110 network reference 2
/C16+	I/O	1	H-MVIP compatibility clock (16.384 MHz, differential)
/C16–	I/O	1	H-MVIP compatibility clock (16.384 MHz, differential)
/C4	I/O	1	MVIP compatibility clock (4.096 MHz)
C2	I/O	1	MVIP compatibility clock (2.048 MHz)
SCLK	I/O	1	SC-BUS compatibility clock
SCLKx2#	I/O	1	SC-BUS compatibility clock
/FR_COMP	I/O	1	Compatibility frame reference

2.1.3 L-Bus (Local) Interface Signals

Table 4. L-Bus (Local) Interface Signals

Signal	I/O	Width	Function
L_D	I/O	32	Local bus data
L_SC	O	4	Local bus clock outputs
FG	I/O	8	Local frame groups

2.1.4 Clock Circuit Interface Signals

Table 5. Clock Circuit Interface Signals

Signal	I/O	Width	Function
XTAL1_IN	I	1	Crystal oscillator #1 input (16.384 MHz)
XTAL1_OUT	O	1	Crystal oscillator #1 feedback
XTAL2_IN	I	1	Crystal oscillator #2 input (6.176 MHz or 12.352 MHz)
XTAL2_OUT	O	1	Crystal oscillator #2 feedback
PTEST	I	1	Analog PLL test mode enable
PEN	I	1	Analog PLL #1 and #2 enable
PSEL	I	1	Analog PLL #1 and #2 test rate selector

2 I/O Description (continued)

Table 5. Clock Circuit Interface Signals (continued)

Signal	I/O	Width	Function
PLOCK	O	1	Analog PLL #1 lock status indicator
PPDN	I	1	Analog PLL #1 and #2 test powerdown
LREF	I	8	Local clock reference inputs
TCLK_OUT	O	1	Internal chip clock output
PRI_REF_OUT	O	1	Main divider reference out for CLAD/DJAT
PRI_REF_IN	I	1	CLAD/DJAT reference in for analog PLL #1
NR1_SEL_OUT	O	1	CT_NETREF1 selection out for CLAD/DJAT
NR1_DIV_IN	I	1	CLAD/DJAT reference in for CT_NETREF1 divider
NR2_SEL_OUT	O	1	CT_NETREF2 selection out for CLAD/DJAT
NR2_DIV_IN	I	1	CLAD/DJAT reference in for CT_NETREF2 divider

2.1.5 GPIO Interface Signals

Table 6. GPIO Interface Signals

Signal	I/O	Width	GPIO Function	Alternate Function
GP0	I/O	1	GPIO bit 0 in/out	A-master indicator out
GP1	I/O	1	GPIO bit 1 in/out	B-master indicator out
GP2	I/O	1	GPIO bit 2 in/out	Forwarded PCI_RST# OUT
GP3	I/O	1	GPIO bit 3 in/out	NA
GP4	I/O	1	GPIO bit 4 in/out	NA
GP5	I/O	1	GPIO bit 5 in/out	NA
GP6	I/O	1	GPIO bit 6 in/out	NA
GP7	I/O	1	GPIO bit 7 in/out	NA

2.1.6 Miscellaneous Interface Signals

Table 7. Miscellaneous Interface Signals

Signal	I/O	Width	Function
RESET#	I	1	Chip reset
SYSERR	O	1	System error indicator
CLKERR	O	1	Clocking error indicator
LPUE	I	1	Pull-up enable for signals, FG, GP, L_D, LREF, MB_D, NR1_DIV_IN, NR2_DIV_IN, PRI_REF_IN
SCAN_EN	I	1	SCAN enable, active-low
EE_CS	O	1	EEPROM chip select
TRST#	I	1	JTAG reset
TCK	I	1	JTAG clock
TMS	I	1	JTAG mode select
TDI	I	1	JTAG data in
TDO	O	1	JTAG data out
VIO	I	1	PCI bus environment, apply GND for ISA interface, apply 3.3 V or 5 V for PCI interface

2 I/O Description (continued)

2.2 T8110 Pinout Information

The T8110 package is a 272-pin BPGAM ball grid array. Refer to the following table for ball assignment, buffer type, and pull-up/pull-down information.

Table 8. T8110 Pinouts

PCI Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
Y18	PCI_AD0	PCI I/O	(NA)
W17	PCI_AD1	PCI I/O	(NA)
V16	PCI_AD2	PCI I/O	(NA)
U16	PCI_AD3	PCI I/O	(NA)
Y17	PCI_AD4	PCI I/O	(NA)
Y16	PCI_AD5	PCI I/O	(NA)
W16	PCI_AD6	PCI I/O	(NA)
V15	PCI_AD7	PCI I/O	(NA)
Y15	PCI_AD8	PCI I/O	(NA)
W15	PCI_AD9	PCI I/O	(NA)
W14	PCI_AD10	PCI I/O	(NA)
V14	PCI_AD11	PCI I/O	(NA)
Y14	PCI_AD12	PCI I/O	(NA)
Y13	PCI_AD13	PCI I/O	(NA)
W13	PCI_AD14	PCI I/O	(NA)
V13	PCI_AD15	PCI I/O	(NA)
Y9	PCI_AD16	PCI I/O	(NA)
W9	PCI_AD17	PCI I/O	(NA)
V9	PCI_AD18	PCI I/O	(NA)
V8	PCI_AD19	PCI I/O	(NA)
Y8	PCI_AD20	PCI I/O	(NA)
W8	PCI_AD21	PCI I/O	(NA)
W7	PCI_AD22	PCI I/O	(NA)
V7	PCI_AD23	PCI I/O	(NA)
Y7	PCI_AD24	PCI I/O	(NA)
Y6	PCI_AD25	PCI I/O	(NA)
W6	PCI_AD26	PCI I/O	(NA)
V6	PCI_AD27	PCI I/O	(NA)
Y5	PCI_AD28	PCI I/O	(NA)
W5	PCI_AD29	PCI I/O	(NA)
V5	PCI_AD30	PCI I/O	(NA)
V4	PCI_AD31	PCI I/O	(NA)
U14	PCI_CBE0#	PCI I/O	(NA)
U12	PCI_CBE1#	PCI I/O	(NA)
U9	PCI_CBE2#	PCI I/O	(NA)

2 I/O Description (continued)

Table 8. T8110 Pinouts (continued)

PCI Interface (continued)			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
U7	PCI_CBE3#	PCI I/O	(NA)
Y3	PCI_CLK	PCI input	(NA)
W11	PCI_DEVSEL#	PCI I/O	(NA)
Y10	PCI_FRAME#	PCI I/O	(NA)
W4	PCI_GNT#	PCI input	(NA)
W10	PCI_IDSEL	PCI input	(NA)
Y4	PCI_INTA#	PCI output/open drain	(NA)
Y11	PCI_IRDY#	PCI I/O	(NA)
V10	PCI_LOCK#	PCI input	(NA)
U11	PCI_PAR	PCI I/O	(NA)
W12	PCI_PERR#	PCI I/O	(NA)
W3	PCI_REQ#	PCI output	(NA)
Y2	PCI_RST#	PCI input	(NA)
V12	PCI_SERR#	PCI output/open drain	(NA)
V11	PCI_STOP#	PCI I/O	(NA)
Y12	PCI_TRDY#	PCI I/O	(NA)
Minibridge Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull Up/Down**
F1	MB_A0/uP_AO	8 mA I/O–Schmitt	20 kΩ down
G1	MB_A1/uP_A1/EE_DO	8 mA I/O–Schmitt	20 kΩ down
J3	MB_A10/uP_A10	8 mA I/O–Schmitt	20 kΩ down
K3	MB_A11/uP_A11	8 mA I/O–Schmitt	20 kΩ down
K1	MB_A12/uP_A12	8 mA I/O–Schmitt	20 kΩ down
K2	MB_A13/uP_A13	8 mA I/O-Schmitt	20 kΩ down
L3	MB_A14/uP_A14	8 mA I/O-Schmitt	20 kΩ down
L4	MB_A15/uP_A15	8 mA I/O-Schmitt	20 kΩ down
G2	MB_A2/uP_A2/EE_DI	8 mA I/O-Schmitt	20 kΩ down
G3	MB_A3/uP_A3/EE_SK	8 mA I/O-Schmitt	20 kΩ down
H1	MB_A4/uP_A4	8 mA I/O-Schmitt	20 kΩ down
H2	MB_A5/uP_A5	8 mA I/O-Schmitt	20 kΩ down
H3	MB_A6/uP_A6	8 mA I/O-Schmitt	20 kΩ down
J4	MB_A7/uP_A7	8 mA I/O-Schmitt	20 kΩ down
J1	MB_A8/uP_A8	8 mA I/O-Schmitt	20 kΩ down
J2	MB_A9/uP_A9	8 mA I/O-Schmitt	20 kΩ down
W1	MB_D0	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
V1	MB_D1	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
V2	MB_D2	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
U3	MB_D3	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
U1	MB_D4	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
U2	MB_D5	8 mA I/O-Schmitt	LPUE: 50 kΩ Up

2 I/O Description (continued)

Table 8. T8110 Pinouts (continued)

Minibridge Interface (continued)			
Ball	**Pin Name**	**Buffer Type**	**Pull Up/Down**
T3	MB_D6	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
T4	MB_D7	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
T1	MB_D8	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
T2	MB_D9	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
R3	MB_D10	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
P4	MB_D11	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
R1	MB_D12	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
R2	MB_D13	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
P2	MB_D14	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
P3	MB_D15	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
N1	MB_RD/uP_RD#(DS#)	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
P1	MB_WR/uP_WR#(R/W#)	8 mA I/O-Schmitt	LPUE: 50 kΩ Up
L1	MB_CS0/uP_A16	8 mA I/O-Schmitt	20 kΩ down
L2	MB_CS1/uP_A17	8 mA I/O-Schmitt	20 kΩ down
M1	MB_CS2/uP_A18	8 mA I/O-Schmitt	20 kΩ down
M2	MB_CS3/uP_A19	8 mA I/O-Schmitt	20 kΩ down
M3	MB_CS4/uP_CSn	8 mA I/O-Schmitt	LPUE: 50 kΩ up
M4	MB_CS5/uP_WB_Sel	8 mA I/O-Schmitt	LPUE: 50 kΩ up
N2	MB_CS6/uP_RDY(DTACK#)	8 mA 3-state	(NA)
N3	MB_CS7/IM_Sel	8 mA I/O-Schmitt	LPUE: 50 kΩ up
C1	VPRECHARGE	—	(NA)
D5	H.110 Enable	Input	20 kΩ down
D7	H.100 Enable	Input	20 kΩ down
A11	CT_D0	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B11	CT_D1	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C11	CT_D2	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C10	CT_D3	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A10	CT_D4	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B10	CT_D5	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B9	CT_D6	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C9	CT_D7	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A9	CT_D8	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B8	CT_D9	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C8	CT_D10	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A8	CT_D11	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C7	CT_D12	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A7	CT_D13	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B7	CT_D14	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C6	CT_D15	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A6	CT_D16	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre

2 I/O Description (continued)

Table 8. T8110 Pinouts (continued)

H-Bus Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
B6	CT_D17	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C5	CT_D18	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A5	CT_D19	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B5	CT_D20	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A4	CT_D21	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B4	CT_D22	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C4	CT_D23	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A3	CT_D24	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B3	CT_D25	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C3	CT_D26	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A2	CT_D27	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B2	CT_D28	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B1	CT_D29	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C2	CT_D30	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
D2	CT_D31	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
A13	CT_C8_A	24 mA I/O-Schmitt	Enabled: 50 kΩ up/20 kΩ down
A12	/CT_FRAME_A	24 mA I/O-Schmitt	Enabled: 50 kΩ up/20 kΩ down
B13	CT_C8_B	24 mA I/O-Schmitt	Enabled: 50 kΩ up/20 kΩ down
B12	/CT_FRAME_B	24 mA I/O-Schmitt	Enabled: 50 kΩ up/20 kΩ down
A14	CT_NETREF1	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
B14	CT_NETREF2	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
D9	/C16+	24 mA I/O-Schmitt	50 kΩ up
D10	/C16–	24 mA I/O-Schmitt	50 kΩ up
D12	/C4	8 mA I/O-Schmitt	50 kΩ up
D14	C2	8 mA I/O-Schmitt	50 kΩ up
C14	SCLK	24 mA I/O-Schmitt	50 kΩ up
C13	SCLKx2	24 mA I/O-Schmitt	50 kΩ up
C12	/FR_COMP	24 mA I/O-Schmitt	50 kΩ up
J20	LD0	8 mA I/O-Schmitt	LPUE: 50 kΩ up
J19	LD1	8 mA I/O-Schmitt	LPUE: 50 kΩ up
J18	LD2	8 mA I/O-Schmitt	LPUE: 50 kΩ up
K17	LD3	8 mA I/O-Schmitt	LPUE: 50 kΩ up
K20	LD4	8 mA I/O-Schmitt	LPUE: 50 kΩ up
K19	LD5	8 mA I/O-Schmitt	LPUE: 50 kΩ up
K18	LD6	8 mA I/O-Schmitt	LPUE: 50 kΩ up
L18	LD7	8 mA I/O-Schmitt	LPUE: 50 kΩ up
L20	LD8	8 mA I/O-Schmitt	LPUE: 50 kΩ up
L19	LD9	8 mA I/O-Schmitt	LPUE: 50 kΩ up
M18	LD10	8 mA I/O-Schmitt	LPUE: 50 kΩ up
M17	LD11	8 mA I/O-Schmitt	LPUE: 50 kΩ up

2 I/O Description (continued)

Table 8. T8110 Pinouts (continued)

H-Bus Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull Up/Down**
M20	LD12	8 mA I/O-Schmitt	LPUE: 50 kΩ up
M19	LD13	8 mA I/O-Schmitt	LPUE: 50 kΩ up
N19	LD14	8 mA I/O-Schmitt	LPUE: 50 kΩ up
N18	LD15	8 mA I/O-Schmitt	LPUE: 50 kΩ up
N20	LD16	8 mA I/O-Schmitt	LPUE: 50 kΩ up
P20	LD17	8 mA I/O-Schmitt	LPUE: 50 kΩ up
P19	LD18	8 mA I/O-Schmitt	LPUE: 50 kΩ up
P18	LD19	8 mA I/O-Schmitt	LPUE: 50 kΩ up
R20	LD20	8 mA I/O-Schmitt	LPUE: 50 kΩ up
R19	LD21	8 mA I/O-Schmitt	LPUE: 50 kΩ up
R18	LD22	8 mA I/O-Schmitt	LPUE: 50 kΩ up
P17	LD23	8 mA I/O-Schmitt	LPUE: 50 kΩ up
T20	LD24	8 mA I/O-Schmitt	LPUE: 50 kΩ up
T19	LD25	8 mA I/O-Schmitt	LPUE: 50 kΩ up
T18	LD26	8 mA I/O-Schmitt	LPUE: 50 kΩ up
U20	LD27	8 mA I/O-Schmitt	LPUE: 50 kΩ up
V20	LD28	8 mA I/O-Schmitt	LPUE: 50 kΩ up
U19	LD29	8 mA I/O-Schmitt	LPUE: 50 kΩ up
L-Bus Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
U18	LD30	8 mA I/O-Schmitt	LPUE: 50 kΩ up
T17	LD31	8 mA I/O-Schmitt	LPUE: 50 kΩ up
H20	L_SC0/PLL2_CK1x	8 mA 3-state	(NA)
H19	L_SC1/PLL_REFCLK1x	8 mA 3-state	(NA)
H18	L_SC2/PLL2_VCOP	8 mA 3-state	(NA)
G19	L_SC3/PLL1_VCOP	8 mA 3-state	(NA)
Y20	FG0	8 mA I/O-Schmitt	LPUE: 50 kΩ up
Y19	FG1	8 mA I/O-Schmitt	LPUE: 50 kΩ up
W20	FG2	8 mA I/O-Schmitt	LPUE: 50 kΩ up
W19	FG3	8 mA I/O-Schmitt	LPUE: 50 kΩ up
W18	FG4	8 mA I/O-Schmitt	LPUE: 50 kΩ up
V19	FG5	8 mA I/O-Schmitt	LPUE: 50 kΩ up
V18	FG6	8 mA I/O-Schmitt	LPUE: 50 kΩ up
V17	FG7	8 mA I/O-Schmitt	LPUE: 50 kΩ up
Clock Circuit Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
B20	XTAL1_IN	Input	(NA)
C19	XTAL1_OUT	Xtal feedback	(NA)
E20	XTAL2_IN	Input	(NA)

2 I/O Description (continued)

Table 8. T8110 Pinouts (continued)

Clock Circuit Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
F19	XTAL2_OUT	Xtal feedback	(NA)
E17	PTest	Input	20 kΩ down
A20	PEN	Input	50 kΩ up
D20	PSEL	Input	20 kΩ down
D16	PLOCK	8 mA 3-state	NA
G17	PPDN	Input	20 kΩ down
A15	LRef0	Input-Schmitt	LPUE: 50 kΩ up
B15	LRef1	Input-Schmitt	LPUE: 50 kΩ up
C15	LRef2	Input-Schmitt	LPUE: 50 kΩ up
C16	LRef3	Input-Schmitt	LPUE: 50 kΩ up
A16	LRef4	Input-Schmitt	LPUE: 50 kΩ up
B16	LRef5	Input-Schmitt	LPUE: 50 kΩ up
B17	LRef6	Input-Schmitt	LPUE: 50 kΩ up
C17	LRef7	Input-Schmitt	LPUE: 50 kΩ up
G20	TCLK_OUT/PLL1_REFCLK1z	8 mA 3-state	(NA)
A17	PRI_REF_OUT/PLL1_CK4x	8 mA 3-state	(NA)
A18	PRI_REF_IN	Input-Schmitt	LPUE: 50 kΩ up
B18	NR1_SEL_OUT/PLL1_CK2x	8 mA 3-state	(NA)
A19	NR1_DIV_IN	Input-Schmitt	LPUE: 50 kΩ up
D19	NR2_SEL_OUT/PLL1_CK1x	8 mA 3-state	(NA)
C20	NR2_DIV_IN	Input-Schmitt	LPUE: 50 kΩ up
GPIO Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull-up-down**
D1	GP0/Amaster	8 mA I/O-Schmitt	LPUE: 50 kΩ up
E1	GP1/BMaster	8 mA I/O-Schmitt	LPUE: 50 kΩ up
E2	GP2/fwd_PCIRST	8 mA I/O-Schmitt	LPUE: 50 kΩ up
F2	GP3	8 mA I/O-Schmitt	LPUE: 50 kΩ up
D3	GP4	8 mA I/O-Schmitt	LPUE: 50 kΩ up
E3	GP5	8 mA I/O-Schmitt	LPUE: 50 kΩ up
F3	GP6	8 mA I/O-Schmitt	LPUE: 50 kΩ up
E4	GP7	8 mA I/O-Schmitt	LPUE: 50 kΩ up
Miscellaneous Interfaces			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
Y1	RESET#	Input-Schmitt	50 kΩ up
V3	SYSERR	8 mA 3-state	(NA)
W2	CLKERR	8 mA 3-state	(NA)
J17	LPUE	Input	50 kΩ up
F20	SCAN_EN	Input-Schmitt	50KΩ up
G4	EE_CS	8 mA 3-state	20 kΩ down

2 I/O Description (continued)

Table 8. T8110 Pinouts (continued)

Miscellaneous Interfaces			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
C18	TRST#	Input-Schmitt	50 kΩ up
E18	TCK	Input-Schmitt	20 kΩ down
D18	TMS	Input-Schmitt	20 kΩ down
F18	TDI	Input-Schmitt	20 kΩ down
G18	TDO	4 mA 3-state	(NA)
U5	VIO	—	—
Power			
Ball	**Pin Name**	**Voltage Type**	**Pull-Up/Down**
B19	$PLLV_{DD1}$	Analog V_{DD}	(NA)
E19	$PLLV_{DD2}$	Analog V_{DD}	(NA)
D6	V_{DD}	—	—
D11	V_{DD}	—	—
D15	V_{DD}	—	—
F4	V_{DD}	—	—
F17	V_{DD}	—	—
K4	V_{DD}	—	—
L17	V_{DD}	—	—
R4	V_{DD}	—	—
R17	V_{DD}	—	—
U6	V_{DD}	—	—
U10	V_{DD}	—	—
U15	V_{DD}	—	—
Ground			
Ball	**Pin Name**	**NA**	**NA**
A1	V_{SS}	—	—
D4	V_{SS}	—	—
D8	V_{SS}	—	—
D13	V_{SS}	—	—
D17	V_{SS}	—	—
H4	V_{SS}	—	—
H17	V_{SS}	—	—
N4	V_{SS}	—	—
N17	V_{SS}	—	—
U4	V_{SS}	—	—
U8	V_{SS}	—	—
U13	V_{SS}	—	—
U17	V_{SS}	—	—

2 I/O Description (continued)

Table 8. T8110 Pinouts (continued)

Thermal Ground			
Ball	**Pin Name**	**NA**	**NA**
J9—12	—	—	—
K9—12	—	—	—
L9—12	—	—	—
M9—12	—	—	—

2.3 Special Buffer Requirements

2.3.1 H.1x0 Bus Signal Internal Pull-Up/pull-Down

The H.1x0 bus pins require special consideration for H.100 and H.110 usage. There are two control pins to select between various internal bus pull-ups:

- H100_ENABLE. Enables internal 50 kΩ pull-ups on all H.1x0 bus signals.
- H110_ENABLE. Enables internal 20 kΩ pull-downs on all H.1x0 bus signals. Additionally, the precharge input provides a 0.7V precharge for the 20 kΩ pull-downs on the H.1x0 Data (CT_D[31:0]) and netref (CT_NETREF1,2) pins.

Note: The two H.1x0 enables are active-high. Only one or the other should ever be asserted.

Warning: Do not assert both at the same time.

Refer to Figure 2 for more detail.

2 I/O Description (continued)

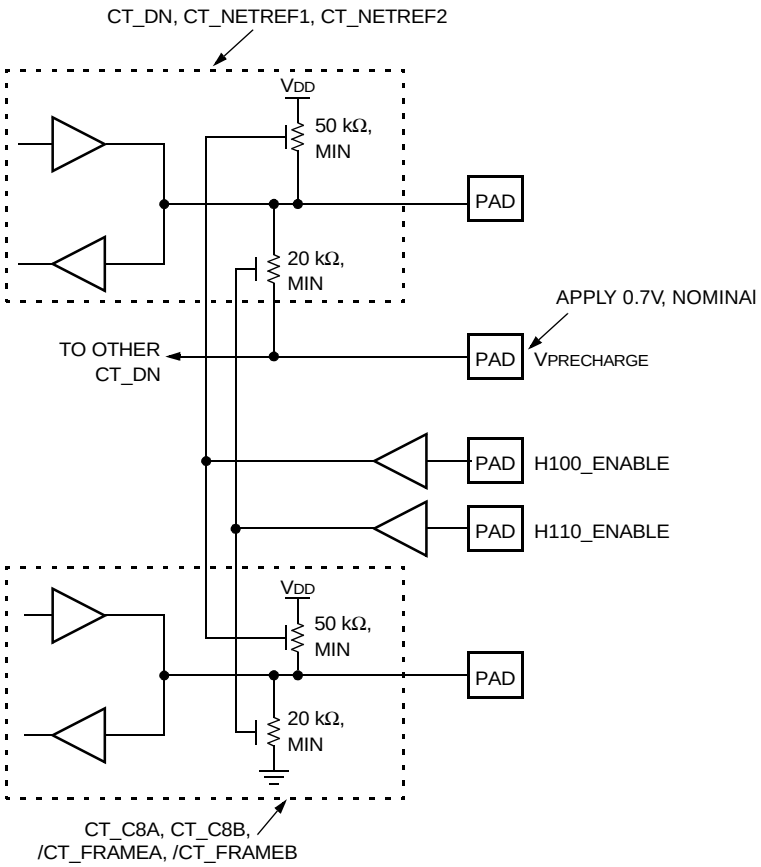


Figure 2. T8110 Pull-Up/Pull-Down Arrangement for H.1x0 Pins

2.3.2 Local Bus Signal Internal Pull-up/Pull-down

The LPUE input is active-high, and is used to activate pull-ups on the following local signals: GP[7:0], FG[7:0], MB_D[15:0], LD[31:0], LREF[7:0], PRI_REF_IN, NR1_DIV_IN, and NR2_DIV_IN

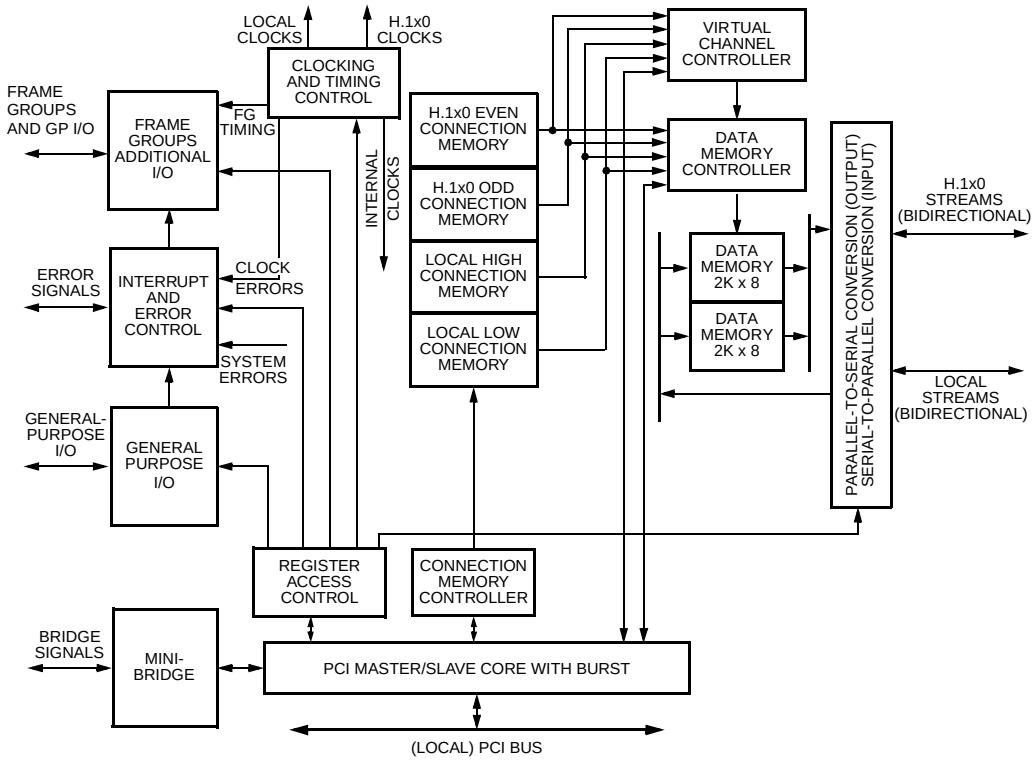
3 Main Architectural Features

3.1 T8110 Architecture

The T8110 includes all of the clocking and standard switching functions found on previous *Ambassador* devices, plus additional functionalities which are described in the following sections. There are two architectures—PCI (see Section 4.1 on page 26) and ISA (see Section 4.2 on page 39).

The local PCI bus interface allows the T8110 to act as a target (access control registers, memories, etc.) and as an initiator. The T8110 performs standard H-bus/L-bus switching, and the capability of the initiator allows an interface for switching packet payloads to/from the H-bus/L-bus; see Section 13 on page 132 more details. With this architecture selection, the minibridge port converts PCI target accesses into a simple handshake, and passes these accesses to external devices connected to this port; see section 10, starting on page 106.

The ISA bus interface allows the T8110 to perform standard H-bus/L-bus switching (i.e., there is no packet payload switching between the H-bus/L-bus and the ISA port). With this architecture, the minibridge port is used as the ISA bus port, and the PCI interface is ignored (see Section 4.2 on page 39).



5-8920 (F)

Figure 3. T8110 Block Diagram

3 Main Architectural Features (continued)

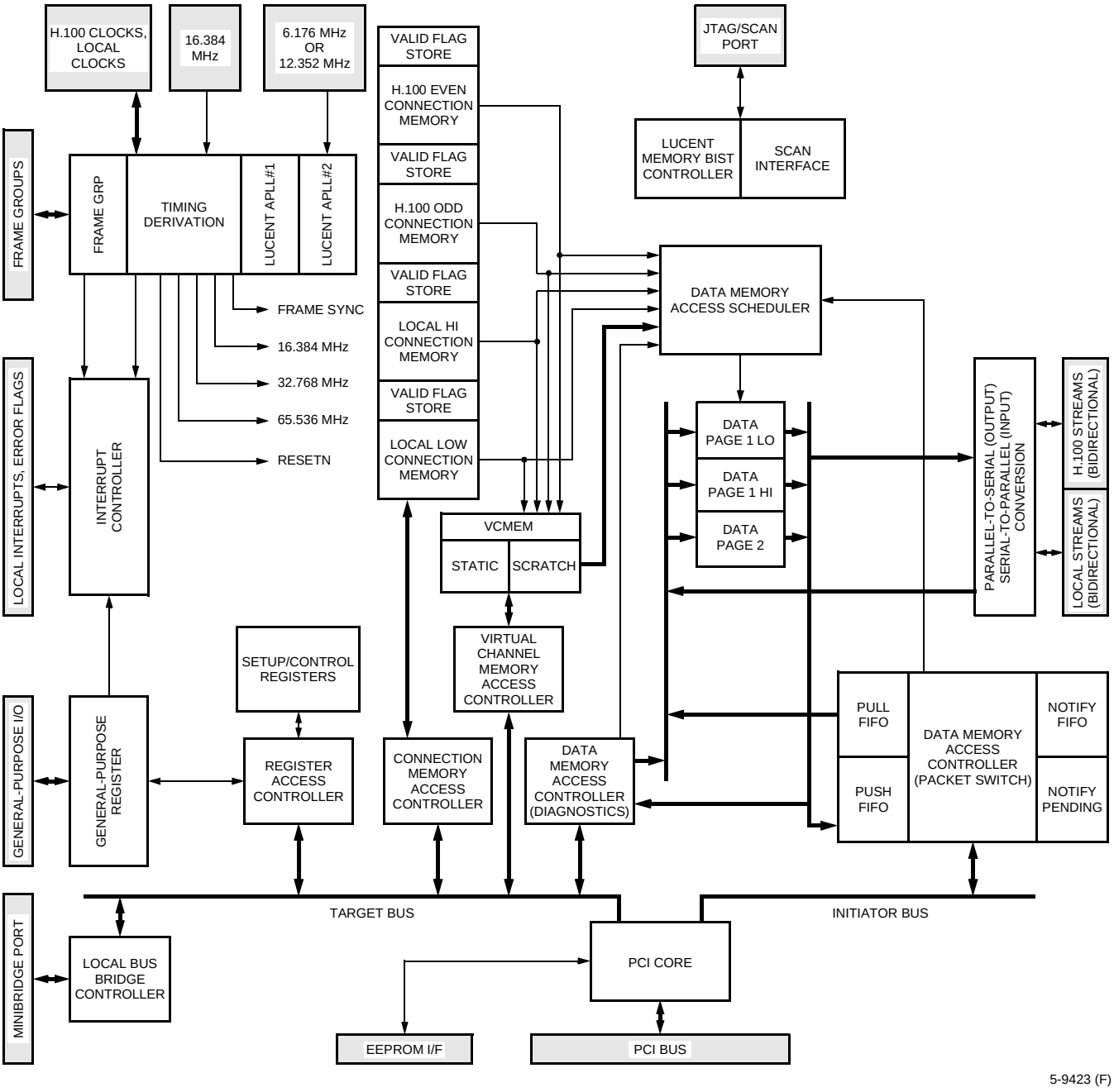


Figure 4. T8110 Architecture 1—PCI Bus Interface

3 Main Architectural Features (continued)

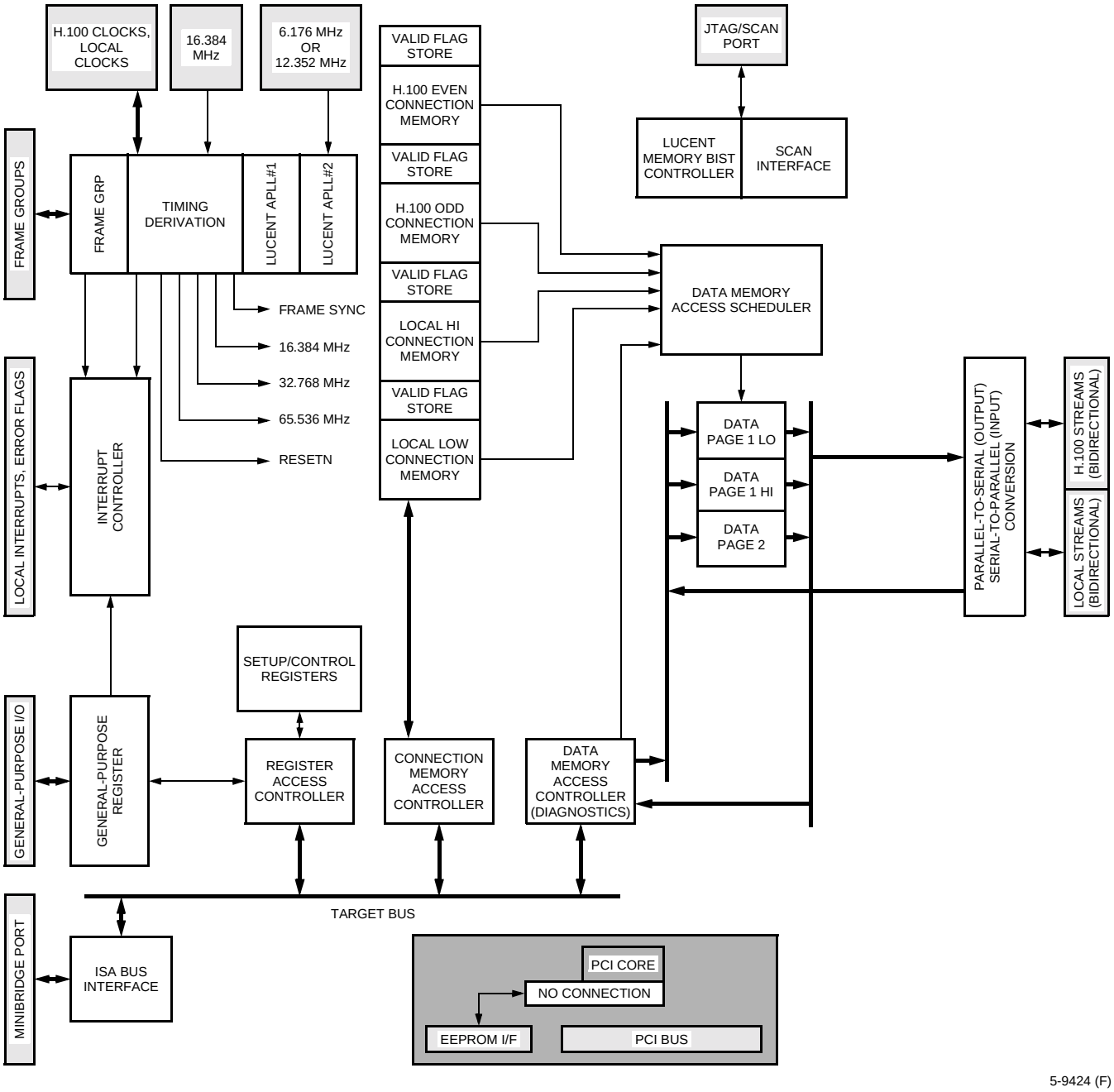


Figure 5. T8110 Architecture 2—ISA Bus Interface

4 Microprocessor Interface

The T8110 provides a selection of two interface mechanisms via the VIO input. This must be a static signal (either pulled high or pulled low).

- VIO tied to GND = T8110 interface to an ISA bus, connected via the minibridge port.
- VIO tied to 3.3 V = T8110 interface to a local PCI bus, 3.3 V signaling.
- VIO tied to 5 V = T8110 interface to a local PCI bus, 5 V signaling.

4.1 PCI Interface

The T8110 is a single function PCI device,;it can act as a target or an initiator. All addressing is DWORD aligned for 32-bit data transfers. Refer to Section 2.1 on page 11 for pin descriptions. When the PCI interface is selected, the minibridge port functions as a bridge to convert the PCI access protocol into a simple handshake protocol for external, non-PCI devices connected to this port. For more details, see Section 10 on page 106.

The PCI interface is arranged to provide a mixture of accesses. Initialization and register programming is typically under coprocessor control. As a result, the T8110 operates as a slave when being programmed by the coprocessor or by the host via a PCI-PCI bridge. Diagnostics and error handling are also defined as slave operations. However, when packets are processed by either taking data from the H.1x0 bus and passing it to memory, or when data is retrieved from memory and sent to the H.1x0 bus, the T8110 operates as a master, arbitrating for the bus and taking control of its own burst transactions. This ensures that the bandwidth required by the T8110 as a local PCI bus owner is kept to a minimum. Packet transactions are not limited to the H.1x0 bus and local time slots can be routed to and from the PCI bus, as well.

4.1.1 Target

The T8110 PCI bus interface allows target access to five internal regions: registers, connection memory, data memory, virtual channel memory, and the minibridge. Target burst transactions are only allowed to the register and connection memory space. No target bursts are allowed to/from the data memory, virtual channel memory, or the minibridge space. All target accesses get synchronized between the PCI's 33 MHz clock domain and the T8110's internal 65.536 MHz clock domain. Of the 32 bits of address provided, the upper 12 decode the base address, while the lower 20 provide addressing for the internal regions of the T8110, as shown in Table 9.

Table 9. T8110 Memory Mapping to PCI Space

Region	Subregion	Range (hex)
Registers	Reserved	0x00000—0x000FF
	Operating control and status	0x00100—0x001FF
	Clocks	0x00200—0x002FF
	Rate control	0x00300—0x003FF
	Frame group	0x00400—0x004FF
	General-purpose I/O	0x00500—0x005FF
	Interrupt control	0x00600—0x006FF
	Minibridge control	0x00700—0x007FF
	Reserved	0x00800—0x0FFFF
Virtual Channel Memory	—	0x10000—0x1FFFF
Data Memory	—	0x20000—0x2FFFF
Reserved	—	0x30000—0x3FFFF
Connection Memory	—	0x40000—0x4FFFF
Reserved	—	0x50000—0x6FFFF

4 Microprocessor Interface (continued)

Table 9. T8110 Memory Mapping to PCI Space (continued)

Region	Subregion	Range (hex)
Minibridge	—	0x70000—0x7FFFF
Reserved	—	0x80000—0xFFFFF

4.1.1.1 PCI Interface Registers

Table 10. PCI Interface Registers Map

DWORD Address (20 bits)	Section Cross Reference	Registers			
		Byte 3	Byte 2	Byte 1	Byte 0
0x00100	5.1.1, 5.1.2	Master enable	Reserved	Reset select	Soft reset
0x00104	5.1.3—5.1.4	Phase alignment select	Clock reg access select	Data mem mode select	VCstart
0x00108	5.1.4	Fallback trigger, upper	Fallback trigger, lower	Fallback type select	Fallback control
0x0010C	5.1.4	Watchdog EN, upper	Watchdog EN, lower	Watchdog select, Netref	Watchdog select, C8
0x00110	12.2.3.4	External buffers descriptor table—base address register [31:0]			
0x00114	4.1.5	Failsafe threshold high	Failsafe threshold low	Failsafe enable and status	Failsafe control
0x00120	5.2.1	Status 3—latched clock errors, upper	Status 2—latched clock errors, lower	Status 1—transient clock errors, upper	Status 0, transient clock errors, lower
0x00124	5.2.2, 5.2.3	Status 7, system errors, upper	Status 6, system errors, lower	Status 5	Status 4
0x00128	5.2.4	Device ID, upper	Device ID, lower	Reserved	Version ID
0x0012C	5.2.4	Reserved	Reserved	Status 9	Status 8
0x00140	12.1	Diag3	Diag2	Diag1	Diag0
0x00144	12.1	Diag7	Diag6	Diag5	Diag4
0x00148	12.1	Reserved	Reserved	Reserved	Diag8
0x00200	6.1	PLL#1 rate	PLL#1 input selector	Main divider	Main input selector
0x00204	6.1	PLL#2 rate	Reserved	Resource divider	Main inversion select
0x00208	6.1	DPLL1 rate	DPLL1 input selector	Reserved	LREF input select
0x0020C	6.1	DPLL2 rate	DPLL2 input selector	Reserved	LREF inversion select
0x00210	6.1	Reserved	Netref1 LREF select	Netref1 divider	Netref1 input selector
0x00214	6.1	Reserved	Netref2 LREF select	Netref2 divider	Netref2 input selector
0x00220	6.2	C8 output rate	FR_COMP width	Netref output enables	Master output enables

4 Microprocessor Interface (continued)

Table 10. PCI Interface Registers Map (continued)

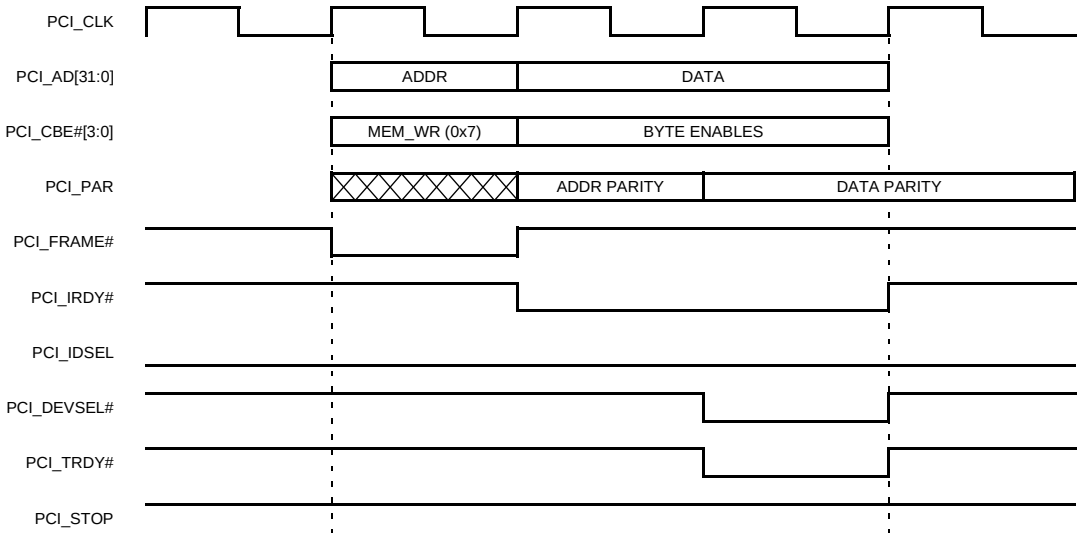
DWORD Address (20 bits)	Section Cross Reference	Registers			
		Byte 3	Byte 2	Byte 1	Byte 0
0x00224	6.2	SCLK output rate	TCLK select	Reserved	CCLK output enables
0x00228	6.2	L_SC3 select	L_SC2 select	L_SC1 select	L_SC0 select
0x00300	9.1	H-bus rate H/G	H-bus rate F/E	H-bus rate D/C	H-bus rate B/A
0x00320	9.2	L-bus rate H/G	L-bus rate F/E	L-bus rate D/C	L-bus rate B/A
0x00400	7.1	FG0 rate	FG0 width	FG0 upper start	FG0 lower start
0x00410	7.1	FG1 rate	FG1 width	FG1 upper start	FG1 lower start
0x00420	7.1	FG2 rate	FG2 width	FG2 upper start	FG2 lower start
0x00430	7.1	FG3 rate	FG3 width	FG3 upper start	FG3 lower start
0x00440	7.1	FG4 rate	FG4 width	FG4 upper start	FG4 lower start
0x00450	7.1	FG5 rate	FG5 width	FG5 upper start	FG5 lower start
0x00460	7.1	FG6 rate	FG6 width	FG6 upper start	FG6 lower start
0x00470	7.1	FG7 rate	FG7 width	FG7 upper start	FG7 lower start
0x00474	7.2	FG7 mode upper	FG7 mode lower	FG7 counter high byte	FG7 counter low byte
0x00480	7.3	Reserved	FGIO R/W	FGIO read mask	FGIO data register
0x00500	8.1	GPIO override	GPIO R/W	GPIO read mask	GPIO data register
0x00600	11.1	FGIO interrupt polarity	Reserved	FGIO interrupt enable	FGIO interrupt pending
0x00604	11.1	GPIO interrupt polarity	Reserved	GPIO interrupt enable	GPIO interrupt pending
0x00608	11.1	System interrupt enable, upper	System interrupt enable, lower	System interrupt pending, upper	System interrupt pending, lower
0x0060C	11.1	Clock interrupt enable, upper	Clock interrupt enable, lower	Clock interrupt pending, upper	Clock interrupt pend-ing, lower
0x00610	11.1	CLKERR output select	SYSERR output select	PCI_INTA output select	Arbitration control
0x00614	11.1	CLKERR pulse width	SYSERR pulse width	Reserved	Reserved
0x00618	11.1	In-service, byte 3	In-service, byte 2	In-service, byte 1	In-service, byte 0
0x00700	10.1	CS0 addr setup wait	CS0 read hold wait	CS0 read width wait	CS0 read setup wait
0x00704	10.1	CS0 addr hold wait	CS0 write hold wait	CS0 write width wait	CS0 write setup wait
0x00710	10.1	CS01 addr setup wait	CS1 read hold wait	CS1 read width wait	CS1 read setup wait

4 Microprocessor Interface (continued)

Table 10. PCI Interface Registers Map (continued)

DWORD Address (20 bits)	Section Cross Reference	Registers			
		Byte 3	**Byte 2**	**Byte 1**	**Byte 0**
0x00714	10.1	CS1 addr hold wait	CS1 write hold wait	CS1 write width wait	CS1 write setup wait
0x00720	10.1	CS2 addr setup wait	CS2 read hold wait	CS2 read width wait	CS2 read setup wait
0x00724	10.1	CS02 addr hold wait	CS2 write hold wait	CS2 write width wait	CS2 write setup wait
0x00730	10.1	CS03 addr setup wait	CS3 read hold wait	CS3 read width wait	CS3 read setup wait
0x00734	10.1	CS3 addr hold wait	CS3 write hold wait	CS3 write width wait	CS3 write setup wait
0x00740	10.1	CS4 addr setup wait	CS4 read hold wait	CS4 read width wait	CS4 read setup wait
0x00744	10.1	CS4 addr hold wait	CS4 write hold wait	CS4 write width wait	CS4 write setup wait
0x00750	10.1	CS5 addr setup wait	CS5 read hold wait	CS5 read width wait	CS5 read setup wait
0x00754	10.1	CS5 addr hold wait	CS5 write hold wait	CS5 write width wait	CS5 write setup wait
0x00760	10.1	CS6 addr setup wait	CS6 read hold wait	CS6 read width wait	CS6 read setup wait
0x00764	10.1	CS6 addr hold wait	CS6 write hold wait	CS6 write width wait	CS6 write setup wait
0x00770	10.1	CS7 addr setup wait	CS7 read hold wait	CS7 read width wait	CS7 read setup wait
0x00774	10.1	CS7 addr hold wait	CS7 write hold wait	CS7 write width wait	CS7 write setup wait
0x00780	10.1	Reserved	Reserved	RD-WR strobe inversion	CS strobe inversion

4 Microprocessor Interface (continued)



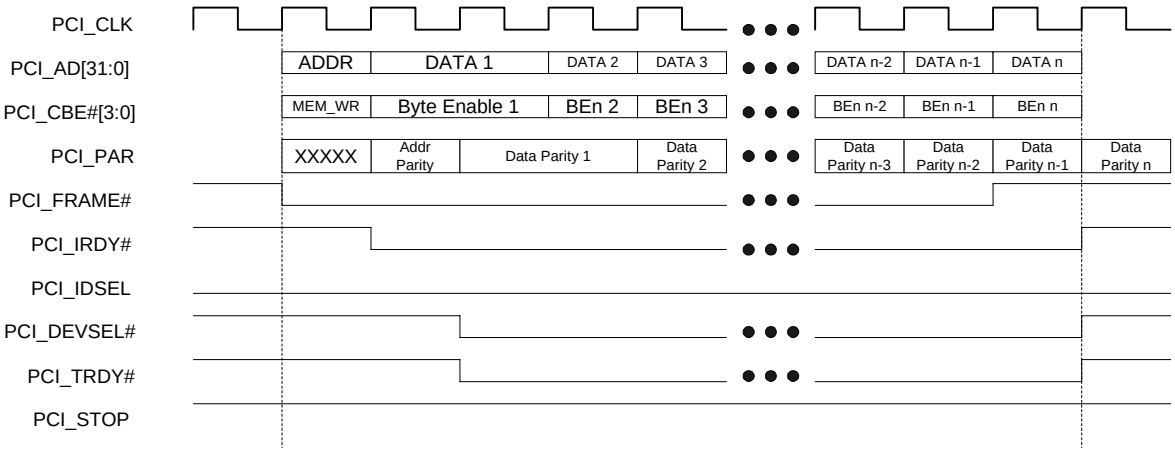
5-9612 (F)

Notes:

T8110 PCI i/f has medium decode speed. There is always a two-cycle turn-around between assertion of FRAME# and assertion of DEVSEL#.

All memory writes get posted to the T8110. Turn around time for a single cycle write is three PCI clocks.

Figure 6. T8110 PCI Interface—Single Write Cycle

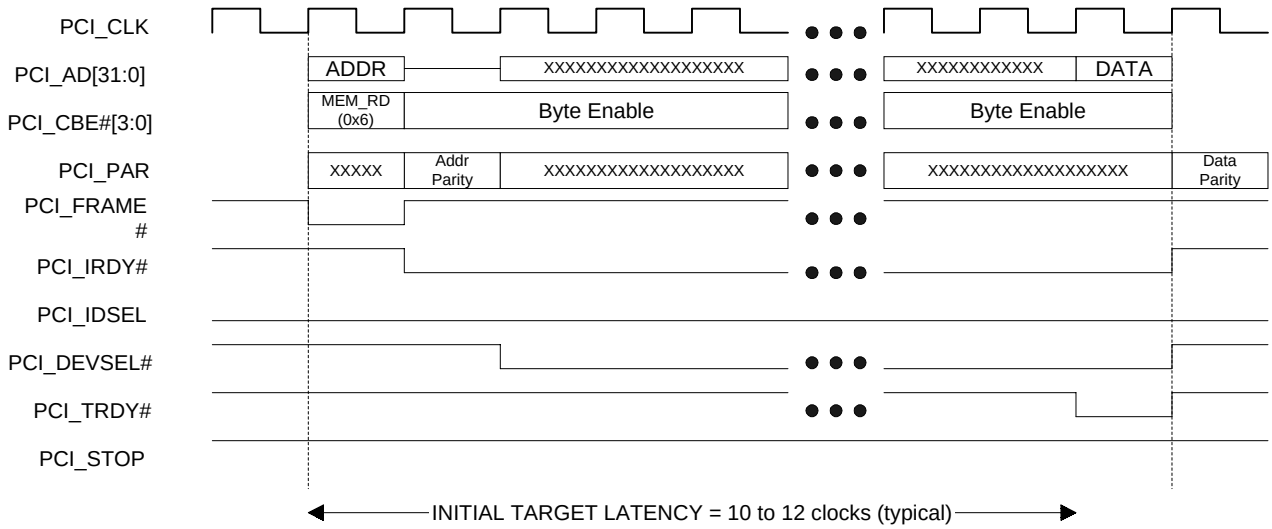


Notes:

1. T8110 PCI i/f has medium decode speed. There is always a two-cycle turn-around between assertion of frame# and assertion of DEVSEL#.
2. All memory writes get posted to the T8110. Turn-around time for the first data phase write is three PCI clocks.
3. PCI core write FIFO depth = 8, so up to 8 data words can immediately get posted.
4. For register region access, the application side operates at a faster rate than the PCI side, so the write FIFO will never become full, and TRDY# will remain active.
5. For connection memory access, the application side operates slightly slower than the PCI side, so it is possible to fill the write FIFO. In this case, the TRDY# signal is deasserted while the application side catches up.

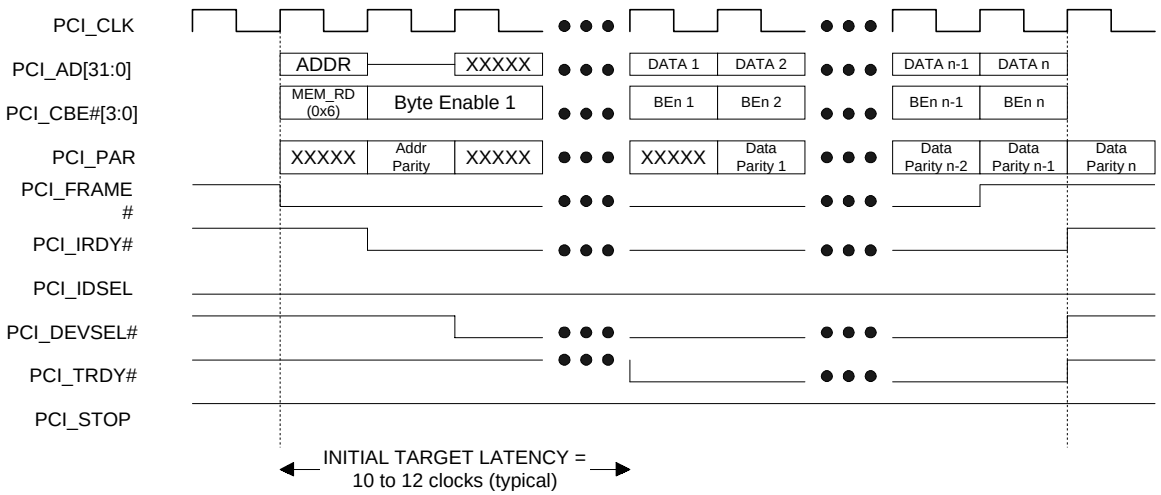
Figure 7. T8110 PCI Interface—Burst Write Cycle

4 Microprocessor Interface (continued)



NOTES:
1. T8110 PCI i/f has MEDIUM decode speed. There is always a two-cycle turn-around between assertion of FRAME# and assertion of DEVSEL#.
2. Turn-around time for memory reads from the T8110 is variable, depending on the region being accessed, and the synchronization time across the PCI clock and application clock domains. Initial target latency is typically between 10-12 PCI clock cycles.

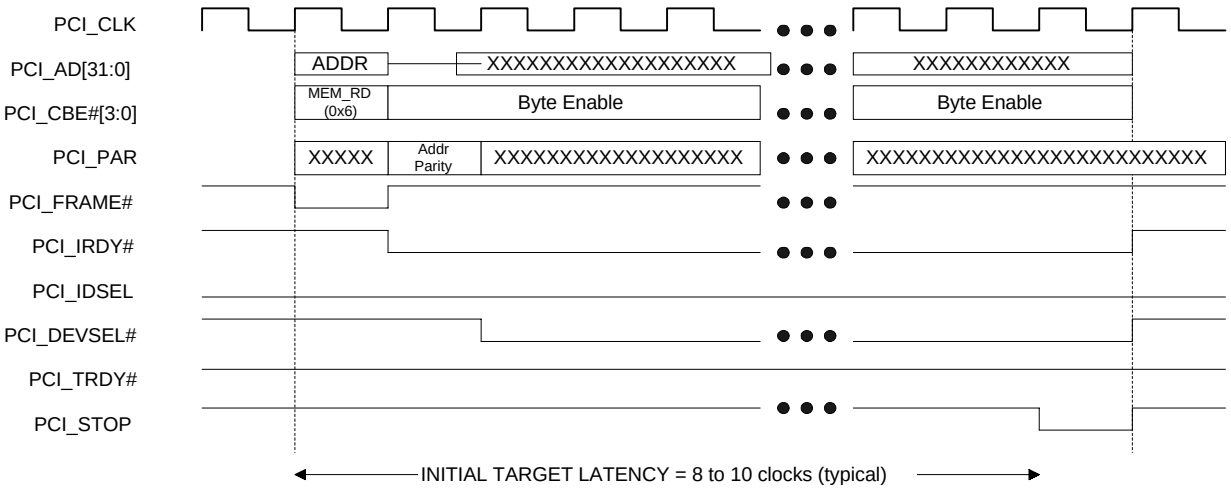
Figure 8. T8110 PCI Interface—Single Read Cycle



Notes:
1. T8110 PCI i/f has medium decode speed. There is always a two-cycle turn-around between assertion of frame# and assertion of DEVSEL#.
2. Turn-around time for memory reads from the T8110 is variable, depending on the region being accessed, and the synchronization time across the PCI clock and application clock domains. Initial target latency is typically between 10-12 PCI clock cycles.
3. PCI core read FIFO depth = 8.
4. For register region access, the application side operates at a faster rate than the PCI side, so the read FIFO will never become empty, and burst read data is returned as quickly as the PCI bus can accept it.
5. For connection memory access, the application side operates slightly slower than the PCI side, so it is possible to empty the read FIFO. In this case, the TRDY# signal is deasserted while the application side catches up.

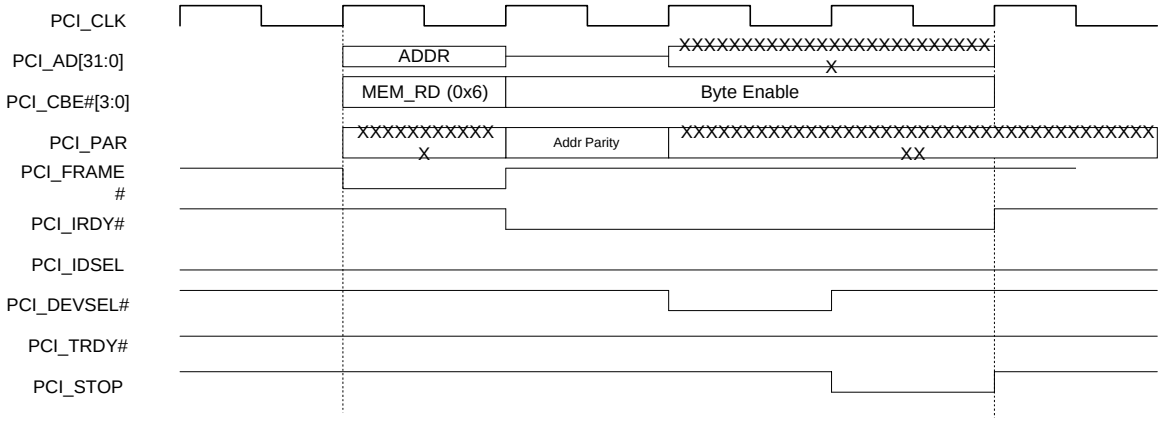
Figure 9. T8110 PCI Interface—Burst Read Cycle

4 Microprocessor Interface (continued)



Notes:
1. T8110 PCI i/f has medium decode speed. There is always a two-cycle turn-around between assertion of frame# and assertion of DEVSEL#.
2. Turn-around time for memory read retry is variable, depending on the region being accessed, and the synchronization time across the PCI clock and application clock domains. Initial target latency for a retry is typically between 8-10 PCI clock cycles.

Figure 10. T8110 PCI Interface—Delayed Read Cycle (Retry)



Notes:
1. T8110 PCI i/f has medium decode speed. There is always a two-cycle turn-around between assertion of frame# and assertion of DEVSEL#.
2. Turn-around time for memory read TARGET ABORT is 4 PCI clocks.

Figure 11. T8110 PCI Interface—Target Abort (Address Parity Error)

4 Microprocessor Interface (continued)

4.1.1.2 Register Space Target Access

The T8110 registers are always immediately available for access. Read and write bursting is allowed to this region. Read access to reserved addresses returns 0x00. For more details, on register programming, refer to Section 5 on page 48 through Section 12 on page 127 and to Figure 6 through Figure 9. A detected address parity error on any read transaction results in a target abort, refer to Figure 11. Address parity errors on write transactions are still posted to the PCI core interface, but are discarded.

For burst transactions to the register space, the application side of the PCI core interface operates faster than the PCI bus, so the PCI core interface FIFOs will never get full. TRDY remains asserted for all valid data phases applied.

4.1.1.3 Connection Memory Space Target Access

The T8110 connection memory is always immediately available for access (via dedicated access times assigned for PCI bus target transactions). Read and write bursting is allowed to this region. For more details, on connection memory programming, see Section 13.1 on page 132 and Figure 6 through Figure 9. A detected address parity error on any read transaction results in a target abort, refer to Figure 11. Address parity errors on write transactions are still posted to the PCI core interface, but are discarded.

For burst transactions to the connection memory space, the application side of the PCI core interface operates slightly slower than the PCI bus, so the PCI core interface FIFOs may get full. In this case, TRDY gets deasserted until the application side catches up.

4.1.1.4 Data Memory Space Target Access

The T8110 data memory is not guaranteed to be immediately available for access. Access to data memory is prioritized for standard H-bus/L-bus switching and packet payload switching, with PCI target access allowed as the lowest priority. Because there is an indeterminate amount of latency, target burst transfers are not allowed to the data memory. Upon reception of a PCI read or write request, if the data memory is immediately available, the transaction is completed as normal single-cycle access, refer to Figure 6 and Figure 8. If the data memory is not available at the time of the request, any write cycle is posted and any read cycle becomes a delayed read. A detected address parity error on any read transaction results in a target abort (refer to Figure 11 on page 32). Address parity errors on write transactions are still posted to the PCI core interface, but are discarded.

4.1.1.4.1 Posted Write Transaction

Only one posted write to the data memory may be queued at a time, refer to Figure 6 for more details. The user **must** monitor a status bit (register Status 8, bit 0, refer to Section 5.2.5) to determine whether a posted write is already queued before attempting more writes. Subsequent posted write attempts to the data memory while a queued posted write has not completed result in an ERROR condition, and both writes (the queued one and the subsequent one) are ignored. Error is reported at register Status 7, bit 0 (refer to Section 5.2.3 on page 62). Subsequent READ attempts from the data memory while a posted WRITE is queued result in a target RETRY (refer to Figure 10).

4.1.1.4.2 Delayed Read Transaction

Only one delayed read from the data memory may be queued at a time. A delayed read transaction latches the address and command information, and issues a retry back to the initiator (refer to Figure 10). If the initiator retries the same transaction **or** attempts a different read transaction from data memory prior to the queued delayed read completion, a retry is issued. The delayed read transaction is completed when the initiator retries the same transaction after the queued delayed read has finished (i.e., a normal completion of a single-cycle read, refer to

4 Microprocessor Interface (continued)

Figure 8). Any subsequent posted write attempts to the data memory while a delayed read is in progress result in an error condition, and the delayed read and the posted write attempts are ignored. Error is reported at register Status 7, bit 0 (refer to Section 5.2.3 on page 62).

4.1.1.5 Virtual Channel Memory Space Target Access

The T8110 virtual channel memory is not guaranteed to be immediately available for access. Access to this memory is prioritized for H-bus/L-bus switching and packet payload switching with PCI target access allowed as the lowest priority. Because there is an indeterminate amount of latency, target burst transfers are not allowed to the virtual channel memory. Upon reception of a PCI read or write request, if the virtual channel memory is immediately available, the transaction is completed as normal single-cycle access (refer to Figure 6 and Figure 8). If the virtual channel memory is not available at the time of the request, any write cycle is posted and any read cycle becomes a delayed read (for more detail on virtual channel memory programming, refer to Section 13.1.1.2 on page 134). A detected address parity error on any read transaction results in a target abort (refer to Figure 11). Address parity errors on write transactions are still posted to the PCI core interface, but are discarded.

4.1.1.5.1 Posted Write Transaction

Only one posted write to the virtual channel memory may be queued at a time. Refer to Figure 6. The user **must** monitor a status bit (register Status 8, bit 1, refer to Section 5.2.5) to determine whether a posted write is already queued before attempting more writes. Subsequent posted write attempts to the virtual channel memory while a queued posted write has not completed result in an error condition, and both writes (the queued one and the subsequent one) are ignored. Error is reported at register Status 7, bit 1 (refer to Section 5.2.3). Subsequent read attempts from the virtual channel memory while a posted WRITE is queued result in a target retry (refer to Figure 10).

4.1.1.5.2 Delayed Read Transaction

Only one delayed read from the virtual channel memory may be queued at a time. A delayed read transaction latches the address and command information, and issues a retry back to the initiator (refer to Figure 10). If the Initiator retries the same transaction **or** attempts a different read transaction from virtual channel memory prior to the queued delayed read completion, a retry is issued. The delayed read transaction is completed when the initiator retries the same transaction after the queued delayed read has finished (i.e., a normal completion of a single-cycle read, refer to Figure 8). Any subsequent posted write attempts to the virtual channel memory while a delayed read is in progress result in an error condition, and the delayed read and the posted write attempts are ignored. Error is reported at register Status 7, bit 1 (refer to Section 5.2.3 on page 62).

4.1.1.6 Minibridge Space Target Access

The T8110 minibridge port is not guaranteed to be immediately available for access. Access time to this space is dependent on wait-state control register setups. Because there is a potential variable amount of latency, target burst transfers are not allowed to the minibridge port. All write cycles are posted writes. All read cycles are delayed reads. Refer to Minibridge, starting on page 106 for more details on minibridge control and operation. A detected address parity error on any read transaction results in a target abort (refer to Figure 11). Address parity errors on write transactions are still posted to the PCI core interface, but are discarded.

4.1.1.6.1 Posted Write Transaction

Only one posted write to the minibridge port may be queued at a time. Refer to Figure 6. The user **must** monitor a status bit (register Status 8, bit 2) to determine whether a posted write is already queued before attempting more writes. Subsequent posted write attempts to the minibridge port while a queued posted write has not completed

4 Microprocessor Interface (continued)

result in an error condition. The queued write is allowed to complete, but the subsequent write is ignored. Error is reported at register Status 7, bit 2 (refer to Section 5.2.3 on page 62). Subsequent read attempts from the minibridge port while a posted write is queued result in a target retry (refer to Figure 10).

4.1.1.6.2 Delayed Read Transaction

Only one delayed read from the minibridge port may be queued at a time. A delayed read transaction latches the address and command information, and issues a retry back to the initiator (refer to Figure 10). If the initiator retries the same transaction **or** attempts a different read transaction from the minibridge port prior to the queued delayed read completion, a retry is issued. The delayed read transaction is completed when the initiator retries the same transaction after the queued delayed read has finished (i.e., a normal completion of a single-cycle read, refer to Figure 8). Any subsequent posted write attempts to the minibridge port while a delayed read is in progress result in an error condition. The delayed read is allowed to complete, but the write request is ignored. Error is reported at register Status 7, bit 2 (refer to Section 5.2.3 on page 62).

4.1.2 Initiator

The T8110 can initiate PCI transactions in order to perform packet payload switching between the local PCI bus and the 64 H-bus/L-bus data streams. The T8110 initiates accesses in order to either send (or push) data received from H-bus/L-bus streams to an external data buffer, or to retrieve (or pull) data from an external data buffer to transmit out to the H-bus/L-bus streams. Each operation requires three PCI burst accesses. An external descriptor table provides current read/write pointer status to the external data buffer. The T8110 fetches pointer information from the descriptor table, transfers data to/from the external data buffer, and then updates the descriptor table pointer information. For more details, see Section 13.2.3 on page 151.

4.1.2.1 PUSH Operation (Upstream Transaction)

The push operation takes data received from incoming H-bus/L-bus streams and passes it to an external data buffer. This is denoted as an upstream transaction. The three required T8110 initiated burst cycles are shown below. For more details, see Section 13.2.3 on page 151.

- Memory read burst (fetch the write pointer information from the descriptor table).
- Memory write burst (upload the received H-bus/L-bus data to external data buffer).
- Memory write burst (update the write pointer information in the descriptor table).

Figure 12. T8110 PCI Interface, Initiated PUSH Operation (TBD)

4.1.2.2 PULL Operation (Downstream Transaction)

The pull operation takes data from an external data buffer and passes it to the T8110 data memory, for transmission onto outgoing H-bus/L-bus streams. This is denoted as a downstream transaction. The three required T8110 initiated burst cycles are shown below. For more information, see Section 13.2.3 on page 151.

- Memory read burst (fetch the read pointer information from the descriptor table).
- Memory read burst (download the external data buffer to T8110 data memory for transmission on H-bus/L-bus streams).
- Memory write burst (update the read pointer information in the descriptor table).

Figure 13. T8110 PCI Interface, Initiated PULL Operation (TBD)

4 Microprocessor Interface (continued)

4.1.3 Configuration Space/EEPROM Interface

The T8110 PCI interface operates at 33 MHz and is a single-function device. As a target, T8110 is a memory-mapped device, and responds to memory write and memory read commands. Dual-address cycles are not supported (32-bit addressing only). As a master, T8110 generates memory write or memory read commands only, as single data phase burst cycles of two or more data phases.

Table 11. T8110 PCI Configuration Registers

Byte Address	Description			
0x00	Device ID		Vendor ID	
0x04	Status register		Command register	
0x08	Class code			Revision ID
0x0C	BIST	Header type	Latency timer	Reserved
0x10	Memory base address			
0x14	Reserved			
0x18	Reserved			
0x1C	Reserved			
0x20	Reserved			
0x24	Reserved			
0x28	Reserved			
0x2C	Subsystem ID		Subsystem vendor ID	
0x30	Reserved			
0x34	Reserved			
0x38	Reserved			
0x3C	MAX_LAT	MIN_GNT	INTERRUPT PIN	INTERRUPT LINE
0x40	Reserved		RETRY TIMEOUT	TRDY TIMEOUT
0x44—0xFF	Reserved			

Access to the configuration registers is shown in Figure 14 and in Figure 15. The configuration register contents include the following:

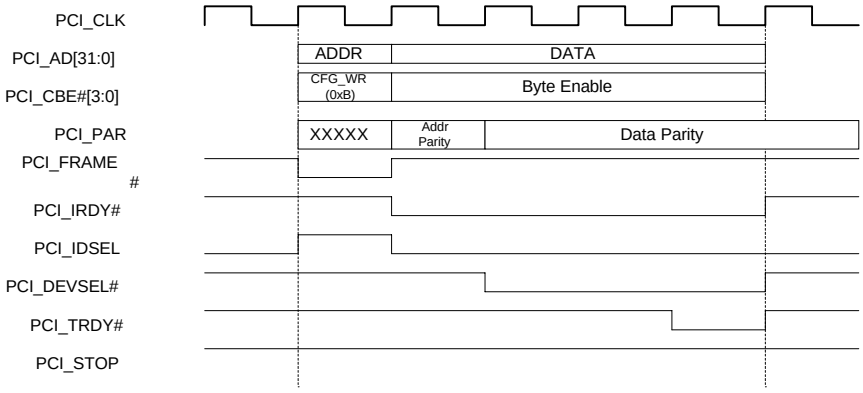
Device ID = 0x TBD, T8110
Vendor ID = 0x TBD, Lucent Technologies
Status Register:
[15]: Detected parity error
[14]: Signaled system error
[13]: Received master abort
[12]: Received target abort
[11]: Signaled target abort
[10:9] = 01, T8110 DEVSEL# timing is medium
[8]: Data parity reported
[7] = 1, T8110 is fast back-to-back capable
[6] = 0, T8110 does not support user-definable features
[5] = 0, T8110 is not 66 MHz capable
[4:0] = 00000, reserved

4 Microprocessor Interface (continued)

Command Register:
[15:10] = 000000, reserved
[9]: Fast back-to-back master enable
[8]: System error enable
[7] = 0, T8110 does not use stepping
[6]: Parity error enable
[5] = 0, T8110 disables palette snoop
[4]: Memory write and invalidate enable
[3] = 0, T8110 ignores special cycles
[2]: Bus master enable
[1]: Memory access enable
[0]: IO access enable
CLASS CODE = 0x TBD
REVISION ID = revision of the device
BIST = 0x00 (no BIST)
Header type = 0x00
Latency timer = *value*—T8110 as a master, number of cycles of retained bus ownership
Memory base address = 0xXXX00000, bits 31:20 are r/w as the static base address, which defines a 1 Mbyte region of addressable space.

Subsystem ID, subsystem vendor ID: user-definable: loaded from the EEPROM i/f at reset (refer to Section 4.1.3.1 on page 38).

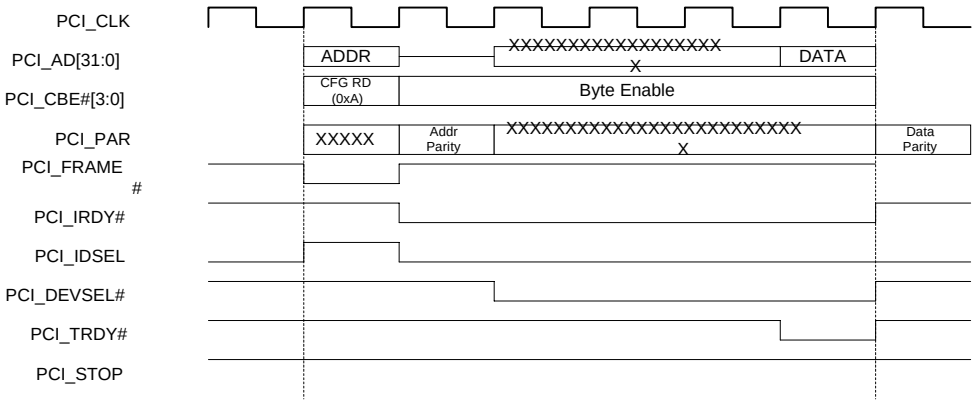
MAX_LAT = value - T8110 as a master, how often it requires access to the PCI bus
MIN_GNT = value - T8110 as a master, how long it retains PCI bus ownership
Interrupt pin = 0x01- T8110 uses INTA#
Interrupt line = value, user-defined
Retry timeout = value [default = 0x80]—T8110 as a master, the number of retries performed
TRDY timeout = value [default = 0x80]—T8110 as a master, how long it will wait for TRDY



NOTES:
1. T8110 PCI i/f has medium decode speed. There is always a two-cycle turn-around between assertion of frame# and assertion of DEVSEL#.
2. A configuration write access cycle takes five PCI clocks.

Figure 14. T8110 PCI Interface—Configuration WRITE Cycle

4 Microprocessor Interface (continued)



NOTES:
1. T8110 PCI i/f has medium decode speed. There is always a two-cycle turn-around between assertion of frame# and assertion of DEVSEL#.
2. A configuration read access cycle takes six PCI clocks.

Figure 15. T8110 PCI Interface—Configuration READ Cycle

4.1.3.1 Loadable PCI Configuration Space via EEPROM

The T8110 allows a user-definable subsystem ID and subsystem vendor ID field (configuration space address 0x2C). Immediately after power-on reset or PCIRST#, the T8110's PCI core loads the read-only configuration registers sequentially from the first 64 bytes in the EEPROM. All values are ignored, except for the subsystem ID and subsystem vendor ID (bytes 44—47). Ignored values (bytes 0—43, 48—63) are don't care, and exist simply as placeholders. During the EEPROM operation, All PCI target accesses to the T8110 result in a target retry.

Four pins are required for the EEPROM interface. The following pins are used for EEPROM just at power-on:

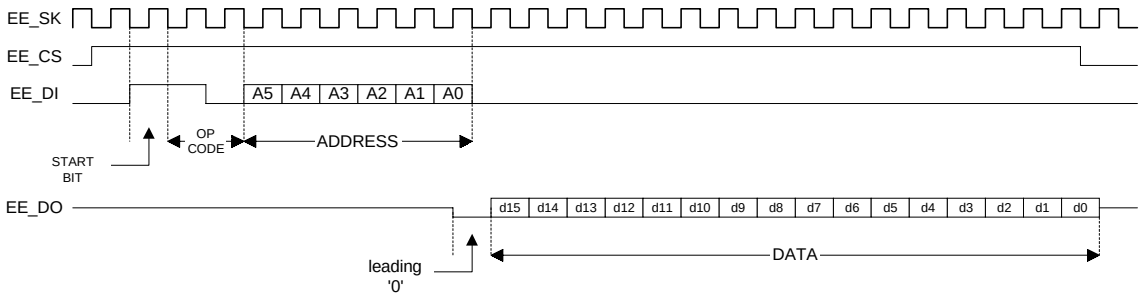
MB_A[1] = EE_DO_in (INPUT, data output from EEPROM)
MB_A[2] = EE_DI_out (OUTPUT, data input to EEPROM)
MB_A[3] = EE_SK_out (OUTPUT, clock input to EEPROM)
EE_CS_OUT = EE_CS_out (OUTPUT, chip select input to EEPROM)

The interface protocol follows the standard 93C46 EEPROM (refer to Figure 16). A state machine within the T8110's PCI core produces nine read cycles, one for each of the read-only configuration register fields. Only the subsystem ID and subsystem vendor ID words are configurable. All other fields returned by the EEPROM are ignored, but must be present as placeholders.

4 Microprocessor Interface (continued)

Table 12. PCI Configuration Space, EEPROM Map

EEPROM Address[5:0]	EEPROM Data[15:0]
000000	Device ID field
000010	Vendor ID field
001000	Class code field (upper 2 bytes)
001010	Class code field (lower byte) and revision ID field
001100	BIST and header fields
101100	Subsystem ID field
101110	Subsystem vendor ID field
111100	MAX_LAT and MIN_GNT fields
111110	Interrupt pin field



NOTES:
1. Signals output from T8110 are driven relative to falling edge of EE_SK, and sampled by the EEPROM on the rising edge.
2. Signals output from the EEPROM are driven relative to rising edge of EE_SK, and sampled by the T8110 on the falling edge.
3. Each read cycle takes 26 EE_SK clocks.
4. There are nine read cycles in all generated by the T8110's PCI core.
5. The following T8110 pinout is used for the EEPROM interface signals:

SIGNAL NAME	EEPROM FUNCTION	BALL ASSIGNMENT
EE_CS_OUT	EE_CS - chip select	G4
MB_A3	EE_SK - clock	G3
MB_A2	EE_DI - data in	G2
MB_A1	EE_DO - data out	G1

Figure 16. EEPROM Interface Protocol

4.2 ISA Interface

The T8110 ISA bus interface allows access to the T8110 internal regions via the minibridge port, see Table 8 on page 14 for pin descriptions. There are two user-selectable input signals that set up the ISA interface, MB_CS7 (*Intel/Motorola* protocol select) and MB_CS5 (word/byte address select).

4 Microprocessor Interface (continued)

4.2.1 *Intel/Motorola* Protocol Selector

MB_CS7 = 1 is the default if left unconnected, and selects an *Intel* handshake protocol.
MB_CS7 = 0 selects a *Motorola* handshake protocol.

Note: The MB_CS7 signal must be static (either pulled high or pulled low).

Table 13. *Intel/Motorola* Protocol Selector

Intel/Motorola Protocol Selector		
Signal	**Intel Mnemonic**	**Motorola Mnemonic**
MB_D[15:0]	D[15:0]	D[15:0]
MB_A[15:0]	A[15:0]	A[15:0]
MB_CS0	A[16]	A[16]
MB_CS1	A[17]	A[17]
MB_CS2	A[18]	A[18]
MB_CS3	A[19]	A[19]
MB_CS4	CSn	CSn
MB_CS6	RDY	DTACKn
MB_RD	RDn (read strobe)	DSn (data strobe)
MB_WR	WRn (write strobe)	R/Wn (read/write selector)
MB_CS5	Default	Default
MB_CS7	Default	Default

4.2.2 Word/Byte Addressing Selector

MB_CS5 = 1 is the default if left unconnected, and selects 16-bit word aligned addressing.
MB_CS5 = 0 selects 8-bit byte aligned addressing.

Note: The MB_CS5 signal may be static or dynamic in nature. If dynamic, MB_CS5 must follow the same timing requirements as the address bus.

Word-aligned addressing produces 16-bit data transfers via MB_D[15:0]. Byte-aligned addressing produces 8-bit data transfers via MB_D[7:0], (MB_D[15:8] is unused). The T8110 internal data bus is 32 bits, so MB_A[1:0] address bits are decoded along with MB_CS5 to control a dword-to-word or dword-to-byte swap function back to the MB_D bus.

4.2.3 Access Via the ISA Bus

The T8110 ISA bus interface allows access to three internal regions: registers, connection memory, and data memory. The virtual channel memory can be made accessible via a special diagnostic mode setting. All ISA bus asynchronous strobes are synchronized to the T8110's internal 65.536 MHz clock domain. There are 20 address bits provided to address the internal regions and these are defined in Table 14.

4 Microprocessor Interface (continued)

Table 14. T8110 Memory Mapping to ISA Space

Region	Subregion	Range (hex)
Registers	Reserved	0x00000—0x000FF
	Operating Control and Status	0x00100—0x001FF
	Clocks	0x00200—0x002FF
	Rate Control	0x00300—0x003FF
	Frame Group	0x00400—0x004FF
	General-Purpose I/O	0x00500—0x005FF
	Interrupt Control	0x00600—0x006FF
	Reserved	0x00700—0x007FF
	Reserved	0x00800—0x0FFFF
Virtual Channel Memory (Diagnostic Only)	—	0x10000—0x1FFFF
Data Memory	—	0x20000—0x2FFFF
Reserved	—	0x30000—0x3FFFF
Connection Memory	—	0x40000—0x4FFFF
Reserved	—	0x50000—0xFFFFF

4.2.3.1 ISA Interface Register Map

The T8110 registers map into the ISA bus space as follows.

Table 15. ISA Interface Register Map

DWORD Address (20 bits)	Section Cross Reference	Register			
		Byte 3	Byte 2	Byte 1	Byte 0
0x00100	5.1.1, 5.1.2	Master enable	Reserved	Reset select	Soft reset
0x00104	5.1.3, 5.1.4	Phase alignment select	Clock register access select	Data memory mode select	Reserved
0x00108	5.1.4	Fallback trigger, upper	Fallback trigger, lower	Fallback type select	Fallback control
0x0010C	5.1.4	Watchdog EN, upper	Watchdog EN, lower	Watchdog select, Netref	Watchdog select, C8
0x00114	4.1.5	Failsafe threshold high	Failsafe threshold low	Failsafe enable and status	Failsafe control
0x00120	5.2.1	Status 3, latched clock errors, upper	Status 2, latched clock errors, lower	Status 1, transient clock errors, upper	Status 0, transient clock errors, lower
0x00124	5.2.2, 5.2.3	Status 7, system errors, upper	Status 6, system errors, lower	Status 5	Status 4
0x00128	5.2.4	Device ID, upper	Device ID, lower	Reserved	Version ID
0x00140	12.1	Diag3	Diag2	Diag1	Diag0

4 Microprocessor Interface (continued)

Table 15. ISA Interface Register Map (continued)

DWORD Address (20 bits)	Section Cross Reference	Register			
		Byte 3	Byte 2	Byte 1	Byte 0
0x00144	12.1	Diag7	Diag6	Diag5	Diag4
0x00148	12.1	Reserved	Reserved	Reserved	Diag8
0x00200	6.1	PLL#1 rate	PLL#1 input selector	Main divider	Main input selector
0x00204	6.1	PLL#2 rate	Reserved	Resource divider	Main inversion select
0x00208	6.1	DPLL1 rate	DPLL1 input selector	Reserved	LREF input select
0x0020C	6.1	DPLL2 rate	DPLL2 input selector	Reserved	LREF inversion select
0x00210	6.1	Reserved	Netref1 LREF select	Netref1 divider	Netref1 input selector
0x00214	6.1	Reserved	Netref2 LREF select	Netref2 divider	Netref2 input selector
0x00220	6.2	C8 output rate	FR_COMP width	Netref output enables	Master output enables
0x00224	6.2	SCLK output rate	TCLK select	Reserved	CCLK output enables
0x00228	6.2	L_SC3 select	L_SC2 select	L_SC1 select	L_SC0 select
0x00300	9.1	H-bus rate H/G	H-bus rate F/E	H-bus rate D/C	H-bus rate B/A
0x00320	9.2	L-bus rate H/G	L-bus rate F/E	L-bus rate D/C	L-bus rate B/A
0x00400	7.1	FG0 rate	FG0 width	FG0 upper start	FG0 lower start
0x00410	7.1	FG1 rate	FG1 width	FG1 upper start	FG1 lower start
0x00420	7.1	FG2 rate	FG2 width	FG2 upper start	FG2 lower start
0x00430	7.1	FG3 rate	FG3 width	FG3 upper start	FG3 lower start
0x00440	7.1	FG4 rate	FG4 width	FG4 upper start	FG4 lower start
0x00450	7.1	FG5 rate	FG5 width	FG5 upper start	FG5 lower start
0x00460	7.1	FG6 rate	FG6 width	FG6 upper start	FG6 lower start
0x00470	7.1	FG7 rate	FG7 width	FG7 upper start	FG7 lower start
0x00474	7.2	FG7 mode upper	FG7 mode lower	FG7 counter high byte	FG7 counter low byte
0x00480	7.3	Reserved	FGIO R/W	FGIO read mask	FGIO data register
0x00500	8.1	GPIO override	GPIO R/W	GPIO read mask	GPIO data register
0x00600	13.1	FGIO interrupt polarity	Reserved	FGIO interrupt enable	FGIO interrupt pending

4 Microprocessor Interface (continued)

Table 15. ISA Interface Register Map (continued)

DWORD Address (20 bits)	Section Cross Reference	Register			
		Byte 3	Byte 2	Byte 1	Byte 0
0x00604	11.1	GPIO interrupt polarity	Reserved	GPIO interrupt enable	GPIO interrupt pending
0x00608	11.1	System interrupt enable, upper	System interrupt enable, lower	System interrupt pending, upper	System interrupt pending, lower
0x0060C	11.1	Clock interrupt enable, upper	Clock interrupt enable, lower	Clock interrupt pending, upper	Clock interrupt pending, lower
0x00610	11.1	CLKERR output select	SYSERR output select	PCI_INTA output select	Arbitration control
0x00614	11.1	CLKERR pulse width	SYSERR pulse width	Reserved	Reserved
0x00618	11.1	In-service, byte 3	In-service, byte 2	In-service, byte 1	In-service, byte 0

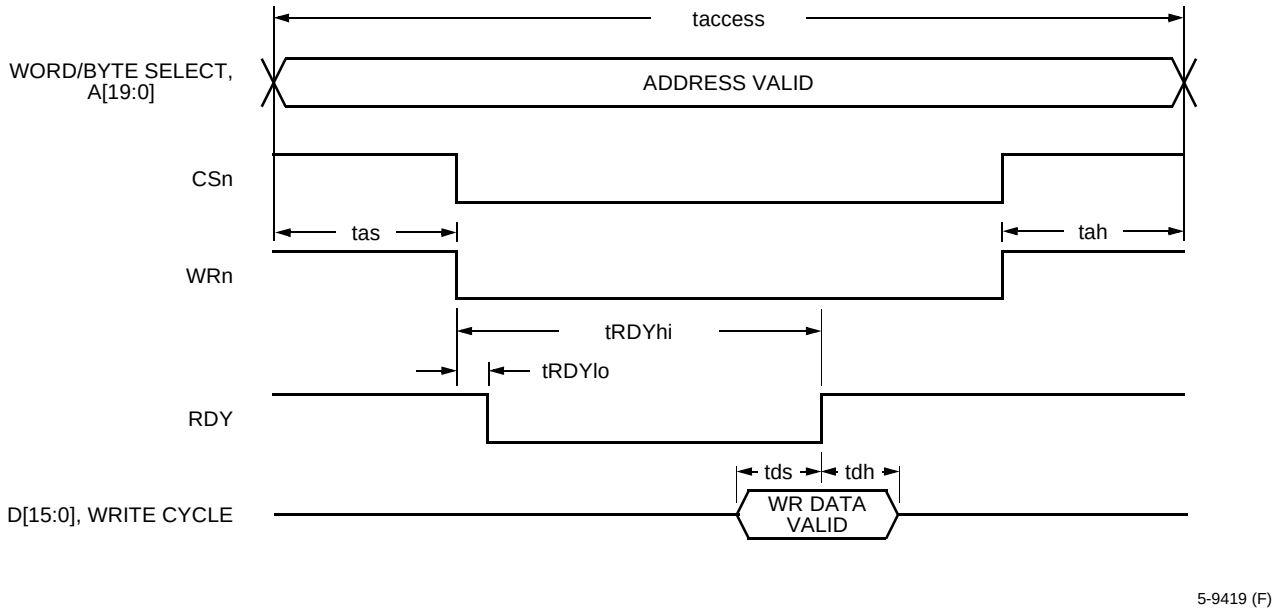


Figure 17. ISA Access Write Cycle, Intel Protocol

4 Microprocessor Interface (continued)

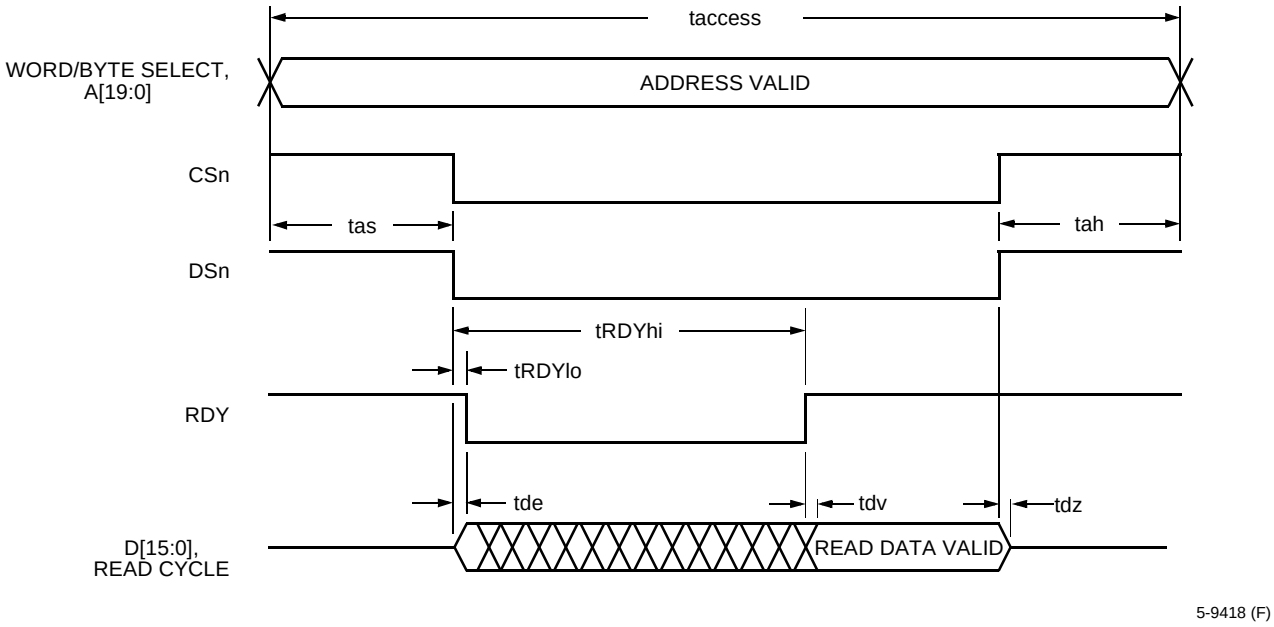


Figure 18. T8110 ISA Access Timing, Intel Protocol

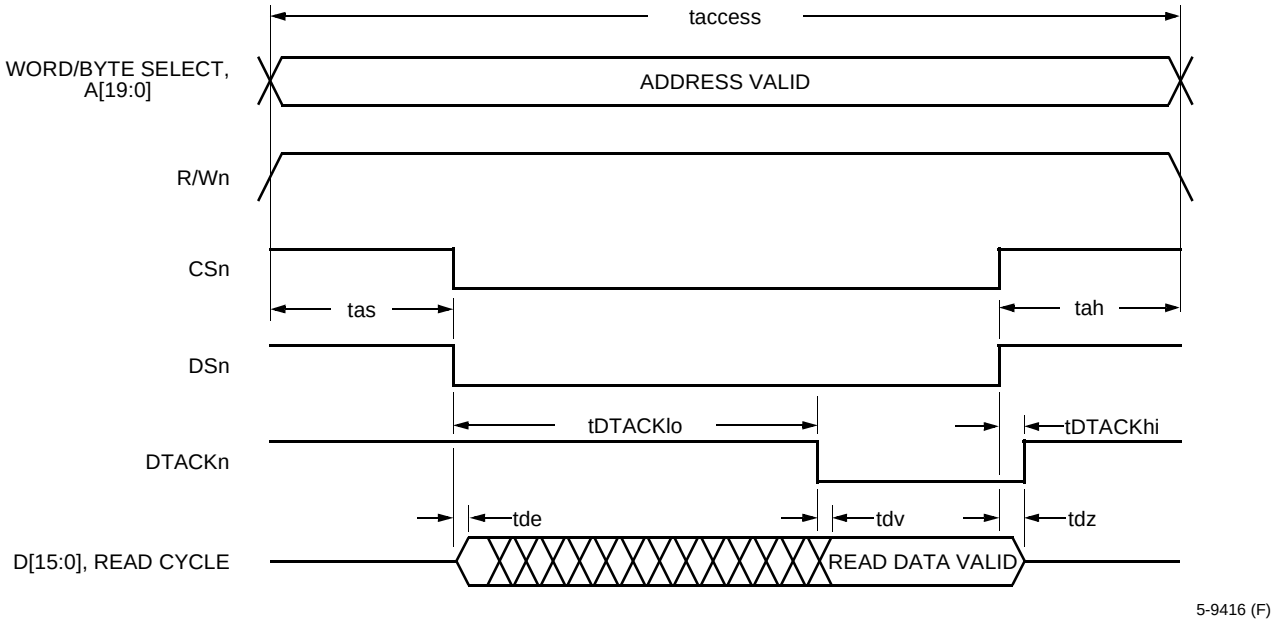


Figure 19. ISA Access Read Cycle, Motorola Protocol

4 Microprocessor Interface (continued)

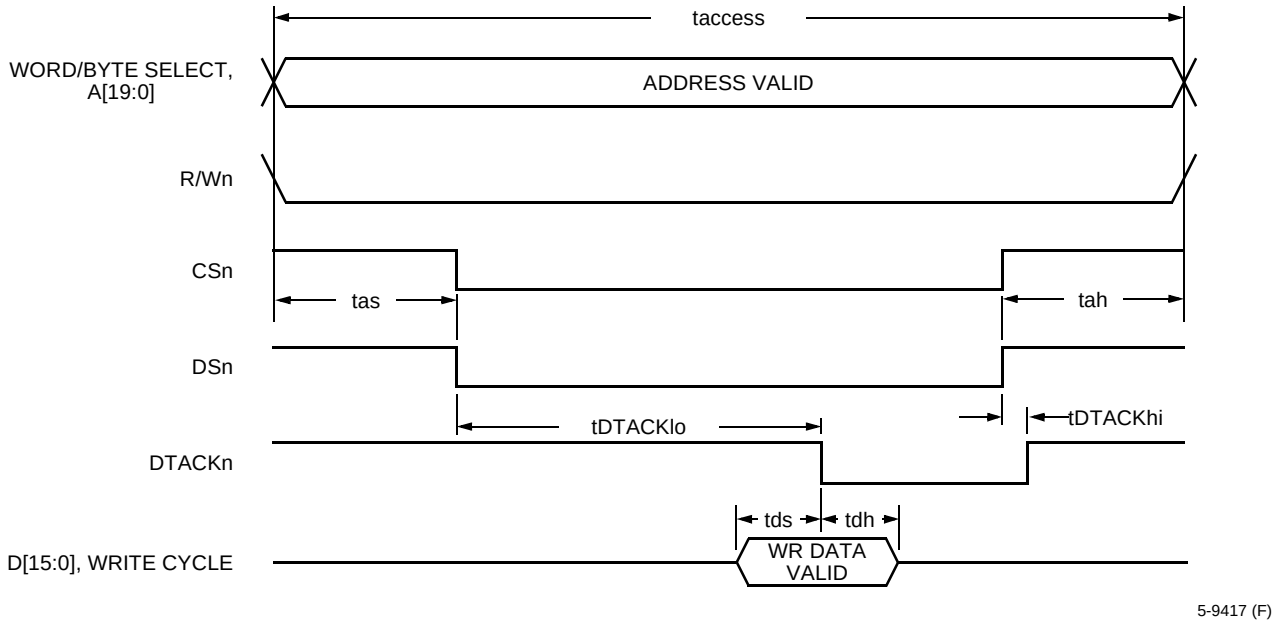


Figure 20. T8110 ISA Access Timing—*Motorola* Protocol

4.2.3.2 Register Space Access

The T8110 registers are always immediately available for access, providing low latency time to acknowledge the transaction. Read access to [reserved] addresses returns 0x00. Register access timing for Figure 17 through Figure 20 is shown below.

Table 16. Register Space Access Timing

Name	Parameter	Min (ns)	Max (ns)
taccess	Overall Access Time	TBD	TBD
tas	Address Setup Time	TBD	TBD
tah	Address Hold Time	TBD	TBD
tRDYlo	*Intel* cycle, Time to RDY Deasserted	TBD	TBD
tRDYhi	*Intel* cycle, Time to RDY Reasserted	TBD	TBD
tDTACKlo	*Motorola* cycle, Time to DTACKn Asserted	TBD	TBD
tDTACKhi	*Motorola* cycle, Time to DTACKn Deasserted	TBD	TBD
tde	Read cycle, Time to Data Enabled	TBD	TBD
tdv	Read cycle, Time to Data Valid	TBD	TBD
tdz	Read cycle, Time to Data Invalid	TBD	TBD
tds	Write cycle, Data Setup Time	TBD	TBD
tdh	Write cycle, Data Hold Time	TBD	TBD

4 Microprocessor Interface (continued)

4.2.3.3 Connection Memory Space Access

The T8110 connection memory is always immediately available for access (via dedicated access times assigned for ISA transactions) providing low latency time to acknowledge the transaction. Connection memory access timing for Figure 17 through Figure 20 is shown below:

Table 17. Connection Memory Space Access Timing

Name	Parameter	Min (ns)	Max (ns)
taccess	Overall Access Time	TBD	TBD
tas	Address Setup Time	TBD	TBD
tah	Address Hold Time	TBD	TBD
tRDYlo	*Intel* cycle, Time to RDY Deasserted	TBD	TBD
tRDYhi	*Intel* cycle, Time to RDY Reasserted	TBD	TBD
tDTACKlo	*Motorola* Cycle, Time to DTACKn Asserted	TBD	TBD
tDTACKhi	*Motorola* Cycle, Time to DTACKn Deasserted	TBD	TBD
tde	Read Cycle, Time to Data Enabled	TBD	TBD
tdv	Read Cycle, Time to Data Valid	TBD	TBD
tdz	Read Cycle, Time to Data Invalid	TBD	TBD
tds	Write Cycle, Data Setup Time	TBD	TBD
tdh	Write Cycle, Data Hold Time	TBD	TBD

4.2.3.4 Data Memory Space Access

The T8110 data memory is not guaranteed to be immediately available for access. Access to data memory is prioritized for standard H-bus/L-bus switching, with ISA bus transaction access allowed as the lowest priority. The latency time to acknowledge these transactions is indeterminate, and depends on the H-bus/L-bus switching configuration.

Table 18. Data Memory Space Access Timing

Name	Parameter	Min (ns)	Max (ns)
taccess	Overall Access Time	TBD	TBD
tas	Address Setup Time	TBD	TBD
tah	Address Hold Time	TBD	TBD
tRDYlo	*Intel* Cycle, Time to RDY Deasserted	TBD	TBD
tRDYhi	*Intel* Cycle, time to RDY reasserted	TBD	TBD
tDTACKlo	*Motorola* Cycle, Time to DTACKn Asserted	TBD	TBD
tDTACKhi	*Motorola* Cycle, Time to DTACKn Deasserted	TBD	TBD
tde	Read Cycle, Time to Data Enabled	TBD	TBD
tdv	Read Cycle, Time to Data Valid	TBD	TBD
tdz	Read Cycle, Time to Data Invalid	TBD	TBD
tds	Write Cycle, Data Setup Time	TBD	TBD
tdh	Write Cycle, Data Hold Time	TBD	TBD

4 Microprocessor Interface (continued)

4.2.3.5 Virtual Channel Memory Space Access

ISA access to the virtual channel memory is provided for diagnostic purpose only and is disabled by default. Access to this region is enabled via the diagnostic registers, see Section 12 on page 127. The T8110 virtual channel memory is not guaranteed to be immediately available for access. Access to data memory is prioritized for standard H-bus/L-bus switching, with ISA bus transaction access allowed as the lowest priority. The latency time to acknowledge these transactions is indeterminate, and depends on the H-bus/L-bus switching configuration. Virtual channel memory access timing for Figure 17 through Figure 20 is shown below.

Table 19. Virtual Channel Memory Space Access Timing

Name	Parameter	Min (ns)	Max (ns)
taccess	Overall Access Time	TBD	TBD
tas	Address Setup Time	TBD	TBD
tah	Address Hold Time	TBD	TBD
tRDYlo	*Intel* Cycle, Time to RDY Deasserted	TBD	TBD
tRDYhi	*Intel* Cycle, Time to RDY Reasserted	TBD	TBD
tDTACKlo	*Motorola* Cycle, Time to DTACKn Asserted	TBD	TBD
tDTACKhi	*Motorola* Cycle, Time to DTACKn Deasserted	TBD	TBD
tde	Read Cycle, Time to Data Enabled	TBD	TBD
tdv	Read cYcle, Time to Data Valid	TBD	TBD
tdz	Read cycle, time to Data Invalid	TBD	TBD
tds	Write Cycle, Data Setup Time	TBD	TBD
tdh	Write Cycle, Data Hold Time	TBD	TBD

5 Operating Control and Status

Overall T8110 operational control and status is configured via registers occupying 0x00100—0x001FC in the address space.

5.1 Control Registers

General control functions are soft reset, reset configuration, overall master output enables, and data memory configuration. Clocking-specific general control functions are clock register access configuration, phase alignment, clock fallback, and clock watchdog configuration.

Table 20. Control Register Map

DWORD Address (20 bits)	Register			
	Byte 3	**Byte 2**	**Byte 1**	**Byte 0**
0x00100	Master enable	Reserved	Reset select	Soft reset
0x00104	Phase alignment select	Clock register access select	Data memory mode select	VCstart*
0x00108	Fallback trigger, upper	Fallback trigger, lower	Fallback type select	Fallback control
0x0010C	Watchdog EN, upper	Watchdog EN, lower	Watchdog select, Netref	Watchdog select, C8
0x00110	External buffers descriptor table—base address register [31:0]*			
0x00114	Failsafe threshold high	Failsafe threshold low	Failsafe enable and status	Failsafe control

* VCstart and external buffers descriptor table registers are only relevant if the T8110 interfaces to the PCI bus. If the selected T8110 interface is to the ISA bus, this register is [reserved].

5.1.1 Reset Registers

The soft reset and reset select registers control soft reset functions and reset signal masking. Writes to the soft reset register trigger the corresponding action, and the set bit(s) are automatically cleared.

Power-on reset, nonmaskable - At power-on, initialize all T8110 registers (including reset select register) and connection valid flags, initialize the T8110 PCI interface. The power-on reset cell test input is controlled via diagnostic register—see Section 12.

HARD reset, maskable via reset select register, HRBEB. On assertion of RESET#, initialize all T8110 registers (excluding reset select register), and connection valid flags.

PCI reset, nonmaskable to PCI interface. Maskable to T8110 back-end via reset select register, PRBEB:
On assertion of PCI_RST#, initialize the T8110 PCI interface. If unmasked (PRBEB = 1), also initialize all T8110 registers (excluding reset select register) and connection valid flags.

Maskable to minibridge port via reset select register, PMBEB, the PCI_RST# input to the T8110 can be forwarded to the minibridge port, using the GP(2) output (via register 0x00503, see Section 8.1 on page 98). Polarity of the forwarded reset is selectable (via register 0x00781, see Section 10.2 on page 109).

SOFT resets: maskable via reset select register, SRBEB, selectable via soft reset register, SRESR.

- SOFT RESET 1: Initialize all T8110 registers (excluding reset select register) and connection valid flags.
- SOFT RESET 2: Initialize all T8110 registers (excluding reset select register).
- SOFT RESET 3: Reset all interrupt pending registers and the interrupt in-service register.
- SOFT RESET 4: Reset the interrupt in-service register only.
- RESET_PENDING_MEM: Reset the virtual channel Notify_pending memory.
- RESET_QUEUE: Reset the virtual channel Notify_queue FIFO.

5 Operating Control and Status (continued)

Table 21. Reset Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00100	Soft Reset	7:0	SRESR	0000 0000 0000 0001 0000 0010 0001 0000 0010 0000 0100 0000 1000 0000	NOP (default value) Reset all registers and connection valid flags Reset all registers Reset interrupt pending and in-service Registers Reset interrupt in-service register only Reset virtual channel NOTIFY_PENDING memory Reset virtual channel NOTIFY_QUEUE
0x00101	Reset Select	7:4	Reserved	0000 0	NOP (default)
		3	PMBEB	0 1	Disable PCI reset to minibridge (default) Enable PCI reset to minibridge
		2	PRBEB	0 1	Disable PCI reset to back end (default) Enable PCI reset to back end
		1	HRBEB	0 1	Disable hard reset to back end Enable hard reset to back end (default)
		0	SRBEB	0 1	Disable soft resets to back end Enable soft resets to back end (default)

5 Operating Control and Status (continued)

5.1.2 Master Output Enable Register

The master output enable register is used to control master output enables to various groups of T8110 signals, including the following:

L-bus data streams (L_D[31:0])
L-bus clocks (L_SC[3:0], FG[7:0] when used as frame group outputs)
H-bus data streams (CT_D[31:0])
H-bus clocks (CT_C8_A, /CT_FRAME_A, CT_C8_B, /CT_FRAME_B, CT_NETREF1,CT_NETREF2, /C16+, /C16–, /C4, C2, SCLK, SCLKx2#, /FR_COMP)
GPIO (GP[7:0])
FGIO (FG[7:0] when used as programmable register outputs)
Minibridge (MB_A[15:0], MB_CS[7:0], MB_RD, MB_WR, MB_D[15:0])

T8110 outputs that are not programmatically enabled (i.e., always driven except during reset) include the following: CLKERR, SYSERR, PRI_REF_OUT, NR1_SEL_OUT, NR2_SEL_OUT.

Table 22. Master Output Enable Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00103	Master Enable	7	AIOEB	0 1	Individual enables via bits [6:0] (default) Enable ALL (same as bits [6:0] = 1111111)
		6	MBREB	0 1	Disable minibridge* (default) Enable minibridge
		5	FGREB	0 1	Disable FGIO (default) Enable FGIO
		4	GPIEB	0 1	Disable GPIO (default) Enable GPIO
		3	HCKEB	0 1	Disable H-bus clocks (default) Enable H-bus clocks
		2	HDBEB	0 1	Disable H-bus data streams (default) Enable H-bus data streams
		1	LCKEB	0 1	Enable L-bus clocks – L_SC, FG (default) Disable L-bus clocks
		0	LDBEB	0 1	Disable L-bus data streams (default) Enable L-bus data streams

* **MBREB** is only relevant if the T8110 interfaces to the PCI bus. If the selected T8110 interface is to the ISA bus, this bit is reserved.

5.1.3 Connection Control—Virtual Channel Enable and Data Memory Selector Register

The VCstart register is used as an overall enable/disable for virtual channel switching activity, with the option to synchronize the enabling of switching activity with the internal 8 kHz frame reference (refer to Section 13.2.3 for more detail). Writes to the VCstart register trigger the corresponding action, and the set bit(s) are automatically cleared.

5 Operating Control and Status (continued)

Table 23. Virtual Channel Enable and Data Memory Selector Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00104	VCstart	7:0	VCSSR	0000 0000 0000 0001 0000 0010 0000 0001 0001 0010	NOP (DEFAULT) START (enable) VC switching, immediate PAUSE (disable) VC switching, immediate START VC switching, synchronized to frame PAUSE VC switching, synchronized to frame

The data memory mode select register MSbit controls subrate switching enable. The lower 7 bits control the T8110 data memory switching configuration. For more details, see Section 13.2.1.2 on page 144.

There are six* data memory configurations.

- 4k Single Buffered Switch. Standard H-bus/L-bus switching only, up to 4096 simplex connections, all connections are minimum delay due to single buffer configuration.
- 2k Double Buffered Switch. Standard H-bus/L-bus switching only, up to 2048 simplex connections, all connections are programmable for minimum or constant delay via the double buffer configuration.
- 2k Single Buffered Switch + 1k double buffered switch. Standard H-bus/L-bus switching only, up to 2048 simplex minimum delay connections (single buffer) and up to 1024 simplex minimum or constant delay connections (double buffer).
- 2k Single Buffered Switch + 256 virtual channels. Standard H-bus/L-bus switching, up to 2048 simplex minimum delay connections (single buffer), PLUS packet payload switching, up to 256 virtual channels.
- 1k Double Buffered Switch + 256 virtual channels. Standard H-bus/L-bus switching, up to 1024 simplex minimum or constant delay connections (double buffer), PLUS packet payload switching, up to 256 virtual channels.
- No standard switching + 512 virtual channels. Packet payload switching only, up to 512 virtual channels.

* If the T8110 interfaces to the PCI bus, all configurations are valid. If the interface selection is to the ISA bus, only the standard switching configurations 1—3 are allowed.

Table 24. Data Memory Mode Select Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00105	Data Mem Mode Select	7	GSREB	0 1	Disable subrate switching Enable subrate switching
		6:0	DMMSP	100 0000 010 0000 011 0000 010 0010 001 0010 000 0100	4k single-buffer switch 2k double-buffer switch 2k single-buffer, 1k double-buffer switch 2k single buffer switch, 256 virtual channels 1k double buffer switch, 256 virtual channels 512 virtual channels

5.1.4 General Clock Control (Phase Alignment, Fallback, Watchdogs) Register

The clock register access select register controls the selection between accessing the ACTIVE vs. the INACTIVE set of T8110 clock registers. The T8110 contains two sets of clock registers, **X** and **Y**. Only one set is used at a time. It is selected based on the clock fallback setup. The clock register set that is currently in use is denoted as the ACTIVE set—see Section 6.3 on page 76 for more details.

5 Operating Control and Status (continued)

Table 25. Clock Register Access Select Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00106	Clock Reg Access Select	7:0	CSASR	0000 0000 0000 0001	Access INACTIVE clock registers (default) Access ACTIVE clock registers

5.1.5 Phase Alignment Select Register

- The phase alignment select register selects the phase alignment configuration. For more details, see Section 6.4.5.1 on page 80. The T8110 internally generates an 8 kHz frame reference. Shown below are three configurations to control phase alignment between this internally generated frame reference and a selected incoming frame reference from the H-bus (/CT_FRAMEA, /CT_FRAMEB, or /FR_COMP) or local clock reference (LREF[4:7]).
- DISABLE alignment, no realignment of unaligned frames
- SNAP alignment, immediate realignment of unaligned frames
- SLIDE alignment, gradual realignment of unaligned frames

Table 26. Phase Alignment Select Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00107	Phase Alignment Select	7:0	PAFSR	0000 0000 0000 0001 0000 0010	Phase alignment is disabled (default) Enable SNAP alignment Enable SLIDE alignment

5.1.6 Fallback Control Register

The fallback control register allows user control over the ACTIVE and INACTIVE clock register sets. For more details, see Section 6.7.1 on page 82. Writes to the fallback control register trigger the corresponding action, and the set bit(s) are automatically cleared. The four commands are shown below:

- GO CLOCKS. At initialization, the clock register Y set is ACTIVE, the X set is inactive, and access is enabled to the X set. The GO CLOCKS command transitions the Y set to INACTIVE and the X set to ACTIVE. This command can either be performed immediately upon issue or can wait to be performed until the next 8 kHz Frame reference (synchronized to frame).
- CLEAR FALLBACK. Forces a state transition for ACTIVE/INACTIVE assignment of the clock register X and Y sets after a fallback event has occurred. This command can either be performed immediately upon issue or can wait to be performed until the next 8 kHz frame reference (synchronized to frame).
- FORCE FALLBACK. Forces a state transition for ACTIVE/INACTIVE assignment of the clock register X and Y sets by creating a fallback event. This command can either be performed immediately upon issue or can wait to be performed until the next 8 kHz frame reference (synchronized to frame).
- COPY ACTIVE to INACTIVE SET. Copies all register values in the current active clock register set to the inactive clock register set. This command is performed immediately upon issue.

5 Operating Control and Status (continued)

Table 27. Fallback Control Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00108	Fallback Control	7:0	FBCSR	0000 0000 0000 0001 0000 0010 0000 0100 0001 0001 0001 0010 0001 0100 0010 0000	NOP (default) Go clocks command Clear fallback command Force fallback command Go clocks synchronized to frame Clear fallback synchronized to frame Force fallback synchronized to frame Copy active to inactive set command

5.1.7 Fallback Type Select Register

The upper nibble configures which H-bus clocks are selected to trigger a clock fallback event. Any of the **legacy** modes have predetermined trigger enables and ignore the fallback trigger register settings. Nonlegacy modes require the fallback trigger register settings. For more details, see Section 6.7.1 on page 82.

The lower nibble configures the state machine that controls clock register set ACTIVE/INACTIVE assignments. There are three possible selections. For more details, see Section 6.7 on page 82.

- DISABLED. No transitions of clock register X and Y sets to ACTIVE/INACTIVE.
- FIXED SECONDARY. Swap the ACTIVE/INACTIVE sets on a fallback event, swap them back when fallback is cleared.
- ROTATING SECONDARY. Swap the ACTIVE/INACTIVE sets on a fallback event, maintain this state when fallback is cleared.

Table 28. Fallback Type Select Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00109	Fallback Type Select	7:4	FTRSN	0001 0010 0100 1000 1001	Legacy, fallback to OSC/4 on main select failure Legacy, fallback X/Y set on main select failure Legacy, fallback X/Y set on H-bus A/B failure Fallback trigger registers control fallback Fallback trigger registers control fallback H-Bus clock enable state machine is enabled
		3:0	FSMSN	0000 0001 0010	Fallback is disabled (default) Enable FIXED SECONDARY fallback Enable ROTATING SECONDARY fallback

5.1.8 Fallback Trigger Registers

The fallback trigger registers are used in conjunction with the fallback type select register and controls which H-bus clocks are enabled to trigger a clock fallback event in case of error. The sync reference inputs to DPLL#1 and DPLL#2 also can trigger a clock fallback event upon detection of an error.

5 Operating Control and Status (continued)

Table 29. Fallback Trigger Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0010A	Fallback Trigger, Lower	7	S2FEB	0 1	Disable SCLKx2# trigger (default) Enable SCLKx2# trigger
		6	SCFEB	0 1	Disable SCLK trigger (default) Enable SCLK trigger
		5	C2FEB	0 1	Disable C2 trigger (default) Enable C2 trigger
		4	C4FEB	0 1	Disable/C4 trigger (default) Enable/C4 trigger
		3	CMFEB	0 1	Disable/C16– trigger (default) Enable/C16– trigger
		2	CPFEB	0 1	Disable/C16+ trigger (default) Enable/C16+ trigger
		1	CBFEB	0 1	Disable CT_C8_B trigger (default) Enable CT_C8_B trigger
		0	CAFEB	0 1	Disable CT_C8_A trigger (default) Enable CT_C8_A trigger
0x0010B	Fallback Trigger, Upper	7	Reserved	0	NOP (default)
		6	D2FEB	0 1	Disable DPLL2 sync trigger (default) Enable DPLL2 sync trigger
		5	D1FEB	0 1	Disable DPLL1 sync trigger (default) Enable DPLL1 sync trigger
		4	N2FEB	0 1	Disable CT_NETREF2 trigger (default) Enable CT_NETREF2 trigger
		3	N1FEB	0 1	Disable CT_NETREF1 trigger (default) Enable CT_NETREF1 trigger
		2	FCFEB	0 1	Disable/FR_COMP trigger (default) Enable/FR_COMP trigger
		1	FBFEB	0 1	Disable/CT_FRAME_B trigger (default) Enable/CT_FRAME_B trigger
		0	FAFEB	0 1	Disable/CT_FRAME_A trigger (default) Enable/CT_FRAME_A trigger

5.1.9 Watchdog Select, C8 and Netref Registers

The watchdog select, c8 Register controls the watchdog circuits to monitor the proper frequency for the **CT_C8_A** and **CT_C8_B** signals. These signals can take on two values, including 8.192 MHz (ECTF mode) and 4.096 MHz (MC1 mode).

The watchdog select, Netref register controls the watchdog circuits to monitor the proper frequency for the CT_NETREF1 and CT_NETREF2 signals. These signals can take on three values depending on system-level clocking architecture, including 8 kHz (frame reference), 1.544 MHz (T1 bit clock), and 2.048 MHz (E1 bit clock).

5 Operating Control and Status (continued)

Table 30. Watchdog Select, C8, Netref Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0010C	Watchdog Select, C8	7:4	CBWSN	0000 0001	CT_C8_B watchdog at 8.192 MHz (default) CT_C8_B watchdog at 4.096 MHz MC1mode
		3:0	CAWSN	0000 0001	CT_C8_A watchdog at 8.192 MHz (default) CT_C8_A watchdog at 4.096 MHz MC1mode
0x0010D	Watchdog Select, Netref	7:4	N2WSN	0000 0001 0010	CT_NETREF2 watchdog at 8 kHz (default) CT_NETREF2 watchdog at 1.544 MHz CT_NETREF2 watchdog at 2.048 MHz
		3:0	N1WSN	0000 0001 0010	CT_NETREF1 watchdog at 8 kHz (default) CT_NETREF1 watchdog at 1.544 MHz CT_NETREF1 watchdog at 2.048 MHz

5.1.10 Watchdog EN Register

The watchdog EN registers are used to enable/disable watchdogs on the individual H-bus clocks and the watchdogs on the sync inputs of DPLL#1 and DPLL#2.

Table 31. Watchdog EN Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0010E	Watchdog EN, Lower	7	S2WEB	0 1	Disable SCLKx2# watchdog (default) Enable SCLKx2# watchdog
		6	SCWEB	0 1	Disable SCLK watchdog (default) Enable SCLK watchdog
		5	C2WEB	0 1	Disable C2 watchdog (default) Enable C2 watchdog
		4	C4WEB	0 1	Disable/C4 watchdog (default) Enable/C4 watchdog
		3	CMWEB	0 1	Disable/C16– watchdog (default) Enable/C16– watchdog
		2	CPWEB	0 1	Disable/C16+ watchdog (default) Enable/C16+ watchdog
		1	CBWEB	0 1	Disable CT_C8_B watchdog (default) Enable CT_C8_B watchdog
		0	CAWEB	0 1	Disable CT_C8_A watchdog (default) Enable CT_C8_A watchdog

5 Operating Control and Status (continued)

Table 31. Watchdog EN Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0010F	Watchdog EN, Upper	7	FSWEB	0 1	Disable FAILSAFE ref watchdog (DEFAULT) Enable FAILSAFE ref watchdog
		6	D2WEB	0 1	Disable DPLL2 sync watchdog (default) Enable DPLL2 sync watchdog
		5	D1WEB	0 1	Disable DPLL1 sync watchdog (default) Enable DPLL1 sync watchdog
		4	N2WEB	0 1	Disable CT_NETREF2 watchdog (default) Enable CT_NETREF2 watchdog
		3	N1WEB	0 1	Disable CT_NETREF1 watchdog (default) Enable CT_NETREF1 watchdog
		2	FCWEB	0 1	Disable/FR_COMP watchdog (default) Enable/FR_COMP watchdog
		1	FBWEB	0 1	Disable/CT_FRAME_B watchdog (default) Enable/CT_FRAME_B watchdog
		0	FAWEB	0 1	Disable/CT_FRAME_A watchdog (default) Enable/CT_FRAME_A watchdog

5.1.11 Failsafe Control Registers

The failsafe control register controls a return from the failsafe state. Writes to the failsafe control register trigger the corresponding action, and the set bit(s) are automatically cleared. From the failsafe state, the user can return to either the primary or secondary clock register sets, please see Section 6.7.2 on page 88.

The failsafe enable and status register controls the enable/disable of failsafe operation, and indicates the failsafe status. For more on failsafe operation, see Section 6.7.2 on page 88.

The failsafe threshold registers allow for programmable threshold times for tripping a failsafe condition. Resolution for the trip value increments is one 32.768 MHz clock period (30.5 ns). The register contains [count -1], a value of 0x0000 yields a 30.5 ns threshold. A value of 0xFFFF yields a 15.6 µs threshold. For more on failsafe operation, see Section 6.7.2 on page 88.

Table 32. Failsafe Control Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00114	Failsafe Control	7:0	FSCSR	0000 0000 0000 0001 0000 0010	Disable failsafe (DEFAULT) Enable failsafe, automatic return Enable failsafe, manual return
0x00115	Failsafe Enable and Status	7:4	FSSON	0000 0001	Failsafe inactive Failsafe state activated
		3:0	FSENN	0000 0001	Failsafe disabled Failsafe enabled

5 Operating Control and Status (continued)

Table 32. Failsafe Control Register (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00116	Failsafe Threshold Lo	7:0	FSTLR	LLLL LLLL	Failsafe threshold value, low byte
0x00117	Failsafe Threshold Hi	7:0	FSTLR	LLLL LLLL	Failsafe threshold value, high byte

5.1.12 External Buffers—Descriptor Table Base Address

Table 33. Extended Buffers Base Addresses

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00110	Base Address Byte0	7:0	NA	LLLL LLLL	32-bit base address value for external buffers descriptor table *
0x00111	Base Address Byte1	7:0	NA	LLLL LLLL	32-bit base address value for external buffers descriptor table *
0x00112	Base Address Byte2	7:0	NA	LLLL LLLL	32-bit base address value for external buffers descriptor table*
0x00113	Base Address Byte3	7:0	NA	LLLL LLLL	32-bit base address value for external buffers descriptor table*

* External buffers descriptor table base address is only relevant if the T8110 interfaces to the PCI bus. If the selected T8110 interface is to the ISA bus, this register is reserved. For more detai, refer to Section 13.2.3.4 on page 152.

5.2 Error and Status Registers

Status 7, 6, and 3—0 registers are writable by the user for clearing specific error bits. Writing a 1 to any of the bits of these registers will clear the corresponding error bit. The remaining error and status registers are read-only.

Table 34. Error and Status Register Map

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00120	Status 3, latched clock errors, upper	Status 2, latched clock errors, lower	Status 1, transient clock errors, upper	Status 0, transient clock errors, lower
0x00124	Status 7, system errors, upper	Status 6, system errors, lower	Status 5 PLL and switching status	Status 4 fallback and failsafe status
0x00128	Device ID, upper	Device ID, lower	Reserved	Version ID
0x0012C	Reserved	Reserved	Status 9, virtual channel status	Status 8, PCI target queue status

5 Operating Control and Status (continued)

5.2.1 Clock Errors

5.2.1.1 Transient Clock Errors Registers

The transient clock error registers are used in conjunction with the watchdog EN registers and indicate error status for H-bus clocks and DPLL sync inputs whose watchdogs are enabled. The transient indicators are dynamic in nature—that is, if a clock is in error only for a short time and then recovers, the error indication is deasserted when the clock recovers. Additionally, an APLL#1 Out-of-Lock indicator is provided, and used in conjunction with the fail-safe clocking mode. For more detail, please see Section 6.7.1 on page 82 and Section 6.7.2 on page 88.

Table 35. Clock Error Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00120	Status 0, Transient Clock Errors, Lower	7	S2TOB	0 1	SCLKx2# no error (default) SCLKx2# error
		6	SCTOB	0 1	SCLK no error (default) SCLK error
		5	C2TOB	0 1	C2 no error (default) C2 error
		4	C4TOB	0 1	/C4 no error (default) /C4 error
		3	CMTOB	0 1	/C16– no error (default) /C16– error
		2	CPTOB	0 1	/C16+ no error (default) /C16+ error
		1	CBTOB	0 1	CT_C8_B no error (default) CT_C8_B error
		0	CATOB	0 1	CT_C8_A no error (default) CT_C8_A error

5 Operating Control and Status (continued)

Table 35. Clock Error Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00121	Status 1, Transient Clock Errors, Upper	7	FSTOB	0 1	Failsafe indicator: APLL1 reference no error APLL1 reference error
		6	D2TOB	0 1	DPLL2 sync no error (default) DPLL2 sync error
		5	D1TOB	0 1	DPLL1 sync no error (default) DPLL1 sync error
		4	N2TOB	0 1	CT_NETREF2 no error (default) CT_NETREF2 error
		3	N1TOB	0 1	CT_NETREF1 no error (default) CT_NETREF1 error
		2	FCTOB	0 1	/FR_COMP no error (default) /FR_COMP error
		1	FBTOB	0 1	/CT_FRAME_B no error (default) /CT_FRAME_B error
		0	FATOB	0 1	/CT_FRAME_A no error (default) /CT_FRAME_A error

5 Operating Control and Status (continued)

5.2.1.2 Latched Clock Error Register

The latched clock error registers capture transient clock errors. The latched indicators capture and hold any transient error status and are used by the clock fallback logic. For more details, see Section 6.7 on page 82, and Section 11 on page 112 for more details.

Table 36. Latched Clock Error Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00122	Status 2, Latched Clock Errors, Lower	7	S2LOB	0 1	SCLKx2# no error (default) SCLKx2# error
		6	SCLOB	0 1	SCLK no error (default) SCLK error
		5	C2LOB	0 1	C2 no error (default) C2 error
		4	C4LOB	0 1	/C4 no error (default) /C4 error
		3	CMLOB	0 1	/C16– no error (default) /C16– error
		2	CPLOB	0 1	/C16+ no error (default) /C16+ error
		1	CBLOB	0 1	CT_C8_B no error (default) CT_C8_B error
		0	CALOB	0 1	CT_C8_A no error (default) CT_C8_A error
0x00123	Status 3, Latched Clock Errors, Upper	7	FSLOB	0 1	Failsafe indicator: APLL1 reference no error APLL1 reference error
		6	D2LOB	0 1	DPLL2 sync no error (default) DPLL2 sync error
		5	D1LOB	0 1	DPLL1 sync no error (default) DPLL1 sync error
		4	N2LOB	0 1	CT_NETREF2 no error (default) CT_NETREF2 error
		3	N1LOB	0 1	CT_NETREF1 no error (default) CT_NETREF1 error
		2	FCLOB	0 1	/FR_COMP no error (default) /FR_COMP error
		1	FBLOB	0 1	/CT_FRAME_B no error (default) /CT_FRAME_B error
		0	FALOB	0 1	/CT_FRAME_A no error (default) /CT_FRAME_A error

5 Operating Control and Status (continued)

5.2.2 System Status

Clock fallback status register: the upper nibble provides status indicators for clock fallback. FBFOB indicates whether the circuit is in a clock fallback state. FBSOP indicates which of five possible states the circuit is in, see Section 6.7.1 on page 82 for more details.

The lower nibble provides status indicators related to the X and Y clock register set ACTIVE/INACTIVE assignments. XYSOB indicates which of the clock register sets is active. The remaining bits indicate a pending status for go clocks, clear fallback and force fallback commands issued (via the fallback control register, 0x00108) which are waiting for a frame sync.

PLL and switching status register: upper 6 bits provide analog and digital PLL lock status. For more details, see Section 6 on page 65. Lower 2 bits provide memory status as follows: CMROB is an indicator for the connection memory actively resetting, see Section 13.2.1.1 on page 142. DMPOB indicates the active data page used for double-buffered standard switching, see Section 13.2.1.2 on page 144.

Table 37. PLL and Switching Status Registers

Byte Address	Register Name	Bit(s)	Mnemonic	Value	Function
0x00124	Status 4, Clock Fallback Status	7	FBFOB	0 1	Indicates not in fallback/failsafe state (default) Indicates fallback/failsafe state
		6:4	FBSOP	111 000 001 010 011 100 101	Fallback state = INITIAL (default) Fallback state = PRIMARY Fallback state = TO_PRIMARY Fallback state = SECONDARY Fallback state = TO_SECONDARY Failsafe state = FS_1 Failsafe state = FS_2
		3	XYSOB	0 1	Clock register Y set is active, X is inactive Clock register X set is active, Y is inactive
		2	GOPOB	0 1	No go clocks pending (default) Go clocks pending, waiting for frame
		1	CFPOB	0 1	No clear fallback pending (default) Clear fallback pending, waiting for frame
		0	FFPOB	0 1	No force fallback pending (default) Force fallback pending, waiting for frame

5 Operating Control and Status (continued)

Table 37. PLL and Switching Status Registers (continued)

Byte Address	Register Name	Bit(s)	Mnemonic	Value	Function
0x00125	Status 5, PLL and Switching Status	7	A1LOB	0 1	Analog PLL#1 in-lock (default) Analog PLL#1 out-of-lock
		6	OOL	0 1	Out-of-lock indicator INACTIVE (default) Out-of-lock indicator ACTIVE
		5:4	D1LOD	00 01 10 11	DPLL#1 in-lock (default) DPLL#1 out-of-lock, SLOW correction DPLL#1 out-of-lock, FAST correction Invalid
		3:2	D2LOD	00 01 10 11	DPLL#2 in-lock (default) DPLL#2 out-of-lock, SLOW correction DPLL#2 out-of-lock, FAST correction Invalid
		1	CMROB	0 1	Connection memory not resetting (default) Connection memory reset loop is active
		0	DMPOB	0 1	Active page = data memory page 1 Active page = date memory page 2

5.2.3 System Errors Register

Table 38. System Errors Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00126	Status 6 – System Errors, Lower	7	TBD	0 1	No error PCI master, PCI bus fatal error
		6	TBD	0 1	No error PCI master, external buffer LOCK error
		5	TBD	0 1	No error PCI master, external buffer STALL error
		4	TBD	0 1	No warning PCI master, external buffer STALL warning
		3	TBD	0 1	No warning PCI master, external buffer OVWRITE warning
		2	TBD	0 1	No warning PCI master, external buffer INIT warning
		1	TBD	0 1	No warning VC memory, scratchpad OVFLOW warning
		0	TBD	0 1	No warning NOTIFY queue, OVFLOW warning

5 Operating Control and Status (continued)

Table 38. System Errors Registers (continued)

Byte Address	Register Name	Bit(s)	Mnemonic	Value	Function
0x00127	Status 7, System Errors, Upper	7	TBD	0 1	No error Clock failsafe indicator
		6	TBD	0 1	No error Clock fallback indicator
		5	TBD	0 1	No time out PCI target, minibridge discard timer expired
		4	TBD	0 1	No time out PCI target, vc memory discard timer expired
		3	TBD	0 1	No time out PCI target, DATA memory discard timer expired
		2	TBD	0 1	No error PCI target, minibridgee protocol error
		1	TBD	0 1	No error PCI target, VC memory protocol error
		0	TBD	0 1	No error PCI target, DATA memory protocol error

5.2.4 Device Identification Registers

These registers identify the device type and revision status, T8110 revision n.

Table 39. Device Identification Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00128	Version ID	7:0	VEROR	0000 0001	REVISION status (value shown = REV1)
0x0012A	Device ID, Lower	7:0	IDLOR	0001 0000	Device ID low status 0x10
0x0012B	Device ID, Upper	7:0	IDHOR	1000 0001	Device ID high status 0x81

5.2.5 Miscellaneous Status

The status 8 register provides the status flags which indicate that there is currently a posted write or a delayed read enqueued, one flag for each access region minibridge, virtual channel memory, and data memory. Refer to Sections 4.1.1.4, 4.1.1.4.1, and 4.1.1.5.1.

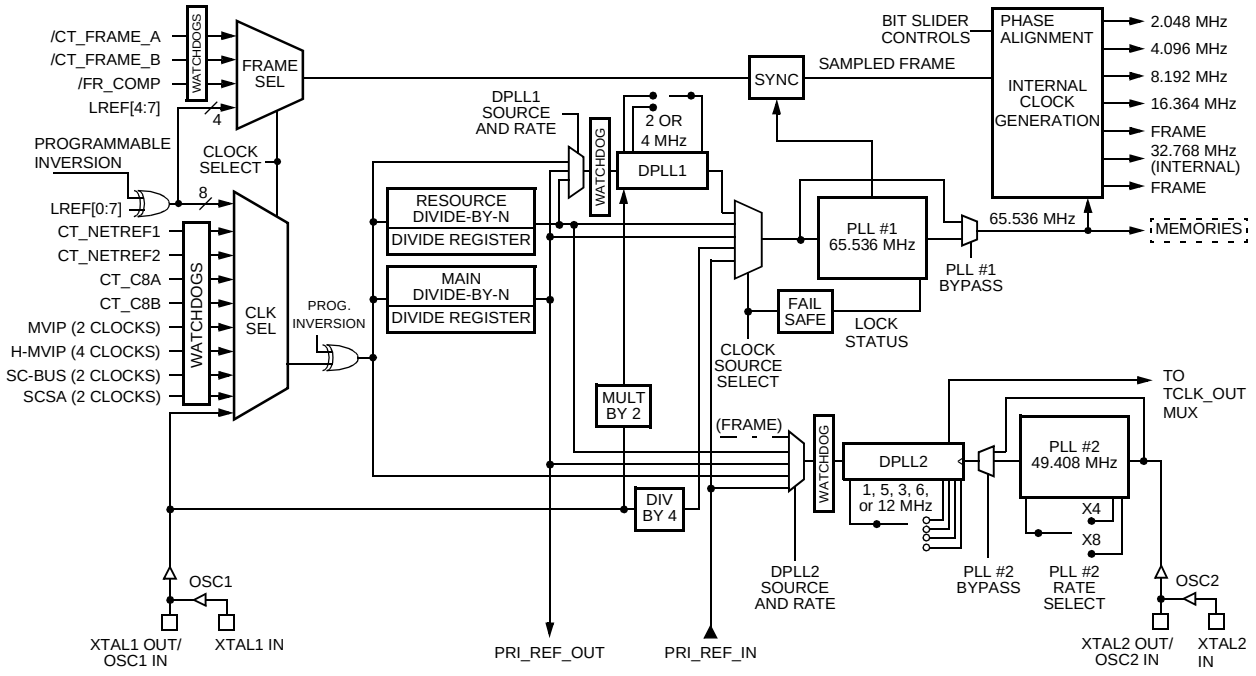
The status 9 register provides the current state of the virtual channel switching (START or PAUSE), and the status of any pending start/pause commands that are waiting for the next frame. Refer to Section 13.2.3.

5 Operating Control and Status (continued)

Table 40. Miscellaneous Status Registers

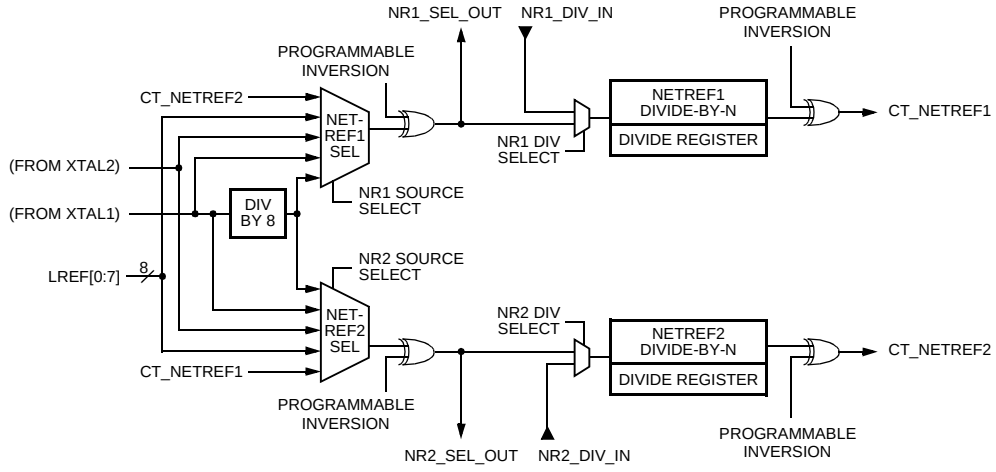
Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0012C	Status 8	7:3	Reserved	0 0000	NOP
		2	TBD	0 1	Minibridge PCI target queue is empty Minibridge PCI target queue is full
		1	TBD	0 1	VC memory PCI target queue is empty VC memory PCI target queue is full
		0	TBD	0 1	DATA memory PCI target queue is empty DATA memory PCI target queue is full
0x0012D	Status 9	7:6	Reserved	00	NOP
		5	TBD	0 1	No PAUSE command pending PAUSE is pending (waiting for frame)
		4	TBD	0 1	No START command pending START is pending (waiting for frame)
		3:0	TBD	0000 0001	PAUSE virtual channel switching (disabled) START virtual channel switching (enabled)

6 Clock Architecture



5-9432 (F)

Figure 21. T8110 Main Clocking Paths



5-9433 (F)

Figure 22. T8110 Netref Paths

6 Clock Architecture (continued)

6.1 Clock Input Control Registers

The following registers control the T8110 main clocking paths and Netref paths.

Table 41. Clock Input Control Registers

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00200	PLL#1 rate	PLL#1 input selector	Main divider	Main input selector
0x00204	PLL#2 rate	Reserved	Resource divider	Main inversion select
0x00208	DPLL1 rate	DPLL1 input selector	Reserved	LREF input select
0x0020C	DPLL2 rate	DPLL2 input selector	Reserved	LREF inversion select
0x00210	Reserved	Netref1 LREF select	Netref1 divider	Netref1 input selector
0x00214	Reserved	Netref2 LREF select	Netref2 divider	Netref2 input selector

6.1.1 Main Input Selector Register

The main input selector register controls clock and frame input selection.

Table 42. Main Input Selector Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00200	Main Input Selector	7:0	CKMSR	0000 0000	Select oscillator/crystal (default)
				0001 0001	Select NETREF1
				0001 0010	Select NETREF2
				0010 0001	Select LREF[0:7] individually
				0010 0010	Select LREF[0:3, 4:7] paired
				0100 0001	Select H-bus A clocks
				0100 0010	Select H-bus B clocks
				0100 0100	Select MC1 R clocks
				0100 1000	Select MC1 L clocks
				1000 0001	Select MVIP clocks (/C4 bit clock)
				1000 0011	Select MVIP clocks (C2 bit clock)*
				1000 0010	Select H-MVIP clocks (C16n bitl clock)
				1000 0100	Select SC-bus clocks 2 MHz
				1000 1000	Select SC-bus clocks 4/8 MHz

Choices include the following:

Oscillator/Crystal Clock = XTAL1_IN (16.384 MHz), no frame
NETREF1 Clock = CT_NETREF1 (8 kHz, 1.544 MHz or 2.048 MHz), no frame
NETREF2 Clock = CT_NETREF2 (8 kHz, 1.544 MHz or 2.048 MHz), no frame
LREF Individual Clock = one of LREF[0:7] †, no frame
LREF Paired Clock = one of LREF[0:3], Frame = one of LREF[4:7] †
H-bus A-Clocks Clock = CT_C8_A (8.192 MHz), frame = /CT_FRAME_A (8 kHz)
H-bus B-Clocks Clock = CT_C8_B (8.192 MHz), frame = /CT_FRAME_B (8 kHz)
MC1 R-Clocks Clock = inverted CT_C8_A (4.096 MHz), frame = /CT_FRAME_A (8 kHz)
MC1 L-Clocks Clock = inverted CT_C8_B (4.096 MHz), frame = /CT_FRAME_B (8 kHz)

6 Clock Architecture (continued)

MVIP Clocks Clock = /C4 (4.096 MHz), frame = /FR_COMP (8 kHz)
MVIP Clocks* Clock = /C2 (2.048 MHz), frame = /FR_COMP (8 kHz)
H-MVIP Clocks Clock = /C16± (16.384 MHz), frame = /FR_COMP (8 kHz)
SC-BUS 2 MHz Clock = SCLKx2#, frame = /FR_COMP (8 kHz)
SC-BUS 4/8 MHz Clock = SCLK, frame = /FR_COMP (8 kHz)

* C2 is allowed as the bit clock input.
† Selection of which LREF is controlled at register 0x00208. Selection of LREF polarity is controlled at register 0x0020C.

6.1.2 Main Divider Register

The main divider register contains [divider value –1]. A value of 0x00 yields a divide-by-1 function. A value of 0xFF yields a divide-by-256 function.

Table 43. Main Divider Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00201	Main Divider	7:0	CKMDR	LLLL LLLL	Divider value, {0x00 to 0xFF} = {div1 to div256}, respectively

6.1.3 PLL#1 Input Selector Register

The PLL#1 input selector register controls APLL#1 reference input selection. The choices include the following:

APLL#1 reference clock = oscillator/4 (4.096 MHz)
APLL#1 reference clock = output of the main divider (4.096 MHz or 2.048 MHz)
APLL#1 reference clock = output of the resource divider (4.096 MHz or 2.048 MHz)
APLL#1 reference clock = output of DPLL1 (4.096 MHz or 2.048 MHz)
APLL#1 reference clock = input from signal PRI_REF_IN (4.096 MHz or 2.048 MHz)

Table 44. PLL#1 Input Selector Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00202	PLL#1 Input Selector	7:0	P1ISR	0000 0000 0000 0001 0000 0010 0000 0100 0000 1000	Select oscillator/4 Select main divider output Select resource divider output Select DPLL1 output Select external input PRI_REF_IN

6.1.4 PLL#1 Rate Register

The PLL#1 rate register provides the rate multiplier value to APLL#1. When APLL#1 reference clock is at 4.096 MHz, the (times 16) value must be selected. When APLL#1 reference clock is at 2.048 MHz, the (times 32) value must be selected. A (times 1) value is provided in order to bypass APLL#1.

6 Clock Architecture (continued)

Table 45. PLL#1 Rate Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00203	PLL#1 Rate	7:0	P1RSR	0000 0000 0000 0001 0001 0000	Times 16 (default) Times 32 Times 1 BYPASS (lower nibble is don't care)

6.1.5 Main Inversion Select Register

The main inversion select register controls progammable inversions at various points within the T8110 main clocking paths and Netref paths. Internal points allowed for programmable inversion include the following:

Main clock selection **clk sel** mux output
Netref2 divider output
Netref2 selection mux output
Netref1 divider output
Netref1 selection mux output

Please reference Figure 21 on page 65 and Figure 22 on page 65.

Table 46. Main Inversion Select Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00204	Main Inversion Select	7:5	Reserved	000	NOP (default)
		4	ICMSB	0 1	Don't invert main clock selection (default) Invert main clock selection
		3	N2DSB	0 1	Don't invert Netref2 divider output (default) Invert Netref2 divider output
		2	N2SSB	0 1	Don't invert Netref2 selection (default) Invert Netref2 selection
		1	N1DSB	0 1	Don't invert Netref1 divider output (default) Invert Netref1 divider output
		0	N1SSB	0 1	Don't invert Netref1 selection (default) Invert Netref1 selection

6.1.6 Resource Divider Register

The resource divider register contains [divider value –1]. A value of 0x00 yields a divide-by-1 function. A value of 0xFF yields a divide-by-256 function.

Table 47. Resource Divider Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00205	Resource Divider	7:0	CKRDR	LLLL LLLL	Divider value, {0x00 to 0xFF} = {div1 to div256}, respectively

6 Clock Architecture (continued)

6.1.7 PLL#2 Rate Register

The PLL#2 rate register provides the rate multiplier value to APLL#2. When the APLL#2 reference clock is at 12.352 MHz, the (times 4) value must be selected. When the APLL#2 reference clock is at 6.176 MHz, the (times 8) value must be selected. A (times 1) value is provided in order to bypass APLL#2.

Table 48. PLL#2 Rate Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00207	PLL#2 rate	7:0	P2RSR	0000 0000 0000 0001 0001 xxxx	Times 4 (default) Times 8 Times 1 BYPASS (lower nibble is don't care)

6.1.8 LREF Input Select Registers

The LREF input select register is used in conjunction with the main input selector (0x00200) and provides the selection control among the eight LREF inputs when the main selection is set for either individual or paired LREFs.

The LREF inversion select register allows programmable inversion for each LREF input. Please refer to Figure 21 on page 65 for further details.

Table 49. LREF Input/Inversion Select Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00208	LREF Input Select	7:0	LRISR	0000 0000 0000 0001 0000 0010 0000 0100 0000 1000 0001 0000 0010 0000 0100 0000 1000 0000 0001 0001 0010 0010 0100 0100 1000 1000	No selection (default) Select LREF0 Select LREF1 Select LREF2 Select LREF3 Select LREF4 Select LREF5 Select LREF6 Select LREF7 Select paired, clock = LREF0, frame = LREF4 Select paired, clock = LREF1, frame = LREF5 Select paired, clock = LREF2, frame = LREF6 Select paired, clock = LREF3, frame = LREF7

6 Clock Architecture (continued)

Table 49. LREF Input/Inversion Select Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0020C	LREF Inversion Select	7	IR7SB	0 1	Don't invert LREF7 (default) Invert LREF7
		6	IR6SB	0 1	Don't invert LREF6 (default) Invert LREF6
		5	IR5SB	0 1	Don't invert LREF5 (default) Invert LREF5
		4	IR4SB	0 1	Don't invert LREF4 (default) Invert LREF4
		3	IR3SB	0 1	Don't invert LREF3 (default) Invert LREF3
		2	IR2SB	0 1	Don't invert LREF2 (default) Invert LREF2
		1	IR1SB	0 1	Don't invert LREF1 (default) Invert LREF1
		0	IR0SB	0 1	Don't invert LREF0 (default) Invert LREF0

6.1.9 DPLL1 Input Selector

The DPLL1 input selector selects one of three sources for DPLL1 synchronization input (see Section 6.4.2 on page 79), including the following:

- Main clock selection **clk sel** mux output.
- Main divider output.
- Resource divider output.

6.1.9.1 DPLL1 Rate Register

The DPLL1 rate register controls the DPLL1 output frequency.

Table 50. DPLL1 Input Selector Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0020A	DPLL1 input selector	7:0	D1ISR	0000 0000 0000 0001 0000 0010	Main selector (default) Main divider Resource divider
0x0020B	DPLL1 rate	7:0	D1RSR	0000 0000 0000 0001	DPLL1 output at 4.096 MHz (default) DPLL1 output at 2.048 MHz

6 Clock Architecture (continued)

6.1.10 DPLL2 Input Selector

The DPLL2 input selector selects one of five sources for DPLL2 synchronization input (see Section 6.5.1 on page 81), including the following:

- Main clock selection **clk sel** mux output
- Main divider output
- Resource divider output
- Internal frame
- External input via PRI_REF_IN signal

6.1.10.1 DPLL2 Rate Register

The DPLL2 rate register controls the DPLL2 output frequency.

Table 51. DPLL2 Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0020E	DPLL2 Input Selector	7:0	D2ISR	0000 0000 0000 0001 0000 0010 0000 0100 0000 1000	Main selector (default) Main divider Resource divider T8110 internally generated frame External input PRI_REF_IN
0x0020F	DPLL2 Rate	7:0	D2RSR	0000 0000 0000 0001 0000 0010 0000 0100 0000 1000	DPLL2 output off (default) DPLL2 output at 1.544 MHz DPLL2 output at 3.088 MHz DPLL2 output at 6.176 MHz DPLL2 output at 12.352 MHz

6.1.11 Netref1 Registers

The Netref1 input selector, Netref1 divider and Netref1 LREF select registers control the signal paths used to generate CT_NETREF1 (see Figure 22 on page 65).

Table 52. Netref1 Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00210	Netref1 Input Selector	7:4	N1DSN	0000 0001	Divider input = selector output (default) Divider input = external input NR1_DIV_IN
		3:0	N1ISN	0000 0001 0010 0100 1000	Oscillator/crystal-div-8, 2.048 MHz (default) oscillator/Crystal, 16.384 MHz CT_NETREF2 Input LREF input* Oscillator/Xtal2, 6.176 MHz, or 12.352 MHz
0x00211	Netref1 Divider	7:0	NR1DR	LLLL LLLL	Divider value, {0x00 to 0xFF} = {div1 to div256}, respectively

6 Clock Architecture (continued)

Table 52. Netref1 Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00212	Netref1 LREF Select	7:0	N1LSR	0000 0000 0000 0001 0000 0010 0000 0100 0000 1000 0001 0000 0010 0000 0100 0000 1000 0000	No selection (default) Select LREF0 Select LREF1 Select LREF2 Select LREF3 Select LREF4 Select LREF5 Select LREF6 Select LREF7

* Selection of which LREF is controlled at register 0x00212.

6.1.12 Netref2 Registers

The Netref2 input selector, Netref2 divider, and Netref2 LREF select registers control the signal paths used to generate CT_NETREF2 (see Figure 22 on page 65).

Table 53. Netref2 Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00214	Netref2 Input Selector	7:4	N2DSN	0000 0001	Divider input = selector output (default) Divider input = external input NR1_DIV_IN
		3:0	N2ISN	0000 0001 0010 0100 1000	Oscillator/crystal 1-div-8, 2.048 MHz (default) Oscillator/crystal 1, 16.384 MHz CT_NETREF2 input LREF input* Oscillator/crystal 2 6.176 MHz, or 12.352 MHz
0x00215	Netref2 Divider	7:0	NR2DR	LLLL LLLL	Divider value, {0x00 to 0xFF} = {div1 to div256}, respectively
0x00216	Netref2 LREF Select	7:0	N2LSR	0000 0000 0000 0001 0000 0010 0000 0100 0000 1000 0001 0000 0010 0000 0100 0000 1000 0000	No selection (default) Select LREF0 Select LREF1 Select LREF2 Select LREF3 Select LREF4 Select LREF5 Select LREF6 Select LREF7

* Selection of which LREF is controlled at register 0x00216.

6 Clock Architecture (continued)

6.2 Clock Output Control Registers

The registers listed below control output enable and rate selection of the T8110 clock path outputs.

Table 54. Clock Output Control Register Map

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00220	C8 output rate	FR_COMP width	Netref output enables	Master output enables
0x00224	SCLK output rate	TCLK select	Reserved	CCLK output enables
0x00228	L_SC3 select	L_SC2 select	L_SC1 select	L_SC0 select

6.2.1 Master Output Enables Register

The master output enables register controls the output enables for H-bus and compatibility clocks* for T8110 clock mastering. A clocks refers to the combination of CT_C8_A bit clock and /CT_FRAME_A frame reference.
B clocks refers to the CT_C8_B bit clock and /CT_FRAME_B frame reference. C clocks refers to the compatibility clocks.

* Overall selection includes all C clocks OFF, ALL C clocks ON, or select individual groups of C clocks to be enabled, in conjunction with register 0x00224.

These programmable enables are used in conjunction with master enable register 0x00102, H-bus clock enables, **HCKEB**.

The Netref output enables register controls the output enables for CT_NETREF1 and CT_NETREF2. These programmable enables are used in conjunction with master enable register 0x00102, H-bus clock enables, **HCKEB**.

The CCLK output enables register is used in conjunction with register 0x00220 and controls the output enables for various groupings of compatibility clocks, including the following:

- H-MVIP bit clock only(/C16±)
- MVIP clocks (/C4, C2)
- H-MVIP clocks (/C16±, /C4, C2)
- SC-bus clocks (SCLK, SCLKx2#)
- /FR_COMP compatibility frame reference

Table 55. Master Output Enables Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00220	Master Output Enables	7:4	ABOEN	0000 0001 0010 0011	Disable A and B clock outputs (default) Enable A clock outputs only Enable B clock outputs only Enable both A and B clock outputs
		3:0	CCOEN	0000 0001 0010	Disable compatibility (C clock) outputs Enable C clocks individually* Enable all C clocks

6 Clock Architecture (continued)

Table 55. Master Output Enables Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00221	Netref Output Enables	7:4	N2OEN	0000 0001	CT_NETREF2 disabled (default) CT_NETREF2 enabled
		3:0	N1OEN	0000 0001	CT_NETREF1 disabled (default) CT_NETREF1 enabled
0x00224	CCLK Output Enables	7:4	FRSEN	0000 0001	/FR_COMP disabled (default) /FR_COMP enabled
		3:0	CCSEN	0000 0001 0010 0011 0100	C Clock bit clocks disabled (default) Enable H-MVIP bit clock Enable MVIP clocks Enable H-MVIP all clocks Enable SC-bus clocks

* Overall selection includes all C clocks OFF, all C clocks ON, or select individual groups of C clocks to be enabled, in conjunction with register 0x00224.

6.2.2 FR_COMP Width Registers

The FR_COMP width registers select the pulse width of the /FR_COMP pulse width.

The C8 output rate register selects the CT_C8_A and CT_C8_B clock output frequency 8.192 MHz for ECTF (H.1x0) mode, or 4.096 MHz for MC1 mode.

The SCLK output rate register selects between three SC-Bus clock configurations, including the following:

- SCLK = 2.048 MHz, SCLKx2# = 4.096 MHz
- SCLK = 4.096 MHz, SCLKx2# = 8.192 MHz
- SCLK = 8.192 MHz, SCLKx2# = 8.192 MHz (phase shifted from SCLK)

Table 56. FR_COMP Width Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00222	FR_COMP Width	7:0	FRWSR	0000 0000 0000 0001	/FR_COMP width is 122 ns (default) /FR_COMP width is 244 ns
0x00223	C8 Output Rate	7:4	BCRSN	0000 0001	CT_C8_B output at 8.192 MHz (default) CT_C8_B output at 4.096 MHz, MC1 mode
		3:0	ACRSN	0000 0001	CT_C8_A output at 8.192 MHz (default) CT_C8_A output at 4.096 MHz, MC1 mode
0x00225	Reserved	—	—	—	—
0x00227	SCLK Output Rate	7:0	SCRSR	0000 0000 0000 0001 0000 0010	SCLK = 2 MHz, SCLKx2# = 4 MHz (default) SCLK = 4 MHz, SCLKx2# = 8 MHz SCLK = 8 MHz, SCLKx2# = 8 MHz phase shifted

6 Clock Architecture (continued)

6.2.3 TCLK and L_SCx Select Registers

The TCLK select register controls the selection of various internally generated clocks for output to the **TCLK_OUT** signal.

The L_SCx select registers control the selection of various internally generated clocks for output to the **L_SC0, L_SC1, L_SC2, and L_SC3** signals.

Table 57. TCLK Select and L_SCx Select Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00226	TCLK Select	7:0	TCOSR	0000 0000	TCLK output disabled (default)
				0000 0001	Select osc1/Xtal1
				0000 0010	Select osc2/Xtal2
				0001 0001	Select osc1/Xtal1 INVERTED
				0001 0010	Select osc2/Xtal2 INVERTED
				0010 0000	Select DPLL2 Output
				0010 0001	Select APLL#1 Output, 65.536 MHz
				0010 0010	Select APLL#2 Output, 49.704 MHz
				0011 0000	Select DPLL2 Output INVERTED
				0011 0001	Select APLL#1 Output INVERTED
				0011 0010	Select APLL#2 Output INVERTED
				0100 0000	Select generated 2.048 MHz
				0100 0001	Select generated 4.096 MHz
				0100 0010	Select generated 8.192 MHz
				0100 0100	Select generated 16.384 MHz
				0100 1000	Select generated 32.768 MHz
				0101 0000	Select generated 2.048 MHz INVERTED
				0101 0001	Select generated 4.096 MHz INVERTED
				0101 0010	Select generated 8.192 MHz INVERTED
				0101 0100	Select generated 16.384 MHz INVERTED
				0101 1000	Select generated 32.768 MHz INVERTED
				1000 0000	Select generated frame
				1000 0001	Select generated CT_NETREF1
				1000 0010	Select generated CT_NETREF2
				1001 0000	Select generated Frame INVERTED
				1001 0001	Select generated CT_NETREF1 INVERTED
				1001 0010	Select generated CT_NETREF2 INVERTED

6 Clock Architecture (continued)

Table 57. TCLK Select and L_SCx Select Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00228 (0x00229) (0x0022A) (0x0022B)	L_SC0 Select L_SC1 Select L_SC2 Select L_SC3 Select	7:0	LC0SR (LC1SR) (LC2SR) (LC3SR)	0000 0000 0000 0001 0001 0001 0100 0000 0100 0001 0100 0010 0100 0100 0100 1000 0101 0000 0101 0001 0101 0010 0101 0100 0101 1000 1000 0000 1000 0001 1000 0010 1001 0000 1001 0001 1001 0010	L_SCx output disabled (default) Select osc1/Xtal1 Select osc1/Xtal1 INVERTED Select generated 2.048 MHz Select generated 4.096 MHz Select generated 8.192 MHz Select generated 16.384 MHz Select generated 32.768 MHz Select generated 2.048 MHz INVERTED Select generated 4.096 MHz INVERTED Select generated 8.192 MHz INVERTED Select generated 16.384 MHz INVERTED Select generated 32.768 MHz INVERTED Select generated frame Select generated CT_NETREF1 Select generated CT_NETREF2 Select generated frame INVERTED Select generated CT_NETREF1 INVERTED Select generated CT_NETREF2 INVERTED

6.3 Clock Register Access

The T8110 clock control registers consist of two identical sets of registers, X and Y. At any given time, only one set is actually controlling the clocking (denoted as the ACTIVE set), while the other is in a standby state (INACTIVE set). Either set, X or Y, may be the ACTIVE set, as determined by a state machine that tracks the clock fallback control and status and assigns either set to be ACTIVE accordingly. For more details, see Section 6.7.1 on page 82. Users may only access one register set at a time. By default, access is allowed to the current INACTIVE set, but access to the ACTIVE set is allowed via the clock reg access select register, 0x00106; see Section 5.1.4 on page 51.

6.4 Clock Circuit Operation—Analog PLL#1

APLL#1 can accept either a 4.096 MHz or 2.048 MHz reference clock and performs a corresponding multiplication function to supply a 65.536 MHz operating clock for the T8110. Additionally, APLL#1 may be bypassed for circuit diagnostic purposes. Please refer to Figure 21 on page 65.

6.4.1 Main Clock Selection, Bit Clock and Frame

APLL#1 clock references are selectable as stand-alone bit clocks, frames, or a pairing of bit clock and frame (see main input selector register, 0x00200). The bit clock output of the main clock selection is available as input to the main divider, resource divider, and DPLL#1.

6 Clock Architecture (continued)

Table 58. Bit Clock and Frame

Bit Clock	Corresponding 8 kHz Frame	Value(s)
CT_NETREF1, CT_NETREF2	NA	1.544 MHz (T1), 2.048 MHz (E1)
NA	CT_NETREF1, CT_NETREF2	8 kHz
CT_C8_A	CT_FRAME_An	8.192 MHz (ECTF) 4.096 MHz (MC1)
CT_C8_B	CT_FRAME_Bn	8.192 MHz (ECTF) 4.096 MHz (MC1)
/C16	FRn_COMP	6.384 MHz (H-MVIP)
/C4	FRn_COMP	4.096 MHz (MVIP)
C2	FRn_COMP	2.048 MHz (MVIP*)
SCLK	FRn_COMP	2.048 MHz, 4.096 MHz 8.192 MHz (SC-bus)
SCLKx2#	FRn_COMP	4.096 MHz, 8.192 MHz (SC-bus)
LREF[0]	LREF[4]†	System-specific
LREF[1]	LREF[5]†	System-specific
LREF[2]	LREF[6]†	System-specific
LREF[3]	LREF[7]†	System-specific
LREF[4]	NA	System-specific
LREF[5]	NA	System-specific
LREF[6]	NA	System-specific
LREF[7]	NA	System-specific

* MVIP /C4 is typically the bit clock. C2 is selectable as the bit clock as well.
† When LREF pairing is enabled.

6.4.1.1 Watchdog Timers

A set of watchdog timers is available for all H.1x0, H-MVIP, MVIP, and SC-bus clocks. No watchdogs are available for LREF[7:0] directly; however, the LREF inputs may be monitored indirectly via watchdogs on the DPLL#1 and DPLL#2 sync inputs, or via the failsafe mechanism, see Section 6.7.2 on page 88. The watchdogs sample the incoming clocks at 32.768 MHz derived from the Xtal1 crystal and monitor for loss of signal, as follows.

Table 59. Watchdog Timer Description

Watchdog	Signal, Value	Description
H.1x0 clock monitors*	CT_C8_A at 8.192 MHz CT_C8_B at 8.192 MHz	ECTF mode. Checks for CT_C8 rising edge within a 35 ns window of its expected arrival.
	CT_C8_A at 4.096 MHz CT_C8_B at 4.096 MHz	MC1 mode. monitors for loss of signal (falling edges).
FRAME monitors	CT_FRAME_An CT_FRAME_Bn FRn_COMP	Monitors for 8 kHz frequency. Detects frame over-flow (i.e., next frame pulse too late) and frame underflow (i.e., next frame pulse too early).

6 Clock Architecture (continued)

Table 59. Watchdog Timer Description (continued)

Watchdog	Signal, Value	Description
NETREF monitors †	CT_NETREF1 at 1.544 MHz CT_NETREF2 at 1.544 MHz	NETREF is T1 bit clock. monitors for loss of signal (rising or falling edges).
	CT_NETREF1 at 2.048 MHz CT_NETREF2 at 2.048 MHz	NETREF is E1 bit clock. monitors for loss of signal (rising or falling edges).
	CT_NETREF1 at 8 kHz CT_NETREF2 at 8 kHz	NETREF is 8 kHz frame reference. Monitors for 8 kHz frequency. Detects frame overflow (i.e., next frame pulse too late) and frame underflow (i.e., next frame pulse too early).
Compatibility clock monitors	C16n at 16.384 MHz C4n at 4.096 MHz C2 at 2.048 MHz SCLK, SCLKx2# at any of their defined values.	Gross loss-of-signal detector—clocks are sampled and normalized to 1.024 MHz. It can take up to 976 ns for these watchdog timers to detect loss of a compatibility clock.
DPLL#1, DPLL#2 Sync Monitors ‡	Output of mux selector to the SYNC input of each DPLL (8 kHz)	Monitors for 8 kHz frequency. Detects frame overflow (i.e., next frame pulse too late) and frame underflow (i.e., next frame pulse too early).

* User selects frequency at which to monitor the CT_C8 clocks via register 0x0010C, watchdog select, C8.
† User selects frequency at which to monitor the CT_NETREF signals via register 0x0010D, watchdog select, Netref.
‡ DPLL sync reference is expected to be 8 kHz.

6.4.1.2 Frame Center Sampling

Frame center samples are used in order to phase-align the incoming frame reference to the internally generated frame reference, see Section 6.4.5.1 on page 80. The incoming frame reference signal is sampled with a recovered clock (output of the APLL#1 feedback divider) to determine the frame center. Frame center sampling is only relevant when the main clock selection is based on a paired bit clock/frame reference, as follows.

Table 60. Frame Center Sampling

Frame Signal	Corresponding Bit Clock	Sample Clock
CT_FRAME_An	CT_C8_A	Recovered 8.192 MHz, rising edge
CT_FRAME_Bn	CT_C8_B	Recovered 8.192 MHz, rising edge
FRn_COMP	C16n (H-MVIP) or C4n (MVIP) or C2 (MVIP)	Recovered 4.096 MHz, falling edge
FRn_COMP	SCLK, or SCLKx2# (SC-bus)	Recovered 2.048 MHz, rising edge
LREF[4]	LREF[0]	Recovered 2.048 MHz, rising edge
LREF[5]	LREF[1]	Recovered 2.048 MHz, rising edge
LREF[6]	LREF[2]	Recovered 2.048 MHz, rising edge
LREF[7]	LREF[3]	Recovered 2.048 MHz, rising edge

6 Clock Architecture (continued)

6.4.2 Main and Resource Dividers

Two independently programmable dividers are available to divide down the main clock selection signal. Function ranges from divide-by-1 (bypass) to divide-by-256.

- For BINARY divider values of 1, 2, 4, 8, 16, 32, 64, 128, and 256, output is 50% duty cycle.
- For a divider value of 193, output is almost 50% duty cycle (low-level duration is one clock cycle shorter than high-level duration).
- For **all** other divider values, output is a pulse whose width is ONE FULL PERIOD of the main clock selection signal.

Output of both dividers is available to the DPLL#1 and the APLL#1 reference selector. Output of the main divider is also available at the PRI_REF_OUT chip output.

Both dividers are reset whenever a changeover between X and Y clock register sets is detected, see Section 6.3 on page 76. This allows for immediate loading of the newly activated divider register values.

6.4.3 DPLL#1

A digital phase-lock loop is provided to generate a 4.096 MHz or 2.048 MHz reference to APLL#1 (selectable via register 0x0020B, DPLL1 rate). The DPLL#1 operates at 32.768 MHz, derived from the Xtal1 crystal input. The DPLL#1 synchronization source is selectable (register 0x0020A, DPLL1 input selector) between the main clock selection signal, the output of the resource divider, or the output of the main divider, and is intended to be presented as an 8 kHz frame reference. DPLL#1 is determined to be in-lock or out-of-lock based on the state of the output clock when an edge transition is detected at the synchronization source. An out-of-lock condition results in a DPLL#1 correction, which can either lengthen or shorten its current output clock period by 30.5 ns.

6.4.4 Reference Selector

The APLL#1 reference clock is selectable between five possible sources via register 0x00202, PLL#1 input selector. A 4.096 MHz or 2.048 MHz reference must be provided.

- Xtal1 crystal (16.384 MHz) divided-by-4
- Main divider output
- Resource divider output
- DPLL#1 output
- PRI_REF_IN external chip input

6.4.5 Internal Clock Generation

The main internal functions of T8110 are synchronous to the 65.536 MHz output of APLL#1. This clock is further divided to generate 32.768 MHz, 16.384 MHz, and 8 kHz internal reference signals. Additional divide-down values to 8.192 MHz, 4.096 MHz, and 2.048 MHz are generated. These generated clocks are the source for H.1x0, H-MVIP, MVIP, and SC-bus clocks when the T8110 is mastering the bus clocks; see Section 6.2 on page 73. These internally generated clocks can either be free-running, or can be aligned to the incoming main selection clock and frame, via a phase alignment circuit (see next section).

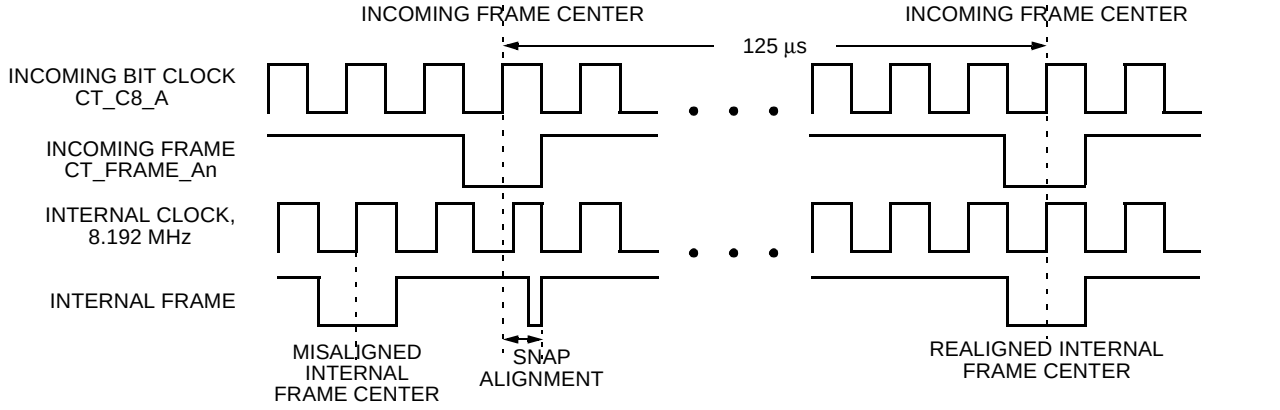
6 Clock Architecture (continued)

6.4.5.1 Phase Alignment

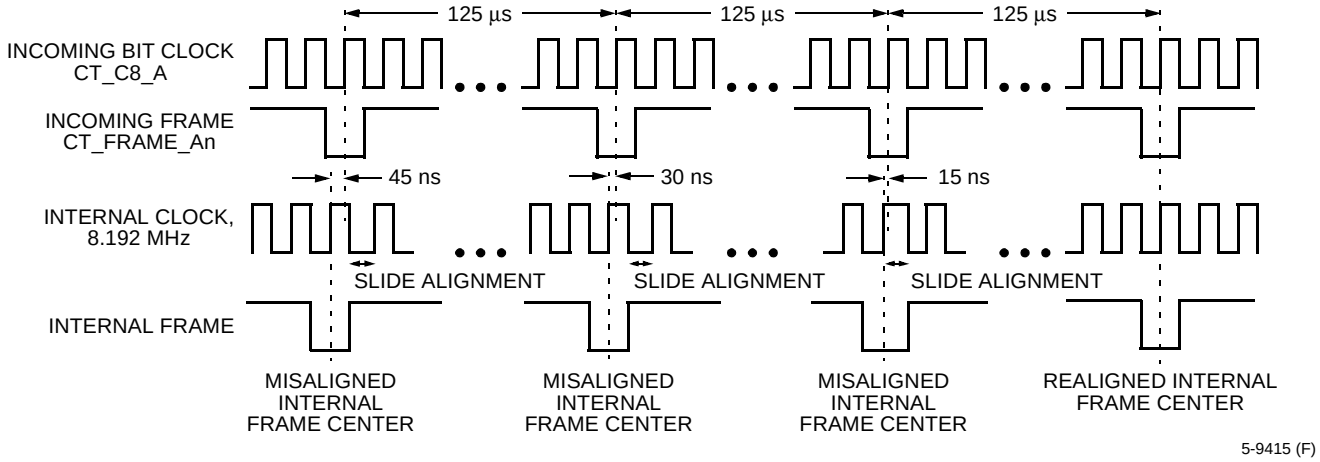
Phase alignment allows the free-running internally generated clocks to be forced into alignment with the incoming main selection clock and frame, under the following conditions. The main selection clock is based on a paired bit clock/frame reference; see Section 6.4.1.2 on page 78, and the phase alignment circuit is enabled (via register 0x00107, phase alignment select).

The incoming frame center is monitored via the frame center samplers (see Section 6.4.1.2 on page 78) and compared to the state of the internally generated frame. The circuit determines whether the frame centers are aligned. If not, three possible actions take place:

- NOP, no corrections when phase alignment is DISABLED.
- SNAP correction, the internally generated clocks and frame immediately snap into alignment with the incoming frame center.
- SLIDE correction, the internally generated clocks and frame gradually slide into alignment with the incoming frame center, at a rate of one 65.536 MHz clock period per frame. The sliding occurs in one direction only and creates frame periods that are 15.25 ns longer than 125 µs until the frames are aligned. Please refer to Figure 23.



Phase Alignment—SNAP



Phase Alignment—SLIDE

Figure 23. T8110 Phase Alignment, SNAP and SLIDE

6 Clock Architecture (continued)

6.4.6 APLL#1 Lock Status

TBD

6.5 Clock Circuit Operation, Analog PLL#2

APLL#2 requires either a 6.176 MHz or 12.352 MHz reference clock, to produce a 49.408 MHz clock for operating DPLL#2. A user-supplied rate multiplier (register 0x00207, PLL#2 rate) provides either a times 8 function (when reference clock = 6.176 MHz) or a times 4 function (for reference clock = 12.352 MHz). Additionally, APLL#2 may be bypassed for circuit diagnostic purposes (see Figure 21 on page 65).

6.5.1 DPLL#2

A second digital phase-lock loop is provided to generate various derivations of T1 operating frequencies, available by selection via the TCLK_OUT output. The possible output frequencies are selectable via register 0x0020F, DPLL2 rate, and include 1.544 MHz, 3.088 MHz, 6.176 MHz, and 12.352 MHz. The DPLL#2 input clock operates at 49.408 MHz, from the APLL#2 output. Synchronization sources for DPLL#2 include the same sources provided to DPLL#1 (selectable between the main clock selection signal, the output of the resource divider, or the output of the main divider), and two additional sources, including the T8110 internally generated frame signal, and the PRI_REF_IN input. These selections are available via register 0x0020E, DPLL2 input selector. DPLL#2 is determined to be in-lock or out-of-lock based on the state of its output when an edge transition is detected at the synchronization source. An out-of-lock condition results in a DPLL#2 correction, which can either lengthen or shorten its current output clock period by 20.2 ns.

6.6 Clock Circuit Operation, CT_NETREF Generation

The T8110 provides two independently programmable paths to generate CT_NETREF1 and CT_NETREF2, via registers 0x00210—0x00216. Each CT_NETREF is individually enabled with register 0x00221, Netref output enables. Each path consists of a source selector MUX and a divider circuit (see Figure 22 on page 65).

6.6.1 Netref Source Select

Xtal1 input DIV 8 (2.048 MHz)

Xtal1 input (16.384 MHz)

Xtal2 input (6.176 MHz or 12.352 MHz)

LREF[7:0]

CT_NETREFx (the other netref—i.e., CT_NETREF1 can be derived from CT_NETREF2, and vise-versa)

The output of the source select mux is made available directly to the Netref divider, and also to chip output (NR1_SEL_OUT, NR2_SEL_OUT).

6 Clock Architecture (continued)

6.6.2 Netref Divider

Each Netref path provides a divider, from a divide-by-1 function up to a divide-by-256 function. The clock source for the divider is selectable between the output of the source select mux, or from external chip input (NR1_DIV_IN, NR2_DIV_IN).

- For binary divider values of 1, 2, 4, 8, 16, 32, 64, and 128, output is 50% duty cycle.
- For divider values of 256, 193, plus all other nonbinary values, output is a pulse whose width is one-half of a clock period, asserted during the second half of the divider clock period.

The Netref dividers are reset whenever a changeover between X and Y clock register sets is detected (see Section 6.3 on page 76). This allows for immediate loading of the newly activated divider register values.

6.7 Clock Circuit Operation—Fallback and Failsafe

Fallback is a means to alter the reference source to APLL#1, by switching between two clock control register sets upon detection of a fallback event. Failsafe is a feature to provide a safety net for the reference source to APLL#1, independent of clock fallback.

6.7.1 Clock Fallback

Clock fallback is a means to alter the APLL#1 reference clock source upon detection of a fallback event, and is controlled by eight registers, 0x00108—0x0010F (refer to Section 5.1.4 on page 51). These registers enable and control the state transitions that determine which of two clock register sets is used to control the APLL#1 reference clock source (see Section 6.1 on page 66 through Section 6.3 on page 76, Table 62 on page 85, and Figure 25 on page 84).

6.7.1.1 Fallback Events

Clock fallback (transition from primary to secondary clock sets) can only occur if the fallback mode is enabled (register 0x00109, lower nibble), and a fallback event occurs. When enabled, there are three ways to trigger the fallback event:

- Software, via a FORCE_FALLBACK command. The user sets bit 2 of the fallback control register, 0x00108, creating a software-invoked fallback event.
- Hardware via the fallback trigger enable registers, 0x0010A—0x0010B. User may enable specific watchdog timers, and corresponding fallback trigger enable bits. If a watchdog timer indicates a clock error, and its corresponding trigger enable bit is set, a hardware-invoked fallback event is produced.
- Hardware, legacy modes, via the fallback type select register, 0x00109, upper nibble. The legacy modes are included to maintain backwards compatibility with earlier *Ambassador* devices. User may enable specific watchdog timers, but the fallback trigger enable registers are ignored. Instead, the watchdogs which are allowed to trigger a fallback event are automatically selected based on the state of the main input selector register, 0x00200 (refer to Table 61). If a watchdog timer indicates a clock error, and its corresponding trigger enable is selected via the main input selector, a hardware-invoked fallback event is produced.

6 Clock Architecture (continued)

Table 61. Legacy Mode Fallback Event Triggers

Main Input Selector Function (Register 0x00200)	Selected Watchdog Triggers (Legacy Modes)
Oscillator/Crystal	None
CT_NETREF1	NETREF1 watchdog
CT_NETREF	NETREF2 watchdog
LREF, Individual	None
LREF, Paired	None
H-bus, A Clocks	CT_C8_A and /CT_FRAME_A watchdogs
H-bus, B Clocks	CT_C8_B and /CT_FRAME_B watchdogs
MC1, R Clocks	None
MC1, L Clocks	None
MVIP clocks (/C4 or C2 bit clock)	/C4, C2, and /FR_COMP watchdogs
H-MVIP Clocks	/C16, /C4, C2, and /FR_COMP watchdogs
SC-bus clocks (2 MHz or 4/8 MHz)	SCLK, SCLKx2#, and /FR_COMP watchdogs

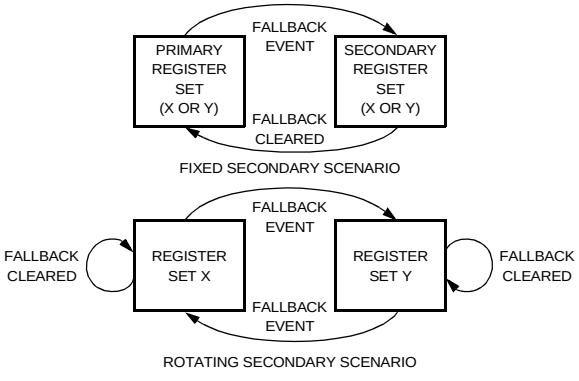
6.7.1.2 Fallback Scenarios—Fixed vs. Rotating Secondary

When clock fallback is enabled (register 0x00109, lower nibble), there are two possible scenarios for transitioning between the primary and secondary clock sets.

In a fixed secondary scheme, a fallback event switches the active clock set from primary to secondary. When the fallback event is cleared (via user-invoked CLEAR_FALLBACK), the active clock set returns to primary.

In a rotating secondary scheme, a fallback event switches the active clock set from primary to secondary. When the fallback event is cleared, the secondary remains as the new active clock set. In effect, the secondary becomes the new primary, and the primary becomes the new secondary.

The concepts are illustrated in the figure below.



5-9420 (F)

Figure 24. Fallback - Fixed vs. Rotating Secondary

6 Clock Architecture (continued)

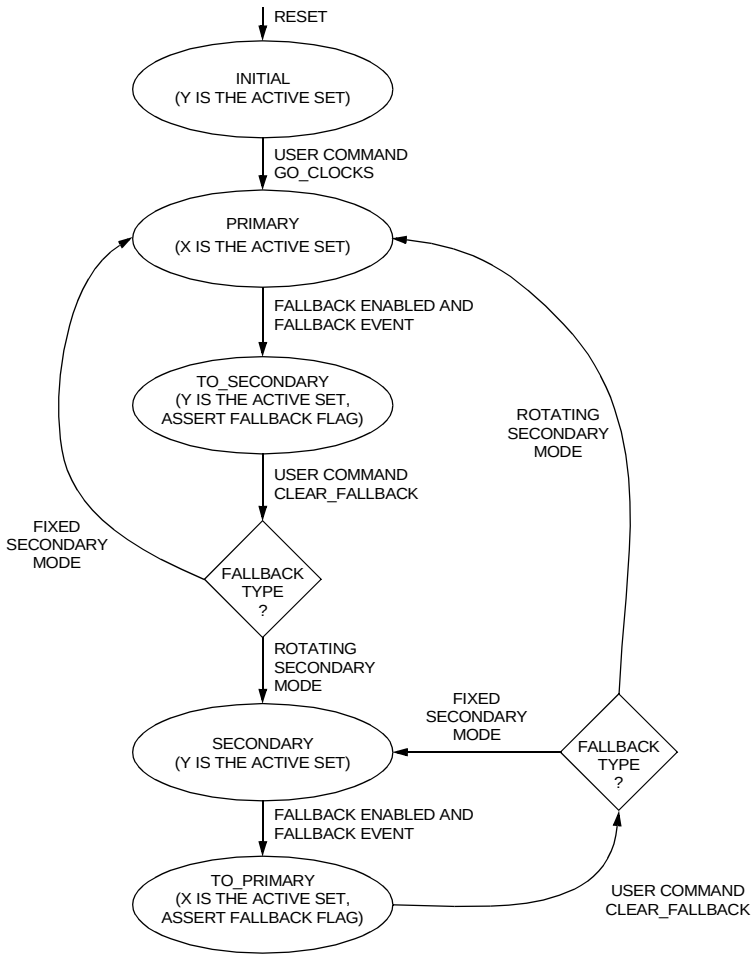


Figure 25. T8110 Clock Fallback States

6 Clock Architecture (continued)

Table 62. Clock Fallback State Descriptions

Clock Fallback State	Description	Exit To	Exit Condition
INITIAL	Y is the active clock register set. Default value provides Xtal1-div-4 reference.	PRIMARY	User issues GO_CLOCKS command (set register 0x00108 bit 0).
PRIMARY	X is the active clock register set and controls APLL#1 refclk.	TO_SECONDARY	Fallback is enabled **and** fallback event*.
TO_SECONDARY	Y is the active clock register set and controls APLL#1 refclk. Fallback flag is asserted.	PRIMARY	User issues CLEAR_FALLBACK command (set register 0x00108 bit 1) and fallback type = fixed secondary†.
		SECONDARY	User issues CLEAR_FALLBACK command (set register 0x00108 bit 1) **and** fallback type = rotating secondary†.
SECONDARY	Y is the active clock register set and controls APLL#1 refclk.	TO_PRIMARY	Fallback is enabled **and** fallback event*.
TO_PRIMARY	X is the active clock register set and controls APLL#1 refclk. Fallback flag is asserted.	SECONDARY	User issues CLEAR_FALLBACK command (set register 0x00108 bit 1) **and** fallback type = fixed secondary†.
		PRIMARY	User issues clear_fallback command (set register 0x00108 bit 1) **and** fallback type = rotating secondary†.

* Fallback event: refer to Section 6.7.1.1 on page 82.
† Fixed, rotating secondary, refer to Section 6.7.1.2 on page 83.

6.7.1.3 H-Bus Clock Enable/Disable on Fallback

The previous *Ambassador* devices allowed a fallback mode (A/B fallback) which automatically allowed an H.1x0 bus clock master to detect an error in its own output clock and remove itself from the bus, or a clock slave to detect an error on its incoming clock and **promote** itself to clock master. The H-bus clocks include:

- A-clocks: CT_C8_A, CT_FRAME_An
- B-clocks: CT_C8_B, CT_FRAME_Bn
- C-clocks: C16n+/-, C4n, C2, SCLK, SCLKx2#, FRn_COMP

Refer to Figure 26–and Table 63. The T8110 allows for this mode of operation in two ways:

Register 0x00109(7:4) = 0100: Legacy mode, A/B fallback—when this mode is selected, the fallback triggers allowed are predefined based on the main input clock selection, and the state machine which controls H-bus clock enable/disable is activated.

Register 0x00109(7:4) = 1001: Nonlegacy mode—when this mode is selected, the fallback trigger enable registers determine what triggers a fallback, and the state machine which controls H-bus clock enable/disable is activated.

6 Clock Architecture (continued)

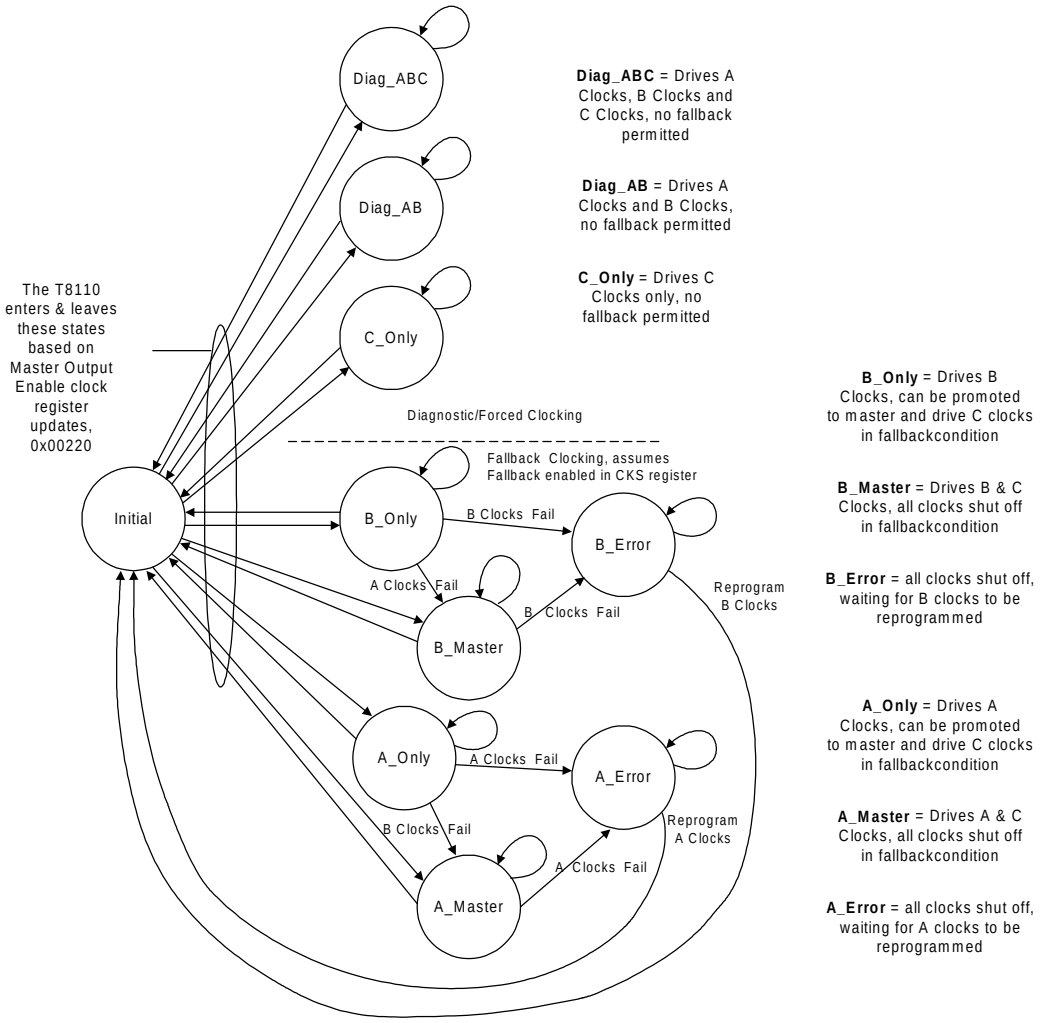


Figure 26. T8110 H-Bus Clock Enable States

6 Clock Architecture (continued)

Table 63. H-Bus Clock Enable State Description

H-Bus Clock Enable State	Description	Exit To	Exit Condition
INITIAL	Initial condition, waiting for clock output control register programming.	Any of the other states	User update of the clock output control register (0x00220, master output enables)
DIAG_ABC	T8110 is driving **all** H-bus clocks (diagnostic mode).	INITIAL	User update of the clock output control register (0x00220, master output enables)
DIAG_AB	T8110 is driving **both** the H-bus A and B clocks (diagnostic mode).	INITIAL	User update of the clock output control register (0x00220, master output enables)
C_ONLY	T8110 is driving **only** the H-bus C clocks.	INITIAL	User update of the clock output control register (0x00220, master output enables)
A_MASTER	T8110 clock output control registers are programmed to drive A clocks and C clocks (T8110 is an A-clock master), **or** T8110 was supplying a backup A clock and has been promoted to A clock master.	A_ERROR	A clock error on CT_C8_A or CT_FRAME_An is detected; disable clock outputs
		INITIAL	User update of the clock output control register (0x00220, master output enables)
A_ONLY	T8110 clock output control registers are programmed to drive A clocks only (T8110 is a B clock slave, and supplies a backup A clock).	A_MASTER	A clock error on CT_C8_B or CT_FRAME_Bn is detected; promote to A clock master
		A_ERROR	A clock error on CT_C8_A or CT_FRAME_An is detected; disable clock outputs
		INITIAL	User update of the clock output control register (0x00220, master output enables)
A_ERROR	T8110 has detected a clock error while driving the A clocks, and has stopped driving any H bus clocks.	INITIAL	User update of the clock output control register (0x00220, master output enables)
B_MASTER	T8110 clock output control registers are programmed to drive B clocks and C clocks (T8110 is a B-clock master), **or** T8110 was supplying a backup B clock and has been promoted to B clock master.	B_ERROR	A clock error on CT_C8_B or CT_FRAME_Bn is detected; disable clock outputs
		INITIAL	User update of the clock output control register (0x00220, master output enables)

6 Clock Architecture (continued)

Table 63. H-Bus Clock Enable State Description (continued)

H-Bus Clock Enable State	Description	Exit To	Exit Condition
B_ONLY	T8110 clock output control registers are programmed to drive B clocks only (T8110 is an A clock slave, and supplies a backup B clock).	B_MASTER	A clock error on CT_C8_A or CT_FRAME_An is detected; promote to B clock master
		B_ERROR	A clock error on CT_C8_B or CT_FRAME_Bn is detected; disable clock outputs
		INITIAL	User update of the clock output control register (0x00220, master output enables)
B_ERROR	T8110 has detected a clock error while driving the B clocks, and has stopped driving any H bus clocks.	INITIAL	User update of the clock output control register (0x00220, master output enables)

6.7.2 Clock Failsafe

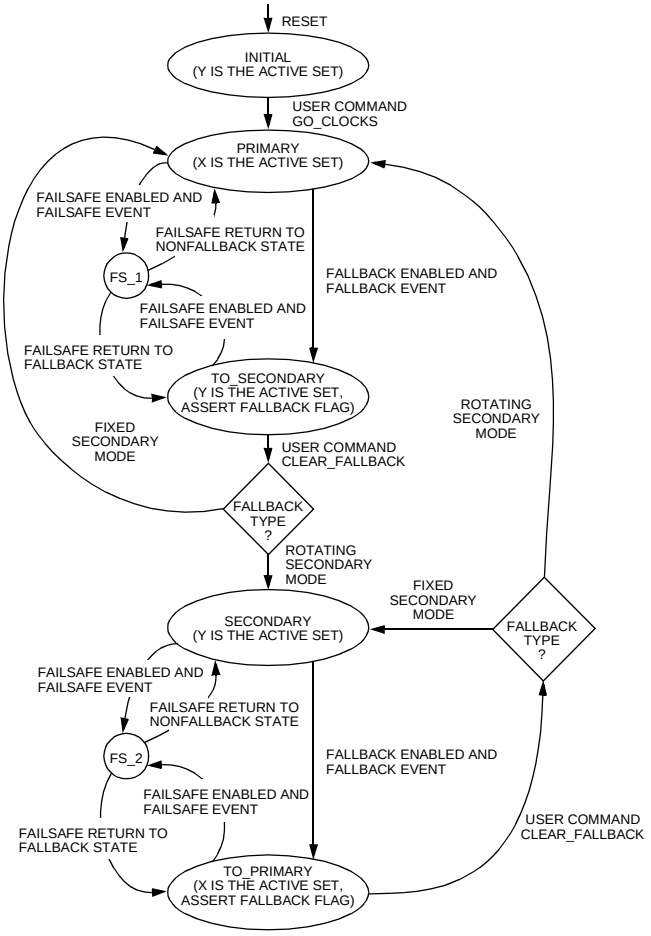
Clock failsafe is a feature to provide a safety net for the APLL#1 reference clock source and is controlled by three registers, 0x00114—0x00116 (refer to Section 5.1.12 on page 57). A failsafe event overrides the active Clock Control registers and forces the APLL#1 clock selection to be a fixed 4.096 MHz, derived from the XTAL1 crystal, divided by four. Transition into one of the failsafe states is independent of clock fallback (i.e., can enter from any state other than INITIAL). Transitions out of the failsafe states are by user command and allow re-entry into either a nonfallback (primary or secondary) or a fallback (TO_SECONDARY or TO_PRIMARY) state. Refer to Table 64 and Figure 27.

6.7.2.1 Failsafe Events

Clock failsafe (transition from either clock register set to a forced XTAL1-div-4 APLL#1 reference clock) can only occur if the failsafe mode is enabled (register 0x00115, lower nibble), and a failsafe event occurs. A failsafe event is triggered by a watchdog error on the APLL#1 reference clock (i.e., loss-of-reference).

Additionally, an out-of-lock (OOL) condition is provided for debug purposes. This does not trigger a failsafe event, but does indicate potential difficulty with the APLL. A lock status flag is provided out of APLL#1, and the OOL is defined by exceeding a user-defined threshold value (registers 0x00117 and 116). The lock status is a flag indicating when APLL#1 is making a correction to maintain synchronization. The flag is continuously sampled. If enough active flags are sampled in a row to exceed the user-defined threshold, this condition is reported via the system status register (0x00125).

6 Clock Architecture (continued)



5-9421 (F)

Figure 27. T8110 Clock Failsafe States

Table 64. Clock Failsafe State Descriptions

Clock Failsafe State	Description	Exit To	Exit Condition
FS_1	APLL#1 refclk is forced to Xtal1-div-4 FAILSAFE FLAG is asserted	PRIMARY	User issues FAILSAFE_RETURN to nonfallback state command (set register 0x00114 bit 0).
		TO_SECONDARY	User issues FAILSAFE_RETURN to fallback state command (set register 0x00114 bit 1).
FS_2	APLL#1 refclk is forced to Xtal1-div-4 FAILSAFE FLAG is asserted	SECONDARY	User issues FAILSAFE_RETURN to non-fallback state command (set register 0x00114 bit 0).
		TO_PRIMARY	User issues FAILSAFE_RETURN to fallback state command (set register 0x00114 bit 1).

7 Frame Group and FG I/O

There are eight independently programmable T8110 frame group/FGIO signals, FG[7:0]. In the frame group mode, the pin is an 8 kHz frame reference output, with programmable pulse width, polarity, and delay offset from the internally generated frame reference. In the FGIO mode, the pin behaves as a general-purpose register bit, with programmable direction (IN or OUT) and read masking. The FG7 signal allows for an additional mode of operation, providing a timer via a 16-bit programmable counter.

Table 65. Frame Group and FG I/O Register Map

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00400	FG0 rate	FG0 width	FG0 upper start	FG0 lower start
0x00410	FG1 rate	FG1 width	FG1 upper start	FG1 lower start
0x00420	FG2 rate	FG2 width	FG2 upper start	FG2 lower start
0x00430	FG3 rate	FG3 width	FG3 upper start	FG3 lower start
0x00440	FG4 rate	FG4 width	FG4 upper start	FG4 lower start
0x00450	FG5 rate	FG5 width	FG5 upper start	FG5 lower start
0x00460	FG6 rate	FG6 width	FG6 upper start	FG6 lower start
0x00470	FG7 rate	FG7 width	FG7 upper start	FG7 lower start
0x00474	FG7 mode upper	FG7 mode lower	FG7 counter high byte	FG7 counter low byte
0x00480	Reserved	FGIO R/W	FGIO read mask	FGIO data register

7.1 Frame Group Control Registers

7.1.1 FGx Lower and Upper Start Registers

The FGx lower and upper start registers provide a 12-bit delay offset value for the corresponding frame group bit. Offsets are relative to the T8110 internally generated 8 kHz frame reference and have a resolution down to one 32.768 MHz clock period (30.5 ns increments).

Table 66. FGx Lower and Upper Start Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00400 0x00410 0x00420 0x00430 0x00440 0x00450 0x00460 0x00470	FG0 Lower Start (FG1 Lower Start) (FG2 Lower Start) (FG3 Lower Start) (FG4 Lower Start) (FG5 Lower Start) (FG6 Lower Start) (FG7 Lower Start)	7:0	F0LLR (F1LLR) (F2LLR) (F3LLR) (F4LLR) (F5LLR) (F6LLR) (F7LLR)	LLLL LLLL	Lower 8 bits of 12-bit start offset

7 Frame Group and FG I/O (continued)

Table 66. FGx Lower and Upper Start Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00401 (0x00411) (0x00421) (0x00431) (0x00441) (0x00451) (0x00461) (0x00471)	FG0 Upper Start (FG1 Upper Start) (FG2 Upper Start) (FG3 Upper Start) (FG4 Upper Start) (FG5 Upper Start) (FG6 Upper Start) (FG7 Upper Start)	7:0	F0ULR (F1ULR) (F2ULR) (F3ULR) (F4ULR) (F5ULR) (F6ULR) (F7ULR)	0000 LLLL	Upper 4 bits of 12 bit start offset

7.1.2 FGx Width Registers

The FGx width registers control the polarity and the pulse widths generated for the corresponding frame group bit. The pulse-width programming works in conjunction with the FGx rate registers to provide 1-bit, 2-bit, 4-bit, 1-byte, and 2-byte wide pulses for any of the available frame group rates (see Table 67).

Table 67. FGx Width Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00402 (0x00412) (0x00422) (0x00432) (0x00442) (0x00452) (0x00462) (0x00472)	FG0 Width (FG1 Width) (FG2 Width) (FG3 Width) (FG4 Width) (FG5 Width) (FG6 Width) (FG7 Width)	7	F0ISB (F1ISB) (F2ISB) (F3ISB) (F4ISB) (F5ISB) (F6ISB) (F7ISB)	0 1	Generate active-high pulse (default) Generate active-low pulse
		6:0	F0WSP (F1WSP) (F2WSP) (F3WSP) (F4WSP) (F5WSP) (F6WSP) (F7WSP)	000 0000 000 0001 000 0010 000 0100 001 0000 010 0000	No pulse generated (default) 1-bit wide pulse 2-bit wide pulse 4-bit wide pulse 1-byte wide pulse 2-byte wide pulse

7.1.3 FGx Rate Registers

The FGx rate registers either enable FGIO operation*, or work in conjunction with FGx width registers to provide various width frame group pulses at rates of 2.048 MHz, 4.096 MHz, 8.192 MHz, or 16.384 MHz.

* FGIO operation is controlled at registers 0x00480—482. Refer to Section 7.3 on page 93.

7 Frame Group and FG I/O (continued)

Table 68. FGx Rate Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00403	FG0 Rate	7:0	F0RSR	0000 0000	Off (default)
(0x00413)	(FG1 Rate)		(F1RSR)	0000 0001	FGIO enabled* (not used as a frame group)
(0x00423)	(FG2 Rate)		(F2RSR)	0000 0010	FGx rate = 2.048 MHz
(0x00433)	(FG3 Rate)		(F3RSR)	0000 0100	FGx rate = 4.096 MHz
(0x00443)	(FG4 Rate)		(F4RSR)	0000 1000	FGx rate = 8.192 MHz
(0x00453)	(FG5 Rate)		(F5RSR)	0000 1001	FGx rate = 16.384 MHz
(0x00463)	(FG6 Rate)		(F6RSR)		
(0x00473)	(FG7 Rate)		(F7RSR)		

* FGIO operation is controlled at registers 0x00480—482. Refer to Section 7.3 on page 93.

7.2 FG7 Timer Option

The FG7 signal allows for an added function of a timer output, via a 16-bit programmable counter.

7.2.1 FG7 Counter (Low and High Byte) Registers

The FG7 counter (low and high byte) registers set the timer value. The timer is actually a divider, so the value entered must be [divider value – 1], i.e., 0000000000000011 would yield a div-by-4 operation. The FG7 mode lower register enables the timer option, with two clock source options: T8110 internal frame or an external timer clock via the FG6 signal. The FG7 mode upper register controls the shape of the timer pulse. For more details, see Section 7.4.3 on page 96.

Table 69. FG7 Counter (Low and High Byte) Registers

Byte Address	Name	Bit	Mnemonic	Value	Function
0x00474	FG7 Counter, Low Byte	7:0	FCLLR	LLLL LLLL	Lower 8 bits of 16-bit counter value
0x00475	FG7 Counter, High Byte	7:0	FCULR	LLLL LLLL	Upper 8 bits of 16-bit counter value
0x00476	FG7 Mode Lower	7:0	F7MSR	0000 0000 0000 0001 0000 0010	Normal operation* (default) Enable timer, clock = internal frame Enable timer, clock = external FG6
0x00477	FG7 Mode Upper	7	F7ISB	0 1	Normal FG7 timer output, high pulses (default) Inverted FG7 timer output, low pulses
		6:4	F7SSP	000 001 010 100	FG7 timer output off (default) FG7 timer output = square wave† FG7 timer output = carry out pulse‡ FG7 timer output = programmable pulse§
		3:0	F7WSN	0001 0010 0100 1000	Programmable pulse width = 30.5 ns Programmable pulse width = 61 ns Programmable pulse width = 91.5 ns Programmable pulse width = 122 ns

* Normal operation allows frame group or FGIO control via registers 0x00470—473. Enabling the counter overrides 0x00470—473 settings.
† Square wave is only available when FG7 Counter high/low value is a binary multiple 1, 2, 4, 8, 16, etc. Other values yield a carry out pulse shape.
‡ Carry Out pulse is active for one FG7 timer clock period.
§ Programmable pulses are based on T8110 internal 32.768 MHz clock periods.

7 Frame Group and FG I/O (continued)

7.3 FGIO Control Registers

7.3.1 FGIO Data Register

The FGIO data register provides read/write access and write storage to/from any FG signals being used as general-purpose register bits. Writes to FGIO work in conjunction with the corresponding FGIO enabled settings in the FGx rate registers. Reads are maskable, controlled via register 0x00481.

Table 70. FGIO Data Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00480	FGIO Data Register	7	F7IOB	L	FGIO bit 7 value
		6	F6IOB	L	FGIO bit 6 value
		5	F5IOB	L	FGIO bit 5 value
		4	F4IOB	L	FGIO bit 4 value
		3	F3IOB	L	FGIO bit 3 value
		2	F2IOB	L	FGIO bit 2 value
		1	F1IOB	L	FGIO bit 1 value
		0	F0IOB	L	FGIO bit 0 value

7.3.2 FGIO Read Mask Register

The FGIO read mask register controls the masking of any FG signals being used as general-purpose register bits on a read access to the FGIO Register.

Table 71. FGIO Read Mask Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00481	FGIO Read Mask	7	F7MEB	0 1	Unmask FGIO bit 7 (default). Mask FGIO bit 7, return 0 on a read.
		6	F6MEB	0 1	Unmask FGIO bit 6 (default). Mask FGIO bit 6, return 0 on a read.
		5	F5MEB	0 1	Unmask FGIO bit 5 (default). Mask FGIO bit 5, return 0 on a read.
		4	F4MEB	0 1	Unmask FGIO bit 4 (default). Mask FGIO bit 4, return 0 on a read.
		3	F3MEB	0 1	Unmask FGIO bit 3 (default). Mask FGIO bit 3, return 0 on a read.
		2	F2MEB	0 1	Unmask FGIO bit 2 (default). Mask FGIO bit 2, return 0 on a read.
		1	F1MEB	0 1	Unmask FGIO bit 1 (default). Mask FGIO bit 1, return 0 on a read.
		0	F0MEB	0 1	Unmask FGIO bit 0 (default). Mask FGIO bit 0, return 0 on a read.

7 Frame Group and FG I/O (continued)

7.3.3 FGIO R/W Register

The FGIO R/W register provides direction control for any of the FG signals being used as general-purpose register bits.

Table 72. FGIO R/W Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00482	FGIO R/W	7	F7DSB	0 1	FGIO bit 7 direction is INPUT (default). FGIO bit 7 direction is OUTPUT.
		6	F6DSB	0 1	FGIO bit 6 direction is INPUT (default). FGIO bit 6 direction is OUTPUT.
		5	F5DSB	0 1	FGIO bit 5 direction is INPUT (default). FGIO bit 5 direction is OUTPUT.
		4	F4DSB	0 1	FGIO bit 4 direction is INPUT (default). FGIO bit 4 direction is OUTPUT.
		3	F3DSB	0 1	FGIO bit 3 direction is INPUT (default). FGIO bit 3 direction is OUTPUT.
		2	F2DSB	0 1	FGIO bit 2 direction is INPUT (default). FGIO bit 2 direction is OUTPUT.
		1	F1DSB	0 1	FGIO bit 1 direction is INPUT (default). FGIO bit 1 direction is OUTPUT.
		0	F0DSB	0 1	FGIO bit 0 direction is INPUT (default). FGIO bit 0 direction is OUTPUT.

7.4 FG Circuit Operation

The eight frame group signals FG[7:0] each operate independently and have multiple uses. Refer to Figure 28 below.

- As programmable 8 kHz frame reference outputs (frame group)
- As general-purpose register I/O bits (FGIO)
- As a programmable timer (FG7 only)
- As external interrupt input signals
- As diagnostic observation points for internal test points

7 Frame Group and FG I/O (continued)

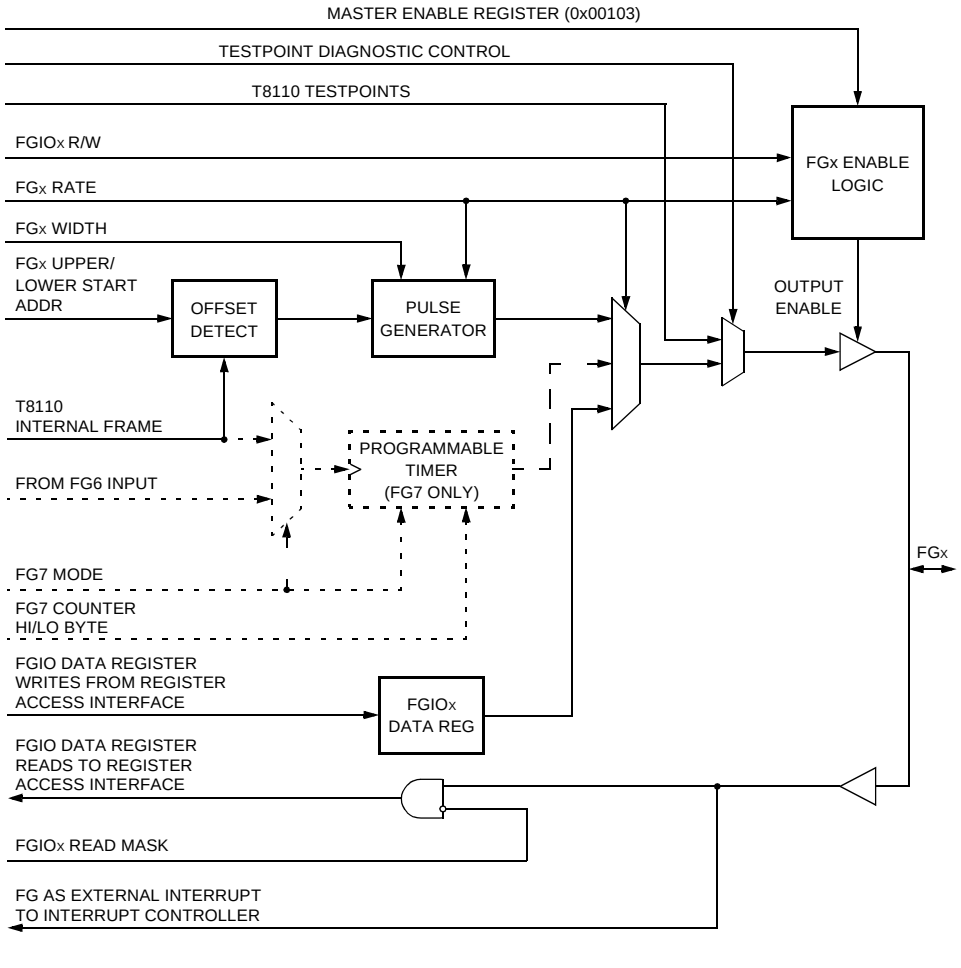


Figure 28. FG[7:0] Functional Paths

7.4.1 Frame Group 8 kHz Reference Generation

Any of the T8110 FG signals may be used as programmable 8 kHz frame reference outputs. There are two sets of control required, an offset delay from internal frame center, and pulse shaping.

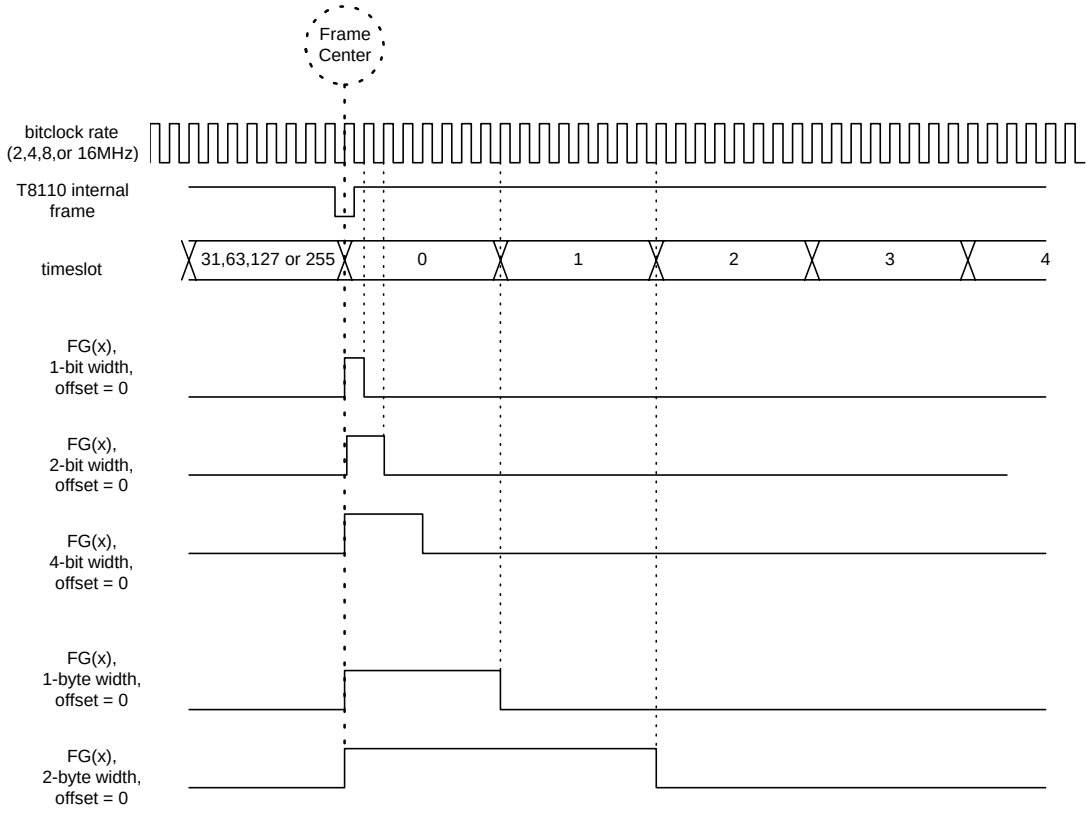
The offset delay is provided via the FGx upper/lower start address registers. The delay is relative to the T8110 internal frame center, and the 12 bits used allow for 4096 different offsets, in increments of one 32.568 MHz clock period (30.5 ns).

Pulse shaping is controlled via the FGx width and FGx rate registers. Pulses may be programmed to be active-high or active-low. Pulse width can be either 1-bit, 2-bit, 4-bit, 1-byte or 2-byte wide (relative to the rate setting*), with allowable rate settings of 2.048 MHz, 4.096 MHz, 8.192 MHz, and 16.384 MHz.

*Pulse widths are bit times or multiples of bit times, for each applicable rate:

RATE	BIT TIME
2.048 Mbits/s	488 ns
4.096 Mbits/s	244 ns
8.192 Mbits/s	122 ns
16.384 Mbits/s	61 ns

7 Frame Group and FG I/O (continued)



NOTES:
1. Frame group signals shown with offset = 0 (default). At offset = 0, the pulse starts at frame center.
2. Non-zero offsets denote 30.568MHz period increments (30.5ns) from frame center. There are up to 4096 increments within an 8KHz frame period. Offsets may be programmed in the range 0 to 4095.
3. Frame group signals are shown as active HI pulses (default) - they may be programmed as active LO pulses.
4. Diagram shows frame group pulse widths relative to bitclock rate and timeslot width. This is applicable for any of the four frame group data rates (2MB/S, 4MB/S, 8MB/S or 16MB/S).

Figure 29. Frame Group 8 kHz Reference Timing

7.4.2 FGIO General-Purpose Bits

Any of the T8110 FG signals may be used as general-purpose I/O bits. Each FG bit used as FGIO is configured by enabling the FGIO function via the FGx rate register(s) and setting the direction via the appropriate bits in the FGIO R/W register. For write access to the FGIO, the FGIO data register is used to hold data for output to the FG pin(s). Read accesses are maskable via the FGIO read mask register. For read access from the FGIO, the logical state of the FG[7:0] signals is returned if unmasked. If an FGIO bit is masked, a read access returns 0.

7.4.3 Programmable Timer (FG7 Only)

The FG7 signal can be used as a programmable timer output, via the FG7 mode upper/lower, and FG7 counter high and low byte registers. The FG7 timer is simply a clock divider. The FG7 counter high/low provides a 16-bit [divider value – 1].

7 Frame Group and FG I/O (continued)

Note: [divider value – 1], i.e., a value of 0000000000000011 yields a div-by-4 operation.

The FG7 mode lower register enables the counter and selects between two clock sources into the counter; either the T8110 internal frame (8 kHz), or an external clock via the FG6 input. The FG7 mode upper register controls the output pulse shape. The output can be inverted or noninverted and shaped as either a square wave, a carryout pulse, or a programmable-width pulse.

- Square wave. This option is applicable only for divide operations that are binary multiples (i.e., div-by-2, div-by-4, div-by-8, div-by-16, div-by-4096). Nonbinary divide operations while square wave is selected result in a carryout pulse.
- Carryout pulse. The output is a pulse, width = one FG7 timer clock period.
- Programmable-width pulse. The timer output is synchronized to the T8110 32.568 MHz clock domain, and can be programmed for 1, 2, 3, or 4 32.768 MHz clock periods in width (30.5 ns, 61 ns, 91.5 ns, or 122 ns).

7.4.4 FG External Interrupts

All FG signals are internally connected as inputs to the interrupt controller logic. Any FG signal, whether an output or an input, may be used to trigger interrupts. When a T8110 FG signal is used as an externally sourced input into the interrupt controller logic, it must be in input mode (i.e., shut OFF, FGx rate register(s) FxRSR = 0000 0000). An FG signal in output mode may also be used for interrupts (i.e., an 8 kHz periodic signal, Section 7.4.1 on page 95). The interrupt control registers (0x00600—603) control how the FG inputs are handled (for more details, refer to Section 11.1 on page 112).

7.4.5 FG Diagnostic Test Point Observation

Any of the T8110 FG signals may be used to observe a predefined set of internal test points. Each FG bit used as a test point output is enabled via diagnostic register 0x00140, FG testpoint enable. Settings in this register override the FGx rate and FGIO R/W register, and force the selected bits to be testpoint outputs, see Section 12.1 on page 127 and Table 101 on page 128.

8 General-Purpose I/O

There are eight independent T8110 GPIO signals, GP[7:0]. These pins behave as general-purpose register bits, with programmable direction (in or out) and read masking. The GP0 and GP1 signals allow for an additional mode of operation, providing dedicated output signals to indicate A-clock and B-clock mastering for H.110 bus applications.

8.1 GPIO Control Registers

Table 73. GPIO Register

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00500	GPIO Override	GPIO R/W	GPIO Read Mask	GPIO Data Register

8.1.1 GPIO Data Register

The GPIO data register provides read/write access and write storage to/from any GP signals being used as general-purpose register bits. Reads from GPIO are maskable, controlled via register 0x00501.

Table 74. GPIO Data Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00500	GPIO Data Register	7	G7IOB	L	GPIO bit 7 value
		6	G6IOB	L	GPIO bit 6 value
		5	G5IOB	L	GPIO bit 5 value
		4	G4IOB	L	GPIO bit 4 value
		3	G3IOB	L	GPIO bit 3 value
		2	G2IOB	L	GPIO bit 2 value
		1	G1IOB	L	GPIO bit 1 value
		0	G0IOB	L	GPIO bit 0 value

8 General-Purpose I/O (continued)

8.1.2 GPIO Read Mask Register

The GPIO read mask register controls the masking of any GP signals being used as general-purpose register bits on a read access to the GPIO register.

Table 75. GPIO Read Mask Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00501	GPIO Read Mask	7	G7MEB	0 1	Unmask GPIO bit 7 (default). Mask GPIO bit 7, return 0 on a read.
		6	G6MEB	0 1	Unmask GPIO bit 6 (default). Mask GPIO bit 6, return 0 on a read.
		5	G5MEB	0 1	Unmask GPIO bit 5 (default). Mask GPIO bit 5, return 0 on a read.
		4	G4MEB	0 1	Unmask GPIO bit 4 (default). Mask GPIO bit 4, return 0 on a read.
		3	G3MEB	0 1	Unmask GPIO bit 3 (default). Mask GPIO bit 3, return 0 on a read.
		2	G2MEB	0 1	Unmask GPIO bit 2 (default). Mask GPIO bit 2, return 0 on a read.
		1	G1MEB	0 1	Unmask GPIO bit 1 (default). Mask GPIO bit 1, return 0 on a read.
		0	G0MEB	0 1	Unmask GPIO bit 0 (default). Mask GPIO bit 0, return 0 on a read.

8 General-Purpose I/O (continued)

8.1.3 GPIO R/W Register

The GPIO R/W Register provides direction control for any of the GP signals being used as general-purpose register bits.

Table 76. GPIO R/W Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00502	GPIO R/W	7	G7DSB	0 1	GPIO bit 7 direction is input (default). GPIO bit 7 direction is output.
		6	G6DSB	0 1	GPIO bit 6 direction is input (default). GPIO bit 6 direction is output.
		5	G5DSB	0 1	GPIO bit 5 direction is input (default). GPIO bit 5 direction is output.
		4	G4DSB	0 1	GPIO bit 4 direction is input (default). GPIO bit 4 direction is output.
		3	G3DSB	0 1	GPIO bit 3 direction is input (default). GPIO bit 3 direction is output.
		2	G2DSB	0 1	GPIO bit 2 direction is input (default). GPIO bit 2 direction is output.
		1	G1DSB	0 1	GPIO bit 1 direction is input (default). GPIO bit 1 direction is output.
		0	G0DSB	0 1	GPIO bit 0 direction is input (default). GPIO bit 0 direction is output.

8.1.4 GPIO Override Register

Table 77. GPIO Override Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00503	GPIO override	7:3	Reserved	0000 0000	NOP (default)
		2	G2OEB	0 1	GPIO bit 2 is GPIO (default). GPIO bit 2 is PCI_RST# indicator.
		1	G1OEB	0 1	GPIO bit 1 is GPIO (default). GPIO bit 1 B-master indicator output.
		0	G0OEB	0 1	GPIO bit 0 is GPIO (default). GPIO bit 0 A-master indicator output.

8 General-Purpose I/O (continued)

8.2 GP Circuit Operation

The eight general-purpose I/O group signals GP[7:0] each operate independently and have multiple uses. Please refer to Figure 30 on page 101.

- As general-purpose register I/O bits (GPIO)
- As H.110 bus clock master indicators (GP0, GP1 only)
- As external interrupt input signals
- As diagnostic observation points for internal test points

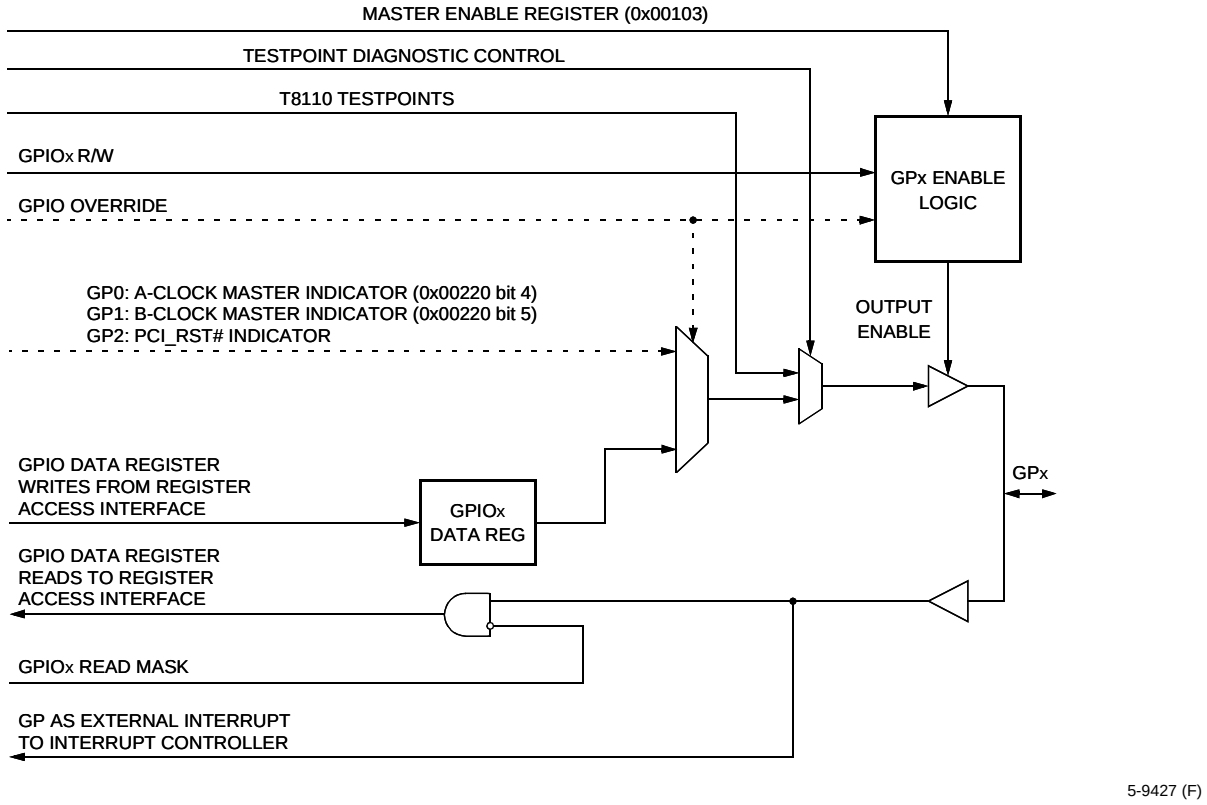


Figure 30. GP[7:0] Functional Paths

8.2.1 GPIO General-Purpose Bits

Any of the T8110 GP signals may be used as general-purpose I/O bits. Each GP bit used as GPIO is configured by setting the direction via the appropriate bits in the GPIO R/W register. For write access to the GPIO, the GPIO data register is used to hold data for output to the GP pin(s). Read accesses are maskable via the GPIO read mask register. For read access from the GPIO, the logical state of the GP[7:0] signals is returned if unmasked. If an GPIO bit is masked, a read access returns 0.

8 General-Purpose I/O (continued)

8.2.2 GP Dual-Purpose Bits GPIO (Override)

8.2.2.1 GP H.110 Clock Master Indicators (GP0, GP1 Only)

An additional function is provided for GP0 and GP1 only, controlled via the GPIO override register.

GP0 may be used as a dedicated output (set GPIO override register bit 0), which transmits the state of the T8110 A-clock master enable (register 0x00220, bit 4). This output is intended to drive the external A-clock FETs required for H.110 bus mastering.

GP1 may be used as a dedicated output (set GPIO override register bit 1), which transmits the state of the T8110 B-clock master enable (register 0x00220, bit 5). This output is intended to drive the external B-clock FETs required for H.110 bus mastering.

8.2.2.2 PCI_RST# Indicator (GP2 Only)

An additional function is provided for GP2 only, controlled via the GPIO override register. GP2 may be used as a dedicated output (set GPIO override register bit 2), which forwards the state of the PCI_RST# signal. Polarity of the transmitted signal is selectable via register 0x00780 (refer to Section 10.2 on page 109). This function provides access to a forwarded PCI_RST# signal by external devices hanging off the minibridge port.

8.2.3 GP External Interrupts

Any of the T8110 GP signals may be used as externally sourced inputs into the interrupt controller logic. Each GP bit used as an interrupt input must be shut off by setting the appropriate GPIO R/W register bit to be input. The interrupt control registers (0x00604—607) control how the GP inputs are handled. For more details, see Section 11.1 on page 112.

8.2.4 GP Diagnostic Test Point Observation

Any of the T8110 GP signals may be used to observe a predefined set of internal test points. Each GP bit used as a test point output is enabled via diagnostic register 0x00142, GP testpoint enable. Settings in this register override the GPIO R/W register, and force the selected bits to be testpoint outputs (refer to Section 12.1 on page 127, and Table 103 on page 128).

9 Stream Rate Control

There are a total of 64 data streams, divided into 16 stream groups of four streams each, as shown below.

Table 78. T8110 Serial Stream Groupings

Stream Group	Stream Bits
H-bus group A	CT_D[0:3]
H-bus group B	CT_D[4:7]
H-bus group C	CT_D[8:11]
H-bus group D	CT_D[12:15]
H-bus group E	CT_D[16:19]
H-bus group F	CT_D[20:23]
H-bus group G	CT_D[24:27]
H-bus group H	CT_D[28:31]
L-bus group A	L_D[0:3]
L-bus group B	L_D[4:7]
L-bus group C	L_D[8:11]
L-bus group D	L_D[12:15]
L-bus group E	L_D[16:19]
L-bus group F	L_D[20:23]
L-bus group G	L_D[24:27]
L-bus group H	L_D[28:31]

The H-bus group operational frequencies are selectable between 2.048 MHz, 4.096 MHz, and 8.192 MHz. The L-bus groups may operate at 2.048 MHz, 4.096 MHz, 8.192 MHz, and 16.384 MHz, which is implemented as multiplexed 8.192 MHz streams. For more details, see Section 9.2.2 on page 104).

9.1 H-Bus Stream Rate Control Registers

9.1.1 H-Bus Rate Registers

The H-bus rate registers control the serial data stream rate of operation for each of the H-bus stream groups, A—H. The upper nibble controls groups B, D, F, and H. The lower nibble controls groups A, C, E, and G.

A—H. The upper nibble controls groups B, D, F, and H. The lower nibble controls groups A, C, E, and G.

Table 79. H-Bus Rate Registers

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00300	H-Bus Rate H/G	H-Bus Rate F/E	H-Bus Rate D/C	H-Bus Rate B/A

9 Stream Rate Control (continued)

Table 79. H-Bus Rate Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00300	H-Bus Rate B/A	7:4	HBRSN	0000	H-bus Group B(D, F, H) off (default)
(0x00301)	(H-Bus Rate D/C)		(HDRSN)	0010	H-bus Group B(D, F, H) rate = 2.048 MHz
(0x00302)	(H-Bus Rate F/E)		(HFRSN)	0100	H-bus Group B(D, F, H) rate = 4.096 MHz
(0x00303)	(H-Bus Rate H/G)		(HHRSN)	1000	H-bus Group B(D, F, H) rate = 8.192 MHz
		3:0	HARSN	0000	H-bus Group A(C, E, G) off (default)
			(HCRSN)	0010	H-bus Group A(C, E, G) rate = 2.048 MHz
			(HERSN)	0100	H-bus Group A(C, E, G) rate = 4.096 MHz
			(HGRSN)	1000	H-bus Group A(C, E, G) rate = 8.192 MHz

9.2 L-Bus Stream Rate Control Registers

9.2.1 L-Bus Rate Registers

The L-bus rate registers control the serial data stream rate of operation for each of the H-bus stream groups, A—H. The upper nibble controls groups B, D, F, and H. The lower nibble controls groups A, C, E, and G. Local streams have a 16.384 MHz rate option (refer to Section 9.2.2 on page 104).

Table 80. L-Bus Rate Registers

DWORD Address (20 Bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00320	L-Bus Rate H/G	L-Bus Rate F/E	L-Bus Rate D/C	L-Bus Rate B/A

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00320	L-Bus Rate B/A	7:4	LBRSN	0000	L-bus Group B(D, F, H) off (default)
(0x00321)	(L-Bus Rate D/C)		(LDRSN)	0010	L-bus Group B(D, F, H) rate = 2.048 MHz
(0x00322)	(L-Bus Rate F/E)		(LFRSN)	0100	L-bus Group B(D, F, H) rate = 4.096 MHz
(0x00323)	(L-Bus Rate H/G)		(LHRSN)	1000	L-bus Group B(D, F, H) rate = 8.192 MHz
				1001	L-bus Group B(D, F, H) rate = 16.384 MHz
		3:0	LARSN	0000	L-bus Group A(C, E, G) off (default)
			(LCRSN)	0010	L-bus Group A(C, E, G) rate = 2.048 MHz
			(LERSN)	0100	L-bus Group A(C, E, G) rate = 4.096 MHz
			(LGRSN)	1000	L-bus Group A(C, E, G) rate = 8.192 MHz
				1001	L-bus Group A(C, E, G) rate = 16.384 MHz

9.2.2 L-Bus 16.384 Mbits/s Operation

Local stream 16.384 Mbits/s operation is implemented as two multiplexed 8.192 Mbits/s streams. Bits are shifted at 16.384 MHz, and 16 bits are shifted per 8.192 Mbits/s time slot (refer to Figure 31). This operation makes use of adjacent pairs of the existing single-byte hold and shift registers for local stream operation, with the local even stream assigned as the incoming stream, and the local odd stream assigned as the outgoing stream. Pairs are assigned as LD[0,1], LD[2,3], … LD[30,31]. When an L-bus group is set to operate at rate of 16.384 Mbits/s, the hold and shift circuitry is configured such that the serial output of the even stream shift register feeds the serial input of the odd stream shift register (refer to Figure 32).

9 Stream Rate Control (continued)

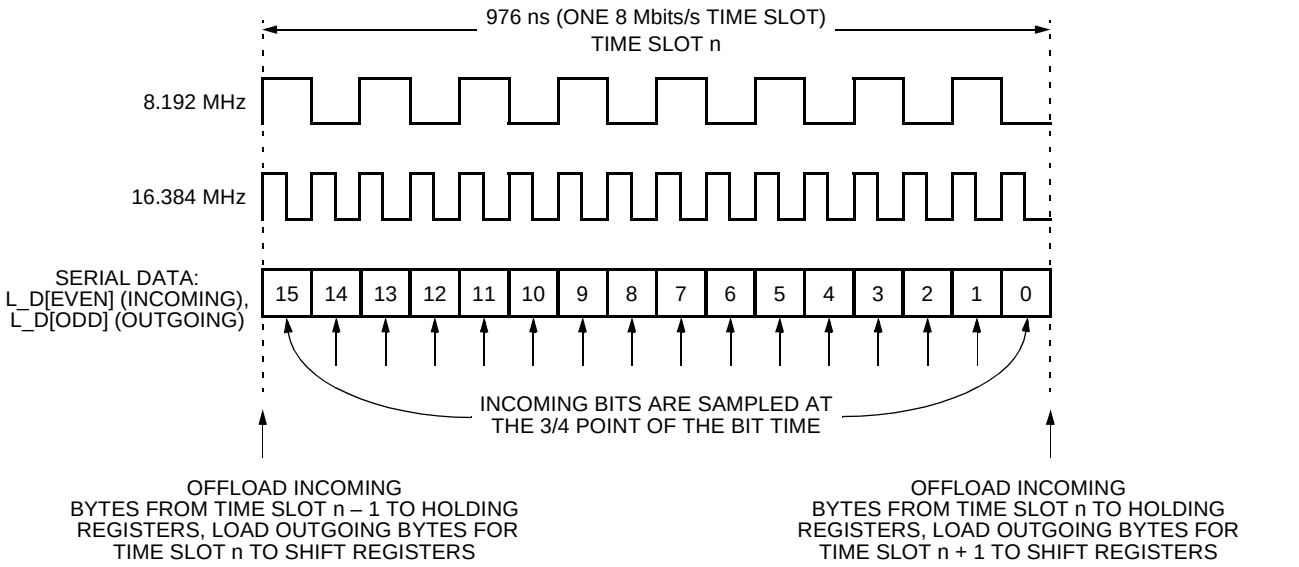


Figure 31. Local Stream 16.384 Mbits/s Timing

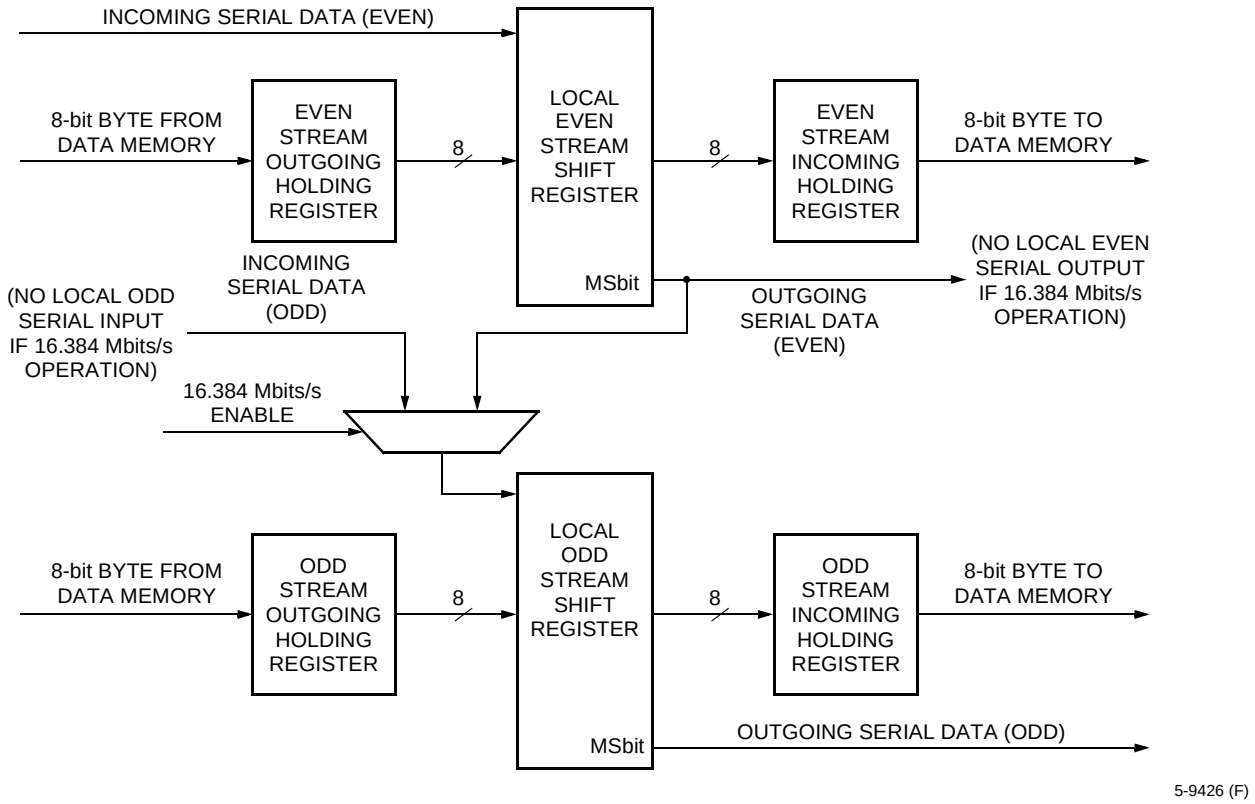


Figure 32. Local Stream 16.384 Mbits/s Circuit

10 Minibridge

The T8110 provides for access to non-PCI devices from the PCI bus via the minibridge port.

Note: If the T8110 is configured to interface to an ISA bus, the minibridge function is not applicable, and the minibridge port pins are used for the ISA interface. Refer to Section 4.2 on page 39.

PCI access requests are converted to a simple interface to external devices hanging on the minibridge port. There are eight chip select outputs, a read strobe, a write strobe, 16-bit address, and 16-bit data bus. Additionally, a forwarded version of the PCI_RST# signal can be made available at the GP(2) output, refer to Section 5.1.1 on page 48 and Section 8.2.2.2 on page 102.

10.1 Wait-State Control Registers

10.1.1 Minibridge Wait-State Control Registers

The minibridge wait-state control registers allow for programmable assertion times for MB_CS[7:0], MB_RD, and MB_WR control outputs. Resolution for the wait-state value increments is one 65.538 MHz clock period (15.25 ns), refer to Figure 31 on page 105.

Table 81. Minibridge Wait-State Control Register Map

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00700	CS0 addr setup wait	CS0 read hold wait	CS0 read width wait	CS0 read setup wait
0x00704	CS0 addr hold wait	CS0 write hold wait	CS0 write width wait	CS0 write setup wait
0x00710	CS1 addr setup wait	CS1 read hold wait	CS1 read width wait	CS1 read setup wait
0x00714	CS1 addr hold wait	CS1 write hold wait	CS1 write width wait	CS1 write setup wait
0x00720	CS2 addr setup wait	CS2 read hold wait	CS2 read width wait	CS2 read setup wait
0x00724	CS2 addr hold wait	CS2 write hold wait	CS2 write width wait	CS2 write setup wait
0x00730	CS3 addr setup wait	CS3 read hold wait	CS3 read width wait	CS3 read setup wait
0x00734	CS3 addr hold wait	CS3 write hold wait	CS3 write width wait	CS3 write setup wait
0x00740	CS4 addr setup wait	CS4 read hold wait	CS4 read width wait	CS4 read setup wait
0x00744	CS4 addr hold wait	CS4 write hold wait	CS4 write width wait	CS4 write setup wait
0x00750	CS5 addr setup wait	CS5 read hold wait	CS5 read width wait	CS5 read setup wait
0x00754	CS5 addr hold wait	CS5 write hold wait	CS5 write width wait	CS5 write setup wait
0x00760	CS6 addr setup wait	CS6 read hold wait	CS6 read width wait	CS6 read setup wait
0x00764	CS6 addr hold wait	CS6 write hold wait	CS6 write width wait	CS6 write setup wait
0x00770	CS7 addr setup wait	CS7 read hold wait	CS7 read width wait	CS7 read setup wait
0x00774	CS7 addr hold wait	CS7 write hold wait	CS7 write width wait	CS7 write setup wait

10 Minibridge (continued)

Table 82. Minibridge Wait-State Control Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00700 (0x00710) (0x00720) (0x00730) (0x00740) (0x00750) (0x00760) (0x00770)	CS0 read setup wait (CS1 read setup wait) (CS2 read setup wait) (CS3 read setup wait) (CS4 read setup wait) (CS5 read setup wait) (CS6 read setup wait) (CS7 read setup wait)	7:0	R0SLR (R1SLR) (R2SLR) (R3SLR) (R4SLR) (R5SLR) (R6SLR) (R7SLR)	LLLL LLLL	Read cycle wait-state value, delay to leading edge of MB_RD
0x00701 (0x00711) (0x00721) (0x00731) (0x00741) (0x00751) (0x00761) (0x00771)	CS0 read width wait (CS1 read width wait) (CS2 read width wait) (CS3 read width wait) (CS4 read width wait) (CS5 read width wait) (CS6 read width wait) (CS7 read width wait)	7:0	R0WLR (R1WLR) (R2WLR) (R3WLR) (R4WLR) (R5WLR) (R6WLR) (R7WLR)	LLLL LLLL	Read cycle wait-state value, assertion time for MB_RD
0x00702 (0x00712) (0x00722) (0x00732) (0x00742) (0x00752) (0x00762) (0x00772)	CS0 read hold wait (CS1 read hold wait) (CS2 read hold wait) (CS3 read hold wait) (CS4 read hold wait) (CS5 read hold wait) (CS6 read hold wait) (CS7 read hold wait)	7:0	R0HLR (R1HLR) (R2HLR) (R3HLR) (R4HLR) (R5HLR) (R6HLR) (R7HLR)	LLLL LLLL	Read cycle wait-state value, delay to deassertion of MB_CS0 (1,2,3,4,5,6,7)
0x00703 (0x00713) (0x00723) (0x00733) (0x00743) (0x00753) (0x00763) (0x00773)	CS0 addr setup wait (CS1 addr setup wait) (CS2 addr setup wait) (CS3 addr setup wait) (CS4 addr setup wait) (CS5 addr setup wait) (CS6 addr setup wait) (CS7 addr setup wait)	7:0	A0SLR (A1SLR) (A2SLR) (A3SLR) (A4SLR) (A5SLR) (A6SLR) (A7SLR)	LLLL LLLL	Any cycle wait-state value, delay from beginning of cycle to assertion of MB_CS0 (1,2,3,4,5,6,7)
0x00704 (0x00714) (0x00724) (0x00734) (0x00744) (0x00754) (0x00764) (0x00774)	CS0 write setup wait (CS1 write setup wait) (CS2 write setup wait) (CS3 write setup wait) (CS4 write setup wait) (CS5 write setup wait) (CS6 write setup wait) (CS7 write setup wait)	7:0	W0SLR (W1SLR) (W2SLR) (W3SLR) (W4SLR) (W5SLR) (W6SLR) (W7SLR)	LLLL LLLL	Write cycle wait-state value, delay to leading edge of MB_WR

10 Minibridge (continued)

Table 82. Minibridge Wait-State Control Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00705 (0x00715) (0x00725) (0x00735) (0x00745) (0x00755) (0x00765) (0x00775)	CS0 write width wait (CS1 write width wait) (CS2 write width wait) (CS3 write width wait) (CS4 write width wait) (CS5 write width wait) (CS6 write width wait) (CS7 write width wait)	7:0	W0WLR (W1WLR) (W2WLR) (W3WLR) (W4WLR) (W5WLR) (W6WLR) (W7WLR)	LLLL LLLL	Write cycle wait-state value, assertion time for MB_WR
0x00706 (0x00716) (0x00726) (0x00736) (0x00746) (0x00756) (0x00766) (0x00776)	CS0 write hold wait (CS1 write hold wait) (CS2 write hold wait) (CS3 write hold wait) (CS4 write hold wait) (CS5 write hold wait) (CS6 write hold wait) (CS7 write hold wait)	7:0	W0HLR (W1HLR) (W2HLR) (W3HLR) (W4HLR) (W5HLR) (W6HLR) (W7HLR)	LLLL LLLL	Write cycle wait-state value, delay to deassertion of MB_CS0 (1, 2, 3, 4, 5, 6, 7)
0x00707 (0x00717) (0x00727) (0x00737) (0x00747) (0x00757) (0x00767) (0x00777)	CS0 addr hold wait (CS1 addr hold wait) (CS2 addr hold wait) (CS3 addr hold wait) (CS4 addr hold wait) (CS5 addr hold wait) (CS6 addr hold wait) (CS7 addr hold wait)	7:0	A0HLR (A1HLR) (A2HLR) (A3HLR) (A4HLR) (A5HLR) (A6HLR) (A7HLR)	LLLL LLLL	Any cycle wait-state value, delay from deassertion of MB_CS0 (1, 2, 3, 4, 5, 6, 7) to end of cycle

10 Minibridge (continued)

10.2 Strobe Control Registers

The CS strobe inversion and RD-WR strobe inversion registers allow for programmable polarity of the MB_CS[7:0], MB_RD and MB_WR minibridge control strobes. Additionally, the polarity of the forwarded PCI_RST# signal (to the GP(2) output) is selectable.

Table 83. Strobe Control Registers

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00780	Reserved	Reserved	rd-wr strobe inversion	cs strobe inversion

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00780	cs Strobe Inversion	7	IC7SB	0 1	CS7 strobe is active-high (default). CS7 strobe is active-low.
		6	IC6SB	0 1	CS6 strobe is active-high (default). CS6 strobe is active-low.
		5	IC5SB	0 1	CS5 strobe is active-high (default). CS5 strobe is active-low.
		4	IC4SB	0 1	CS4 strobe is active-high (default). CS4 strobe is active-low.
		3	IC3SB	0 1	CS3 strobe is active-high (default). CS3 strobe is active-low.
		2	IC2SB	0 1	CS2 strobe is active-high (default). CS2 strobe is active-low.
		1	IC1SB	0 1	CS1 strobe is active-high (default). CS1 strobe is active-low.
		0	IC0SB	0 1	CS0 strobe is active-high (default). CS0 strobe is active-low.
0x00781	rd-wr Strobe Inversion	7:3	Reserved	0000 00	NOP (default)
		2	IPRSB	0 1	Forward direct PCI_RST# (default). Forward inverted PCI_RST#.
		1	IMRSB	0 1	MB_RD strobe is active-high (default). MB_RD strobe is active-low.
		0	IMWSB	0 1	MB_WR strobe is active-high (default). MB_WR strobe is active-low.

10.3 Minibridge Circuit Operation

The minibridge circuit accepts PCI memory read and memory write transactions and translates them into simple asynchronous* control strobes for external devices hanging off the minibridge port. Refer to Section 4.1.1.6 on page 34 for more detail on the PCI side of the transactions. Refer to Figure 33 for the access cycle descriptions of the minibridge side of the transactions.

* Asynchronous relative to the PCI clock. Strobes are generated relative to the internal chip clock, in multiples of 65.536 MHz clock periods.

10 Minibridge (continued)

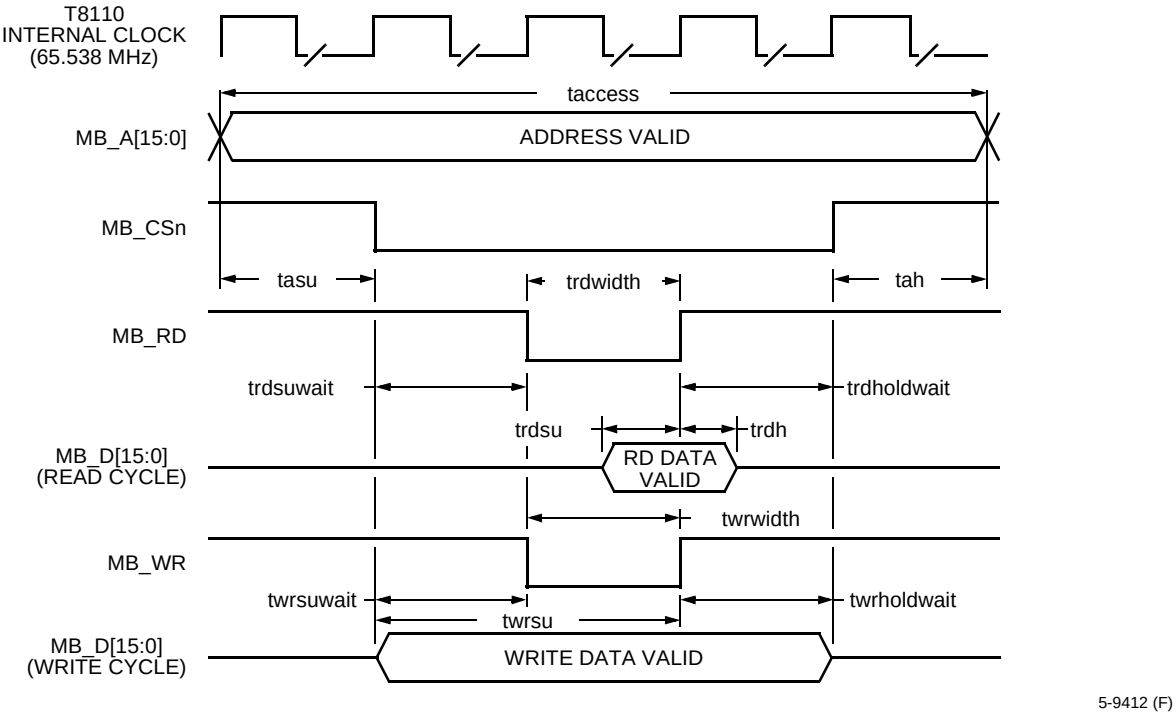


Figure 33. Minibridge Read/Write Access Cycles

Notes:

- Strobes are created based on the T8110 internal 65.536 MHz clock. MB_CS, MB_RD, and MB_WR are shown here as active-low, but may be programmed active-high via the CS strobe inversion and RD-WR strobe inversion registers.
- User-programmable wait-states are allowed at five points for either read or write cycles:
  - — Delay from valid address to MB_CSn assertion (address wait)
  - — Delay from MB_CSn assertion to the leading edge of the MB_RD (or MB_WR) strobe (setup wait)
  - — Pulse width of the MB_RD (or MB_WR) strobe (width wait)
  - — Delay from MB_RD (or MB_WR) trailing edge to the deassertion of MB_CSn (hold wait)
  - — Delay from deassertion of MB_CSn to address invalid (address wait)
- With no wait-states, the minimum access time (read or write) for the minibridge interface is 76.3 ns (five 65.536 MHz clock cycles).

10 Minibridge (continued)

Notes:

- Timing protocol. Any user-programmable wait-states are defined in increments of 65.536 MHz clock periods (15.25 ns).
  - taccess: total access time. Minimum = 76.3 ns (five clock cycles), maximum = 19.5 μs (accumulation of user-programmed wait-states).
  - tasu: address setup to MB_CSn active. Minimum = 15.25 ns (one clock cycle), maximum = 3.9 μs (256 clock cycles) user-programmable via CSn address wait register.
  - tah: address hold from MB_CSn inactive. Minimum = 15.25 ns (one clock cycle), maximum = 3.9 μs (256 clock cycles) user-programmable via CSn address wait register.
  - trdsuwait: delay from MB_CSn active to leading edge of MB_RD strobe. Minimum = 15.25 ns (one clock cycle), maximum = 3.9 μs (256 clock cycles) user-programmable via CSn RD setup wait register
  - trdwidth: MB_RD strobe pulse width. Minimum = 15.25 ns (one clock cycle), maximum = 3.9 μs (256 clock cycles) user-programmable via CSn RD width wait register.
  - trdholdwait: delay from MB_RD strobe trailing edge to MB_CSn inactive. Minimum = 15.25 ns (one clock cycle), maximum = 3.9 μs (256 clock cycles) user-programmable via CSn RD hold wait register.
  - trdsu: read cycle data setup to trailing edge RDn. Minimum = 10 ns.
  - trdh: read cycle data hold from trailing edge RDn. Minimum = 0 ns.
  - twrsuwait: delay from MB_CSn active to leading edge of MB_WR strobe. Minimum = 15.25 ns (one clock cycle), maximum = 3.9 μs (256 clock cycles) user-programmable via CSn WR setup wait register.
  - twrwidth: MB_WR strobe pulse width. Minimum = 15.25 ns (one clock cycle), maximum = 3.9 μs (256 clock cycles) user-programmable via CSn WR width wait register.
  - twrholdwait: delay from MB_WR strobe trailing edge to MB_CSn inactive. Minimum = 15.25 ns (one clock cycle), maximum = 3.9 μs (256 clock cycles) user-programmable via CSn WR hold wait register.
  - twrsu: write cycle data setup to trailing edge of MB_WR. Minimum = 30.5 ns (two clock cycles).

11 Error Reporting and Interrupt Control

11.1 Interrupt Control Registers

Table 84. Interrupt Control Register Map

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00600	FGIO Polarity	Reserved	FGIO Interrupt Enable	FGIO Interrupt Pending
0x00604	GPIO Polarity	Reserved	GPIO Interrupt Enable	GPIO Interrupt Pending
0x00608	System Interrupt Enable High	System Interrupt Enable Low	System Interrupt Pending High	System Interrupt Pending Low
0x0060C	Clock Interrupt Enable High	Clock Interrupt Enable Low	Clock Interrupt Pending High	Clock Interrupt Pending Low
0x00610	CLKERR Output Select	SYSERR Output Select	PCI_INTA Output Select	Arbitration Control
0x00614	CLKERR Pulse Width	SYSERR Pulse Width	Reserved	Reserved
0x00618	In-Service, Byte 3	In-Service, Byte 2	In-Service, Byte 1	In-Service, Byte 0

11.1.1 Interrupts via External FG[7:0] Registers

11.1.1.1 FGIO Interrupt Pending Register

The FGIO interrupt pending register stores detected interrupts via the FG[7:0] signals. User can clear specific pending bits by writing 1 to that bit (write-1 to clear). Interrupts via these signals are maskable via the FGIO interrupt enable register.

Table 85. FGIO Interrupt Pending Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00600	FGIO Interrupt Pending	7	JF7OB	0 1	No pending interrupts via FG7 (default) Pending interrupt via FG7
		6	JF6OB	0 1	No pending interrupts via FG6 (default) Pending interrupt via FG6
		5	JF5OB	0 1	No pending interrupts via FG5 (default) Pending interrupt via FG5
		4	JF4OB	0 1	No pending interrupts via FG4 (default) Pending interrupt via FG4
		3	JF3OB	0 1	No pending interrupts via FG3 (default) Pending interrupt via FG3
		2	JF2OB	0 1	No pending interrupts via FG2 (default) Pending interrupt via FG2
		1	JF1OB	0 1	No pending interrupts via FG1 (default) Pending interrupt via FG1
		0	JF0OB	0 1	No pending interrupts via FG0 (default) Pending interrupt via FG0

11 Error Reporting and Interrupt Control (continued)

Table 85. FGIO Interrupt Pending Registers (continued)

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00601	FGIO Interrupt Enable	7	JF7EB	0 1	Disable (mask) interrupts via FG7 (default). Enable (unmask) interrupts via FG7.
		6	JF6EB	0 1	Disable (mask) interrupts via FG6 (default). Enable (unmask) interrupts via FG6.
		5	JF5EB	0 1	Disable (mask) interrupts via FG5 (default). Enable (unmask) interrupts via FG5.
		4	JF4EB	0 1	Disable (mask) interrupts via FG4 (default). Enable (unmask) interrupts via FG4.
		3	JF3EB	0 1	Disable (mask) interrupts via FG3 (default). Enable (unmask) interrupts via FG3.
		2	JF2EB	0 1	Disable (mask) interrupts via FG2 (default). Enable (unmask) interrupts via FG2.
		1	JF1EB	0 1	Disable (mask) interrupts via FG1 (default). Enable (unmask) interrupts via FG1.
		0	JF0EB	0 1	Disable (mask) interrupts via FG0 (default). Enable (unmask) interrupts via FG0.

The GPIO edge/level and GPIO polarity registers control how interrupts are interpreted on the GP[7:0] signals (negative edge, positive edge, low level or high level).

Table 86. FGIO Edge/Level and Polarity Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00603	FGIO Polarity	7	IF7SB	0 1	FG7 interrupts are negative edge or low level (default) FG7 interrupts are positive edge or high level
		6	IF6SB	0 1	FG6 interrupts are negative edge or low level (default) FG6 interrupts are positive edge or high level
		5	IF5SB	0 1	FG5 interrupts are negative edge or low level (default) FG5 interrupts are positive edge or high level
		4	IF4SB	0 1	FG4 interrupts are negative edge or low level (default) FG4 interrupts are positive edge or high level
		3	IF3SB	0 1	FG3 interrupts are negative edge or low level (default) FG3 interrupts are positive edge or high level
		2	IF2SB	0 1	FG2 interrupts are negative edge or low level (default) FG2 interrupts are positive edge or high level
		1	IF1SB	0 1	FG1 interrupts are negative edge or low level (default) FG1 interrupts are positive edge or high level
		0	IF0SB	0 1	FG0 interrupts are negative edge or low level (default) FG0 interrupts are positive edge or high level

11 Error Reporting and Interrupt Control (continued)

11.1.2 Interrupts via External GP[7:0]

11.1.2.1 GPIO Interrupt Pending Register

The GPIO interrupt pending register stores detected interrupts via the GP[7:0] signals. User can clear specific pending bits by writing 1 to that bit (write-1-to-clear). Interrupts via these signals are maskable via the GPIO interrupt enable register.

Table 87. GPIO Interrupt Pending Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00604	GPIO Interrupt Pending	7	JG7OB	0 1	No pending interrupts via GP7 (default) Pending interrupt via GP7
		6	JG6OB	0 1	No pending interrupts via GP6 (default) Pending interrupt via GP6
		5	JG5OB	0 1	No pending interrupts via GP5 (default) Pending interrupt via GP5
		4	JG4OB	0 1	No pending interrupts via GP4 (default) Pending interrupt via GP4
		3	JG3OB	0 1	No pending interrupts via GP3 (default) Pending interrupt via GP3
		2	JG2OB	0 1	No pending interrupts via GP2 (default) Pending interrupt via GP2
		1	JG1OB	0 1	No pending interrupts via GP1 (default) Pending interrupt via GP1
		0	JG0OB	0 1	No pending interrupts via GP0 (default) Pending interrupt via GP0
0x00605	GPIO Interrupt Enable	7	JG7EB	0 1	Disable (mask) interrupts via GP7 (default) Enable (unmask) interrupts via GP7
		6	JG6EB	0 1	Disable (mask) interrupts via GP6 (default) Enable (unmask) interrupts via GP6
		5	JG5EB	0 1	Disable (mask) interrupts via GP5 (default) Enable (unmask) interrupts via GP5
		4	JG4EB	0 1	Disable (mask) interrupts via GP4 (default) Enable (unmask) interrupts via GP4
		3	JG3EB	0 1	Disable (mask) interrupts via GP3 (default) Enable (unmask) interrupts via GP3
		2	JG2EB	0 1	Disable (mask) interrupts via GP2 (default) Enable (unmask) interrupts via GP2
		1	JG1EB	0 1	Disable (mask) interrupts via GP1 (default) Enable (unmask) interrupts via GP1
		0	JG0EB	0 1	Disable (mask) interrupts via GP0 (default) Enable (unmask) interrupts via GP0

11 Error Reporting and Interrupt Control (continued)

11.1.2.2 GPIO Edge/Level and GPIO Polarity Registers

The GPIO edge/level and GPIO polarity registers control how interrupts are interpreted on the GP[7:0] signals (negative edge, positive edge, low level or high level).

Table 88. GPIO Edge/Level and GPIO Polarity Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00607	GPIO Polarity	7	IG7SB	0 1	GP7 interrupts are negative edge or low level (default) GP7 interrupts are positive edge or high level
		6	IG6SB	0 1	GP6 interrupts are negative edge or low level (default) GP6 interrupts are positive edge or high level
		5	IG5SB	0 1	GP5 interrupts are negative edge or low level (default) GP5 interrupts are positive edge or high level
		4	IG4SB	0 1	GP4 interrupts are negative edge or low level (default) GP4 interrupts are positive edge or high level
		3	IG3SB	0 1	GP3 interrupts are negative edge or low level (default) GP3 interrupts are positive edge or high level
		2	IG2SB	0 1	GP2 interrupts are negative edge or low level (default) GP2 interrupts are positive edge or high level
		1	IG1SB	0 1	GP1 interrupts are negative edge or low level (default) GP1 interrupts are positive edge or high level
		0	IF0SB	0 1	GP0 interrupts are negative edge or low level (default) GP0 interrupts are positive edge or high level

11.1.3 Interrupts via Internal System Errors

Table 89. System Error Interrupt Assignments

System Interrupt Bit	Description
SYS15	Clock Failsafe Indicator
SYS14	Clock Fallback Indicator
SYS13	PCI Target, Minibridge Discard Timer Expired
SYS12	PCI Target, VC Memory Discard Timer Expired
SYS11	PCI Target, DATA Memory Discard Timer Expired
SYS10	PCI Target, Minibridge Protocol Error
SYS9	PCI Target, VC Memory Protocol Error
SYS8	PCI Target, DATA Memory Protocol Error
SYS7	PCI Master, PCI Bus Fatal Error
SYS6	PCI Master, External Buffer LOCK Error
SYS5	PCI Master, External Buffer STALL Error
SYS4	PCI Master, External Buffer STALL Warning
SYS3	PCI Master, External Buffer OVWRITE Warning
SYS2	PCI Master, External Buffer INIT Warning
SYS1	VC Memory, Scratchpad OVFLOW Warning
SYS0	NOTIFY Queue, OVFLOW Warning

11 Error Reporting and Interrupt Control (continued)

11.1.4 System Interrupt Pending High/Low Registers

The system interrupt pending high/low registers store detected interrupts via the internal system error signals (refer to Section 5.2.3 on page 62). User can clear specific bits by writing 1 to that bit (write-1-to-clear).

Table 90. System Interrupt Pending High/Low Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00608	System Interrupt Pending low	7	JS7OB	0 1	No pending interrupts via SYS7 (default) Pending interrupt via SYS7
		6	JS6OB	0 1	No pending interrupts via SYS6 (default) Pending interrupt via SYS6
		5	JS5OB	0 1	No pending interrupts via SYS5 (default) Pending interrupt via SYS5
		4	JS4OB	0 1	No pending interrupts via SYS4 (default) Pending interrupt via SYS4
		3	JS3OB	0 1	No pending interrupts via SYS3 (default) Pending interrupt via SYS3
		2	JS2OB	0 1	No pending interrupts via SYS2 (default) Pending interrupt via SYS2
		1	JS1OB	0 1	No pending interrupts via SYS1 (default) Pending interrupt via SYS1
		0	JS0OB	0 1	No pending interrupts via SYS0 (default) Pending interrupt via SYS0
0x00609	System Interrupt Pending High	7	JSFOB	0 1	No pending interrupts via SYS15 (default) Pending interrupt via SYS15
		6	JSEOB	0 1	No pending interrupts via SYS14 (default) Pending interrupt via SYS14
		5	JSDOB	0 1	No pending interrupts via SYS13 (default) Pending interrupt via SYS13
		4	JSCOB	0 1	No pending interrupts via SYS12 (default) Pending interrupt via SYS12
		3	JSBOB	0 1	No pending interrupts via SYS11 (default) Pending interrupt via SYS11
		2	JSAOB	0 1	No pending interrupts via SYS10 (default) Pending interrupt via SYS10
		1	JS9OB	0 1	No pending interrupts via SYS9 (default) Pending interrupt via SYS9
		0	JS8OB	0 1	No pending interrupts via SYS8 (default) Pending interrupt via SYS8

11 Error Reporting and Interrupt Control (continued)

11.1.5 System Interrupt Enable High/Low Registers

The system interrupt enable high/low registers allow for masking of interrupts via the internal system error signals.

Table 91. System Interrupt Enable High/Low Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0060A	System Interrupt Enable low	7	JS7EB	0 1	Disable (mask) interrupts via SYS7 (default). Enable (unmask) interrupts via SYS7.
		6	JS6EB	0 1	Disable (mask) interrupts via SYS6 (default). Enable (unmask) interrupts via SYS6.
		5	JS5EB	0 1	Disable (mask) interrupts via SYS5 (default). Enable (unmask) interrupts via SYS5.
		4	JS4EB	0 1	Disable (mask) interrupts via SYS4 (default). Enable (unmask) interrupts via SYS4.
		3	JS3EB	0 1	Disable (mask) interrupts via SYS3 (default). Enable (unmask) interrupts via SYS3.
		2	JS2EB	0 1	Disable (mask) interrupts via SYS2 (default). Enable (unmask) interrupts via SYS2.
		1	JS1EB	0 1	Disable (mask) interrupts via SYS1 (default). Enable (unmask) interrupts via SYS1.
		0	JS0EB	0 1	Disable (mask) interrupts via SYS0 (default). Enable (unmask) interrupts via SYS0.
0x0060B	System Interrupt Enable High	7	JSFEB	0 1	Disable (mask) interrupts via SYS15 (default). Enable (unmask) interrupts via SYS15.
		6	JSEEB	0 1	Disable (mask) interrupts via SYS14 (default). Enable (unmask) interrupts via SYS14.
		5	JSDEB	0 1	Disable (mask) interrupts via SYS13 (default). Enable (unmask) interrupts via SYS13.
		4	JSCEB	0 1	Disable (mask) interrupts via SYS12 (default). Enable (unmask) interrupts via SYS12.
		3	JSBEB	0 1	Disable (mask) interrupts via SYS11 (default). Enable (unmask) interrupts via SYS11.
		2	JSAEB	0 1	Disable (mask) interrupts via SYS10 (default). Enable (unmask) interrupts via SYS10.
		1	JS9EB	0 1	Disable (mask) interrupts via SYS9 (default). Enable (unmask) interrupts via SYS9.
		0	JS8EB	0 1	Disable (mask) interrupts via SYS8 (default). Enable (unmask) interrupts via SYS8.

11 Error Reporting and Interrupt Control (continued)

11.1.6 Interrupts via Internal Clock Errors

Table 92. Clock Error Interrupt Assignments

Clock Interrupt Bit	Description
CLK15	Failsafe indicator—APLL#1 reference error
CLK14	DPLL#2 sync input error
CLK13	DPLL#1 sync input error
CLK12	CT_NETREF2 error
CLK11	CT_NETREF1 error
CLK10	/FR_COMP error
CLK9	/CT_FRAME_B error
CLK8	/CT_FRAME_A error
CLK7	SCLKx2# error
CLK6	SCLK error
CLK5	C2 error
CLK4	/C4 error
CLK3	/C16– error
CLK2	/C16+ error
CLK1	CT_C8_B error
CLK0	CT_C8_A error

11 Error Reporting and Interrupt Control (continued)

11.1.7 Clock Interrupt Pending High/Low Registers

The clock interrupt pending high/low registers store detected interrupts via the internal clock error signals (refer to Section 5.2.1 on page 58). The user can clear specific bits by writing 1 to that bit (write-1-to-clear).

Table 93. Clock Interrupt Pending High/Low Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0060C	Clock Interrupt Pending Low	7	JC7OB	0 1	No pending interrupts via CLK7 (default) Pending interrupt via CLK7
		6	JC6OB	0 1	No pending interrupts via CLK6 (default) Pending interrupt via CLK6
		5	JC5OB	0 1	No pending interrupts via CLK5 (default) Pending interrupt via CLK5
		4	JC4OB	0 1	No pending interrupts via CLK4 (default) Pending interrupt via CLK4
		3	JC3OB	0 1	No pending interrupts via CLK3 (default) Pending interrupt via CLK3
		2	JC2OB	0 1	No pending interrupts via CLK2 (default) Pending interrupt via CLK2
		1	JC1OB	0 1	No pending interrupts via CLK1 (default) Pending interrupt via CLK1
		0	JC0OB	0 1	No pending interrupts via CLK0 (default) Pending interrupt via CLK0
0x0060D	Clock Interrupt Pending High	7	JCFOB	0 1	No pending interrupts via CLK15 (default) Pending interrupt via CLK15
		6	JCEOB	0 1	No pending interrupts via CLK14 (default) Pending interrupt via CLK14
		5	JCDOB	0 1	No pending interrupts via CLK13 (default) Pending interrupt via CLK13
		4	JCCOB	0 1	No pending interrupts via CLK12 (default) Pending interrupt via CLK12
		3	JCBOB	0 1	No pending interrupts via CLK11 (default) Pending interrupt via CLK11
		2	JCAOB	0 1	No pending interrupts via CLK10 (default) Pending interrupt via CLK10
		1	JC9OB	0 1	No pending interrupts via CLK9 (default) Pending interrupt via CLK9
		0	JC8OB	0 1	No pending interrupts via CLK8 (default) Pending interrupt via CLK8

11 Error Reporting and Interrupt Control (continued)

11.1.8 Clock Interrupt Enable High/Low Registers

The clock interrupt enable high/low registers allow for masking of interrupts via the internal clock error signals.

Table 94. Clock Interrupt Enable High/Low Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0060E	Clock Interrupt Enable low	7	JC7EB	0 1	Disable (mask) interrupts via CLK7 (default). Enable (unmask) interrupts via CLK7.
		6	JC6EB	0 1	Disable (mask) interrupts via CLK6 (default). Enable (unmask) interrupts via CLK6.
		5	JC5EB	0 1	Disable (mask) interrupts via CLK5 (default). Enable (unmask) interrupts via CLK5.
		4	JC4EB	0 1	Disable (mask) interrupts via CLK4 (default). Enable (unmask) interrupts via CLK4.
		3	JC3EB	0 1	Disable (mask) interrupts via CLK3 (default). Enable (unmask) interrupts via CLK3.
		2	JC2EB	0 1	Disable (mask) interrupts via CLK2 (default). Enable (unmask) interrupts via CLK2.
		1	JC1EB	0 1	Disable (mask) interrupts via CLK1 (default). Enable (unmask) interrupts via CLK1.
		0	JC0EB	0 1	Disable (mask) interrupts via CLK0 (default). Enable (unmask) interrupts via CLK0.
0x0060F	Clock Interrupt Enable High	7	JCFEB	0 1	Disable (mask) interrupts via CLK15 (default). Enable (unmask) interrupts via CLK15.
		6	JCEEB	0 1	Disable (mask) interrupts via CLK14 (default). Enable (unmask) interrupts via CLK14.
		5	JCDEB	0 1	Disable (mask) interrupts via CLK13 (default). Enable (unmask) interrupts via CLK13.
		4	JCCEB	0 1	Disable (mask) interrupts via CLK12 (default). Enable (unmask) interrupts via CLK12.
		3	JCBEB	0 1	Disable (mask) interrupts via CLK11 (default). Enable (unmask) interrupts via CLK11.
		2	JCAEB	0 1	Disable (mask) interrupts via CLK10 (default). Enable (unmask) interrupts via CLK10.
		1	JC9EB	0 1	Disable (mask) interrupts via CLK9 (default). Enable (unmask) interrupts via CLK9.
		0	JC8EB	0 1	Disable (mask) interrupts via CLK8 (default). Enable (unmask) interrupts via CLK8.

11 Error Reporting and Interrupt Control (continued)

11.1.9 Interrupt Servicing Registers

11.1.9.1 Arbitration Control Register

The arbitration control register allows for four modes of interrupt control operation as shown below:

- Disabled. This mode bypasses any interrupt controller operation. No FG or GP inputs are allowed as external interrupt inputs. SYSERR assertion is a simple logical OR of the internal system error bits. CLKERR assertion is a simple logical OR of the internal clock error bits.
- Flat. This mode treats all 48 possible inputs (eight from external FG[7:0], eight from external GP[7:0], 16 from internal system errors, 16 from internal clock errors) with equal weight, and queues them for in-service via a round-robin arbitration.
- Tier, no pre-empting. This mode assigns three priority levels. The highest level is internal clock errors CLK[15:0], next level is internal system errors SYS[15:0], lowest level is external errors FG[7:0] and GP[7:0]. Arbitration priority encodes between the three levels. Multiple interrupts within a level are queued round-robin.
- Tier, with pre-empting. This mode is the same as tier, with the added ability to preempt a current in-service interrupt according to the three priority levels.

Table 95. Arbitration Control Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00610	Arbitration Control	7:0	JAMSR	0000 0000 0000 0001 0000 0010 0001 0010	Disable interrupt controller (default) Flat structure (round-robin arbiter) Tier structure (three levels), no pre-empting Tier structure (three levels), pre-empting

11.1.10 PCI_INTA Output Select Register

The PCI_INTA output select register controls whether the internal signal which generates SYSERR also generates a PCI interrupt PCI_INTA#.

Table 96. PCI_INTA Output Select Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00611	PCI_INTA Output Select	7:0	JSPSR	0000 0000 0000 0001	Do not route SYSERR to PCI_INTA (default) Route SYSERR to PCI_INTA

11.1.10.1 SYSERR and CLKERR Output Select Register

The SYSERR output select register controls how the SYSERR signal is asserted (active-high level, active-low level, active-high pulse or active-low pulse).

The CLKERR output select register controls how the **CLKERR** signal is asserted (active-high level, active-low level, active-high pulse or active-low pulse).

11 Error Reporting and Interrupt Control (continued)

Table 97. SYSERR Output Select Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00612	SYSERR Output Select	7:0	JSOSR	0000 0000 0000 0001 0001 0000 0001 0001	SYSERR is active-high level * (default). SYSERR is active-low level *. SYSERR is active-high single pulse. SYSERR is active-low single pulse.
0x00616	SYSERR Pulse Width	7:0	JSPSR	LLLL LLLL	SYSERR pulse width value.
0x00613	CLKERR Output Select	7:0	JCOSR	0000 0000 0000 0001 0001 0000 0001 0001	CLKERR is active-high level * (default). CLKERR is active-low level *. CLKERR is active-high single pulse. CLKERR is active-low single pulse.
0x00617	CLKERR Pulse Width	7:0	JCPSR	LLLL LLLL	CLKERR pulse width value.

* When the arbitration control is disabled (0x00610 = 0000 0000), SYSERR or CLKERR levels remain asserted until all the internal system (or clock) pending bits are cleared.

11.1.10.2 Interrupt In-Service Registers

The interrupt in-service registers provide a 16-bit interrupt vector, with unique encoding to indicate which of the 48 possible interrupts is currently in-service.

Table 98. Interrupt In-Service Register

Byte Address	Register Name	Bit	Mnemonic	Value	Function
0x00618	Interrupt In-service low	7:0	Reserved	0000 0000	Lower byte of in-service vector; returns zero
0x00619	Interrupt In-service High	7:0	JISOR	0000 0000	No interrupt in-service (default)
				0001 0000	FG0 interrupt in-service
				0001 0001	FG1 interrupt in-service
				0001 0010	FG2 interrupt in-service
				0001 0011	FG3 interrupt in-service
				0001 0100	FG4 interrupt in-service
				0001 0101	FG5 interrupt in-service
				0001 0110	FG6 interrupt in-service
				0001 0111	FG7 interrupt in-service
				0010 0000	GP0 interrupt in-service
				0010 0001	GP1 interrupt in-service
				0010 0010	GP2 interrupt in-service
				0010 0011	GP3 interrupt in-service
				0010 0100	GP4 interrupt in-service
				0010 0101	GP5 interrupt in-service
				0010 0110	GP6 interrupt in-service
				0010 0111	GP7 interrupt in-service
				0100 0000	SYS0 interrupt in-service
				0100 0001	SYS1 interrupt in-service
				0100 0010	SYS2 interrupt in-service
				0100 0011	SYS3 interrupt in-service
				0100 0100	SYS4 interrupt in-service
				0100 0101	SYS5 interrupt in-service

11 Error Reporting and Interrupt Control (continued)

Table 98. Interrupt In-Service Register (continued)

Byte Address	Register Name	Bit	Mne-monic	Value	Function
0x00619	Interrupt In-Service High	7:0		0100 0110	SYS6 interrupt in-service
				0100 0111	SYS7 interrupt in-service
				0100 1000	SYS8 interrupt in-service
				0100 1001	SYS9 interrupt in-service
				0100 1010	SYS10 interrupt in-service
				0100 1011	SYS11 interrupt in-service
				0100 1100	SYS12 interrupt in-service
				0100 1101	SYS13 interrupt in-service
				0100 1110	SYS14 interrupt in-service
				0100 1111	SYS15 interrupt in-service
				1000 0000	CLK0 interrupt in-service
				1000 0001	CLK1 interrupt in-service
				1000 0010	CLK2 interrupt in-service
				1000 0011	CLK3 interrupt in-service
				1000 0100	CLK4 interrupt in-service
				1000 0101	CLK5 interrupt in-service
				1000 0110	CLK6 interrupt in-service
				1000 0111	CLK7 interrupt in-service
				1000 1000	CLK8 interrupt in-service
				1000 1001	CLK9 interrupt in-service
				1000 1010	CLK10 interrupt in-service
				1000 1011	CLK11 interrupt in-service
				1000 1100	CLK12 interrupt in-service
				1000 1101	CLK13 interrupt in-service
				1000 1110	CLK14 interrupt in-service
				1000 1111	CLK15 interrupt in-service
0x0061A	Interrupt In-Service, Byte 2	7:0		LLLL LLLL	Low byte, virtual channel identifier
0x0061B	Interrupt In-Service, Byte 3	7:1		0000 0000	NOP
		0		L	MSBit, virtual channel identifier

11 Error Reporting and Interrupt Control (continued)

11.2 Error Reporting and Interrupt Controller Circuit Operation

T8110 errors are reported via three output signals, CLKERR, SYSERR, and PCI_INTA#. These outputs are generated by an interrupt controller circuit, refer to Figure 34. The interrupt control circuit accepts 48 interrupt inputs in all. The way in which these interrupts are arbitrated is selectable, and the means of reporting the interrupts out to the system is also selectable.

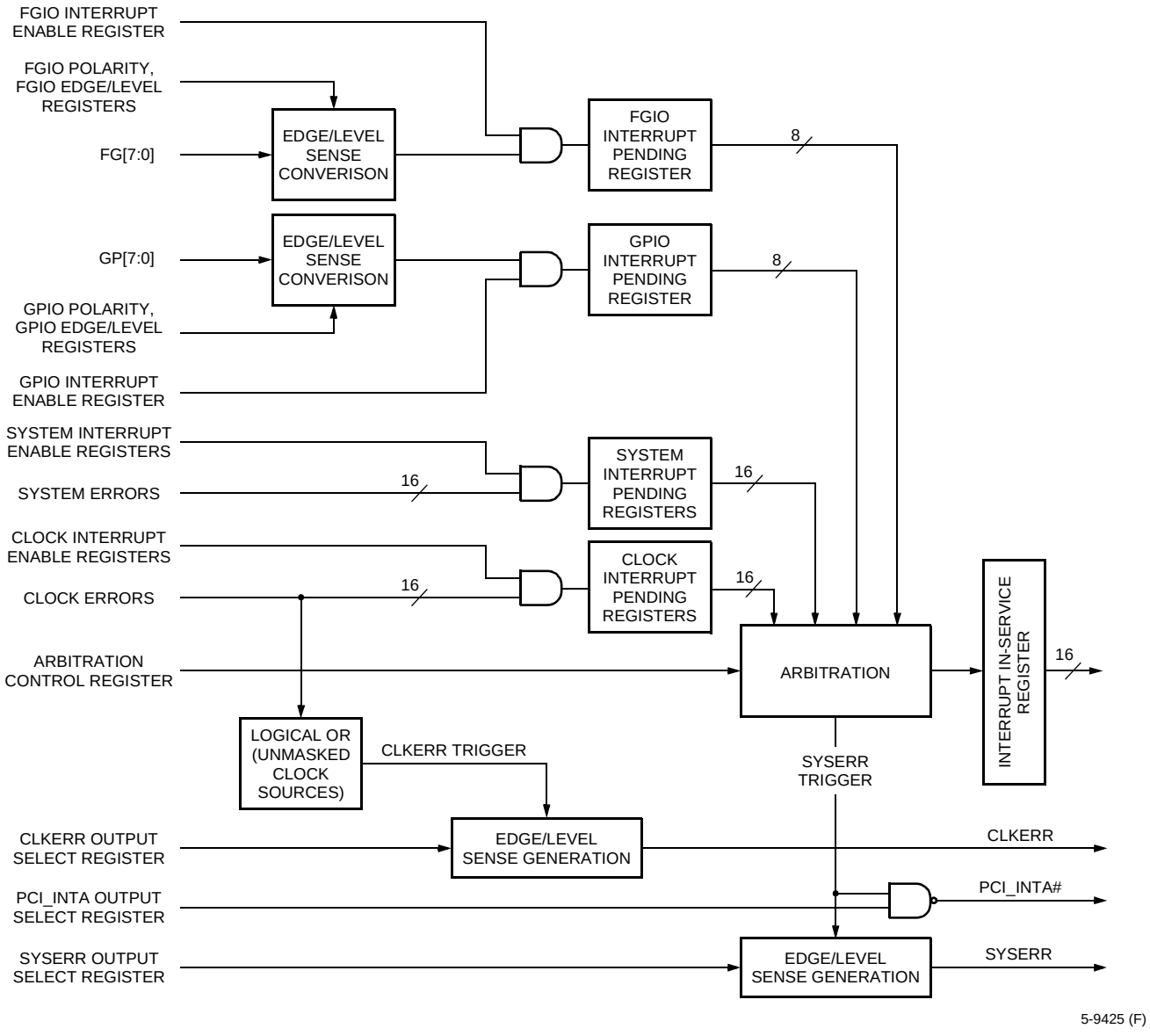


Figure 34. Interrupt Controller

11 Error Reporting and Interrupt Control (continued)

11.2.1 Externally Sourced Interrupts via FG[7:0], GP[7:0]

Up to 16 of the 48 interrupt inputs are sourced external to the T8110, via the FG[7:0] and GP[7:0] signals. Each input is independently controlled via the Interrupt control registers (refer to Section 11.1.1 on page 112 and Section 11.1.2 on page 114). Any externally sourced interrupt may be presented as active-high level, active-low level, positive edge or negative edge sense. Each external interrupt is maskable. Any detected interrupt which is unmasked is held in an interrupt pending register, and presented to the arbitration circuit for servicing.

11.2.2 Internally Sourced System Error Interrupts

Another set of 16 of the 48 interrupt inputs are sourced internally via the system error register bits (0x00126—127, refer to Section 5.2.3 on page 62). Each of these inputs is independently controlled via the interrupt control registers (refer to Section 11.1.3 on page 115). All internal system error bit interrupts are presented as active-high level sense. Each system error bit interrupt is maskable. Any detected interrupt which is unmasked is held in an interrupt pending register and presented to the arbitration circuit for servicing.

11.2.3 Internally Sourced Clock Error Interrupts

Another set of 16 of the 48 interrupt inputs are sourced internally via the latched clock error register bits (0x00122—123, refer to Section 5.2.1 on page 58). Each of these inputs is independently controlled via the interrupt control registers (refer to Section 11.1.6 on page 118). All internal clock error bit interrupts are presented as active-high level sense. Each clock error bit interrupt is maskable. Any detected interrupt that is unmasked is held in an interrupt pending register and presented to the arbitration circuit for servicing.

11.2.4 Arbitration of Pending Interrupts

The arbitration of the pending interrupts can be handled in one of four selectable modes. Arbitration off, FLAT arbitration, TIER arbitration with pre-empting disabled, and TIER arbitration with pre-empting enabled. Interrupts are reported to the system via the SYSERR signal (and the PCI_INTA# signal, if enabled to do so).

11.2.4.1 Arbitration OFF

This mode only allows the 16 internal system error register bits to generate interrupts, and no arbitration takes place. The trigger for the SYSERR output is simply a logical OR of the internal system error register bits. All bits of the internal system error register must be cleared in order to rearm the SYSERR trigger in this mode.

11.2.4.2 FLAT Arbitration

The FLAT arbitration mode performs a round-robin arbitrations on all 48 interrupt sources. When a pending interrupt wins the arbitration, the in-service register is loaded with its corresponding interrupt vector, SYSERR is triggered, and that pending bit is cleared, removing it from the next round-robin arbitration cycle. The system must respond to the current in-service interrupt (refer to Section 11.2.8 on page 126), after which the next arbitration cycle takes place.

11.2.4.3 TIER Arbitration

The TIER arbitration creates three prioritized groups as shown below:

- Highest priority. The 16 internal latched clock error register bits.
- Next highest priority. The 16 internal system error register bits.
- Lowest priority. The 16 external FG[7:0] and GP[7:0] bits.

11 Error Reporting and Interrupt Control (continued)

Arbitration assigns interrupt servicing priority to the three groups. Multiple pending interrupts within the same group are arbitrated round-robin. When a pending interrupt wins the arbitration, the in-service register is loaded with its corresponding interrupt vector, SYSERR is triggered, and that pending bit is cleared, removing it from the next arbitration cycle.

11.2.4.3.1 Pre-Empting Disabled

With pre-empting disabled, once a pending interrupt wins the arbitration and the in-service register is loaded with its corresponding interrupt vector, new incoming pending interrupts of higher priority must wait for the system to respond to the current in-service interrupt (refer to Section 11.2.8 on page 126), at which time another arbitration cycle takes place.

11.2.4.3.2 Pre-Empting Enabled

With pre-empting enabled, an interrupt that is in-service (i.e., its interrupt vector is loaded in the in-service register and SYSERR has been triggered) can be overridden by new incoming pending interrupts of higher priority. The current in-service interrupt is pushed onto a stack for storage, the higher-priority interrupt vector is loaded into the in-service register and SYSERR is retriggered. Once all interrupts of higher priority have been serviced by the system (refer to Section 11.2.8 on page 126), the stack is popped and the original lower-priority interrupt is reissued.

11.2.5 CLKERR Output

The CLKERR output signal is used to indicate any internal clocking errors. The trigger for the CLKERR output is simply a logical OR of the internal latched clock error register bits. All bits of the internal clock error register must be cleared in order to rearm the CLKERR trigger. The CLKERR trigger induces a state machine to generate the CLKERR signal in one of four possible ways: active-high level, active-low level, active-high single pulse, or active-low single pulse.

11.2.6 SYSERR Output

The T8110 SYSERR output signal is used to report interrupts. Internally, the arbitration circuit provides a SYSERR trigger, which induces a state machine to generate the SYSERR signal in one of four possible ways: active-high level, active-low level, active-high single pulse, or active-low single pulse.

11.2.7 PCI_INTA# Output

The internal SYSERR trigger can be enabled to also trigger a PCI interrupt via the PCI_INTA# signal.

11.2.8 System Handling of Interrupts

The T8110 interrupt controller presents an interrupt to the system by triggering the SYSERR output and providing a predefined interrupt vector value at the interrupt in-service register (ISR). The system may acknowledge the interrupt in three ways:

- System reads the T8110 ISR Register. This allows the arbiter to advance, and if more pending interrupts are active, reloads the ISR with the winner of the arbitration, and retriggers SYSERR.
- System clears the T8110 ISR register (via register 0x00100, soft reset; write 0x20 clears the ISR). The arbiter advances, and if more pending interrupts are active, reloads the ISR and retriggers SYSERR.
- System resets the interrupt controller (via register 0x00100, Soft Reset, write 0x10 clears the ISR and all the pending interrupt registers). All pending interrupts are cleared, and the arbiter is reset.

12 Test and Diagnostics

12.1 Diagnostics Control Registers

The diagnostic control registers allow for various diagnostic modes (refer to Section 12.2 on page 130).

Table 99. Diagnostics Control Register Map

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00140	Diag3, GP Testpoint Select	Diag2, GP Testpoint Enable	Diag1, FG Testpoint Select	Diag0, FG Testpoint Enable
0x00144	Diag7, External Buffer RETRY Timer	Diag6, Miscellaneous Diagnostics low	Diag5, State Counter Modes High	Diag4, State Counter Modes low
0x00148	Reserved	Reserved	Reserved	Diag8, Interrupt Controller Diagnostics

12.1.1 FG Testpoint Enable Register

The FG testpoint enable register allows individual programming of FG[7:0] bits for either standard operation (as FG or FGIO) or as test point outputs. FG testpoint select controls the MUX selection for which testpoints are selected. Refer to Table 101 on page 128 for test point assignments for each FG bit.

Table 100. FG Testpoint Enable Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00140	Diag0 FG Testpoint Enable	7	FT7EB	0 1	FG7 is standard FG or FGIO bit (default). FG7 is a testpoint.
		6	FT6EB	0 1	FG6 is standard FG or FGIO bit (default). FG6 is a testpoint.
		5	FT5EB	0 1	FG5 is standard FG or FGIO bit (default). FG5 is a testpoint.
		4	FT4EB	0 1	FG4 is standard FG or FGIO bit (default). FG4 is a testpoint.
		3	FT3EB	0 1	FG3 is standard FG or FGIO bit (default). FG3 is a testpoint.
		2	FT2EB	0 1	FG2 is standard FG or FGIO bit (default). FG2 is a testpoint.
		1	FT1EB	0 1	FG1 is standard FG or FGIO bit (default). FG1 is a testpoint.
		0	FT0EB	0 1	FG0 is standard FG or FGIO bit (default). FG0 is a testpoint.
0x00141	Diag1 FG Testpoint Select	7:0	FTPSR	LLLL LLLL	Value for mux selection of testpoints output to FG[7:0]—see Table 101 on page 128.

12 Test and Diagnostics (continued)

Table 101. FG[7:0] Internal Testpoint Assignments

FG Testpoint Select value	FG7	FG6	FG5	FG4	FG3	FG2	FG1	FG0
TBD	TBD	TBD	TBD	TBD	TBD	TBD	TBD	TBD

12.1.2 GP Testpoint Enable Register

The GP testpoint enable register allows individual programming of GP[7:0] bits for either standard operation (as GPIO) or as test point outputs. GP testpoint select controls the MUX selection for which testpoints are selected. Refer to Table 102 on page 128 for test point assignments for each GP bit.

Table 102. Testpoint Enable Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00142	Diag2 GP Testpoint Enable	7	GT7EB	0 1	GP7 is standard GPIO bit (default). GP7 is a testpoint.
		6	GT6EB	0 1	GP6 is standard GPIO bit (default). GP6 is a testpoint.
		5	GT5EB	0 1	GP5 is standard GPIO bit (default). GP5 is a testpoint.
		4	GT4EB	0 1	GP4 is standard GPIO bit (default). GP4 is a testpoint.
		3	GT3EB	0 1	GP3 is standard GPIO bit (default). GP3 is a testpoint.
		2	GT2EB	0 1	GP2 is standard GPIO bit (default). GP2 is a testpoint.
		1	GT1EB	0 1	GP1 is standard GPIO bit (default). GP1 is a testpoint.
		0	GT0EB	0 1	GP0 is standard GPIO bit (default). GP0 is a testpoint.
0x00143	Diag3 GP Testpoint Select	7:0	GTPSR	LLLL LLLL	Value for MUX selection of testpoints output to GP[7:0]—see Table 102 on page 128.

Table 103. GP[7:0] Internal Testpoint Assignments

GP Testpoint Select Value	GP7	GP6	GP5	GP4	GP3	GP2	GP1	GP0
TBD	TBD	TBD	TBD	TBD	TBD	TBD	TBD	TBD

12 Test and Diagnostics (continued)

12.1.3 State Counter Modes Registers

The state counter modes registers control state counter diagnostics, including the breaking of state counter carry chains, using /FR_COMP as the internal frame reference, and allowing the state counter to roll over early via a modulo function. For more details, refer to Section 12.2 on page 130.

Table 104. State Counter Modes Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00144	Diag4 State Counter Modes Low	7:0	SCLSR	LLLL LLLL	Lower 8 bits of state counter modulo load value
0x00145	Diag5 State Counter Modes High	7:6	Reserved	00	NOP (default)
		5	SCMSB	0 1	Normal carry chain operation (default) Break state counter carry chains
		4	FRMSB	0 1	Normal internal frame operation (default) Use /FR_COMP as internal frame
		3	SCLSB	0 1	Normal counting (default) State counter modulo counting
		2:0	SCUSP	LLL	Upper 3 bits of state counter modulo load value

12.1.4 Miscellaneous Diagnostics Low Register

The miscellaneous diagnostics low register: bit 5 enables a mode to shorten the PCI target discard timers for minibridge, VC memory and DATA memory target accesses. Bit 4 enables ISA access to the minibridge register space. Bit 3 enables ISA access to the virtual channel memory region. Bits 2 and 1 allow direct reset of the analog PLL feedback dividers. Bit 0 controls the TST input of the power-on reset cell.

Table 105. Miscellaneous Diagnostics Low Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00146	Diag6 Miscellaneous Diagnostics Low	7:6	Reserved	00	NOP (default)
		5	TBD	0 1	PCI target discard timers normal (default) PCI target discard timers shortened
		4	MBREB	0 1	ISA access to minibridge registers disabled (default) ISA access to minibridge registers enabled
		3	VCMEB	0 1	ISA access to VCMEM disabled (default) ISA access to VCMEM enabled
		2	FB2SB	0 1	PLL#2 feedback divider reset inactive (default) PLL#2 feedback divider reset active
		1	FB1SB	0 1	PLL#1 feedback divider reset inactive (default) PLL#1 feedback divider reset active
		0	PWREB	0 1	Power-on reset TST input inactive (default) Power-on reset TST input active

12 Test and Diagnostics (continued)

12.1.5 External Buffer RETRY Timer Register

The external buffer RETRY timer register provides a timeout value for external buffer accesses. Value indicates [number of 65.536 MHz periods –1]. The RETRY in this context refers to receiving a BUFFER LOCKED status on the first descriptor table fetch attempt, and then retrying the fetch after the retry timer expires. Refer to Section 13.2.3.4 for more detail.

Table 106. External Buffer RETRY Timer Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00147	Diag7, External Buffer RETRY Timer	7:0	EBOSR	LLLL LLLL	RETRY timer value for external buffer access
0x00148	Diag8, Interrupt Controller Diagnostic	7:6	ICDSP	00 01	Interrupt controller, normal mode (DEFAULT) Interrupt controller, DIAG mode
		5:4	ICKSP	LL	DIAG mode, force CLK[1:0] errors
		3:2	ISYSP	LL	DIAG mode, force SYS[1:0] errors
		1:0	IEXSP	LL	DIAG mode, force EXT[8, 0] errors

12.2 Diagnostic Circuit Operation

The T8110 internal diagnostic modes are intended primarily for chip manufacturing test. The diagnostic functions include the following:

- DIAG0—3: Observability of internal test points via FG[7:0], GP[7:0]. Internal test points are brought to chip I/O at FG and GP signals. Refer to Table 101 and Table 102 for test point assignment.
- DIAG4—5: Internal state counter diagnostic modes.
  - — Break counter carry chains—this is used in conjunction with monitoring of the state counter bits at FG and GP, and breaks the 11-bit state counter into three separate pieces (bits [10:8], [7:4], and [3:0]).
  - — Shorten frame operation—the internally generated 8 kHz frame is bypassed in favor of the /FR_COMP input. The /FR_COMP input still denotes the frame center and may be presented at a higher frequency than 8 kHz. This is used in conjunction with the state counter modulo function, which when properly programmed allows the internal state counter to roll over coincident with the /FR_COMP frame center.
- DIAG6: ISA access to the virtual channel memory and minibridge register regions. The VCMEM region is only functionally applicable for packet payload switching, which is only available when the T8110 interface to a local PCI bus is selected. When interface to ISA bus is selected, this diagnostic setting allows direct access to the virtual channel memory. The minibridge registers are only functionally applicable for minibridge port operation and are only available when the T8110 interface to a local PCI bus is selected. When interface to ISA bus is selected, this diagnostic setting allows direct access to the minibridge registers.
- DIAG6: Forced RESET of analog PLL feedback dividers. The APLL feedback dividers are typically not reset. This diagnostic mode allows each feedback divider to be held in a reset state.
- DIAG6: Power-on reset cell test. The TST input to the power-on reset cell is directly controlled via a diagnostic register bit.
- DIAG7: External buffer RETRY timer. The external buffer access protocol allows for one RETRY of a descriptor table fetch in the case of a locked external buffer. This diagnostic register allows for manipulation of the amount of time to wait before retrying a descriptor table fetch. Refer to Section 13.2.3.4 for more detail.

12 Test and Diagnostics (continued)

- DIAG8: Interrupt controller diagnostics. When the diagnostic mode is enabled (DIAG8 register, bits 7:6 = 01), then bits 5:4 override the CLK error[1:0] inputs, bits [3:2] override the SYS error[1:0] inputs, bit 1 overrides the GP[0] input, and bit 0 overrides the FG[0] input to the interrupt controller. This allows for direct manipulation to set/clear a portion of interrupt bits from each tier group. Refer to Section 11.2.

12.3 Other Test Functions

12.3.1 Analog PLL Test Modes

There are five dedicated input pins for APLL testing. Additionally, other functional pins are used as APLL clock monitors when the APLLs are placed in test mode, as follows.

Table 107. Analog PLL Test Mode Signals

Signal	Description
PTEST	0 = Normal mode: clock registers control PLL refclk, frequency selections. 1 = Test mode: PLL#1 refclk forced to PRI_REF_IN, PSEL controls frequency, enable monitor outputs.
PEN	0 = Bypass mode: Disable PLLs, force PLL#1 and 2 into BYPASS. 1 = Normal mode: clock registers control PLL bypass.
PSEL	Rate selector when PTEST is active-high: 0 = PLL#1 times 16, PLL#2 times 4. 1 = PLL#1 times 32, PLL#2 times 8.
PLOCK	APLL#1 lock indicator.
PPDN	For powerdown IDDQ testing.
NR1_SEL_OUT NR2_SEL_OUT PRI_REF_OUT TCLK_OUT	PLL#1 VCOP monitor when PTEST is active-high. PLL#1 VCO monitor when PTEST is active-high. PLL#1 lock status monitor when PTEST is active-high. PLL#1 feedback clock monitor when PTEST is active-high.
L_SC3 L_SC2 L_SC1	PLL#2 VCOP monitor when PTEST is active-high. PLL#2 VCO monitor when PTEST is active-high. PLL#2 feedback clock monitor when PTEST is active-high.

12.3.2 Memory Cell BIST

TBD

12.3.3 JTAG

TBD

12.3.4 Internal Scan

TBD

13 Connection Control—Standard and Virtual Channel

13.1 Programming Interface

Programming the T8110 for standard and virtual channel switching requires specific access cycles to the connection memory and virtual channel memory regions. Access to any other region (data memory, minibridge or registers) is made through a standard direct access via the interface (described starting in Section 4 on page 26).

13.1.1 PCI Interface

When the T8110 PCI interface is the selected bus interface, both standard telephony and virtual channel switching are allowed. A standard telephony connection is made by writing to a location in the connection memory. A virtual channel connection requires a write to the connection memory plus a corresponding write to the virtual channel memory.

13.1.1.1 PCI Connection Memory Programming

The connection memory is divided into four 2K regions, each of which handles up to 128 time slots worth of connectivity for each of 16 serial data streams. The regions include H.1x0 EVEN streams (CT_D[30 ,28, . . . 0]), H.1x0 ODD streams (CT_D[31, 29, . . . 1]), local low streams (L_D[15:0]), and local high streams (L_D[31:16]). The connection memory locations are addressed relative to time slot and stream. BURST access is allowed to the connection memory. In order to SKIP making/breaking a connection to a particular time slot and stream combination that gets addressed during a burst, the user must disable all the byte enables for that particular data phase.

Connection memory commands are as follows:

RESET PAGE, refer to Figure 35: resets any (up to all four) connection memory region. Address bit 15 determines whether or not it's a reset page command. The reset page command relies on a valid internal chip clock and loops through all addresses within the connection memory region, resetting the VALID bit field. The reset page command must be presented as a PCI memory write command.

MAKE/BREAK/QUERY, telephony connection, refer to Figure 36.

MAKE/BREAK/QUERY, virtual channel nonbonded connection, refer to Figure 37.

MAKE/BREAK/QUERY, virtual channel bonded connection, refer to Figure 38.

MAKE and BREAK commands above must be presented as PCI memory write commands. QUERY commands are presented as PCI memory read commands.

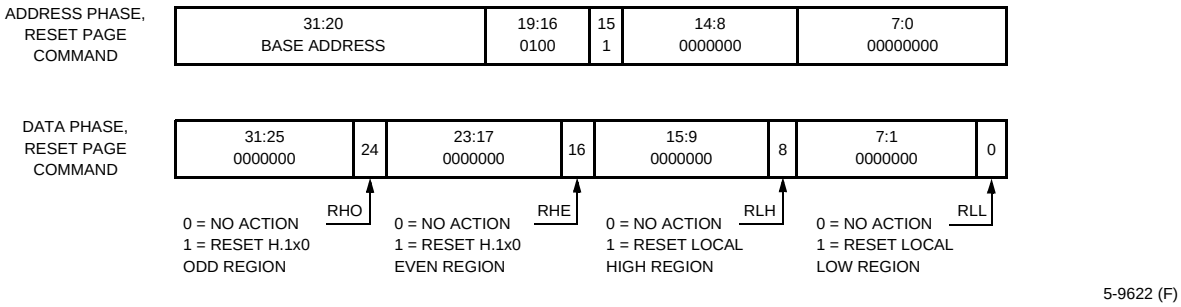
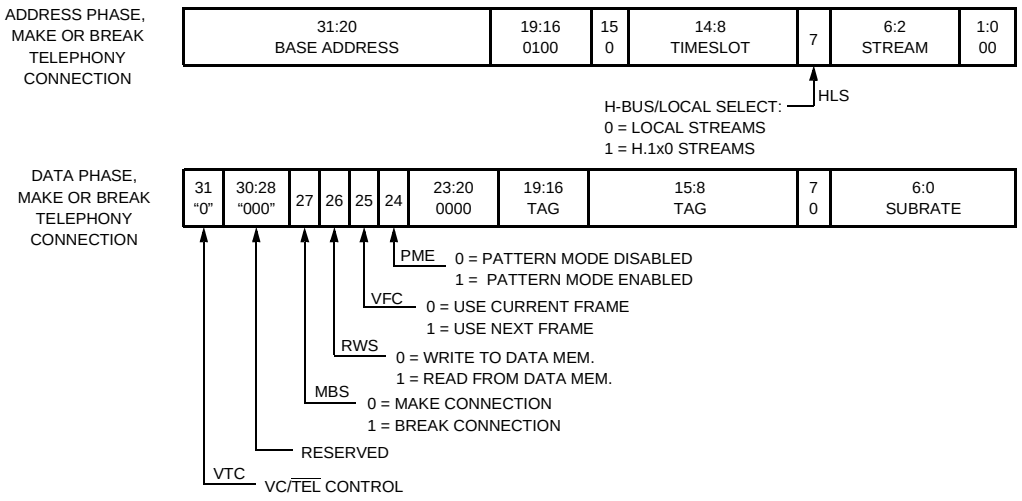


Figure 35. PCI Programming—Reset Page Command

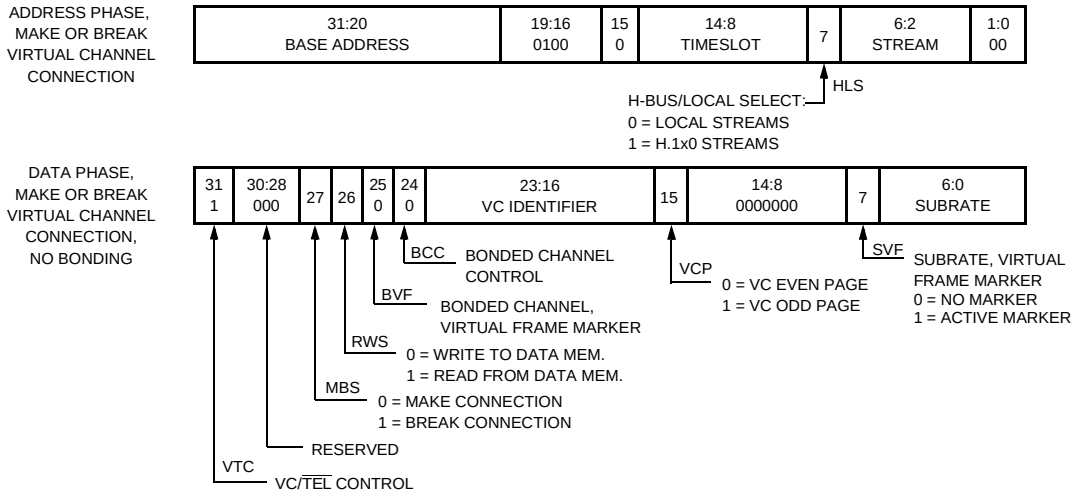
13 Connection Control—Standard and Virtual Channel (continued)



5-9623 (F)

Note: For QUERY (read cycle), a 0 in bit 27 of the data phase indicates an invalid, or broken connection. A 1 in bit 27 indicates a valid connection.

Figure 36. PCI Programming—Make/Break/Query Telephony Connection

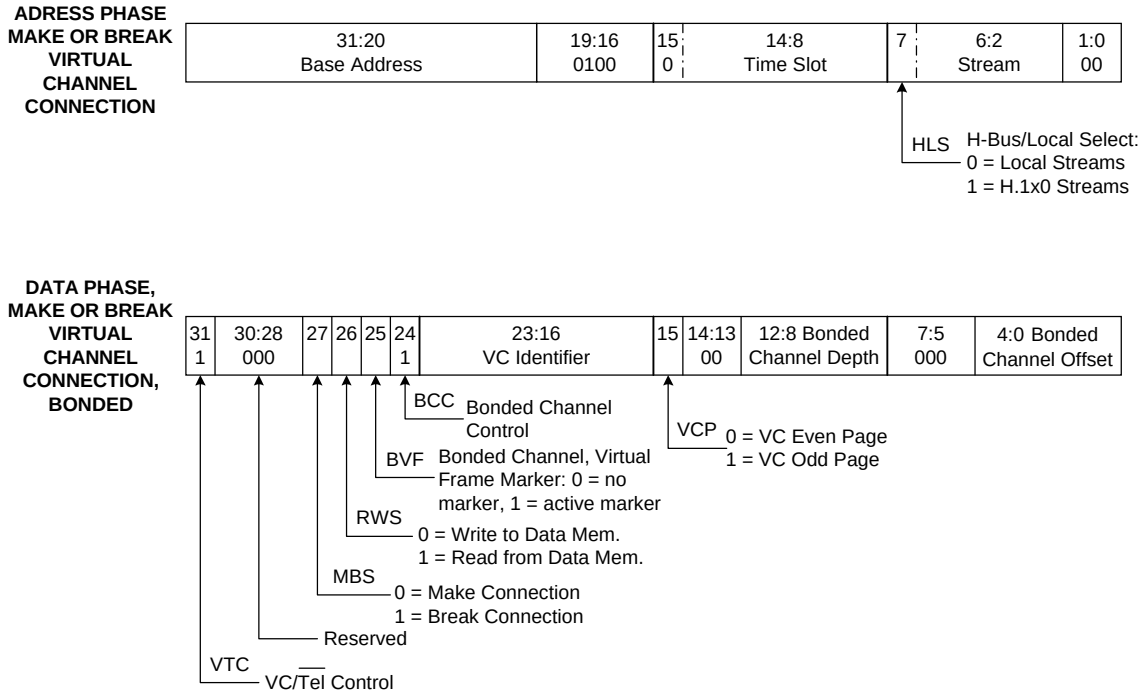


5-9624 (F)

Note: For QUERY (read cycle), a 0 in bit 27 of the data phase indicates an invalid, or broken connection. A 1 in bit 27 indicates a valid connection.

Figure 37. PCI Programming—Make/Break/Query Virtual Channel Nonbonded Connection

13 Connection Control—Standard and Virtual Channel (continued)



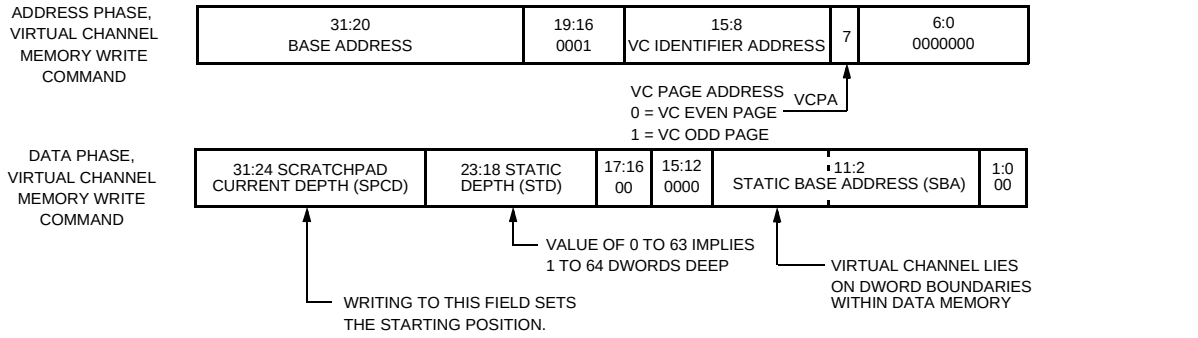
Note: For QUERY (read cycle), a 0 in bit 27 of the data phase indicates an invalid or broken connection. A 1 in bit 27 indicates a valid connection.

Figure 38. PCI Programming—Make/Break/Query Virtual Channel Bonded Connection

13.1.1.2 PCI Virtual Channel Memory Programming

The virtual channel memory is divided into two regions: the static portion and the scratchpad portion. The static portion contains two read/write fields, defining a particular virtual channel's base address and depth. The scratchpad portion contains one read/write field (depth) and one read-only field (current offset). On any write to the virtual channel memory, the scratchpad current offset is reset to zero. Virtual channel memory commands include:

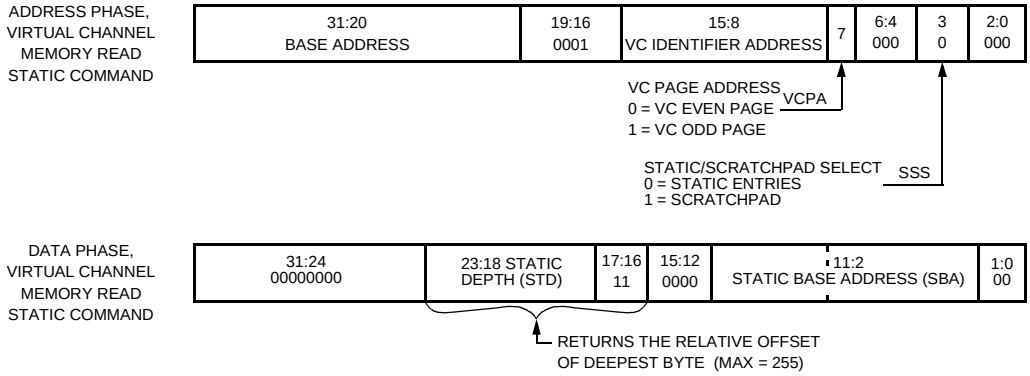
Write (see Figure 39 on page 134). This is presented as a PCI memory write command.
Read, static (see Figure 40 on page 135). This is presented as a PCI memory read command.
Read scratchpad (see Figure 41 on page 135). This is presented as a PCI memory read command.



5-9628 (F)

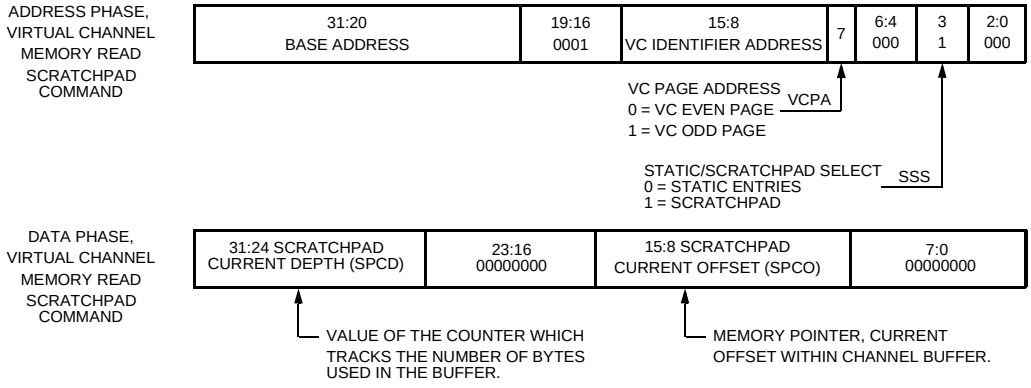
Figure 39. PCI Programming—Write Virtual Channel Command

13 Connection Control—Standard and Virtual Channel (continued)



5-9627 (F)

Figure 40. PCI Programming—Read Virtual Channel Static Command



5-9630 (F)

Figure 41. PCI Programming—Read Virtual Channel Scratchpad Command

13.1.2 ISA Interface

When the T8110 ISA interface is the selected bus interface, only standard telephony switching is allowed (no virtual channel switching). A standard telephony connection is made by writing to a location in the connection memory. There is a mode in which the virtual channel memory may be accessed via the ISA interface, included for diagnostic purpose only.

13.1.2.1 ISA Connection Memory Programming

Because the ISA interface only allows word or byte accesses, multiple write accesses must occur. The ISA connection memory access mimics the 32-bit PCI access by using a combination of the lower two address bits [1:0] and holding registers. For byte access, there are a total of three byte-wide holding registers. For word access, there is one word-wide holding register. The user must load the holding registers with the proper information first, then write to the upper byte (or upper word) to actually move data into the connection memory, refer to Table 108.

13 Connection Control—Standard and Virtual Channel (continued)

The connection memory is divided into four 2K regions, each of which handle up to 128 time slots worth of connectivity for each of 16 serial data streams. The regions include H.1x0 even streams (CT_D[30,28, . . . 0]), H.1x0 odd streams (CT_D[31,29, . . . 1]), local low streams (L_D[15:0]), and local high streams (L_D[31:16]). The connection memory locations are addressed relative to time slot and stream.

Connection memory commands are as follows:

RESET PAGE (see Figure 42 on page 137) resets any (up to all four) connection memory region. Address bit 15 determines whether or not it's a reset page command. The reset page command relies on a valid internal chip clock and loops through all addresses within the connection memory region, resetting the VALID bit field. The RESET PAGE command is presented as either two ISA WORD writes, or four ISA BYTE writes, see Table 108.

MAKE/BREAK/QUERY, telephony connection (see Figure 43 on page 137).

MAKE/BREAK/QUERY, virtual channel nonbonded connection (see Figure 44 on page 138).

MAKE/BREAK/QUERY, virtual channel bonded connection (see Figure 45 on page 139).

Note: Making virtual channel connections in the connection memory is for diagnostic purpose only when the ISA interface is selected.

The MAKE and BREAK commands are presented as multiple ISA write cycles. The QUERY command is presented as multiple ISA read cycles, refer to Table 108.

Table 108. ISA Programming, Connection Memory Access

Word/Byte (MB_CS5)	A[1:0]	D[15:8]	D[7:0]	Access Description
Byte	00	X	Data Byte 0	Write data byte 0 to a holding register, or read data byte 0 information.
Byte	01	X	Data Byte 1	Write data byte 1 to a holding register, or read data byte 1 information.
Byte	10	X	Data Byte 2	Write data byte 2 to a holding register, or read data byte 2 information.
Byte	11	X	Data Byte 3	Write data byte 3 plus the holding register data to connection memory, or read data byte 3 information.
Word	0X	Data Byte 1	Data Byte 0	Write data bytes 1 and 0 to a holding register, or read data bytes 1 and 0 information.
Word	1X	Data Byte 3	Data Byte 2	Write data bytes 3 and 2 plus the holding register data to connection memory, or read data bytes 3 and 2 information.

13 Connection Control—Standard and Virtual Channel (continued)

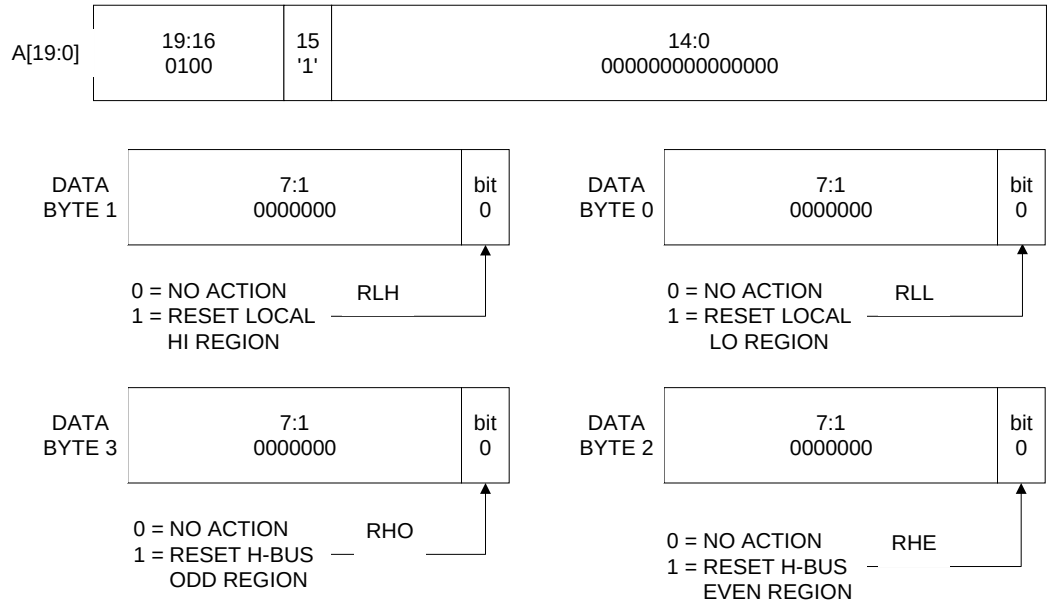


Figure 42. ISA Programming—Reset Page Command

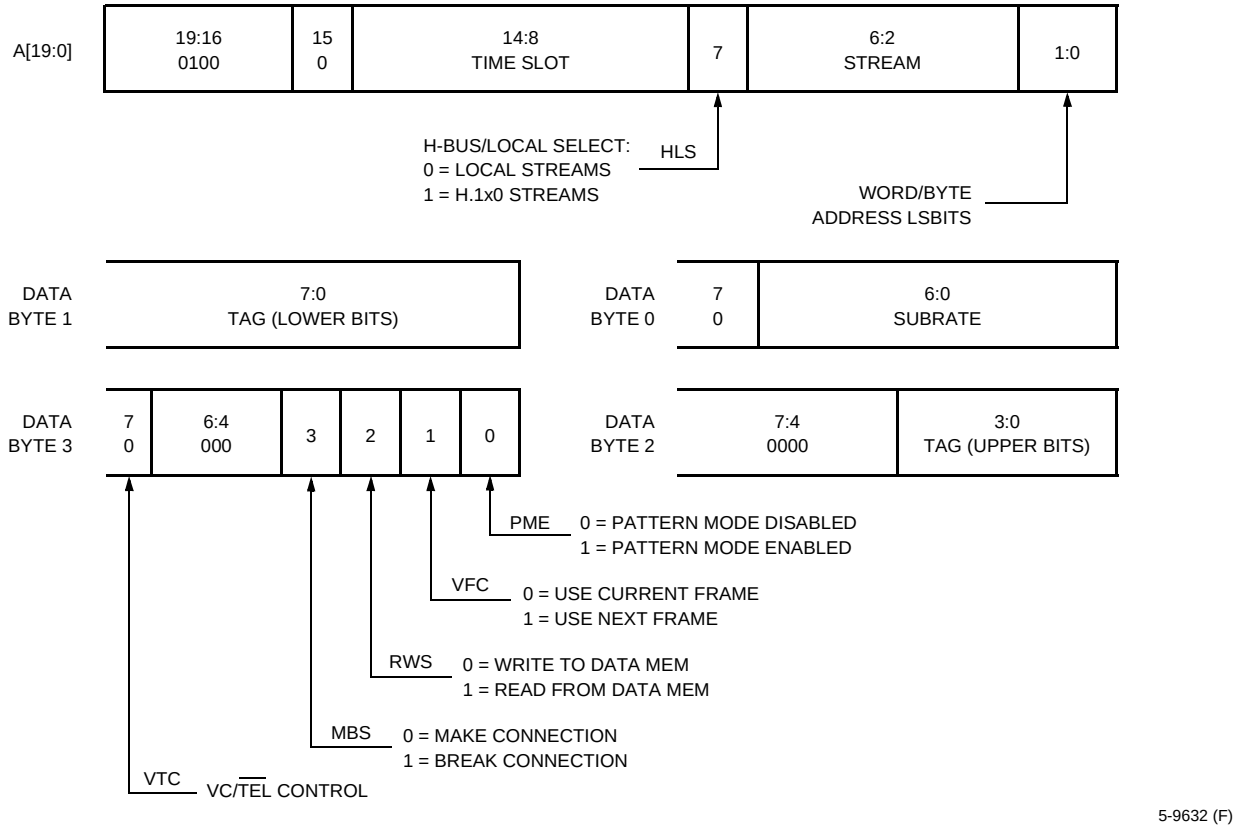


Figure 43. ISA Programming—Make/Break/Query Telephony Connections

13 Connection Control—Standard and Virtual Channel (continued)

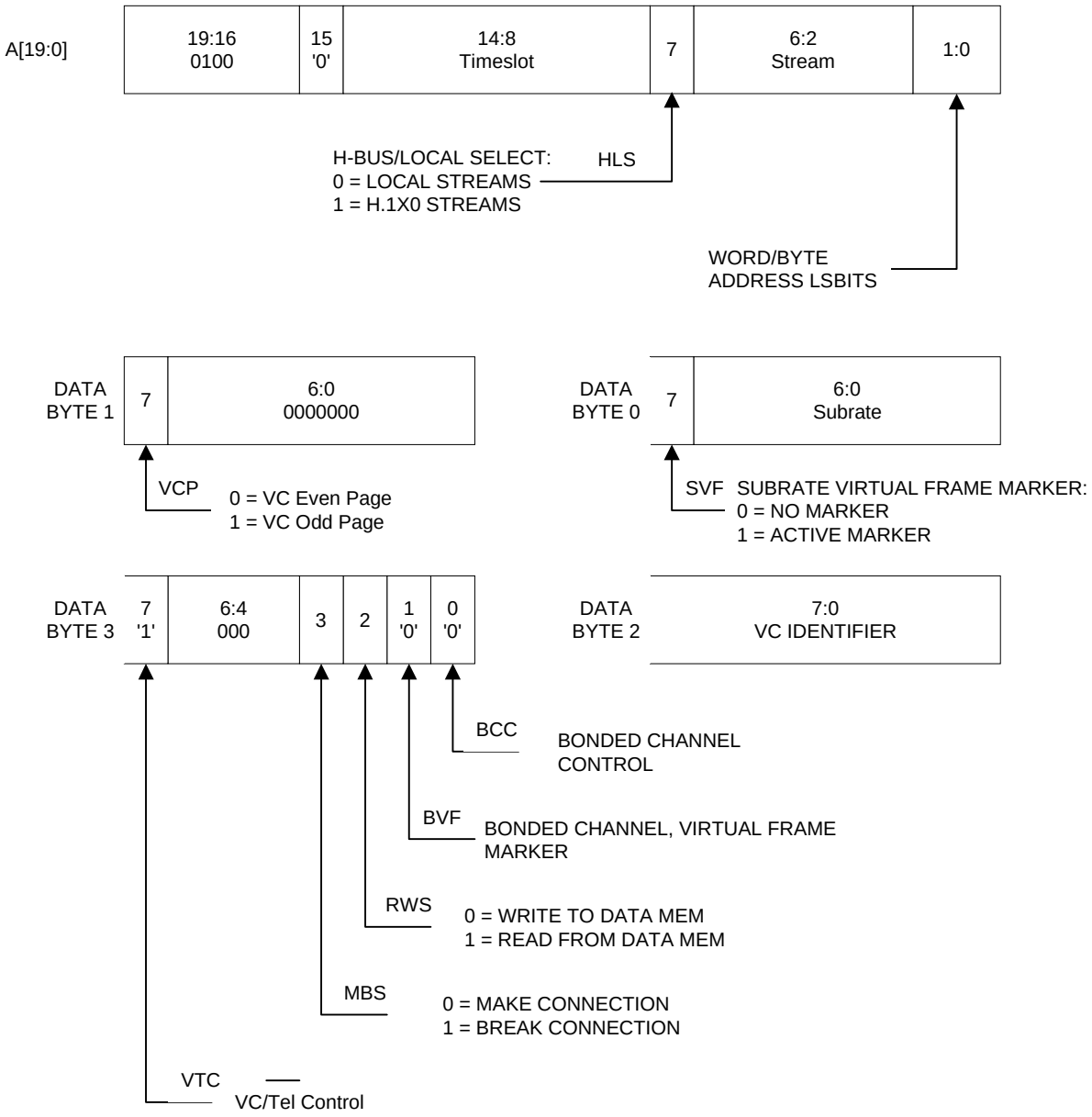


Figure 44. ISA Programming—Make/Break/Query Virtual Channel Nonbonded Connections

13 Connection Control—Standard and Virtual Channel (continued)

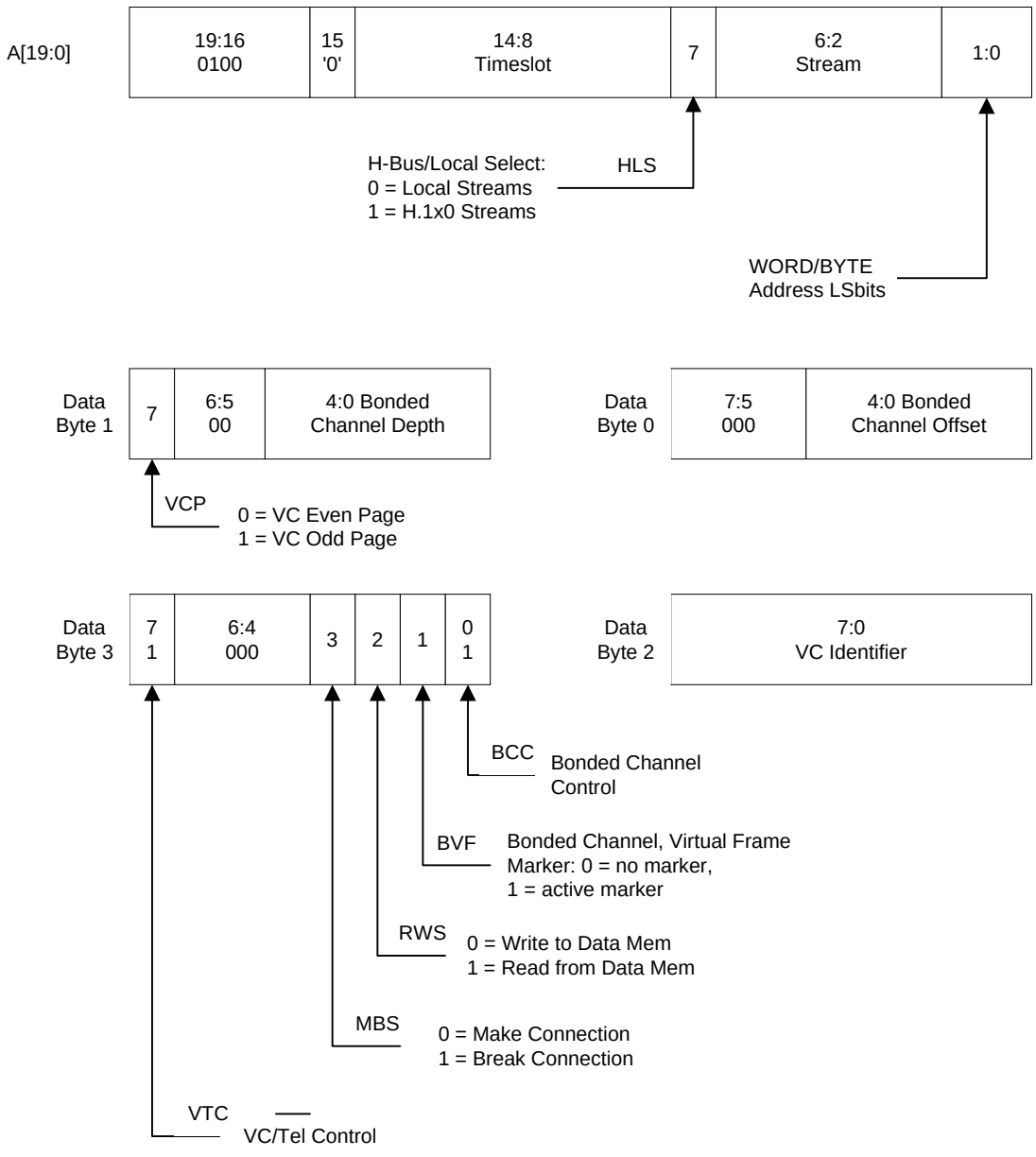


Figure 45. ISA Programming—Make/Break/Query Virtual Channel Bonded Connection

13 Connection Control—Standard and Virtual Channel (continued)

13.1.2.2 ISA Virtual Channel Memory Programming

Because the ISA interface only allows word or byte accesses, multiple write accesses must occur. The ISA virtual channel memory access mimics the 32-bit PCI access by using a combination of the lower two address bits [1:0] and holding registers. For byte access, there are a total of three byte-wide holding registers. For word access, there is one word-wide holding register. The user must load the holding registers with the proper information first, and then write to the upper byte (or upper word) to actually move data into the virtual channel memory, refer to Table 109.

The virtual channel memory is divided into two regions: the static portion and the scratchpad portion. The static portion contains two read/write fields, defining a particular virtual channel's base address and depth. The scratchpad portion contains one read/write field (depth) and one read-only field (current offset). On any write to the virtual channel memory, the scratchpad current offset is reset to zero. Virtual channel memory commands include:

WRITE (see Figure 46 on page 141). This command is presented as an ISA write cycle.

READ STATIC (see Figure 47 on page 141). This command is presented as an ISA read cycle.

READ SCRATCHPAD (see Figure 48 on page 142). This command is presented as an ISA read cycle.

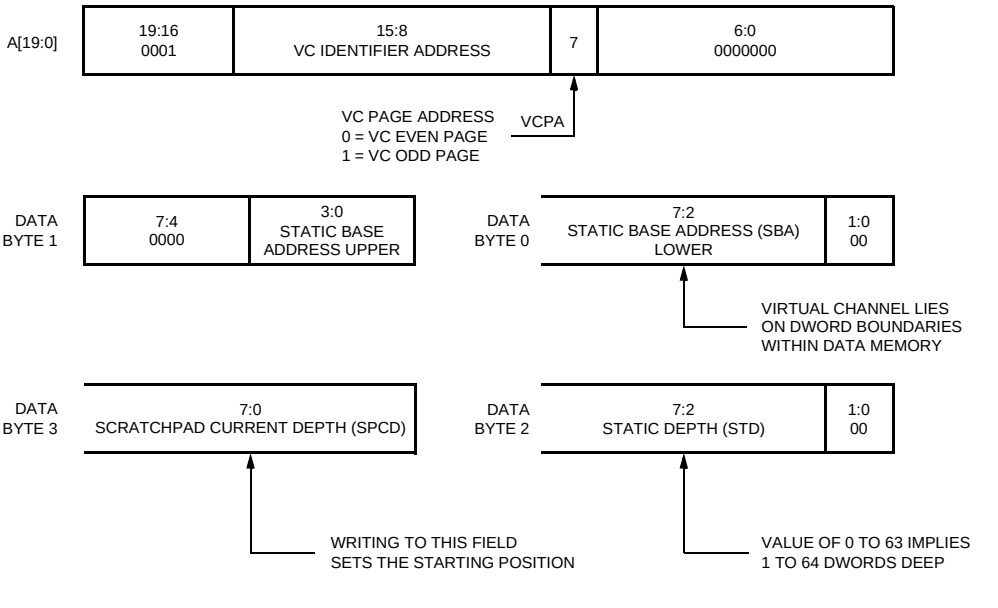
Note: Accessing the virtual channel memory is for diagnostic purpose only when the ISA interface is selected.

Table 109. Virtual Channel Memory Access

Word/Byte (MB_CS5)	A[1:0]	D[15:8]	D[7:0]	Access Description
Byte	00	X	Data byte 0	Write data byte 0 to a holding register, or read data byte 0 information.
Byte	01	X	Data byte 1	Write data byte 1 to a holding register, or read data byte 1 information.
Byte	10	X	Data byte 2	Write data byte 2 to a holding register, or read data byte 2 information.
Byte	11	X	Data byte 3	Write data byte 3 plus the holding register data to a virtual channel memory, or read data byte 3 information.
Word	0X	Data byte 1	Data byte 0	Write data bytes 1 and 0 to a holding register, or read data bytes 1 and 0 information.
Word	1X	Data byte 3	Data byte 2	Write data bytes 3 and 2 plus the holding register data to a virtual channel memory, or read data bytes 3 and 2 information.

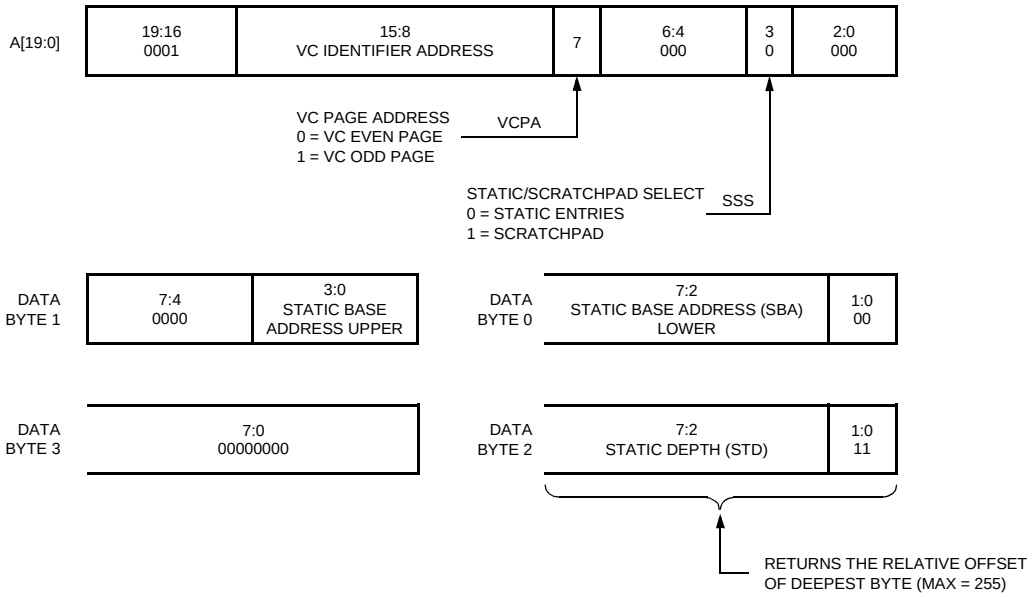
Note: Data byte n required information is shown in Figure 46, Figure 47, and Figure 48.

13 Connection Control—Standard and Virtual Channel (continued)



5-9635 (F)

Figure 46. ISA Programming—Write Virtual Channel Memory Command



5-9636 (F)

Figure 47. ISA Programming—Read Virtual Channel Static Command

13 Connection Control—Standard and Virtual Channel (continued)

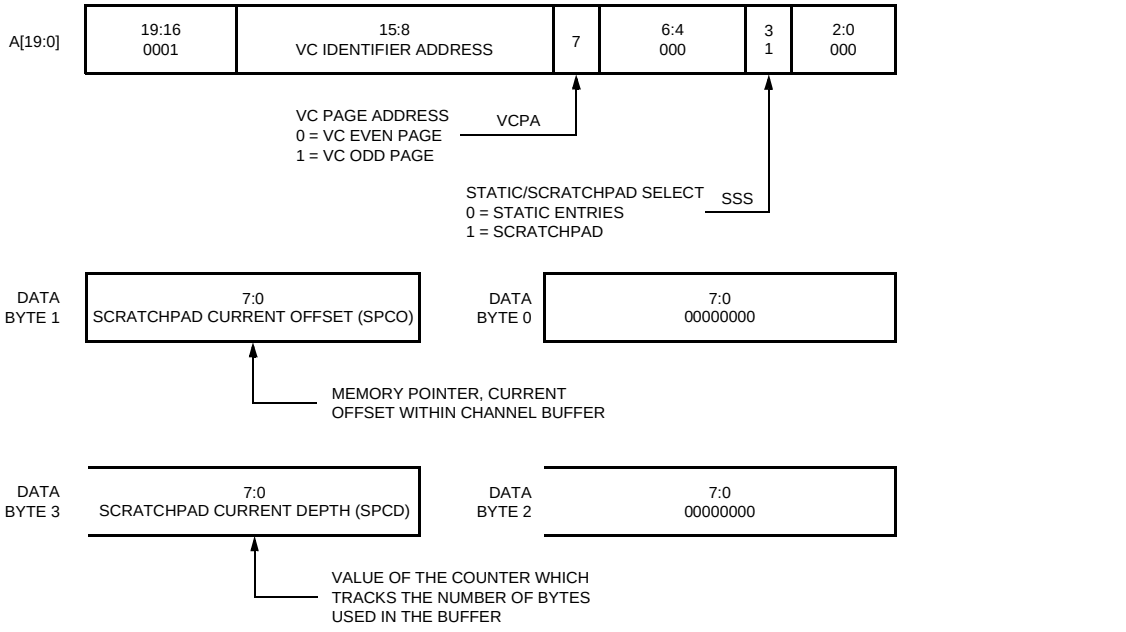


Figure 48. ISA Programming—Read Virtual Channel Scratchpad Command

13.2 Switching Operation

T8110 provides two main switching operations, standard (telephony) switching and virtual channel (packet payload) switching. The basic building block of switching is one half simplex connections loaded into the connection memory. Each connection memory location controls data flow, either from a serial stream input to a location in data memory, or from data memory to a serial stream output. A typical telephony simplex switch connection would use one **from** and one **to** connection, each using the same data memory location. A virtual channel switch connection would only use one half simplex connection (**to** or **from**), plus control provided in virtual channel memory to initiate transfers between the data memory and an external buffer in the PCI space.

13.2.1 Memory Architecture and Configuration

13.2.1.1 Connection Memory

The T8110 connection memory consists of 8192 locations, one location for each of the possible stream/time-slot combinations, to provide a full nonblocking switch for up to 128 time slots on 32 H.1x0 streams (CT_D[31:0]) and 32 local streams (L_D[31:0]). Connection memory is physically addressed by time slot (7 bits), H.1x0/local select (1 bit), and stream (5 bits).

The 8192 locations are divided into four pages of 2048, with each page dedicated to a set of 16 serial streams:

- H.1x0 EVEN streams (CT_D[30, 28, . . . 0])
- H.1x0 ODD streams (CT_D[31, 29, . . . 1])
- Local high streams (L_D[31:16])
- Local low streams (L_D[15:0])

13 Connection Control—Standard and Virtual Channel (continued)

Each of these connection memory pages are initialized at reset (valid bit entries are reset to invalid). Additionally, each page may be initialized individually via software command, RESET PAGE (refer to Figure 35 on page 132 and Figure 42 on page 137).

For standard telephony switching connections, the connection memory locations contain one half simplex switch control information (refer to Figure 36 on page 133 and Figure 43 on page 137), including:

- VALID bit: indicates that a valid switch connection exists for this stream/time slot.
- VTC: indicates whether the connection is a virtual channel connection or a telephony connection. A 0 denotes a telephony connection.
- RWS: indicates whether the connection is **from** (from serial stream to data mem) or **to** (from DATA mem to serial stream).
- VFC: virtual framing control, controls which data page is used in double-buffer scenarios.

  Note: There are three data memory configurations that allow double-buffering of the data, in order to create constant frame delay connections. Refer to Section 13.2.1.2 on page 144 and Section 13.2.2.1 on page 145.
- PME: indicates a pattern mode connection.
- TAG: the data memory location used for this one half simplex switch connection (or the data pattern sent to serial output for pattern mode connections).
- SUBRATE information: Subrate switching control (bitswap).

For virtual channel (packet payload) switching connections, there are two possible control fields, depending on whether the virtual channel is nonbonded (refer to Figure 37) or bonded (refer to Figure 38).

13.2.1.1.1 Virtual Channel Switching, Nonbonded Connections

- VALID bit: indicates that a valid switch connection exists for this stream/time slot.
- VTC: indicates whether the connection is a virtual channel connection or a telephony connection. A 1 denotes a virtual channel connection.
- RWS: indicates whether the connection is **from** (from serial stream to data mem) or **to** (from DATA mem to serial stream).
- BVF: Bonded virtual frame marker (unused for nonbonded channels).
- BCC: Bonded channel control indicator, 0 denotes a nonbonded channel.
- VC identifier and VCP: which virtual channel this information is for (0 up to 511 virtual channels).
- SVF: Subrate virtual frame marker, indicator for the last piece of a packed subrate byte.
- SUBRATE information: Subrate switching control (bitswap); refer to Section 13.2.2.3.

13.2.1.1.2 Virtual Channel Switching, Bonded Connections

- VALID bit: Indicates that a valid switch connection exists for this stream/time slot.
- VTC: Indicates whether the connection is a virtual channel connection or a telephony connection. A 1 denotes a virtual channel connection.
- RWS: Indicates whether the connection is **from** (from serial stream to data mem) or **to** (from DATA mem to serial stream).
- BVF: Bonded virtual frame marker, indicator for the last byte switched in a frame.

13 Connection Control—Standard and Virtual Channel (continued)

- BCC: Bonded channel control indicator, 1 denotes a bonded channel.
- VC identifier and VCP: which virtual channel this information is for (0 up to 511 virtual channels).
- Bonded channel depth: defines how many bytes within one frame are switched.
- Bonded channel offset: a specific data memory pointer offset for the data byte related to this connection.

13.2.1.2 Data Memory

The T8110 data memory is 4096 bytes, which can be programmatically configured in six ways, via the Data mem mode sel register (0x00105, refer to Section 5.1.3 on page 50). The data memory is allocated either for 4 Kbytes telephony, 4 Kbytes of virtual channels, or 2 Kbytes of telephony and 2 Kbytes of virtual channels (see Figure 49 on page 144).

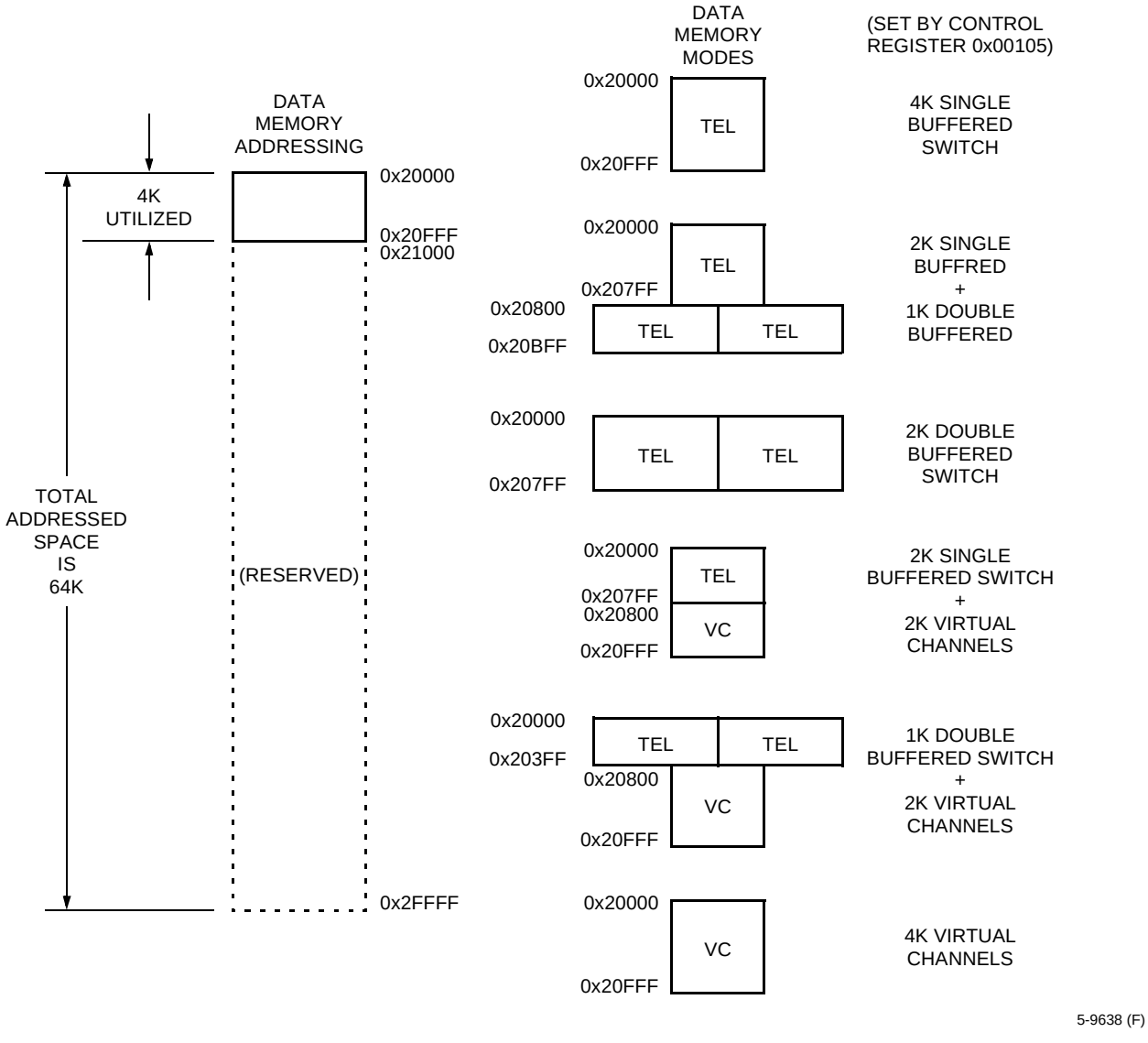


Figure 49. T8110 Data Memory Map and Configurations

13 Connection Control—Standard and Virtual Channel (continued)

13.2.1.3 Virtual Channel Memory

The T8110 virtual channel memory consists of 512 locations, one location for each possible virtual channel. Virtual channel memory consists of two portions, static and scratchpad, and controls how the VC portion of the data memory is partitioned when the data memory is configured to allow for virtual channels (refer to Figure 49). The data memory partition for a virtual channel is defined by the static portion of the virtual channel memory (refer to Figure 39 and Figure 40), which includes the following information:

- SBA, static base address: the physical start address for this particular virtual channel in the data memory VC space.
- STD, static depth: the total number of bytes (in DWORDS) allotted for this virtual channel. A virtual channel can occupy 2 DWORDS (minimum) up to 64 DWORDS (maximum) in the data memory VC space.

The scratchpad portion of the virtual channel memory (refer to Figure 39 and Figure 41) keeps track of the current data memory address within the data memory partition. Information includes:

- SPCD: Scratchpad current depth, the current number of bytes transferred to/from serial streams.
- SPCO: Scratchpad current offset, the data memory address pointer.

13.2.2 Standard Switching

Standard telephony switching is achieved by loading control fields into the connection memory for one half simplex connections (refer to Figure 36 on page 133, Figure 43 on page 137, and Section 13.2.1.1 on page 142).

13.2.2.1 Constant Delay and Minimum Delay Connections

The VFC control bit in connection memory determines which of two data pages is accessed, when the data memory is configured to double-buffering for telephony connections (refer to Figure 49). This bit always affects **to** connections (read the data memory, send it out to a serial stream output) in a double-buffer configuration. This bit can control a **from** connection in a double-buffer configuration, only if it is a subrate connection; otherwise, the VFC bit has no bearing on **from** connections.

The double-buffering configuration creates two data pages. During a particular frame (125 µs time boundary, partitioned into time slots), one page is the active page, the other is the inactive page. The active/inactive page status toggles at every frame boundary. For all **from** connections (except for subrate connections), incoming serial data is always written to the active page. For all **to** connections, the VFC control bit indicates whether to read from the active or inactive page. Manipulation of this bit affects the latency between the incoming **from** data and the outgoing **to** data. This latency defines whether or not a connection is constant delay or minimum delay.

Refer to Appendix A for more detail on constant and minimum delay connections.

13.2.2.2 Pattern Mode

The PME control bit in connection memory affects only **to** connections. Instead of reading a value out of the data memory for subsequent output to a serial stream, the lower 8 bits of the TAG field provide a byte pattern for the serial output.

13.2.2.3 Subrate

The subrate control bit field in connection memory is used only by **from** connections, and controls how individual bits or groups of bits of an incoming serial byte are shuffled prior to writing them to the data memory, in order to achieve subrate switching.

13 Connection Control—Standard and Virtual Channel (continued)

13.2.2.3.1 Subrate Switching Overview

Traditional byte-oriented TDM data switching provides 8 bits of data per time slot, or channel, regardless of the TDM stream bit rate. A particular channel occurs once every 8 kHz frame, and there are 8K frames per second. This allows for a channel data propagation rate of (8 bits/frame * 8K frames/s = 64 Kbits/s).

Refer to Figure 50 and Table 111.

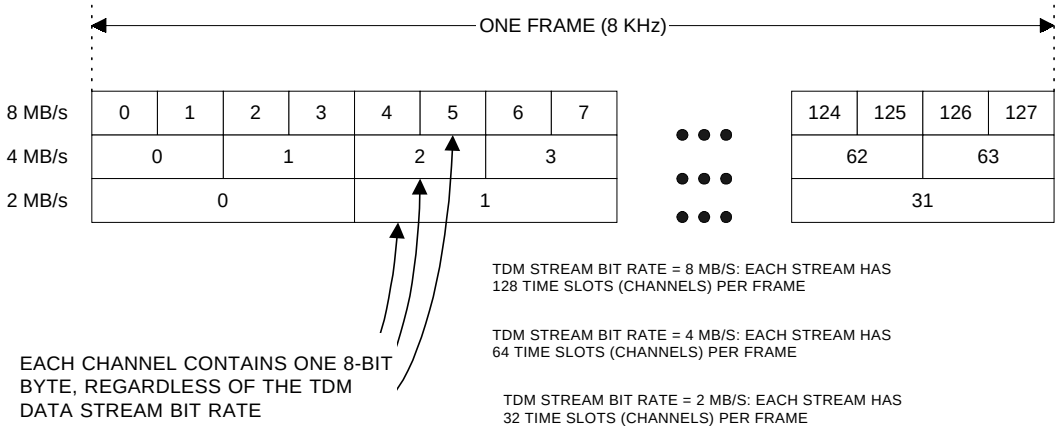


Figure 50. TDM Data Stream Bit Rates

Subrate refers to switching fractional portions of the byte-oriented TDM data streams. The T8110 allows the 8 bits of a byte-oriented channel to be broken into multiple channels of fewer bits, either two 4-bit channels, four 2-bit channels, or eight 1-bit channels. This lowers the data propagation rate per channel, but increases the overall channel capacity for a given time slot. Refer to Table 110 and Table 111.

Table 110. TDM Data Stream

One Time Slot (or Channel)

Bit	7	6	5	4	3	2	1	0
Di-Bit	7:6		5:4		3:2		1:0	
Nibble	7:4				3:0			
Byte	7:0							

Notes: Bit subrate = 8 channels per time slot, 1 bit per channel.
Di-bit = 4 channels per time slot, 2 bits per channel.
Nibble subrate = 2 channels per time slot, 4 bits per channel.
Byte (no subrate) = 1 channel per time slot, 8 bits per channel.

13 Connection Control—Standard and Virtual Channel (continued)

Table 111. Subrate Switching, Data Propagation Rate vs. Channel Capacity

Subrate Type	Bits per Channel	Channel Data Propagation Rate (Bits/Frame x 8 Kframes/s)	Channel Capacity (Relative to Byte Switching)
Bit	1	8 Kbits/s	8X
Di-bit	2	16 Kbits/s	4X
Nibble	4	32 Kbits/s	2X
Byte (no subrate)	8	64 Kbits/s	1X

13.2.2.3.2 Subrate Switching Using T8110

The H.1x0 bus and the local stream bus are based on byte-oriented TDM data streams—data is always switched as whole bytes. The subrate data must be packed into these bytes prior to switching (refer to Sections 13.2.2.3.3 and 13.2.2.3.4). The data bytes are not necessarily constrained to using fully packed bytes—any portion of a byte may be used. Subrate switching using T8110 requires:

- Overall subrate enable mode is activated (register 0x00105, data mem mode sel bit 7 is set; refer to Section 5.1.3).
- The subrate field of the connection memory entry for that switch connection is set up. This field contains 7 bits which control the type of subrate (i.e., bit, di-bit, nibble, or byte), and the data bit shuffling within the TDM byte data, **from** and **to** (refer to Figure 36, Figure 43, and Table 112).
- The VFC connection memory bit for cases where a double-buffering configuration is set up in the data memory (refer to Figure 36, Figure 43 and Sections 13.2.1.2, 13.2.2.1).

In order to program a subrate simplex connection, the subrate field is only required for the **from** half of that connection. Incoming serial byte data has its bit positions rearranged based on the subrate field contents prior to being written into the data memory. For double-buffered data memory configurations, the VCF bit controls which of two data pages the rearranged byte is written to. The **to** half of a subrate simplex connection simply outputs the entire byte found at the data memory location used for that connection, and its connection memory subrate field is ignored.

Table 112. Subrate Switching, Connection Memory Programming Setup

Subrate Type	Subrate Connection Memory Bit Field (6:0)						
	6	5	4	3	2	1	0
Bit	1	000 = **from** bit 0 001 = **from** bit 1 010 = **from** bit 2 011 = **from** bit 3 100 = **from** bit 4 101 = **from** bit 5 110 = **from** bit 6 111 = **from** bit 7			000 = **to** bit 0 001 = **to** bit 1 010 = **to** bit 2 011 = **to** bit 3 100 = **to** bit 4 101 = **to** bit 5 110 = **to** bit 6 111 = **to** bit 7		
	Subrate Connection Memory Bit Field (6:0						
	6	5	4	3	2	1	0
Di-Bit	01		00 = **from** bits[1:0]" 01 = **from** bits[3:2] 10 = **from** bits [5:4] 11 = **from** bits[7:6]		Reserved	00 = **to** bits[1:0] 01 = **to** bits[3:2] 10 = **to** bits [5:4] 11 = **to** bits[7:6]	

13 Connection Control—Standard and Virtual Channel (continued)

Table 112. Subrate Switching, Connection Memory Programming Setup (continued)

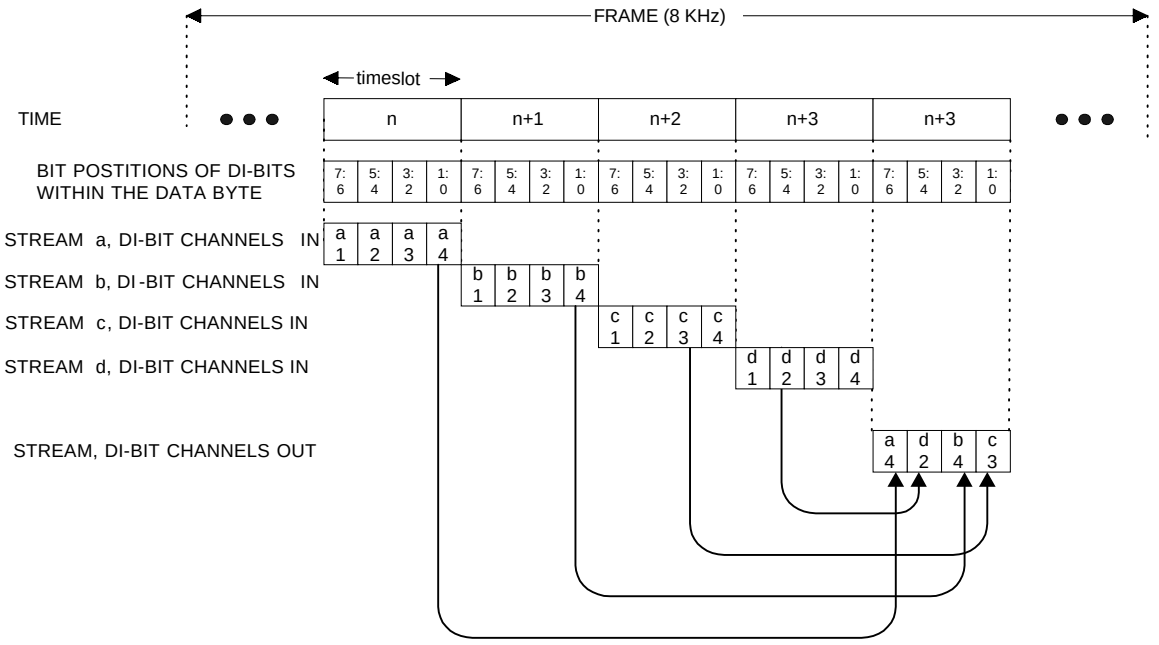
Subrate Type	Subrate Connection Memory Bit Field (6:0						
	6	5	4	3	2	1	0
Nibble	001			0 = **from** bits[3:0] 1 = **from** bits[7:4]	Reserved		0 = **to** bits[3:0] 1 = **to** bits[7:4]
	Subrate Connection Memory Bit Field (6:0						
	6	5	4	3	2	1	0
Byte	000			Reserved			

13.2.2.3.3 Subrate Packing of Outgoing Bytes

The outgoing **to** half of a simplex connection reads an entire byte from a data memory location. The packing of separate incoming subrate pieces into this byte is achieved by setting up multiple **from** one half simplex connections for one **to** one half simplex connection, all using the same data memory location. An example is illustrated in Figure 51. This example shows the packing of four separate incoming di-bits from four different channels into one outgoing byte on one channel.

Note: The limitation that multiple di-bits from the same time slot cannot be switched simultaneously. This would require the byte of that time slot to be unpacked first, which is discussed in the next section.

13 Connection Control—Standard and Virtual Channel (continued)



Connectivity:

1. From stream a, time slot n, bits[1:0] to stream e, timeslot n+4, bits[7:6]
2. From stream b, time slot n+1, bits[1:0] to stream e, timeslot n+4, bits[3:2]
3. From stream c, time slot n+2, bits[3:2] to stream e, timeslot n+4, bits[1:0]
4. From stream d, time slot n+3, bits[5:4] to stream e, timeslot n+4, bits[5:4]

Required connection memory programming:

Five half simplex connections are required to pack four incoming di-bits into an outgoing byte.

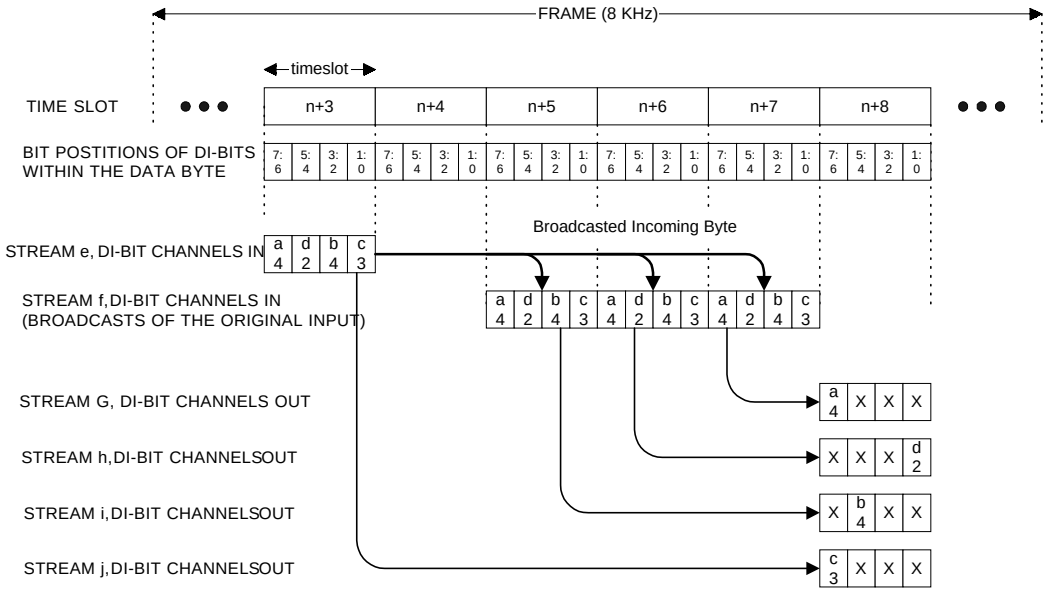
1. From stream a, time slot n. Connection memory subrate field = 0100X11
2. From stream b, time slot n+1. Connection memory subrate field = 0100X01
3. From stream c, time slot n+2. Connection memory subrate field = 0101X00
4. From stream d, time slot n+3. Connection memory subrate field = 0110X10
5. To stream e, time slot n+4. Connection memory subrate field is don't care.

Figure 51. Subrate Switching Example, Byte Packing

13.2.2.3.4 Subrate Unpacking of Incoming Bytes

Because the H.1x0 bus and the local stream bus are based on byte-oriented TDM data streams, and the T8110 architecture is geared towards standard byte switching, it is not possible to simultaneously switch subrate portions of a single byte to different places. This limitation is overcome by application. To gain access to each subrate piece contained in one incoming byte, that byte must be broadcast onto additional channels, one channel for each sub-rate piece required. The means of broadcasting is up to the application—either the source device of the packed subrate byte can broadcast it, or the device receiving that byte can broadcast it over unused channels and loop the broadcast bytes back in. The example from Figure 51 is extended in Figure 52. This example shows the unpacking of the packed byte created in Figure 51, output to four different channels.

13 Connection Control—Standard and Virtual Channel (continued)



Connectivit:y:

1. From stream e, time slot n+3, bits[1:0] to stream j, time slot n+8, bits[7:6]
2. From stream f, time slot n+5, bits[3:2] to stream i, time slot n+8, bits[5:4]
3. From stream f, time slot n+6, bits[5:4] to stream h, time slot n+8, bits[1:0]
4. From stream f, timeslot n+7, bits[7:6] to stream g, time slot n+8, bits[7:6]

Required connection memory programming:

Eight half simplex connections are required to unpack one incoming byte to four separate outgoing di-bits.

1. From stream e, time slot n+3. Connection memory subrate field = 0100X11
2. From stream f, time slot n+5. Connection memory subrate field = 0101X10
3. From stream f, time slot n+6. Connection memory subrate field = 0110X00
4. From stream f, time slot n+7. Connection memory subrate field = 0111X11
5. To stream g, time slot n+8. Connection memory subrate field is don't care.
6. To stream h, time slot n+8. Connection memory subrae field is don't care.
7. To stream i, time slot n+8. Connection memory subrate field is don't care.
8. To stream j, time slot n+8. Connection memory subrte field is don't care.

Figure 52. Subrate Switching Example, Byte Unpacking

13 Connection Control—Standard and Virtual Channel (continued)

13.2.3 Virtual Channel (Packet Payload) Switching

Packet payload switching is achieved by loading control fields into the connection memory for one half simplex connections (refer to Figure 37 and Figure 38), and loading control fields for a corresponding virtual channel into the virtual channel memory (refer to Figure 39— Figure 41). A virtual channel connection consists of the following:

- A one half simplex connection **to** (or **from**) a channel (or channels) in the H.1x0 or local bus TDM switching domain. These connections are made in a similar manner to standard telephony switching connections—by loading the proper control fields into the T8110 connection memory (refer to Section 13.2.1.1, Figure 38, and Figure 39). There are two types of virtual channel connections: nonbonded refers to switching of a single TDM channel each frame, bonded refers to switching of multiple TDM channels each frame. Additionally, subrate switching is allowed for nonbonded virtual channels. Refer to Sections 13.2.3.1—13.2.3.3 for more detail.
- A store-and-forward buffer that has access **from** (or **to**) an external buffer which is defined somewhere in the PCI bus space (see Section 13.2.3.4). The T8110 uses its data memory as the store-and-forward buffer. Depending on the data memory configuration (refer to Figure 49), there can be as many as 512 unique virtual channels defined simultaneously (using all 4 Kbytes of the available space). Configuration of the data memory space for virtual channels is achieved by loading control fields into the virtual channel memory (refer to Section 13.2.1.2, 13.2.1.3, and Figure 39—Figure 41).

The above two parts define a virtual channel. Each virtual channel can be either:

- **From** TDM domain **to** PCI domain. The data flow from incoming serial TDM data to the PCI external buffer is referred to as PUSH. In this case, the T8110 controls writes to the external buffer. Another agent on the PCI bus (such as a coprocessor) would control the reads from the external buffer. The handshake between the T8110 and the other agent is described in Section 13.2.3.4.
- **From** PCI domain **to** TDM domain. The data flow from the PCI external buffer to outgoing serial TDM data is referred to as PULL. In this case, the T8110 controls reads from the external buffer. Another agent on the PCI bus (such as a coprocessor) would control the writes to the external buffer. The handshake between the T8110 and the other agent is described in Section 13.2.3.4.

13.2.3.1 Nonbonded Channels

A nonbonded virtual channel means that only one byte of information per 8 kHz frame is switched in the TDM domain for that channel. The T8110 data memory configuration for any virtual channel holds multiple data bytes at a time (minimum of 8 bytes, maximum of 256 bytes, in increments of 4 bytes). Since only 1 byte per frame is switched, the data memory depth defined for a nonbonded virtual channel directly corresponds to the number of TDM frames worth of data stored at one time. The concept is illustrated in Figure 53 and in Figure 54.

TBD*

Figure 53. Bonded Virtual Channel in the PUSH Direction (TBD)

.**TBD***

Figure 54. Bonded Virtual Channel in the PULL Direction (TBD)

13.2.3.2 Subrate

Nonbonded virtual channels in the PUSH direction allow for subrate switching and the packing of incoming subrate pieces into bytes (refer to Section 13.2.2.3 for a general description of subrate. For the PULL direction, all bits of

13 Connection Control—Standard and Virtual Channel (continued)

the byte are output to the TDM domain so subrate is not applicable). In addition to the normal subrate field loaded into the connection memory (refer to Table 112), one additional control bit, SVF subrate virtual frame marker (refer to Section 13.2.1.1) is required as a means to denote the completion of a byte packing operation. Please refer to Figure 55.

TBD

Figure 55. Nonbonded Virtual Channel with Subrate and Packed Bytes (TBD)

13.2.3.3 Bonded Channels

A bonded virtual channel means that more than one byte of information per 8 kHz frame is switched in the TDM domain for that channel. The number of TDM frames worth of data stored at one time in the T8110 data memory depends on the number of bytes switched per frame, plus the overall data memory depth defined for that channel. For example, if a virtual channel is defined to switch 4 bytes per frame, and the data memory depth is set to 8 bytes, then the data memory for that channel can store two TDM frames worth of data at a time. An additional control bit, BVF bonded virtual frame marker (refer to Section 13.2.1.1) is required as a means to mark the last byte switched in a frame. The concept is illustrated in Figure 56 and in Figure 57.

TBD

Figure 56. Bonded Virtual Channel in the PUSH Direction (TBD)

TBD

Figure 57. Bonded Virtual Channel in the PULL Direction (TBD)

13.2.3.4 External Buffer Access

13.2.3.4.1 Overview

The T8110's general operation is to gather TDM serial stream data in small internal buffers, which are programmable on a perchannel basis (i.e., virtual channels), and then when full, burst the data to a larger external buffer where a corresponding PCI bus agent (such as a coprocessor) can operate on the larger buffers. For purposes of further discussion, the other PCI bus agent will be referred to as the **user**.

The T8110's onboard storage is 4096 bytes total. For complete usage, this can be defined from as many as 512 buffers of 8 bytes each, to as few as 16 buffers of 256 bytes each, or any combinations that adhere to the following:

- The total allotment does not exceed 4096 bytes.
- The T8110 internal buffer sizes are defined in multiples of DWORDS, from 2 DWORDS (minimum) to 64 DWORDS (maximum).

For a virtual channel in the **push** direction, the T8110 fills its internal buffer with incoming TDM serial stream data. Once the internal buffer for that channel is full, the T8110 initiates a burst transfer of that data to the external buffer. For a virtual channel in the **pull** direction, the T8110 empties its internal buffer to outgoing TDM serial stream data. Once the internal buffer for that channel is empty, the T8110 initiates a burst transfer to fetch more data from the external buffer.

13 Connection Control—Standard and Virtual Channel (continued)

13.2.3.4.2 Descriptor Table

The first portion of a T8110-initiated transfer is to fetch control information associated with a particular virtual channel. The third portion is an update of that control information, with the transfer of data to/from the external buffer in between (refer to Figure 12 on page 35 and to Figure 13 on page 35). The control information is stored in a descriptor table, which exists in the PCI bus address space, and is accessible by both the T8110 and the **user**. The T8110 must have access to the descriptor table's base address, which must be loaded into T8110 registers 0x00110—113 prior to any virtual channel activity. There is one entry in the descriptor table for each of the possible 512 virtual channels. Each entry consists of 8 bytes (2 DWORDS). The maximum space required for the descriptor table is (512*8) = 4 Kbytes. A descriptor table entry contains the following information.

Table 113. Descriptor Table

DWORD	Bits[31:24]					Bits[23:16]	Bits[15:8]		Bits[7:0]
First DWORD	External Buffer Base Address								External Buffer Last Address
Second DWORD	SE	OE	L	UF	UOR[11:0]		GBS[2:0]	TF	TOR[11:0]

- External Buffer Base Address: The base address where the external buffer space for this particular virtual channel begins. Note that only bits [31:12] are defined, which means the external buffer space for a virtual channel is addressed on 4 Kbyte boundaries in the PCI space. This region is read-only for T8110, read-write for the USER.
- External Buffer Last Address: The last address offset in the external buffer space for this particular virtual channel (DWORD-aligned). This region is read-only for T8110, read-write for the USER.
- Control Flags: All flags are read-only for T8110, read-write for the USER.
- SE, Stop Enable: Determines whether the external buffer is treated as circular for this virtual channel (refer to Section 13.2.3.4.4). 0 = the buffer is not circular (T8110 must stop at end of buffer), 1 = buffer is circular (T8110 can roll over at end-of-buffer).
- OE, Overwrite Enable: Determines whether the T8110 can overwrite unread data in the external buffer for this virtual channel (refer to Section 13.2.3.4.4). 0 = overwrite is disabled, 1 = overwrite is enabled.
- L, Lock: Determines whether the T8110 is allowed access to the external buffer for this virtual channel. 0 = not locked, 1 = locked.
- UF: Toggled by the **user** whenever the UOR pointer rolls over at the end of the external buffer. Read-only for the T8110.
- UOR: User-updated offset pointer within the defined 4K external buffer space for this virtual channel. Read-only for the T8110.
- GBS (status): Supplemental information on the status of the external buffer for this virtual channel (refer to Section 13.2.3.4.4). These flags are read-write for both the T8110 and the USER. Refer to Table 114.
- TF: Toggled by the T8110 whenever the TOR pointer rolls over at the end of the external buffer. Read-write for both the T8110 and the **user**.
- TOR: T8110 updated offset pointer within the defined 4K external buffer space for this virtual channel. Read-write for both the T8110 and the **user**.

13 Connection Control—Standard and Virtual Channel (continued)

Table 114. Descriptor Table GBS Status Descriptions

GBS Value	Status Description
000	**User** has intialized the external buffer.
001	T8110 has completed with normal status.
010	T8110 has completed with a boundary condition.
011	T8110 has overwritten a portion of the external buffer unread by **user.**
100	**User** has initialized the T8110 pointer (TF and TOR).
101	T8110 did not complete due to a locked buffer.
110	T8110 did not complete due to stalled buffer (boundary condition).
111	**User** has disabled the external buffer.

13.2.3.4.3 External Buffer

Each virtual channel has an allotment of external buffer space. The external buffer for a virtual channel can be thought of as a FIFO, with the T8110 controlling one side, and the **user** controlling the other. The base address and the last address offset of the external buffer is stored in the descriptor table (first DWORD of the entry for the corresponding virtual channel; see previous section). Each external buffer is defined as 4 Kbytes, with the base address at 4 Kbyte boundaries. An external buffer is not limited to exactly 4 Kbytes—it can be smaller or larger, which requires special on-the-fly manipulation of the descriptor table by the **user** side (refer to Section 13.2.3.4.4). The only basic requirement for the external buffer size is that it be an integral multiple of the T8110 internal buffer size for a given virtual channel.

13.2.3.4.4 Transfer Protocol

The transfer mechanism between the T8110 and the **user** is three T8110-initiated PCI transfers: descriptor table fetch, external buffer data transfer, and descriptor table update (refer to Figure 12 and Figure 13). The second transfer (external buffer data transfer) would normally occur; however, it may not occur if the state of the descriptor table control and status flags shows the external buffer not accessible by the T8110.

13.2.3.4.4.1 Descriptor Table Fetch

T8110 uses the descriptor table base address stored in its control register field (address 0x00110-113), and adds an address offset determined by which of the possible 512 virtual channels initiated the action to create the descriptor table address for that channel. The transfer is a PCI memory read burst of 2 DWORDS. The descriptor table contents is decoded and a number of calculations are performed on the descriptor table data. The results of these calculations determine the sequence of further PCI transfers (i.e., whether or not to skip the second external buffer data transfer, and what value(s) to update the descriptor table with). Refer to Figure 58 and Table 114.

TBD

Figure 58. Descriptor Table Fetch Decode (TBD)

13.2.3.4.5 External Buffer Data Transfer

After the descriptor table fetch, the T8110 then does the following:

- It either dumps its internal buffer of gathered TDM data to the external buffer space (**push**, a PCI memory write burst of internal buffer size)

13 Connection Control—Standard and Virtual Channel (continued)

- Or fetches data from the external buffer space into its internal buffer for outgoing TDM data (**pull**, a PCI memory read burst of internal buffer size).
- Or skips the external buffer data transfer (based on descriptor table status, such as a pointer stall, end-of-buffer stall, buffer locked status, which indicate that the external buffer is not currently available to the T8110).

13.2.3.4.6 Descriptor Table Update

The T8110 then updates the descriptor table (second DWORD only), with values calculated based on the descriptor table fetch results. The only allowable portions writable by T8110 include the GBS status, TF and TOR. The transfer is a PCI memory write of 1 DWORD.

13.2.3.5 T8110 Packet Switching, Circuit Operation

Each programmed T8110 virtual channel operates independently, and tracks both its current position in the T8110 internal buffer space and the buffer full (**push**) or empty (**pull**) status, in the scratchpad portion of the virtual channel memory (refer to Section 13.2.1.3). Upon determination of a full (or empty) internal buffer, that channel places an entry into a notify queue, and sets a bit in the notify pending memory. Entries in the notify queue get translated into the T8110-initiated external buffer transfer protocol (refer to Section 13.2.3.4.4). Upon completion of that transfer protocol, the notify pending memory bit for that channel is reset.

13.2.3.5.1 System Errors Due to Packet Switching

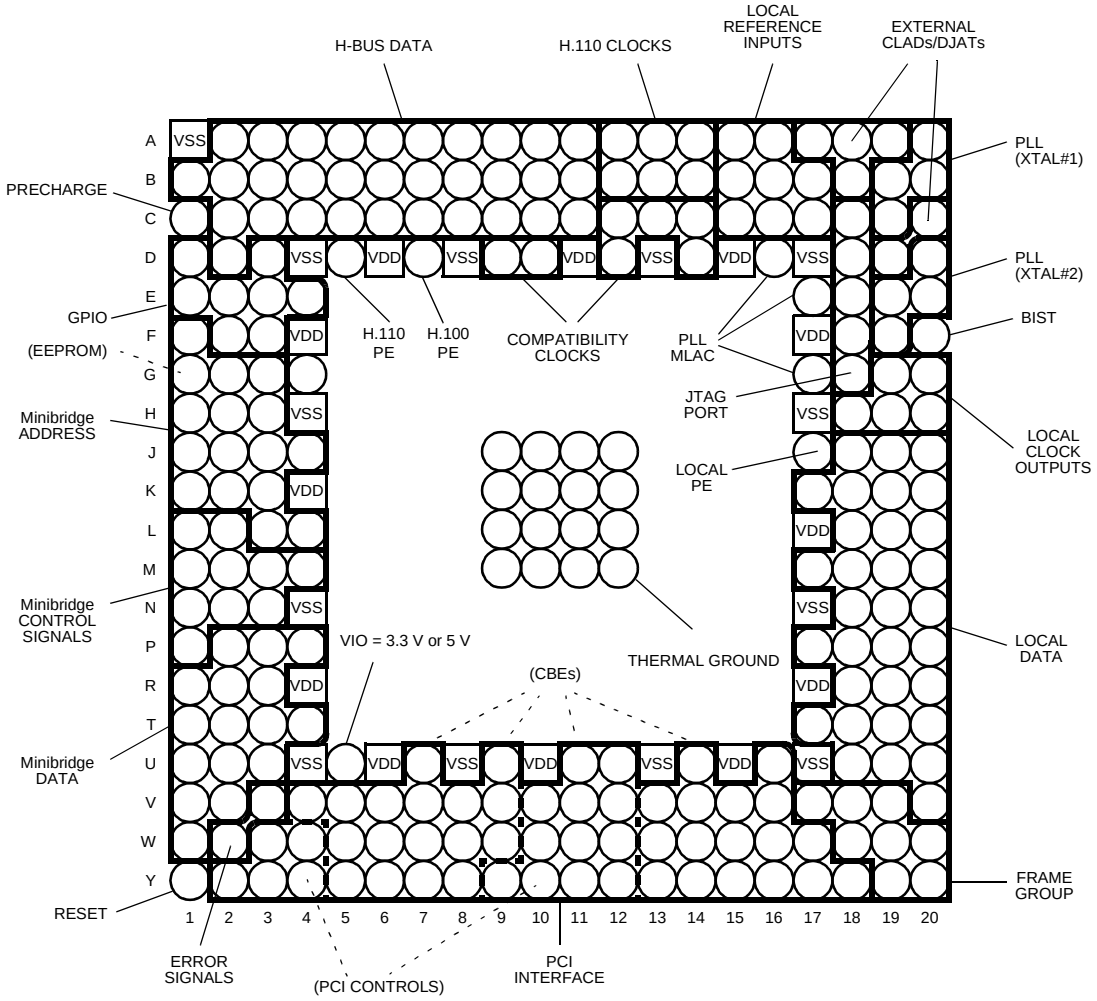
TBD

14 Electrical Characteristics

TBD - similar to information in the existing *Ambassador* data sheet.

15 Package Outline

15.1 Pin and Pad Assignments

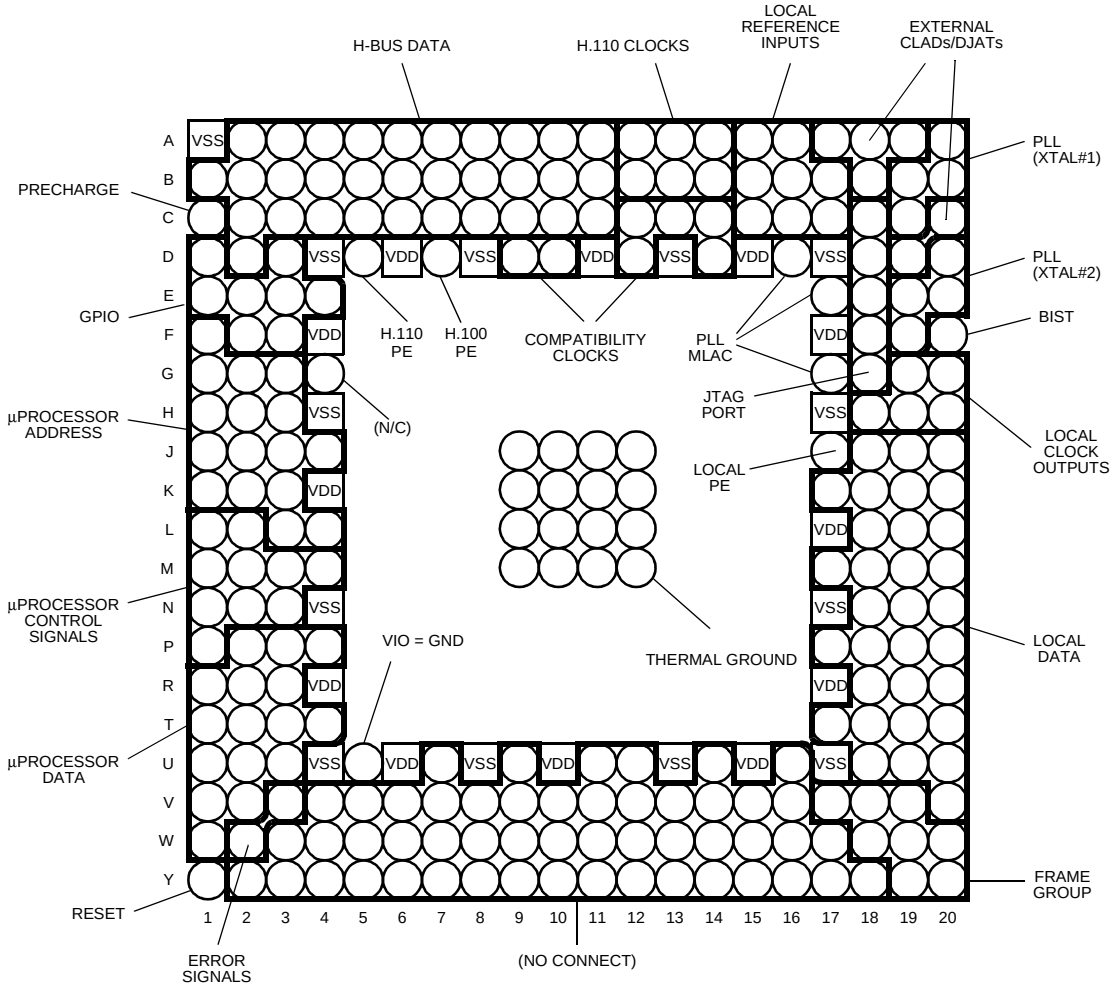


T8110 Pin Assignment by Mayor Function, PCI Interface Enabled, Microprocessor Port Disabled

5-8906F(a).

Figure 59. T8110 Pins by Functional Group, PCI Interface

15 Package Outline (continued)



T8110 Pin Assignment by Mayor Function, Microprocessor Port Enabled, PCI I/F Disabled

5-8906F(b).

Figure 60. T8110 Pins by Functional Group, ISA Interface

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) 7000 582 368**, FAX (44) 1189 328 148
Technical Inquiries: GERMANY: **(49) 89 95086 0** (Munich), UNITED KINGDOM: **(44) 1344 865 900** (Ascot),
FRANCE: **(33) 1 40 83 68 00** (Paris), SWEDEN: **(46) 8 594 607 00** (Stockholm), FINLAND: **(358) 9 4354 2800** (Helsinki),
ITALY: **(39) 02 6608131** (Milan), SPAIN: **(34) 1 807 1441** (Madrid)

June 2000
DS00-012CTI

microelectronics group

Lucent Technologies
Bell Labs Innovations