Advisory
September 2001

Ambassador ® T8110 Version History

Introduction

The purpose of this advisory is to provide information on the different versions of the *Ambassador* T8110.

T8110 Version 1

Models of the T8110 V1 had two device issues. The two device issues only affect the microprocessor interface and packet switching capabilities. The T8110 V1 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.

T8110 version 1 models can be identified by the markings on the device or by reading the version ID register. If the last line of the device markings is a 7 digit number followed by no version number, then the device is a version 1. Reading the version ID register 0x00128 will read back a value of 01h, indicating the device is version 1.

Samples of version 1 are no longer available (version 2 samples are now available).

T8110 Version 2

Models of the T8110 V2 have one device issue. The device issue only affects the packet switching capabilities. The T8110 V2 can function as a 4096 connection standard telephony switch when using either PCI or microprocessor interface.

Issue 1 (from version 1): Fixed. The microprocessor interface issue has been resolved.

Issue 2 (from version 1): Will be fixed in version 3.

T8110 version 2 models can be identified by the markings on the device or by reading the version ID register. If the last line of the device markings is a 7 digit number followed by V2, then the device is a version 2. Reading the version ID register 0x00128 will read back a value of 02h, indicating the device is a version 2.

Samples of version 2 are currently available.

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@agere.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: Agere Systems Hong Kong Ltd., Suites 3201 & 3210-12, 32/F, Tower 2, The Gateway, Harbour City, Kowloon
Tel. (852) 3129-2000, FAX (852) 3129-2020
CHINA: **(86) 21-5047-1212** (Shanghai), **(86) 10-6522-5566** (Beijing), **(86) 755-695-7224** (Shenzhen)
JAPAN: **(81) 3-5421-1600** (Tokyo), KOREA: **(82) 2-767-1850** (Seoul), SINGAPORE: **(65) 778-8833**, TAIWAN: **(886) 2-2725-5858** (Taipei)
EUROPE: **Tel. (44) 7000 624624**, FAX (44) 1344 488 045

September 2001
AY01-038CTI (Replaces AY01-021CTI and must accompany DS00-434CTI)

Data Sheet
May 2001

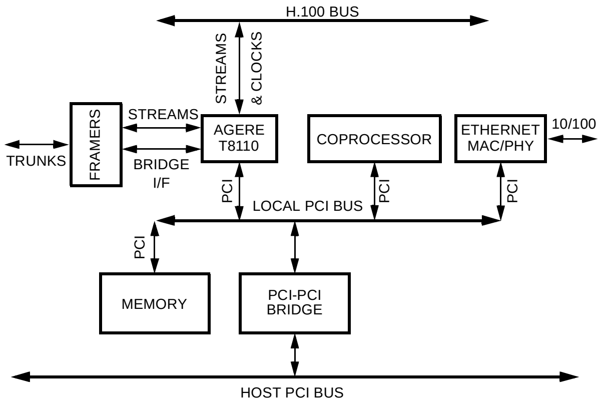
Ambassador® T8110 PCI-Based H.100/H.110 Switch and Packet Payload Engine

1 Introduction

The T8110 is Agere Systems Inc.'s newest addition to the *Ambassador* series of computer telephony integrated circuits. This device not only has the capabilities of previous members of the *Ambassador* series 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. Packets may also be switched between the local PCI bus and local TDM streams. 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.

1.1 Features

- 4,096-connection unified switch
- Packet payload engine supports up to 512 virtual channels
- Full H.100/H.110 support (32 data lines, all clock modes)
- 32 local I/O lines (2, 4, 8, or 16 Mbits/s)
- PCI interface: combined master/slave with burst
- Microprocessor interface: *Motorola*[*]/*Intel*[†] modes
- Minibridge with programmable chip selects
- Interrupt controller with external inputs
- Eight independent general-purpose I/O lines
- Eight independently programmed framing signals
- Four local clocks
- T1/E1 rate adaptation
- Two clock-fallback modes
- Stratum 4/4E and AT&T 62411 MTIE compliant
- Incorporates 38 H.100 and 34 H.110 termination resistors
- Subrate switching of 4 bits, 2 bits, or 1 bit
- Backward compatible to all T810x devices
- JTAG/boundary-scan testing support
- BSDL files available
- Assists H.110 hot swap
- Single 3.3 V supply with 5 V tolerant inputs and TTL compatible outputs
- 272 PBGA package
- Evaluation boards available, PCI and *CompactPCI*[‡] Hot Swap



5-8921F

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

---

[*] *Motorola* is a registered trademark of Motorola, Inc.
[†] *Intel* is a registered trademark of Intel Corporation.
[‡] *CompactPCI* is a registered trademark of the PCI Industrial Computer Manufacturers Group.

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 **Page**

Table of Contents (continued)

Contents **Page**

2 Pin 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	In	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#	In	1	PCI bus grant.
PCI_IDSEL	In	1	PCI bus initialization device select.
PCI_INTA#	Out	1	PCI bus interrupt.
PCI_IRDY#	I/O	1	PCI bus initiator ready.
PCI_LOCK#	In	1	PCI bus lock.
PCI_PAR	I/O	1	PCI bus parity.
PCI_PERR#	I/O	1	PCI bus parity error.
PCI_REQ#	Out	1	PCI bus request.
PCI_RST#	In	1	PCI bus reset.
PCI_SERR#	Out	1	PCI bus system error.
PCI_STOP#	I/O	1	PCI bus stop.
PCI_TRDY#	I/O	1	PCI bus target ready.

Table 2. Minibridge Interface Signals

Signal	I/O	Width	Minibridge Function	Microprocessor 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 output.	RDn(DSn) in.
MB_WR	I/O	1	Write strobe output.	WRn(R/Wn) in.
MB_CS0	I/O	1	Chip select 0 output.	Address[16] in.
MB_CS1	I/O	1	Chip select 1 output.	Address[17] in.
MB_CS2	I/O	1	Chip select 2 output.	Address[18] in.
MB_CS3	I/O	1	Chip select 3 output.	Address[19] in.
MB_CS4	I/O	1	Chip select 4 output.	CSn in.
MB_CS5	I/O	1	Chip select 5 output.	Word/byte select in.
MB_CS6	Out	1	Chip select 6 output.	RDY(DTACKn) out.
MB_CS7	I/O	1	Chip select 7 output.	*Intel**/*Motorola*† select in.

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

2 Pin Description (continued)

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

Signal	I/O	Width	Function
VPRECHARGE	In	1	Precharge voltage for pull-downs, H.110 bus signals: CT_D, CT_NETREF1, CT_NETREF2.
H110_ENABLE	In	1	Pull-down enable for H.110 bus signals: CT_D, CT_NETREF1, CT_NETREF2.
H100_ENABLE	In	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.

* *MVIP* is a trademark of Natural MicroSystems Corporation.

Table 4. L-Bus (Local) Interface Signals

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

Table 5. Clock Circuit Interface Signals

Signal	I/O	Width	Function
XTAL1_IN	In	1	Crystal oscillator #1 input (16.384 MHz).
XTAL1_OUT	Out	1	Crystal oscillator #1 feedback.
XTAL2_IN	In	1	Crystal oscillator #2 input (6.176 MHz or 12.352 MHz).
XTAL2_OUT	Out	1	Crystal oscillator #2 feedback.

2 Pin Description (continued)

Table 5. Clock Circuit Interface Signals (continued)

Signal	I/O	Width	Function
LREF	In	8	Local clock reference inputs.
TCLK_OUT	Out	1	Internal chip clock output.
PRI_REF_OUT	Out	1	Main divider reference out for CLAD/DJAT.
PRI_REF_IN	In	1	CLAD/DJAT reference in for APLL1.
NR1_SEL_OUT	Out	1	CT_NETREF1 selection out for CLAD/DJAT.
NR1_DIV_IN	In	1	CLAD/DJAT reference in for CT_NETREF1 divider.
NR2_SEL_OUT	Out	1	CT_NETREF2 selection out for CLAD/DJAT.
NR2_DIV_IN	In	1	CLAD/DJAT reference in for CT_NETREF2 divider.

Table 6. GPIO Interface Signals

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

Table 7. Miscellaneous Interface Signals

Signal	I/O	Width	Function
RESET#	In	1	Chip reset.
SYSERR	Out	1	System error indicator.
CLKERR	Out	1	Clocking error indicator.
LPUE	In	1	Pull-up enable for signals: FG, GP, L_D, LREF, MB_D, NR1_DIV_IN, NR2_DIV_IN, PRI_REF_IN.
EE_CS	Out	1	EEPROM chip select.
VIO/µP_SELECT	In	1	PCI bus environment, apply GND for microprocessor interface, apply 3.3 V or 5 V for PCI interface.

Table 8. JTAG Signals

Signal	I/O	Width	Function
TRST#	In	1	JTAG reset.
TCK	In	1	JTAG clock.
TMS	In	1	JTAG mode select.
TDI	In	1	JTAG data in.
TDO	Out	1	JTAG data out.

2 Pin Description (continued)

2.2 T8110 Pinout Information

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

Note: The pull-up/down column in the following table is defined as follows:

- 20 kΩ down—20 kΩ pull-down resistor is always in-circuit.
- 50 kΩ up—50 kΩ pull-up resistor is always in-circuit.
- LPUE: 50 kΩ up—when LPUE = 1, a 50 kΩ pull-up resistor is in-circuit.
- Enabled: 50 kΩ up/20 kΩ Vpre—when H100_ENABLE = 1, a 50 kΩ pull-up resistor is in-circuit (see Figure 2 on page 18). When H110_ENABLE = 1, a 20 kΩ pull-down resistor from the VPRECHARGE input to this signal is in-circuit.

Table 9. T8110 Pinouts

PCI Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down**
Y18	PCI_AD0	PCI I/O	—
W17	PCI_AD1	PCI I/O	—
V16	PCI_AD2	PCI I/O	—
U16	PCI_AD3	PCI I/O	—
Y17	PCI_AD4	PCI I/O	—
Y16	PCI_AD5	PCI I/O	—
W16	PCI_AD6	PCI I/O	—
V15	PCI_AD7	PCI I/O	—
Y15	PCI_AD8	PCI I/O	—
W15	PCI_AD9	PCI I/O	—
W14	PCI_AD10	PCI I/O	—
V14	PCI_AD11	PCI I/O	—
Y14	PCI_AD12	PCI I/O	—
Y13	PCI_AD13	PCI I/O	—
W13	PCI_AD14	PCI I/O	—
V13	PCI_AD15	PCI I/O	—
Y9	PCI_AD16	PCI I/O	—
W9	PCI_AD17	PCI I/O	—
V9	PCI_AD18	PCI I/O	—
V8	PCI_AD19	PCI I/O	—
Y8	PCI_AD20	PCI I/O	—
W8	PCI_AD21	PCI I/O	—
W7	PCI_AD22	PCI I/O	—
V7	PCI_AD23	PCI I/O	—
Y7	PCI_AD24	PCI I/O	—
Y6	PCI_AD25	PCI I/O	—

2 Pin Description (continued)

Table 9. T8110 Pinouts (continued)

PCI Interface (continued)			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down** (see note on page 11)
W6	PCI_AD26	PCI I/O	—
V6	PCI_AD27	PCI I/O	—
Y5	PCI_AD28	PCI I/O	—
W5	PCI_AD29	PCI I/O	—
V5	PCI_AD30	PCI I/O	—
V4	PCI_AD31	PCI I/O	—
U14	PCI_CBE0#	PCI I/O	—
U12	PCI_CBE1#	PCI I/O	—
U9	PCI_CBE2#	PCI I/O	—
U7	PCI_CBE3#	PCI I/O	—
Y3	PCI_CLK	PCI input	—
W11	PCI_DEVSEL#	PCI I/O	—
Y10	PCI_FRAME#	PCI I/O	—
W4	PCI_GNT#	PCI input	—
W10	PCI_IDSEL	PCI input	—
Y4	PCI_INTA#	PCI output/open drain	—
Y11	PCI_IRDY#	PCI I/O	—
V10	PCI_LOCK#	PCI input	—
U11	PCI_PAR	PCI I/O	—
W12	PCI_PERR#	PCI I/O	—
W3	PCI_REQ#	PCI output	—
Y2	PCI_RST#	PCI input	—
V12	PCI_SERR#	PCI output/open drain	—
V11	PCI_STOP#	PCI I/O	—
Y12	PCI_TRDY#	PCI I/O	—
Minibridge Interface			
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
K3	MB_A10/UP_A10	8 mA I/O-Schmitt	20 kΩ down
J3	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

2 Pin Description (continued)

Table 9. T8110 Pinouts (continued)

Minibridge Interface (continued)			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down** (see note on page 11)
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
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	External pull-up required
N3	MB_CS7/IM_SEL	8 mA I/O-Schmitt	LPUE: 50 kΩ up
H-Bus Interface			
C1	VPRECHARGE	Op amp noninvert	—
D5	H110_ENABLE	Input	20 kΩ down
D7	H100_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
C10	CT_D2	PCI I/O	Enabled: 50 kΩ up/20 kΩ Vpre
C11	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

2 Pin Description (continued)

Table 9. T8110 Pinouts (continued)

H-Bus Interface (continued)			
Ball	**Pin Name**	**Buffer Type**	**Pull-Up/Down** (see note on page 11)
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
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
A12	/CT_FRAME_A	24 mA I/O-Schmitt	Enabled: 50 kΩ up
B13	CT_C8_B	24 mA I/O-Schmitt	Enabled: 50 kΩ up
B12	/CT_FRAME_B	24 mA I/O-Schmitt	Enabled: 50 kΩ up
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

2 Pin Description (continued)

Table 9. T8110 Pinouts (continued)

L-Bus Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull Up/Down** (see note on page 11)
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
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
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	8 mA 3-state	—
H19	L_SC1	8 mA 3-state	—
H18	L_SC2	8 mA 3-state	—
G19	L_SC3	8 mA 3-state	—
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

2 Pin Description (continued)

Table 9. T8110 Pinouts (continued)

Clock Circuit Interface			
Ball	**Pin Name**	**Buffer Type**	**Pull Up/Down** (see note on page 11)
B20	XTAL1_IN	Input	—
C19	XTAL1_OUT	Crystal feedback	—
E20	XTAL2_IN	Input	—
F19	XTAL2_OUT	Crystal feedback	—
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	8 mA 3-state	—
A17	PRI_REF_OUT	8 mA 3-state	—
A18	PRI_REF_IN	Input-Schmitt	LPUE: 50 kΩ up
B18	NR1_SEL_OUT	8 mA 3-state	—
A19	NR1_DIV_IN	Input-Schmitt	LPUE: 50 kΩ up
D19	NR2_SEL_OUT	8 mA 3-state	—
C20	NR2_DIV_IN	Input-Schmitt	LPUE: 50 kΩ up
GPIO Interface			
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
F3	GP5	8 mA I/O-Schmitt	LPUE: 50 kΩ up
E3	GP6	8 mA I/O-Schmitt	LPUE: 50 kΩ up
E4	GP7	8 mA I/O-Schmitt	LPUE: 50 kΩ up
Miscellaneous Interfaces			
Y1	RESET#	Input-Schmitt	50 kΩ up
V3	SYSERR	8 mA 3-state	—
W2	CLKERR	8 mA 3-state	—
J17	LPUE	Input	50 kΩ up
G4	EE_CS	8 mA 3-state	—
U5	VIO/µP_SELECT	—	20 kΩ down
JTAG Interface			
C18	TRST#	Input-Schmitt	50 kΩ up
E18	TCK	Input-Schmitt	50 kΩ up
D18	TMS	Input-Schmitt	50 kΩ up
F18	TDI	Input-Schmitt	50 kΩ up
G18	TDO	4 mA 3-state	—

2 Pin Description (continued)

Table 9. T8110 Pinouts (continued)

Power			
Ball	**Pin Name**	**Buffer Type**	**Pull Up/Down**
B19	APLL1VDD	Analog VDD	—
E19	APLL2VDD	Analog VDD	—
D6	VDD	—	—
D11	VDD	—	—
D15	VDD	—	—
F4	VDD	—	—
F17	VDD	—	—
K4	VDD	—	—
L17	VDD	—	—
R4	VDD	—	—
R17	VDD	—	—
U6	VDD	—	—
U10	VDD	—	—
U15	VDD	—	—
Ground			
A1	VSS	—	—
D4	VSS	—	—
D8	VSS	—	—
D13	VSS	—	—
D17	VSS	—	—
H4	VSS	—	—
H17	VSS	—	—
N4	VSS	—	—
N17	VSS	—	—
U4	VSS	—	—
U8	VSS	—	—
U13	VSS	—	—
U17	VSS	—	—
Thermal Ground			
J9—12	—	—	—
K9—12	—	—	—
L9—12	—	—	—
M9—12	—	—	—
No Connects			
A20	No connects must be left unconnected.		
D16			
D20			
E17			
F20			
G17			

2 Pin Description (continued)

2.3 Special Buffer Requirements

2.3.1 H1x0 Bus Signal Internal Pull-Up/Pull-Down

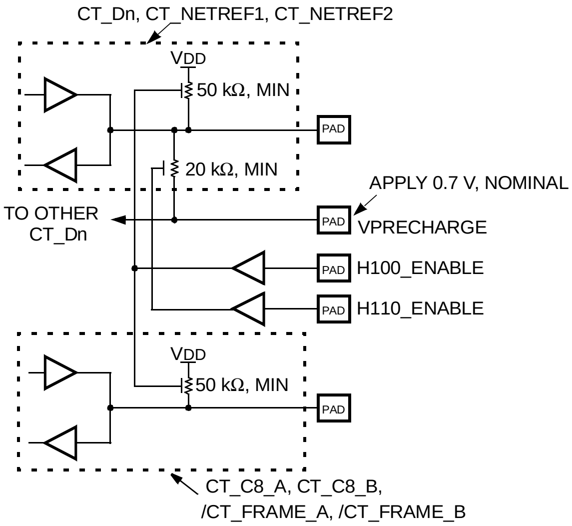
The H1x0 bus pins require special consideration for H.100 and H.110 usage. There are two control pins to select between various internal bus pull-ups/pull-downs, as shown below:

- H100_ENABLE. Enables internal 50 kΩ pull-ups on CT_Dn, CT_NETREF1, CT_NETREF2, CT_C8_A, CT_C8_B, /CT_FRAME_A, and /CT_FRAME_B signals.
- H110_ENABLE. Enables internal 20 kΩ pull-downs on all 32 CT_Dn signals, CT_NETREF1, and CT_NETREF2 to the VPRECHARGE signal.

Note: The two H1x0 enables are active-high. Only one or the other should ever be asserted.

Warning: Do not assert both at the same time.

Please refer to Figure 2 for more detail.



5-9611 (F)

Figure 2. T8110 Pull-Up/Pull-Down Arrangement for H1x0 Pins

2.3.2 Local Bus Signal Internal Pull-Up

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.

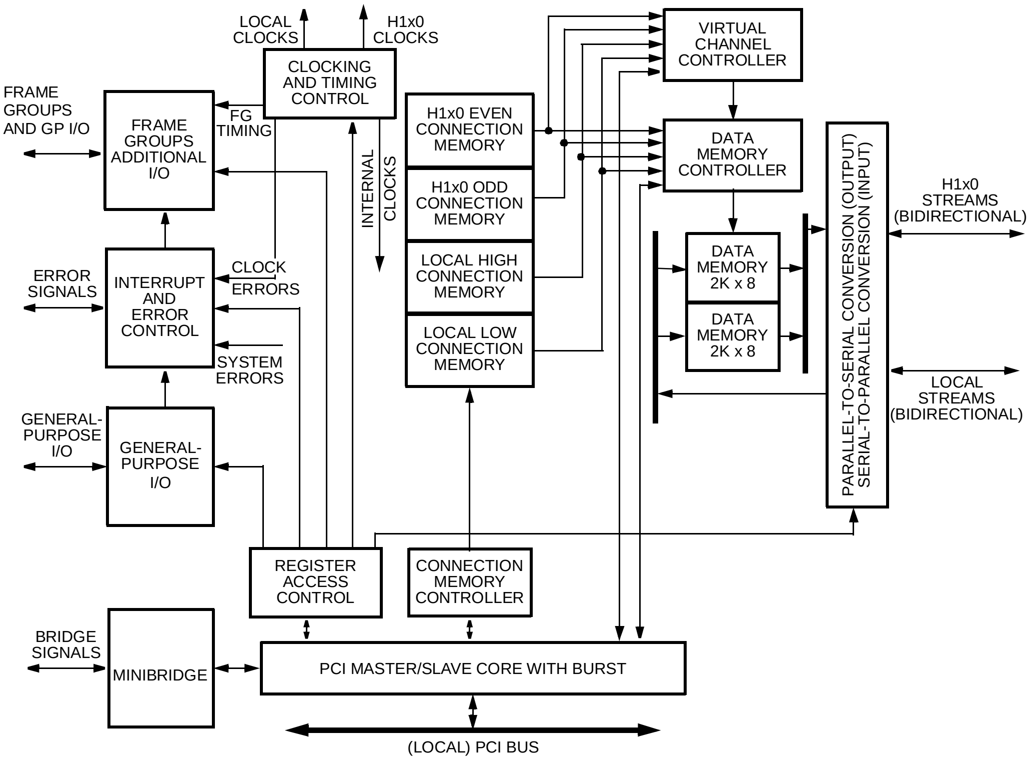
3 Main Architectural Features

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 on page 22) and microprocessor (see Section 5 on page 38).

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 14, starting on page 136, for 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 11, starting on page 107.

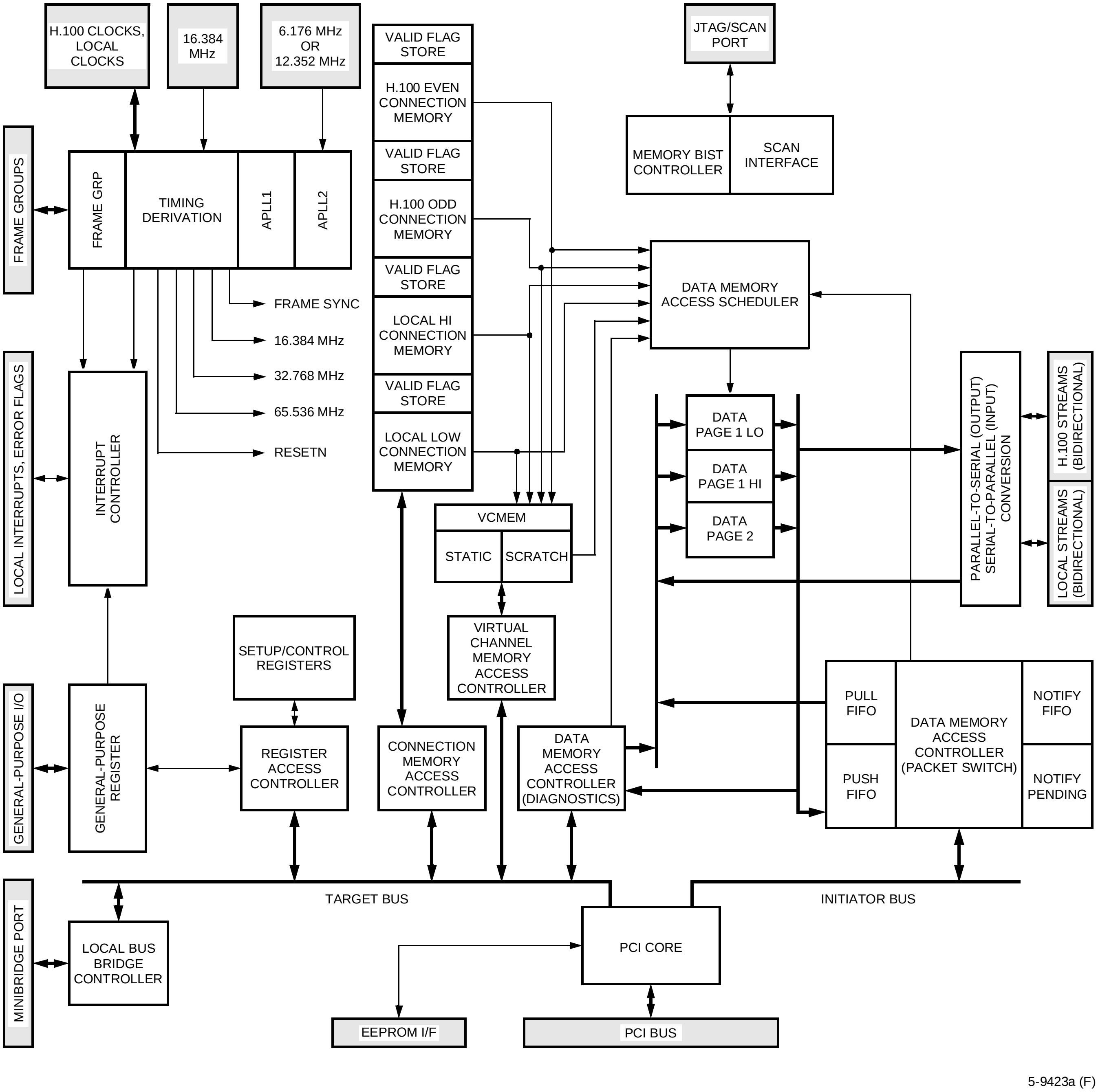
The microprocessor bus interface allows the T8110 to perform standard H-bus/L-bus switching (i.e., there is no packet payload switching between the H-bus/L-bus and the microprocessor port). With this architecture, the minibridge port is used as the microprocessor bus port, and the PCI interface is ignored.



5-8920 (F)

Figure 3. T8110 Block Diagram

3 Main Architectural Features (continued)



5-9423a (F)

Figure 4. T8110 Architecture 1—PCI Bus Interface

3 Main Architectural Features (continued)

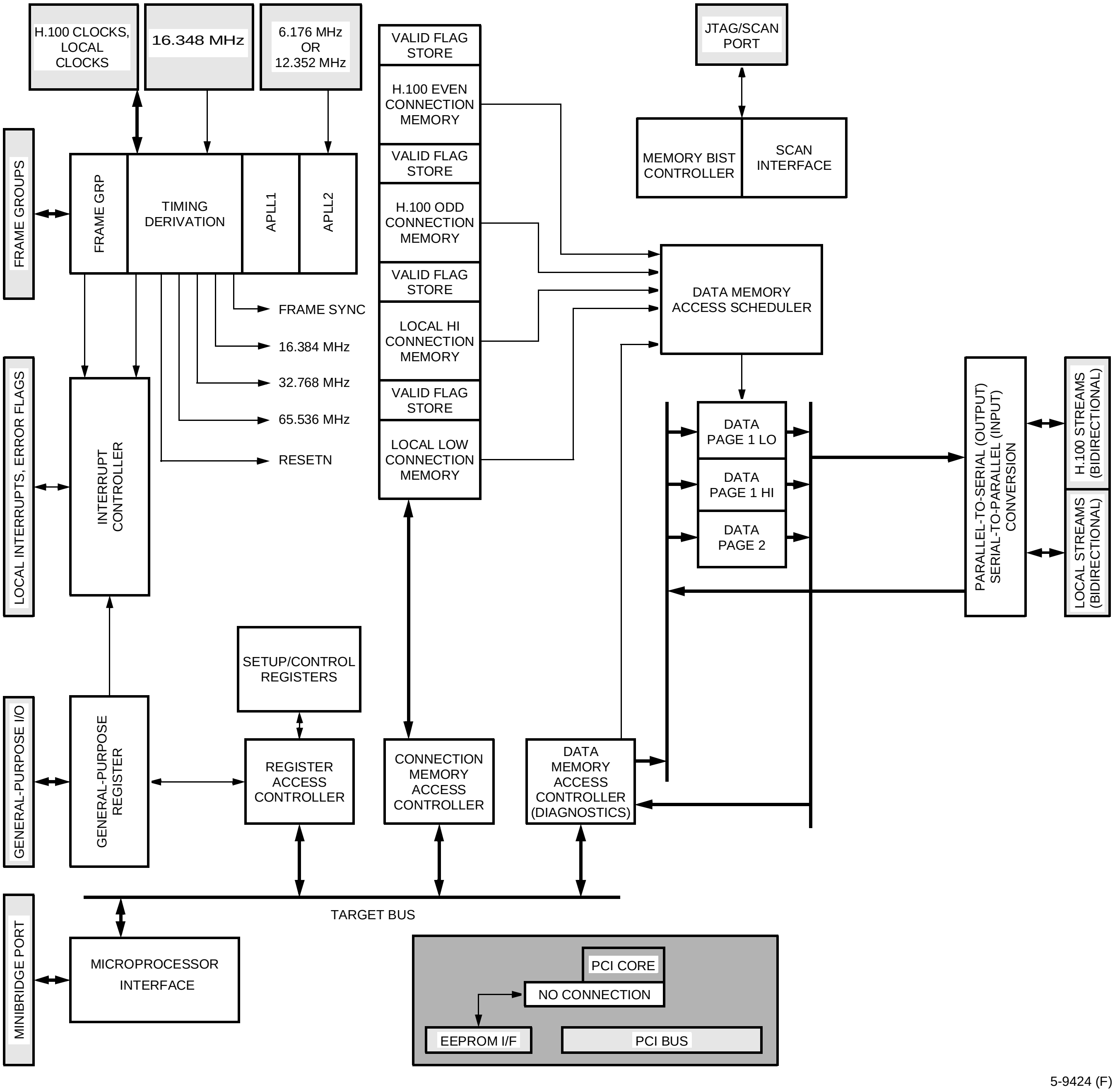


Figure 5. T8110 Architecture 2—Microprocessor Bus Interface

4 PCI Interface

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

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

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 8 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 11, starting on page 107.

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 H1x0 bus and passing it to memory, or when data is retrieved from memory and sent to the H1x0 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 H1x0 bus and local time slots can be routed to and from the PCI bus as well.

4.1 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 10.

Table 10. 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
Minibridge	—	0x70000—0x7FFFF
Reserved	—	0x80000—0xFFFFF

4 PCI Interface (continued)

4.1.1 PCI Interface Registers

Table 11. PCI Interface Registers Map

DWORD Address (20 bits)	Section Cross Reference	Registers			
		Byte 3	Byte 2	Byte 1	Byte 0
0x00100	6.1.1, 6.1.2	Master enable	Reserved	Reset select	Soft reset
0x00104	6.1.3, 6.1.4	Phase alignment select	Clock register access select	Data memory mode select	VCSTART
0x00108	6.1.4	Fallback trigger, upper	Fallback trigger, lower	Fallback type select	Fallback control
0x0010C	6.1.4	Watchdog EN, upper	Watchdog EN, lower	Watchdog select, NETREF	Watchdog select, C8
0x00110	14.2.3.4.2	External buffers descriptor table—base address register[31:0]			
0x00114	4.1.5	Reserved	Failsafe threshold low	Failsafe enable and status	Failsafe control
0x00120	6.2.1	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	6.2.2, 6.2.5	Status 7, system errors, upper	Status 6, system errors, lower	Status 5	Status 4
0x00128	6.2.6	Device ID, upper	Device ID, lower	Reserved	Version ID
0x0012C	6.2.6	Reserved	Reserved	Status 9	Status 8
0x00140	13.1	Diag3	Diag2	Diag1	Diag0
0x00144	13.1	Diag7	Diag6	Diag5	Diag4
0x00148	13.1	Reserved	Reserved	Reserved	Diag8
0x00200	7.1	APLL1 rate	APLL1 input selector	Main divider	Main input selector
0x00204	7.1	APLL2 rate	Reserved	Resource divider	Main inversion select
0x00208	7.1	DPLL1 rate	DPLL1 input selector	Reserved	LREF input select
0x0020C	7.1	DPLL2 rate	DPLL2 input selector	Reserved	LREF inversion select
0x00210	7.1	Reserved	NETREF1 LREF select	NETREF1 divider	NETREF1 input selector
0x00214	7.1	Reserved	NETREF2 LREF select	NETREF2 divider	NETREF2 input selector
0x00220	7.2	C8 output rate	/FR_COMP width	NETREF output enables	Master output enables
0x00224	7.2	SCLK output rate	TCLK select	Reserved	CCLK output enables
0x00228	7.2	L_SC3 select	L_SC2 select	L_SC1 select	L_SC0 select
0x00300	10.1	H-bus rate H/G	H-bus rate F/E	H-bus rate D/C	H-bus rate B/A
0x00320	10.2	L-bus rate H/G	L-bus rate F/E	L-bus rate D/C	L-bus rate B/A

4 PCI Interface (continued)

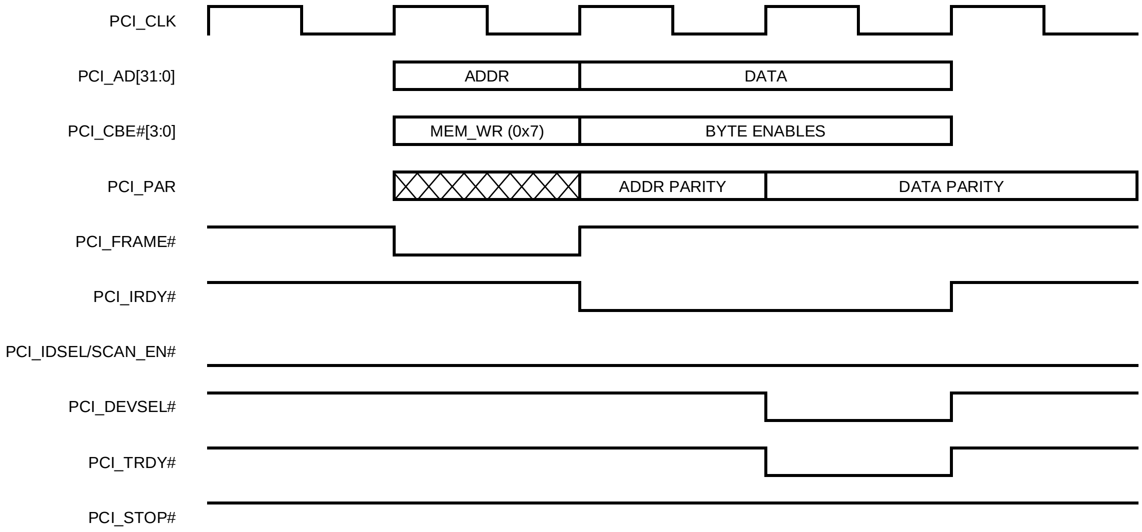
Table 11. PCI Interface Registers Map (continued)

DWORD Address (20 bits)	Section Cross Reference	Registers			
		Byte 3	Byte 2	Byte 1	Byte 0
0x00400	8.1	FG0 rate	FG0 width	FG0 upper start	FG0 lower start
0x00410	8.1	FG1 rate	FG1 width	FG1 upper start	FG1 lower start
0x00420	8.1	FG2 rate	FG2 width	FG2 upper start	FG2 lower start
0x00430	8.1	FG3 rate	FG3 width	FG3 upper start	FG3 lower start
0x00440	8.1	FG4 rate	FG4 width	FG4 upper start	FG4 lower start
0x00450	8.1	FG5 rate	FG5 width	FG5 upper start	FG5 lower start
0x00460	8.1	FG6 rate	FG6 width	FG6 upper start	FG6 lower start
0x00470	8.1	FG7 rate	FG7 width	FG7 upper start	FG7 lower start
0x00474	8.2	FG7 mode upper	FG7 mode lower	FG7 counter high byte	FG7 counter low byte
0x00480	8.3	Reserved	FGIO R/W	FGIO read mask	FGIO data register
0x00500	9.1	GPIO override	GPIO R/W	GPIO read mask	GPIO data register
0x00600	12.1	FGIO interrupt polarity	Reserved	FGIO interrupt enable	FGIO interrupt pending
0x00604	12.1	GPIO interrupt polarity	Reserved	GPIO interrupt enable	GPIO interrupt pending
0x00608	12.1	System interrupt enable, upper	System interrupt enable, lower	System interrupt pending, upper	System interrupt pending, lower
0x0060C	12.1	Clock interrupt enable, upper	Clock interrupt enable, lower	Clock interrupt pending, upper	Clock interrupt pending, lower
0x00610	12.1	CLKERR output select	SYSERR output select	PCI_INTA output select	Arbitration control
0x00614	12.1	CLKERR pulse width	SYSERR pulse width	Reserved	Reserved
0x006FC	12.1	In-service, byte 3	In-service, byte 2	In-service, byte 1	In-service, byte 0
0x00700	11.1	CS0 address setup wait	CS0 read hold wait	CS0 read width wait	CS0 read setup wait
0x00704	11.1	CS0 address hold wait	CS0 write hold wait	CS0 write width wait	CS0 write setup wait
0x00710	11.1	CS1 address setup wait	CS1 read hold wait	CS1 read width wait	CS1 read setup wait
0x00714	11.1	CS1 address hold wait	CS1 write hold wait	CS1 write width wait	CS1 write setup wait
0x00720	11.1	CS2 address setup wait	CS2 read hold wait	CS2 read width wait	CS2 read setup wait
0x00724	11.1	CS2 address hold wait	CS2 write hold wait	CS2 write width wait	CS2 write setup wait
0x00730	11.1	CS3 address setup wait	CS3 read hold wait	CS3 read width wait	CS3 read setup wait
0x00734	11.1	CS3 address hold wait	CS3 write hold wait	CS3 write width wait	CS3 write setup wait
0x00740	11.1	CS4 address setup wait	CS4 read hold wait	CS4 read width wait	CS4 read setup wait
0x00744	11.1	CS4 address hold wait	CS4 write hold wait	CS4 write width wait	CS4 write setup wait
0x00750	11.1	CS5 address setup wait	CS5 read hold wait	CS5 read width wait	CS5 read setup wait

4 PCI Interface (continued)

Table 11. PCI Interface Registers Map (continued)

DWORD Address (20 bits)	**Section Cross Reference**	**Registers**			
		Byte 3	**Byte 2**	**Byte 1**	**Byte 0**
0x00754	11.1	CS5 address hold wait	CS5 write hold wait	CS5 write width wait	CS5 write setup wait
0x00760	11.1	CS6 address setup wait	CS6 read hold wait	CS6 read width wait	CS6 read setup wait
0x00764	11.1	CS6 address hold wait	CS6 write hold wait	CS6 write width wait	CS6 write setup wait
0x00770	11.1	CS7 address setup wait	CS7 read hold wait	CS7 read width wait	CS7 read setup wait
0x00774	11.1	CS7 address hold wait	CS7 write hold wait	CS7 write width wait	CS7 write setup wait
0x00780	11.1	Reserved	Reserved	RD-WR strobe inversion	CS strobe inversion



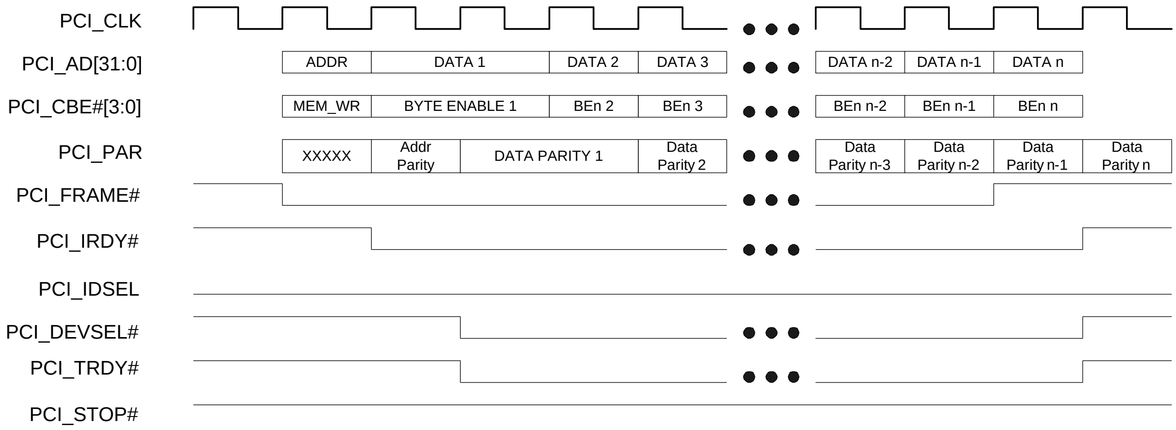
5-9612 (F)

Notes:
T8110 PCI I/F has medium decode speed. There is always a two-cycle turnaround between assertion of PCI_ FRAME# and assertion of PCI_DEVSEL#.

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

Figure 6. T8110 PCI Interface—Single Write Cycle

4 PCI Interface (continued)



Notes:

T8110 PCI I/F has medium decode speed. There is always a two-cycle turnaround between assertion of PCI_FRAME# and assertion of PCI_DEVSEL#.

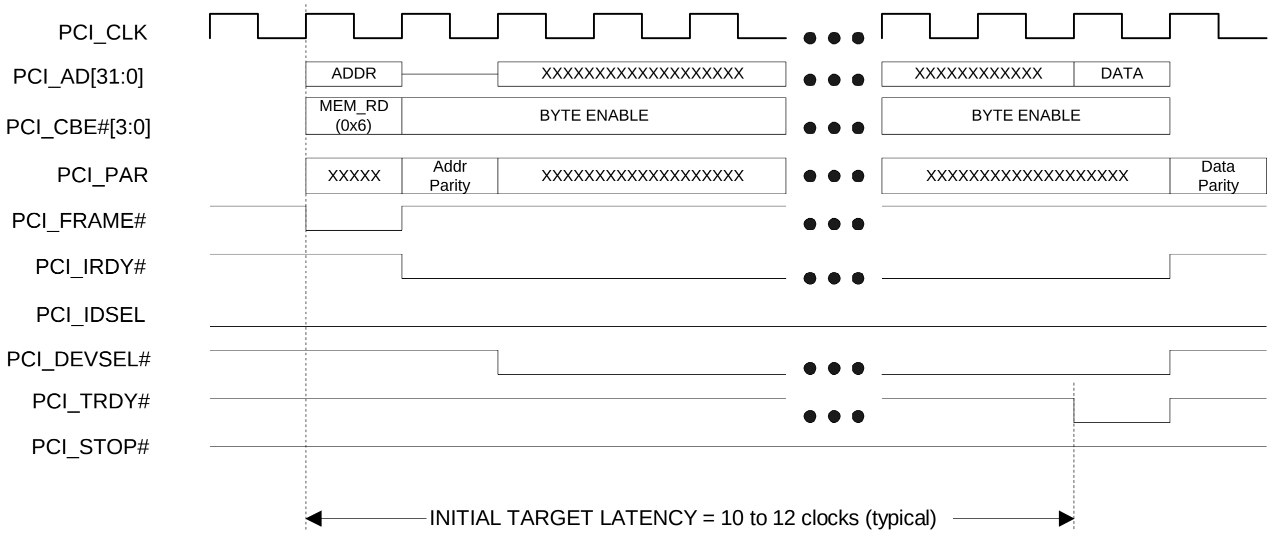
All memory writes get posted to the T8110. Turnaround time for the first data phase write is three PCI clocks.

PCI core write FIFO depth = 8, so up to 8 data words can immediately get posted.

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 PCI_TRDY# will remain active.

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 PCI_TRDY# signal is deasserted while the application side catches up.

Figure 7. T8110 PCI Interface—Burst Write Cycle



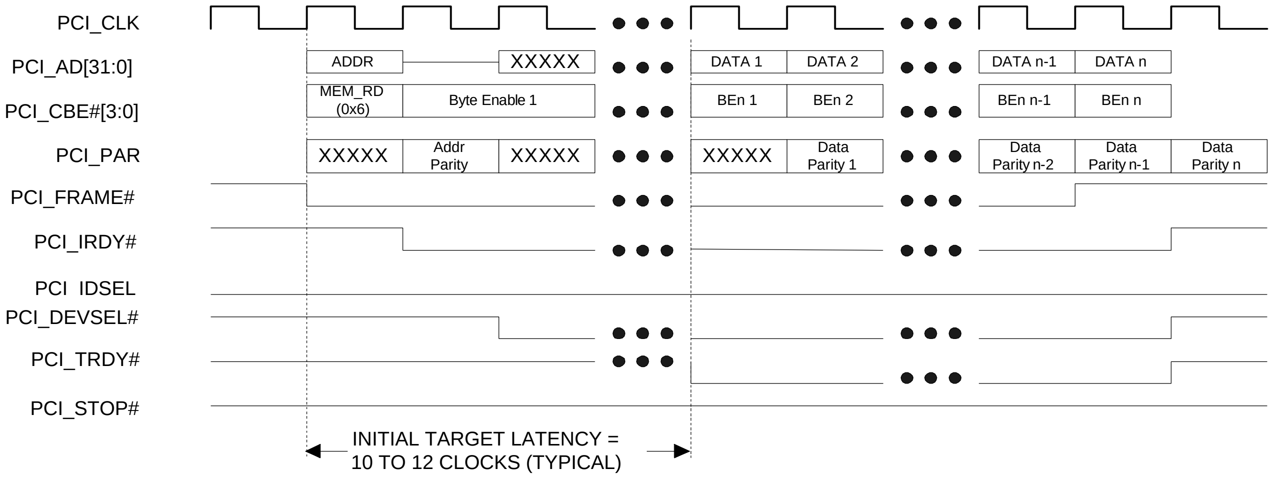
Notes:

T8110 PCI I/F has medium decode speed. There is always a two-cycle turnaround between assertion of PCI_FRAME# and assertion of PCI_DEVSEL#.

Turnaround 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

4 PCI Interface (continued)



Notes:
T8110 PCI I/F has medium decode speed. There is always a two-cycle turnaround between assertion of PCI_FRAME# and assertion of PCI_DEVSEL#.

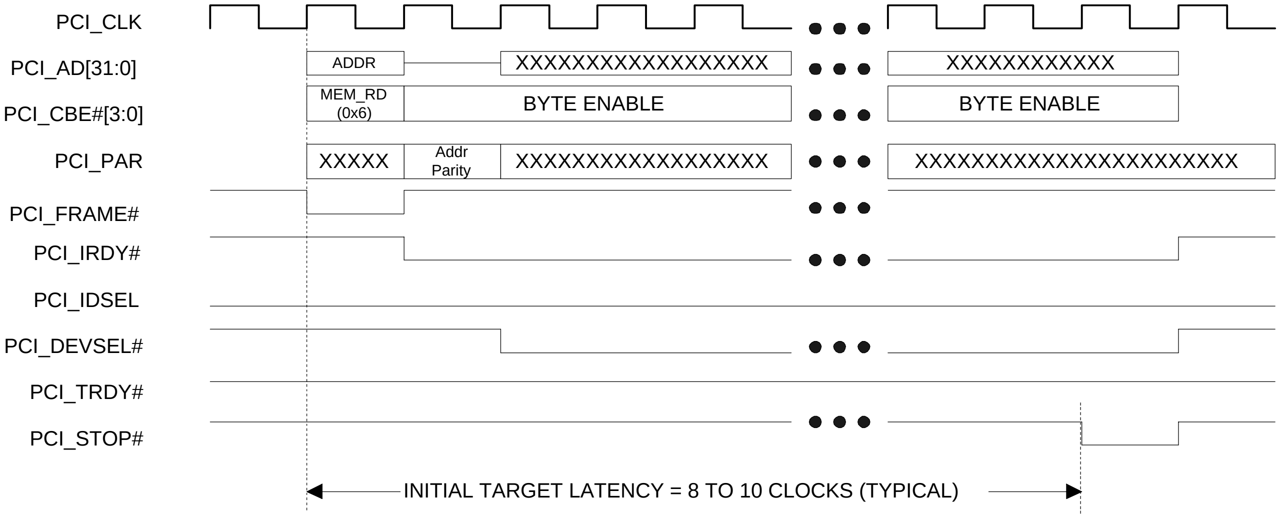
Turnaround time for memory reads from the T8110 is variable, depending on the region being accessed, and the synchronization time across the PCI clock and application clock domains. Initial target latency is typically between 10—12 PCI clock cycles.

PCI core read FIFO depth = 8.

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.

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 PCI_TRDY# signal is deasserted while the application side catches up.

Figure 9. T8110 PCI Interface—Burst Read Cycle

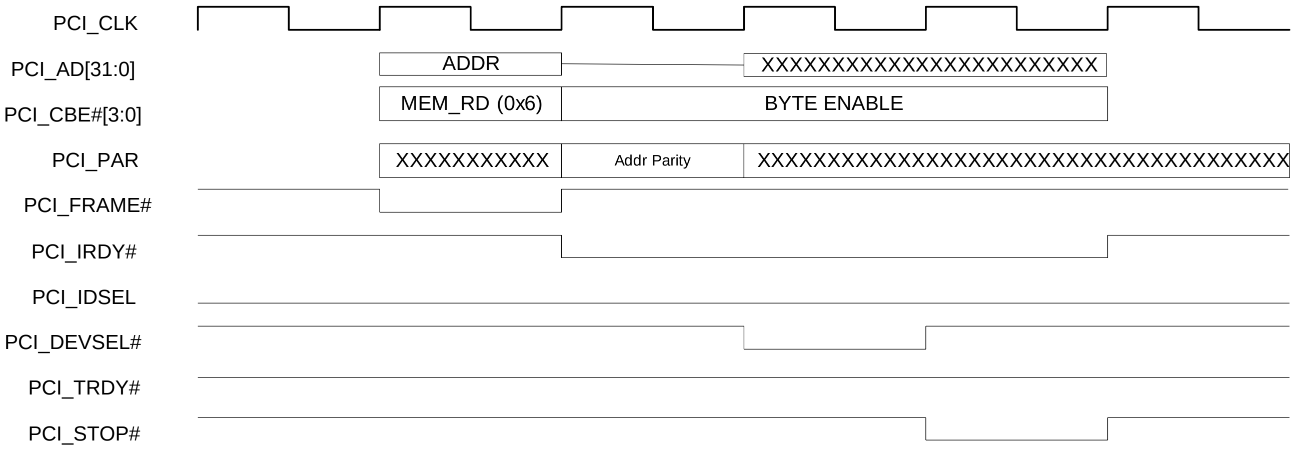


Notes:
T8110 PCI I/F has medium decode speed. There is always a two-cycle turnaround between assertion of PCI_FRAME# and assertion of PCI_DEVSEL#.

Turnaround 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)

4 PCI Interface (continued)



Notes:
T8110 PCI I/F has medium decode speed. There is always a two-cycle turnaround between assertion of PCI_FRAME# and assertion of PCI_DEVSEL#.

Turnaround time for memory read target ABORT is 4 PCI clocks.

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

4 PCI Interface (continued)

4.1.2 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 6, starting on page 46, through Section 13, and to Figure 6 on page 25, through Figure 9. A detected address parity error on any read transaction results in a target abort; refer to Figure 11 on page 28. 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. PCI_TRDY# remains asserted for all valid data phases applied.

4.1.3 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 14.1 on page 136, 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, PCI_TRDY# gets deasserted until the application side catches up.

4.1.4 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 7. 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 28). Address parity errors on write transactions are still posted to the PCI core interface, but are discarded.

4.1.4.1 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 6.2.7) 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 6.2.5 on page 59). Subsequent read attempts from the data memory while a posted write is queued result in a target RETRY; please refer to Figure 10.

4 PCI Interface (continued)

4.1.4.2 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; please refer to 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 6.2.5 on page 59).

4.1.5 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 14.1.1.2 on page 138). 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.

4.1.5.1 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 6.2.7) 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 6.2.5 on page 59). Subsequent read attempts from the virtual channel memory while a posted WRITE is queued result in a target retry (refer to Figure 10).

4.1.5.2 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 6.2.5 on page 59).

4.1.6 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 the Minibridge section, starting on page 107, 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 PCI Interface (continued)

4.1.6.1 Posted Write Transaction

Only one posted write to the minibridge port may be queued at a time; please 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 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 6.2.5 on page 59). Subsequent read attempts from the minibridge port, while a posted write is queued, result in a target retry (refer to Figure 10).

4.1.6.2 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 6.2.5 on page 59).

4.2 Initiator

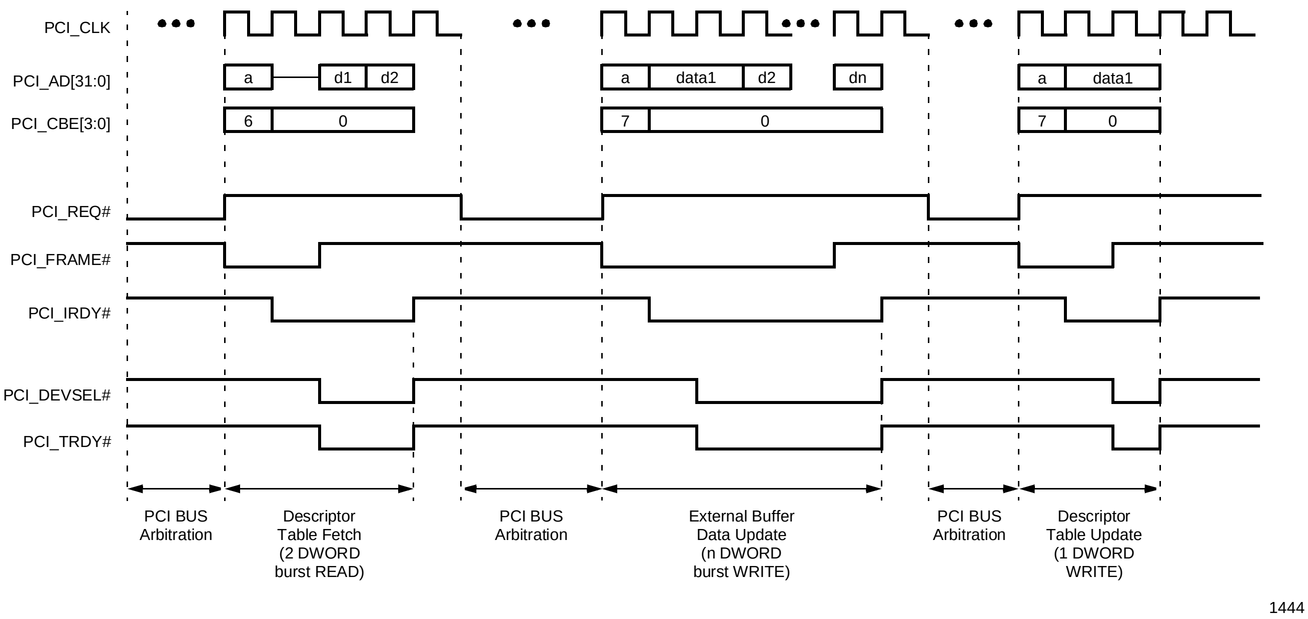
The T8110 can initiate PCI transactions in order to perform packet payload switching between the local PCI bus and the 64 H-bus/L-bus data streams. The T8110 initiates accesses in order to either send (or push) data received from H-bus/L-bus streams to an external data buffer, or to retrieve (or pull) 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 14.2.3 on page 155.

4.2.1 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 14.2.3 on page 155.

- 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).

4 PCI Interface (continued)



Notes:

This diagram depicts a target with medium decode speed (two-cycle turnaround to assertion of PCI_DEVSEL#).

Each of the three separate PCI transactions requires a PCI bus arbitration (PCI_REQ# active, system responds with PCI_GNT#).

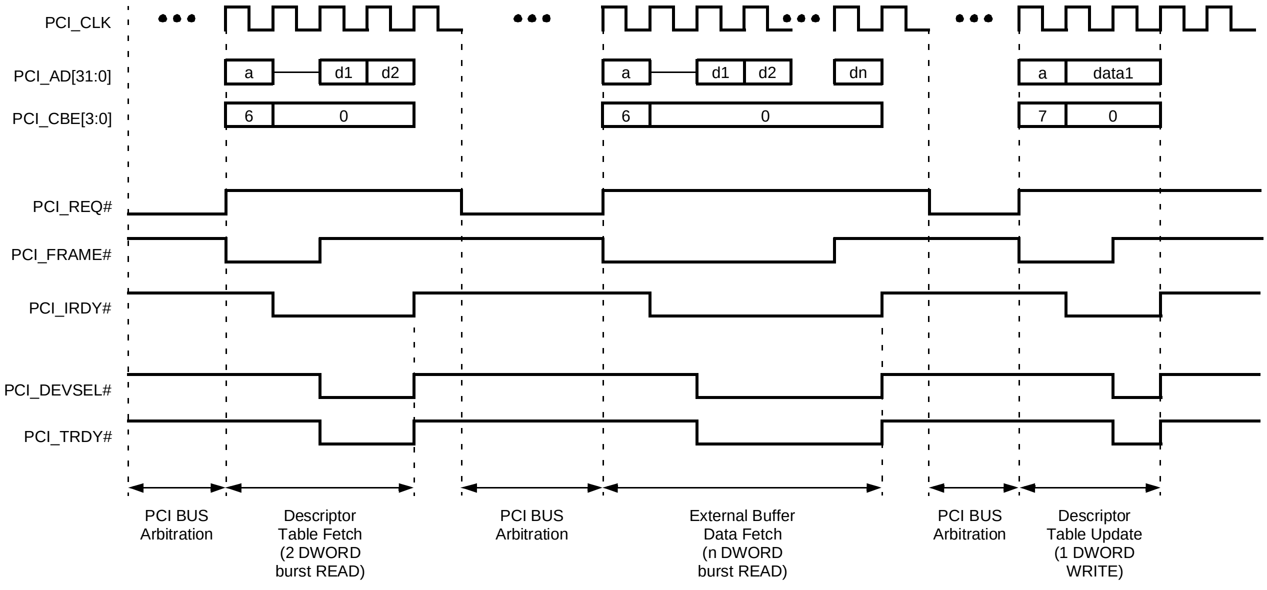
Figure 12. T8110 PCI Interface, Initiated PUSH Operation

4.2.2 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 14.2.3 on page 155.

- 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).

4 PCI Interface (continued)



Notes:
This diagram depicts a target with medium decode speed (two-cycle turnaround to assertion of PCI_DEVSEL#).

Each of the three separate PCI transactions requires a PCI bus arbitration (PCI_REQ# active, system responds with PCI_GNT#).

Figure 13. T8110 PCI Interface, Initiated PULL Operation

4 PCI Interface (continued)

4.3 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 12. T8110 PCI Configuration Registers

<table>
<tr><th>Byte Address</th><th colspan="4">Description</th></tr>
<tr><td>0x00</td><td colspan="2">Device ID</td><td colspan="2">Vendor ID</td></tr>
<tr><td>0x04</td><td colspan="2">Status register</td><td colspan="2">Command register</td></tr>
<tr><td>0x08</td><td colspan="3">Class code</td><td>Revision ID</td></tr>
<tr><td>0x0C</td><td>BIST</td><td>Header type</td><td>Latency timer</td><td>Cacheline size</td></tr>
<tr><td>0x10</td><td colspan="4">Memory base address</td></tr>
<tr><td>0x14</td><td colspan="4">Reserved</td></tr>
<tr><td>0x18</td><td colspan="4">Reserved</td></tr>
<tr><td>0x1C</td><td colspan="4">Reserved</td></tr>
<tr><td>0x20</td><td colspan="4">Reserved</td></tr>
<tr><td>0x24</td><td colspan="4">Reserved</td></tr>
<tr><td>0x28</td><td colspan="4">Reserved</td></tr>
<tr><td>0x2C</td><td colspan="2">Subsystem ID</td><td colspan="2">Subsystem vendor ID</td></tr>
<tr><td>0x30</td><td colspan="4">Reserved</td></tr>
<tr><td>0x34</td><td colspan="4">Reserved</td></tr>
<tr><td>0x38</td><td colspan="4">Reserved</td></tr>
<tr><td>0x3C</td><td>MAX_LAT</td><td>MIN_GNT</td><td>Interrupt pin</td><td>Interrupt line</td></tr>
<tr><td>0x40</td><td colspan="2">Reserved</td><td>Retry timeout</td><td>PCI_TRDY# timeout</td></tr>
<tr><td>0x44—0xFF</td><td colspan="4">Reserved</td></tr>
</table>

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

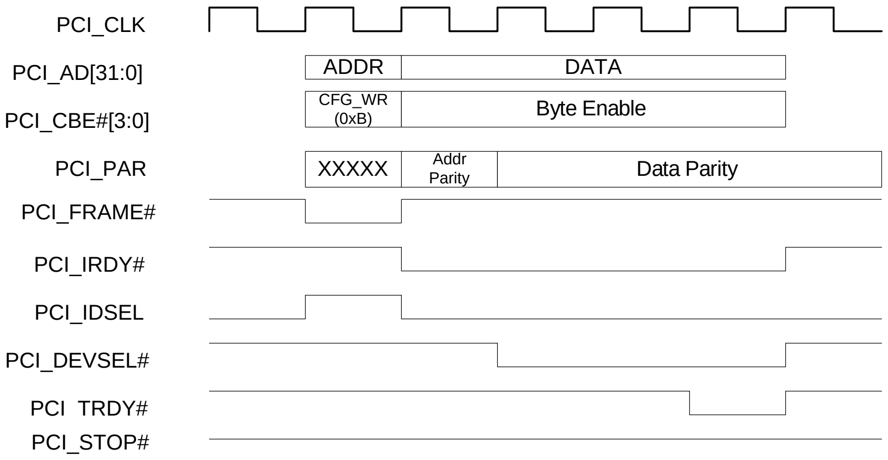
Device ID = 0x8110, T8110
Vendor ID = 0x11C1, Agere
Status register:
[15]: Detected parity error
[14]: Signaled system error
[13]: Received master abort
[12]: Received target abort
[11]: Signaled target abort
[10:9] = 01, T8110 DEVSEL# timing is medium
[8]: Data parity reported
[7] = 1, T8110 is fast 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 PCI 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]: I/O access enable
Class code = 0x02800, network controller—other
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 4.3.1 on page 36).

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
PCI_TRDY# timeout = value [default = 0x80]—T8110 as a master, how long it will wait for PCI_TRDY#

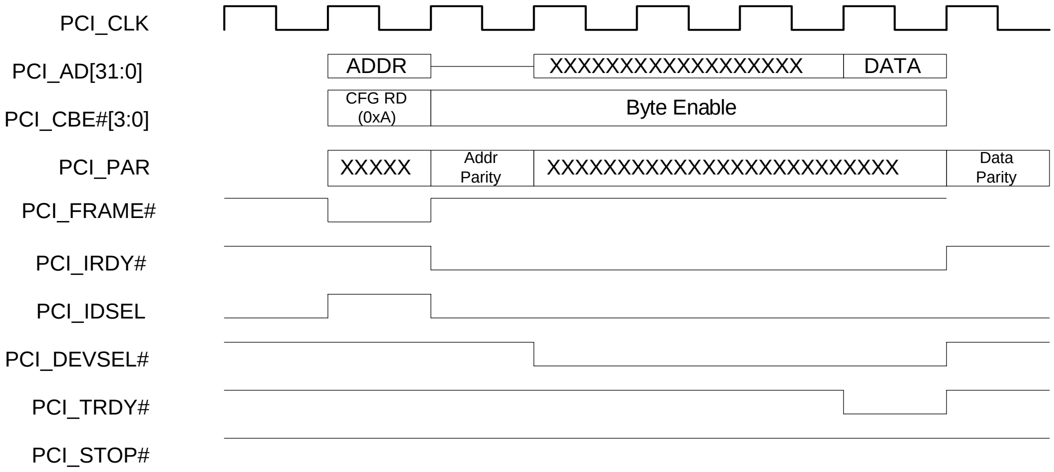


Notes:
T8110 PCI I/F has medium decode speed. There is always a two-cycle turnaround between assertion of PCI_FRAME# and assertion of PCI_DEVSEL#.

A configuration write access cycle takes five PCI clocks.

Figure 14. T8110 PCI Interface—Configuration WRITE Cycle

4 PCI Interface (continued)



Notes:
T8110 PCI I/F has medium decode speed. There is always a two-cycle turnaround between assertion of PCI_FRAME# and assertion of PCI_DEVSEL#.

A configuration read access cycle takes six PCI clocks.

Figure 15. T8110 PCI Interface—Configuration READ Cycle

4.3.1 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, subsystem vendor ID, MAX_LAT, MIN_GNT, and INTERRUPT_PIN (bytes 44—63). Ignored values (bytes 0—43) 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.

Note: If no EEPROM is present, internal pull-down resistors will set the values for subsystem ID, subsystem vendor ID, MAX_LAT, MIN_GNT, and INTERRUPT_PIN to zero. After the PCI core loads the values into configuration registers, this space is read-only. The only way to change the values from 0 is from an external EEPROM.

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)

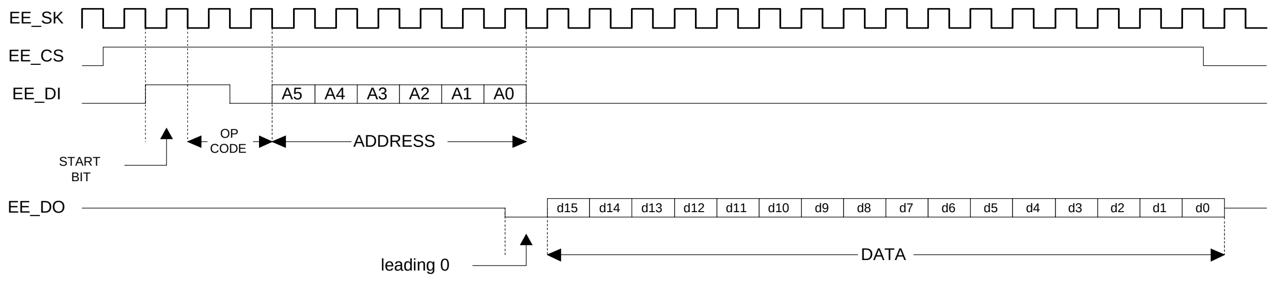
The interface protocol follows the standard 93C46 EEPROM (refer to Figure 16). A state machine within the T8110's PCI core produces nine read cycles, one for each of the read-only configuration register fields. Only the subsystem ID, subsystem vendor ID, MAX_LAT, MIN_GNT, and interrupt pin fields are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

4 PCI Interface (continued)

Table 13. 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

Note: The EEPROM is not write-accessible via the T8110. The T8110's PCI core only generates control signals to read the EEPROM.



Notes:

Signals output from T8110 are driven relative to the falling edge of EE_SK and are sampled by the EEPROM on the rising edge.

Signals output from the EEPROM are driven relative to the rising edge of EE_SK and are sampled by the T8110 on the falling edge.

Each read cycle takes 26 EE_SK clocks.

There are nine read cycles in all generated by the T8110's PCI core.

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

5 Microprocessor Interface

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

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

The T8110 microprocessor bus interface allows access to the T8110 internal regions via the minibridge port; see Table 9 on page 11 for pin descriptions. There are two user-selectable input signals that set up the microprocessor interface, MB_CS7 (*Intel/Motorola* protocol select) and MB_CS5 (word/byte address select).

5.1 *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 14. *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

5.2 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.

5 Microprocessor Interface (continued)

5.3 Access Via the Microprocessor Bus

The T8110 microprocessor 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 microprocessor 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 15.

Table 15. T8110 Memory Mapping to Microprocessor 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

5 Microprocessor Interface (continued)

5.3.1 Microprocessor Interface Register Map

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

Table 16. Microprocessor Interface Register Map

DWORD Address (20 bits)	Cross Reference	Register			
		Byte 3	Byte 2	Byte 1	Byte 0
0x00100	6.1.1, 6.1.2	Master enable	Reserved	Reset select	Soft reset
0x00104	6.1.3, 6.1.4	Phase alignment select	Clock register access select	Data memory mode select	Reserved
0x00108	6.1.4	Fallback trigger, upper	Fallback trigger, lower	Fallback type select	Fallback control
0x0010C	6.1.4	Watchdog EN, upper	Watchdog EN, lower	Watchdog select, NETREF	Watchdog select, C8
0x00114	4.1.5	Reserved	Failsafe sensitivity	Failsafe enable	Failsafe control
0x00118	6.1.11	Reserved	OOL monitor	OOL threshold high	OOL threshold low
0x00120	6.2.1	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	6.2.2, 6.2.5	Status 7, system errors, upper	Status 6, system errors, lower	Status 5	Status 4
0x00128	6.2.6	Device ID, upper	Device ID, lower	Reserved	Version ID
0x00140	13.1	Diag3	Diag2	Diag1	Diag0
0x00144	13.1	Diag7	Diag6	Diag5	Diag4
0x00148	13.1	Diag11	Diag10	Diag9	Diag8
0x00200	7.1	APLL1 rate	APLL1 input selector	Main divider	Main input selector
0x00204	7.1	APLL2 rate	Reserved	Resource divider	Main inversion select
0x00208	7.1	DPLL1 rate	DPLL1 input selector	Reserved	LREF input select
0x0020C	7.1	DPLL2 rate	DPLL2 input selector	Reserved	LREF inversion select
0x00210	7.1	Reserved	NETREF1 LREF select	NETREF1 divider	NETREF1 input selector
0x00214	7.1	Reserved	NETREF2 LREF select	NETREF2 divider	NETREF2 input selector
0x00220	7.2	C8 output rate	/FR_COMP width	NETREF output enables	Master output enables
0x00224	7.2	SCLK output rate	TCLK select	Reserved	CCLK output enables
0x00228	7.2	L_SC3 select	L_SC2 select	L_SC1 select	L_SC0 select
0x00300	10.1	H-bus rate H/G	H-bus rate F/E	H-bus rate D/C	H-bus rate B/A
0x00320	10.2	L-bus rate H/G	L-bus rate F/E	L-bus rate D/C	L-bus rate B/A
0x00400	8.1	FG0 rate	FG0 width	FG0 upper start	FG0 lower start

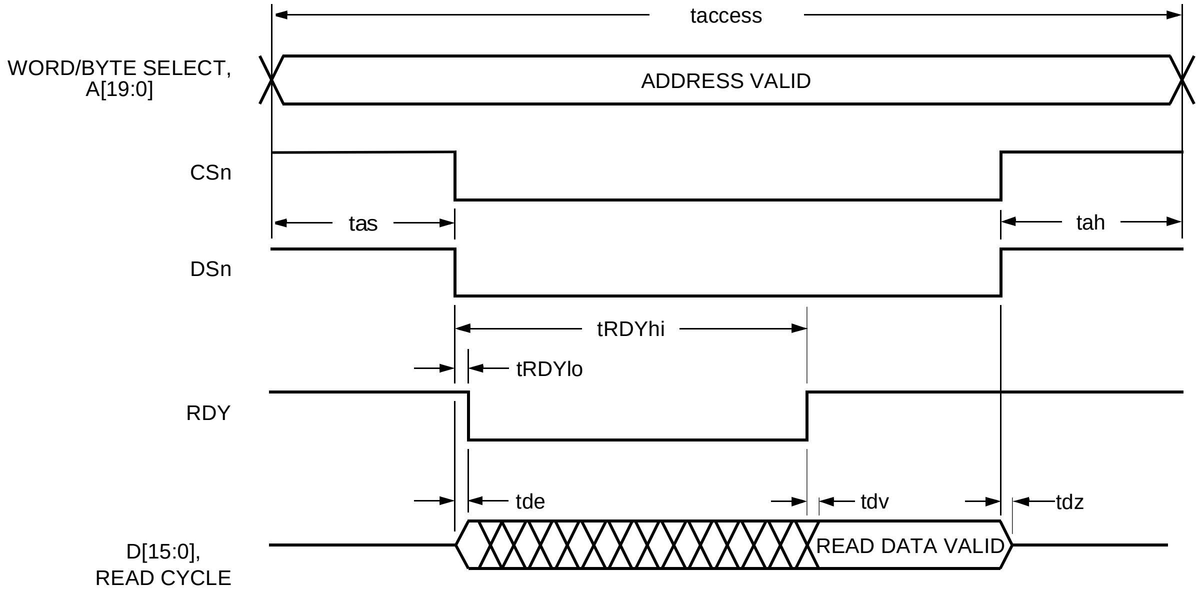
5 Microprocessor Interface (continued)

Table 16. Microprocessor Interface Register Map (continued)

DWORD Address (20 bits)	Cross Reference	Register			
		Byte 3	**Byte 2**	**Byte 1**	**Byte 0**
0x00410	8.1	FG1 rate	FG1 width	FG1 upper start	FG1 lower start
0x00420	8.1	FG2 rate	FG2 width	FG2 upper start	FG2 lower start
0x00430	8.1	FG3 rate	FG3 width	FG3 upper start	FG3 lower start
0x00440	8.1	FG4 rate	FG4 width	FG4 upper start	FG4 lower start
0x00450	8.1	FG5 rate	FG5 width	FG5 upper start	FG5 lower start
0x00460	8.1	FG6 rate	FG6 width	FG6 upper start	FG6 lower start
0x00470	8.1	FG7 rate	FG7 width	FG7 upper start	FG7 lower start
0x00474	8.2	FG7 mode upper	FG7 mode lower	FG7 counter high byte	FG7 counter low byte
0x00480	8.3	Reserved	FGIO R/W	FGIO read mask	FGIO data register
0x00500	9.1	GPIO override	GPIO R/W	GPIO read mask	GPIO data register
0x00600	14.1	FGIO interrupt polarity	Reserved	FGIO interrupt enable	FGIO interrupt pending
0x00604	12.1	GPIO interrupt polarity	Reserved	GPIO interrupt enable	GPIO interrupt pending
0x00608	12.1	System interrupt enable, upper	System interrupt enable, lower	System interrupt pending, upper	System interrupt pending, lower
0x0060C	12.1	Clock interrupt enable, upper	Clock interrupt enable, lower	Clock interrupt pending, upper	Clock interrupt pending, lower
0x00610	12.1	CLKERR output select	SYSERR output select	PCI_INTA output select	Arbitration control
0x00614	12.1	CLKERR pulse width	SYSERR pulse width	Reserved	Reserved
0x006FC	12.1	In-service, byte 3	In-service, byte 2	In-service, byte 1	In-service, byte 0

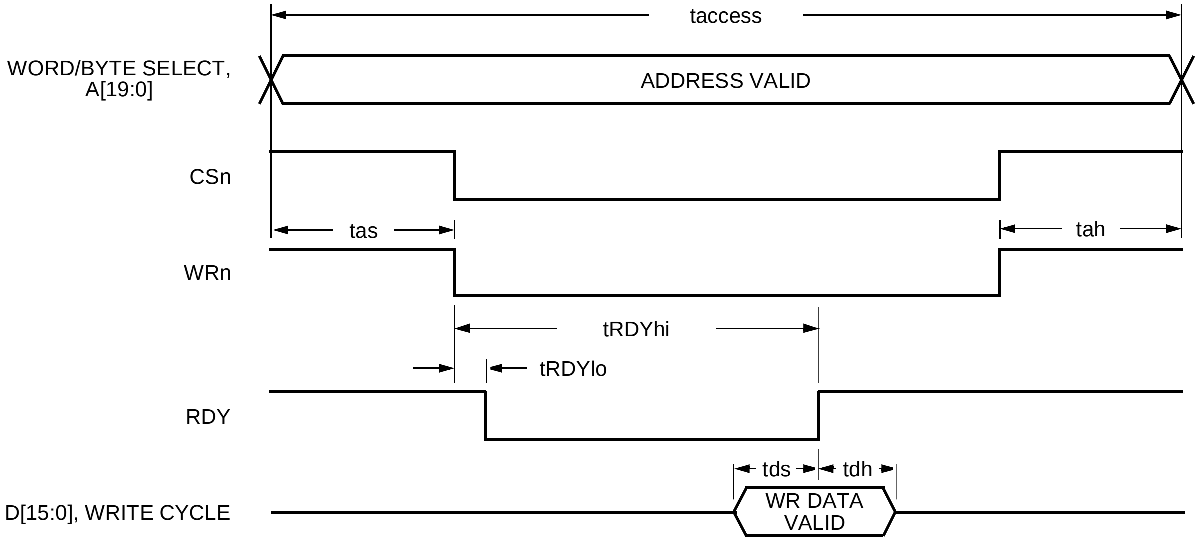
5 Microprocessor Interface (continued)

Microprocessor Access Read Cycle, *Intel* Protocol



5-9418 (F)

Microprocessor Access Write Cycle, *Intel* Protocol

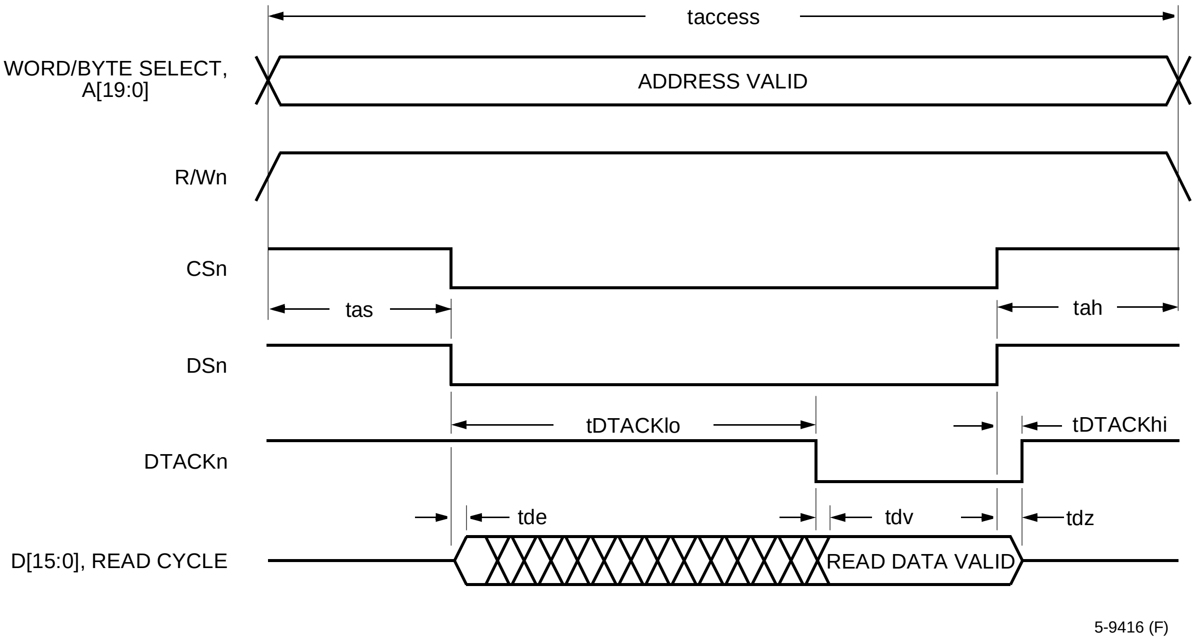


5-9419 (F)

Figure 17. Microprocessor Access Timing, *Intel* Protocol

5 Microprocessor Interface (continued)

Microprocessor Access Read Cycle, *Motorola* Protocol



Microprocessor Access Write Cycle, *Motorola* Protocol

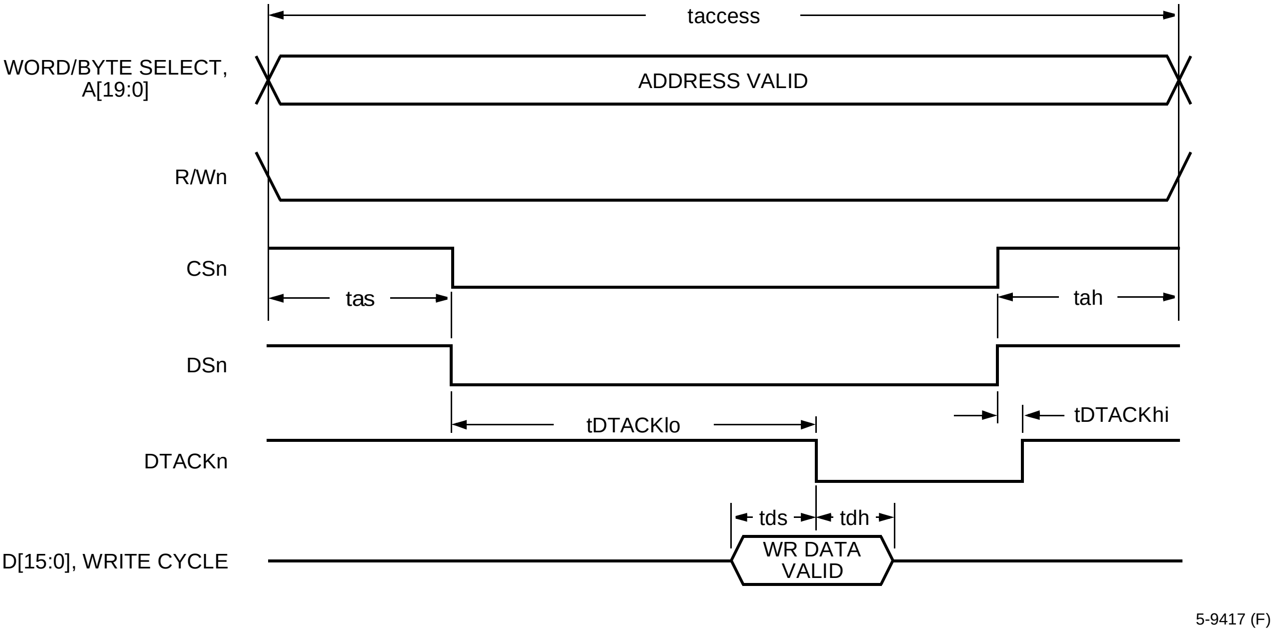


Figure 18. Microprocessor Access Timing, *Motorola* Protocol

5 Microprocessor Interface (continued)

5.3.2 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 and Figure 18 is shown below.

Table 17. Register Space Access Timing

Name	Parameter	Min (ns)	Max (ns)
taccess	Overall Access Time	41	—
tas	Address Setup Time	5	—
tah	Address Hold Time	0	—
tRDYlo	*Intel* Cycle, Time to RDY Deasserted	6	12
tRDYhi	*Intel* Cycle, Time to RDY Reasserted	36	72
tDTACKlo	*Motorola* Cycle, Time to DTACKn Asserted	36	70
tDTACKhi	*Motorola* Cycle, Time to DTACKn Deasserted	10	15
tde	Read Cycle, Time to Data Enabled	7	14
tdv	Read Cycle, Time to Data Valid	5	9
tdz	Read Cycle, Time to Data Invalid	10	16
tds	Write Cycle, Data Setup Time	25	—
tdh	Write Cycle, Data Hold Time	0	—

5.3.3 Connection Memory Space Access

The T8110 connection memory is always immediately available for access (via dedicated access times assigned for microprocessor transactions) providing low latency time to acknowledge the transaction. Connection memory access timing for Figure 17 and Figure 18 is shown below.

Table 18. Connection Memory Space Access Timing

Name	Parameter	Min (ns)	Max (ns)
taccess	Overall Access Time	41	—
tas	Address Setup Time	5	—
tah	Address Hold Time	0	—
tRDYlo	*Intel* Cycle, Time to RDY Deasserted	6	12
tRDYhi	*Intel* Cycle, Time to RDY Reasserted	36	72
tDTACKlo	*Motorola* Cycle, Time to DTACKn Asserted	36	70
tDTACKhi	*Motorola* Cycle, Time to DTACKn Deasserted	10	15
tde	Read Cycle, Time to Data Enabled	7	14
tdv	Read Cycle, Time to Data Valid	5	9
tdz	Read Cycle, Time to Data Invalid	10	16
tds	Write Cycle, Data Setup Time	25	—
tdh	Write Cycle, Data Hold Time	0	—

5 Microprocessor Interface (continued)

5.3.4 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 microprocessor 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. Data memory access timing for Figure 17 and Figure 18 is shown below.

Table 19. Data Memory Space Access Timing

Name	Parameter	Min (ns)	Max (ns)
taccess	Overall Access Time	41	*
tas	Address Setup Time	5	—
tah	Address Hold Time	0	—
tRDYlo	*Intel* Cycle, Time to RDY Deasserted	6	12
tRDYhi	*Intel* Cycle, Time to RDY Reasserted	36	*
tDTACKlo	*Motorola* Cycle, Time to DTACKn Asserted	36	*
tDTACKhi	*Motorola* Cycle, Time to DTACKn Deasserted	10	15
tde	Read Cycle, Time to Data Enabled	7	14
tdv	Read Cycle, Time to Data Valid	5	9
tdz	Read Cycle, Time to Data Invalid	10	16
tds	Write Cycle, Data Setup Time	25	—
tdh	Write Cycle, Data Hold Time	0	—

* Max data memory space access time is indeterminate, and depends on how much of the data memory access bandwidth is being taken by TDM switch connections.

5.3.5 Virtual Channel Memory Space Access

Microprocessor access to the virtual channel memory is provided for diagnostic purposes only and is disabled by default. Access to this region is enabled via the diagnostic registers; see Section 13 on page 128. The T8110 virtual channel memory is not guaranteed to be immediately available for access. Access to virtual channel memory is prioritized for standard H-bus/L-bus switching, with microprocessor 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 and Figure 18 is shown below.

Table 20. Virtual Channel Memory Space Access Timing

Name	Parameter	Min (ns)	Max (ns)
taccess	Overall Access Time	41	—
tas	Address Setup Time	5	—
tah	Address Hold Time	0	—
tRDYlo	*Intel* Cycle, Time to RDY Deasserted	6	12
tRDYhi	*Intel* Cycle, Time to RDY Reasserted	36	57*
tDTACKlo	*Motorola* Cycle, Time to DTACKn Asserted	36	55*
tDTACKhi	*Motorola* Cycle, Time to DTACKn Deasserted	10	15
tde	Read Cycle, Time to Data Enabled	7	14
tdv	Read Cycle, Time to Data Valid	5	9
tdz	Read Cycle, Time to Data Invalid	10	16
tds	Write Cycle, Data Setup Time	25	—
tdh	Write Cycle, Data Hold Time	0	—

* Immediate response, same as register access, assuming no virtual channel connections are programmed into the connection memory (virtual channels aren't supported with the microprocessor interface protocol selected).

6 Operating Control and Status

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

6.1 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 21. 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	Reserved	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 microprocessor bus, this register is [reserved].

6.1.1 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, and initialize the T8110 PCI interface. The power-on reset cell test input is controlled via diagnostic register; see Section 13.
- **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 9.1 on page 98). Polarity of the forwarded reset is selectable (via register 0x00781; see Section 11.2 on page 110).

Soft resets are maskable via reset select register, SRBEB, and 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.

6 Operating Control and Status (continued)

- 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.

Table 22. 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	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).

6.1.2 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, and NR2_SEL_OUT.

6 Operating Control and Status (continued)

Table 23. 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. **Note:** If AIOEB is set to 1 to enable all then MBREB must also be set to 1 to enable the 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	Disable L-bus clocks, L_SC, FG (default). Enable 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 microprocessor bus, this bit is reserved.

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

The VC start 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 14.2.3 for more detail). Writes to the VC start register trigger the corresponding action, and the set bit(s) are automatically cleared.

Table 24. Virtual Channel Enable and Data Memory Selector Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00104	VCSTART	7:0	VCSSR	0000 0000 0001 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 14.2.1.2 on page 148.

There are six data memory configurations as outlined below. (If the T8110 interfaces to the PCI bus, all configurations are valid. If the interface selection is to the microprocessor bus, only the standard switching configurations 1—3 are allowed.)

6 Operating Control and Status (continued)

1. 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.
2. 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.
3. 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).
4. 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.
5. 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.
6. No standard switching + 512 virtual channels. Packet payload switching only, up to 512 virtual channels.

Table 25. Data Memory Mode Select Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00105	Data Memory Mode Select	7	GSREB	0 1	Disable subrate switching (default). Enable subrate switching.
		6:0	DMMSP	100 0000 010 0000 011 0000 010 0010 001 0010 000 0100	4k single-buffer switch (default). 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.

6.1.4 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. The X and Y register sets are comprised of the registers listed in Table 43 on page 63, Clock Input Control Register Map, and Table 56 on page 72, Clock Output Control Register Map. 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 7.3 on page 76 for more details.

Table 26. Clock Register Access Select Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00106	Clock Register Access Select	7:0	CSASR	0000 0000 0000 0001	Access inactive clock registers (default). Access active clock registers.

6 Operating Control and Status (continued)

6.1.5 Phase Alignment Select Register

The phase alignment select register selects the phase alignment configuration. For more details, see Section 7.4.5.1 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_FRAME_A, /CT_FRAME_B, 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 27. 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.

6.1.6 Fallback Control Register

The fallback control register allows user control over the active and inactive clock register sets. For more details, see Section 7.7.1 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.

6 Operating Control and Status (continued)

Table 28. Fallback Control Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00108	Fallback Control	7:0	FBCSR	0000 0000	NOP (default).
				0000 0001	GO_CLOCKS command.
				0000 0010	CLEAR_FALLBACK command.
				0000 0100	FORCE_FALLBACK command.
				0001 0001	GO_CLOCKS synchronized to frame*.
				0001 0010	CLEAR_FALLBACK synchronized to frame*.
				0001 0100	FORCE_FALLBACK synchronized to frame*.
				0010 0000	COPY ACTIVE TO INACTIVE SET command.

* The synchronized to frame command also has a diagnostic element—instead of performing the command right at the frame boundary, the user can elect to perform the command at a specified offset time from the frame boundary, by programming the Diag11 and Diag10 registers, 0x0014B—0x0014A.

6.1.7 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 7.7.1 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 7.7 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 29. Fallback Type Select Register

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

6.1.8 Fallback Trigger Registers

The fallback trigger registers are used in conjunction with the fallback type select register and control which H-bus clocks are enabled to trigger a clock fallback event in case of error. The sync reference inputs to DPLL1 and DPLL2 can also trigger a clock fallback event upon detection of an error.

6 Operating Control and Status (continued)

Table 30. 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.

6.1.9 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).

6 Operating Control and Status (continued)

Table 31. 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.

6.1.10 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 DPLL1 and DPLL2.

Table 32. 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.

6 Operating Control and Status (continued)

Table 32. 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.

6.1.11 Failsafe Control Registers

Table 33. Failsafe Control Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00114	Failsafe Control	7:0	FSCSR	0000 0000 0000 0001 0000 0010	NOP (default). Return from failsafe to nonfallback condition. Return from failsafe to fallback condition.
0x00115	Failsafe Enable	7:0	FSEER	0000 0000 0000 0001	Failsafe disabled. Failsafe enabled.
0x00116	Failsafe Sensitivity	7:0	FSSSR	0000 0000 0000 0001 0000 0010 0000 0100 0000 1000	Failsafe watchdog highest sensitivity. Failsafe watchdog + 30.5 ns. Failsafe watchdog + 121.0 ns. Failsafe watchdog + 244.0 ns. Failsafe watchdog + 488.0 ns.
0x00118	OOL Threshold Low	7:0	OLLLR	LLLL LLLL	Failsafe threshold value, low byte.
0x00119	OOL Threshold High	7:0	OLHLR	LLLL LLLL	Failsafe threshold value, high byte.
0x0011A	OOL Monitor	7:0	OOLER	0000 0000 0000 0001	Monitor direct APLL1 lock detect at PLOCK. Monitor user threshold lock detect at PLOCK.

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. For more on failsafe, please see Section 7.7.2 on page 88.

6 Operating Control and Status (continued)

The failsafe enable register controls the enable/disable of failsafe operation. For more on failsafe operation, please see Section 7.7.2 on page 88.

The failsafe sensitivity register allows the failsafe watchdog timer to be desensitized by either 1, 4, 8, or 16 watchdog sample clock periods.

The OOL threshold registers allow for programmable threshold times which indicate the APLL1 out-of-lock. Resolution for the threshold 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 1.99 ms threshold. For more on OOL operation, please see Section 7.7.2 on page 88.

The OOL monitor register allows the user to monitor either the raw APLL1 out-of-lock status, OR the status flag that indicates that the APLL1 has been out-of-lock for more than the threshold defined in the OOL threshold registers.

6.1.12 External Buffers—Descriptor Table Base Address

Table 34. Extended Buffers Base Addresses

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00110	Base Address Byte 0	7:0	NA	LLLL LLLL	32-bit base address value for external buffers descriptor table*.
0x00111	Base Address Byte 1	7:0	NA	LLLL LLLL	32-bit base address value for external buffers descriptor table*.
0x00112	Base Address Byte 2	7:0	NA	LLLL LLLL	32-bit base address value for external buffers descriptor table*.
0x00113	Base Address Byte 3	7:0	NA	LLLL LLLL	32-bit base address value for external buffers descriptor table*.

* The 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 microprocessor bus, this register is reserved. For more details, refer to Section 14.2.3.4 on page 160.

6.2 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 35. 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

6 Operating Control and Status (continued)

6.2.1 Clock Errors

6.2.1.1 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 DPLL1/DPLL2 sync inputs whose watchdogs are enabled. The transient indicators are dynamic in nature; 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 APLL1 out-of-lock indicator is provided, and used in conjunction with the failsafe clocking mode. For more details, please see Section 7.7.1 on page 82 and Section 7.7.2 on page 88.

Table 36. 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.
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.

6 Operating Control and Status (continued)

6.2.1.2 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 7.7 on page 82, and Section 12 on page 113 for more details.

Table 37. 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.

6 Operating Control and Status (continued)

6.2.2 System Status

6.2.3 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 7.7.1 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.

Table 38. Fallback and Failsafe Status Register

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.

6.2.4 PLL and Switching Status Register

The upper 6 bits provide APLL1, DPLL1, and DPLL2 lock status. For more details, see Section 7 on page 62. The lower 2 bits provide memory status as follows: CMROB is an indicator for the connection memory actively resetting; see Section 14.2.1.1 on page 146. DMPOB indicates the active data page used for double-buffered standard switching; see Section 14.2.1.2 on page 148.

6 Operating Control and Status (continued)

Table 39. PLL and Switching Status Register

Byte Address	Register Name	Bit(s)	Mnemonic	Value	Function
0x00125	Status 5, PLL and Switching Status	7	A1LOB	0 1	APLL1 in-lock. APLL1 out-of-lock.
		6	OOLOB	0 1	Out-of-lock indicator inactive. Out-of-lock indicator active.
		5:4	D1LOP	00 01 10 11	DPLL1 in-lock (default). DPLL1 out-of-lock, slow correction. DPLL1 out-of-lock, fast correction. Invalid.
		3:2	D2LOP	00 01 10 11	DPLL2 in-lock (default). DPLL2 out-of-lock, slow correction. DPLL2 out-of-lock, fast correction. Invalid.
		1	CMROB	0 1	Connection memory not resetting (default). Connection memory reset loop is active.
		0	DPGOB	0 1	Active page = data memory page 1. Active page = date memory page 2.

6.2.5 System Errors Register

Table 40. System Errors Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00126	Status 6, System Errors, Lower	7	PMFOB	0 1	No error. PCI master, PCI bus fatal error.
		6	PMLOB	0 1	No error. PCI master, external buffer LOCK error.
		5	PMEOB	0 1	No error. PCI master, external buffer STALL error.
		4	PMWOB	0 1	No warning. PCI master, external buffer STALL warning.
		3	PMOOB	0 1	No warning. PCI master, external buffer overwrite warning.
		2	PMIOB	0 1	No warning. PCI master, external buffer INITIAL warning.
		1	VCOOB	0 1	No warning. VC memory, scratchpad overflow warning.
		0	NQOOB	0 1	No warning. NOTIFY_QUEUE, overflow warning.

6 Operating Control and Status (continued)

Table 40. System Errors Registers (continued)

Byte Address	Register Name	Bit(s)	Mnemonic	Value	Function
0x00127	Status 7, System Errors, Upper	7	CFSOB	0 1	No error. Clock failsafe indicator.
		6	CFBOB	0 1	No error. Clock fallback indicator.
		5	MBTOB	0 1	No time-out. PCI target, minibridge discard timer expired.
		4	VCTOB	0 1	No time-out. PCI target, VC memory discard timer expired.
		3	DMTOB	0 1	No time-out. PCI target, data memory discard timer expired.
		2	MBPOB	0 1	No error. PCI target, minibridge protocol error.
		1	VCPOB	0 1	No error. PCI target, VC memory protocol error.
		0	DMPOB	0 1	No error. PCI target, data memory protocol error.

6.2.6 Device Identification Registers

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

Table 41. 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.

6 Operating Control and Status (continued)

6.2.7 Miscellaneous Status

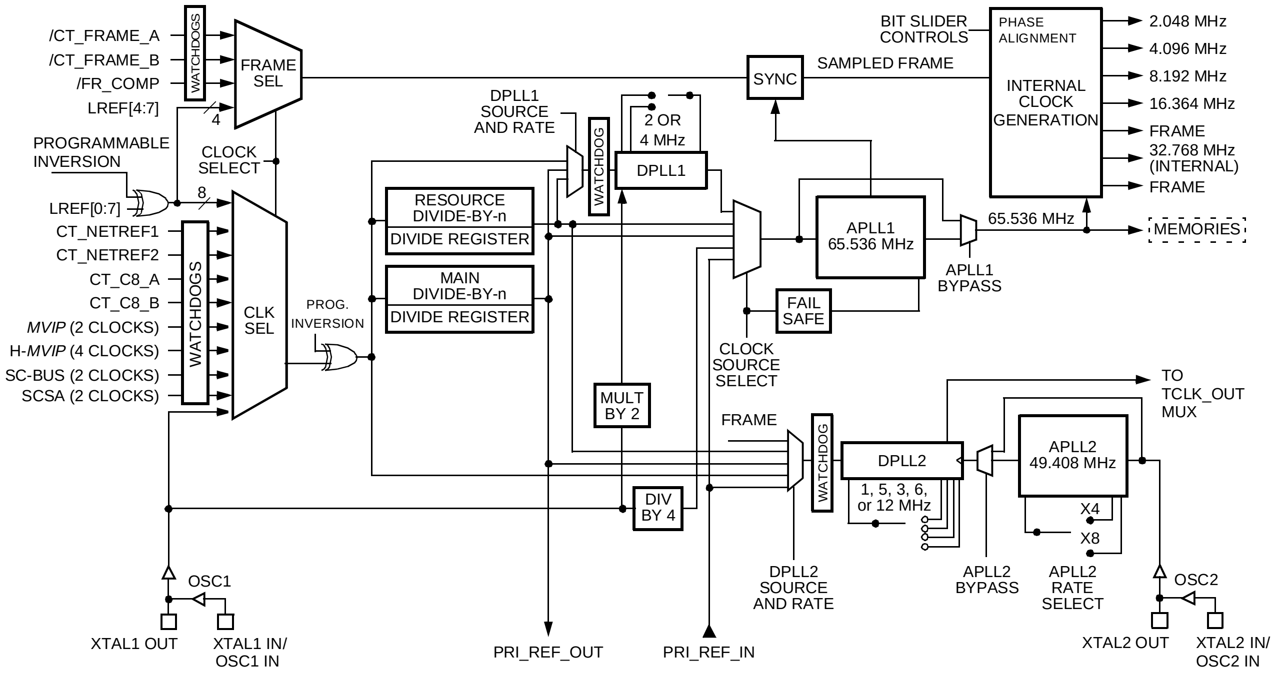
The status 8 register provides the status flags which indicate that there is currently a posted write or a delayed read enqueued, one flag for each access region minibridge, virtual channel memory, and data memory. Please refer to Sections 4.1.4, 4.1.4.1, and 4.1.5.1 for more details.

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. Please refer to Section 14.2.3. for more details.

Table 42. Miscellaneous Status Registers

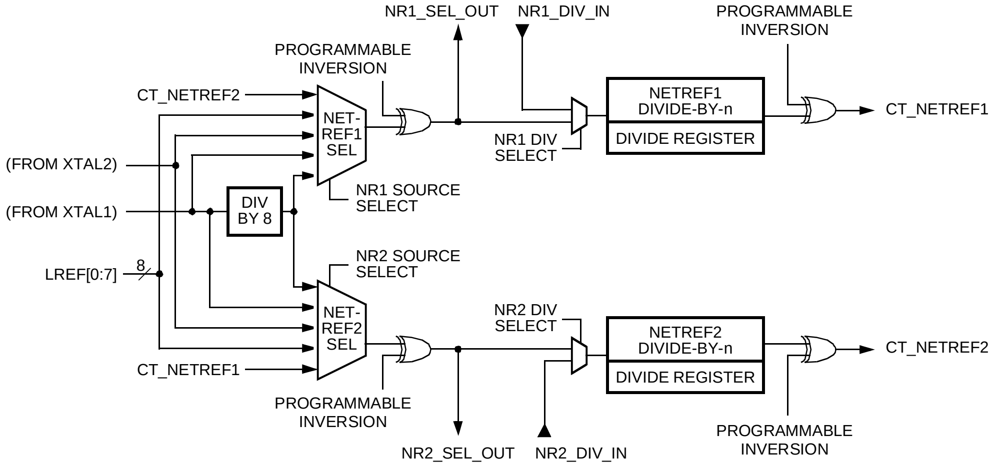
Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x0012C	Status 8	7:3	Reserved	0000 0	NOP.
		2	MBSOB	0 1	Minibridge PCI target queue is empty. Minibridge PCI target queue is full.
		1	VCSOB	0 1	VC memory PCI target queue is empty. VC memory PCI target queue is full.
		0	DMSOB	0 1	Data memory PCI target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

7 Clock Architecture



5-9432 (F)

Figure 19. T8110 Main Clocking Paths



5-9433 (F)

Figure 20. T8110 NETREF Paths

7 Clock Architecture (continued)

7.1 Clock Input Control Registers

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

Table 43. Clock Input Control Register Map

DWORD Address (20 bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00200	APLL1 rate	APLL1 input selector	Main divider	Main input selector
0x00204	APLL2 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

7.1.1 Main Input Selector Register

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

Table 44. 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 0000	Select *MVIP* clocks (C2 bit clock)*.
				1000 0001	Select *MVIP* clocks (/C4 bit clock).
				1000 0010	Select H-*MVIP* clocks (/C16± bit clock).
				1000 0100	Select SC-bus clocks 2 MHz.
				1000 1000	Select SC-bus clocks 4/8 MHz.

* C2 is allowed as the bit clock input.

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)

* Selection of which LREF is controlled at register 0x00208. Selection of LREF polarity is controlled at register 0x0020C.

7 Clock Architecture (continued)

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)
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.

7.1.2 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 45. 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.

7.1.3 Analog PLL1 (APLL1) Input Selector Register

The APLL1 input selector register controls APLL1 reference input selection. The choices include the following:

- APLL1 reference clock = oscillator/4 (4.096 MHz)
- APLL1 reference clock = output of the main divider (4.096 MHz or 2.048 MHz)
- APLL1 reference clock = output of the resource divider (4.096 MHz or 2.048 MHz)
- APLL1 reference clock = output of DPLL1 (4.096 MHz or 2.048 MHz)
- APLL1 reference clock = input from signal PRI_REF_IN (4.096 MHz or 2.048 MHz)

Table 46. APLL1 Input Selector Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00202	APLL1 Input Selector	7:0	P1ISR	0000 0000	Select oscillator/4 (default).
				0000 0001	Select main divider output.
				0000 0010	Select resource divider output.
				0000 0100	Select DPLL1 output.
				0000 1000	Select external input PRI_REF_IN.

7 Clock Architecture (continued)

7.1.4 APLL1 Rate Register

The APLL1 rate register provides the rate multiplier value to APLL1. When APLL1 reference clock is at 4.096 MHz, the [x16 (multiplied by)] value must be selected. When APLL1 reference clock is at 2.048 MHz, the [x32 (multiplied by)] value must be selected. A [x1 (multiplied by)] value is provided in order to bypass APLL1.

Table 47. APLL1 Rate Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00203	APLL1 Rate	7:0	P1RSR	0000 0000 0000 0001 0001 xxxx	Times 16 (default). Times 32. Times 1 BYPASS (lower nibble is don't care).

7.1.5 Main Inversion Select Register

The main inversion select register controls programmable 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; see Figure 19 on page 62.
- NETREF2 divider output; see Figure 20 on page 62.
- NETREF2 selection MUX output
- NETREF1 divider output
- NETREF1 selection MUX output

Table 48. 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.

7 Clock Architecture (continued)

7.1.6 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 49. 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.

7.1.7 Analog PLL2 (APLL2) Rate Register

The APLL2 rate register provides the rate multiplier value to APLL2. When the APLL2 reference clock is at 12.352 MHz, the (times 4) value must be selected. When the APLL2 reference clock is at 6.176 MHz, the (times 8) value must be selected. A (times 1) value is provided in order to bypass APLL2.

Table 50. APLL2 Rate Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00207	APLL2 Rate	7:0	P2RSR	0000 0000	Times 4 (default).
				0000 0001	Times 8.
				0001 xxxx	Times 1 BYPASS (lower nibble is don't care).

7 Clock Architecture (continued)

7.1.8 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 19 on page 62 for further details.

Table 51. 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	Select LREF0 (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.
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.

7 Clock Architecture (continued)

7.1.9 DPLL1 Input Selector

The DPLL1 input selector selects one of three sources for DPLL1 synchronization input (see Section 7.4.2 on page 78), including the following:

- Main clock selection CLK SEL MUX output
- Main divider output
- Resource divider output

7.1.9.1 DPLL1 Rate Register

The DPLL1 rate register controls the DPLL1 output frequency.

Table 52. 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.

7 Clock Architecture (continued)

7.1.10 DPLL2 Input Selector

The DPLL2 input selector selects one of five sources for DPLL2 synchronization input (see Section 7.5.1 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

7.1.10.1 DPLL2 Rate Register

The DPLL2 rate register controls the DPLL2 output frequency.

Table 53. 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.

7 Clock Architecture (continued)

7.1.11 NETREF1 Registers

The NETREF1 input selector, NETREF1 divider, and NETREF1 LREF select registers control the signal paths used to generate CT_NETREF1 (see Figure 20 on page 62).

Table 54. 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/XTAL1-div-8, 2.048 MHz (default). Oscillator/XTAL1, 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.
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	Select LREF0 (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.

7 Clock Architecture (continued)

7.1.12 NETREF2 Registers

The NETREF2 input selector, NETREF2 divider, and NETREF2 LREF select registers control the signal paths used to generate CT_NETREF2 (see Figure 20 on page 62).

Table 55. 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/XTAL1-div-8, 2.048 MHz (default). Oscillator/XTAL1, 16.384 MHz. CT_NETREF1 input. LREF input*. Oscillator/XTAL2, 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	Select LREF0 (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.

7 Clock Architecture (continued)

7.2 Clock Output Control Registers

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

Table 56. 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

7.2.1 Master Output Enables Register

The master output enables register controls the output enables for H-bus and compatibility clocks (CCLK) 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.

These programmable enables are used in conjunction with master enable register 0x00103, 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 0x00103, 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

7 Clock Architecture (continued)

Table 57. 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 (default). Enable C clocks individually*. Enable all C clocks.
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.

7 Clock Architecture (continued)

7.2.2 Clock Output Format Registers

The clock output format 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 (H1x0) 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 58. Clock Output Format 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.
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.

7.2.3 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.

7 Clock Architecture (continued)

Table 59. 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 APLL1 output, 65.536 MHz.
				0010 0010	Select APLL2 output, 49.704 MHz.
				0011 0000	Select DPLL2 output inverted.
				0011 0001	Select APLL1 output inverted.
				0011 0010	Select APLL2 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.
0x00228 (0x00229) (0x0022A) (0x0022B)	L_SC0 Select L_SC1 Select L_SC2 Select L_SC3 Select	7:0	LC0SR (LC1SR) (LC2SR) (LC3SR)	0000 0000	L_SCx output disabled (default).
				0000 0001	Select OSC1/XTAL1.
				0001 0001	Select OSC1/XTAL1 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.

7 Clock Architecture (continued)

7.3 Clock Register Access

The T8110 clock control registers, 0x00200—0x002FF, 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 7.7.1 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 register access select register, 0x00106; see Section 6.1.4 on page 49.

7.4 Clock Circuit Operation—APLL1

APLL1 can accept either a 4.096 MHz or 2.048 MHz reference clock, and perform a corresponding multiplication function to supply a 65.536 MHz operating clock for the T8110. Additionally, APLL1 may be bypassed for circuit diagnostic purposes. Please refer to Figure 19 on page 62.

7.4.1 Main Clock Selection, Bit Clock, and Frame

APLL1 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 DPLL1.

Table 60. Bit Clock and Frame

Bit Clock	Corresponding 8 kHz Frame	Value(s)
CT_NETREF1, CT_NETREF2	—	1.544 MHz (T1), 2.048 MHz (E1)
NA	CT_NETREF1, CT_NETREF2	8 kHz
CT_C8_A	/CT_FRAME_A	8.192 MHz (ECTF), 4.096 MHz (MC1)
CT_C8_B	/CT_FRAME_B	8.192 MHz (ECTF), 4.096 MHz (MC1)
/C16±	/FR_COMP	6.384 MHz (H-*MVIP*)
/C4	/FR_COMP	4.096 MHz (*MVIP*)
C2	/FR_COMP	2.048 MHz (*MVIP**)
SCLK	/FR_COMP	2.048 MHz, 4.096 MHz, 8.192 MHz (SC-bus)
/SCLKx2	/FR_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]	—	System-specific
LREF[5]	—	System-specific
LREF[6]	—	System-specific
LREF[7]	—	System-specific

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

7 Clock Architecture (continued)

7.4.1.1 Watchdog Timers

A set of watchdog timers is available for all H1x0, 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 DPLL1 and DPLL2 sync inputs, or via the failsafe mechanism; see Section 7.7.2 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 shown below.

Table 61. Watchdog Timer Description

Watchdog	Signal, Value	Description
H1x0 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_A /CT_FRAME_B /FR_COMP	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).
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	/C16± at 16.384 MHz /C4 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.
DPLL1, DPLL2 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.

† DPLL sync reference is expected to be 8 kHz.

7 Clock Architecture (continued)

7.4.1.2 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 7.4.5.1 on page 80. The incoming frame reference signal is sampled with a recovered clock (output of the APLL1 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 62. Frame Center Sampling

Frame Signal	Corresponding Bit Clock	Sample Clock
/CT_FRAME_A	CT_C8_A	Recovered 8.192 MHz, rising edge.
/CT_FRAME_B	CT_C8_B	Recovered 8.192 MHz, rising edge.
/FR_COMP	/C16± (H-*MVIP*) or /C4 (*MVIP*) or C2 (*MVIP*)	Recovered 4.096 MHz, falling edge.
/FR_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.

7.4.2 Main and Resource Dividers

Two independently programmable dividers are available to divide down the main clock selection signal. The 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, the output is 50% duty cycle.
- For a divider value of 193, the output is almost 50% duty cycle (low-level duration is one clock cycle shorter than high-level duration).
- For **all** other divider values, the output is a pulse whose width is one full period of the main clock selection signal.

Output of both dividers is available to the DPLL1 and the APLL1 reference selector. The 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 7.3 on page 76. This allows for immediate loading of the newly activated divider register values.

7 Clock Architecture (continued)

7.4.3 DPLL1

A digital phase-lock loop is provided to generate a 4.096 MHz or 2.048 MHz reference to APLL1, selectable via register 0x0020B (DPLL1 rate). The DPLL1 operates at 32.768 MHz, derived from the XTAL1 crystal input. The DPLL1 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. DPLL1 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 DPLL1 correction, which can either lengthen or shorten its current output clock period by 30.5 ns.

7.4.4 Reference Selector

The APLL1 reference clock is selectable between five possible sources via register 0x00202, APLL1 input selector. A 4.096 MHz or 2.048 MHz reference must be provided. The five possible sources are shown below:

- XTAL1 crystal (16.384 MHz) divided-by-4
- Main divider output
- Resource divider output
- DPLL1 output
- PRI_REF_IN external chip input

7.4.5 Internal Clock Generation

The main internal functions of T8110 are synchronous to the 65.536 MHz output of APLL1. 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 H1x0, H-*MVIP*, *MVIP*, and SC-bus clocks when the T8110 is mastering the bus clocks; see Section 7.2 on page 72. 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 Section 7.4.5.1).

7 Clock Architecture (continued)

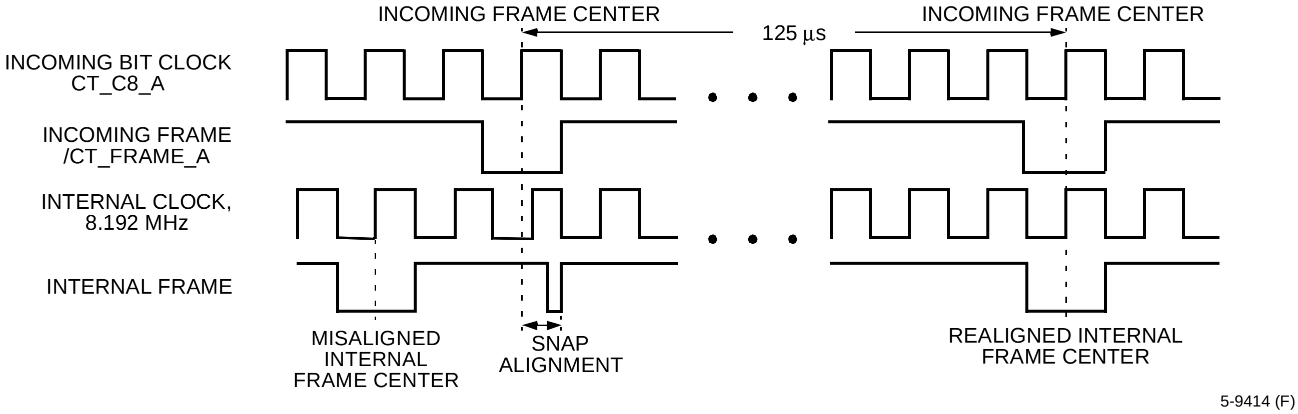
7.4.5.1 Phase Alignment

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:

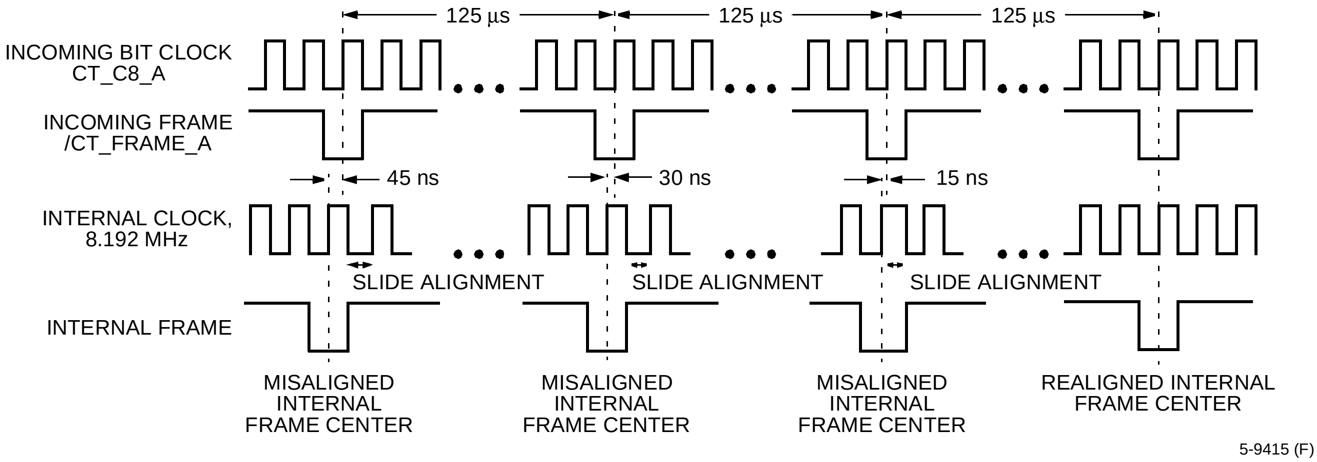
- The main selection clock is based on a paired bit clock/frame reference (see Section 7.4.1.2 on page 78), **and** the phase alignment circuit is enabled (via register 0x00107, phase alignment select).

The incoming frame center is monitored via the frame center samplers (see Section 7.4.1.2 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 as shown below:

- 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 21.



A. Phase Alignment—SNAP



B. Phase Alignment—SLIDE

Figure 21. T8110 Phase Alignment, SNAP and SLIDE

7 Clock Architecture (continued)

7.5 Clock Circuit Operation, APLL2

APLL2 requires either a 6.176 MHz or 12.352 MHz reference clock to produce a 49.408 MHz clock for operating DPLL2. A user-supplied rate multiplier (register 0x00207, APLL2 rate) provides either a times 8 function (when reference clock = 6.176 MHz) or a times 4 function (when reference clock = 12.352 MHz). Additionally, APLL2 may be bypassed for circuit diagnostic purposes (see Figure 19 on page 62).

7.5.1 DPLL2

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 DPLL2 input clock operates at 49.408 MHz from the APLL2 output. Synchronization sources for DPLL2 include the same sources provided to DPLL1 (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. DPLL2 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 DPLL2 correction, which can either lengthen or shorten its current output clock period by 20.2 ns.

7.6 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 20 on page 62).

7.6.1 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).

7.6.2 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 7.3 on page 76). This allows for immediate loading of the newly activated divider register values.

7 Clock Architecture (continued)

7.7 Clock Circuit Operation—Fallback and Failsafe

Fallback is a means to alter the reference source to APLL1 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 APLL1, independent of clock fallback.

7.7.1 Clock Fallback

Clock fallback is a means to alter the APLL1 reference clock source upon detection of a fallback event and is controlled by eight registers, 0x00108—0x0010F (refer to Section 6.1.4 on page 49). These registers enable and control the state transitions that determine which of two clock register sets is used to control the APLL1 reference clock source (see Section 7.1 on page 63 through Section 7.3, Table 64 on page 85, and Figure 23 on page 84).

7.7.1.1 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 63). 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.

7 Clock Architecture (continued)

Table 63. 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

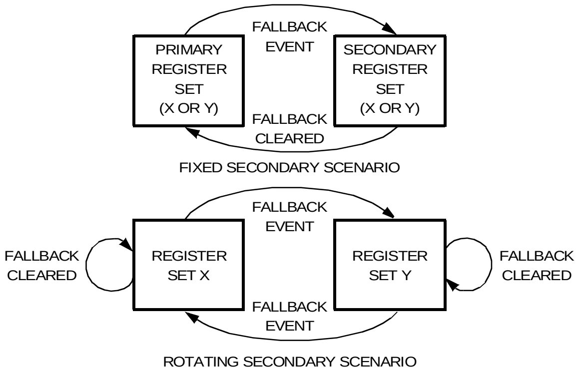
7.7.1.2 Fallback Scenarios—Fixed vs. Rotating Secondary

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.

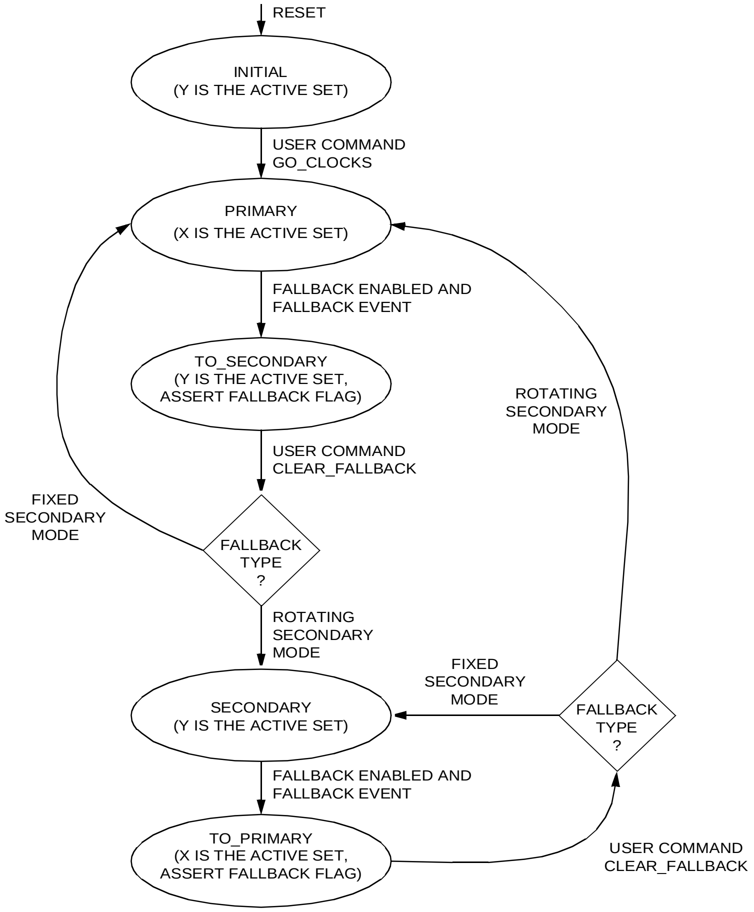
The concepts are illustrated in the figure below.



5-9420 (F)

Figure 22. Fallback—Fixed vs. Rotating Secondary

7 Clock Architecture (continued)



5-9422 (F)

Figure 23. T8110 Clock Fallback States

7 Clock Architecture (continued)

Table 64. Clock Fallback State Description

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 APLL1 REFCLK.	TO_SECONDARY	Fallback is enabled and fallback event* occurs.
TO_SECONDARY	Y is the active clock register set and controls APLL1 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 APLL1 REFCLK.	TO_PRIMARY	Fallback is enabled and fallback event* occurs.
TO_PRIMARY	X is the active clock register set and controls APLL1 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 7.7.1.1 on page 82.

† Fixed, rotating secondary; refer to Section 7.7.1.2 on page 83.

7 Clock Architecture (continued)

7.7.1.3 H-Bus Clock Enable/Disable on Fallback

The previous *Ambassador* devices allowed a fallback mode (A/B fallback) which automatically allowed an H1x0 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_A
- B clocks: CT_C8_B, /CT_FRAME_B
- C clocks: /C16±, /C4, C2, SCLK, /SCLKx2, /FR_COMP

Refer to Figure 24 and Table 65. 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.

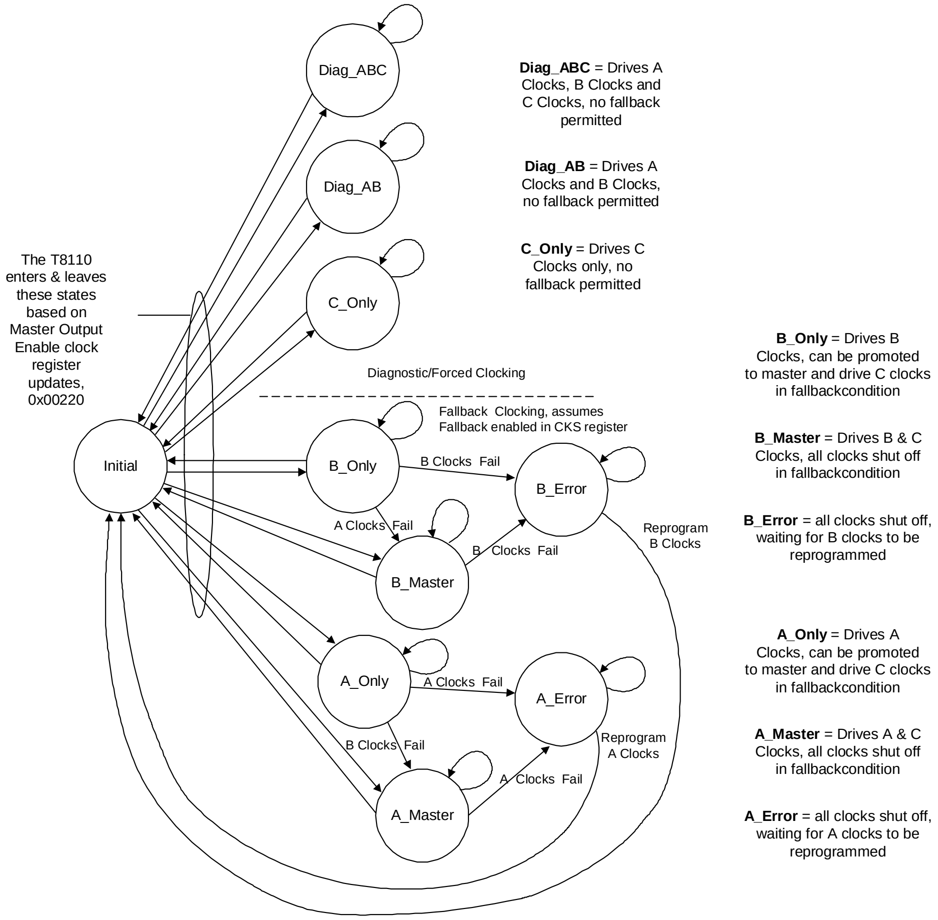


Figure 24. T8110 H-Bus Clock Enable States

7 Clock Architecture (continued)

Table 65. 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_A 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_B is detected; promote to A clock master.
		A_ERROR	A clock error on CT_C8_A or /CT_FRAME_A 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_B is detected; disable clock outputs.
		INITIAL	User update of the clock output control register (0x00220, master output enables).
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_A is detected; promote to B clock master.
		B_ERROR	A clock error on CT_C8_B or /CT_FRAME_B 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).

7 Clock Architecture (continued)

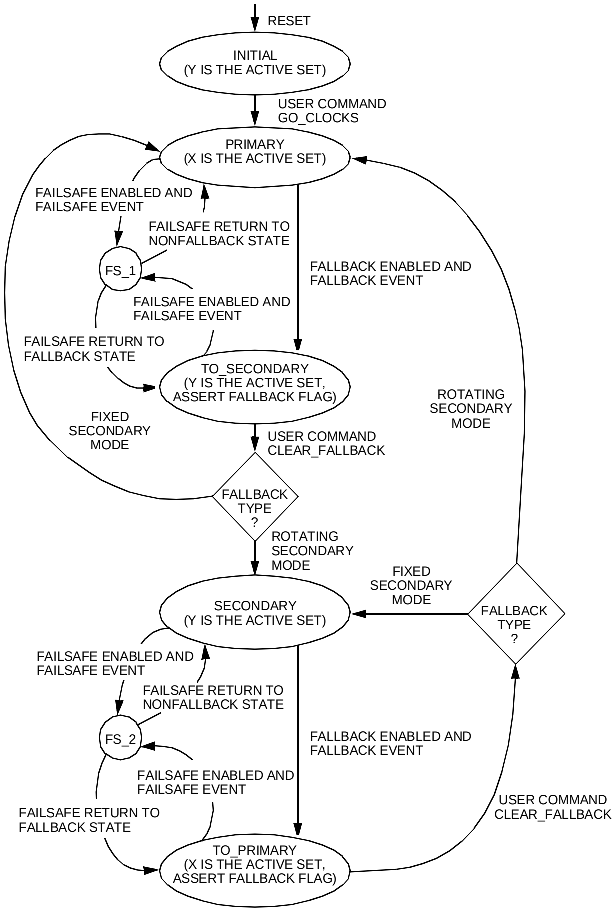
7.7.2 Clock Failsafe

Clock failsafe provides a safety net for the APLL1 reference clock source and is controlled by three registers, 0x00114—0x00116; see Section 6.1.11 on page 54. A failsafe event overrides the active clock control registers and forces the APLL1 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 66 and Figure 25.

7.7.2.1 Failsafe Events

Clock failsafe (transition from either clock register set to a forced XTAL1-div-4 APLL1 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 APLL1 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 APLL1. A lock status flag is provided out of APLL1, and the OOL is defined by exceeding a user-defined threshold value (register 0x00116). The lock status is a flag indicating when APLL1 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).



5-9421 (F)

Figure 25. T8110 Clock Failsafe States

7 Clock Architecture (continued)

Table 66. Clock Failsafe State Descriptions

Clock Failsafe State	Description	Exit To	Exit Condition
FS_1	APLL1 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	APLL1 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).

8 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 67. 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

8.1 Frame Group Control Registers

8.1.1 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 68. 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.
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.

8 Frame Group and FG I/O (continued)

8.1.2 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 69).

Table 69. 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	1-bit wide pulse (default). 1-bit wide pulse. 2-bit wide pulse. 4-bit wide pulse. 1-byte wide pulse. 2-byte wide pulse.

8.1.3 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.

Table 70. FGx Rate Registers

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

* FGIO operation is controlled at registers 0x00480—482. Refer to Section 8.3 on page 93.

8 Frame Group and FG I/O (continued)

8.2 FG7 Timer Option

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

8.2.1 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 8.4.3 on page 97.

Table 71. 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	FCISB	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 = 61.0 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.

8 Frame Group and FG I/O (continued)

8.3 FGIO Control Registers

8.3.1 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 72. 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.

8.3.2 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 73. 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.

8 Frame Group and FG I/O (continued)

8.3.3 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 74. FGIO R/W Register

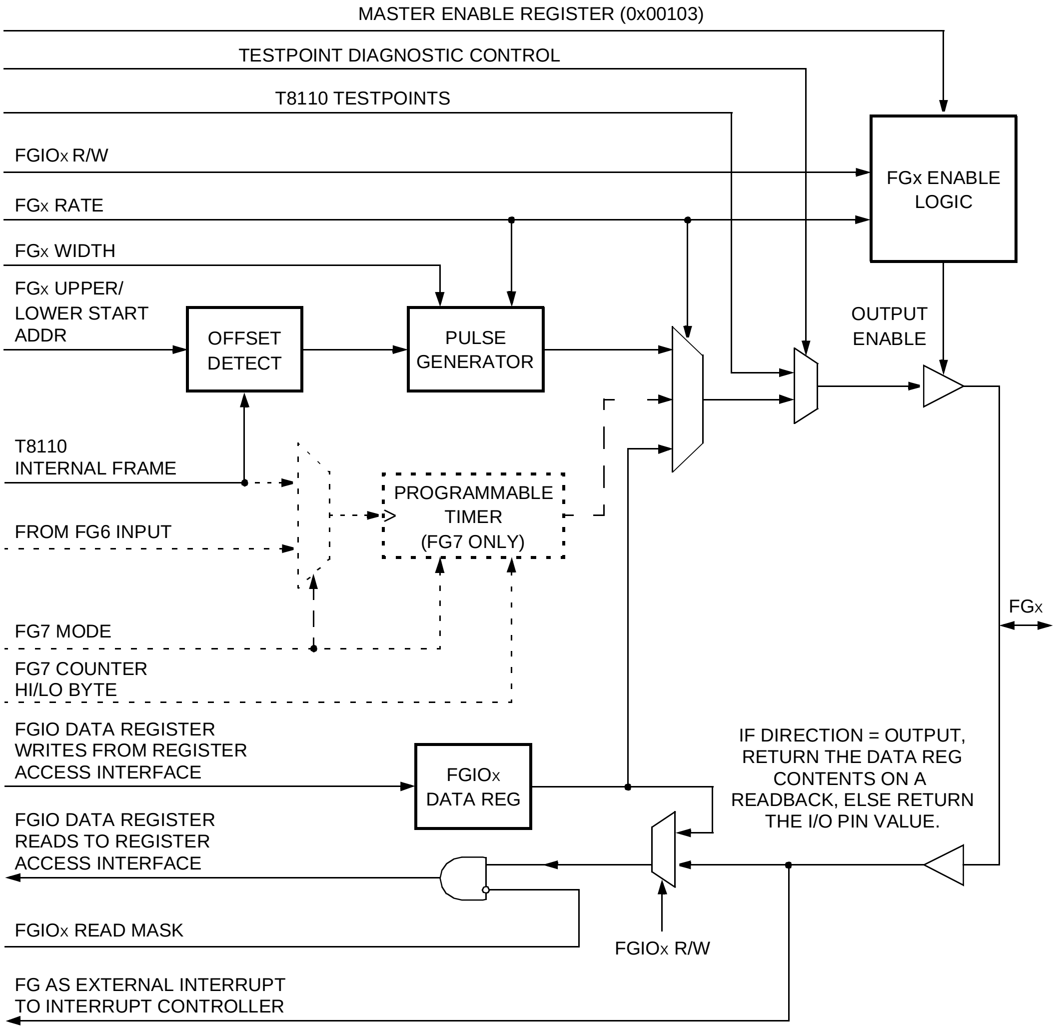
Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00482	FGIO R/W	7	F7DSB	0 1	FGIO bit 7 direction is input (default). FGIO bit 7 direction is output.
		6	F6DSB	0 1	FGIO bit 6 direction is input (default). FGIO bit 6 direction is output.
		5	F5DSB	0 1	FGIO bit 5 direction is input (default). FGIO bit 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.

8 Frame Group and FG I/O (continued)

8.4 FG Circuit Operation

Each of the eight frame group signals FG[7:0] operate independently and have multiple uses. Refer to Figure 26 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



5-9428a (F)

Figure 26. FG[7:0] Functional Paths

8 Frame Group and FG I/O (continued)

8.4.1 Frame Group 8 kHz Reference Generation

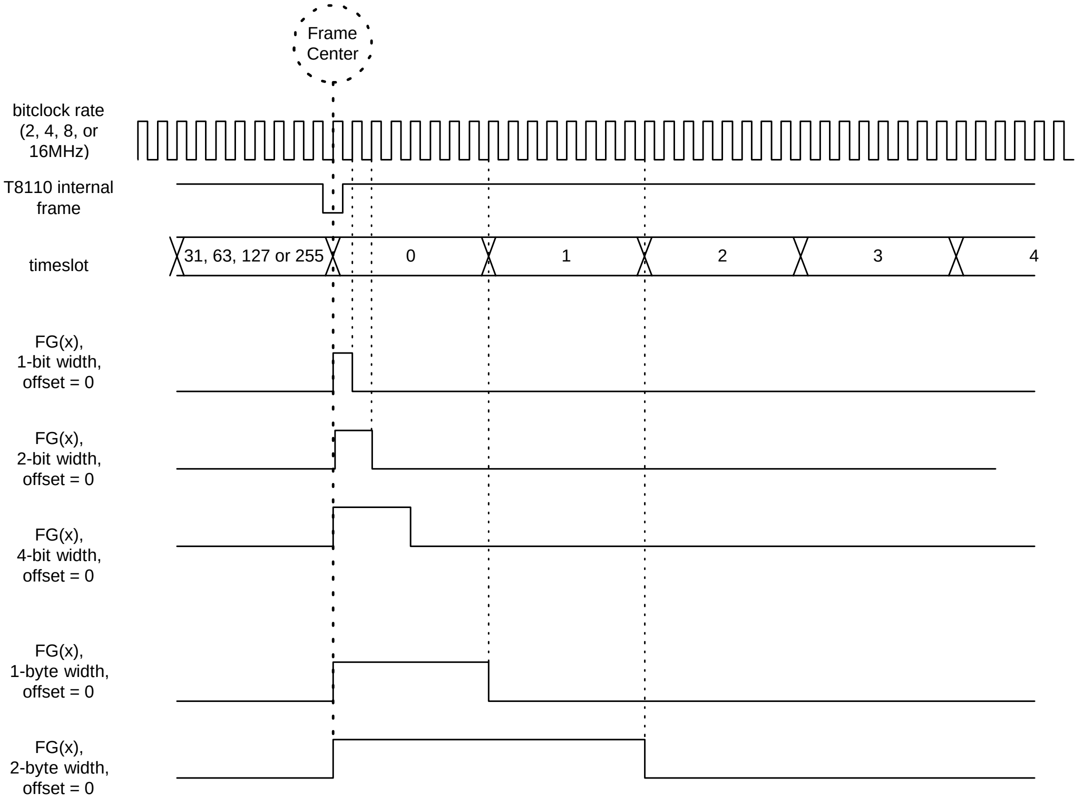
Any of the T8110 FG signals may be used as programmable 8 kHz frame reference outputs. There are two sets of control required, an offset delay from internal frame center, and pulse shaping.

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 32.768 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



Notes:

Frame group signals shown with offset = 0 (default). At offset = 0, the pulse starts at frame center.

Nonzero offsets denote 32.768 MHz period increments (30.5 ns) from frame center. There are up to 4096 increments within an 8 kHz frame period. Offsets may be programmed in the range from 0—4095.

Frame group signals are shown as active high pulses (default)—they may be programmed as active-low pulses.

Diagram shows frame group pulse widths relative to bit-clock rate and time-slot width. This is applicable for any of the four frame group data rates (2 Mbits/s, 4 Mbits/s, 8 Mbits/s, or 16 Mbits/s).

Figure 27. Frame Group 8 kHz Reference Timing

8 Frame Group and FG I/O (continued)

8.4.2 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.

8.4.3 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].

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-65536). 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 32.768 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).

8.4.4 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, see Section 8.4.1 on page 96). The interrupt control registers (0x00600—603) control how the FG inputs are handled (for more details, refer to Section 12.1 on page 113).

8.4.5 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 test-point enable. Settings in this register override the FGx rate and FGIO R/W register, and force the selected bits to be test-point outputs, see Section 13.1 on page 128 and Table 103 on page 128.

9 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.

9.1 GPIO Control Registers

Table 75. 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

9.1.1 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 76. 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.

9 General-Purpose I/O (continued)

9.1.2 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 77. 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.

9.1.3 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 78. 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.

9 General-Purpose I/O (continued)

9.1.4 GPIO Override Register

Table 79. GPIO Override Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00503	GPIO Override	7:3	Reserved	0000 0	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.

9.2 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 28 on page 100.

- 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

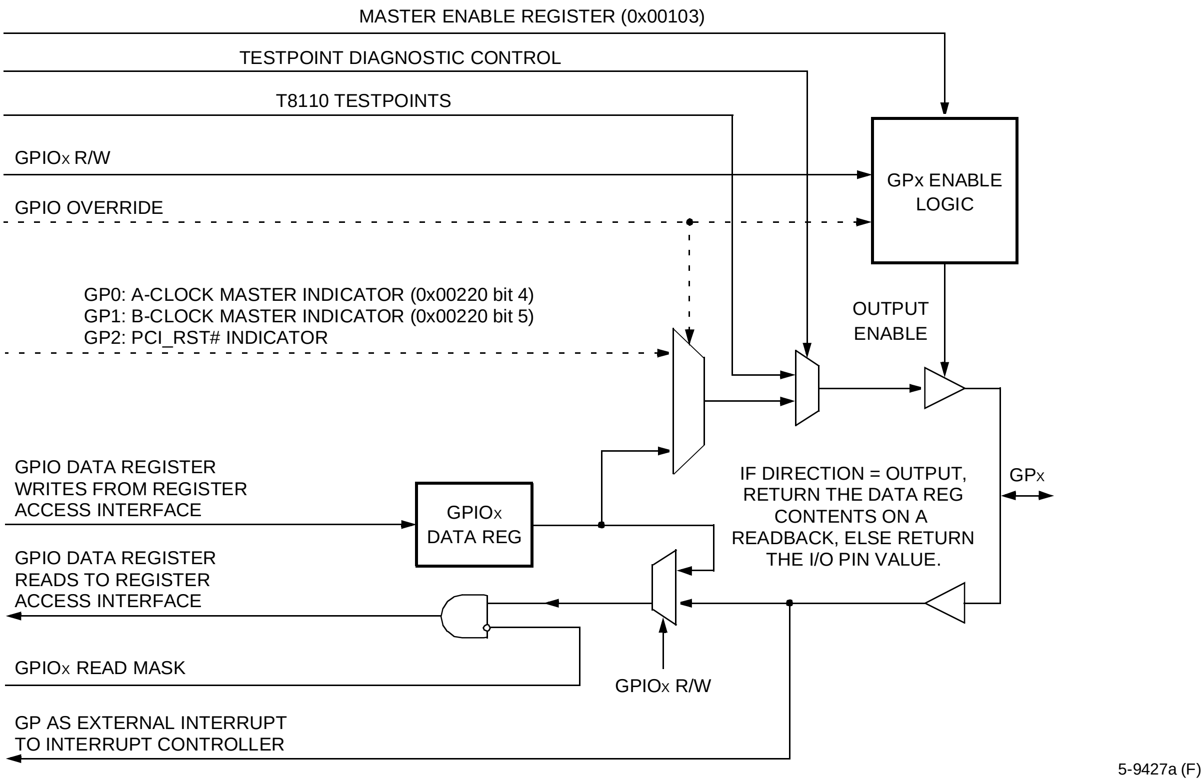


Figure 28. GP[7:0] Functional Paths

9 General-Purpose I/O (continued)

9.2.1 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 a GPIO bit is masked, a read access returns 0.

9.2.2 GP Dual-Purpose Bits GPIO (Override)

9.2.2.1 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.

9.2.2.2 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 11.2 on page 110). This function provides access to a forwarded PCI_RST# signal by external devices hanging off the minibridge port.

9.2.3 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 12.1 on page 113.

9.2.4 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 test-point enable. Settings in this register override the GPIO R/W register and force the selected bits to be test-point outputs (refer to Section 13.1 on page 128, and Table 105 on page 130).

10 Stream Rate Control

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

Table 80. 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 10.2.2 on page 104).

10 Stream Rate Control (continued)

10.1 H-Bus Stream Rate Control Registers

10.1.1 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.

Table 81. 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

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

10.2 L-Bus Stream Rate Control Registers

10.2.1 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 10.2.2 on page 104).

Table 82. L-Bus Rate Registers

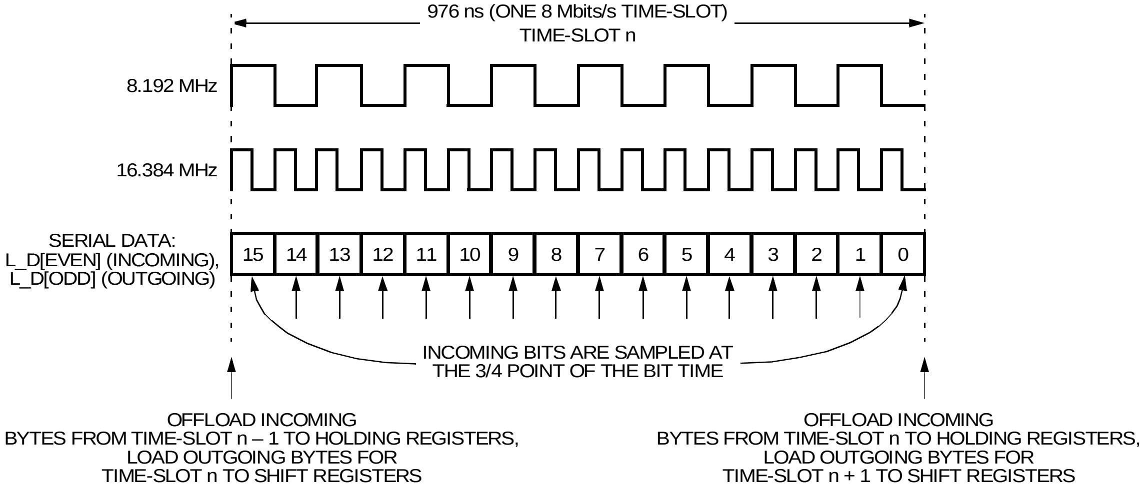
DWORD Address (20 Bits)	Register			
	Byte 3	Byte 2	Byte 1	Byte 0
0x00320	L-bus rate H/G	L-bus rate F/E	L-bus rate D/C	L-bus rate B/A

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

10 Stream Rate Control (continued)

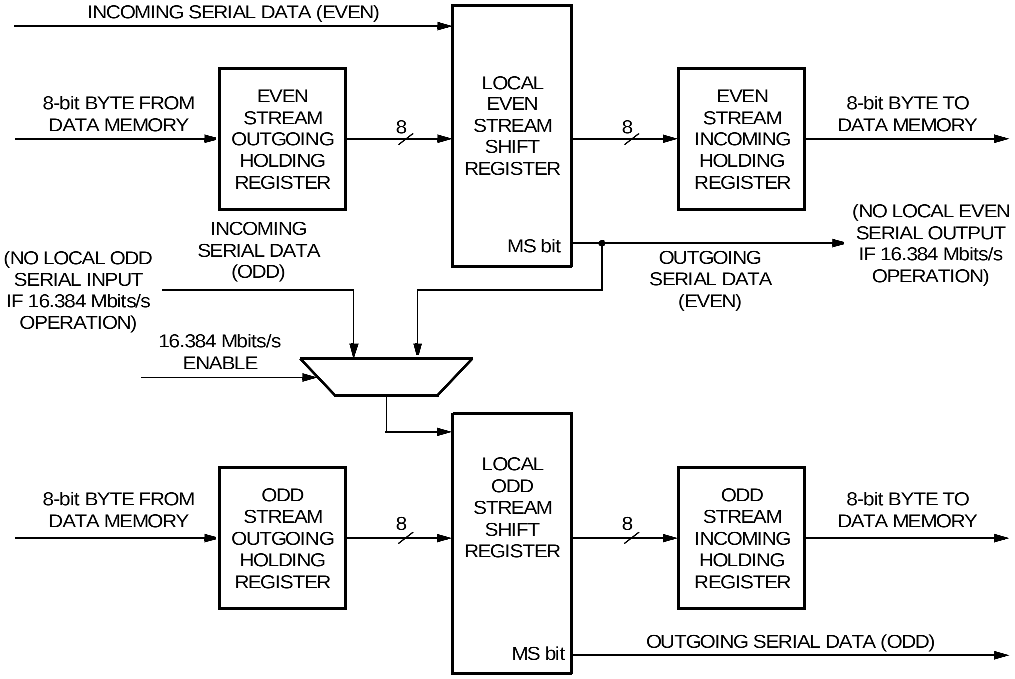
10.2.2 L-Bus 16.384 Mbits/s Operation

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 29). 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 30).



5-9411 (F)

Figure 29. Local Stream 16.384 Mbits/s Timing



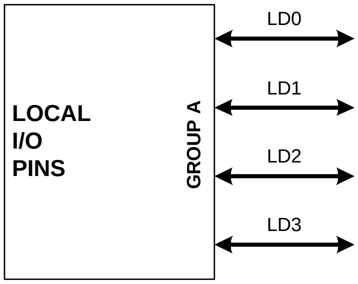
5-9426 (F)

Figure 30. Local Stream 16.384 Mbits/s Circuit

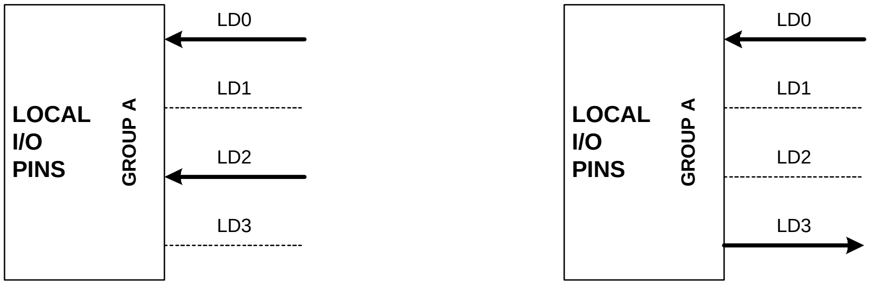
10 Stream Rate Control (continued)

10.2.3 16.384 Mbits/s Local I/O Superrate

This 16.384 Mbits/s rate option is available only on the local I/O streams (i.e., it is not supported as a part of the H.100/H.110 specifications). When applying the superrate option to a local I/O group, the I/O for the group is redefined and divided into two pairs of input and output. An input or an output can be selected from each pair, but both can't be used simultaneously. This leads to four possible configurations for each group. Note that inputs are always on even signals and outputs are always on odd signals. Thus, if all local groups are operated at the superrate, then the application can have 16 lines, all at 16.384 Mbits/s, in contrast to the 32 I/O lines at normal rates.

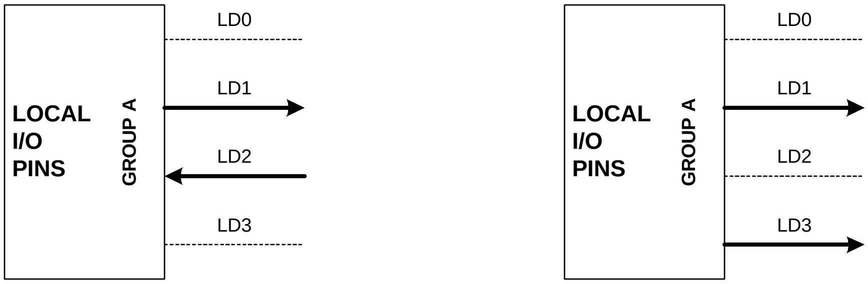


GROUP A CONFIGURATION AT 2.048, 4.096,
OR 8.192 Mbits/s



GROUP A CONFIGURATION 1
AT 16.384 Mbits/s SUPERRATE

GROUP A CONFIGURATION 2
AT 16.384 Mbits/s SUPERRATE



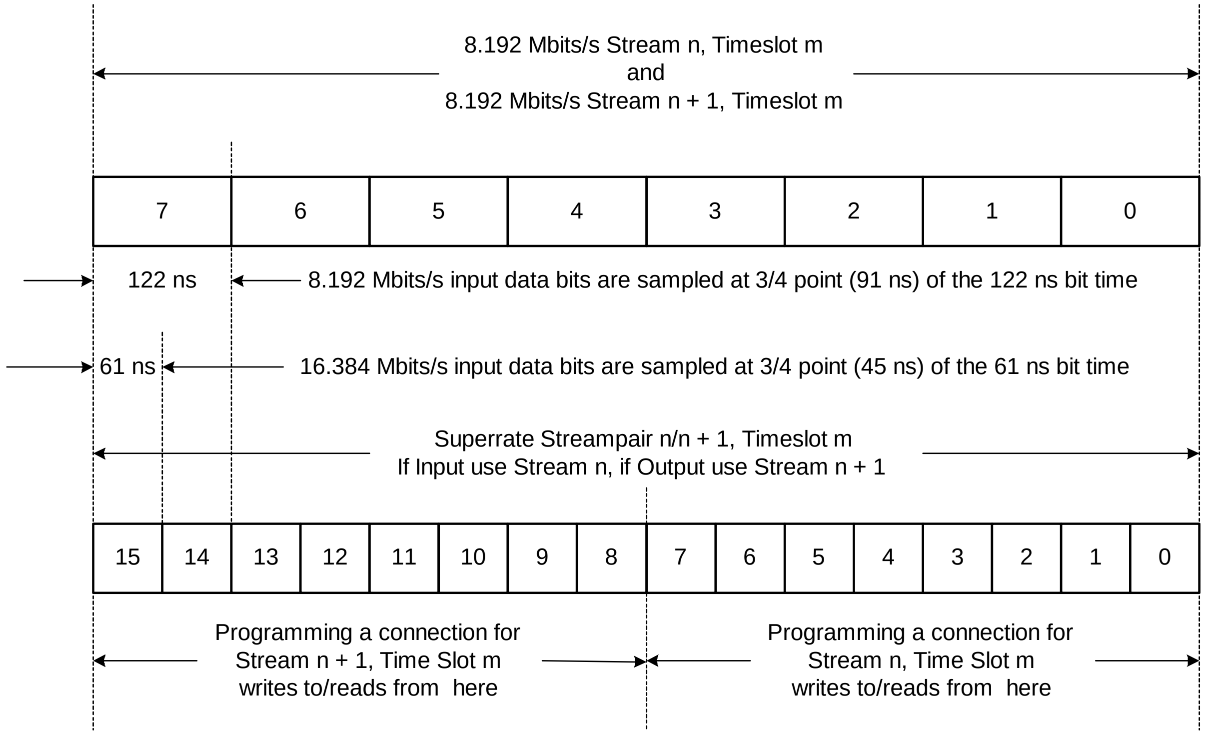
GROUP A CONFIGURATION 3
AT 16.384 Mbits/s SUPERRATE

GROUP A CONFIGURATION 4
AT 16.384 Mbits/s SUPERRATE

Figure 31. Superrate I/O Configuration

10 Stream Rate Control (continued)

The configurations are selected as a consequence of the connection programming. The data is inputted or outputted as a true 16-bit at 16.384 Mbits/s signal. Programming a 16-bit connection requires two separate byte connections, one for the MS-byte and the other for the LS-byte.



Note: n = even number, m = integer.

Figure 32. Relationship Between 8.192 Mbits/s and 16.384 Mbits/s Time-Slots

Thus, programming a connection **to** stream n + 1 is programming a connection to the MS-byte on output pin n + 1 and programming a connection **to** stream n is programming a connection to the LS-byte on output pin n + 1. Similarly, programming a connection **from** stream n + 1 is programming a connection from the MS-byte on input pin n and programming a connection **from** stream n is programming a connection from the LS-byte on input pin n.
(An easier way to remember this is that the even/odd identifier becomes the MS-byte/LS-byte identifier.)

As a consequence of this arrangement, the T8110 permits byte-packing at the superrate in analogous manner to subrate bit-packing.

11 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 a microprocessor bus, the minibridge function is not applicable, and the minibridge port pins are used for the microprocessor interface. Refer to Section 5 on page 38.

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, a 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 6.1.1 on page 46 and Section 9.2.2.2 on page 101. 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].

There is a direct mapping of the PCI address bits to the minibridge address bits. Users 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 for PCI MEMORY transactions (which is the only type of transaction T8110 responds to), the value is always 00.

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.

11.1 Wait-State Control Registers

11.1.1 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 29 on page 104.

Table 83. 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

11 Minibridge (continued)

Table 84. 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.

11 Minibridge (continued)

Table 84. 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.

11 Minibridge (continued)

11.2 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 85. 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	R/W Strobe Inversion	7:3	Reserved	0000 0	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.

11.3 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.

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 4.1.6 on page 30 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.

11 Minibridge (continued)

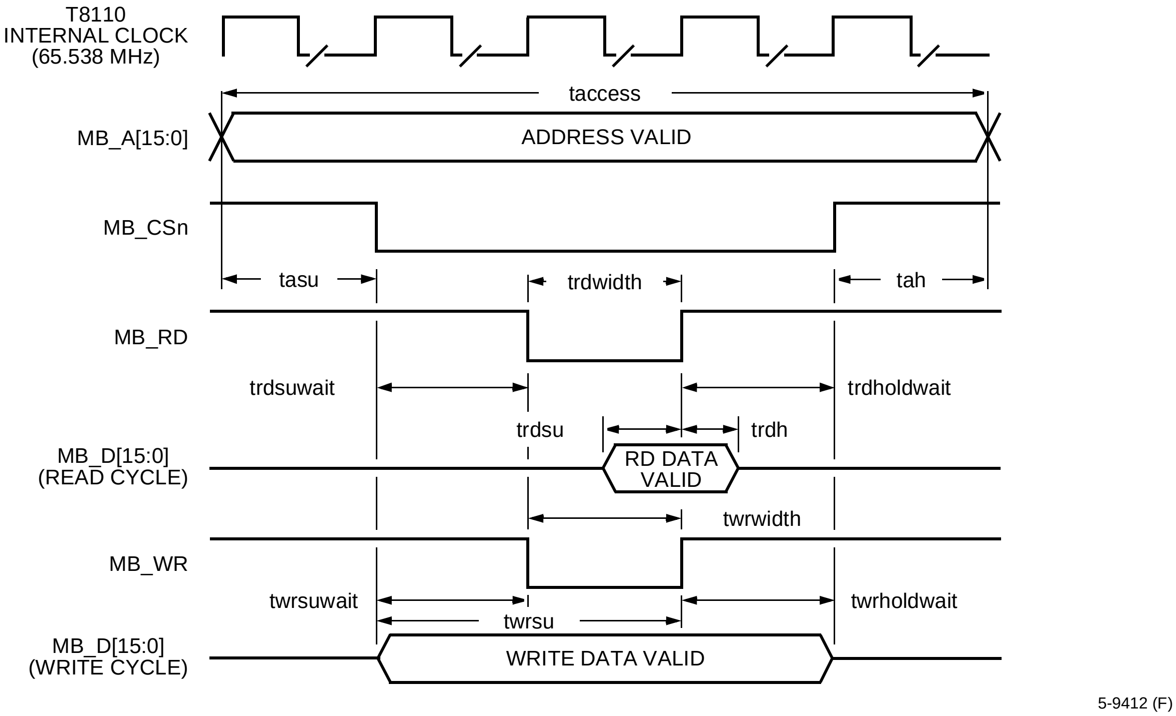


Figure 33. Minibridge Read/Write Access Cycles

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).

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.

11 Minibridge (continued)

— 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).

11.4 Minibridge Operational Addressing

The operating space (in PCI) is from 0x70000—0x7FFFF. 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 86. Minibridge Operating Space (PCI)

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.

12 Error Reporting and Interrupt Control

12.1 Interrupt Control Registers

Table 87. 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
0x006FC	In-service, byte 3	In-service, byte 2	In-service, byte 1	In-service, byte 0

12.1.1 Interrupts Via External FG[7:0] Registers

12.1.1.1 FGIO Interrupt Pending Register

The FGIO interrupt pending register stores detected interrupts via the FG[7:0] signals. The 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 88. 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.

12 Error Reporting and Interrupt Control (continued)

Table 88. 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 FGIO edge/level and FGIO polarity registers control how interrupts are interpreted on the GP[7:0] signals (negative edge, positive edge, low level, or high level).

Table 89. 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.

12 Error Reporting and Interrupt Control (continued)

12.1.2 Interrupts Via External GP[7:0]

12.1.2.1 GPIO Interrupt Pending Register

The GPIO interrupt pending register stores detected interrupts via the GP[7:0] signals. The 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 90. 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.

12 Error Reporting and Interrupt Control (continued)

12.1.2.2 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 91. 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.

12.1.3 Interrupts Via Internal System Errors

Table 92. 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 overwrite warning.
SYS2	PCI master, external buffer initial warning.
SYS1	VC memory, scratchpad overflow warning.
SYS0	NOTIFY_QUEUE, overflow warning.

12 Error Reporting and Interrupt Control (continued)

12.1.4 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 6.2.5 on page 59). The user can clear specific bits by writing 1 to that bit (write-1-to-clear).

Table 93. 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.

12 Error Reporting and Interrupt Control (continued)

12.1.5 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 94. 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.

12 Error Reporting and Interrupt Control (continued)

12.1.6 Interrupts Via Internal Clock Errors

Table 95. Clock Error Interrupt Assignments

Clock Interrupt Bit	Description
CLK15	Failsafe indicator—APLL1 reference error.
CLK14	DPLL2 sync input error.
CLK13	DPLL1 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.

12 Error Reporting and Interrupt Control (continued)

12.1.7 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 6.2.1 on page 56). The user can clear specific bits by writing 1 to that bit (write 1 to clear).

Table 96. 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.

12 Error Reporting and Interrupt Control (continued)

12.1.8 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 97. 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.

12 Error Reporting and Interrupt Control (continued)

12.1.9 Interrupt Servicing Registers

12.1.9.1 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 pre-empt a current in-service interrupt according to the three priority levels.

Table 98. 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.

12.1.10 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 99. 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.

12.1.10.1 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 SYSERR pulse-width register controls how wide the SYSERR pulse is (when selected output format = high or low pulse). Value corresponds to the number of 32.768 MHz periods – 1.

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).

The CLKERR pulse-width register controls how wide the CLKERR pulse is (when selected output format = high or low pulse). Value corresponds to the number of 32.768 MHz periods – 1.

12 Error Reporting and Interrupt Control (continued)

Table 100. 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	JSWSR	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	JCWSR	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.

12.1.10.2 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 101. Interrupt In-Service Register

Byte Address	Register Name	Bit(s)	Mnemonic	Value	Function
0x006FC	Interrupt In-service Low	7:0	Reserved	0000 000	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.

12 Error Reporting and Interrupt Control (continued)

Table 101. Interrupt In-Service Register (continued)

Byte Address	Register Name	Bit(s)	Mnemonic	Value	Function
0x00619	Interrupt In-service High	7:0		0100 0110 0100 0111 0100 1000 0100 1001 0100 1010 0100 1011 0100 1100 0100 1101 0100 1110 0100 1111 1000 0000 1000 0001 1000 0010 1000 0011 1000 0100 1000 0101 1000 0110 1000 0111 1000 1000 1000 1001 1000 1010 1000 1011 1000 1100 1000 1101 1000 1110 1000 1111	SYS6 interrupt in-service. SYS7 interrupt in-service. SYS8 interrupt in-service. SYS9 interrupt in-service. SYS10 interrupt in-service. SYS11 interrupt in-service. SYS12 interrupt in-service. SYS13 interrupt in-service. SYS14 interrupt in-service. SYS15 interrupt in-service. CLK0 interrupt in-service. CLK1 interrupt in-service. CLK2 interrupt in-service. CLK3 interrupt in-service. CLK4 interrupt in-service. CLK5 interrupt in-service. CLK6 interrupt in-service. CLK7 interrupt in-service. CLK8 interrupt in-service. CLK9 interrupt in-service. CLK10 interrupt in-service. CLK11 interrupt in-service. CLK12 interrupt in-service. CLK13 interrupt in-service. CLK14 interrupt in-service. 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 000	NOP.
		0		L	MS bit, virtual channel identifier.

12 Error Reporting and Interrupt Control (continued)

12.2 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.

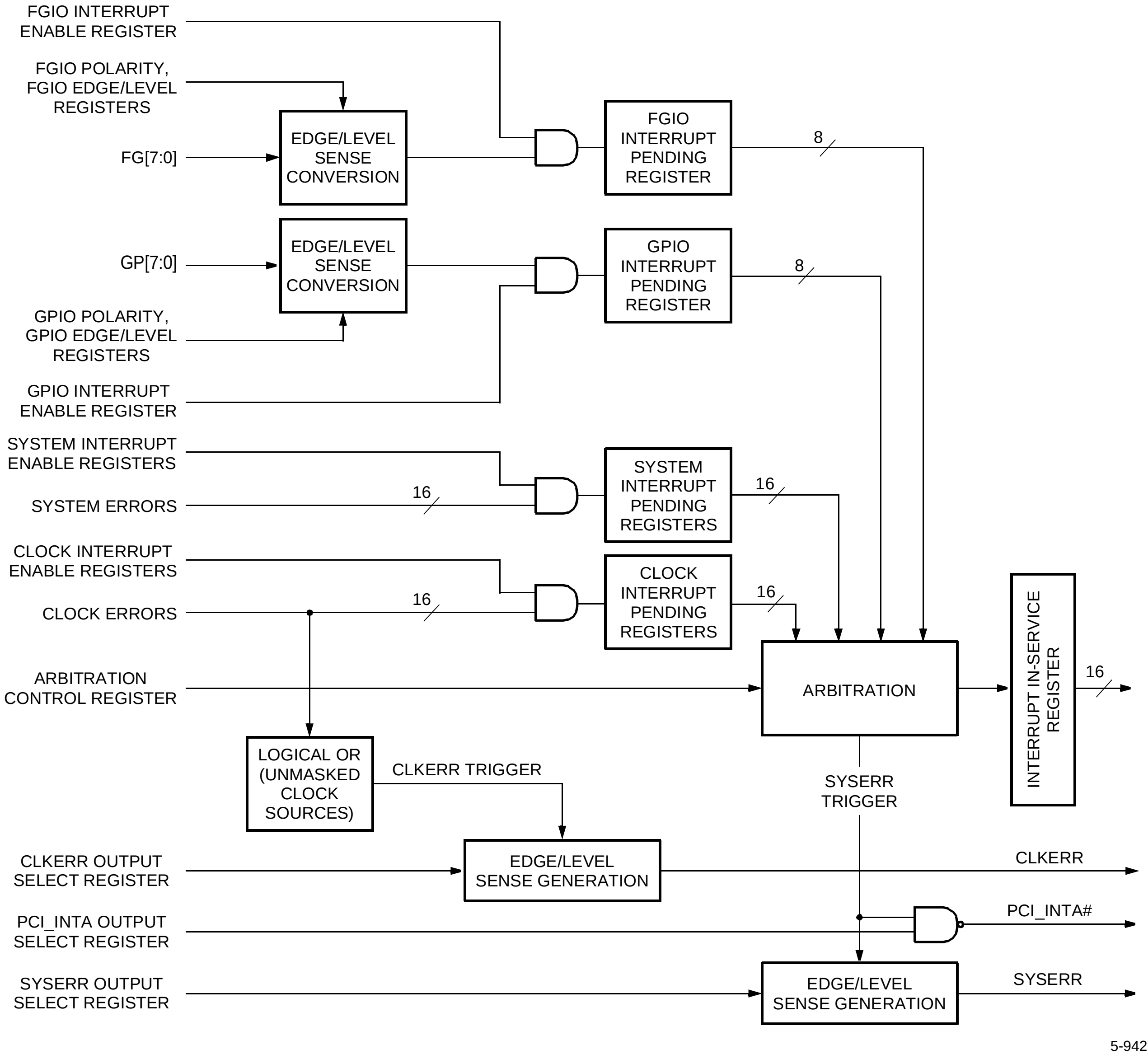


Figure 34. Interrupt Controller

12 Error Reporting and Interrupt Control (continued)

12.2.1 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 12.1.1 on page 113 and Section 12.1.2 on page 115). 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.

12.2.2 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 6.2.5 on page 59). Each of these inputs is independently controlled via the interrupt control registers (refer to Section 12.1.3 on page 116). 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.

12.2.3 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 6.2.1 on page 56). Each of these inputs is independently controlled via the interrupt control registers (refer to Section 12.1.6 on page 119). 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.

12.2.4 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).

12.2.4.1 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.

12.2.4.2 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 12.2.8 on page 127), after which the next arbitration cycle takes place.

12.2.4.3 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.

12 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.

12.2.4.3.1 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 12.2.8 on page 127), at which time another arbitration cycle takes place.

12.2.4.3.2 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 12.2.8 on page 127), the stack is popped and the original lower-priority interrupt is reissued.

12.2.5 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.

12.2.6 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.

12.2.7 PCI_INTA# Output

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

12.2.8 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 as shown below:

- 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.

13 Test and Diagnostics

13.1 Diagnostics Control Registers

The diagnostic control registers allow for various diagnostic modes (refer to Section 13.2 on page 135).

Table 102. Diagnostics Control Register Map

DWORD Address (20 bits)	Register			
	Byte 3	**Byte 2**	**Byte 1**	**Byte 0**
0x00140	Diag3, GP test-point select	Diag2, GP test-point enable	Diag1, FG test-point select	Diag0, FG test-point enable
0x00144	Diag7, external buffer RETRY timer	Diag6, miscellaneous diagnostics low	Diag5, state counter modes high	Diag4, state counter modes low
0x00148	Diag11, sync-to-frame command offset high	Diag10, sync-to-frame command offset low	Diag9, interrupt controller SYSERR delay	Diag8, interrupt controller diagnostics

13.1.1 FG Testpoint Enable Register

The FG test-point enable register allows individual programming of FG[7:0] bits for either standard operation (as FG or FGIO) or as test-point outputs. FG test-point select controls the MUX selection for which test-points are selected. Refer to Table 104 on page 129 for test-point assignments for each FG bit.

Table 103. 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 test-point.
		6	FT6EB	0 1	FG6 is standard FG or FGIO bit (default). FG6 is a test-point.
		5	FT5EB	0 1	FG5 is standard FG or FGIO bit (default). FG5 is a test-point.
		4	FT4EB	0 1	FG4 is standard FG or FGIO bit (default). FG4 is a test-point.
		3	FT3EB	0 1	FG3 is standard FG or FGIO bit (default). FG3 is a test-point.
		2	FT2EB	0 1	FG2 is standard FG or FGIO bit (default). FG2 is a test-point.
		1	FT1EB	0 1	FG1 is standard FG or FGIO bit (default). FG1 is a test-point.
		0	FT0EB	0 1	FG0 is standard FG or FGIO bit (default). FG0 is a test-point.
0x00141	Diag1, FG Testpoint Select	7:0	FTPSR	LLLL LLLL	Value for MUX selection of test-points output to FG[7:0]—see Table 103 on page 128.

13 Test and Diagnostics (continued)

Table 104. FG[7:0] Internal Testpoint Assignments

FG Testpoint Select Value	FG7	FG6	FG5	FG4
0000 0001	i_FRAME	STATE_COUNT[10:4] (actual time-slot)		
0000 0010	i_FRAME	STATE_COUNT_LOOKAHEAD (lookahead time-slot)		
0000 0100	i_FRAME	STATE_COUNT_LOOKBEHIND (lookbehind time-slot)		
0000 1000	TAR_VALID_CMD	TAR_FORCE_RETRY	Reserved	TRCV FIFO FLUSH
0001 0000	MAS_START_DMA	HOST1_COMPLETE	HOST1_FATAL	MRCV FIFO FLUSH
0010 0000	Reserved			
0100 0000	P_S_SELECTOR	Reserved	OOL threshold flag	APLL1 lock indicator
1000 0000	Stalled	Snapping	C clock enable	B clock enable

—	FG3	FG2	FG1	FG0
0000 0001	STATE_COUNT[10:4] (actual time-slot)			
0000 0010	STATE_COUNT_LOOKAHEAD (lookahead time-slot)			
0000 0100	STATE_COUNT_LOOKBEHIND (lookbehind time-slot)			
0000 1000	TRCV FIFO PERR	TRCV FIFO EMPTY	TRCV FIFO LASTOUT	TRCV FIFO READ
0001 0000	Terminate master done	MRCV FIFO empty	Reserved	MRCV FIFO READ
0010 0000	Reserved			
0100 0000	Failsafe flag	Force-to-OSC4 flag	Return from FS2 flag	Return from FS1 flag
1000 0000	A clock enable	Encoded ABC states: 000 or 100 = DIAGS 001 = A_ONLY 010 = A_MASTER 011 = A_ERROR 101 = B_ONLY 110 = B_MASTER 111 = B_ERROR		

13.1.2 GP Testpoint Enable Register

The GP test-point enable register allows individual programming of GP[7:0] bits for either standard operation (as GPIO) or as test-point outputs. GP test-point select controls the MUX selection for which test-points are selected. Refer to Table 106 on page 131 for test-point assignments for each GP bit.

13 Test and Diagnostics (continued)

Table 105. 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 test-point.
		6	GT6EB	0 1	GP6 is standard GPIO bit (default). GP6 is a test-point.
		5	GT5EB	0 1	GP5 is standard GPIO bit (default). GP5 is a test-point.
		4	GT4EB	0 1	GP4 is standard GPIO bit (default). GP4 is a test-point.
		3	GT3EB	0 1	GP3 is standard GPIO bit (default). GP3 is a test-point.
		2	GT2EB	0 1	GP2 is standard GPIO bit (default). GP2 is a test-point.
		1	GT1EB	0 1	GP1 is standard GPIO bit (default). GP1 is a test-point.
		0	GT0EB	0 1	GP0 is standard GPIO bit (default). GP0 is a test-point.
0x00143	Diag3, GP Testpoint Select	7:0	GTPSR	LLLL LLLL	Value for MUX selection of test-points output to GP[7:0]—see Table 105 on page 130.

13 Test and Diagnostics (continued)

Table 106. GP[7:0] Internal Testpoint Assignments

GP Testpoint Select Value	**GP7**	**GP6**	**GP5**	**GP4**
0000 0001	BYTEREF_16	BYTEREF_8	BYTEREF_4	BYTEREF_2
0000 0010	i_FRAME	Reserved	CP8 read	CP8 write
0000 0100	Reserved			
0000 1000	TAR_VALID_CMD	TAR_FORCE_RETRY	Reserved	TXMT FIFO flush
0001 0000	MAS_START_DMA	HOST1_COMPLETE	HOST1_FATAL	MXMT FIFO flush
0010 0000	DPLL2 lock	DPLL1 lock	Reserved	
0100 0000	P_S_SELECTOR	Fallback encoded states: 000 = PRIMARY 001 = TO_PRIMARY 010 = SECONDARY 011 = TO_SECONDARY 100 = FS1 101 = FS2 110 = [reserved] 111 = INITIAL		
1000 0000	Stalled	Snapping	Reserved	Reserved

—	**GP3**	**GP2**	**GP1**	**GP0**
0000 0001	STATE_COUNT[3:0] (stream)			
0000 0010	CP4 read	CP4 write	CP2 read	CP2 write
0000 0100	Reserved			
0000 1000	TXMT FIFO full	TXMT FIFO empty	TXMT FIFO stop	TXMT FIFO write
0001 0000	MXMT FIFO full	MXMT FIFO empty	MXMT FIFO lastin	MXMT FIFO write
0010 0000	Reserved			
0100 0000	Fallback flag	Go_clocks indicator	CLEAR_FALLBACK indicator	FORCE_FALLBACK indicator
1000 0000	Phase alignment frame event	APLL1 feedback, 8 MHz tap	APLL1 feedback, 4 MHz tap	APLL1 feedback, 2 MHz tap

13 Test and Diagnostics (continued)

13.1.3 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 13.2 on page 135.

Table 107. State Counter Modes Registers

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00144	Diag4, State Counter Modes Low	7:0	SCMLR	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	SCULP	LLL	Upper 3 bits of state counter modulo load value.

13 Test and Diagnostics (continued)

13.1.4 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 microprocessor access to the minibridge register space. Bit 3 enables microprocessor access to the virtual channel memory region. Bits 2 and 1 allow direct reset of the APLL2 and APLL1 feedback dividers. Bit 0 controls the TST input of the power-on reset cell.

Table 108. Miscellaneous Diagnostics Low Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00146	Diag6, Miscellaneous Diagnostics Low	7:6	Reserved	00	NOP (default).
		5	PDTSB	0 1	PCI target discard timers normal (default). PCI target discard timers shortened.
		4	MBIEB	0 1	Microprocessor access to minibridge registers disabled (default). Microprocessor access to minibridge registers enabled.
		3	VCMEB	0 1	Microprocessor access to VC memory disabled (default). Microprocessor access to VC memory enabled.
		2	FB2SB	0 1	APLL2 feedback divider reset inactive (default). APLL2 feedback divider reset active.
		1	FB1SB	0 1	APLL1 feedback divider reset inactive (default). APLL1 feedback divider reset active.
		0	Reserved	—	—

13 Test and Diagnostics (continued)

13.1.5 External Buffer Retry Timer Register

The external buffer retry timer register provides a time-out 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 14.2.3.4 for more detail.

Table 109. External Buffer Retry Timer Register

Byte Address	Name	Bit(s)	Mnemonic	Value	Function
0x00147	Diag7, External Buffer Retry Timer	7:0	EBOLR	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	ICKLP	LL	DIAG mode, force CLK[1:0] errors.
		3:2	ISYLP	LL	DIAG mode, force SYS[1:0] errors.
		1:0	IEXLP	LL	DIAG mode, force EXT[8, 0] errors.
0x00149	Diag9, Interrupt Controller Deassertion Delay	7:0	IASLR	LLLL LLLL	Programmable delay to control the deassertion time of SYSERR.
0x0014A	Diag10, Sync-to-frame Command Delay (Lower)	7:0	CFLLR	LLLL LLLL	Low byte of 12-bit offset value for sync-to-frame clock commands (GO_CLOCKS, CLEAR_FALLBACK, FORCE_FALLBACK).
0x0014B	Diag11, Sync-to-frame Command Delay (Upper)	7:4	CFSEN	0000 0001	Disable delay mode (default). Enable delay mode.
		3:0	CFHLN	LLLL	Upper 4 bits of 12-bit offset value for sync-to-frame clock commands (GO_CLOCKS, CLEAR_FALLBACK, FORCE_FALLBACK).

13 Test and Diagnostics (continued)

13.2 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 104 on page 129 and Table 106 on page 131 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, microprocessor access to the virtual channel memory and minibridge register regions:
  - — The VC memory 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 microprocessor 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 microprocessor bus is selected, this diagnostic setting allows direct access to the minibridge registers.
- DIAG6, forced RESET of analog APLL1 feedback dividers:
  - — The APLL1 feedback dividers are typically not reset. This diagnostic mode allows each feedback divider to be held in a reset state.
- 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. Please see Section 14.2.3.4 on page 160 for more details.
- 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. Please see Section 12.2 on page 125 for more details.
- DIAG9, interrupt controller deassertion delay:
  - — Allows a programmable deassertion time for the SYSERR signal in between back-to-back interrupts.
- DIAG10—11, sync-to-frame command delay:
  - — Allows a programmable delay time from the FRAME boundary for execution of the sync-to-frame clock commands, GO_CLOCKS, CLEAR_FALLBACK, FORCE_FALLBACK.

14 Connection Control—Standard and Virtual Channel

14.1 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 22).

14.1.1 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.

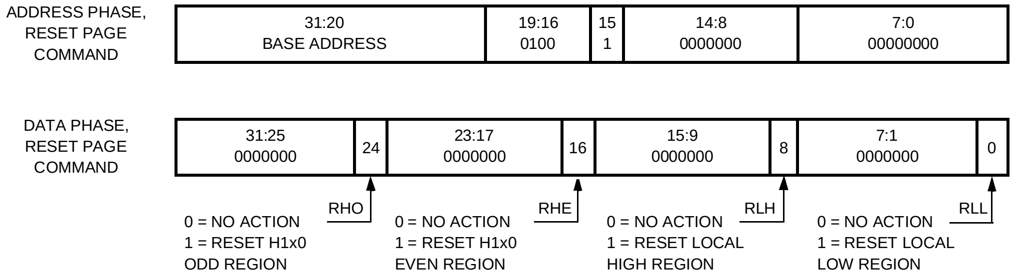
14.1.1.1 PCI Connection Memory Programming

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 H1x0 EVEN streams (CT_D[30, 28, . . . 0]), H1x0 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— 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; refer to Figure 35.
- 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.

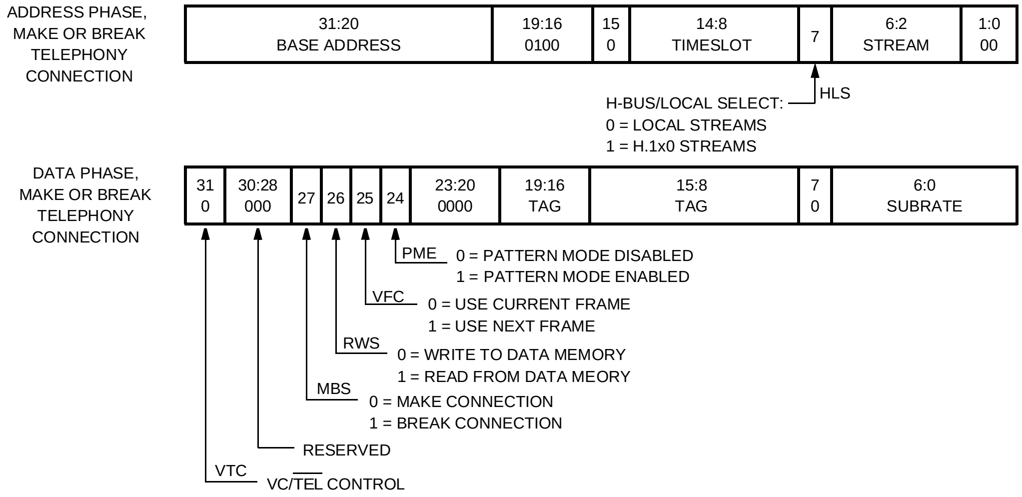
MAKE and BREAK commands above must be presented as PCI memory write commands. QUERY commands are presented as PCI memory read commands.



5-9622 (F)

Figure 35. PCI Programming—Reset Page Command

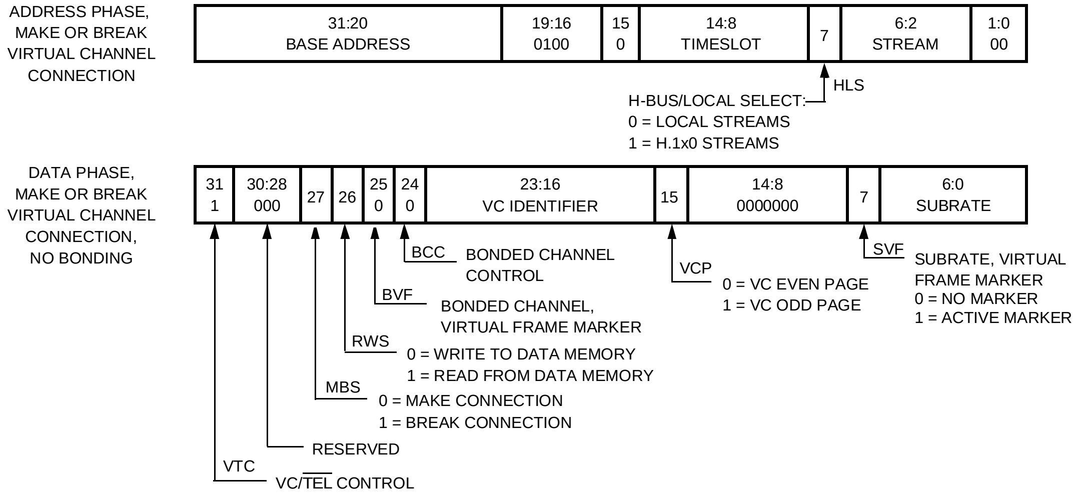
14 Connection Control—Standard and Virtual Channel (continued)



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

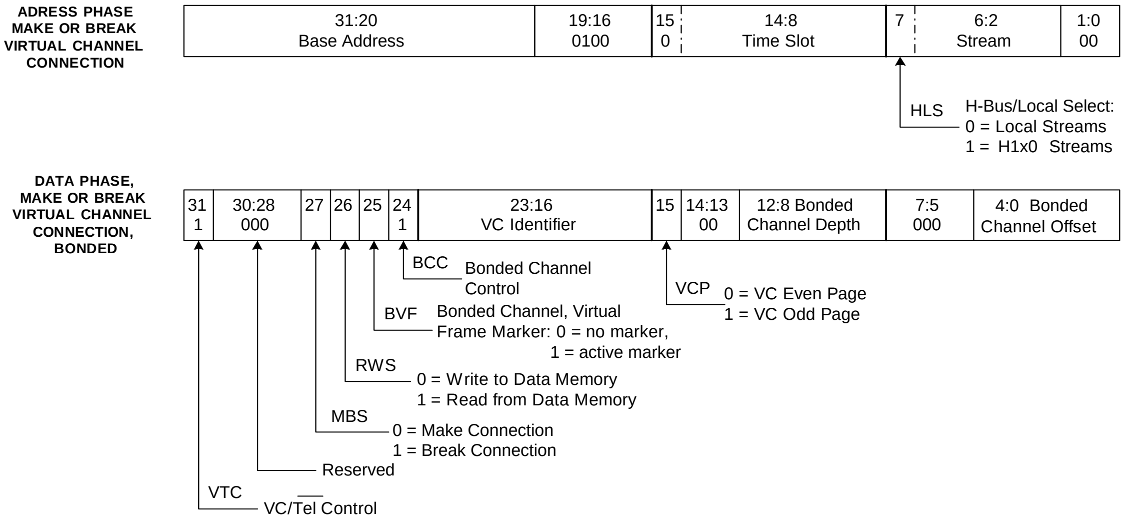


5-9624 (F)

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

14 Connection Control—Standard and Virtual Channel (continued)



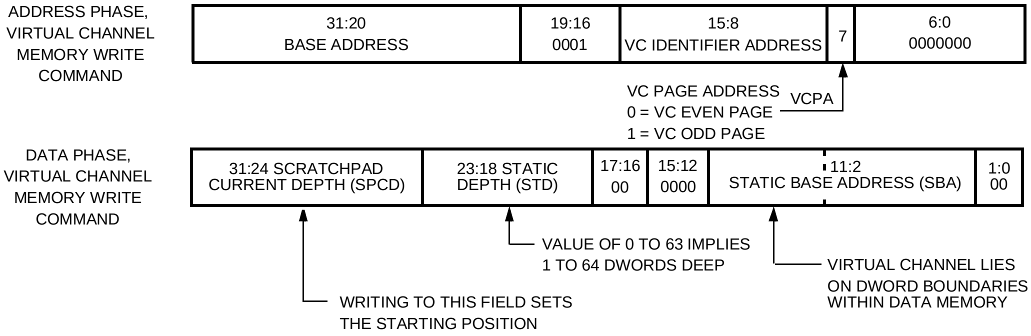
Note: For QUERY (read cycle), a 0 in bit 27 of the data phase indicates an invalid or broken connection. A 1 in bit 27 indicates a valid connection.

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

14.1.1.2 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 are as follows:

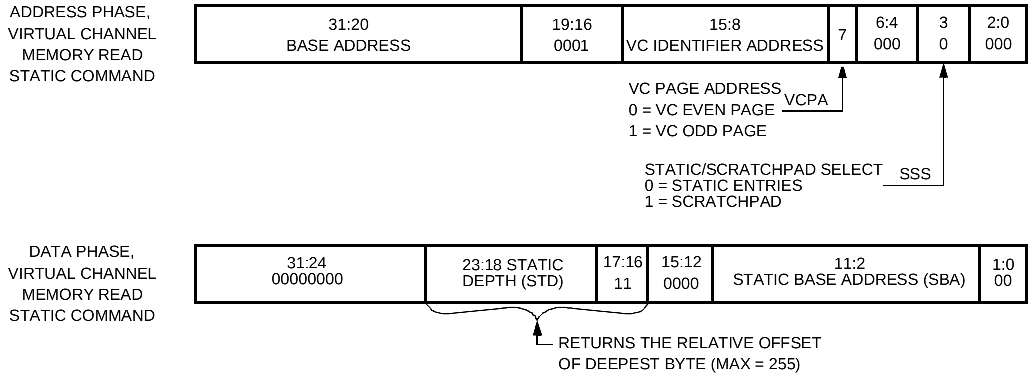
- A WRITE is presented as a PCI memory write command Figure 39 below.
- A READ STATIC is presented as a PCI memory read command (see Figure 40 on page 139).
- A READ SCRATCHPAD is presented as a PCI memory read command (see Figure 41 on page 139).



5-9628 (F)

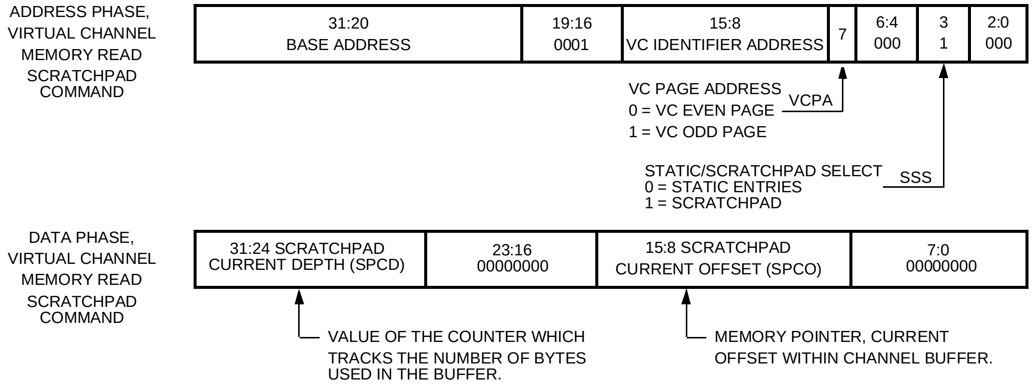
Figure 39. PCI Programming—Write Virtual Channel Command

14 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

14.1.2 Microprocessor Interface

When the T8110 microprocessor 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 microprocessor interface, included for diagnostic purposes only.

14.1.2.1 Microprocessor Connection Memory Programming

Because the microprocessor interface only allows word or byte accesses, multiple write accesses must occur. The microprocessor 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, and then write to the upper byte (or upper word) to actually move data into the connection memory; refer to Table 110.

14 Connection Control—Standard and Virtual Channel (continued)

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 H1x0 even streams (CT_D[30, 28, . . . 0]), H1x0 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 resets any (up to all four) connection memory region (see Figure 42 on page 141). 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 microprocessor WORD writes, or four microprocessor BYTE writes, see Table 110.
- MAKE/BREAK/QUERY, telephony connection (see Figure 43 on page 141).
- MAKE/BREAK/QUERY, virtual channel nonbonded connection* (see Figure 44 on page 142).
- MAKE/BREAK/QUERY, virtual channel bonded connection* (see Figure 45 on page 143).

The MAKE and BREAK commands are presented as multiple microprocessor write cycles. The QUERY command is presented as multiple microprocessor read cycles; refer to Table 110.

* Making virtual channel connections in the connection memory is for diagnostic purpose only when the microprocessor interface is selected.

Table 110. Microprocessor 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.

Note: Data byte n required information is shown in Figure 43—Figure 45.

14 Connection Control—Standard and Virtual Channel (continued)

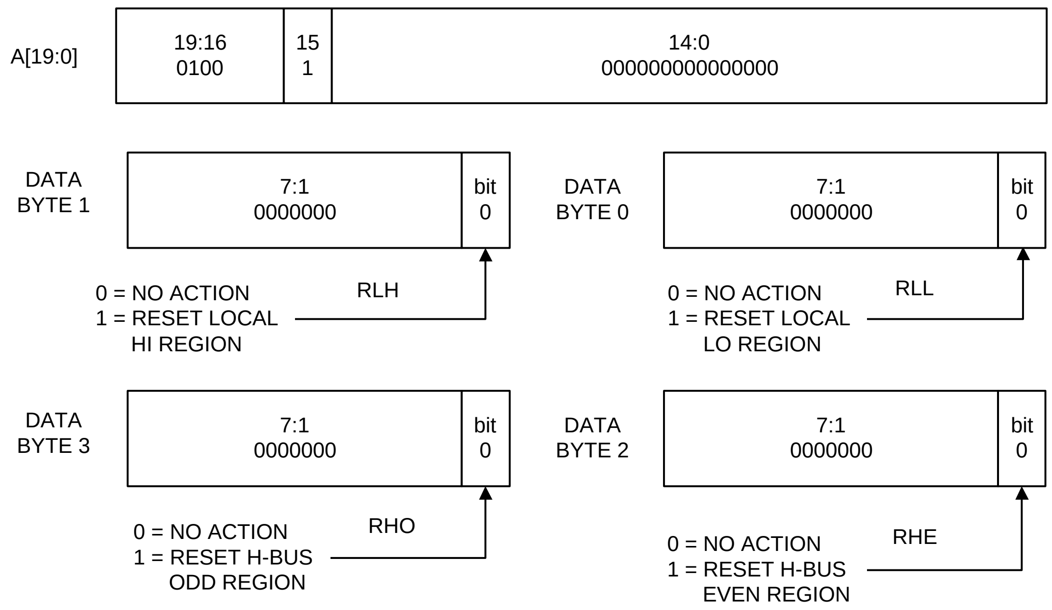


Figure 42. Microprocessor Programming—Reset Page Command

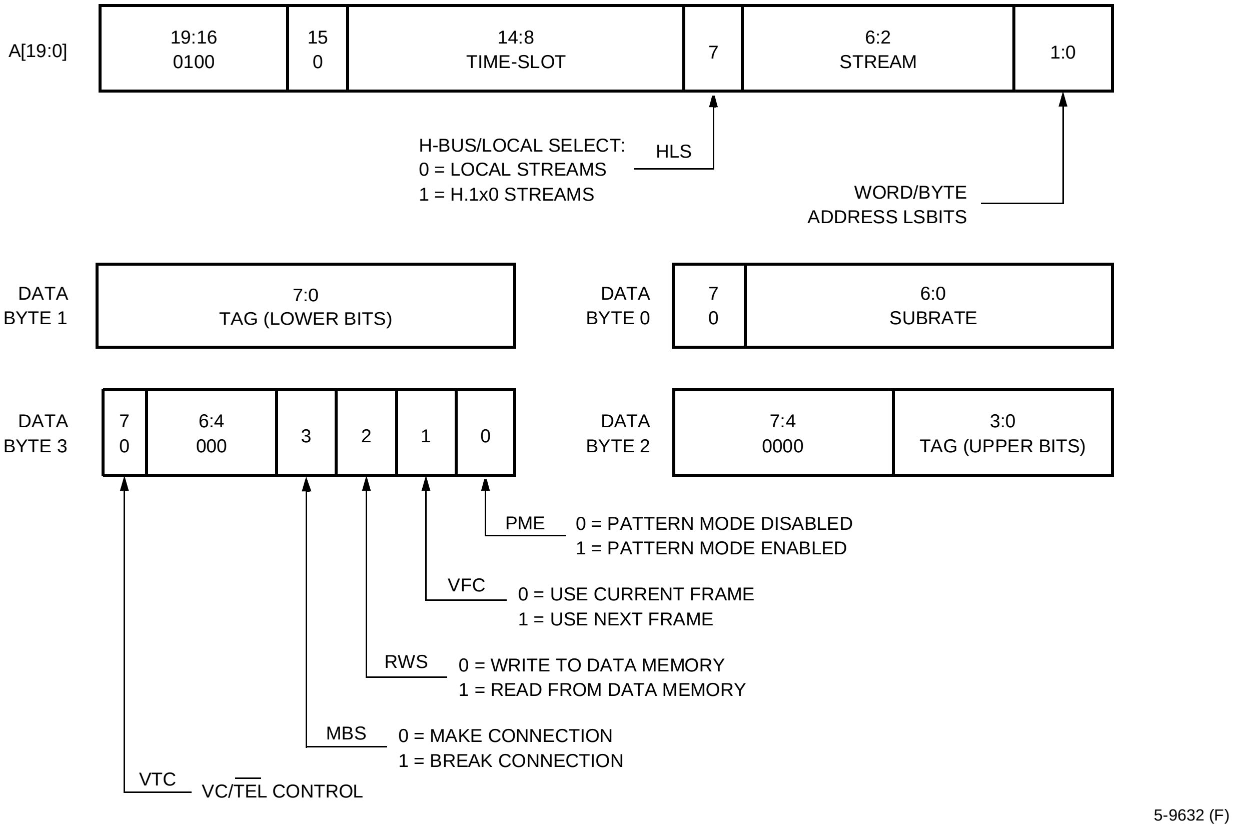


Figure 43. Microprocessor Programming—Make/Break/Query Telephony Connections

14 Connection Control—Standard and Virtual Channel (continued)

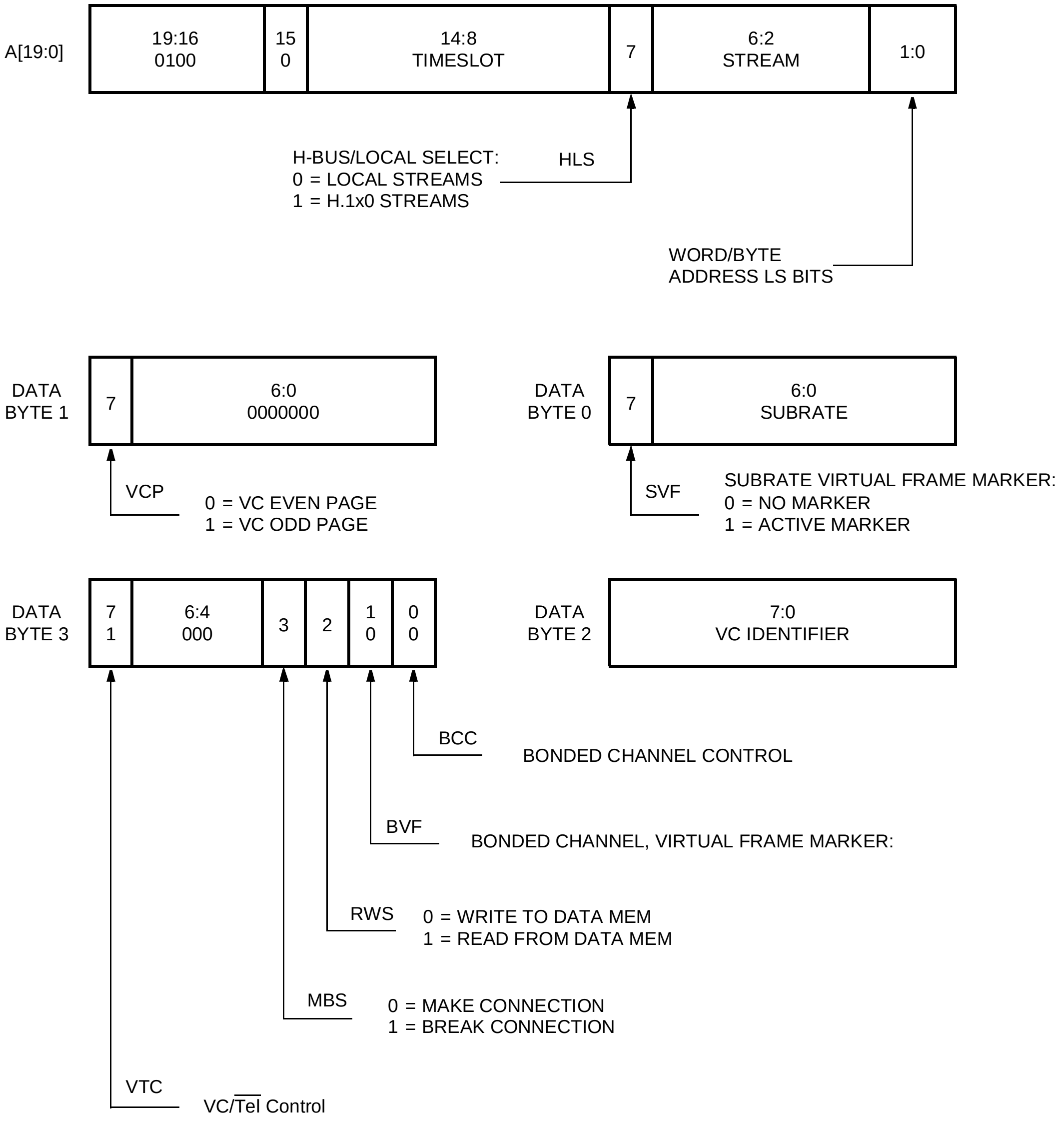
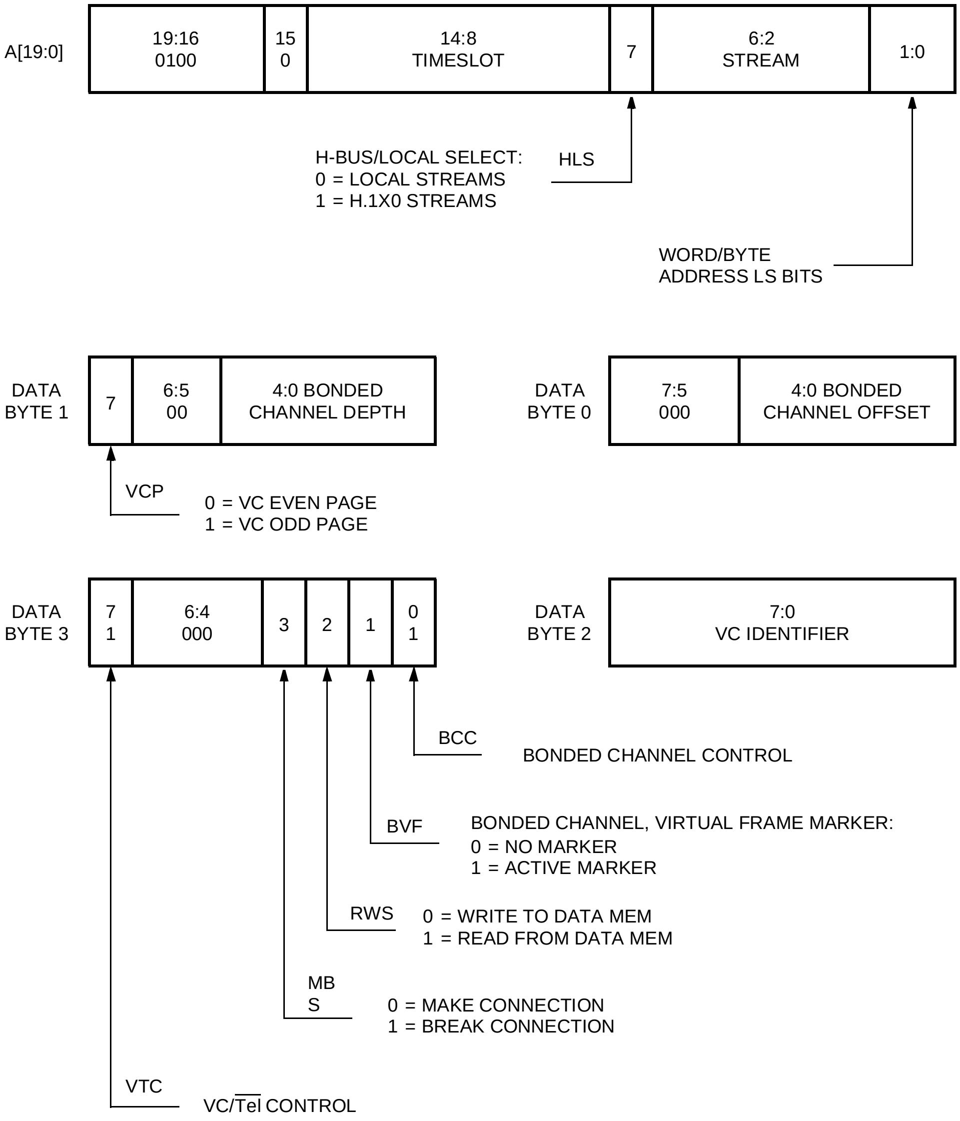


Figure 44. Microprocessor Programming—Make/Break/Query Virtual Channel Nonbonded Connections

14 Connection Control—Standard and Virtual Channel (continued)



1446

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

14 Connection Control—Standard and Virtual Channel (continued)

14.1.2.2 Microprocessor Virtual Channel Memory Programming

Because the microprocessor interface only allows word or byte accesses, multiple write accesses must occur. The microprocessor 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 111.

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 are as follows:

- The WRITE command is presented as a microprocessor write cycle (see Figure 46 on page 145).
- The READ STATIC command is presented as a microprocessor read cycle (see Figure 47 on page 145).
- The READ SCRATCHPAD command is presented as a microprocessor read cycle (see Figure 48 on page 146).

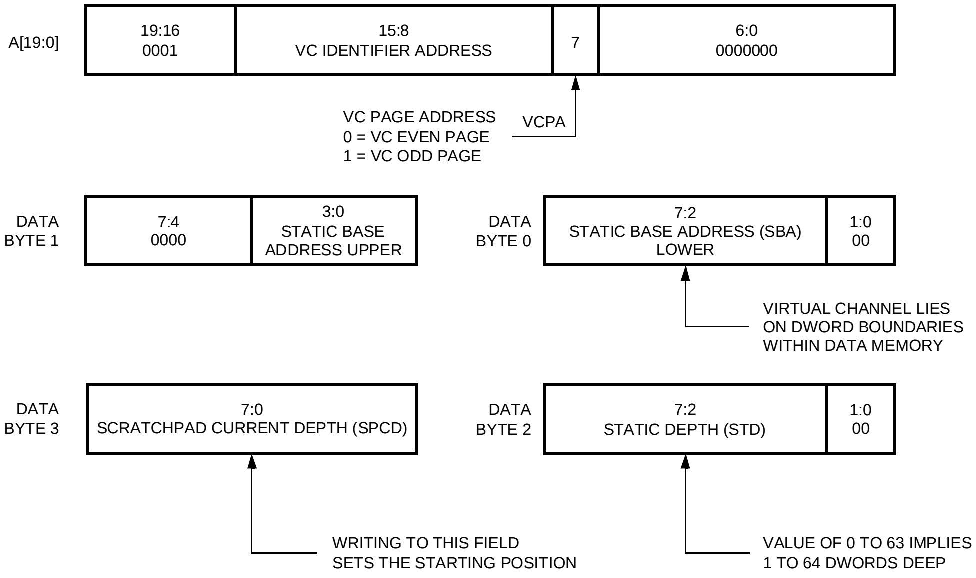
Note: Accessing the virtual channel memory is for diagnostic purpose only when the microprocessor interface is selected.

Table 111. 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.

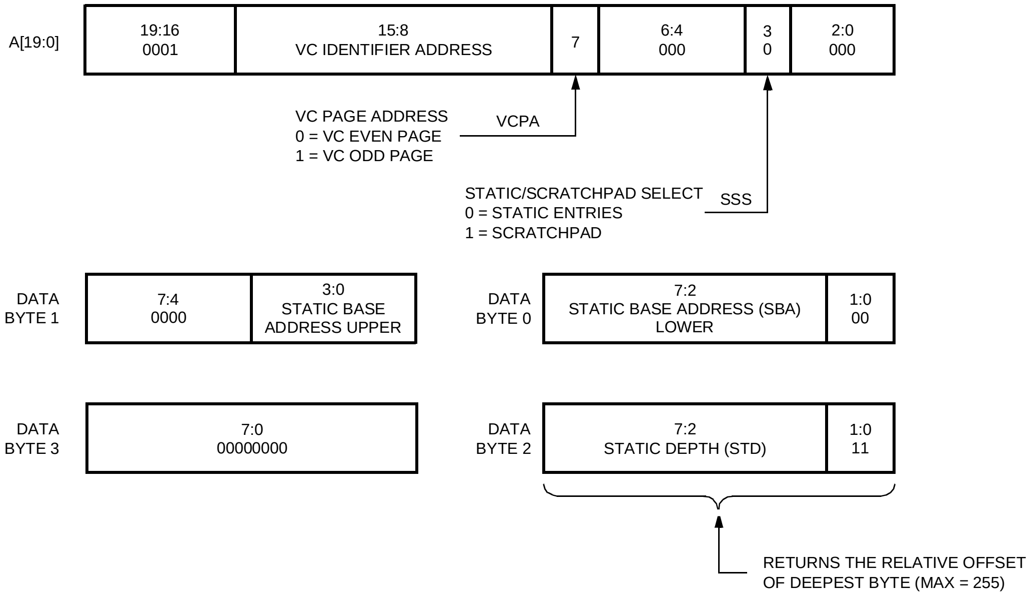
Note: Data byte n required information is shown in Figure 46—Figure 48.

14 Connection Control—Standard and Virtual Channel (continued)



5-9635 (F)

Figure 46. Microprocessor Programming—Write Virtual Channel Memory Command



5-9636 (F)

Figure 47. Microprocessor Programming—Read Virtual Channel Static Command

14 Connection Control—Standard and Virtual Channel (continued)

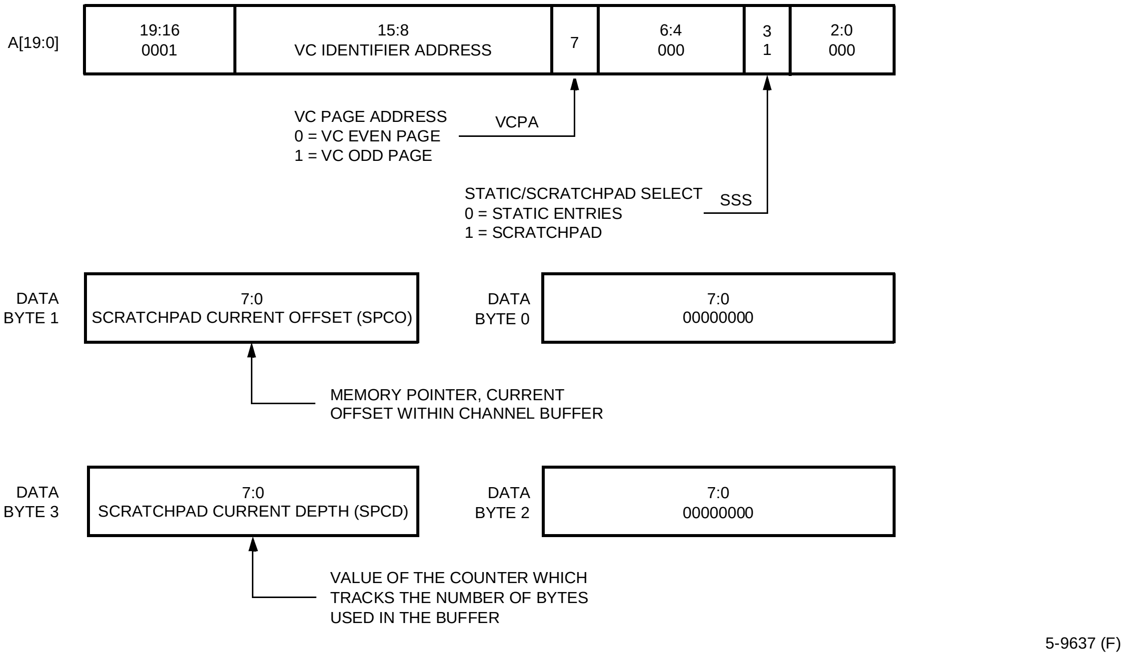


Figure 48. Microprocessor Programming—Read Virtual Channel Scratchpad Command

14.2 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.

14.2.1 Memory Architecture and Configuration

14.2.1.1 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 H1x0 streams (CT_D[31:0]) and 32 local streams (L_D[31:0]). Connection memory is physically addressed by time slot (7 bits), H1x0/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 as follows:

- H1x0 even streams (CT_D[30, 28, . . . 0])
- H1x0 odd streams (CT_D[31, 29, . . . 1])
- Local high streams (L_D[31:16])
- Local low streams (L_D[15:0])

14 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 136 and Figure 42 on page 141).

For standard telephony switching connections, the connection memory locations contain one-half simplex switch control information (refer to Figure 36 on page 137 and Figure 43 on page 141), as follows:

- 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 memory) or **to** (from DATA memory 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 14.2.1.2 on page 148 and Section 14.2.2.1 on page 149.

- PME indicates a pattern mode connection.
- TAG is 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 is 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).

14.2.1.1.1 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 memory) or **to** (from DATA memory to serial stream).
- BVF is bonded virtual frame marker (unused for nonbonded channels).
- BCC is bonded channel control indicator; 0 denotes a nonbonded channel.
- VC identifier and VCP indicates which virtual channel this information is for (0 up to 511 virtual channels).
- SVF (subrate virtual frame marker) is an indicator for the last piece of a packed subrate byte.
- SUBRATE information is subrate switching control (bitswap); refer to Section 14.2.2.3.

14.2.1.1.2 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 memory) or **to** (from data memory to serial stream).
- BVF (bonded virtual frame marker) is an indicator for the last byte switched in a frame.

14 Connection Control—Standard and Virtual Channel (continued)

- BCC (bonded channel control indicator), a 1 denotes a bonded channel.
- VC identifier and VCP indicates 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 is a specific data memory pointer offset for the data byte related to this connection.

14.2.1.2 Data Memory

The T8110 data memory is 4096 bytes, which can be programmatically configured in six ways, via the data memory mode select register (0x00105; refer to Section 6.1.3 on page 48). 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.

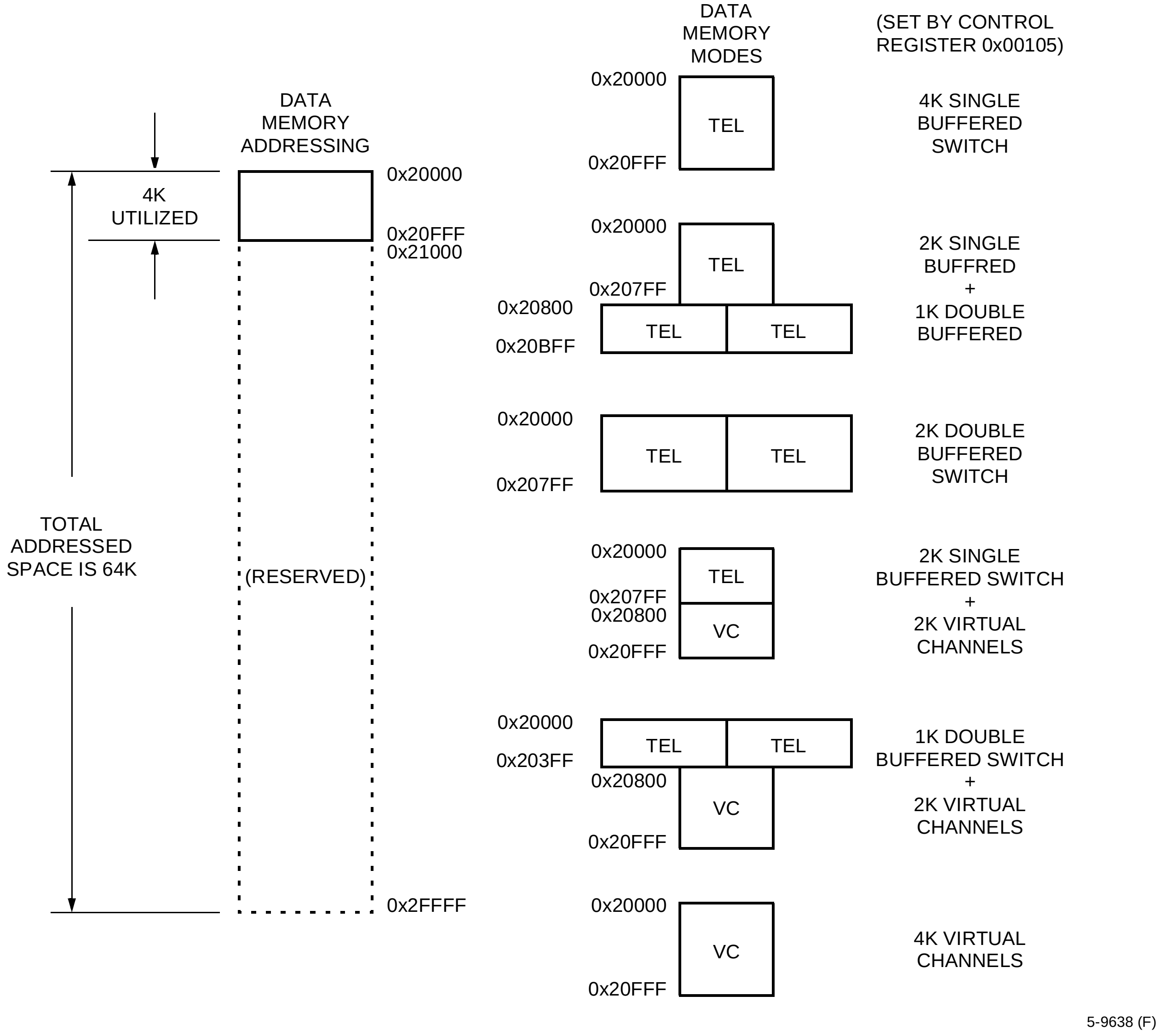


Figure 49. T8110 Data Memory Map and Configurations

14 Connection Control—Standard and Virtual Channel (continued)

14.2.1.3 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) is the physical start address for this particular virtual channel in the data memory VC space.
- STD (static depth) is 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 as follows:

- SPCD (scratchpad current depth) is the current number of bytes transferred to/from serial streams.
- SPCO (scratchpad current offset) is the data memory address pointer.

14.2.2 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 137, Figure 43 on page 141, and Section 14.2.1.1 on page 146).

14.2.2.1 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.

Please see Appendix A on page 190 for more details on constant and minimum delay connections.

14.2.2.2 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.

14.2.2.3 Subrate

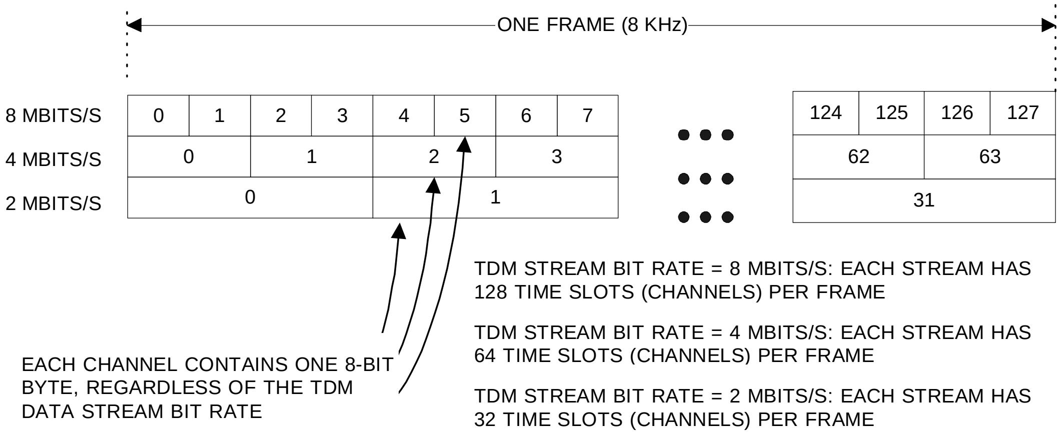
The subrate control bit field in connection memory is used only by **from** connections and controls how individual bits or groups of bits of an incoming serial byte are shuffled prior to writing them to the data memory, in order to achieve subrate switching.

14 Connection Control—Standard and Virtual Channel (continued)

14.2.2.3.1 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 113.



EACH CHANNEL CONTAINS ONE 8-BIT BYTE, REGARDLESS OF THE TDM DATA STREAM BIT RATE

TDM STREAM BIT RATE = 8 MBITS/S: EACH STREAM HAS 128 TIME SLOTS (CHANNELS) PER FRAME

TDM STREAM BIT RATE = 4 MBITS/S: EACH STREAM HAS 64 TIME SLOTS (CHANNELS) PER FRAME

TDM STREAM BIT RATE = 2 MBITS/S: EACH STREAM HAS 32 TIME SLOTS (CHANNELS) PER FRAME

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 112 and Table 113.

Table 112. 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.

14 Connection Control—Standard and Virtual Channel (continued)

Table 113. Subrate Switching, Data Propagation Rate vs. Channel Capacity

Subrate Type	Bits per Channel	Channel Data Propagation Rate (Bits/Frame x 8K Frames/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

14.2.2.3.2 Subrate Switching Using T8110

The H1x0 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 14.2.2.3.3 and 14.2.2.3.4). 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 the following:

- Overall subrate enable mode is activated (register 0x00105, data memory mode select bit 7 is set; see Section 6.1.3 on page 48).
- 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 on page 137, Figure 43 on page 141, and Table 114).
- 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 14.2.1.2, 14.2.2.1).

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 114. Subrate Switching, Connection Memory Programming Setup

<table>
<tr><th rowspan="2">Subrate Type</th><th colspan="7">Subrate Connection Memory Bit Field (6:0)</th></tr>
<tr><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td rowspan="3">Bit</td><td>1</td><td colspan="3">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</td><td colspan="3">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</td></tr>
<tr><th colspan="7">Subrate Connection Memory Bit Field (6:0)</th></tr>
<tr><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>Di-Bit</td><td colspan="2">01</td><td colspan="2">00 = from bits[1:0]
01 = from bits[3:2]
10 = from bits [5:4]
11 = from bits[7:6]</td><td>Reserved</td><td colspan="2">00 = to bits[1:0]
01 = to bits[3:2]
10 = to bits[5:4]
11 = to bits[7:6]</td></tr>
</table>

14 Connection Control—Standard and Virtual Channel (continued)

Table 114. 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			

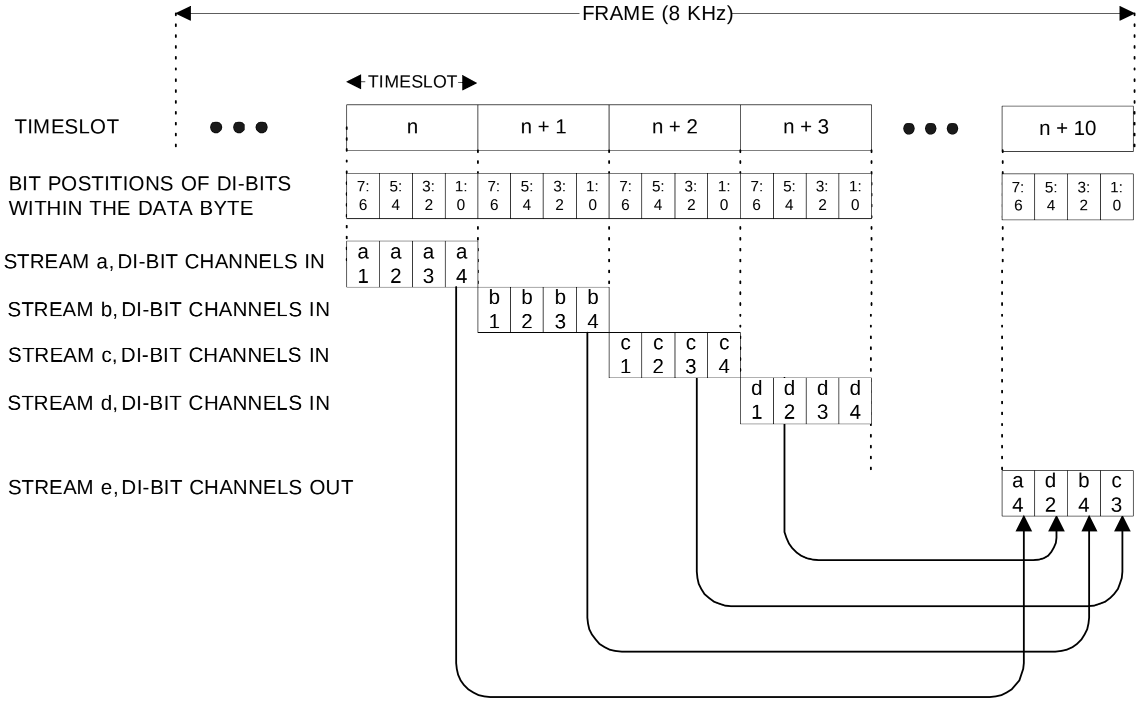
14.2.2.3.3 Subrate Packing of Outgoing Bytes

The output from subrate connections is always an entire byte so that it does not violate the H.100 or H.110 specifications. The output byte is composed of smaller, i.e., subrate pieces. The process of combining the incoming pieces into a whole byte suitable for output is called **packing**. In the T8110 (and other subrate-capable *Ambassador* devices), **packing** is accomplished by making several **from** connections for each single **to** connection. For example, in Figure 48, four **from** connections (of different stream/time-slot origins), all di-bits, are used to construct a byte that will be output as defined by the **to** connection.

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 48. This example shows the packing of four separate incoming di-bits from four different channels into one outgoing byte on one channel.

Note: Please 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 Section 14.2.2.3.4 on page 153.

14 Connection Control—Standard and Virtual Channel (continued)



Notes:
Connectivity is as follows:

- **From** stream a, time slot n, bits[1:0] **to** stream e, time slot n + 10, bits[7:6].
- **From** stream b, time slot n + 1, bits[1:0] **to** stream e, time slot n + 10, bits[3:2].
- **From** stream c, time slot n + 2, bits[3:2] **to** stream e, time slot n + 10, bits[1:0].
- **From** stream d, time slot n + 3, bits[5:4] **to** stream e, time slot n + 10, bits[5:4].

Required connection memory programming is as follows:

Five 1/2 simplex connections are required to pack four incoming di-bits into an outgoing byte.

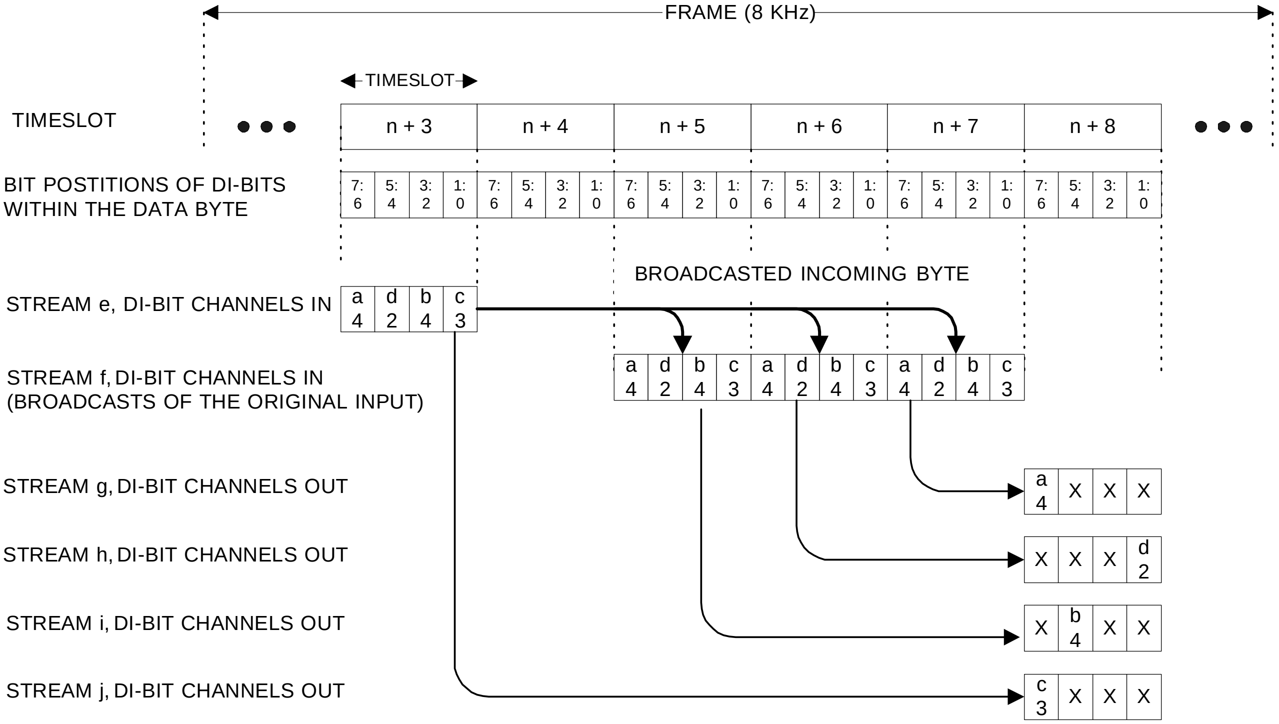
- **From** stream a, time slot n. Connection memory subrate field = 0100X11.
- **From** stream b, time slot n + 1. Connection memory subrate field = 0100X01.
- **From** stream c, time slot n + 2. Connection memory subrate field = 0101X00.
- **From** stream d, time slot n + 3. Connection memory subrate field = 0110X10.
- **To** stream e, time slot n + 10. Connection memory subrate field is don’t care.

Figure 51. Subrate Switching Example, Byte Packing

14.2.2.3.4 Subrate Unpacking of Incoming Bytes

Because the H1x0 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 subrate 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.

14 Connection Control—Standard and Virtual Channel (continued)



Notes:

Connectivity is as follows:

From stream e, time slot n + 3, bits[1:0] **to** stream j, time slot n + 8, bits[7:6].

From stream f, time slot n + 5, bits[3:2] **to** stream i, time slot n + 8, bits[5:4].

From stream f, time slot n + 6, bits[5:4] **to** stream h, time slot n + 8, bits[1:0].

From stream f, time slot n + 7, bits[7:6] **to** stream g, time slot n + 8, bits[7:6].

Required connection memory programming is as follows:

Eight 1/2 simplex connections are required to unpack one incoming byte to four separate outgoing di-bits.

From stream e, time slot n + 3. Connection memory subrate field = 0100X11.

From stream f, time slot n + 5. Connection memory subrate field = 0101X10.

From stream f, time slot n + 6. Connection memory subrate field = 0110X00.

From stream f, time slot n + 7. Connection memory subrate field = 0111X11.

To stream g, time slot n + 8. Connection memory subrate field is don't care.

To stream h, time slot n + 8. Connection memory subrate field is don't care.

To stream i, time slot n + 8. Connection memory subrate field is don't care.

To stream j, time slot n + 8. Connection memory subrate field is don't care.

Figure 52. Subrate Switching Example, Byte Unpacking

14 Connection Control—Standard and Virtual Channel (continued)

14.2.3 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 two parts:

- A one-half simplex connection **to** (or **from**) a channel (or channels) in the H1x0 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 14.2.1.1, 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 14.2.3.1—14.2.3.3 for more details.
- 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 14.2.3.4). 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 14.2.1.2, 14.2.1.3, 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 14.2.3.4.
- **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 14.2.3.4.

14.2.3.1 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.

14 Connection Control—Standard and Virtual Channel (continued)

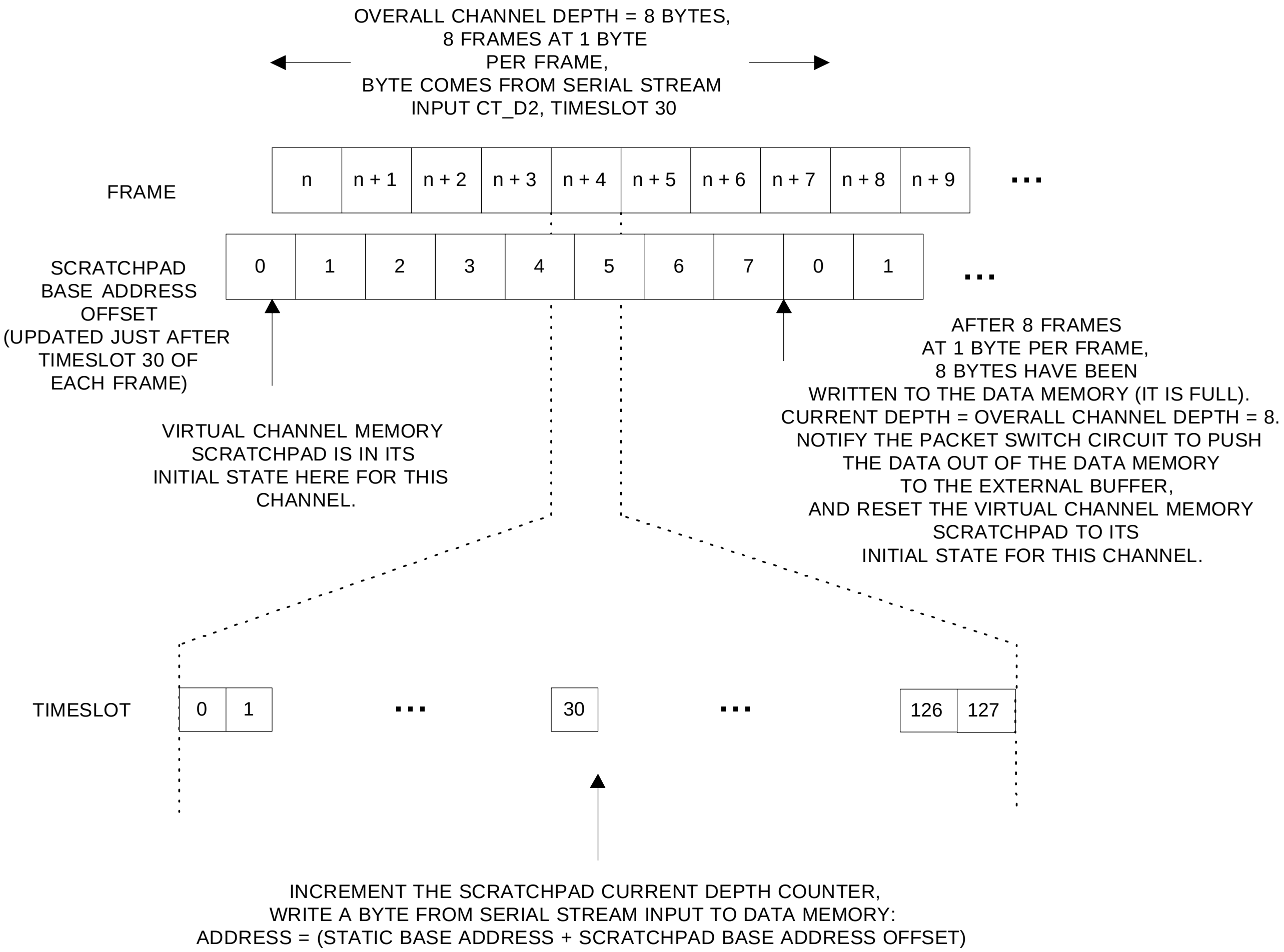


Figure 53. Nonbonded Virtual Channel in the PUSH Direction

14 Connection Control—Standard and Virtual Channel (continued)

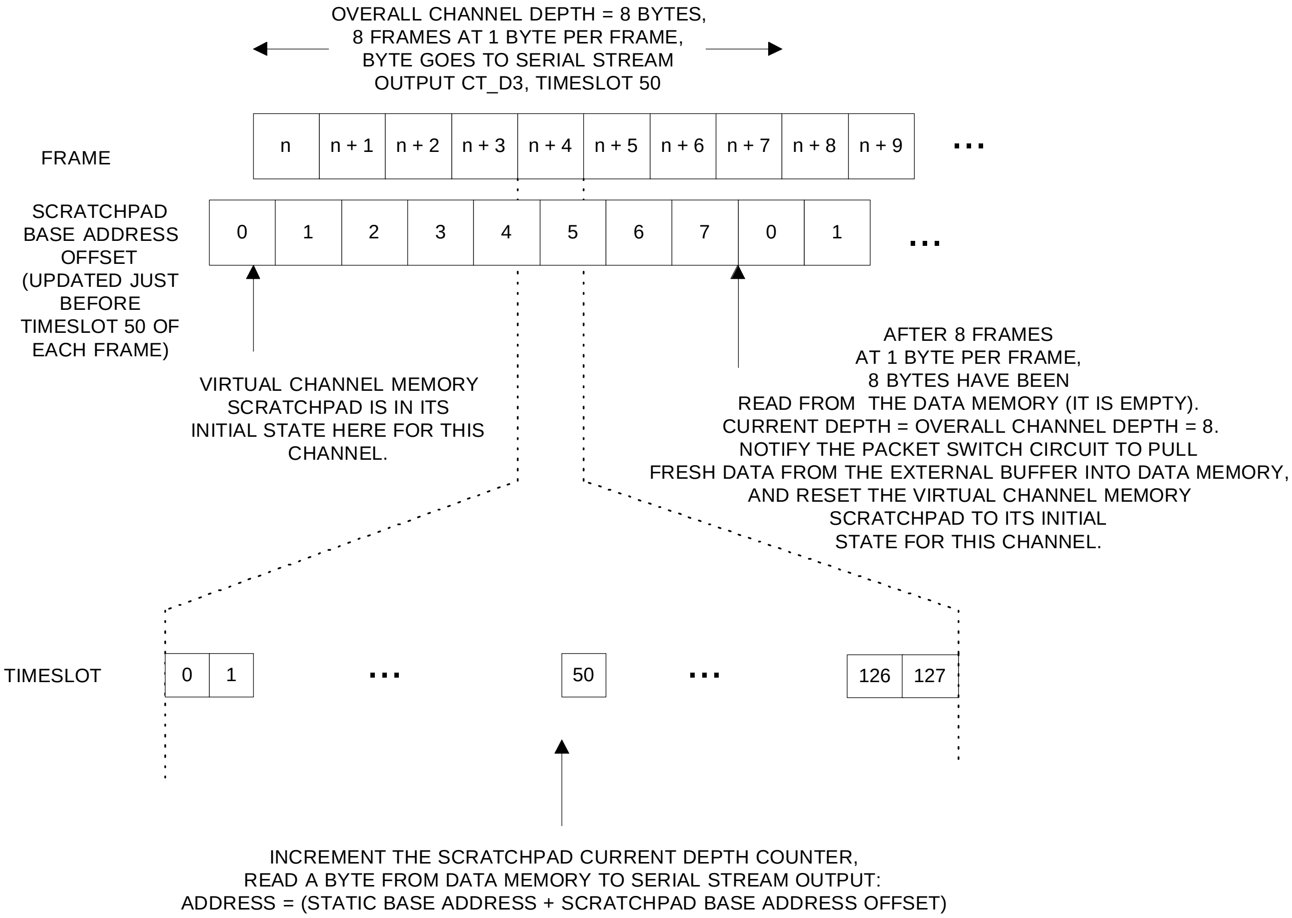


Figure 54. Nonbonded Virtual Channel in the PULL Direction

14.2.3.2 Subrate

Nonbonded virtual channels in the PUSH direction allow for subrate switching and the packing of incoming subrate pieces into bytes. (Refer to Section 14.2.2.3 for a general description of subrate. For the PULL direction, all bits of 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 114), one additional control bit, SVF subrate virtual frame marker (refer to Section 14.2.1.1), is required as a means to denote the completion of a byte packing operation. Please refer to Figure 55.

14 Connection Control—Standard and Virtual Channel (continued)

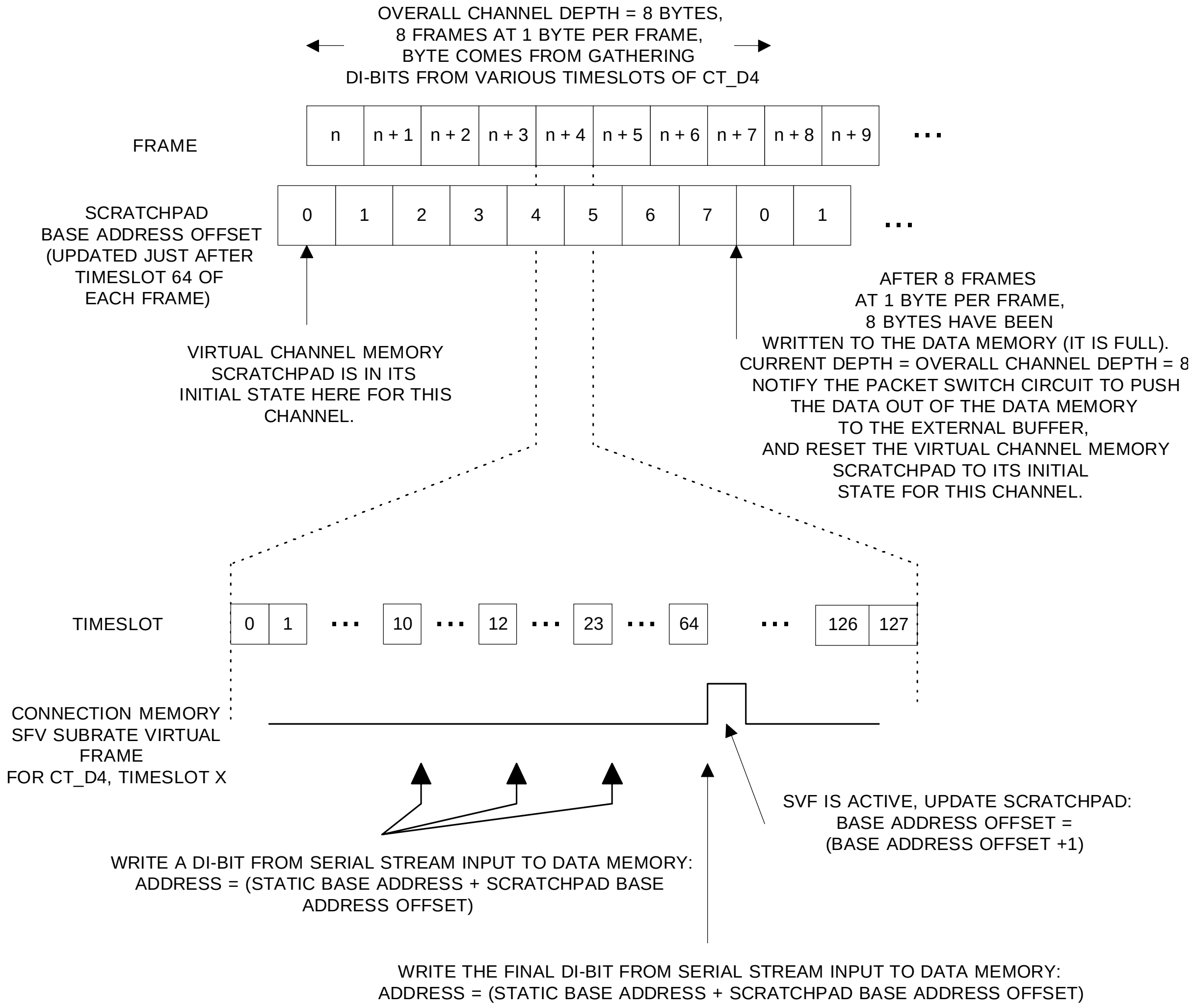
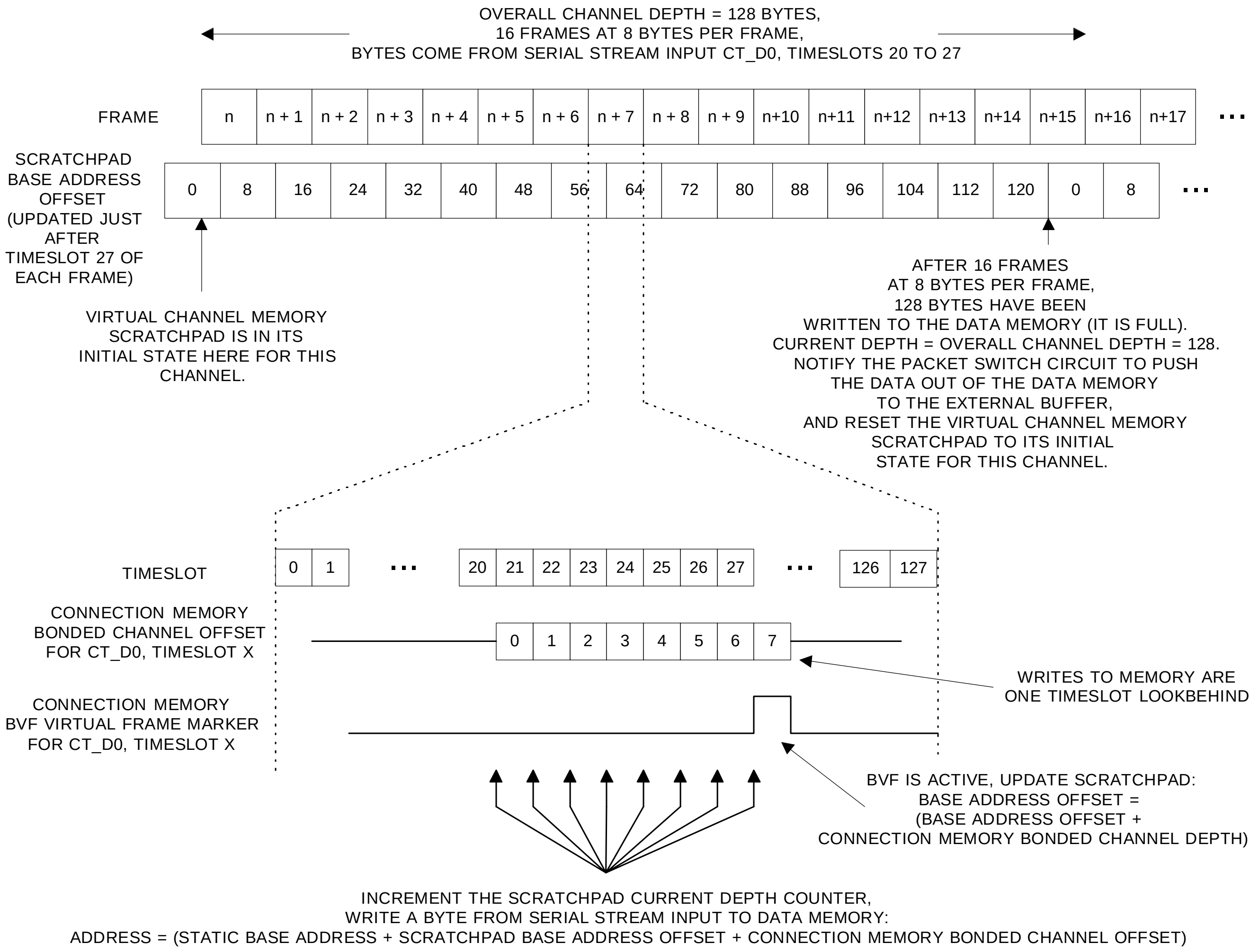


Figure 55. Nonbonded Virtual Channel with Subrate and Packed Bytes

14.2.3.3 Bonded Channels

A bonded virtual channel means that more than 1 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 8 bytes per frame, and the data memory depth is set to 128 bytes, then the data memory for that channel can store 16 TDM frames worth of data at a time. An additional control bit, BVF (bonded virtual frame marker; refer to Section 14.2.1.1), 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.

14 Connection Control—Standard and Virtual Channel (continued)



OVERALL CHANNEL DEPTH = 128 BYTES,
16 FRAMES AT 8 BYTES PER FRAME,
BYTES COME FROM SERIAL STREAM INPUT CT_D0, TIMESLOTS 20 TO 27

VIRTUAL CHANNEL MEMORY SCRATCHPAD IS IN ITS INITIAL STATE HERE FOR THIS CHANNEL.

AFTER 16 FRAMES
AT 8 BYTES PER FRAME,
128 BYTES HAVE BEEN
WRITTEN TO THE DATA MEMORY (IT IS FULL).
CURRENT DEPTH = OVERALL CHANNEL DEPTH = 128.
NOTIFY THE PACKET SWITCH CIRCUIT TO PUSH
THE DATA OUT OF THE DATA MEMORY
TO THE EXTERNAL BUFFER,
AND RESET THE VIRTUAL CHANNEL MEMORY
SCRATCHPAD TO ITS INITIAL
STATE FOR THIS CHANNEL.

WRITES TO MEMORY ARE ONE TIMESLOT LOOKBEHIND

BVF IS ACTIVE, UPDATE SCRATCHPAD:
BASE ADDRESS OFFSET =
(BASE ADDRESS OFFSET +
CONNECTION MEMORY BONDED CHANNEL DEPTH)

INCREMENT THE SCRATCHPAD CURRENT DEPTH COUNTER,
WRITE A BYTE FROM SERIAL STREAM INPUT TO DATA MEMORY:
ADDRESS = (STATIC BASE ADDRESS + SCRATCHPAD BASE ADDRESS OFFSET + CONNECTION MEMORY BONDED CHANNEL OFFSET)

Figure 56. Bonded Virtual Channel in the PUSH Direction

14 Connection Control—Standard and Virtual Channel (continued)

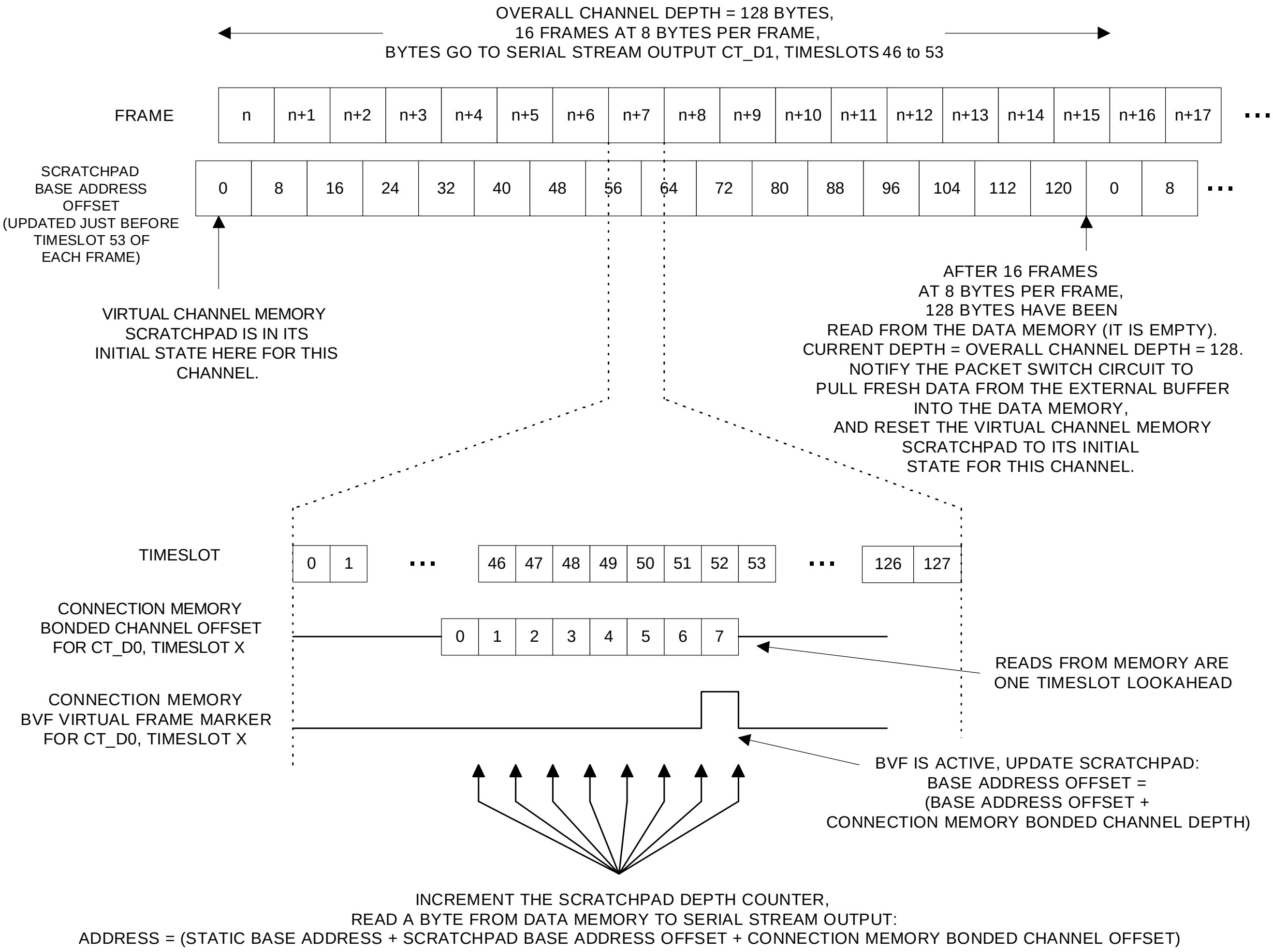


Figure 57. Bonded Virtual Channel in the PULL Direction

14.2.3.4 External Buffer Access

14.2.3.4.1 Overview

The T8110’s general operation is to gather TDM serial stream data in small internal buffers, which are programmable on a per-channel 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).

14 Connection Control—Standard and Virtual Channel (continued)

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.

14.2.3.4.2 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; see Section 4.2.2 on page 32 and to Figure 13 on page 33. 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 115. Descriptor Table

<table>
<tr><th>DWORD</th><th colspan="5">Bits[31:24]</th><th>Bits[23:16]</th><th colspan="2">Bits[15:8]</th><th>Bits[7:0]</th></tr>
<tr><td>First DWORD</td><td colspan="8">External buffer base address</td><td>External buffer last address</td></tr>
<tr><td>Second DWORD</td><td>SE</td><td>OE</td><td>L</td><td>UF</td><td colspan="2">UOR[11:0]</td><td>GBS [2:0]</td><td>TF</td><td>TOR[11:0]</td></tr>
</table>

- External buffer base address. This is 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. This is 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. This determines whether the external buffer is treated as circular for this virtual channel (refer to Section 14.2.3.4.4). 0 = buffer is circular (T8110 can roll over at end-of-buffer), 1 = the buffer is not circular (T8110 must stop at end of buffer).
  — OE, overwrite enable. This determines whether the T8110 can overwrite unread data in the external buffer for this virtual channel (refer to Section 14.2.3.4.4). 0 = overwrite is disabled, 1 = overwrite is enabled.
  — L, lock. This 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. This is a **USER**-updated offset pointer within the defined 4K external buffer space for this virtual channel. Read-only for the T8110.
- GBS (status). This is supplemental information on the status of the external buffer for this virtual channel (refer to Section 14.2.3.4.4). These flags are read-write for both the T8110 and the **USER**. Refer to Table 116.
- 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**.

14 Connection Control—Standard and Virtual Channel (continued)

Table 116. Descriptor Table GBS Status Descriptions

GBS Value	Status Description
000	**USER** has initialized 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.

14.2.3.4.3 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 14.2.3.4.4). 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.

14.2.3.4.4 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.

14.2.3.4.4.1 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 are 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 116.

14 Connection Control—Standard and Virtual Channel (continued)

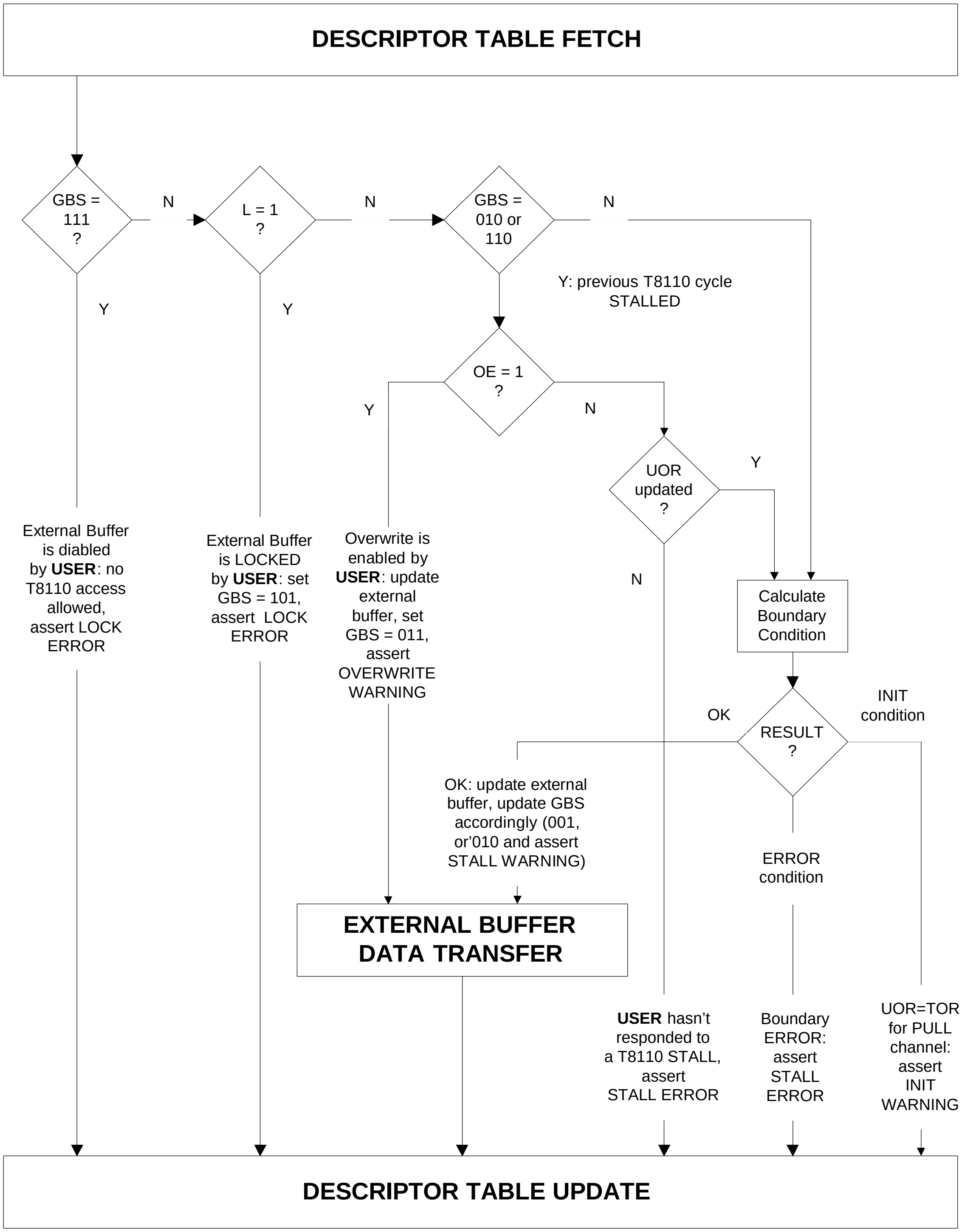


Figure 58. Descriptor Table Fetch Decode

14 Connection Control—Standard and Virtual Channel (continued)

14.2.3.4.5 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)
- 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).

14.2.3.4.6 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.

14.2.3.5 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 14.2.1.3). 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 14.2.3.4.4). Upon completion of that transfer protocol, the notify pending memory bit for that channel is reset.

14 Connection Control—Standard and Virtual Channel (continued)

14.2.3.5.1 System Errors Due to Packet Switching

The system error indicators for packet switch activity are located in the system error register, 0x00126. These indicators are inputs to the interrupt controller, and are maskable interrupts. There are eight indicators, as shown below.

Table 117. System Error Register Address 0x00126

BIT	Mnemonic	Description
7	PMFOB	PCI master, fatal error—this bit is set when a T8110-initiated PCI cycle results in abnormal termination, including the following: ■ Requested PCI target does not respond (master abort). ■ Requested PCI target terminates (target abort). ■ Requested PCI target responds with PCI_DEVSEL#, but does not follow with PCI_TRDY# or PCI_STOP# to allow the cycle to complete (PCI_TRDY# time-out, see Table 12 on page 34). ■ Requested PCI target retries beyond the retry count (RETRY time-out, see Table 12 on page 34).
6	PMLOB	PCI master, external buffer LOCK error—this bit is set when a T8110 descriptor table fetch indicates the **USER** has locked the external buffer, and T8110 access to the external buffer is denied.
5	PMEOB	PCI master, external buffer STALL error—this bit is set when a T8110 descriptor table fetch indicates that there is a pointer boundary condition (end-of-buffer, or T8110 pointer has caught up to **USER** pointer), and T8110 access to the external buffer is denied.
4	PMWOB	PCI master, external buffer STALL error—this bit is set when a T8110 descriptor table fetch indicates that there is only one external buffer access left before a pointer boundary condition (end-of-buffer, or T8110 pointer has caught up to **USER** pointer). T8110 access to the external buffer is allowed.
3	PMOOB	PCI master, external buffer OVERWRITE warning—this bit is set when a T8110 descriptor table fetch indicates the **USER** has enabled the T8110 to overwrite unread data in the external buffer (i.e., override a boundary condition). This is applicable for **push** operation only.
2	PMIOB	PCI master, external buffer INITIAL warning—this bit is set when a T8110 descriptor table fetch indicates the **USER** has not written anything into the external buffer (the initial state of a **pull** operation).
1	VCOOB	Virtual channel memory, scratchpad OVERFLOW warning—indicates the calculated scratchpad current depth has exceeded the overall buffer depth (VC programming error).
0	NQOOB	NOTIFY_QUEUE OVERFLOW warning—indicates that a request to **push** (or **pull**) a packet of data did not occur within one frame time (125 us), and T8110's internal data buffer will get overwritten (**push**) or contain stale data (**pull**).

15 Electrical Characteristics

15.1 Absolute Maximum Ratings

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

Table 118. Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Unit
Supply Voltage	V_{DD}	—	4.2	V
XTAL1_IN, XTAL2_IN, XTAL1_OUT, XTAL2_OUT pins	—	V_{SS}	V_{DD}	V
Voltage Applied to I/O Pins	—	V_{SS} – 0.3	V_{DD} 5.5	V
Operating Temperature	—	–40	85	°C
Storage Temperature	T_{stg}	–55	125	°C

15.1.1 Handling Precautions

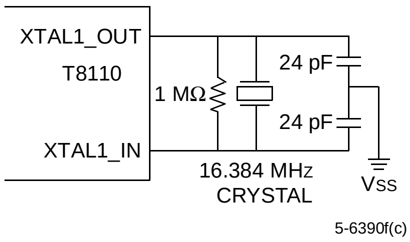
Although protection circuitry has been designed into this device, proper precautions should be taken to avoid exposure to electrostatic discharge (ESD) during handling and mounting. Agere employs a human-body model (HBM) and a charged-device model (CDM) for ESD-susceptibility testing and protection design evaluation. ESD voltage thresholds are dependent on the circuit parameters used to define the model. No industry-wide standard has been adopted for CDM. However, a standard HBM (resistance = 1500 W, capacitance = 100 pF) is widely used and, therefore, can be used for comparison purposes. The T8110 has a HBM ESD threshold voltage rating of 1500 V minimum.

15.2 Crystal Specifications

15.2.1 XTAL1 Crystal

The T8110 requires a 16.384 MHz clock source derived from an oscillator or a crystal. If a crystal is used it has to be a 16.384 MHz crystal and must be connected between the XTAL1_IN and the XTAL1_OUT pins. External 24 pF, 5% capacitors must be connected from XTAL1_IN and XTAL1_OUT to Vss, as shown in the diagram below.
The ±32 ppm tolerance is the suggested value if the oscillator is used as the clocking source while mastering the bus. Otherwise, a crystal with a lesser tolerance can be used. The crystal specifications are shown below.

Table 119. XTAL1 Specifications

Parameter	Value	
Frequency	16.384 MHz	
Oscillation Mode	Fundamental, parallel resonance	
Effective Series Resistance	50 Ω maximum	
Load Capacitance	18 pF	
Shunt Capacitance	7 pF maximum	
Frequency Tolerance and Stability	32 ppm	

15 Electrical Characteristics (continued)

If an oscillator is used (see Section 7.4.4 on page 79), the signal has to be connected to the XTAL1_IN pin. XTAL1_OUT must be left unconnected in this configuration. XTAL1_IN and XTAL1_OUT are not 5 V tolerant. The oscillator must meet the requirements shown below.

Table 120. 16.384 MHz Oscillator Requirements

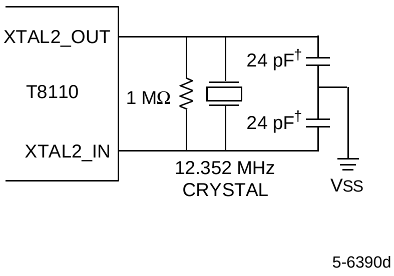
Parameter	Value	
Frequency	16.384 MHz	
Maximum Rise or Fall Time	10 ns, 10%—90% V_{DD}	
Minimum Pulse Width	Low	High
	20 ns	20 ns

15.2.2 XTAL2 Crystal

XTAL2 is an optional crystal oscillator input. If a crystal is used, it has to be a 6.176 MHz or a 12.352 MHz crystal and must be connected between the XTAL2_IN and the XTAL2_OUT pins, as shown in the diagram below. External 24 pF, 5% capacitors must be connected from XTAL2_IN and XTAL2_OUT to Vss. The ±32 ppm tolerance is the suggested value if the oscillator is used as the clocking source while mastering the bus. Otherwise, a crystal with a lesser tolerance can be used (see Table 121).

If XTAL2 is not used, XTAL2_IN should be tied to V_{DD} and XTAL2_OUT should be left unconnected.

Table 121. XTAL2 Specifications

Parameter	Value	
Frequency	12.352 MHz	
Oscillation Mode	Fundamental, parallel resonance	
Effective Series Resistance	75* Ω maximum	
Load Capacitance	18‡ pF	
Shunt Capacitance	7 pF maximum	
Frequency Tolerance and Stability	32 ppm	

* 120 Ω maximum for 6.176 MHz crystal.
† 24 pF for 6.176 MHz crystal also.
‡ 18 pF for 6.176 MHz crystal also.

If an oscillator is used (see Section 7.5.1 on page 81), the signal has to be connected to the XTAL2_IN pin. XTAL2_OUT must be left unconnected in this configuration. XTAL2_IN and XTAL2_OUT are not 5 V tolerant. The oscillator must meet the requirements shown below.

Table 122. 6.176 MHz/12.352 MHz Oscillator Requirements

Parameter	Value		Value	
Frequency	6.176 MHz		12.352 MHz	
Maximum Rise or Fall Time	10 ns, 10%—90% V_{DD}		10 ns, 10%—90% V_{DD}	
Minimum Pulse Width	Low	High	Low	High
	54 ns	54 ns	27 ns	27 ns

15 Electrical Characteristics (continued)

15.2.3 Reset Pulse

Table 123. Reset Pulse

Parameter	Min	Max	Unit
RESET# Minimum Pulse Width	61	—	ns

15.3 Thermal Considerations for the 272 PBGA

Table 124. Thermal Considerations

Parameter	Body Size (mm sq.)	Array Details	Ball Pitch (mm)	Number of Layers	Theta-JA (°C/W)			Maximum Power (Natural Convection) (Watts)
					Natural convection	200 fpm	500 fpm	
4-layer JEDEC Test Board	27	Peripheral + T.A.	1.27	2	22.5	19	17.5	2.44

15.4 dc Electrical Characteristics

15.4.1 PCI Signals

All PCI signals meet the electrical requirements as specified in the *PCI Local Bus*, Rev 2.2, specification. PCI interface timing diagrams can be found in Figure 6—Figure 15, starting on page 25.

15.4.2 Electrical Drive Specifications, CT_C8 and /CT_FRAME

V_{DD} = 3.3 V and V_{SS} = 0.0 V, unless otherwise specified.

Table 125. Electrical Drive Specifications, CT_C8 and /CT_FRAME

Parameter	Symbol	Condition	Min	Max	Unit
Output High Voltage	V_{OH}	I_{OUT} = –24 mA	2.4	3.3	V
Output Low Voltage	V_{OL}	I_{OUT} = 24 mA	–0.25	0.4	V
Positive-going Threshold	Vt+	—	1.2	2.0	V
Negative-going Threshold	Vt–	—	0.6	1.6	V
Hysteresis (Vt+—Vt–)	V_{HYS}	—	0.4	—	V
Input Pin Capacitance	C_{IN}	—	—	10	pF

PCI-compliant data line I/O cells are used for the CT bus data lines. (See *PCI Specification*, Rev. 2.2, Chapter 4.)

/C16, /C4, C2, SCLK, /SCLKx2, and /FR_COMP all use the same driver/receiver pairs as those specified for the CT_C8 and /CT_FRAME signals, though this is not explicitly stated as a part of the H.1x0 specification.

15 Electrical Characteristics (continued)

15.4.3 All Other Pins

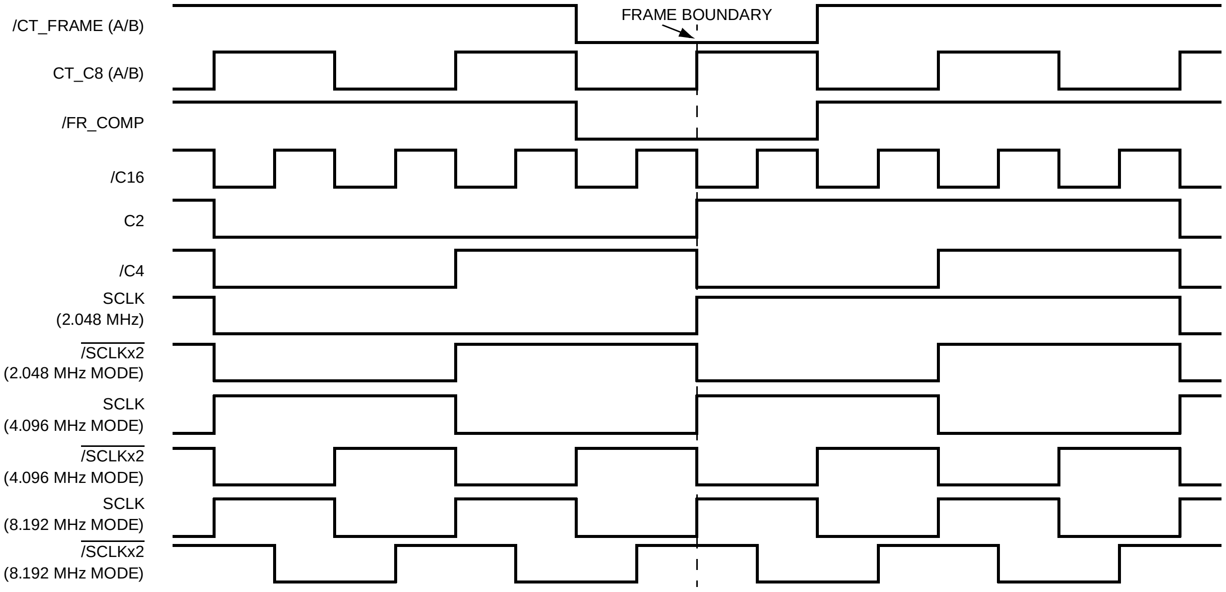
V_{DD} = 3.3 V and V_{SS} = 0.0 V, unless otherwise specified.

Table 126. dc Electrical Characteristics, All Other Pins

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Supply Current	I_{DD}	2000 H-bus/L-bus connections	—	300	—	mA
Supply Voltage	V_{DD}	—	3.0	—	3.6	V
Input High Voltage	V_{IH}	—	2.0	—	—	V
Input Low Voltage	V_{IL}	—	—	—	0.8	V
Input Current	I_I	—	—	—	1	µA
Input Capacitance (input only)	C_I	—	—	—	5	pF
Input Capacitance (I/O pins)	C_{IO}	—	—	—	10	pF
Leakage Current (3-state)	I_{LEAK}	—	—	—	10	µA
Input Clamp Voltage	V_C	—	—	—	–1.0	V
Output High Voltage	V_{OH}	I = 8 mA	2.4	—	—	V
Output Low Voltage	V_{OL}	I = 8 mA	—	—	0.4	V
Output Short-circuit Current	I_{OS}	V_{OH} tied to GND	—	—	100	mA

15.5 H-Bus Timing

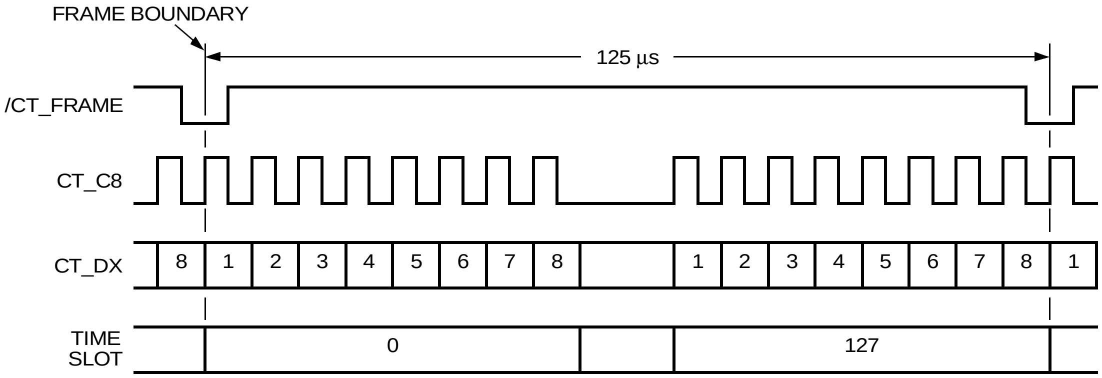
15.5.1 Timing Diagrams



5-6119F

Figure 59. Clock Alignment

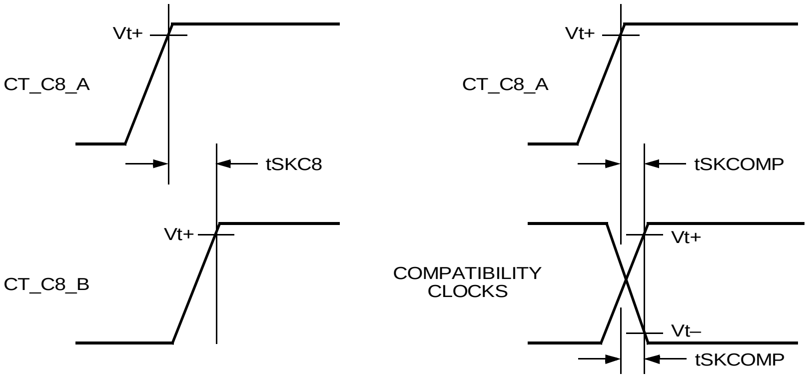
15 Electrical Characteristics (continued)



Note: Bit 1 is the MSB and Bit 8 is the LSB. MSB is always transmitted first in all transfers.

5-6120F

Figure 60. Frame Timing Diagram



5-6122F

Figure 61. Detailed Clock Skew Timing Diagram

15.6 ac Electrical Characteristics

15.6.1 Skew Timing, H-Bus

Table 127. Skew Timing, H-Bus

Symbol	Parameter	Min	Typical	Max	Unit
tSKC8	Maximum Skew Between CT_C8_A and CT_C8_B* †‡ §	—	—	±10, ±Φ	ns
TSKCOMP	Maximum Skew Between CT_C8_A and any Compatibility Clock*	—	—	±5	ns
—	Maximum Skew Between CT_C8_A and L_SCx Clock*	—	—	±2	ns

* Test load—50 pF.

† Assumes A and B masters in adjacent slots.

‡ When static skew is 10 ns and in the same clock cycle, each clock performs a 10 ns phase correction in opposite directions, a maximum skew of 30 ns will occur during that clock cycle.

§ Meeting the skew requirements in Table 127 and the requirements of Section 15.5 H-Bus Timing on page 169 could require the PLLs generating CT_C8 to have different time constants when acting as primary and secondary clock masters.

15 Electrical Characteristics (continued)

Table 128. L_SC[3:0] and Frame Group Rise and Fall Time

Parameter	Min	Typ	Max	Unit*
L_SCx Rise Time	—	—	5	ns
L_SCx Fall Time	—	—	4	ns
Frame Group Rise Time	—	—	3	ns
Frame Group Fall Time	—	—	3	ns

* Worst-case loading of 50 pF on all outputs.

15.7 Hot-Swap

The T8110 has features which assist in H.110 hot swap applications. All H.110 bus signals are put in high impedance (3-state and/or input) during the early power phase of board insertion/removal. The ECTF H.110 specification requires that all CT data lines and CT_NETREF clocks have 0.7 V applied through 18 kΩ resistors before plugging into and releasing from the H.110 bus. A feature on the T8110 incorporates all 34 18 kΩ precharge resistors internally (32 for the CT data signals, 2 for NETREFs). These resistors accept 0.7 V directly through the VPRECHARGE input. The ECTF H.110 specification requires that the T8110 must be powered from early power in hot swap applications. The circuit that generates the 0.7 V precharge voltage must also be powered from early power. Refer to ECTF H.110 and *PICMG CompactPCI Hot Swap* specifications for hot swap requirements.

15.7.1 LPUE (Local Pull-Up Enable)

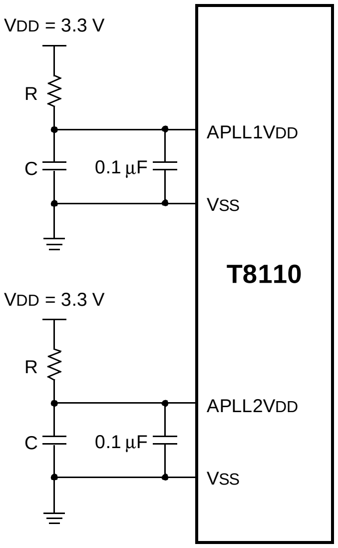
LPUE is used as an assist in *CompactPCI* specifically for Hot Swap; see Section 2.3.2 on page 18. During live board insertion/removal, the only devices which should be on early power are the power controller and interface parts (PCI interface attached to J1, H.110 interface attached to J4). Without the LPUE, any device connected to the T8110 would get current flow from the early power through the pull-up resistors. When late power parts power up, they already have current flowing through the I/O and these devices could possibly latch up. The current flow is eliminated by LPUE disabling the pull-up resistors. LPUE is typically controlled by the power controller. The power controller will pull LPUE low during board insertion/removal and will release LPUE high so that the pull ups are re-enabled with late power turning on. Signals that have pull-ups disabled by LPUE are 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.

15.8 Decoupling

Decoupling the T8110 VDDs with 0.1 µF capacitors is recommended. 1000 pF or 0.01 µF capacitors may be used in addition to the 0.1 µF capacitors to provide additional decoupling.

15.9 APLL VDD Filter

Separate VDDs are provided for APLL1 and APLL2 for filtering purposes. VDD filtering will provide stability in the APLL, primarily the VCOs. An R/C low pass filter should be applied to the PLL VDDs, see Figure 62. Depending on the quality of VDD and board layout characteristics, the R/C values should be selected to filter out unwanted frequencies above a targeted frequency. For example, a 25 Ω resistor and 10 µF capacitor will have a cut-off frequency of 636 Hz. Characterize the quality of your VDD and select component values accordingly. 25 Ω is the maximum recommended resistor value. At high frequencies the ESR of a bulk cap becomes a problem (no longer effectively low passes) so a high-frequency cap of 0.1 µF or so is required to compensate for some of the higher clocks and various harmonics. This needs to be placed as close to the T8110 device as possible to minimize the radiational pick-up in the remaining trace length. APLL1 and APLL2 each draw approximately 7 mA at 3.3 V. Hot swap applications can use late power to ensure the capacitance and in-rush current do not violate the PICMG Hot Swap specification.



0995(F)

Figure 62. APLL VDD Filtering

15 Electrical Characteristics (continued)

15.10 PC Board PBGA Considerations

The T8110 is a 272-ball plastic ball grid array package. The 16 centrally located thermal balls should be connected to board ground. While there are no special printed-circuit board requirements for the T8110, there are specification requirements for PC board layout that must be adhered to. For instance, per the ECTF H.110 specification, all CT bus data signals must not exceed 4 inches in length from connector to I/O cell and all CT bus clock signals must not exceed 2 inches in length from connector to I/O cell. We advise the customer to become familiar with applicable specifications for any PC board requirements.

15.11 Unused Pins

If the PCI interface is not used, these signals should be pulled up/down to their inactive state. Multiple pins may share a common resistor. Signals with pull-up/down resistors may be left unconnected if unused. Unused 8 mA 3-state signals may be left unconnected. If XTAL1_IN and/or XTAL2_IN are being driven from an oscillator, then XTAL1_OUT and/or XTAL2_OUT must be left unconnected. If XTAL2 is not used, XTAL2_IN should be pulled-up or tied directly to V_{DD} and XTAL2_OUT should be left unconnected. If VPRECHARGE is unused, this signal may be left unconnected. All signals listed as **no connect** (A20, D16, D20, E17, F20, and G17) in Table 9 must be left unconnected.

15.12 T8110 Evaluation Boards

Two development kits are available for the T8110, one for PCI and one for *CompactPCI*. Each kit contains an evaluation board, software, documentation and unrestricted access to the T8110 help desk. The *CompactPCI* evaluation board is Full Hot Swap; it includes the T8110 chip, a PCI transparent bridge, a dual T1/E1 line interface, dual codec, and dual SLIC. The PCI evaluation board is a full-length PCI board and is comparably featured (no hot swap). The *CompactPCI* evaluation board is designed to use the PCI interface only, and the PCI board has an option to use the PCI interface **or** the microprocessor interface. Software includes full source code, including rights to re-use. Documentation CD includes evaluation board schematics (ORCAD and .pdf formats) evaluation board bill of material (BOM), data sheets, and advisories. Refer to website www.agere.com/ambassador for additional information.

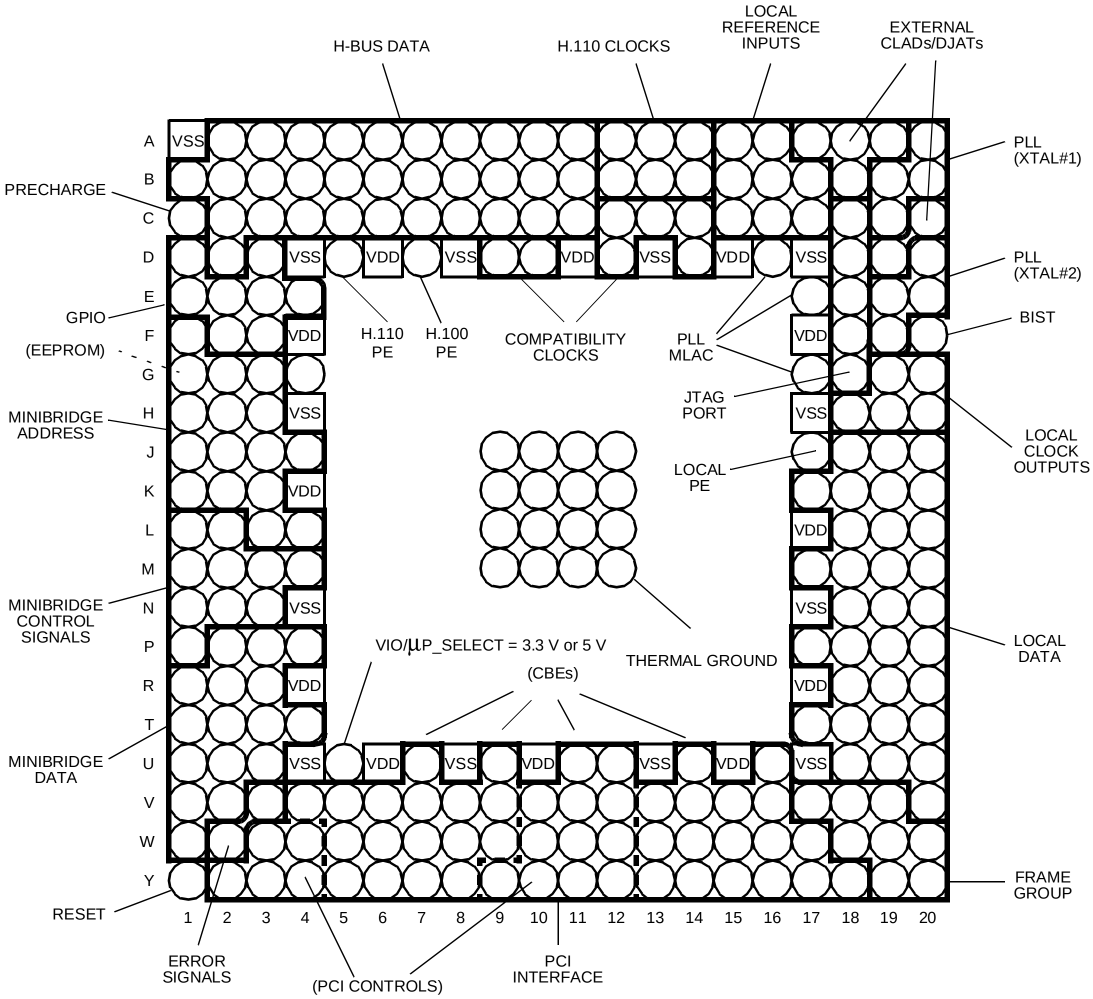
15.13 T8110 Ordering Information

Table 129. T8110 Ordering Information

Device Part Number	Package	Comcode
T-8110---BAL-DB	272-Ball PBGA	108560269

16 Package Outline

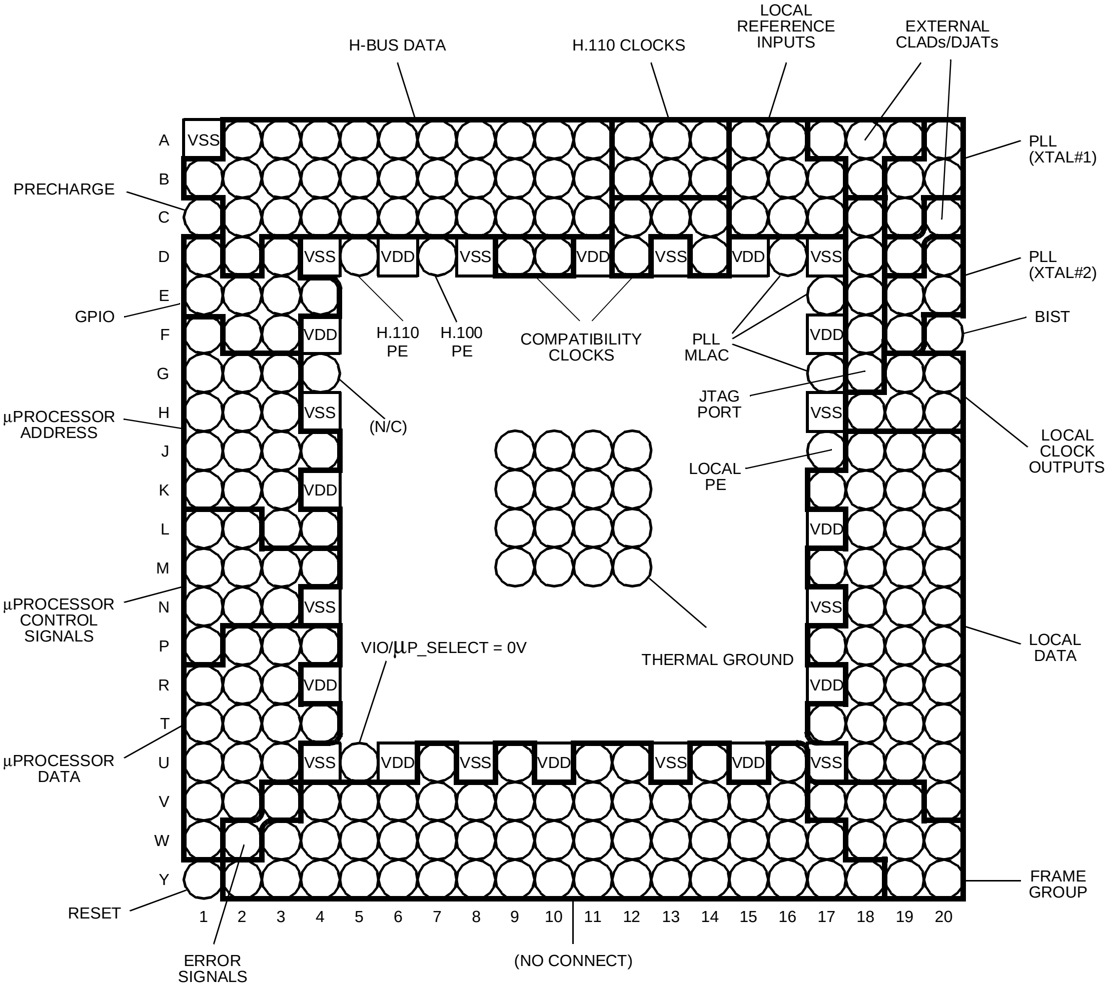
16.1 Pin and Pad Assignments



5-8906F(a).

Figure 63. T8110 Pins by Functional Group, PCI I/F Enabled, Microprocessor I/F Disabled

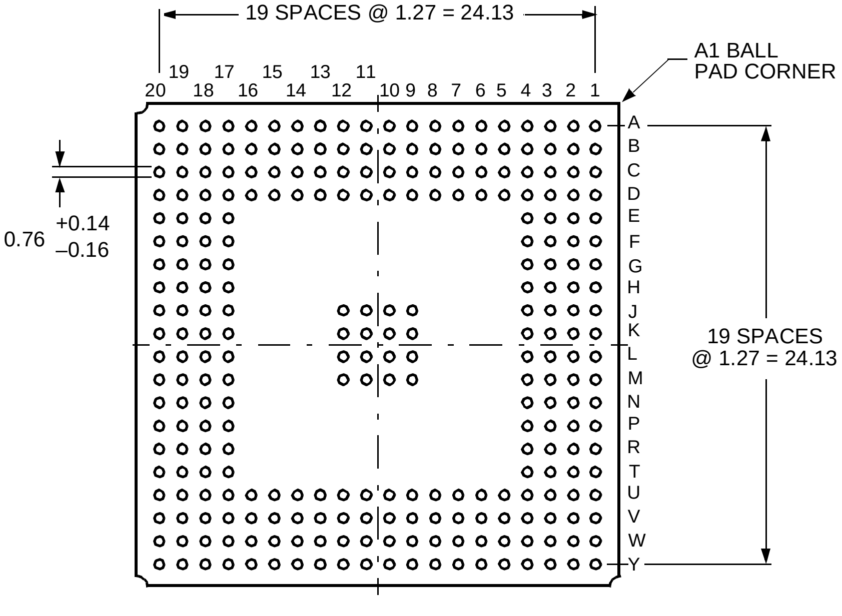
16 Package Outline (continued)



5-8906F(b).

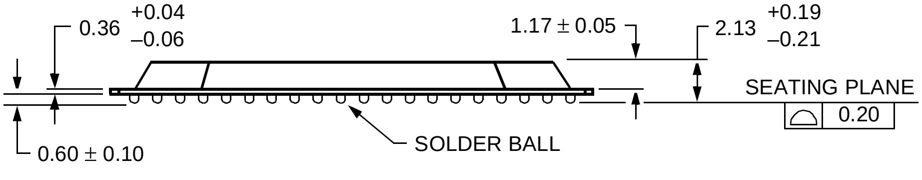
Figure 64. T8110 Pins by Functional Group, Microprocessor I/F Enabled, PCI I/F Disabled

16 Package Outline (continued)



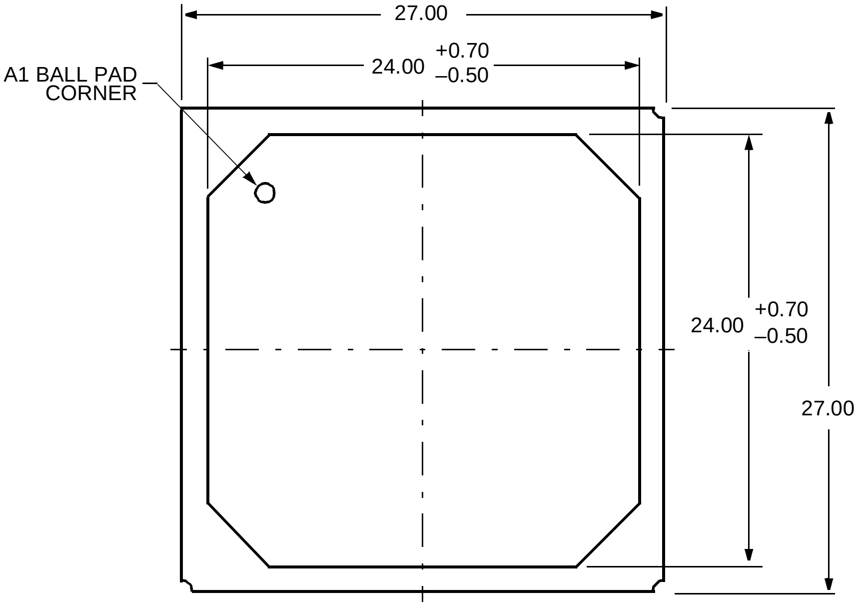
BOTTOM VIEW

16 Package Outline (continued)



5-4406

SIDE VIEW



5-4406

TOP VIEW

17 JTAG/Boundary Scan

17.1 The Principle of Boundary-Scan Architecture

Each primary input signal and primary output signal is supplemented with a multipurpose memory element called a boundary-scan cell. Cells on device primary inputs are referred to as input cells and cells on primary outputs are referred to as output cells. Input and output is relative to the core logic of the device.

At any time, only one register can be connected from TDI to TDO. For example, instruction register (IR), bypass, boundary-scan, ident, or even some appropriate register internal to the core logic (see Figure 65). The selected register is identified by the decoded output of the instruction register. Certain instructions are mandatory, such as EXTEST (boundary-scan register selected), whereas others are optional, such as the IDCODE instruction (Ident register selected).

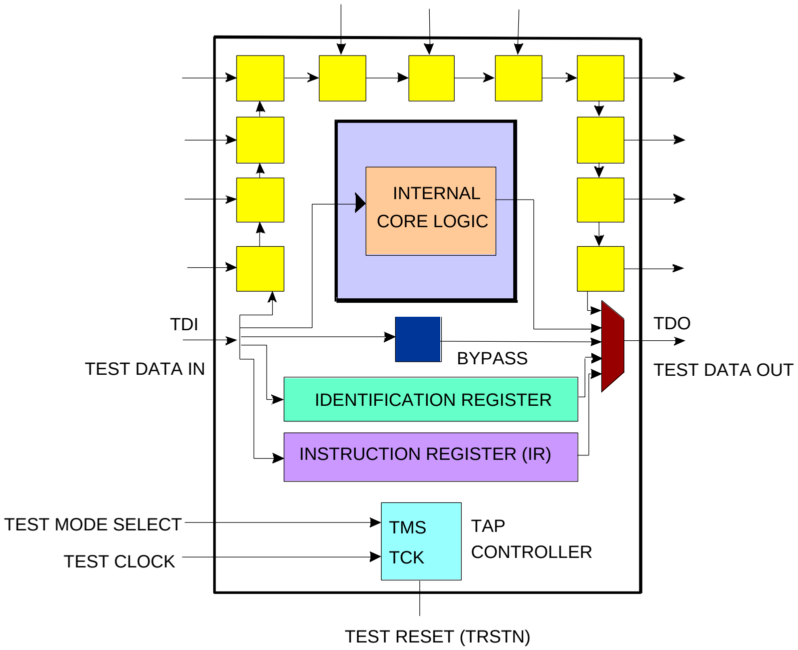


Figure 65. *IEEE 1149.1 Boundary-Scan Architecture**

Figure 65 shows the following elements:

- A set of four dedicated test pins, test data in (TDI), test mode select (TMS), test clock (TCK), test data out (TDO), and one optional test pin test reset (TRSTN). These pins are collectively referred to as the test access port (TAP).
- A boundary-scan cell on each device's primary input and primary output pin, connected internally to form a serial boundary-scan register (boundary scan).
- A finite-state machine TAP controller with inputs TCK and TMS.
- An n-bit (n = 3) instruction register (IR), holding the current instruction.
- A 1-bit bypass register (BYPASS).
- An optional 32-bit identification register (ident) capable of being loaded with a permanent device identification code.

* *IEEE* is a registered trademark of The Institute of Electrical and Electronic Engineers, Inc.

17 JTAG/Boundary Scan

17.1.1 Instruction Register

The instruction register is 3 bits long and the capture value is 001.

Table 130. Instruction Register

Instruction	Binary Code	Description
EXTEST	000	Places the boundary-scan register in EXTEST mode.
SAMPLE	001	Places the boundary-scan register in sample mode.
IDCODE	101	Identification code.
BYPASS	110, 111	Places the bypass register in the scan chain.
HIGH Z	010	Places all outputs and I/Os in 3-state mode.

17.2 Boundary-Scan Register

Note: The control column of the following table indicates the value for boundary-scan control of this pin.

Table 131. Boundary-Scan Register Description

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
0	VSS	A1	Linkage	—	—	—
1	BOUT_CT_D_EN(27)	—	Controller	—	—	—
2	CT_D27	A2	I/O	BOUT_CT_D_EN(27)	0	High Z
3	BOUT_CT_D_EN(24)	—	Controller	—	—	—
4	CT_D24	A3	I/O	BOUT_CT_D_EN(24)	0	High Z
5	BOUT_CT_D_EN(21)	—	Controller	—	—	—
6	CT_D21	A4	I/O	BOUT_CT_D_EN(21)	0	High Z
7	BOUT_CT_D_EN(19)	—	Controller	—	—	—
8	CT_D19	A5	I/O	BOUT_CT_D_EN(19)	0	High Z
9	BOUT_CT_D_EN(16)	—	Controller	—	—	—
10	CT_D16	A6	I/O	BOUT_CT_D_EN(16)	0	High Z
11	BOUT_CT_D_EN(13)	—	Controller	—	—	—
12	CT_D13	A7	I/O	BOUT_CT_D_EN(13)	0	High Z
13	BOUT_CT_D_EN(11)	—	Controller	—	—	—
14	CT_D11	A8	I/O	BOUT_CT_D_EN(11)	0	High Z
15	BOUT_CT_D_EN(8)	—	Controller	—	—	—
16	CT_D8	A9	I/O	BOUT_CT_D_EN(8)	0	High Z
17	BOUT_CT_D_EN(4)	—	Controller	—	—	—
18	CT_D4	A10	I/O	BOUT_CT_D_EN(4)	0	High Z
19	BOUT_CT_D_EN(0)	—	Controller	—	—	—
20	CT_D0	A11	I/O	BOUT_CT_D_EN(0)	0	High Z
21	BOUT_CT_A_EN	—	Controller	—	—	—

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
22	/CT_FRAME_A	A12	I/O	BOUT_CT_A_EN	0	High Z
23	CT_C8_A	A13	I/O	BOUT_CT_A_EN	0	High Z
24	BOUT_CT_NETREF1_En	—	Controller	—	—	—
25	CT_NETREF1	A14	I/O	BOUT_CT_NETREF1_En	0	High Z
26	L_REF0	A15	Input	BOUT_CT_NETREF1_En	—	—
27	L_REF4	A16	Input	BOUT_CT_NETREF1_En	—	—
28	GENERAL_EN	—	Controller	—	—	—
29	PRI_REF_OUT	A17	Output3	GENERAL_EN	0	High Z
30	PRI_REF_IN	A18	Clock	GENERAL_EN	—	—
31	NR1_DIV_IN	A19	Input	GENERAL_EN	—	—
32	PEN	A20	Enable0	GENERAL_EN(0)	—	50 KΩ up
33	BOUT_CT_D_EN(29)	—	Controller	—	—	—
34	CT_D29	B1	I/O	BOUT_CT_D_EN(29)	0	High Z
35	BOUT_CT_D_EN(28)	—	Controller	—	—	—
36	CT_D28	B2	I/O	BOUT_CT_D_EN(28)	0	High Z
37	BOUT_CT_D_EN(25)	—	Controller	—	—	—
38	CT_D25	B3	I/O	BOUT_CT_D_EN(25)	0	High Z
39	BOUT_CT_D_EN(22)	—	Controller	—	—	—
40	CT_D22	B4	I/O	BOUT_CT_D_EN(22)	0	High Z
41	BOUT_CT_D_EN(20)	—	Controller	—	—	—
42	CT_D20	B5	I/O	BOUT_CT_D_EN(20)	0	High Z
43	BOUT_CT_D_EN(17)	—	Controller	—	—	—
44	CT_D17	B6	I/O	BOUT_CT_D_EN(17)	0	High Z
45	BOUT_CT_D_EN(14)	—	Controller	—	—	—
46	CT_D14	B7	I/O	BOUT_CT_D_EN(14)	0	High Z
47	BOUT_CT_D_EN(9)	—	Controller	—	—	—
48	CT_D9	B8	I/O	BOUT_CT_D_EN(9)	0	High Z
49	BOUT_CT_D_EN(6)	—	Controller	—	—	—
50	CT_D6	B9	I/O	BOUT_CT_D_EN(6)	0	High Z
51	BOUT_CT_D_EN(5)	—	Controller	—	—	—
52	CT_D5	B10	I/O	BOUT_CT_D_EN(5)	0	High Z
53	BOUT_CT_D_EN(1)	—	Controller	—	—	—
54	CT_D1	B11	I/O	BOUT_CT_D_EN(1)	0	High Z
55	BOUT_CT_B_EN	—	Controller	—	—	—
56	/CT_FRAME_B	B12	I/O	BOUT_CT_B_EN	0	High Z
57	CT_C8_B	B13	I/O	BOUT_CT_B_EN	0	High Z

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
58	BOUT_CT_NETREF2_En	—	Controller	—	—	—
59	CT_NETREF2	B14	I/O	BOUT_CT_NETREF2_En	0	High Z
60	L_REF1	B15	Input	BOUT_CT_NETREF2_En	—	—
61	L_REF5	B16	Input	BOUT_CT_NETREF2_En	—	—
62	L_REF6	B17	Input	BOUT_CT_NETREF2_En	—	—
63	NR1_SEL_OUT	B18	Output3	GENERAL_EN	0	High Z
64	APLL1VDD	B19	Linkage	—	—	—
65	XTAL1_IN	B20	Linkage	—	—	—
66	VPRECHARGE	C1	Linkage	—	—	—
67	BOUT_CT_D_EN(30)	—	Controller	—	—	—
68	CT_D30	C2	I/O	BOUT_CT_D_EN(30)	0	High Z
69	BOUT_CT_D_EN(26)	—	Controller	—	—	—
70	CT_D26	C3	I/O	BOUT_CT_D_EN(26)	0	High Z
71	BOUT_CT_D_EN(23)	—	Controller	—	—	—
72	CT_D23	C4	I/O	BOUT_CT_D_EN(23)	0	High Z
73	BOUT_CT_D_EN(18)	—	Controller	—	—	—
74	CT_D18	C5	I/O	BOUT_CT_D_EN(18)	0	High Z
75	BOUT_CT_D_EN(15)	—	Controller	—	—	—
76	CT_D15	C6	I/O	BOUT_CT_D_EN(15)	0	High Z
77	BOUT_CT_D_EN(12)	—	Controller	—	—	—
78	CT_D12	C7	I/O	BOUT_CT_D_EN(12)	0	High Z
79	BOUT_CT_D_EN(10)	—	Controller	—	—	—
80	CT_D10	C8	I/O	BOUT_CT_D_EN(10)	0	High Z
81	BOUT_CT_D_EN(7)	—	Controller	—	—	—
82	CT_D7	C9	I/O	BOUT_CT_D_EN(7)	0	High Z
83	BOUT_CT_D_EN(2)	—	Controller	—	—	—
84	CT_D3	C10	I/O	BOUT_CT_D_EN(2)		High Z
85	BOUT_CT_D_EN(3)	—	Controller	—	—	—
86	CT_D2	C11	I/O	BOUT_CT_D_EN(3)	0	High Z
87	BOUT_FRN_COMP_EN	—	Controller	—	—	—
88	/FRN_COMP	C12	I/O	BOUT_FRN_COMP_EN	0	50 kΩ up
89	BOUT_SCBUS_CLOCKS_En	—	Controller	—	—	—
90	/SCLKX2	C13	I/O	BOUT_SCBUS_CLOCKS_En	0	50 kΩ up
91	SCLK	C14	I/O	BOUT_SCBUS_CLOCKS_En	0	50 kΩ up
92	L_REF2	C15	Input	BOUT_SCBUS_CLOCKS_En	—	—
93	L_REF3	C16	Input	BOUT_SCBUS_CLOCKS_En	—	—

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
94	L_REF7	C17	Input	BOUT_SCBUS_CLOCKS_En	—	—
95	TRST#	C18	TRST	BOUT_SCBUS_CLOCKS_En	—	50 kΩ up
96	XTAL1_OUT	C19	Linkage	BOUT_SCBUS_CLOCKS_En	—	—
97	NR2_DIV_IN	C20	Input	BOUT_SCBUS_CLOCKS_En	—	—
98	BOUT_GP_EN(0)	—	Controller	—	—	—
99	GP0	D1	I/O	BOUT_GP_EN(0)	0	High Z
100	BOUT_CT_D_EN(31)	—	Controller	—	—	—
101	CT_D31	D2	I/O	BOUT_CT_D_EN(31)	0	High Z
102	BOUT_GP_EN(4)	—	Controller	—	—	—
103	GP4	D3	I/O	BOUT_GP_EN(4)	0	High Z
104	V_{SS}	D4	Linkage	—	—	—
105	H110_ENABLE	D5	Enable1	BOUT_GP_EN(4)	—	20 kΩ down
106	V_{DD}	D6	Linkage	—	—	—
107	H100_ENABLE	D7	Enable0	BOUT_GP_EN(4)	—	20 kΩ down
108	V_{SS}	D8	Linkage	—	—	—
109	BOUT_HMVIP_CLOCKS_En	—	Controller	BOUT_GP_EN(4)	—	—
110	/C16+	D9	I/O	BOUT_HMVIP_CLOCKS_En	0	50 kΩ up
111	/C16–	D10	I/O	BOUT_HMVIP_CLOCKS_En	0	50 kΩ up
112	VDD	D11	Linkage	—	—	—
113	BOUT_MVIP_CLOCKS_En	—	Controller	—	—	—
114	/C4	D12	I/O	BOUT_MVIP_CLOCKS_En	0	50 kΩ up
115	V_{SS}	D13	Linkage	—	—	—
116	C2	D14	I/O	BOUT_MVIP_CLOCKS_En	0	50 kΩ up
117	V_{DD}	D15	Linkage	—	—	—
118	PLOCK	D16	Linkage	BOUT_MVIP_CLOCKS_En	—	—
119	V_{SS}	D17	Linkage	—	—	—
120	TMS	D18	TMS	BOUT_MVIP_CLOCKS_En	—	50 kΩ up
121	NR2_SEL_OUT	D19	Output3	GENERAL_EN	0	High Z
122	PSEL	D20	Linkage	GENERAL_EN	—	20 kΩ down
123	BOUT_GP_EN(1)	—	Controller	—	—	—
124	GP1	E1	I/O	BOUT_GP_EN(1)	0	High Z
125	BOUT_GP_EN(2)	—	Controller	—	—	—
126	GP2	E2	I/O	BOUT_GP_EN(2)	0	High Z
127	BOUT_GP_EN(6)	—	Controller	—	—	—

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
128	GP5	E3	I/O	BOUT_GP_EN(6)	0	High Z
129	BOUT_GP_EN(7)	—	Controller	—	—	—
130	GP7	E4	I/O	BOUT_GP_EN(7)	0	High Z
131	PTEST	E17	Linkage	BOUT_GP_EN(7)	—	20 kΩ down
132	TCK	E18	TCK	BOUT_GP_EN(7)	—	50 kΩ up
133	APLL2VDD	E19	Linkage		—	—
134	XTAL2_IN	E20	Linkage	BOUT_GP_EN(7)	—	—
135	BOUT_MB_EN	—	Controller	—	—	—
136	MB_A0	F1	I/O	BOUT_MB_EN	0	20 kΩ down
137	BOUT_GP_EN(3)	—	Controller	—	—	—
138	GP3	F2	I/O	BOUT_GP_EN(3)	0	High Z
139	BOUT_GP_EN(5)	—	Controller	—	—	—
140	GP6	F3	I/O	BOUT_GP_EN(5)	0	High Z
141	VDD	F4	Linkage	—	—	—
142	VDD	F17	Linkage	—	—	—
143	TDI	F18	TDI	BOUT_GP_EN(5)	—	50 kΩ up
144	XTAL2_OUT	F19	Linkage	BOUT_GP_EN(5)	—	—
145	TESTMODE	F20	Enable0	BOUT_GP_EN(5)	—	20 kΩ down
146	BOUT_MB_A1_EN	—	Controller	—	—	—
147	MB_A1	G1	I/O	BOUT_MB_A1_EN	0	20 kΩ down
148	BOUT_MB_A2_EN	—	Controller	—	—	—
149	MB_A2	G2	I/O	BOUT_MB_A2_EN	0	20 kΩ down
150	MB_A3	G3	I/O	BOUT_MB_A2_EN	0	20 kΩ down
151	OUT_EE_CS_EN	—	Controller	—	—	—
152	EE_CS	G4	Output3	OUT_EE_CS_EN	0	High Z
153	PPDN	G17	Linkage	OUT_EE_CS_EN	—	20 kΩ down
154	TDO	G18	TDO	OUT_EE_CS_EN	—	—
155	OUT_L_SC_En(3)	—	Controller	—	—	—
156	L_SC3	G19	Output3	OUT_L_SC_En(3)	0	High Z
157	OUT_TCLK_OUT_En	—	Controller	—	—	—
158	TCLK_OUT	G20	Output3	OUT_TCLK_OUT_En	0	High Z

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
159	MB_A4	H1	I/O	BOUT_MB_EN	0	20 kΩ down
160	MB_A5	H2	I/O	BOUT_MB_EN	0	20 kΩ down
161	MB_A6	H3	I/O	BOUT_MB_EN	0	20 kΩ down
162	Vss	H4	Linkage	—	—	—
163	Vss	H17	Linkage	—	—	—
164	OUT_L_SC_En(2)	—	Controller	—	—	—
165	L_SC2	H18	Output3	OUT_L_SC_En(2)	0	High Z
166	OUT_L_SC_En(1)	—	Controller	—	—	—
167	L_SC1	H19	Output3	OUT_L_SC_En(1)	0	High Z
168	OUT_L_SC_En(0)	—	Controller	—	—	—
169	L_SC0	H20	Output3	OUT_L_SC_En(0)	0	High Z
170	MB_A8	J1	I/O	BOUT_MB_EN	0	20 kΩ down
171	MB_A9	J2	I/O	BOUT_MB_EN	0	20 kΩ down
172	MB_A10	J3	I/O	BOUT_MB_EN	0	20 kΩ down
173	MB_A7	J4	I/O	BOUT_MB_EN	0	20 kΩ down
174	LPUE	J17	Enable1	BOUT_MB_EN	—	50 kΩ up
175	BOUT_L_D_EN(2)	—	Controller	—	—	—
176	L_D2	J18	I/O	BOUT_L_D_EN(2)	0	High Z
177	BOUT_L_D_EN(1)	—	Controller	—	—	—
178	L_D1	J19	I/O	BOUT_L_D_EN(1)	0	High Z
179	BOUT_L_D_EN(0)	—	Controller	—	—	—
180	L_D0	J20	I/O	BOUT_L_D_EN(0)	0	High Z
181	MB_A12	K1	I/O	BOUT_MB_EN	0	20 kΩ down
182	MB_A13	K2	I/O	BOUT_MB_EN	0	20 kΩ down
183	MB_A11	K3	I/O	BOUT_MB_EN	0	20 kΩ down
184	VDD	K4	Linkage	—	—	—
185	BOUT_L_D_EN(3)	—	Controller	—	—	—
186	L_D3	K17	I/O	BOUT_L_D_EN(3)	0	High Z
187	BOUT_L_D_EN(6)	—	Controller	—	—	—

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
188	L_D6	K18	I/O	BOUT_L_D_EN(6)	0	High Z
189	BOUT_L_D_EN(5)	—	Controller	—	—	—
190	L_D5	K19	I/O	BOUT_L_D_EN(5)	0	High Z
191	BOUT_L_D_EN(4)	—	Controller	—	—	—
192	L_D4	K20	I/O	BOUT_L_D_EN(4)	0	High Z
193	MB_CS0	L1	I/O	BOUT_MB_EN	0	20 kΩ down
194	MB_CS1	L2	I/O	BOUT_MB_EN	0	20 kΩ down
195	MB_A14	L3	I/O	BOUT_MB_EN	0	20 kΩ down
196	MB_A15	L4	I/O	BOUT_MB_EN	0	20 kΩ down
197	VDD	L17	Linkage	—	—	—
198	BOUT_L_D_EN(7)	—	Controller	—	—	—
199	L_D7	L18	I/O	BOUT_L_D_EN(7)	0	High Z
200	BOUT_L_D_EN(9)	—	Controller	—	—	—
201	L_D9	L19	I/O	BOUT_L_D_EN(9)	0	High Z
202	BOUT_L_D_EN(8)	—	Controller	—	—	—
203	L_D8	L20	I/O	BOUT_L_D_EN(8)	0	High Z
204	MB_CS2	M1	I/O	BOUT_MB_EN	0	20 kΩ down
205	MB_CS3	M2	I/O	BOUT_MB_EN	0	20 kΩ down
206	MB_CS4	M3	I/O	BOUT_MB_EN	0	High Z
207	MB_CS5	M4	I/O	BOUT_MB_EN	0	High Z
208	BOUT_L_D_EN(11)	—	Controller	—	—	—
209	L_D11	M17	I/O	BOUT_L_D_EN(11)	0	High Z
210	BOUT_L_D_EN(10)	—	Controller	—	—	—
211	L_D10	M18	I/O	BOUT_L_D_EN(10)	0	High Z
212	BOUT_L_D_EN(13)	—	Controller	—	—	—
213	L_D13	M19	I/O	BOUT_L_D_EN(13)	0	High Z
214	BOUT_L_D_EN(12)	—	Controller	—	—	—
215	L_D12	M20	I/O	BOUT_L_D_EN(12)	0	High Z
216	MB_RD	N1	I/O	BOUT_MB_EN	0	High Z
217	OUT_MB_CS6_EN	—	Controller	—	—	—
218	MB_CS6	N2	Output3	OUT_MB_CS6_EN	0	High Z
219	MB_CS7	N3	I/O	BOUT_MB_EN	0	High Z

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
220	Vss	N4	Linkage	—	—	—
221	Vss	N17	Linkage	—	—	—
222	BOUT_L_D_EN(15)	—	Controller	—	—	—
223	L_D15	N18	I/O	BOUT_L_D_EN(15)	0	High Z
224	BOUT_L_D_EN(14)	—	Controller	—	—	—
225	L_D14	N19	I/O	BOUT_L_D_EN(14)	0	High Z
226	BOUT_L_D_EN(16)	—	Controller	—	—	—
227	L_D16	N20	I/O	BOUT_L_D_EN(16)	0	High Z
228	MB_WR	P1	I/O	BOUT_MB_EN	0	High Z
229	BOUT_MB_D_En	—	Controller	—	—	—
230	MB_D14	P2	I/O	BOUT_MB_D_En	0	High Z
231	MB_D15	P3	I/O	BOUT_MB_D_En	0	High Z
232	MB_D11	P4	I/O	BOUT_MB_D_En	0	High Z
233	BOUT_L_D_EN(23)	—	Controller	—	—	—
234	L_D23	P17	I/O	BOUT_L_D_EN(23)	0	High Z
235	BOUT_L_D_EN(19)	—	Controller	—	—	—
236	L_D19	P18	I/O	BOUT_L_D_EN(19)	0	High Z
237	BOUT_L_D_EN(18)	—	Controller	—	—	—
238	L_D18	P19	I/O	BOUT_L_D_EN(18)	0	High Z
239	BOUT_L_D_EN(17)	—	Controller	—	—	—
240	L_D17	P20	I/O	BOUT_L_D_EN(17)	0	High Z
241	MB_D12	R1	I/O	BOUT_MB_D_En	0	High Z
242	MB_D13	R2	I/O	BOUT_MB_D_En	0	High Z
243	MB_D10	R3	I/O	BOUT_MB_D_En	0	High Z
244	VDD	R4	Linkage	—	—	—
245	VDD	R17	Linkage	—	—	—
246	BOUT_L_D_EN(22)	—	Controller	—	—	—
247	L_D22	R18	I/O	BOUT_L_D_EN(22)	0	High Z
248	BOUT_L_D_EN(21)	—	Controller	—	—	—
249	L_D21	R19	I/O	BOUT_L_D_EN(21)	0	High Z
250	bout_L_D_EN(20)	—	Controller	—	—	—
251	L_D20	R20	I/O	BOUT_L_D_EN(20)	0	High Z
252	MB_D8	T1	I/O	BOUT_MB_D_En	0	High Z
253	MB_D9	T2	I/O	BOUT_MB_D_En	0	High Z
254	MB_D6	T3	I/O	BOUT_MB_D_En	0	High Z
255	MB_D7	T4	I/O	BOUT_MB_D_En	0	High Z

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
256	BOUT_L_D_EN(31)	—	Controller	—	—	—
257	L_D31	T17	I/O	BOUT_L_D_EN(31)	0	High Z
258	BOUT_L_D_EN(26)	—	Controller	—	—	—
259	L_D26	T18	I/O	BOUT_L_D_EN(26)	0	High Z
260	BOUT_L_D_EN(25)	—	Controller	—	—	—
261	L_D25	T19	I/O	BOUT_L_D_EN(25)	0	High Z
262	BOUT_L_D_EN(24)	—	Controller	—	—	—
263	L_D24	T20	I/O	BOUT_L_D_EN(24)	0	High Z
264	MB_D4	U1	I/O	BOUT_MB_D_En	0	High Z
265	MB_D5	U2	I/O	BOUT_MB_D_En	0	High Z
266	MB_D3	U3	I/O	BOUT_MB_D_En	0	High Z
267	VSS	U4	Linkage	—	—	—
268	VIO/µP_SELECT	U5	Input	—	—	—
269	VDD	U6	Linkage	—	—	—
270	INV_PCI_CBE3_EN_N	—	Controller	—	—	—
271	PCI_CBE3#	U7	I/O	INV_PCI_CBE3_EN_N	0	High Z
272	VSS	U8	Linkage	—	—	—
273	INV_PCI_CBE2_EN_N	—	Controller	—	—	—
274	PCI_CBE2#	U9	I/O	INV_PCI_CBE2_EN_N	0	High Z
275	VDD	U10	Linkage	—	—	—
276	INV_PCI_PAR_EN_N	—	Controller	—	—	—
277	PCI_PAR	U11	I/O	INV_PCI_PAR_EN_N	0	High Z
278	INV_PCI_CBE1_EN_N	—	Controller	—	—	—
279	PCI_CBE1#	U12	I/O	INV_PCI_CBE1_EN_N	0	High Z
280	VSS	U13	Linkage	—	—	—
281	INV_PCI_CBE0_EN_N	—	Controller	—	—	—
282	PCI_CBE0#	U14	I/O	INV_PCI_CBE0_EN_N	0	High Z
283	VDD	U15	Linkage	—	—	—
284	INV_PCI_ADEN_N(3)	—	Controller	—	—	—
285	PCI_AD3	U16	I/O	INV_PCI_ADEN_N(3)	0	High Z
286	VSS	U17	Linkage	—	—	—
287	BOUT_L_D_EN(30)	—	Controller	—	—	—
288	L_D30	U18	I/O	BOUT_L_D_EN(30)	0	High Z
289	BOUT_L_D_EN(29)	—	Controller	—	—	—
290	L_D29	U19	I/O	BOUT_L_D_EN(29)	0	High Z
291	BOUT_L_D_EN(27)	—	Controller	—	—	—

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
292	L_D27	U20	I/O	BOUT_L_D_EN(27)	0	High Z
293	MB_D1	V1	I/O	BOUT_MB_D_En	0	High Z
294	MB_D2	V2	I/O	BOUT_MB_D_En	0	High Z
295	SYSERR	V3	Output3	GENERAL_EN	0	High Z
296	INV_PCI_ADEN_N(31)	—	Controller	—	—	—
297	PCI_AD31	V4	I/O	INV_PCI_ADEN_N(31)	0	High Z
298	INV_PCI_ADEN_N(30)	—	Controller	—	—	—
299	PCI_AD30	V5	I/O	INV_PCI_ADEN_N(30)	0	High Z
300	INV_PCI_ADEN_N(27)	—	Controller	—	—	—
301	PCI_AD27	V6	I/O	INV_PCI_ADEN_N(27)	0	High Z
302	INV_PCI_ADEN_N(23)	—	Controller	—	—	—
303	PCI_AD23	V7	I/O	INV_PCI_ADEN_N(23)	0	High Z
304	INV_PCI_ADEN_N(19)	—	Controller	—	—	—
305	PCI_AD(19)	V8	I/O	INV_PCI_ADEN_N(19)	0	High Z
306	INV_PCI_ADEN_N(18)	—	Controller	—	—	—
307	PCI_AD18	V9	I/O	INV_PCI_ADEN_N(18)	0	High Z
308	PCI_LOCK#	V10	Input	INV_PCI_ADEN_N(18)		
309	INV_PCI_CTRL_EN_N	—	Controller	—	—	—
310	PCI_STOP#	V11	I/O	INV_PCI_CTRL_EN_N	0	High Z
311	INV_PCI_SERR_OUT_N	—	Controller	—	—	—
312	PCI_SERR#	V12	Output3	INV_PCI_SERR_OUT_N	0	High Z
313	INV_PCI_ADEN_N(15)	—	Controller	—	—	—
314	PCI_AD15	V13	I/O	INV_PCI_ADEN_N(15)	0	High Z
315	INV_PCI_ADEN_N(11)	—	Controller	—	—	—
316	PCI_AD11	V14	I/O	INV_PCI_ADEN_N(11)	0	High Z
317	INV_PCI_ADEN_N(7)	—	Controller	—	—	—
318	PC_AD7	V15	I/O	INV_PCI_ADEN_N(7)	0	High Z
319	INV_PCI_ADEN_N(2)	—	Controller	—	—	—
320	PCI_AD2	V16	I/O	INV_PCI_ADEN_N(2)	0	High Z
321	BOUT_FG_EN(7)	—	Controller	—	—	—
322	FG7	V17	I/O	BOUT_FG_EN(7)	0	High Z
323	BOUT_FG_EN(6)	—	Controller	—	—	—
324	FG6	V18	I/O	BOUT_FG_EN(6)	0	High Z
325	BOUT_FG_EN(5)	—	Controller	—	—	—
326	FG5	V19	I/O	BOUT_FG_EN(5)	0	High Z
327	BOUT_L_D_EN(28)	—	Controller	—	—	—
328	L_D28	V20	I/O	BOUT_L_D_EN(28)	0	High Z

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
329	MB_D0	W1	I/O	BOUT_MB_D_En	0	High Z
330	CLKERR	W2	Output3	GENERAL_EN	0	High Z
331	INV_PCI_REQ_EN_N	—	Controller	—	—	—
332	PCI_REQ#	W3	Output3	INV_PCI_REQ_EN_N	0	High Z
333	PCI_GNT#	W4	Input	INV_PCI_REQ_EN_N	—	—
334	INV_PCI_ADEN_N(29)	—	Controller	INV_PCI_REQ_EN_N	—	—
335	PCI_AD29	W5	I/O	INV_PCI_ADEN_N(29)	0	High Z
336	INV_PCI_ADEN_N(26)	—	Controller	—	—	—
337	PCI_AD26	W6	I/O	INV_PCI_ADEN_N(26)	0	High Z
338	INV_PCI_ADEN_N(22)	—	Controller	—	—	—
339	PCI_AD22	W7	I/O	INV_PCI_ADEN_N(22)	0	High Z
340	INV_PCI_ADEN_N(21)	—	Controller	—	—	—
341	PCI_AD21	W8	I/O	INV_PCI_ADEN_N(21)	0	High Z
342	INV_PCI_ADEN_N(17)	—	Controller	—	—	—
343	PCI_AD17	W9	I/O	INV_PCI_ADEN_N(17)	0	High Z
344	PCI_IDSEL#	W10	Input	—	—	—
345	PCI_DEVSEL#	W11	I/O	INV_PCI_CTRL_EN_N	0	High Z
346	INV_PCI_PERR_EN_N	—	Controller	—	—	—
347	PCI_PERR#	W12	I/O	INV_PCI_PERR_EN_N	0	High Z
348	INV_PCI_ADEN_N(14)	—	Controller	—	—	—
349	PCI_AD14	W13	I/O	INV_PCI_ADEN_N(14)	0	High Z
350	INV_PCI_ADEN_N(10)	—	Controller	—	—	—
351	PCI_AD10	W14	I/O	INV_PCI_ADEN_N(10)	0	High Z
352	INV_PCI_ADEN_N(9)	—	Controller	—	—	—
353	PCI_AD9	W15	I/O	INV_PCI_ADEN_N(9)	0	High Z
354	INV_PCI_ADEN_N(6)	—	Controller	—	—	—
355	PCI_AD6	W16	I/O	INV_PCI_ADEN_N(6)	0	High Z
356	INV_PCI_ADEN_N(1)	—	Controller	—	—	—
357	PCI_AD1	W17	I/O	INV_PCI_ADEN_N(1)	0	High Z
357	BOUT_FG_EN(4)	—	Controller	—	—	—
359	FG4	W18	I/O	BOUT_FG_EN(4)	0	High Z
360	BOUT_FG_EN(3)	—	Controller	—	—	—
361	FG3	W19	I/O	BOUT_FG_EN(3)	0	High Z
362	BOUT_FG_EN(2)	—	Controller	—	—	—
363	FG2	W20	I/O	BOUT_FG_EN(2)	0	High Z
364	RESET#	Y1	Input	BOUT_FG_EN(2)	—	50 kΩ up
365	PCI_RST#	Y2	Input	BOUT_FG_EN(2)	—	—
366	PCI_CLK	Y3	Clock	BOUT_FG_EN(2)	—	—

17 JTAG/Boundary Scan (continued)

Table 131. Boundary-Scan Register Description (continued)

Boundary-Scan Register Bit Pin	Pin Name	Ball	Enabled State	Pin Grouping	Control	Disabled State
367	INV_OUT_PCI_INTAN	—	Controller	—	—	—
368	PCI_INTA#	Y4	Output3	INV_OUT_PCI_INTAn	0	High Z
369	INV_PCI_ADEN_N(28)	—	Controller	—	—	—
370	PCI_AD28	Y5	I/O	INV_PCI_ADEN_N(28)	0	High Z
371	INV_PCI_ADEN_N(25)	—	Controller	—	—	—
372	PCI_AD25	Y6	I/O	INV_PCI_ADEN_N(25)	0	High Z
373	INV_PCI_ADEN_N(24)	—	Controller	—	—	—
374	PCI_AD24	Y7	I/O	INV_PCI_ADEN_N(24)	0	High Z
375	INV_PCI_ADEN_N(20)	—	Controller	—	—	—
376	PCI_AD20	Y8	I/O	INV_PCI_ADEN_N(20)	0	High Z
377	INV_PCI_ADEN_N(16)	—	Controller	—	—	—
378	PCI_AD16	Y9	I/O	INV_PCI_ADEN_N(16)	0	High Z
379	INV_PCI_FRAME_EN_N	—	Controller	—	—	—
380	PCI_FRAME#	Y10	I/O	INV_PCI_FRAME_EN_N	0	High Z
381	INV_PCI_IRDY_EN_N	—	Controller	—	—	—
382	PCI_IRDY#	Y11	I/O	INV_PCI_IRDY_EN_N	0	High Z
383	PCI_TRDY#	Y12	I/O	INV_PCI_CTRL_EN_N	0	High Z
384	INV_PCI_ADEN_N(13)	—	Controller	—	—	—
385	PCI_AD13	Y13	I/O	INV_PCI_ADEN_N(13)	0	High Z
386	INV_PCI_ADEN_N(12)	—	Controller	—	—	—
387	PCI_AD12	Y14	I/O	INV_PCI_ADEN_N(12)	0	High Z
388	INV_PCI_ADEN_N(8)	—	Controller	—	—	—
389	PCI_AD8	Y15	I/O	INV_PCI_ADEN_N(8)	0	High Z
390	INV_PCI_ADEN_N(5)	—	Controller	—	—	—
391	PCI_AD5	Y16	I/O	INV_PCI_ADEN_N(5)	0	High Z
392	INV_PCI_ADEN_N(4)	—	Controller	—	—	—
393	PCI_AD4	Y17	I/O	INV_PCI_ADEN_N(4)	0	High Z
394	INV_PCI_ADEN_N(0)	—	Controller	—	—	—
395	PC_AD0	Y18	I/O	INV_PCI_ADEN_N(0)	0	High Z
396	BOUT_FG_EN(1)	—	Controller	—	—	—
397	FG1	Y19	I/O	BOUT_FG_EN(1)	0	High Z
398	BOUT_FG_EN(0)	—	Controller	—	—	—
399	FG0	Y20	I/O	BOUT_FG_EN(0)	0	High Z

Appendix A. Constant and Minimum Delay Connections

A.1 Connection Definitions

A forward connection is defined as one in which the output **to** time slot has a greater value than the input **from** time-slot, or, put another way, the delta between them is positive.

A reverse connection is defined as one in which the output **to** time slot has a lesser value than the input **from** time slot, and the delta between them is negative.

For example, going from TS(1) to TS(38) is a forward connection, and the $TS\Delta$ is +37, but going from TS(38) to TS(1) is a reverse connection, with a $TS\Delta$ of –37:

where $TS\Delta = TS(\textbf{to}) - TS(\textbf{from})$.

Similarly, a delta can be introduced for streams which will have a bearing in certain exceptions (discussed later):

$STR\Delta = STR(\textbf{to}) - STR(\textbf{from})$.

There is only one combination which forms a $TS\Delta$ of +127 or –127:

$TS\Delta = TS(127) - TS(0) = +127$, and
$TS\Delta = TS(0) - TS(127) = -127$,

but there are two combinations which form $TS\Delta$s of +126 or –126:

$TS\Delta = TS(127) - TS(1) = TS(126) - TS(0) = +126$, and
$TS\Delta = TS(1) - TS(127) = TS(0) - TS(126) = -126$,

there are three combinations which yield +125 or –125, and so on.

The user can utilize the $TS\Delta$ to control the latency of the resulting connection. In some cases, the latency must be minimized. In other cases, such as a block of connections which must maintain some relative integrity while crossing a frame boundary, the required latency of some of the connections may exceed a one frame (>128 time-slots) to maintain the integrity of this virtual frame.

The device uses a control bit at each connection memory location, VFC, for controlling latency, allowing each connection to select one of two alternating data buffers.

A.2 Delay Type Definitions

Constant Delay—This is a well-defined, predictable, and linear region of latency in which the **to** time slot is at least 128 time slots after the **from** time-slot, but no more than 256 time slots after the **from** time-slot.

Mathematically, constant delay latency is described as follows*, with L denoting latency, and VFC set to the value indicated:

Forward connections, VFC = 1: $L = 128 + TS\Delta$ $(0 \le TS\Delta \le 127)$

Reverse connections, VFC = 0: $L = 256 + TS\Delta$ $(-127 \le TS\Delta \le 0)$

Example: Switching from TS(37) to TS(1) as a constant delay, the delta is –36, so FME is set to 0 and the resulting latency is 256 – 36 = 220 time slots. Thus, the connection will be made from TS(37) of frame(n) to TS(1) of frame(n + 2).

Simple summary: Use constant delay for latencies of 128 to 256 time slots,
set VFC = 1 for forward connections,
set VFC = 0 for reverse connections.

* Since $TS\Delta = TS(\textbf{to}) - TS(\textbf{from})$, the user can modify the equations to solve for either TS(**to**) or TS(**from**).

Appendix A. Constant and Minimum Delay Connections (continued)

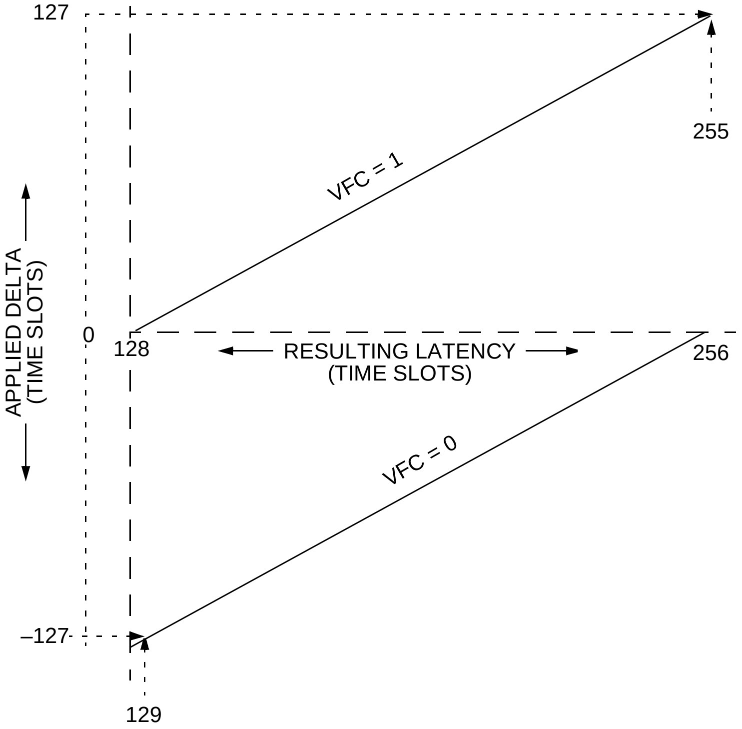


Figure 66. Constant Delay Connection Latency

Minimum Delay—This is the most common type of switching, but has a shorter range than constant delay, and the user must be aware of exceptions caused by interactions between the device's internal pipeline and the dual buffering. The **to** time slot is at least 3 time slots after the **from** time slot, but no more than 128 time slots after the **from** time slot. Exceptions exist at TSΔs of +1, +2, –126, and –127.

Forward connections, VFC = 0: L = TSΔ (3 ≤ TSΔ ≤ 127).

Reverse connections, VFC = 1: L = 128 + TSΔ (–125 ≤ TSΔ ≤ 0).

Example: Using the same switching from the example above, TS(37) to TS(1), the delta is –36, so VFC is set to 1 to effect the minimum delay (setting to 0 effects constant delay), and the resulting latency is 128 – 36 = 92 time slots. The relative positions of the end time slots are the same in both minimum and constant delay, i.e., they both switch to TS(1)], but the actual data is delayed by an additional frame in the constant delay case.

Simple summary: Use minimum delay for latencies of 3 to 128 time slots,
set VFC = 0 for forward connections,
set VFC = 1 for reverse connections.

Exceptions to minimum delay—Up until this point in the discussion, the STRΔs have not been discussed because the **to** and **from** streams have been irrelevant in the switching process.

Note: The one universally disallowed connection on the device is a TSΔ of 0 and a STRΔ of 0. This is, of course, a stream + time-slot switching to itself!

Rather than try to list the exceptions mathematically, a table is provided. The latencies in these cases may exceed two frames due to the interaction of the intrinsic pipeline delays with the double buffering.

Appendix A. Constant and Minimum Delay Connections (continued)

Table 132. Special Cases (Exceptions)

VFC Value	TSΔ	Latency for STRΔ < 0	Latency for STRΔ ≥ 0
0	+1	257	257
0	+2	258	2
1	–126	258	2
1	–127	257	257

Graphically, the minimum delay latency equations are illustrated below. The exceptions to the minimum delay have been included in the diagram, connected to the main function by dashed lines.

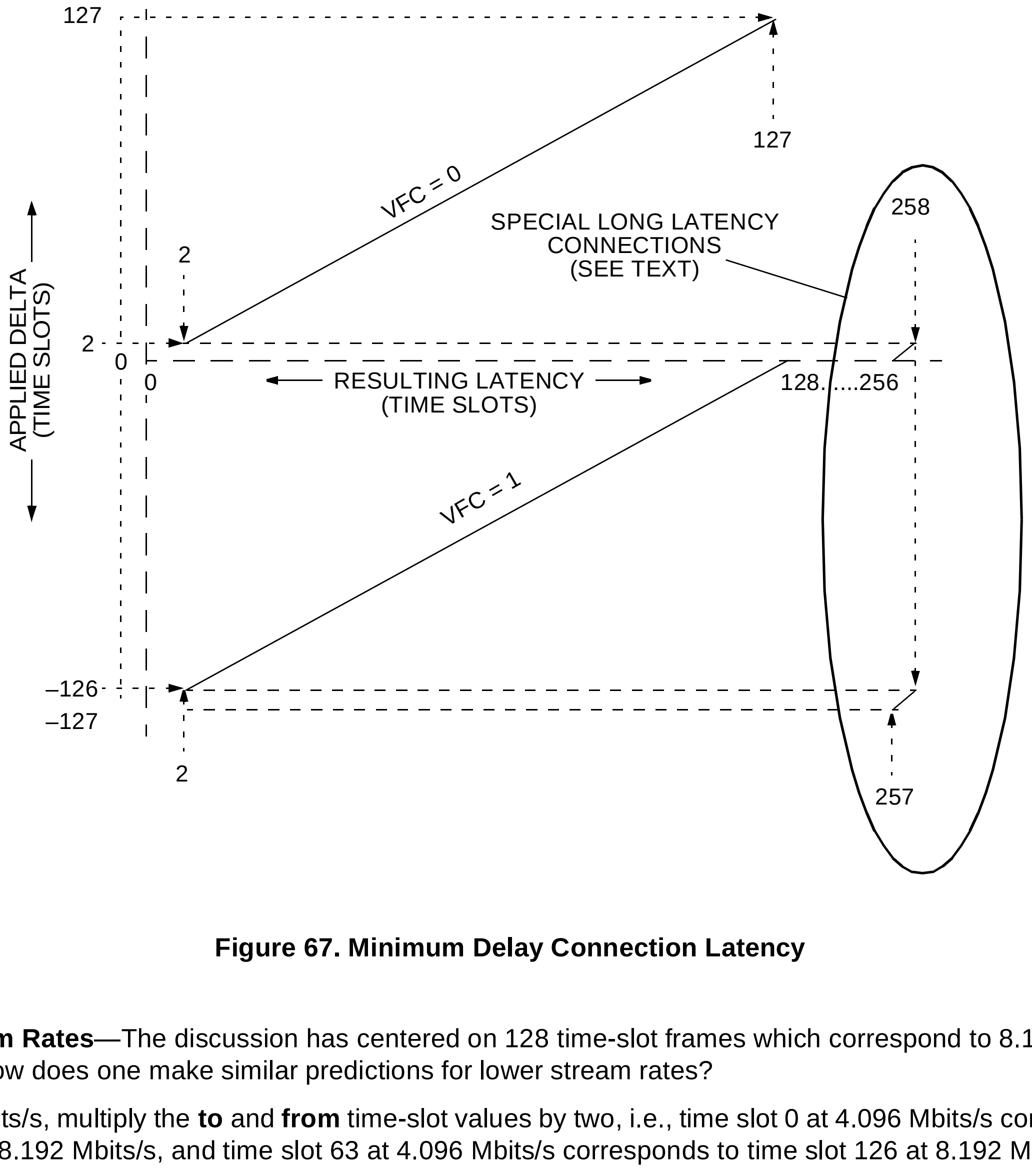


Figure 67. Minimum Delay Connection Latency

Lower Stream Rates—The discussion has centered on 128 time-slot frames which correspond to 8.192 Mbits/s data rates. How does one make similar predictions for lower stream rates?

For 4.096 Mbits/s, multiply the **to** and **from** time-slot values by two, i.e., time slot 0 at 4.096 Mbits/s corresponds to time slot 0 at 8.192 Mbits/s, and time slot 63 at 4.096 Mbits/s corresponds to time slot 126 at 8.192 Mbits/s. Similarly, multiply values by four to convert 2.048 Mbits/s values. The latency equations can then be applied directly.

Appendix B. Register Bit Field Mnemonic Summary

Key to using the table below:

- Five character alphanumeric designation
- Character 4 indicates the general register type as follows:
  - — Divide = load value for divider
  - — Enable = bit or bits to enable a function
  - — Load = load value, typically for counter
  - — Output = output only
  - — Select = bit or bits select multiple functions
- Character 5 indicates the size as follows:
  - — B = bit
  - — N = nibble
  - — P = partial register (2, 3, 5, 6, or 7 bits)
  - — R = register
- Position column identifies the bit position in the register:
  - — 0, 1, 2, 3, 4, 5, 6, 7 for bits
  - — L for lower nibble
  - — U for upper nibble
  - — n-m for bit positions in a partial register

Table 133. Mnemonic Summary, Sorted by Name

Mnemonic	Description	Type	Register	Bit Position
A1HLR	MB_CS1 address hold	Load	0x00717	—
A1LOB	APLL1 lock indicator	Output	0x00125	7
A1SLR	MB_CS1 address setup	Load	0x00713	—
A2HLR	MB_CS2 address hold	Load	0x00727	—
A2SLR	MB_CS2 address setup	Load	0x00723	—
A3HLR	MB_CS3 address hold	Load	0x00737	—
A3SLR	MB_CS3 address setup	Load	0x00733	—
A4HLR	MB_CS4 address hold	Load	0x00747	—
A4SLR	MB_CS4 address setup	Load	0x00743	—
A5HLR	MB_CS5 address hold	Load	0x00757	—
A5SLR	MB_CS5 address setup	Load	0x00753	—
A6HLR	MB_CS6 address hold	Load	0x00767	—
A6SLR	MB_CS6 address setup	Load	0x00763	—
A7HLR	MB_CS7 address hold	Load	0x00777	—
A7SLR	MB_CS7 address setup	Load	0x00773	—
ABOEN	A and B clocks output	Enable	0x00220	U
ACRSN	A clocks rate	Select	0x00223	L
AIOEB	All I/O (master)	Enable	0x00103	7
BA0LR	DT base address byte 0	Load	0x00110	—
BA1LR	DT base address byte 1	Load	0x00111	—
BA2LR	DT base address byte 2	Load	0x00112	—
BA3LR	DT base address byte 3	Load	0x00113	—
BCRSN	B clocks rate	Select	0x00223	U
C2FEB	C2 fallback trigger	Enable	0x0010A	5

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
C2LOB	C2 latched error	Output	0x00122	5
C2TOB	C2 transient error	Output	0x00120	5
C2WEB	C2 watchdog	Enable	0x0010E	5
C4FEB	/C4 fallback trigger	Enable	0x0010A	4
C4LOB	C4 latched error	Output	0x00122	4
C4TOB	C4 transient error	Output	0x00120	4
C4WEB	/C4 watchdog	Enable	0x0010E	4
CAFEB	C8A fallback trigger	Enable	0x0010A	0
CALOB	C8A latched error	Output	0x00122	0
CATOB	C8A transient error	Output	0x00120	0
CAWEB	C8A watchdog	Enable	0x0010E	0
CAWSN	C8A watchdog	Select	0x0010C	L
CBFEB	C8B fallback trigger	Enable	0x0010A	1
CBLOB	C8B latched error	Output	0x00122	1
CBTOB	C8B transient error	Output	0x00120	1
CBWEB	C8B watchdog	Enable	0x0010E	1
CBWSN	C8B watchdog	Select	0x0010C	U
CCOEN	C clocks output	Enable	0x00220	L
CCSEN	C clocks separate	Enable	0x00224	L
CFBOB	Fallback status	Output	0x00127	6
CFHLN	Diag sync-to-frame high	Load	0x0014B	L
CFLLR	Diag sync-to-frame low	Load	0x0014A	—
CFPOB	CLEAR_FALLBACK pending	Output	0x00124	1
CFSEN	Diag sync-to-frame EN	Enable	0x0014B	U
CFSOB	Failsafe status	Output	0x00127	7
CKMDR	Clock main	Divide	0x00201	—
CKMSR	Clock main	Select	0x00200	—
CKRDR	Clock resource	Divide	0x00205	—
CMFEB	/C16– fallback trigger	Enable	0x0010A	3
CMLOB	/C16– latched error	Output	0x00122	3
CMROB	Connection memory reset active	Output	0x00125	1
CMTOB	/C16– transient error	Output	0x00120	3
CMWEB	/C16– watchdog	Enable	0x0010E	3
CPFEB	/C16+ fallback trigger	Enable	0x0010A	2
CPLOB	/C16+ latched error	Output	0x00122	2
CPTOB	/C16+ transient error	Output	0x00120	2
CPWEB	/C16+ watchdog	Enable	0x0010E	2
CSASR	Clock set access	Select	0x00106	—
D1FEB	DPLL1 sync trigger	Enable	0x0010B	5
D1ISR	DPLL1 input	Select	0x0020A	—
D1LOB	DPLL1 sync latched error	Output	0x00123	5

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
D1LOP	DPLL1 lock status	Output	0x00125	5:4
D1RSR	DPLL1 rate	Select	0x0020B	—
D1TOB	DPLL1 sync transient error	Output	0x00121	5
D1WEB	DPLL1 sync watchdog	Enable	0x0010F	5
D2FEB	DPLL2 sync trigger	Enable	0x0010B	6
D2ISR	DPLL2 input	Select	0x0020E	—
D2LOB	DPLL2 sync latched error	Output	0x00123	6
D2LOP	DPLL2 lock status	Output	0x00125	3:2
D2RSR	DPLL2 rate	Select	0x0020F	—
D2TOB	DPLL2 sync transient error	Output	0x00121	6
D2WEB	DPLL2 sync watchdog	Enable	0x0010F	6
DMMSP	Data memory mode	Select	0x00105	6:0
DMPOB	Data memory PCI error status	Output	0x00127	0
DMSOB	Data memory PCI queue status	Output	0x00126	0
DMTOB	Data memory PCI timer status	Output	0x00127	3
DPGOB	Data memory active page	Output	0x00125	0
EBOLR	Diag external buffer retry	Load	0x00147	—
F0DSB	FGIO 0 R/W direction	Select	0x00482	0
F0IOB	FGIO 0 data	Load	0x00480	0
F0ISB	Frame 0 pulse inversion	Enable	0x00402	7
F0LLR	Frame 0 lower start time	Load	0x00400	—
F0MEB	FGIO 0 read mask	Enable	0x00481	0
F0RSR	Frame 0 pulse width rate	Select	0x00403	—
F0ULR	Frame 0 upper start time	Load	0x00401	—
F0WSP	Frame 0 pulse width	Select	0x00402	6:0
F1DSB	FGIO 1 R/W direction	Select	0x00482	1
F1IOB	FGIO 1 data	Load	0x00480	1
F1ISB	Frame 1 pulse inversion	Enable	0x00412	7
F1LLR	Frame 1 lower start time	Load	0x00410	—
F1MEB	FGIO 1 read mask	Enable	0x00481	1
F1RSR	Frame 1 pulse width rate	Select	0x00413	—
F1ULR	Frame 1 upper start time	Load	0x00411	—
F1WSP	Frame 1 pulse width	Select	0x00412	6:0
F2DSB	FGIO 2 R/W direction	Select	0x00482	2
F2IOB	FGIO 2 data	Load	0x00480	2
F2ISB	Frame 2 pulse inversion	Enable	0x00422	7
F2LLR	Frame 2 lower start time	Load	0x00420	—
F2MEB	FGIO 2 read mask	Enable	0x00481	2
F2RSR	Frame 2 pulse width rate	Select	0x00423	—
F2ULR	Frame 2 upper start time	Load	0x00421	—
F2WSP	Frame 2 pulse width	Select	0x00422	6:0

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
F3DSB	FGIO 3 R/W direction	Select	0x00482	3
F3IOB	FGIO 3 data	Load	0x00480	3
F3ISB	Frame 3 pulse inversion	Enable	0x00432	7
F3LLR	Frame 3 lower start time	Load	0x00430	—
F3MEB	FGIO 3 read mask	Enable	0x00481	3
F3RSR	Frame 3 pulse width rate	Select	0x00433	—
F3ULR	Frame 3 upper start time	Load	0x00431	—
F3WSP	Frame 3 pulse width	Select	0x00432	6:0
F4DSB	FGIO 4 R/W direction	Select	0x00482	4
F4IOB	FGIO 4 data	Load	0x00480	4
F4ISB	Frame 4 pulse inversion	Enable	0x00442	7
F4LLR	Frame 4 lower start time	Load	0x00440	—
F4MEB	FGIO 4 read mask	Enable	0x00481	4
F4RSR	Frame 4 pulse width rate	Select	0x00443	—
F4ULR	Frame 4 upper start time	Load	0x00441	—
F4WSP	Frame 4 pulse width	Select	0x00442	6:0
F5DSB	FGIO 5 R/W direction	Select	0x00482	5
F5IOB	FGIO 5 data	Load	0x00480	5
F5ISB	Frame 5 pulse inversion	Enable	0x00452	7
F5LLR	Frame 5 lower start time	Load	0x00450	—
F5MEB	FGIO 5 read mask	Enable	0x00481	5
F5RSR	Frame 5 pulse width rate	Select	0x00453	—
F5ULR	Frame 5 upper start time	Load	0x00451	—
F5WSP	Frame 5 pulse width	Select	0x00452	6:0
F6DSB	FGIO 6 R/W direction	Select	0x00482	6
F6IOB	FGIO 6 data	Load	0x00480	6
F6ISB	Frame 6 pulse inversion	Enable	0x00462	7
F6LLR	Frame 6 lower start time	Load	0x00460	—
F6MEB	FGIO 6 read mask	Enable	0x00481	6
F6RSR	Frame 6 pulse width rate	Select	0x00463	—
F6ULR	Frame 6 upper start time	Load	0x00461	—
F6WSP	Frame 6 pulse width	Select	0x00462	6:0
F7DSB	FGIO 7 R/W direction	Select	0x00482	7
F7IOB	FGIO 7 data	Load	0x00480	7
F7ISB	Frame 7 pulse inversion	Enable	0x00472	7
F7LLR	Frame 7 lower start time	Load	0x00470	—
F7MEB	FGIO 7 read mask	Enable	0x00481	7
F7MSR	Frame 7 mode	Select	0x00476	—
F7RSR	Frame 7 pulse width rate	Select	0x00473	—
F7SSP	FG7 timer pulse shape	Select	0x00477	—
F7ULR	Frame 7 upper start time	Load	0x00471	—

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
F7WSN	FG7 timer pulse width	Select	0x00477	L
F7WSP	Frame 7 pulse width	Select	0x00472	6:0
FAFEB	/FRAMEA fallback trigger	Enable	0x0010B	0
FALOB	/FRAMEA latched error	Output	0x00123	0
FATOB	/FRAMEA transient error	Output	0x00121	0
FAWEB	/FRAMEA watchdog	Enable	0x0010F	0
FB1SB	APLL1 feedback reset	Select	0x00146	1
FB2SB	APLL2 feedback reset	Select	0x00146	2
FBCSR	Fallback control	Select	0x00108	—
FBFEB	/FRAMEB fallback trigger	Enable	0x0010B	1
FBFOB	Fallback enable status	Output	0x00124	7
FBLOB	/FRAMEB latched error	Output	0x00123	1
FBSOP	Fallback states	Output	0x00124	6:4
FBTOB	/FRAMEB transient error	Output	0x00121	1
FBWEB	/FRAMEB watchdog	Enable	0x0010F	1
FCFEB	/FR_COMP fallback trigger	Enable	0x0010B	2
FCISB	FG7 timer invert output	Select	0x00477	7
FCLLR	Frame group 7 lower count	Load	0x00474	—
FCLOB	/FR_COMP latched error	Output	0x00123	2
FCTOB	/FR_COMP transient error	Output	0x00121	2
FCULR	Frame group 7 upper count	Load	0x00475	—
FCWEB	/FR_COMP watchdog	Enable	0x0010F	2
FFPOB	FORCE_FALLBACK pending	Output	0x00124	0
FGREB	Frame group	Enable	0x00103	5
FPASR	Frame phase alignment	Select	0x00107	—
FRMSB	Diag /FR_COMP input	Select	0x00145	4
FRSEN	/FR_COMP separate	Enable	0x00224	U
FRWSR	/FR_COMP width	Select	0x00222	—
FSCSR	Failsafe return command	Select	0x00114	—
FSEER	Failsafe enable	Enable	0x00115	—
FSLOB	Failsafe latched error	Output	0x00123	7
FSMSN	Fallback secondary mode	Select	0x00109	L
FSSSR	Failsafe sensitivity	Select	0x00116	—
FSTOB	Failsafe transient error	Output	0x00121	7
FSWEB	Failsafe watchdog	Enable	0x0010F	7
FT0EB	FG0 test-point	Enable	0x00140	0
FT1EB	FG1 test-point	Enable	0x00140	1
FT2EB	FG2 test-point	Enable	0x00140	2
FT3EB	FG3 test-point	Enable	0x00140	3
FT4EB	FG4 test-point	Enable	0x00140	4
FT5EB	FG5 test-point	Enable	0x00140	5

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
FT6EB	FG6 test-point	Enable	0x00140	6
FT7EB	FG7 test-point	Enable	0x00140	7
FTPSR	FG test-point MUX	Select	0x00141	—
FTRSN	Fallback type	Select	0x00109	U
G0DSB	GPIO 0 R/W direction	Select	0x00502	0
G0IOB	GPIO 0 data	Load	0x00500	0
G0MEB	GPIO 0 read mask	Enable	0x00501	0
G0OEB	GPIO 0 override	Enable	0x00503	0
G1DSB	GPIO 1 R/W direction	Select	0x00502	1
G1IOB	GPIO 1 data	Load	0x00500	1
G1MEB	GPIO 1 read mask	Enable	0x00501	1
G1OEB	GPIO 0 override	Enable	0x00503	1
G2DSB	GPIO 2 R/W direction	Select	0x00502	2
G2IOB	GPIO 2 data	Load	0x00500	2
G2MEB	GPIO 2 read mask	Enable	0x00501	2
G2OEB	GPIO 2 override	Enable	0x00503	2
G3DSB	GPIO 3 R/W direction	Select	0x00502	3
G3IOB	GPIO 3 data	Load	0x00500	3
G3MEB	GPIO 3 read mask	Enable	0x00501	3
G4DSB	GPIO 4 R/W direction	Select	0x00502	4
G4IOB	GPIO 4 data	Load	0x00500	4
G4MEB	GPIO 4 read mask	Enable	0x00501	4
G5DSB	GPIO 5 R/W direction	Select	0x00502	5
G5IOB	GPIO 5 data	Load	0x00500	5
G5MEB	GPIO 5 read mask	Enable	0x00501	5
G6DSB	GPIO 6 R/W direction	Select	0x00502	6
G6IOB	GPIO 6 data	Load	0x00500	6
G6MEB	GPIO 6 read mask	Enable	0x00501	6
G7DSB	GPIO 7 R/W direction	Select	0x00502	7
G7IOB	GPIO 7 data	Load	0x00500	7
G7MEB	GPIO 7 read mask	Enable	0x00501	7
GOPOB	Go clocks pending	Output	0x00124	2
GPIEB	General purpose I/O	Enable	0x00103	4
GSREB	Global subrate	Enable	0x00105	7
GT0EB	GP0 test-point	Enable	0x00142	0
GT1EB	GP1 test-point	Enable	0x00142	1
GT2EB	GP2 test-point	Enable	0x00142	2
GT3EB	GP3 test-point	Enable	0x00142	3
GT4EB	GP4 test-point	Enable	0x00142	4
GT5EB	GP5 test-point	Enable	0x00142	5
GT6EB	GP6 test-point	Enable	0x00142	6

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
GT7EB	GP7 test-point	Enable	0x00142	7
GTPSR	GP test-point MUX	Select	0x00143	—
HARSN	H1x0 group A rate	Select	0x00300	L
HBRSN	H1x0 group B rate	Select	0x00300	U
HCKEB	H1x0 clocks	Enable	0x00103	3
HCRSN	H1x0 group C rate	Select	0x00301	L
HDBEB	H1x0 data bus	Enable	0x00103	2
HDRSN	H1x0 group D rate	Select	0x00301	U
HERSN	H1x0 group E rate	Select	0x00302	L
HFRSN	H1x0 group F rate	Select	0x00302	U
HGRSN	H1x0 group G rate	Select	0x00303	L
HHRSN	H1x0 group H rate	Select	0x00303	U
HRBEB	Hard reset of back end	Enable	0x00101	1
IASLR	Diag, SYSERR assertion	Load	0x00149	—
IC0SB	Invert MB CS0 strobe	Select	0x00780	0
IC1SB	Invert MB CS1 strobe	Select	0x00780	1
IC2SB	Invert MB CS2 strobe	Select	0x00780	2
IC3SB	Invert MB CS3 strobe	Select	0x00780	3
IC4SB	Invert MB CS4 strobe	Select	0x00780	4
IC5SB	Invert MB CS5 strobe	Select	0x00780	5
IC6SB	Invert MB CS6 strobe	Select	0x00780	6
IC7SB	Invert MB CS7 strobe	Select	0x00780	7
ICDSP	Diag, internal control mode	Select	0x00148	7:6
ICKLP	Diag, internal control CLKERR	Load	0x00148	5:4
ICMSB	Invert clock main	Select	0x00204	4
IDHOR	Device ID high	Output	0x0012B	—
IDLOR	Device ID low	Output	0x0012A	—
IEXLP	Diag, internal control EXTERR	Load	0x00148	1:0
IF0SB	Invert interrupt FGIO 0	Select	0x00603	0
IF1SB	Invert interrupt FGIO 1	Select	0x00603	1
IF2SB	Invert interrupt FGIO 2	Select	0x00603	2
IF3SB	Invert interrupt FGIO 3	Select	0x00603	3
IF4SB	Invert interrupt FGIO 4	Select	0x00603	4
IF5SB	Invert interrupt FGIO 5	Select	0x00603	5
IF6SB	Invert interrupt FGIO 6	Select	0x00603	6
IF7SB	Invert interrupt FGIO 7	Select	0x00603	7
IG0SB	Invert interrupt GPIO 0	Select	0x00607	0
IG1SB	Invert interrupt GPIO 1	Select	0x00607	1
IG2SB	Invert interrupt GPIO 2	Select	0x00607	2
IG3SB	Invert interrupt GPIO 3	Select	0x00607	3
IG4SB	Invert interrupt GPIO 4	Select	0x00607	4

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
IG5SB	Invert interrupt GPIO 5	Select	0x00607	5
IG6SB	Invert interrupt GPIO 6	Select	0x00607	6
IG7SB	Invert interrupt GPIO 7	Select	0x00607	7
IMRSB	Invert MB read strobe	Select	0x00781	1
IMWSB	Invert MB write strobe	Select	0x00781	0
IPRSB	Invert MB PCIRSTn, forwarded	Select	0x00781	2
IR0SB	Invert local reference 0	Select	0x0020C	0
IR1SB	Invert local reference 1	Select	0x0020C	1
IR2SB	Invert local reference 2	Select	0x0020C	2
IR3SB	Invert local reference 3	Select	0x0020C	3
IR4SB	Invert local reference 4	Select	0x0020C	4
IR5SB	Invert local reference 5	Select	0x0020C	5
IR6SB	Invert local reference 6	Select	0x0020C	6
IR7SB	Invert local reference 7	Select	0x0020C	7
ISYLP	Diag, internal control SYSERR	Load	0x00148	3:2
JAMSR	Interrupt arbitration mode	Select	0x00610	—
JC0EB	Interrupt from CLKERR 0	Enable	0x0060E	0
JC0OB	Interrupt pending CLKERR 0	Output	0x0060C	0
JC1EB	Interrupt from CLKERR 1	Enable	0x0060E	1
JC1OB	Interrupt pending CLKERR 1	Output	0x0060C	1
JC2EB	Interrupt from CLKERR 2	Enable	0x0060E	2
JC2OB	Interrupt pending CLKERR 2	Output	0x0060C	2
JC3EB	Interrupt from CLKERR 3	Enable	0x0060E	3
JC3OB	Interrupt pending CLKERR 3	Output	0x0060C	3
JC4EB	Interrupt from CLKERR 4	Enable	0x0060E	4
JC4OB	Interrupt pending CLKERR 4	Output	0x0060C	4
JC5EB	Interrupt from CLKERR 5	Enable	0x0060E	5
JC5OB	Interrupt pending CLKERR 5	Output	0x0060C	5
JC6EB	Interrupt from CLKERR 6	Enable	0x0060E	6
JC6OB	Interrupt pending CLKERR 6	Output	0x0060C	6
JC7EB	Interrupt from CLKERR 7	Enable	0x0060E	7
JC7OB	Interrupt pending CLKERR 7	Output	0x0060C	7
JC8EB	Interrupt from CLKERR 8	Enable	0x0060F	0
JC8OB	Interrupt pending CLKERR 8	Output	0x0060D	0
JC9EB	Interrupt from CLKERR 9	Enable	0x0060F	1
JC9OB	Interrupt pending CLKERR 9	Output	0x0060D	1
JCAEB	Interrupt from CLKERR A	Enable	0x0060F	2
JCAOB	Interrupt pending CLKERR A	Output	0x0060D	2
JCBEB	Interrupt from CLKERR B	Enable	0x0060F	3
JCBOB	Interrupt pending CLKERR B	Output	0x0060D	3
JCCEB	Interrupt from CLKERR C	Enable	0x0060F	4

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
JCCOB	Interrupt pending CLKERR C	Output	0x0060D	4
JCDEB	Interrupt from CLKERR D	Enable	0x0060F	5
JCDOB	Interrupt pending CLKERR D	Output	0x0060D	5
JCEEB	Interrupt from CLKERR E	Enable	0x0060F	6
JCEOB	Interrupt pending CLKERR E	Output	0x0060D	6
JCFEB	Interrupt from CLKERR F	Enable	0x0060F	7
JCFOB	Interrupt pending CLKERR F	Output	0x0060D	7
JCOSR	Interrupt CLKERR output mode	Select	0x00613	—
JCWSR	Interrupt CLKERR pulse width	Select	0x00617	—
JF0EB	Interrupt from FGIO 0	Enable	0x00601	0
JF0OB	Interrupt pending FGIO 0	Output	0x00600	0
JF1EB	Interrupt from FGIO 1	Enable	0x00601	1
JF1OB	Interrupt pending FGIO 1	Output	0x00600	1
JF2EB	Interrupt from FGIO 2	Enable	0x00601	2
JF2OB	Interrupt pending FGIO 2	Output	0x00600	2
JF3EB	Interrupt from FGIO 3	Enable	0x00601	3
JF3OB	Interrupt pending FGIO 3	Output	0x00600	3
JF4EB	Interrupt from FGIO 4	Enable	0x00601	4
JF4OB	Interrupt pending FGIO 4	Output	0x00600	4
JF5EB	Interrupt from FGIO 5	Enable	0x00601	5
JF5OB	Interrupt pending FGIO 5	Output	0x00600	5
JF6EB	Interrupt from FGIO 6	Enable	0x00601	6
JF6OB	Interrupt pending FGIO 6	Output	0x00600	6
JF7EB	Interrupt from FGIO 7	Enable	0x00601	7
JF7OB	Interrupt pending FGIO 7	Output	0x00600	7
JG0EB	Interrupt from GPIO 0	Enable	0x00605	0
JG0OB	Interrupt pending GPIO 0	Output	0x00604	0
JG1EB	Interrupt from GPIO 1	Enable	0x00605	1
JG1OB	Interrupt pending GPIO 1	Output	0x00604	1
JG2EB	Interrupt from GPIO 2	Enable	0x00605	2
JG2OB	Interrupt pending GPIO 2	Output	0x00604	2
JG3EB	Interrupt from GPIO 3	Enable	0x00605	3
JG3OB	Interrupt pending GPIO 3	Output	0x00604	3
JG4EB	Interrupt from GPIO 4	Enable	0x00605	4
JG4OB	Interrupt pending GPIO 4	Output	0x00604	4
JG5EB	Interrupt from GPIO 5	Enable	0x00605	5
JG5OB	Interrupt pending GPIO 5	Output	0x00604	5
JG6EB	Interrupt from GPIO 6	Enable	0x00605	6
JG6OB	Interrupt pending GPIO 6	Output	0x00604	6
JG7EB	Interrupt from GPIO 7	Enable	0x00605	7
JG7OB	Interrupt pending GPIO 7	Output	0x00604	7

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
JISOR	Interrupt in-service	Output	0x006FC	—
JS0EB	Interrupt from SYSERR 0	Enable	0x0060A	0
JS0OB	Interrupt pending SYSERR 0	Output	0x00608	0
JS1EB	Interrupt from SYSERR 1	Enable	0x0060A	1
JS1OB	Interrupt pending SYSERR 1	Output	0x00608	1
JS2EB	Interrupt from SYSERR 2	Enable	0x0060A	2
JS2OB	Interrupt pending SYSERR 2	Output	0x00608	2
JS3EB	Interrupt from SYSERR 3	Enable	0x0060A	3
JS3OB	Interrupt pending SYSERR 3	Output	0x00608	3
JS4EB	Interrupt from SYSERR 4	Enable	0x0060A	4
JS4OB	Interrupt pending SYSERR 4	Output	0x00608	4
JS5EB	Interrupt from SYSERR 5	Enable	0x0060A	5
JS5OB	Interrupt pending SYSERR 5	Output	0x00608	5
JS6EB	Interrupt from SYSERR 6	Enable	0x0060A	6
JS6OB	Interrupt pending SYSERR 6	Output	0x00608	6
JS7EB	Interrupt from SYSERR 7	Enable	0x0060A	7
JS7OB	Interrupt pending SYSERR 7	Output	0x00608	7
JS8EB	Interrupt from SYSERR 8	Enable	0x0060B	0
JS8OB	Interrupt pending SYSERR 8	Output	0x00609	0
JS9EB	Interrupt from SYSERR 9	Enable	0x0060B	1
JS9OB	Interrupt pending SYSERR 9	Output	0x00609	1
JSAEB	Interrupt from SYSERR A	Enable	0x0060B	2
JSAOB	Interrupt pending SYSERR A	Output	0x00609	2
JSBEB	Interrupt from SYSERR B	Enable	0x0060B	3
JSBOB	Interrupt pending SYSERR B	Output	0x00609	3
JSCEB	Interrupt from SYSERR C	Enable	0x0060B	4
JSCOB	Interrupt pending SYSERR C	Output	0x00609	4
JSDEB	Interrupt from SYSERR D	Enable	0x0060B	5
JSDOB	Interrupt pending SYSERR D	Output	0x00609	5
JSEEB	Interrupt from SYSERR E	Enable	0x0060B	6
JSEOB	Interrupt pending SYSERR E	Output	0x00609	6
JSFEB	Interrupt from SYSERR F	Enable	0x0060B	7
JSFOB	Interrupt pending SYSERR F	Output	0x00609	7
JSOSR	Interrupt SYSERR output mode	Select	0x00612	—
JSPSR	Interrupt SYSERR-to-PCI_INTA#	Select	0x00611	—
JSWSR	Interrupt SYSERR pulse width	Select	0x00616	—
JVHOB	Interrupt in-service VC ID high	Output	0x006FF	0
JVLOR	Interrupt in-service VC ID low	Output	0x006FE	—
LARSN	Local group A rate	Select	0x00320	L
LBRSN	Local group B rate	Select	0x00320	U
LC0SR	Local clock 0 output	Select	0x00228	—

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
LC1SR	Local clock 1 output	Select	0x00229	—
LC2SR	Local clock 2 output	Select	0x0022A	—
LC3SR	Local clock 3 output	Select	0x0022B	—
LCKEB	Local clocks	Enable	0x00103	1
LCRSN	Local group C rate	Select	0x00321	L
LDBEB	Local data bus	Enable	0x00103	0
LDRSN	Local group D rate	Select	0x00321	U
LERSN	Local group E rate	Select	0x00322	L
LFRSN	Local group F rate	Select	0x00322	U
LGRSN	Local group G rate	Select	0x00323	L
LHRSN	Local group H rate	Select	0x00323	U
LRISR	Local reference input	Select	0x00208	—
MBIEB	Diag MB microprocessor access	Enable	0x00146	4
MBPOB	MB PCI error status	Output	0x00127	2
MBREB	Minibridge	Enable	0x00103	6
MBSOB	MB PCI queue status	Output	0x00126	2
MBTOB	MB PCI timer status	Output	0x00127	5
N1DSB	NR1 divider inversion	Select	0x00204	1
N1DSN	NETREF1 divider input	Select	0x00210	U
N1FEB	NETREF1 fallback trigger	Enable	0x0010B	3
N1ISN	NETREF1 main input	Select	0x00210	L
N1LOB	NETREF1 latched error	Output	0x00123	3
N1LSR	NETREF1 local reference	Select	0x00212	—
N1OEN	NETREF1 output	Enable	0x00221	L
N1SSB	NR1 selector inversion	Select	0x00204	0
N1TOB	NETREF1 transient error	Output	0x00121	3
N1WEB	NETREF1 watchdog	Enable	0x0010F	3
N1WSN	NETREF1 watchdog	Select	0x0010D	L
N2DSB	NR2 divider inversion	Select	0x00204	3
N2DSN	NETREF1 divider input	Select	0x00214	U
N2FEB	NETREF2 fallback trigger	Enable	0x0010B	4
N2ISN	NETREF1 main input	Select	0x00214	L
N2LOB	NETREF2 latched error	Output	0x00123	4
N2LSR	NETREF1 local reference	Select	0x00216	—
N2OEN	NETREF1 output	Enable	0x00221	U
N2SSB	NR2 selector inversion	Select	0x00204	2
N2TOB	NETREF2 transient error	Output	0x00121	4
N2WEB	NETREF2 watchdog	Enable	0x0010F	4
N2WSN	NETREF2 watchdog	Select	0x0010D	U
NQOOB	NOTIFY_QUEUE overflow error	Output	0x00126	0
NR1DR	NETREF1	Divide	0x00211	—

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
NR2DR	NETREF2	Divide	0x00215	—
OLHLR	Out-of-lock threshold, high	Load	0x00119	—
OLLLR	Out-of-lock threshold, low	Load	0x00118	—
OOLER	Out-of-lock monitor	Enable	0x0011A	—
OOLOB	Out-of-lock status	Output	0x00125	6
P1ISR	APLL1 input	Select	0x00202	—
P1RSR	APLL1 rate	Select	0x00203	—
P2RSR	APLL2 rate	Select	0x00207	—
PAFSR	Phase align frame	Enable	0x00107	—
PDTSB	Diag PCI discard timer	Select	0x00146	5
PMBEB	PCI reset to minibridge	Enable	0x00101	3
PMEOB	PCI master stall error	Output	0x00126	5
PMFOB	PCI master fatal error	Output	0x00126	7
PMIOB	PCI master initial warning	Output	0x00126	2
PMLOB	PCI master lock error	Output	0x00126	6
PMOOB	PCI master overwrite	Output	0x00126	3
PMWOB	PCI master stall warning	Output	0x00126	4
PRBEB	PCI reset of back end	Enable	0x00101	2
R0HLR	MB_CS0 read cycle hold	Load	0x00702	—
R0SLR	MB_CS0 read cycle setup	Load	0x00700	—
R0WLR	MB_CS0 read cycle width	Load	0x00701	—
R1HLR	MB_CS1 read cycle hold	Load	0x00712	—
R1SLR	MB_CS1 read cycle setup	Load	0x00710	—
R1WLR	MB_CS1 read cycle width	Load	0x00711	—
R2HLR	MB_CS2 read cycle hold	Load	0x00722	—
R2SLR	MB_CS2 read cycle setup	Load	0x00720	—
R2WLR	MB_CS2 read cycle width	Load	0x00721	—
R3HLR	MB_CS3 read cycle hold	Load	0x00732	—
R3SLR	MB_CS3 read cycle setup	Load	0x00730	—
R3WLR	MB_CS3 read cycle width	Load	0x00731	—
R4HLR	MB_CS4 read cycle hold	Load	0x00742	—
R4SLR	MB_CS4 read cycle setup	Load	0x00740	—
R4WLR	MB_CS4 read cycle width	Load	0x00741	—
R5HLR	MB_CS5 read cycle hold	Load	0x00752	—
R5SLR	MB_CS5 read cycle setup	Load	0x00750	—
R5WLR	MB_CS5 read cycle width	Load	0x00751	—
R6HLR	MB_CS6 read cycle hold	Load	0x00762	—
R6SLR	MB_CS6 read cycle setup	Load	0x00760	—
R6WLR	MB_CS6 read cycle width	Load	0x00761	—
R7HLR	MB_CS7 read cycle hold	Load	0x00772	—
R7SLR	MB_CS7 read cycle setup	Load	0x00770	—

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
R7WLR	MB_CS7 read cycle width	Load	0x00771	—
S2FEB	/SCLKx2 fallback trigger	Enable	0x0010A	7
S2LOB	/SCLKx2 latched error	Output	0x00122	7
S2TOB	/SCLKx2 transient error	Output	0x00120	7
S2WEB	/SCLKx2 watchdog	Enable	0x0010E	7
SCFEB	SCLK fallback trigger	Enable	0x0010A	6
SCLOB	SCLK latched error	Output	0x00122	6
SCLSB	Diag state counter mode EN	Select	0x00145	3
SCMLR	Diag state counter mode low	Load	0x00144	—
SCMSB	Diag state counter carry	Select	0x00145	5
SCRSR	SCLK/SCLKx2 rate	Select	0X00227	—
SCTOB	SCLK transient error	Output	0x00120	6
SCULP	Diag state counter mode high	Load	0x00145	2:0
SCWEB	SCLK watchdog	Enable	0x0010E	6
SRBEB	Soft reset of back end	Enable	0x00101	0
SRESR	Soft reset	Select	0x00100	—
TCOSR	T clock output	Select	0x00226	—
VCEON	VC enable status	Output	0x0012D	L
VCMEB	Diag VC microprocessor access	Enable	0x00146	3
VCOOB	VC memory overflow warning	Output	0x00126	1
VCPOB	VC memory PCI error	Output	0x00127	1
VCSOB	VC memory PCI queue	Output	0x0012C	1
VCSSR	VC start command reg	Select	0x00104	—
VCTOB	VC memory PCI timer	Output	0x00127	4
VEROR	Version ID register	Output	0x00128	—
VPPOB	VC pause pending	Output	0x0012D	5
VSPOB	VC start pending	Output	0x0012D	4
W0HLR	MB_CS0 write cycle hold	Load	0x00706	—
W0SLR	MB_CS0 write cycle setup	Load	0x00704	—
W0WLR	MB_CS0 write cycle width	Load	0x00705	—
W1HLR	MB_CS1 write cycle hold	Load	0x00716	—
W1SLR	MB_CS1 write cycle setup	Load	0x00714	—
W1WLR	MB_CS1 write cycle width	Load	0x00715	—
W2HLR	MB_CS2 write cycle hold	Load	0x00726	—
W2SLR	MB_CS2 write cycle setup	Load	0x00724	—
W2WLR	MB_CS2 write cycle width	Load	0x00725	—
W3HLR	MB_CS3 write cycle hold	Load	0x00736	—
W3SLR	MB_CS3 write cycle setup	Load	0x00734	—
W3WLR	MB_CS3 write cycle width	Load	0x00735	—
W4HLR	MB_CS4 write cycle hold	Load	0x00746	—
W4SLR	MB_CS4 write cycle setup	Load	0x00744	—

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 133. Mnemonic Summary, Sorted by Name (continued)

Mnemonic	Description	Type	Register	Bit Position
W4WLR	MB_CS4 write cycle width	Load	0x00745	—
W5HLR	MB_CS5 write cycle hold	Load	0x00756	—
W5SLR	MB_CS5 write cycle setup	Load	0x00754	—
W5WLR	MB_CS5 write cycle width	Load	0x00755	—
W6HLR	MB_CS6 write cycle hold	Load	0x00766	—
W6SLR	MB_CS6 write cycle setup	Load	0x00764	—
W6WLR	MB_CS6 write cycle width	Load	0x00765	—
W7HLR	MB_CS7 write cycle hold	Load	0x00776	—
W7SLR	MB_CS7 write cycle setup	Load	0x00774	—
W7WLR	MB_CS7 write cycle width	Load	0x00775	—
XYSOB	Active clock set	Output	0x00124	3

Table 134. Mnemonic Summary, Sorted by Register

Mnemonic	Description	Type	Register	Bit Position
SRESR	Soft reset	Select	0x00100	—
SRBEB	Soft reset of back end	Enable	0x00101	0
HRBEB	Hard reset of back end	Enable	0x00101	1
PRBEB	PCI reset of back end	Enable	0x00101	2
PMBEB	PCI reset to minibridge	Enable	0x00101	3
LDBEB	Local data bus	Enable	0x00103	0
LCKEB	Local clocks	Enable	0x00103	1
HDBEB	H1x0 data bus	Enable	0x00103	2
HCKEB	H1x0 clocks	Enable	0x00103	3
GPIEB	General-purpose I/O	Enable	0x00103	4
FGREB	Frame group	Enable	0x00103	5
MBREB	Minibridge	Enable	0x00103	6
AIOEB	All I/O (master)	Enable	0x00103	7
VCSSR	VC start command reg	Select	0x00104	—
GSREB	Global subrate	Enable	0x00105	7
DMMSP	Data memory mode	Select	0x00105	6:0
CSASR	Clock set access	Select	0x00106	—
FPASR	Frame phase alignment	Select	0x00107	—
PAFSR	Phase align frame	Enable	0x00107	—
FBCSR	Fallback control	Select	0x00108	—
FSMSN	Fallback secondary mode	Select	0x00109	L
FTRSN	Fallback type	Select	0x00109	U
CAFEB	C8A fallback trigger	Enable	0x0010A	0
CBFEB	C8B fallback trigger	Enable	0x0010A	1
CPFEB	/C16+ fallback trigger	Enable	0x0010A	2
CMFEB	/C16– fallback trigger	Enable	0x0010A	3

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
C4FEB	/C4 fallback trigger	Enable	0x0010A	4
C2FEB	C2 fallback trigger	Enable	0x0010A	5
SCFEB	SCLK fallback trigger	Enable	0x0010A	6
S2FEB	/SCLKx2 fallback trigger	Enable	0x0010A	7
FAFEB	/FRAMEA fallback trigger	Enable	0x0010B	0
FBFEB	/FRAMEB fallback trigger	Enable	0x0010B	1
FCFEB	/FR_COMP fallback trigger	Enable	0x0010B	2
N1FEB	NETREF1 fallback trigger	Enable	0x0010B	3
N2FEB	NETREF2 fallback trigger	Enable	0x0010B	4
D1FEB	DPLL1 sync trigger	Enable	0x0010B	5
D2FEB	DPLL2 sync trigger	Enable	0x0010B	6
CAWSN	C8A watchdog	Select	0x0010C	L
CBWSN	C8B watchdog	Select	0x0010C	U
N1WSN	NETREF1 watchdog	Select	0x0010D	L
N2WSN	NETREF2 watchdog	Select	0x0010D	U
CAWEB	C8A watchdog	Enable	0x0010E	0
CBWEB	C8B watchdog	Enable	0x0010E	1
CPWEB	/C16+ watchdog	Enable	0x0010E	2
CMWEB	/C16– watchdog	Enable	0x0010E	3
C4WEB	/C4 watchdog	Enable	0x0010E	4
C2WEB	C2 watchdog	Enable	0x0010E	5
SCWEB	SCLK watchdog	Enable	0x0010E	6
S2WEB	/SCLKx2 watchdog	Enable	0x0010E	7
FAWEB	/FRAMEA watchdog	Enable	0x0010F	0
FBWEB	/FRAMEB watchdog	Enable	0x0010F	1
FCWEB	/FR_COMP watchdog	Enable	0x0010F	2
N1WEB	NETREF1 watchdog	Enable	0x0010F	3
N2WEB	NETREF2 watchdog	Enable	0x0010F	4
D1WEB	DPLL1 sync watchdog	Enable	0x0010F	5
D2WEB	DPLL2 sync watchdog	Enable	0x0010F	6
FSWEB	Failsafe watchdog	Enable	0x0010F	7
BA0LR	DT base address byte 0	Load	0x00110	—
BA1LR	DT base address byte 1	Load	0x00111	—
BA2LR	DT base address byte 2	Load	0x00112	—
BA3LR	DT base address byte 3	Load	0x00113	—
FSCSR	Failsafe return command	Select	0x00114	—
FSEER	Failsafe enable	Enable	0x00115	—
FSSSR	Failsafe sensitivity	Select	0x00116	—
OLLLR	Out-of-lock threshold, low	Load	0x00118	—
OLHLR	Out-of-lock threshold, high	Load	0x00119	—
OOLER	Out-of-lock monitor	Enable	0x0011A	—

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
CATOB	C8A transient error	Output	0x00120	0
CBTOB	C8B transient error	Output	0x00120	1
CPTOB	/C16+ transient error	Output	0x00120	2
CMTOB	/C16– transient error	Output	0x00120	3
C4TOB	C4 transient error	Output	0x00120	4
C2TOB	C2 transient error	Output	0x00120	5
SCTOB	SCLK transient error	Output	0x00120	6
S2TOB	/SCLKx2 transient error	Output	0x00120	7
FATOB	/FRAMEA transient error	Output	0x00121	0
FBTOB	/FRAMEB transient error	Output	0x00121	1
FCTOB	/FR_COMP transient error	Output	0x00121	2
N1TOB	NETREF1 transient error	Output	0x00121	3
N2TOB	NETREF2 transient error	Output	0x00121	4
D1TOB	DPLL1 sync transient error	Output	0x00121	5
D2TOB	DPLL2 sync transient error	Output	0x00121	6
FSTOB	Failsafe transient error	Output	0x00121	7
CALOB	C8A latched error	Output	0x00122	0
CBLOB	C8B latched error	Output	0x00122	1
CPLOB	/C16+ latched error	Output	0x00122	2
CMLOB	/C16– latched error	Output	0x00122	3
C4LOB	C4 latched error	Output	0x00122	4
C2LOB	C2 latched error	Output	0x00122	5
SCLOB	SCLK latched error	Output	0x00122	6
S2LOB	/SCLKx2 latched error	Output	0x00122	7
FALOB	/FRAMEA latched error	Output	0x00123	0
FBLOB	/FRAMEB latched error	Output	0x00123	1
FCLOB	/FR_COMP latched error	Output	0x00123	2
N1LOB	NETREF1 latched error	Output	0x00123	3
N2LOB	NETREF2 latched error	Output	0x00123	4
D1LOB	DPLL1 sync latched error	Output	0x00123	5
D2LOB	DPLL2 sync latched error	Output	0x00123	6
FSLOB	Failsafe latched error	Output	0x00123	7
FFPOB	FORCE_FALLBACK pending	Output	0x00124	0
CFPOB	CLEAR_FALLBACK pending	Output	0x00124	1
GOPOB	Go clocks pending	Output	0x00124	2
XYSOB	Active clock set	Output	0x00124	3
FBFOB	Fallback enable status	Output	0x00124	7
FBSOP	Fallback states	Output	0x00124	6:4
DPGOB	Data memory active page	Output	0x00125	0
CMROB	Connection memory reset active	Output	0x00125	1
OOLOB	Out-of-lock status	Output	0x00125	6

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
A1LOB	APLL1 lock indicator	Output	0x00125	7
D2LOP	DPLL2 lock status	Output	0x00125	3:2
D1LOP	DPLL1 lock status	Output	0x00125	5:4
DMSOB	Data memory PCI queue status	Output	0x00126	0
NQOOB	NOTIFY_QUEUE overflow error	Output	0x00126	0
VCOOB	VC memory overflow warning	Output	0x00126	1
MBSOB	MB PCI queue status	Output	0x00126	2
PMIOB	PCI master initial warning	Output	0x00126	2
PMOOB	PCI master overwrite	Output	0x00126	3
PMWOB	PCI master stall warning	Output	0x00126	4
PMEOB	PCI master stall error	Output	0x00126	5
PMLOB	PCI master lock error	Output	0x00126	6
PMFOB	PCI master fatal error	Output	0x00126	7
DMPOB	Data memory PCI error status	Output	0x00127	0
VCPOB	VC memory PCI error	Output	0x00127	1
MBPOB	MB PCI error status	Output	0x00127	2
DMTOB	Data memory PCI timer status	Output	0x00127	3
VCTOB	VC memory PCI timer	Output	0x00127	4
MBTOB	MB PCI timer status	Output	0x00127	5
CFBOB	Fallback status	Output	0x00127	6
CFSOB	Failsafe status	Output	0x00127	7
VEROR	Version ID register	Output	0x00128	—
IDLOR	Device ID low	Output	0x0012A	—
IDHOR	Device ID high	Output	0x0012B	—
VCSOB	VC memory PCI queue	Output	0x0012C	1
VSPOB	VC start pending	Output	0x0012D	4
VPPOB	VC pause pending	Output	0x0012D	5
VCEON	VC enable status	Output	0x0012D	L
FT0EB	FG0 test-point	Enable	0x00140	0
FT1EB	FG1 test-point	Enable	0x00140	1
FT2EB	FG2 test-point	Enable	0x00140	2
FT3EB	FG3 test-point	Enable	0x00140	3
FT4EB	FG4 test-point	Enable	0x00140	4
FT5EB	FG5 test-point	Enable	0x00140	5
FT6EB	FG6 test-point	Enable	0x00140	6
FT7EB	FG7 test-point	Enable	0x00140	7
FTPSR	FG test-point MUX	Select	0x00141	—
GT0EB	GP0 test-point	Enable	0x00142	0
GT1EB	GP1 test-point	Enable	0x00142	1
GT2EB	GP2 test-point	Enable	0x00142	2
GT3EB	GP3 test-point	Enable	0x00142	3

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
GT4EB	GP4 test-point	Enable	0x00142	4
GT5EB	GP5 test-point	Enable	0x00142	5
GT6EB	GP6 test-point	Enable	0x00142	6
GT7EB	GP7 test-point	Enable	0x00142	7
GTPSR	GP test-point MUX	Select	0x00143	—
SCMLR	Diagnostic, state counter mode low	Load	0x00144	—
SCLSB	Diagnostic, state counter mode EN	Select	0x00145	3
FRMSB	Diagnostic, /FR_COMP input	Select	0x00145	4
SCMSB	Diagnostic, state counter carry	Select	0x00145	5
SCULP	Diagnostic, state counter mode high	Load	0x00145	2:0
FB1SB	APLL1 feedback reset	Select	0x00146	1
FB2SB	APLL2 feedback reset	Select	0x00146	2
VCMEB	Diagnostic, VC microprocessor access	Enable	0x00146	3
MBIEB	Diagnostic, MB microprocessor access	Enable	0x00146	4
PDTSB	Diagnostic, PCI discard timer	Select	0x00146	5
EBOLR	Diagnostic, external buffer retry	Load	0x00147	—
IEXLP	Diagnostic, interrupt control EXTERR	Load	0x00148	1:0
ISYLP	Diagnostic, interrupt control SYSERR	Load	0x00148	3:2
ICKLP	Diagnostic, interrupt control CLKERR	Load	0x00148	5:4
ICDSP	Diagnostic, interrupt control mode	Select	0x00148	7:6
IASLR	Diagnostic, SYSERR assertion	Load	0x00149	—
CFLLR	Diagnostic sync-to-frame low	Load	0x0014A	—
CFHLN	Diagnostic sync-to-frame high	Load	0x0014B	L
CFSEN	Diagnostic sync-to-frame EN	Enable	0x0014B	U
CKMSR	Clock main	Select	0x00200	—
CKMDR	Clock main	Divide	0x00201	—
P1ISR	APLL1 input	Select	0x00202	—
P1RSR	APLL1 rate	Select	0x00203	—
N1SSB	NR1 selector inversion	Select	0x00204	0
N1DSB	NR1 divider inversion	Select	0x00204	1
N2SSB	NR2 selector inversion	Select	0x00204	2
N2DSB	NR2 divider inversion	Select	0x00204	3
ICMSB	Invert clock main	Select	0x00204	4
CKRDR	Clock resource	Divide	0x00205	—
P2RSR	APLL2 rate	Select	0x00207	—
LRISR	Local reference input	Select	0x00208	—
D1ISR	DPLL1 input	Select	0x0020A	—
D1RSR	DPLL1 rate	Select	0x0020B	—
IR0SB	Invert local reference 0	Select	0x0020C	0
IR1SB	Invert local reference 1	Select	0x0020C	1
IR2SB	Invert local reference 2	Select	0x0020C	2

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
IR3SB	Invert local reference 3	Select	0x0020C	3
IR4SB	Invert local reference 4	Select	0x0020C	4
IR5SB	Invert local reference 5	Select	0x0020C	5
IR6SB	Invert local reference 6	Select	0x0020C	6
IR7SB	Invert local reference 7	Select	0x0020C	7
D2ISR	DPLL2 input	Select	0x0020E	—
D2RSR	DPLL2 rate	Select	0x0020F	—
N1ISN	NETREF1 main input	Select	0x00210	L
N1DSN	NETREF1 divider input	Select	0x00210	U
NR1DR	NETREF1	Divide	0x00211	—
N1LSR	NETREF1 local reference	Select	0x00212	—
N2ISN	NETREF1 main input	Select	0x00214	L
N2DSN	NETREF1 divider input	Select	0x00214	U
NR2DR	NETREF2	Divide	0x00215	—
N2LSR	NETREF1 local reference	Select	0x00216	—
CCOEN	C clocks output	Enable	0x00220	L
ABOEN	A and B clocks output	Enable	0x00220	U
N1OEN	NETREF1 output	Enable	0x00221	L
N2OEN	NETREF1 output	Enable	0x00221	U
FRWSR	/FR_COMP width	Select	0x00222	—
ACRSN	A clocks rate	Select	0x00223	L
BCRSN	B clocks rate	Select	0x00223	U
CCSEN	C clocks separate	Enable	0x00224	L
FRSEN	/FR_COMP separate	Enable	0x00224	U
TCOSR	T clock output	Select	0x00226	—
SCRSR	SCLK/SCLKx2 rate	Select	0x00227	—
LC0SR	Local clock 0 output	Select	0x00228	—
LC1SR	Local clock 1 output	Select	0x00229	—
LC2SR	Local clock 2 output	Select	0x0022A	—
LC3SR	Local clock 3 output	Select	0x0022B	—
HARSN	H1x0 group A rate	Select	0x00300	L
HBRSN	H1x0 group B rate	Select	0x00300	U
HCRSN	H1x0 group C rate	Select	0x00301	L
HDRSN	H1x0 group D rate	Select	0x00301	U
HERSN	H1x0 group E rate	Select	0x00302	L
HFRSN	H1x0 group F rate	Select	0x00302	U
HGRSN	H1x0 group G rate	Select	0x00303	L
HHRSN	H1x0 group H rate	Select	0x00303	U
LARSN	Local group A rate	Select	0x00320	L
LBRSN	Local group B rate	Select	0x00320	U
LCRSN	Local group C rate	Select	0x00321	L

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
LDRSN	Local group D rate	Select	0x00321	U
LERSN	Local group E rate	Select	0x00322	L
LFRSN	Local group F rate	Select	0x00322	U
LGRSN	Local group G rate	Select	0x00323	L
LHRSN	Local group H rate	Select	0x00323	U
F0LLR	Frame 0 lower start time	Load	0x00400	—
F0ULR	Frame 0 upper start time	Load	0x00401	—
F0ISB	Frame 0 pulse inversion	Enable	0x00402	7
F0WSP	Frame 0 pulse width	Select	0x00402	6:0
F0RSR	Frame 0 pulse width rate	Select	0x00403	—
F1LLR	Frame 1 lower start time	Load	0x00410	—
F1ULR	Frame 1 upper start time	Load	0x00411	—
F1ISB	Frame 1 pulse inversion	Enable	0x00412	7
F1WSP	Frame 1 pulse width	Select	0x00412	6:0
F1RSR	Frame 1 pulse width rate	Select	0x00413	—
F2LLR	Frame 2 lower start time	Load	0x00420	—
F2ULR	Frame 2 upper start time	Load	0x00421	—
F2ISB	Frame 2 pulse inversion	Enable	0x00422	7
F2WSP	Frame 2 pulse width	Select	0x00422	6:0
F2RSR	Frame 2 pulse width rate	Select	0x00423	—
F3LLR	Frame 3 lower start time	Load	0x00430	—
F3ULR	Frame 3 upper start time	Load	0x00431	—
F3ISB	Frame 3 pulse inversion	Enable	0x00432	7
F3WSP	Frame 3 pulse width	Select	0x00432	6:0
F3RSR	Frame 3 pulse width rate	Select	0x00433	—
F4LLR	Frame 4 lower start time	Load	0x00440	—
F4ULR	Frame 4 upper start time	Load	0x00441	—
F4ISB	Frame 4 pulse inversion	Enable	0x00442	7
F4WSP	Frame 4 pulse width	Select	0x00442	6:0
F4RSR	Frame 4 pulse width rate	Select	0x00443	—
F5LLR	Frame 5 lower start time	Load	0x00450	—
F5ULR	Frame 5 upper start time	Load	0x00451	—
F5ISB	Frame 5 pulse inversion	Enable	0x00452	7
F5WSP	Frame 5 pulse width	Select	0x00452	6:0
F5RSR	Frame 5 pulse width rate	Select	0x00453	—
F6LLR	Frame 6 lower start time	Load	0x00460	—
F6ULR	Frame 6 upper start time	Load	0x00461	—
F6ISB	Frame 6 pulse inversion	Enable	0x00462	7
F6WSP	Frame 6 pulse width	Select	0x00462	6:0
F6RSR	Frame 6 pulse width rate	Select	0x00463	—
F7LLR	Frame 7 lower start time	Load	0x00470	—

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
F7ULR	Frame 7 upper start time	Load	0x00471	—
F7ISB	Frame 7 pulse inversion	Enable	0x00472	7
F7WSP	Frame 7 pulse width	Select	0x00472	6:0
F7RSR	Frame 7 pulse width rate	Select	0x00473	—
FCLLR	Frame group 7 lower count	Load	0x00474	—
FCULR	Frame group 7 upper count	Load	0x00475	—
F7MSR	Frame 7 mode	Select	0x00476	—
FCISB	FG7 timer invert output	Select	0x00477	7
F7WSN	FG7 timer pulse width	Select	0x00477	L
F7SSP	FG7 timer pulse shape	Select	0x00477	—
F0IOB	FGIO 0 data	Load	0x00480	0
F1IOB	FGIO 1 data	Load	0x00480	1
F2IOB	FGIO 2 data	Load	0x00480	2
F3IOB	FGIO 3 data	Load	0x00480	3
F4IOB	FGIO 4 data	Load	0x00480	4
F5IOB	FGIO 5 data	Load	0x00480	5
F6IOB	FGIO 6 data	Load	0x00480	6
F7IOB	FGIO 7 data	Load	0x00480	7
F0MEB	FGIO 0 read mask	Enable	0x00481	0
F1MEB	FGIO 1 read mask	Enable	0x00481	1
F2MEB	FGIO 2 read mask	Enable	0x00481	2
F3MEB	FGIO 3 read mask	Enable	0x00481	3
F4MEB	FGIO 4 read mask	Enable	0x00481	4
F5MEB	FGIO 5 read mask	Enable	0x00481	5
F6MEB	FGIO 6 read mask	Enable	0x00481	6
F7MEB	FGIO 7 read mask	Enable	0x00481	7
F0DSB	FGIO 0 R/W direction	Select	0x00482	0
F1DSB	FGIO 1 R/W direction	Select	0x00482	1
F2DSB	FGIO 2 R/W direction	Select	0x00482	2
F3DSB	FGIO 3 R/W direction	Select	0x00482	3
F4DSB	FGIO 4 R/W direction	Select	0x00482	4
F5DSB	FGIO 5 R/W direction	Select	0x00482	5
F6DSB	FGIO 6 R/W direction	Select	0x00482	6
F7DSB	FGIO 7 R/W direction	Select	0x00482	7
G0IOB	GPIO 0 data	Load	0x00500	0
G1IOB	GPIO 1 data	Load	0x00500	1
G2IOB	GPIO 2 data	Load	0x00500	2
G3IOB	GPIO 3 data	Load	0x00500	3
G4IOB	GPIO 4 data	Load	0x00500	4
G5IOB	GPIO 5 data	Load	0x00500	5
G6IOB	GPIO 6 data	Load	0x00500	6

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
G7IOB	GPIO 7 data	Load	0x00500	7
G0MEB	PIO 0 read mask	Enable	0x00501	0
G1MEB	GPIO 1 read mask	Enable	0x00501	1
G2MEB	GPIO 2 read mask	Enable	0x00501	2
G3MEB	GPIO 3 read mask	Enable	0x00501	3
G4MEB	GPIO 4 read mask	Enable	0x00501	4
G5MEB	GPIO 5 read mask	Enable	0x00501	5
G6MEB	GPIO 6 read mask	Enable	0x00501	6
G7MEB	GPIO 7 read mask	Enable	0x00501	7
G0DSB	GPIO 0 R/W direction	Select	0x00502	0
G1DSB	GPIO 1 R/W direction	Select	0x00502	1
G2DSB	GPIO 2 R/W direction	Select	0x00502	2
G3DSB	GPIO 3 R/W direction	Select	0x00502	3
G4DSB	GPIO 4 R/W direction	Select	0x00502	4
G5DSB	GPIO 5 R/W direction	Select	0x00502	5
G6DSB	GPIO 6 R/W direction	Select	0x00502	6
G7DSB	GPIO 7 R/W direction	Select	0x00502	7
G0OEB	GPIO 0 override	Enable	0x00503	0
G1OEB	GPIO 0 override	Enable	0x00503	1
G2OEB	GPIO 2 override	Enable	0x00503	2
JF0OB	Interrupt pending FGIO 0	Output	0x00600	0
JF1OB	Interrupt pending FGIO 1	Output	0x00600	1
JF2OB	Interrupt pending FGIO 2	Output	0x00600	2
JF3OB	Interrupt pending FGIO 3	Output	0x00600	3
JF4OB	Interrupt pending FGIO 4	Output	0x00600	4
JF5OB	Interrupt pending FGIO 5	Output	0x00600	5
JF6OB	Interrupt pending FGIO 6	Output	0x00600	6
JF7OB	Interrupt pending FGIO 7	Output	0x00600	7
JF0EB	Interrupt from FGIO 0	Enable	0x00601	0
JF1EB	Interrupt from FGIO 1	Enable	0x00601	1
JF2EB	Interrupt from FGIO 2	Enable	0x00601	2
JF3EB	Interrupt from FGIO 3	Enable	0x00601	3
JF4EB	Interrupt from FGIO 4	Enable	0x00601	4
JF5EB	Interrupt from FGIO 5	Enable	0x00601	5
JF6EB	Interrupt from FGIO 6	Enable	0x00601	6
JF7EB	Interrupt from FGIO 7	Enable	0x00601	7
IF0SB	Invert interrupt FGIO 0	Select	0x00603	0
IF1SB	Invert interrupt FGIO 1	Select	0x00603	1
IF2SB	Invert interrupt FGIO 2	Select	0x00603	2
IF3SB	Invert interrupt FGIO 3	Select	0x00603	3
IF4SB	Invert interrupt FGIO 4	Select	0x00603	4

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
IF5SB	Invert interrupt FGIO 5	Select	0x00603	5
IF6SB	Invert interrupt FGIO 6	Select	0x00603	6
IF7SB	Invert interrupt FGIO 7	Select	0x00603	7
JG0OB	Interrupt pending GPIO 0	Output	0x00604	0
JG1OB	Interrupt pending GPIO 1	Output	0x00604	1
JG2OB	Interrupt pending GPIO 2	Output	0x00604	2
JG3OB	Interrupt pending GPIO 3	Output	0x00604	3
JG4OB	Interrupt pending GPIO 4	Output	0x00604	4
JG5OB	Interrupt pending GPIO 5	Output	0x00604	5
JG6OB	Interrupt pending GPIO 6	Output	0x00604	6
JG7OB	Interrupt pending GPIO 7	Output	0x00604	7
JG0EB	Interrupt from GPIO 0	Enable	0x00605	0
JG1EB	Interrupt from GPIO 1	Enable	0x00605	1
JG2EB	Interrupt from GPIO 2	Enable	0x00605	2
JG3EB	Interrupt from GPIO 3	Enable	0x00605	3
JG4EB	Interrupt from GPIO 4	Enable	0x00605	4
JG5EB	Interrupt from GPIO 5	Enable	0x00605	5
JG6EB	Interrupt from GPIO 6	Enable	0x00605	6
JG7EB	Interrupt from GPIO 7	Enable	0x00605	7
IG0SB	Invert interrupt GPIO 0	Select	0x00607	0
IG1SB	Invert interrupt GPIO 1	Select	0x00607	1
IG2SB	Invert interrupt GPIO 2	Select	0x00607	2
IG3SB	Invert interrupt GPIO 3	Select	0x00607	3
IG4SB	Invert interrupt GPIO 4	Select	0x00607	4
IG5SB	Invert interrupt GPIO 5	Select	0x00607	5
IG6SB	Invert interrupt GPIO 6	Select	0x00607	6
IG7SB	Invert interrupt GPIO 7	Select	0x00607	7
JS0OB	Interrupt pending SYSERR 0	Output	0x00608	0
JS1OB	Interrupt pending SYSERR 1	Output	0x00608	1
JS2OB	Interrupt pending SYSERR 2	Output	0x00608	2
JS3OB	Interrupt pending SYSERR 3	Output	0x00608	3
JS4OB	Interrupt pending SYSERR 4	Output	0x00608	4
JS5OB	Interrupt pending SYSERR 5	Output	0x00608	5
JS6OB	Interrupt pending SYSERR 6	Output	0x00608	6
JS7OB	Interrupt pending SYSERR 7	Output	0x00608	7
JS8OB	Interrupt pending SYSERR 8	Output	0x00609	0
JS9OB	Interrupt pending SYSERR 9	Output	0x00609	1
JSAOB	Interrupt pending SYSERR A	Output	0x00609	2
JSBOB	Interrupt pending SYSERR B	Output	0x00609	3
JSCOB	Interrupt pending SYSERR C	Output	0x00609	4
JSDOB	Interrupt pending SYSERR D	Output	0x00609	5

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
JSEOB	Interrupt pending SYSERR E	Output	0x00609	6
JSFOB	Interrupt pending SYSERR F	Output	0x00609	7
JS0EB	Interrupt from SYSERR 0	Enable	0x0060A	0
JS1EB	Interrupt from SYSERR 1	Enable	0x0060A	1
JS2EB	Interrupt from SYSERR 2	Enable	0x0060A	2
JS3EB	Interrupt from SYSERR 3	Enable	0x0060A	3
JS4EB	Interrupt from SYSERR 4	Enable	0x0060A	4
JS5EB	Interrupt from SYSERR 5	Enable	0x0060A	5
JS6EB	Interrupt from SYSERR 6	Enable	0x0060A	6
JS7EB	Interrupt from SYSERR 7	Enable	0x0060A	7
JS8EB	Interrupt from SYSERR 8	Enable	0x0060B	0
JS9EB	Interrupt from SYSERR 9	Enable	0x0060B	1
JSAEB	Interrupt from SYSERR A	Enable	0x0060B	2
JSBEB	Interrupt from SYSERR B	Enable	0x0060B	3
JSCEB	Interrupt from SYSERR C	Enable	0x0060B	4
JSDEB	Interrupt from SYSERR D	Enable	0x0060B	5
JSEEB	Interrupt from SYSERR E	Enable	0x0060B	6
JSFEB	Interrupt from SYSERR F	Enable	0x0060B	7
JC0OB	Interrupt pending CLKERR 0	Output	0x0060C	0
JC1OB	Interrupt pending CLKERR 1	Output	0x0060C	1
JC2OB	Interrupt pending CLKERR 2	Output	0x0060C	2
JC3OB	Interrupt pending CLKERR 3	Output	0x0060C	3
JC4OB	Interrupt pending CLKERR 4	Output	0x0060C	4
JC5OB	Interrupt pending CLKERR 5	Output	0x0060C	5
JC6OB	Interrupt pending CLKERR 6	Output	0x0060C	6
JC7OB	Interrupt pending CLKERR 7	Output	0x0060C	7
JC8OB	Interrupt pending CLKERR 8	Output	0x0060D	0
JC9OB	Interrupt pending CLKERR 9	Output	0x0060D	1
JCAOB	Interrupt pending CLKERR A	Output	0x0060D	2
JCBOB	Interrupt pending CLKERR B	Output	0x0060D	3
JCCOB	Interrupt pending CLKERR C	Output	0x0060D	4
JCDOB	Interrupt pending CLKERR D	Output	0x0060D	5
JCEOB	Interrupt pending CLKERR E	Output	0x0060D	6
JCFOB	Interrupt pending CLKERR F	Output	0x0060D	7
JC0EB	Interrupt from CLKERR 0	Enable	0x0060E	0
JC1EB	Interrupt from CLKERR 1	Enable	0x0060E	1
JC2EB	Interrupt from CLKERR 2	Enable	0x0060E	2
JC3EB	Interrupt from CLKERR 3	Enable	0x0060E	3
JC4EB	Interrupt from CLKERR 4	Enable	0x0060E	4
JC5EB	Interrupt from CLKERR 5	Enable	0x0060E	5
JC6EB	Interrupt from CLKERR 6	Enable	0x0060E	6

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
JC7EB	Interrupt from CLKERR 7	Enable	0x0060E	7
JC8EB	Interrupt from CLKERR 8	Enable	0x0060F	0
JC9EB	Interrupt from CLKERR 9	Enable	0x0060F	1
JCAEB	Interrupt from CLKERR A	Enable	0x0060F	2
JCBEB	Interrupt from CLKERR B	Enable	0x0060F	3
JCCEB	Interrupt from CLKERR C	Enable	0x0060F	4
JCDEB	Interrupt from CLKERR D	Enable	0x0060F	5
JCEEB	Interrupt from CLKERR E	Enable	0x0060F	6
JCFEB	Interrupt from CLKERR F	Enable	0x0060F	7
JAMSR	Interrupt arbitration mode	Select	0x00610	—
JSPSR	Interrupt SYSERR-to-PCI_INTA	Select	0x00611	—
JSOSR	Interrupt SYSERR output mode	Select	0x00612	—
JCOSR	Interrupt CLKERR output mode	Select	0x00613	—
JSWSR	Interrupt SYSERR pulse width	Select	0x00616	—
JCWSR	Interrupt CLKERR pulse width	Select	0x00617	—
JISOR	Interrupt in-service	Output	0x006FC	—
JVLOR	Interrupt in-service VC ID low	Output	0x006FE	—
JVHOB	Interrupt in-service VC ID high	Output	0x006FF	0
R0SLR	MB_CS0 read cycle setup	Load	0x00700	—
R0WLR	MB_CS0 read cycle width	Load	0x00701	—
R0HLR	MB_CS0 read cycle hold	Load	0x00702	—
A0SLR	MB_CS0 address setup	Load	0x00703	—
W0SLR	MB_CS0 write cycle setup	Load	0x00704	—
W0WLR	MB_CS0 write cycle width	Load	0x00705	—
W0HLR	MB_CS0 write cycle hold	Load	0x00706	—
R1SLR	MB_CS1 read cycle setup	Load	0x00710	—
R1WLR	MB_CS1 read cycle width	Load	0x00711	—
R1HLR	MB_CS1 read cycle hold	Load	0x00712	—
A1SLR	MB_CS1 address setup	Load	0x00713	—
W1SLR	MB_CS1 write cycle setup	Load	0x00714	—
W1WLR	MB_CS1 write cycle width	Load	0x00715	—
W1HLR	MB_CS1 write cycle hold	Load	0x00716	—
A1HLR	MB_CS1 address hold	Load	0x00717	—
R2SLR	MB_CS2 read cycle setup	Load	0x00720	—
R2WLR	MB_CS2 read cycle width	Load	0x00721	—
R2HLR	MB_CS2 read cycle hold	Load	0x00722	—
A2SLR	MB_CS2 address setup	Load	0x00723	—
W2SLR	MB_CS2 write cycle setup	Load	0x00724	—
W2WLR	MB_CS2 write cycle width	Load	0x00725	—
W2HLR	MB_CS2 write cycle hold	Load	0x00726	—
A2HLR	MB_CS2 address hold	Load	0x00727	—

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
R3SLR	MB_CS3 read cycle setup	Load	0x00730	—
R3WLR	MB_CS3 read cycle width	Load	0x00731	—
R3HLR	MB_CS3 read cycle hold	Load	0x00732	—
A3SLR	MB_CS3 address setup	Load	0x00733	—
W3SLR	MB_CS3 write cycle setup	Load	0x00734	—
W3WLR	MB_CS3 write cycle width	Load	0x00735	—
W3HLR	MB_CS3 write cycle hold	Load	0x00736	—
A3HLR	MB_CS3 address hold	Load	0x00737	—
R4SLR	MB_CS4 read cycle setup	Load	0x00740	—
R4WLR	MB_CS4 read cycle width	Load	0x00741	—
R4HLR	MB_CS4 read cycle hold	Load	0x00742	—
A4SLR	MB_CS4 address setup	Load	0x00743	—
W4SLR	MB_CS4 write cycle setup	Load	0x00744	—
W4WLR	MB_CS4 write cycle width	Load	0x00745	—
W4HLR	MB_CS4 write cycle hold	Load	0x00746	—
A4HLR	MB_CS4 address hold	Load	0x00747	—
R5SLR	MB_CS5 read cycle setup	Load	0x00750	—
R5WLR	MB_CS5 read cycle width	Load	0x00751	—
R5HLR	MB_CS5 read cycle hold	Load	0x00752	—
A5SLR	MB_CS5 address setup	Load	0x00753	—
W5SLR	MB_CS5 write cycle setup	Load	0x00754	—
W5WLR	MB_CS5 write cycle width	Load	0x00755	—
W5HLR	MB_CS5 write cycle hold	Load	0x00756	—
A5HLR	MB_CS5 address hold	Load	0x00757	—
R6SLR	MB_CS6 read cycle setup	Load	0x00760	—
R6WLR	MB_CS6 read cycle width	Load	0x00761	—
R6HLR	MB_CS6 read cycle hold	Load	0x00762	—
A6SLR	MB_CS6 address setup	Load	0x00763	—
W6SLR	MB_CS6 write cycle setup	Load	0x00764	—
W6WLR	MB_CS6 write cycle width	Load	0x00765	—
W6HLR	MB_CS6 write cycle hold	Load	0x00766	—
A6HLR	MB_CS6 address hold	Load	0x00767	—
R7SLR	MB_CS7 read cycle setup	Load	0x00770	—
R7WLR	MB_CS7 read cycle width	Load	0x00771	—
R7HLR	MB_CS7 read cycle hold	Load	0x00772	—
A7SLR	MB_CS7 address setup	Load	0x00773	—
W7SLR	MB_CS7 write cycle setup	Load	0x00774	—
W7WLR	MB_CS7 write cycle width	Load	0x00775	—
W7HLR	MB_CS7 write cycle hold	Load	0x00776	—
A7HLR	MB_CS7 address hold	Load	0x00777	—
IC0SB	Invert MB CS0 strobe	Select	0x00780	0

Appendix B. Register Bit Field Mnemonic Summary (continued)

Table 134. Mnemonic Summary, Sorted by Register (continued)

Mnemonic	Description	Type	Register	Bit Position
IC1SB	Invert MB CS1 strobe	Select	0x00780	1
IC2SB	Invert MB CS2 strobe	Select	0x00780	2
IC3SB	Invert MB CS3 strobe	Select	0x00780	3
IC4SB	Invert MB CS4 strobe	Select	0x00780	4
IC5SB	Invert MB CS5 strobe	Select	0x00780	5
IC6SB	Invert MB CS6 strobe	Select	0x00780	6
IC7SB	Invert MB CS7 strobe	Select	0x00780	7
IMWSB	Invert MB write strobe	Select	0x00781	0
IMRSB	Invert MB read strobe	Select	0x00781	1
IPRSB	Invert MB PCIRST, forwarded	Select	0x00781	2

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)

Ambassador is a registered trademark of Agere Systems Inc.

May 2001
DS00-434CTI (Replaces DS00-012CTI, AY00-030CTI, and AY01-009CTI; must accompany AY01-021CTI)