4539 PMC Quad T1/E1/J1 Communications Controller

Hardware Reference Manual

Document No. UG04539-001-B

Print Date: October 2001

Copyright Notice

Interphase Corporation
13800 Senlac
Dallas, Texas 75234

Phone: (214) 654-5000
Fax: (214) 654-5506

Disclaimer

Trademark Acknowledgments

Assistance

Product Purchased from Reseller

Contact the reseller or distributor if

- You need ordering, service or any technical assistance.
- You received a damaged, incomplete or incorrect product.

Product Purchased Directly from Interphase Corporation

Contact Interphase Corporation directly for assistance with this, or any other Interphase Corporation product. Please have your purchase order and serial numbers ready.

Customer Support

United States:	Telephone:	(214) 654-5555
	Fax:	(214) 654-5506
	E-Mail:	intouch@iphase.com
Europe:	Telephone:	+ 33 (0) 1 41 15 44 00
	Fax:	+ 33 (0) 1 41 15 12 13

World Wide Web

http://www.interphase.com

Anonymous FTP Server

ftp.interphase.com

END-USER LICENSE AGREEMENT FOR INTERPHASE CORPORATION SOFTWARE

IMPORTANT NOTICE TO USER–READ CAREFULLY

THIS END-USER LICENSE AGREEMENT FOR INTERPHASE CORPORATION SOFTWARE (“AGREEMENT”) IS A LEGAL AGREEMENT BETWEEN YOU (EITHER AN INDIVIDUAL OR SINGLE ENTITY) AND INTERPHASE CORPORATION FOR THE SOFTWARE PRODUCTS ENCLOSED HEREIN WHICH INCLUDES COMPUTER SOFTWARE AND PRINTED MATERIALS (“SOFTWARE”). BY INSTALLING, COPYING, OR OTHERWISE USING THE ENCLOSED SOFTWARE, YOU AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT, PROMPTLY RETURN, WITHIN THIRTY DAYS, THE UNUSED SOFTWARE TO THE PLACE FROM WHICH YOU OBTAINED IT FOR A FULL REFUND.

The Software is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The Software is licensed, not sold.

Grant of License: You are granted a personal license to install and use the Software on a single computer solely for internal use and to make one copy of the Software in machine readable form solely for backup purposes.

Restrictions on Use: You may not reverse engineer, decompile, or disassemble the Software. You may not distribute copies of the Software to others or electronically transfer the Software from one computer to another over a network. You may not use the Software from multiple locations of a multi-user or networked system at any time. You may not use this software on any product for which it was not intended. You may not use this software on any non-Interphase product. LICENSEE MAY NOT RENT, LEASE, LOAN, OR RESELL THE SOFTWARE OR ANY PART THEREOF.

Ownership of Software: Interphase or its vendors retain all title to the Software, and all copies thereof, and no title to the Software, or any intellectual property in the Software, is being transferred.

Software Transfer: You may permanently transfer all of your rights under this Agreement, provided you retain no copies, you transfer all the Software, and the recipient agrees to the terms of this Agreement.

Limited Warranty: Interphase Corporation (“Seller”) warrants that (i) the hardware provided to Buyer (“Products”) shall, at the F.O.B. point, be free from defects in materials and workmanship for a period of one (1) year from the date of shipment to Buyer; (ii) the software and/or firmware associated with or embedded in the Products shall comply with the applicable specifications for a period of six (6) months from the date of shipment to Buyer; and (iii) its services will, when performed, be of good quality. Defective and nonconforming Products and software must be held for Seller’s inspection and returned at Seller’s request, freight prepaid, to the original F.O.B. point.

Upon Buyer’s submission of a claim in accordance with Seller’s Return and Repair Policy, Seller will, at its option either (i) repair or replace the nonconforming Product; (ii) correct or replace the software/firmware; (iii) rework the nonconforming services; or (iv) refund an equitable portion of the purchase price attributable to such nonconforming Products, software, or services. Seller shall not be liable for the cost of removal or installation of products or any unauthorized warranty work, nor shall Seller be responsible for any transportation costs, unless expressly authorized in writing by Seller. This warranty does not cover damage to the Product resulting from accident, disaster, misuse, negligence, improper maintenance, or modification or repair of the Product other than by Seller. Any Products or software replaced by Seller will become the property of Seller.

REMEDIES AND EXCLUSIONS. THE SOLE LIABILITY OF SELLER AND BUYER’S SOLE REMEDY FOR BREACH OF THESE WARRANTIES SHALL BE LIMITED TO REPAIR OR REPLACEMENT OF THE PRODUCTS OR CORRECTION OF THAT PART OF THE SOFTWARE, WHICH FAILS TO CONFORM TO THESE WARRANTIES. EXCEPT AS EXPRESSLY STATED HEREIN, AND EXCEPT AS TO TITLE, THERE ARE NO OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE, IN CONNECTION WITH OR ARISING OUT OF ANY PRODUCT OR SOFTWARE PROVIDED TO BUYER.

IN NO EVENT SHALL SELLER HAVE ANY LIABILITY FOR INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, ARISING OUT OF THESE WARRANTIES, INCLUDING BUT NOT LIMITED TO LOSS OF ANTICIPATED PROFITS, LOSS OF DATA, USE OR GOODWILL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. (IC-199, 1/97)

Limitation of Liability: NEITHER INTERPHASE NOR ITS LICENSORS SHALL BE LIABLE FOR ANY GENERAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL, OR OTHER DAMAGES ARISING OUT OF THIS AGREEMENT EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Confidentiality: The Software is copyrighted and contains proprietary and confidential trade secret information of Interphase and its vendors. Licensee agrees to maintain the Software in confidence and not to disclose the Software to any third party without the express written consent of Interphase. Licensee further agrees to take all reasonable precautions to prevent access to the Software by unauthorized persons.

Termination: Without prejudice to any other rights, Interphase may terminate this Agreement if you fail to comply with any term or condition of the Agreement. In such event you must destroy the Software together with all copies, updates, or modifications thereof.

Export: You agree to comply with all export and re-export restrictions and regulations of the U.S. Department of Commerce or other applicable U.S. agency. You must not transfer the Software to a prohibited country or otherwise violate any such restrictions or regulations.

U.S. Government Restricted Rights: Use, duplication, or disclosure of the Software to or by the U.S. Government is subject to restrictions as set forth in the applicable U.S. federal procurement regulations covering commercial/restricted rights software. You are responsible for complying with the notice requirements contained in such regulations.

General: You acknowledge that you have read and understand this Agreement, and by installing and using the Software you agree to be bound by the terms and conditions herein. You further agree that this is the complete and exclusive Agreement between Interphase and yourself. No variation of the terms of this Agreement or any different terms will be enforceable against Interphase unless agreed to in writing by Interphase and yourself. The validity of this Agreement and the rights, obligations, and relations of the parties hereunder shall be determined under the substantive laws of the State of Texas. If any provision of this Agreement is held invalid, illegal, or unenforceable, the remaining provisions shall in no way be affected or impaired thereby. All rights in the Software not specifically granted in this Agreement are reserved by Interphase.

Contents

CHAPTER 4 Accessing the 4539 on the PCI Side

CHAPTER 5 Connectors and Front Panel

CHAPTER 6 Operating environment

List of Figures

List of Tables

List of Examples

Using This Guide

Purpose

This *4539 Hardware Reference Manual* is designed for software developers in Interphase customer organizations who intend to develop embedded software and/or host drivers for the 4539 CompactPCI T1/E1/J1 communications controller.

The 4539 is delivered with an Interphase Boot Firmware located in the FLASH memory. This firmware initializes and configures the 4539 hardware at each boot. It also includes built-in self tests and a monitor. Software developers can decide to keep the Interphase Boot Firmware and develop an operational firmware that will start after the completion of this Interphase Boot Firmware. This solution is recommended for the following reasons:

- PowerQUICC II™ and 4539 initial hardware configuration code is already developed and validated by Interphase in the boot firmware.
- Interphase Boot Firmware provides several ways to download and execute developer's operational firmware.
- Interphase Boot Firmware can be used during the life of the product for operational firmware updates and field unit tests.

However, if the software developers decide to develop their own Boot Firmware, this manual describes in detail, the 4539 Hardware and provides information relative to its initialization and configuration.

NOTES

To install the 4539 Communications Controller in your CompactPCI Machine, please refer to the *4539 Board Installation and Maintenance Manual* (UG04539-000). Different cautions and warnings are described to avoid damage to your communications controller.

All the booting process and software elements composing the Interphase Boot Firmware are described in the *4539 Built-In Self Test and Monitor Manual* (UG04539-004). Refer to this document when using Boot Firmware.

Audience

This guide was written assuming that readers have extensive knowledge of the C programming language and of methods for developing and installing software drivers.

Byte Ordering and Bit Coding Convention

The PCI bus uses the Little Endian Byte ordering: byte 0 in a 32-bit word is the Least Significant Byte (LSB) from an arithmetic point of view and is noted D(7:0). The PowerPC architecture uses the Big Endian Byte ordering: byte 0 in a 32-bit word is the Most Significant Byte (MSB) from an arithmetic point of view and is noted D(31:24).

The PowerPC architecture uses the very unusual Little Endian Bit convention, where bit 0 is on the left and is the most significant bit. Unless otherwise noted, this document does not use this convention. Instead, it uses the classical bit coding convention, where bit 0 (on the right) is the least significant bit and bit i is the 2^i weight bit. This is the Big Endian Bit convention. This coding convention applies to data, addresses, and bit fields. In the following figure, MSB means Most Significant Byte and LSB Least Significant Byte:

31 24			7 0
MSB			LSB

The standard C convention is used to identify the numeric format of arithmetical values:

- No prefix for decimal values
- 0x prefix for a hexadecimal value

For example 0x12 = 18.

Type Definition

Only a few basic types are used:

- byte: unsigned, coded as 8 bits
- word: unsigned, coded as two contiguous bytes, most significant first
- dword: unsigned, coded as two contiguous words, most significant first

Code Examples

This document provides several algorithm descriptions presented in PowerPC assembly language and in C language.

Icon Conventions

Icons draw your attention to especially important information:

NOTE

The Note icon indicates important points of interest related to the current subject.

CAUTION

The Caution icon brings to your attention those items or steps that, if not properly followed, could cause problems in your machine's configuration or operating system.

WARNING

The Warning icon alerts you to steps or procedures that could be hazardous to your health, cause permanent damage to the equipment, or impose unpredictable results on the surrounding environment.

Text Conventions

The following conventions are used in this manual. Computer-generated text is shown in typewriter font. Examples of computer-generated text are: program output (such as the screen display during the software installation procedure), commands, directory names, file names, variables, prompts, and sections of program code.

```
Computer-generated text example
```

Commands to be entered by the user are printed in **`bold Courier`** type. For example:

`cd /usr/tmp`

Pressing the return key (↵ **`Return`**) at the end of the command line entry is assumed, when not explicitly shown. For example:

`/bin/su`

is the same as:

`/bin/su` ↵ **`Return`**

Required user input, when mixed with program output, is printed in **`bold Courier`** type. References to UNIX programs and manual page entries follow the standard UNIX conventions.

When a user command, system prompt, or system response is too long to fit on a single line, it will be shown as

```
Do you want the new kernel moved into
\vmunix?[y]
```

with a backslash at either the beginning of the continued line or at the end of the previous line.

Checking and Downloading from the Interphase WWW/FTP Site

The latest production software drivers, firmware, and documentation (in Adobe Acrobat PDF or text format) for our current products are available on our WWW / FTP site. Interphase recommends our customers visit the web site often to verify that they have the latest version of driver, firmware, or documentation.

WWW Method

1. Access the following web page:

   http://www.interphase.com

2. Move the mouse (or other pointer) and click on the `Products` option. A menu will appear on the left side with Telecom and Server I/O options. Choose the appropriate menu item.

3. A new web page with a list of the currently offered products will appear. Choose the correct communications controller for your system bus and configuration needs by clicking on the product number (i.e. 4538, 6535, 4575, etc.).

4. The Product Description page appears for that product number. At the left side of the page is a list showing further information web pages for that product. Choose the SW Downloads item.

5. A new web page appears with a list of the latest released drivers available for that adapter based on the operating system. Click on the line that describes your driver requirement. Depending on your browser configuration, the driver will now download to your system. If this doesn't work correctly, try to 'right-click' on the proper driver and choose an option that will save the file to your local file system.

FTP Method

1. From your command line, enter following:

   ftp://ftp.iphase.com

2. At the Login: prompt, enter **anonymous**

3. At the Password: prompt, enter your E-mail address.

4. At the ftp prompt, enter **binary**

5. Enter **cd pub**

6. To list all the available product technology directories, enter **dir**. The `00README.txt` file in each directory gives a description of the files and subdirectories in that directory.

7. To access the directory for the technology that you require, enter **cd** followed by the appropriate directory name. Most technology directories also have bus and

operating system subdirectories. In these cases, you must choose the proper bus and operating system by typing **cd <directory>** for the appropriate subdirectories.

8. To download one or more files to your local directory, enter **get <filename>**
9. To exit the FTP site, enter **quit** or **bye**

Hardware Description 1

Overview

The Interphase 4539 PMC Quad T1/E1/J1 Communications Controller is a network interface PCI Mezzanine Card (PMC) equipped with four software-selectable T1/E1/J1 interfaces available with rear access. The 4539 board is intended for 2G and 3G wireless networks, Internet access, and Advanced Intelligent Network (AIN) applications.

This chapter provides the functional specification of the 4539. It describes how the different main components of the board are arranged together.

The main components of the 4539 are:

- The PowerQUICC II™, a Motorola® MPC8260 RISC embedded processor (133 MHz or 200 MHz core frequency)
- The Tundra® PowerSpan®, a dedicated PCI bridge that controls the interface between the card and the host 64-bit PCI bus
- 4 MB of 8-bit FLASH EEPROM memory
- 32 MB or 64 MB of 64-Bit SDRAM system memory
- 8 MB of 32-bit SDRAM connection memory (only on ATM versions)
- One 16K-entry CAM memory (only on ATM versions)
- One INFINEON QuadFALC™ framer that controls four independent T1/E1/J1 interfaces. For each interface, the QuadFALC includes a framer and a Line Interface Unit (LIU) with data and clock recovery.
- An MT90221 or MT90220 (only on ATM versions). This ATM device supports two major modes of operation: the IMA mode (as defined by the ATM Forum IMA Specification) and the User Network Interface (UNI) mode.
- An Intel® LXT971A, an IEEE compliant Fast Ethernet transceiver that supports 10BaseT/100BaseTX auto-negotiation and parallel detection.

4539 Hardware Structure

Figure 1-1 shows the 4539 hardware structure.

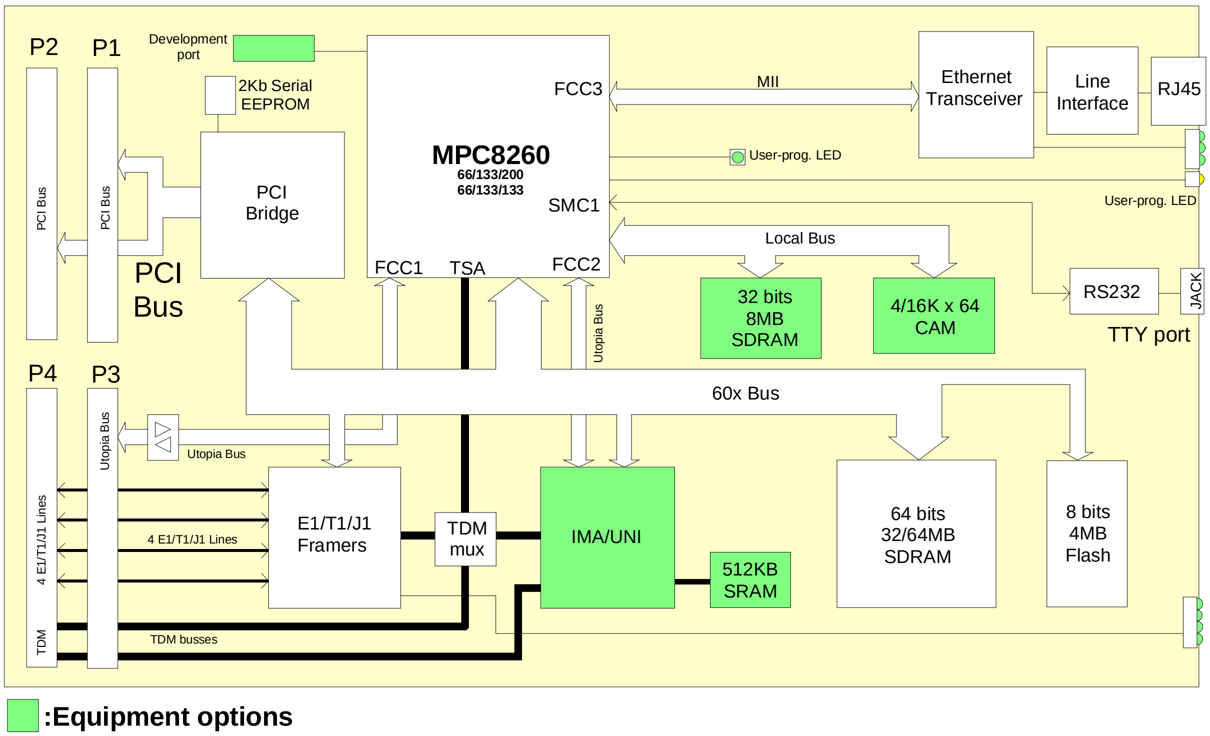


Figure 1-1. 4539 Structure

The PowerQUICC II

The local CPU is a Motorola MPC8260 RISC embedded processor. The MPC8260 includes three major parts:

- An MPC603e core
- A System Interface Unit (SIU)
- A Communication Processor Module (CPM)

The MPC603e core is derived from the PowerPC™ 603e core and includes mainly the integer core and the 16 KB data and 16 KB instruction caches.

The SIU includes a memory management unit and enables control of the external 60x local bus (64-bit data width). The SIU also provides a local bus (32-bit data, 32-bit internal/18-bit external address) used to enhance the operation of the Fast Communication Controllers (FCCs). It can be used to store connection tables for ATM, buffer descriptors, or raw data that is transmitted between channels. It is synchronized with the 60x bus and runs at the same frequency.

The Communication Processor Module (CPM) is a super-set of the PowerQUICC II CPM with additional capabilities. It features:

- Two Multichannel Communications Controllers (MCCs)
- Three Fast Serial Communications Controllers (FCCs). Two are used to control the UTOPIA busses; one is used to control the Ethernet Media-Independent Interface (MII).
- Four Serial Communication Controllers (SCCs)
- Two Serial Management Controllers (SMCs)
- A debug serial port
- A Serial Peripheral Interface (SPI)
- Four timers and an interrupt controller

PowerQUICC II Resets

Once the card is powered-up and the power stabilized, the PowerQUICC II enters into a sequence where it will define certain vital parameters, such as the type of its bus and the PLL multiplication factors. Then it will wait for various conditions, such as PLL stabilization and PCI reset signal de-asserted, before booting.

The PowerQUICC II is controlled by three reset signals:

- –PORESET: Power-on reset
- –HRESET: Hardware reset
- –SRESET: Software Reset

When –PORESET is activated, this also activates –HRESET and –SRESET. –PORESET is the strongest reset. When –HRESET is activated, this also activates –SRESET. When –SRESET is activated, it does not interfere with the other resets (–SRESET is the weakest reset).

A power supervisor controls the MPC8260 input signal –PORESET. It activates –PORESET (0) when the power is not stabilized (at power-up or during power failures). The –PORESET is maintained active for 150 ms after stabilization of the power.

After –PORESET is de-asserted (set to 1), the MPC8260 waits 1024 input clock cycles and samples the MODCK[1:3] bits, which define the default clock multiplication factor and input clock used for the SPLL. The MPC8260 starts its PLL at this time. It maintains –HRESET and –SRESET asserted while the PLL is not locked.

Through its pin –PB_RESET, the PowerSpan also maintains –HRESET asserted as long as the PCI reset signal is activated.

PowerQUICC II –RSTCONF pin is tied to ground, indicating that the MPC8260 is the configuration master. At the rising edge of –HRESET, the MPC8260 generates 64-bit reads into its boot memory (the FLASH) with address starting at 0 and incriminated by 8. The first eight bytes set its Hard Reset Configuration (for detailed initialization see *PowerSpan Hardware Configuration Word* on page 71).

The PowerSpan has no dedicated pin to control the PowerQUICC II hard reset signal –HRESET and soft reset signal –SRESET. Instead, two of its interrupt pins, –INT2 and –INT3 respectively, configured as an output are used. These interrupt are controlled with doorbell bits (see *Hardware and Software Resets Through the PowerSpan* on page 26).

Once all the resets are de-asserted, the PowerQUICC II boots using its 8-bit FLASH device.

The MPC8260 can control the reset of the various communication peripherals through certain CPM I/O ports. When the PowerQUICC II is in reset state, and until it configures these I/O ports as outputs, these reset signals are activated.

System Clocks

The MPC8260 gets its reference clock in its CLOCKIN input pin from a 65.536 MHz reference oscillator.

The MPC8260 input pins MODCK[1:3], along with the MODCK_H field from the Reset Configuration Word define the input clock used for the SPLL and the default clock multiplication factors. The resulting internal system frequencies are:

- PowerPC core frequency: 131.072 MHz (x2) or 196.608 MHz (x3)
- CPM frequency: 131.072 MHz (x2)

Local Space Mapping

The PowerPC local processor can address a 4 GB logical space. In this space, the following elements are mapped:

- The vector table (including the reset entry point)
- The MPC8260 internal registers
- The main SDRAM memory
- The FLASH memory
- The PCI bridge (the PowerSpan) and its local-to-PCI window(s)
- The communications peripherals (QuadFALC, MT90220/1)
- The local SDRAM (local bus)
- The CAM (local bus)

When the MPC8260 boots, it is configured to select the FLASH memory, regardless of the address generated. This will allow the PowerPC to always find the boot start entry in the FLASH. After having booted, having executed a proper jump, and initialized the memory controllers, both the vector table address and the FLASH address can be configured and mapped in other areas: the developer will typically prefer to implement the vector table in a R/W memory device (the main memory SDRAM).

The memory mapping is common to both buses (60x and local). The location of the peripherals on either bus is configured in the Bank Registers. The MPC8260 includes 12 banks with their respective Chip Selects.

The memory mapping has been defined in a way that allows use of the MMU Block Address Translation (BAT) mechanism, which is simpler than the segments-and-pages mechanism. This mechanism divides the memory into several areas that have their own cache properties.

Depending on the device selected, the corresponding memory area can be defined as "cachable" for better performance or must be set as "non-cachable". For instance, the FLASH memory can be cachable. The areas in the SDRAM that are only accessed by the local processor can also be cachable. The peripherals cannot be cachable. The area of SDRAM memory used for the transfer of data cannot be cachable either, because it can be modified by elements other than the PowerQUICC II, such as the PowerSpan DMA.

In order to simultaneously support cachable and non-cachable areas in the SDRAM memories, they are mapped twice in the local space. One mapping area will be defined as cachable and the other will be defined as non-cachable.

Table 1-1 and Figure 1-2 indicate the organization of the local space as defined in the current 4539 Boot Firmware code, with the instruction and data BAT blocks and CS banks used.

Table 1-1. Local Space Mapping

Address Area	Size	Element Accessed	IBAT/ DBAT	CS Bank	Property
0x0000 0000 – 0x03FF FFFF	64MB	60x bus Main memory	0/0	1	Cachable
0x0800 0000 – 0x087F FFFF	8MB	Local bus connection memory (only for the CPM and the external masters)	–/–	2	No access from core
0x8000 0000 – 0x83FF FFFF	64MB	60x bus Main memory (duplicated)	–/1	1	Not cachable
0x8800 0000 – 0x887F FFFF	8MB	Local bus connection memory (duplicated) See **Note**.	–/1	2	Not cachable
0xC000 0000 – 0xCFFF FFFF	256MB	Local to PCI window(s)	–/2	–	Not cachable
0xF002 0000 – 0xF002 FFFF	64KB	PowerSpan internal registers	–/3	–	Not cachable
0xF004 0000 – 0xF004 FFFF	64KB	CAM Control Port	–/3	3	Not cachable
0xF006 0000 – 0xF006 FFFF	64KB	CAM Match Port	–/3	5	Not cachable
0xF008 0000 – 0xF008 FFFF	64KB	QuadFALC	–/3	7	Not cachable
0xF00C 0000 – 0xF00C FFFF	64KB	Inverse Mux. ATM device (IMA)	–/3	6	Not cachable
0xFF00 0000 – 0xFF01 FFFF	128 KB	MPC internal registers (IMMR initial value)	–/3	–	Not cachable
0xFF80 0000 – 0xFFFF FFFF	8 MB	FLASH (initial vector table at 0xFFF0 0000)	3/3	0	Cachable

Note: The local bus connection memory cannot be set as cachable. The PowerQUICC II does not support cachable accesses to the local bus.

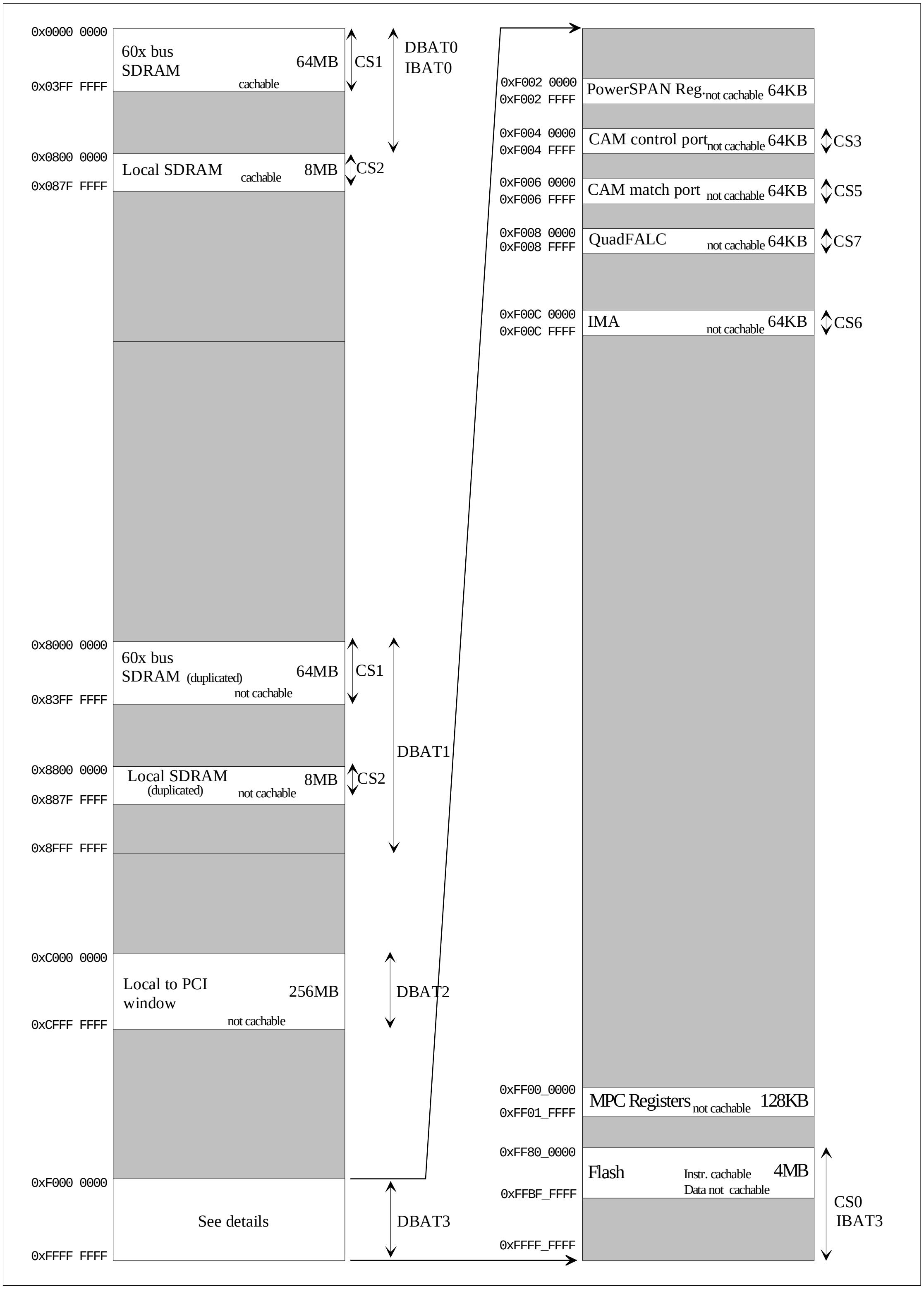


Figure 1-2. Local Space Mapping

NOTE

The CPM and the PowerSpan must use the physical addresses (most significant bit = 0) when accessing the SDRAMs. This is because only the accesses performed by the core go through the Memory Management Unit (MMU).

Interrupts

The PCI bridge PowerSpan and the communication peripherals generate interrupt requests to the PowerQUICC II. These interrupts are level sensitive, active low.

Table 1-2. Local Interrupts

Source	MPC2860 Pin	MPC8260 IRQ
PowerSpan interrupt (ATN)	–IRQ1/DP1/–EXT_BG2	–IRQ1
Ethernet LIU (LXT971A) interrupt	–IRQ2/DP2/–TLBISYNC/–EXT_DBG2	–IRQ2
QuadFALC Interrupt	–IRQ3/DP3/–CKSTP_OUT/–EXT_BR3	–IRQ3
IMA (MT90220/1) Interrupt	–IRQ4/DP4/–CORE_SRESET/–EXT_BR3	–IRQ4

Memory Controllers

The sophisticated memory controller units included in the PowerQUICC II are used on the 4539 boards to control all the external devices, except the PowerSpan, which is directly a 60x bus-compatible device. These units are a General Purpose Chip-select Machine (GPCM) for SRAM, FLASH, and peripherals control, three User Programmable Machines (UPM), and two SDRAM control machines.

The memory controller unit to be used is defined bank per bank. Each bank is defined by its Base Register (BRx) and its Option Register (ORx). The memory machine selection is done in the Option register.

Table 1-3. PowerQUICC II Memory Controller Machine Usage

Element Accessed	Bank	Memory Controller
Flash EEPROM	0	60x bus GPCM
60x bus Main memory	1	60x bus SDRAM machine
Local bus connection memory	2	Local bus SDRAM machine
CAM Control Port	3	UPMB
CAM Match Port	5	UPMA
MT90220/1	6	60x bus GPCM

Table 1-3. PowerQUICC II Memory Controller Machine Usage (cont)

Element Accessed	Bank	Memory Controller
QuadFALC	7	60x bus GPCM

Communication Processor Module (CPM) I/O Ports

The CPM part of the PowerQUICC II provides several communication functions. These functions use multi-mode pins that are grouped in four I/O ports: Port A, B, C, and D. The 4539 communications controller uses these ports as shown in the following tables:

Table 1-4. CPM Port A Usage

CPM I/O Port	Pin Configuration	Dir	Usage
PA(0:2)	FCC2:TxAddr[2:0]	O	FCC2: UTOPIA MPHY Master
PA(3:5)	FCC2:RxAddr[0:2]	O	FCC2: UTOPIA MPHY Master
PA(6)	TDMa1:L1RSYNC	I	TDMa1
PA(7)	Input	I	CONFIG0: Optional configuration input. Set to 0 or 1 by resistors.
PA(8)	TDMa1:L1RXD	I	TDMa1
PA(9)	TDMa1:L1TXD	O	TDMa1
PA(10:17)	FCC1:RxD[0:7]	I	FCC1: UTOPIA 8
PA(18:25)	FCC1:TxD[7:0]	O	FCC1: UTOPIA 8
PA(26)	FCC1:RxClav	O(I)	FCC1: UTOPIA Slave (or Master)
PA(27)	FCC1:RxSOC	I	FCC1: UTOPIA 8
PA(28)	FCC1:RxEnb	I(O)	FCC1: UTOPIA Slave (or Master)
PA(29)	FCC1:TxSOC	O	FCC1: UTOPIA 8
PA(30)	FCC1:TxClav	O(I)	FCC1: UTOPIA Slave (or Master)
PA(31)	FCC1:TxEnb	I(O)	FCC1: UTOPIA Slave (or Master)

Table 1-5. CPM Port B Usage

CPM I/O Port	Pin Configuration	Dir	Usage
PB(4:7)	FCC3:TXD[3:0]	O	Fast Ethernet
PB(8:11)	FCC3:RXD[0:3]	I	Fast Ethernet
PB(12)	FCC3:CRS	I	Fast Ethernet
PB(13)	FCC3:COL	I	Fast Ethernet
PB(14)	FCC3:TX_EN	O	Fast Ethernet
PB(15)	FCC3:TX_ER	O	Fast Ethernet
PB(16)	FCC3:RX_ER	I	Fast Ethernet
PB(17)	FCC3:RX_DV	I	Fast Ethernet

Table 1-5. CPM Port B Usage (cont)

CPM I/O Port	Pin Configuration	Dir	Usage
PB(18:21)	FCC2:RXD[4:7]	I	FCC2: UTOPIA MPHY Master
PB(22:23)	FCC2:TXD[7:6]	O	FCC2: UTOPIA MPHY Master
PB(24)	FCC2:TXD[5] or TDMc2:L1RSYNC	O or I	FCC2: UTOPIA MPHY Master in IMA/UNI mode or TDMc2 in other modes. **Before configuring this port as output, select the IMA/UNI mode (see Table 1-34) to avoid signals conflict.**
PB(25)	FCC2:TXD[4]	O	FCC2: UTOPIA MPHY Master
PB(26)	FCC2:TXD[1] or TDMc2:L1RXD	O or I	FCC2: UTOPIA MPHY Master in IMA/UNI mode or TDMc2 in other modes. **Before configuring this port as output, select the IMA/UNI mode (see Table 1-34) to avoid signals conflict.**
PB(27)	FCC2:TXD[0] or TDMc2:L1TXD	O	FCC2: UTOPIA MPHY Master in IMA/UNI mode or TDMc2 in other mode
PB(28)	Input	I	CONFIG1: Optional configuration input. Set to 0 or 1 by resistors.
PB(29)	FCC2:RxClav	I	FCC2: UTOPIA MPHY Master
PB(30)	FCC2:TxSoc	O	FCC2: UTOPIA MPHY Master
PB(31)	FCC2:RxSoc	I	FCC2: UTOPIA MPHY Master

Table 1-6. CPM Port C Usage

CPM I/O Port	Pin Configuration	Dir	Usage
PC(0)	Output	O	COMCLK_N: selection signal
PC(1)	Input	I	PTMC $\overline{\text{PTENB}}$ signal acquisition or interrupt.
PC(2:3)	FCC2:TxD[3:2]	O	FCC2: UTOPIA MPHY Master
PC(4)	FCC2:RxEnb	O	FCC2: UTOPIA MPHY Master
PC(5)	FCC2:TxClav	I	FCC2: UTOPIA MPHY Master
PC(6)	FCC1:RxAddr[2]	I(O)	FCC1: UTOPIA Slave (or Master)
PC(7)	FCC1:TxAddr[2]	I(O)	FCC1: UTOPIA Slave (or Master)
PC(8)	Output	O	SELSYNC: selection signal
PC(9)	Output	O	LTMODE: selection signal
PC(10:11)	FCC2:RxD[3:2]	I	FCC2: UTOPIA MPHY Master
PC(12)	FCC1:RxAddr[1]	I(O)	FCC1: UTOPIA Slave (or Master)
PC(13)	FCC1:TxAddr[1]	I(O)	FCC1: UTOPIA Slave (or Master)

Table 1-6. CPM Port C Usage (cont)

CPM I/O Port	Pin Configuration	Dir	Usage
PC(14)	FCC1:RxAddr[0]	I(O)	FCC1: UTOPIA Slave (or Master)
PC(15)	FCC1:TxAddr[0]	I(O)	FCC1: UTOPIA Slave (or Master)
PC(16)	CLK16	I	Fast Ethernet TX Clock
PC(17)	CLK15	I	25MHz reference clock
PC(18)	CLK14	I	Fast Ethernet RX Clock
PC(19)	CLK13	I	TDMa2 Clock
PC(20)	CLK12	I	FCC1 UTOPIA RxClk, externally connected to PD(17) through a tristate buffer.
PC(21)	Output	O	Not used
PC(22)	CLK10	I	FCC1 UTOPIA TxClk, externally connected to PD(17): Input or BRG2
PC(23)	Output	O	Fast Ethernet MDC
PC(24)	Input/Output	I/O	Fast Ethernet MDIO
PC(25)	BRG4	O	12.5MHz clock to the QuadFALC
PC(26)	Output	O	QuadFALC reset (0=reset)
PC(27)	CLK5	I	TDMc1 Clock
PC(28)	Output	O	MT90220/1 reset (0 = reset)
PC(29)	CLK3	I	TDMc2 Clock
PC(30)	Output	O	U1MASTER_N selection signal
PC(31)	CLK1	I	TDMa1 Clock

Table 1-7. CPM Port D Usage

CPM I/O Port	Pin Configuration	Dir	Usage
PD(4)	Output	O	Unused
PD(5)	Input	I	CONFIG2: Optional configuration input. Set to 0 or 1 by resistors.
PD(6)	Output	O	IMAMODE selection signal
PD(7)	FCC1:TxAddr[3]	I(O)	FCC1: UTOPIA Slave (or Master)
PD(8)	SMC1:SMRxD	I	SMC1 UART
PD(9)	SMC1:SMTxD	O	SMC1 UART
PD(10:11)	FCC2:RxD[1:0]	I	FCC2: UTOPIA MPHY Master
PD(12)	Output	O	$\overline{\text{PTMODE}}$: Selection signal
PD(13)	Output	O	$\overline{\text{SWMODE}}$: Selection signal
PD(14)	Output	O	PT4TO3: Selection signal
PD(15)	Output	O	PT2TO1: Selection signal

Table 1-7. CPM Port D Usage (cont)

CPM I/O Port	Pin Configuration	Dir	Usage
PD(16)	Output	O	Not used
PD(17)	BRG2	I/O	Not used (input) when FCC1 is in slave mode. Provides UTOPIA RxCLK and TxCLK when FCC1 is in master mode. Directly connected to PC(22):CLK10, and indirectly connected to PC(20):CLK12.
PD(18)	FCC2:RxAddr[3]	O	FCC2: UTOPIA MPHY Master
PD(19)	FCC2:TxAddr[3]	O	FCC2: UTOPIA MPHY Master
PD(20)	TDMa2:L1RSYNC	I	TDMa2
PD(21)	TDMa2:L1RXD	I	TDMa2
PD(22)	TDMa2:L1TXD	O	TDMa2
PD(23)	Output	O	LXT971A reset (0 = reset)
PD(24)	Output	O	User-programmable front panel yellow LED (LED5) command
PD(25)	Output	O	User-programmable board green LED (LED0) command
PD(26)	TDMc1:L1RSYNC	I	TDMc1
PD(27)	TDMc1:L1RXD	I	TDMc1
PD(28)	TDMc1:L1TXD	O	TDMc1
PD(29)	FCC1:RxAddr[3]	I(O)	FCC1: UTOPIA Slave (or Master)
PD(30)	FCC2:TxEnb	O	FCC2: UTOPIA MPHY Master
PD(31)	Output	O	CAM reset (0 = reset)

The unused I/O ports must be configured as general purpose outputs (the logical level does not matter) in order to avoid their electrical level to float.

CPM TDM Busses

The CPM in the MPC8260 features two Serial Interfaces, each one featuring four TDM busses, for a total of eight TDM busses (TDMa1 ... TDMd1 and TDMa2 ... TDMd2). Two TDM busses are used on each of the two Serial Interfaces. Each of the four TDM busses has its own clock and frame pulse with a 2.048 Mb/s bit rates. These clock and frame pulse are common for receive and transmit sections (configured in the SIxxMR registers).

Table 1-8. CPM SI1 TDM Bus Wiring

TSA Signal	TDMa1	TDMc1	Dir
L1RCLK	PC(31)	PC(27)	I
L1RSYNC	PA(6)	PD(26)	I

Table 1-8. CPM SI1 TDM Bus Wiring (cont)

TSA Signal	TDMa1	TDMc1	Dir
L1RXD	PA(8)	PD(27)	I
L1TXD	PA(9)	PD(28)	O

Table 1-9. CPM SI2 TDM Bus Wiring

TSA Signal	TDMa2	TDMc2	Dir
L1RCLK	PC(19)	PC(29)	I
L1RSYNC	PD(20)	PB(24)	I
L1RXD	PD(21)	PB(26)	I
L1TXD	PD(22)	PB(27)	O

CPM UTOPIA Busses

The MPC8260 CPM includes three Fast Communications Controllers (FCCs), two of which are able to handle ATM traffic (up to 155 Mbps full duplex) and control their own UTOPIA bus.

FCC1 UTOPIA bus

FCC1 is dedicated to handle the ATM traffic to/from the Host through the PMC P3 connector. Its UTOPIA bus is configured in multi PHY slave mode (FCC1 is considered as an ATM PHY), or in 'normal' master mode.

As per the PTMC specification, the UTOPIA bus operates in multiplexed polling mode; Rx signals are outputs and Tx signals are inputs of the PowerQuicc II. Depending on the operating mode the clocks configuration must be changed.

- UTOPIA signals, **Slave mode**

UTOPIA clocks are provided by the PMC P3 connector. As a result, PD(17) port shall be configured as input (not used). PC(30) port (U1MASTER_N signal) shall be configured as output set to 1 to disable the connection between PD(17) and PC(20).

Table 1-10. FCC1 UTOPIA Slave Bus on the CPM

CPM I/O Configuration	CPM I/O Port	Port Dir	Signal on P3	P3 Connector Pin
CLK12 (FCC1 TXCLK)	PC(20)	I	RXCLK	43
TxSOC	PA(29)	O	RXSOC	64
TxEnb	PA(31)	I	RXENB_N	16
TxClav	PA(30)	O	RXCLAV	18
TxD[0:7]	PA(25:18)	O	RXD[0:7]	60,58,54,52,48, 46,42,40

Table 1-10. FCC1 UTOPIA Slave Bus on the CPM

CPM I/O Configuration	CPM I/O Port	Port Dir	Signal on P3	P3 Connector Pin
TxAddr[0:3]	PC(15,13,7),PD(7)	I	RXADR[0:3]	34,30,24,7
CLK10 (FCC1 RXCLK)	PC(22)	I	TXCLK	25
RxSOC	PA(27)	I	TXSOC	1
RxEnb	PA(28)	I	TXENB_N	22
RxClav	PA(26)	O	TXCLAV	5
RxD[0:7]	PA(10:17)	I	TXD[0:7]	61,59,55,53,49, 47,37,35
RxAddr[0:3]	PC(14,12,6),PD(29)	I	TXADR[0:3]	29,28,19,17

- UTOPIA signals, **Master mode**

UTOPIA clocks are provided by the CPM port PD(17), which shall be configured as BRG2:BRGO. PC(30) port (U1MASTER_N signal) must be configured as output set to 0, in order to establish connection between PD(17) and PC(20).

Table 1-11. FCC1 UTOPIA Master Bus on the CPM

CPM I/O Configuration	CPM I/O Port	Port Dir	Signal on P3	P3 Connector Pin
CLK12 (FCC1 TXCLK)	PC(20)	I	RXCLK	43
TxSOC	PA(29)	O	RXSOC	64
TxEnb	PA(31)	O	RXENB_N	16
TxClav	PA(30)	I	RXCLAV	18
TxD[0:7]	PA(25:18)	O	RXD[0:7]	60,58,54,52,48, 46,42,40
TxAddr[0:3]	PC(15,13,7),PD(7)	O	RXADR[0:3]	34,30,24,7
CLK10 (FCC1 RXCLK)	PC(22)	I	TXCLK	25
RxSOC	PA(27)	I	TXSOC	1
RxEnb	PA(28)	O	TXENB_N	22
RxClav	PA(26)	I	TXCLAV	5
RxD[0:7]	PA(10:17)	I	TXD[0:7]	61,59,55,53,49, 47,37,35
RxAddr[0:3]	PC(14,12,6),PD(29)	O	TXADR[0:3]	29,28,19,17

FCC2 UTOPIA bus

FCC2 is used for the ATM Inverse Multiplexer (IMA). The IMA device appears on the UTOPIA bus as multiple ATM PHY interfaces (as many as the number of IMA groups of T1/E1/J1 lines). It can appear as up to eight PHY interfaces if the eight T1/E1/J1 lines are independent (ATM over T1/E1/J1 without IMA). For multi PHY (MPHY) operation, multiplexed polling is used: the TxClav and RxClav signals are common to all PHYs, and the ATM controller (FCC2) polls all active PHYs by setting the address bits.

The UTOPIA Rx and Tx Clocks are generated by an external 25.00MHz oscillator. This frequency is provided on PC(17) which should be configured as CLK15.

Table 1-12. FCC2 UTOPIA Bus on the CPM

CPM I/O Configuration	CPM I/O Port	Dir
CLK15 (FCC2 TxClk & RxClk)	PC(17)	I
TxSOC	PB(30)	O
TxEnb	PD(30)	O
TxClav	PC(5)	I
TxD[0:7]	PB(27,26),PC(3,2), PB(25:22)	O
TxAddr[0:3]	PA(2:0),PD(19)	O
RxSOC	PB(31)	I
RxEnb	PC(4)	O
RxClav	PB(29)	I
RxD[0:7]	PD(11,10),PC(11,10), PB(18:21)	I
RxAddr[0:3]	PA(3:5),PD(18)	O

Clocks and Baud-Rate Generators

The PowerQUICC II CPM features a bank of clocks that can be selected independently for each device used. However, the choice for each device is limited. In addition to the ports configuration as clock inputs, it is necessary to configure the clock source of each TDM bus and of each communication controller.

Table 1-13. 4539 CPM Bank of Clocks Usage

Clock	CPM I/O Port	Usage
CLK1	PC(31)	TDMa1 Rx & Tx Clock
CLK3	PC(29)	TDMc2 Rx & Tx Clock
CLK5	PC(27)	TDMc1 Rx & Tx Clock
CLK10	PC(22)	FCC1 UTOPIA RxClk, externally connected to PD(17) which is configured as input in slave mode and as BRG2:BRGO in master mode.

Table 1-13. 4539 CPM Bank of Clocks Usage (cont)

Clock	CPM I/O Port	Usage
CLK12	PC(20)	FCC1 UTOPIA TxClk, indirectly connected to PD(17) which is configured as BRG2:BRGO in master mode. The connection is enabled by PC(30) (U1MASTER_N signal).
CLK13	PC(19)	TDMa2 Rx & Tx Clock
CLK14	PC(18)	FCC3: Ethernet Rx Clock
CLK15	PC(17)	25 MHz for BRG4 and FCC2 UTOPIA clocks
CLK16	PC(16)	FCC3: Ethernet Tx Clock

The CPM features eight Baud-Rate Generators (BRG) which can be assigned to different clocks and timers. Some BRG are assigned to a particular single function, while others can have multiple uses. BRG2 is used only when FCC1 UTOPIA bus is in master mode, otherwise the port should be configured as input.

Table 1-14. 4539 CPM Baud-Rate Generators Usage

BRGs	Source	CPM I/O Port	Usage
BRG1	BRGCLK	Internal	SMC1or PIT
BRG2	BRGCLK	PD(17)	FCC1: UTOPIA master clocks externally connected to PC(22):CLK10 and PC(20):CLK12
BRG4	CLK15	PC(25)	12.5 MHz to the PEB22554
BRG5	BRGCLK	Internal	FCC internal rate
BRG6	BRGCLK	Internal	FCC internal rate
BRG7	BRGCLK	Internal	SMC1

Figure 1-3 shows recommended clock and BRG use. It is not the only possible configuration.

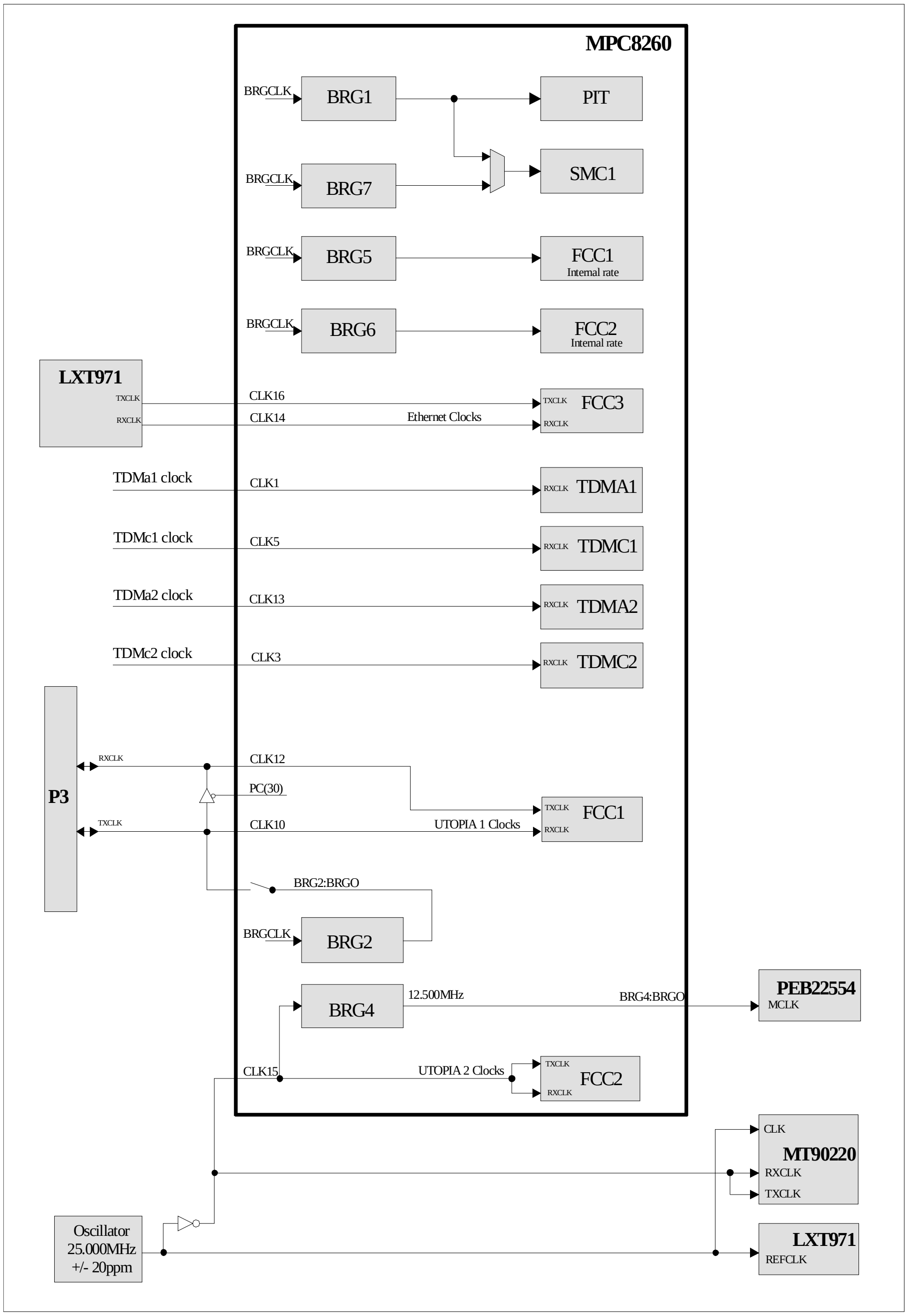


Figure 1-3. Recommended Clocks and BRGs use

Ethernet 10/100BaseT

The FCC3 part of the CPM is used to control an Ethernet 10/100baseT port. An on-board LXT971A Line Interface Unit (LIU) controls the Ethernet interface to RJ45 connector J1. The CPM interface to the LIU is a Media-Independent Interface (MII) bus.

Table 1-15. Ethernet Signals on the CPM

Ethernet Signal	CPM I/O Port	Dir	Description
FE_TXD[0:3]	PB(7:4)	O	Transmit Nibble Data
FE_RXD[0:3]	PB(8:11)	I	Receive Nibble Data
FE_CRS	PB(12)	I	Carrier Sense
FE_COL	PB(13)	I	Collision Detect
FE_TX_EN	PB(14)	O	Transmit Enable
FE_TX_ER	PB(15)	O	Transmit Error
FE_RX_ER	PB(16)	I	Receive Error
FE_RX_DV	PB(17)	I	Receive Data Valid
FE_MDC	PC(23)	O	Management Data Clock
FE_MDIO	PC(24)	I/O	Management Data I/O
FE_TX_CLK	PC(16)	I	Transmit clock
FE_RX_CLK	PC(18)	I	Receive Clock
FE_RESET	PD(23)	O	LXT971A reset control (low active)
FE_INT	IRQ2	I	LXT971A interrupt

Three green Ethernet LEDs, LED6, LED7, and LED8, driven by the LXT971A LED/CFG(1:3) outputs, are provided on the front panel.

TTY Console Serial Port

The SMC1 part of the CPM is used as a simple asynchronous serial port for connection to a TTY console. An on-board RS232 transceiver translates the signals to RS232 electrical levels which are routed to the 2.5mm stereo jack connector J2.

Table 1-16. Asynchronous Console Serial Port Wiring

SMC1 Signal	CPM I/O Port	Dir	J2 Connector
GND	–	–	Ring
TXD	PD(9)	O	Tip
RXD	PD(8)	I	Sleeve

User-Programmable LEDs

Two user-programmable LEDs are provided, a green LED (LED0) on the board and a yellow LED (LED5) on the front panel. They are controlled through CPM I/O ports used as simple outputs. The front panel yellow LED is always illuminated when the PowerQUICC II is in reset state.

.

Table 1-17. User-programmable LED Control Ports

CPM I/O	Signal Name	Description
PD(25)	CPU_LED0	Board user-programmable green LED0 control (LED0). 0=On, 1=Off.
PD(24)	CPU_LED5	Front panel user-programmable yellow LED5 control (LED5). 0=Off, 1=On.

The PCI Bridge

A dedicated PCI bridge, the Tundra PowerSpan, controls the interface between the card and the host 64-bit PCI bus.

The PowerSpan implements all the registers needed by the PCI 2.2 standard, providing the Plug-and-Play capability, as well as the Hot-Swap Friendly capabilities. It supports Target and Master accesses between the PCI bus and the local 60x bus.

It also implements windows and different mechanisms to interface between the PCI host and the card. Exchanges can use the following elements from the PowerSpan:

- Runtime registers (mailboxes, doorbells, semaphores)
- Four memory windows from the PCI memory space to the Local memory space
- Eight memory windows from the Local space to the PCI memory or I/O space
- Four independent bidirectional DMA engines
- An I²O messaging unit

This chip implements FIFO buffers for all the exchanges through the different windows between the two buses, so that the local bus clock is independent from the PCI bus clock.

All the PowerSpan internal registers are grouped in a 4 KB memory space that can be accessed by the PCI host and the local processor. On the PCI side, the PCI base address of this register space is defined by PCI configuration register PCIBAR1 (offset 0x14). On the local side, the local base address has been conventionally fixed to 0xF0020000.

The PowerSpan internal register set can be split into six different functional groups:

- PCI configuration registers (these registers, defined by the PCI specification, can be accessed in the standard PCI configuration space or in the local PowerSpan internal registers space)
- PCI registers
- Processor bus registers
- DMA registers

- Miscellaneous registers (Mailboxes, Doorbells, Interrupts, Semaphores)
- I²O messaging registers

The details of the PowerSpan registers can be found by consulting the PowerSpan data sheet available in the Tundra Web site:

http://www.tundra.com

PowerSpan PCI Configuration Registers

As defined by the PCI specification, the communications controller has a unique 256-byte memory space, called configuration space, that maps all the PCI configuration registers. Access to this area is done through CompactPCI Configuration Read and PCI Configuration Write cycles.

Table 1-18. PCI Configuration Registers

Register	Size	PCI cfg Address	Local Offset	Description
PCIIDR	32	0x00	0x00	Vendor and Device Identification
PCICR	16	0x04	0x06	PCI Command
PCISR	16	0x06	0x04	PCI Status
PCIREV	8	0x08	0x0B	Revision Identification
PCICCR	24	0x09	0x08	Class Code
PCICLSR	8	0x0C	0x0F	Cache Line Size
PCILTR	8	0x0D	0x0E	Master Latency Timer
PCIHTR	8	0x0E	0x0D	Header Type
PCIBISTR	8	0x0F	0x0C	Built-in Self Test
PCIBAR0	32	0x10	0x10	I²O registers base address
PCIBAR1	32	0x14	0x14	PowerSpan internal registers base address
PCIBAR2	32	0x18	0x18	PCI-to-Local Window 0 PCI base address
PCIBAR3	32	0x1C	0x1C	PCI-to-Local Window 1 PCI base address
PCIBAR4	32	0x20	0x20	PCI-to-Local Window 2 PCI base address
PCIBAR5	32	0x24	0x24	PCI-to-Local Window 3 PCI base address
PCISVID	16	0x2C	0x2E	Subsystem Vendor ID
PCISID	16	0x2E	0x2C	Subsystem Device ID
PCICAP	8	0x34	0x37	Capabilities pointer
PCIILR	8	0x3C	0x3F	Interrupt Line
PCIIPR	8	0x3D	0x3E	Interrupt Pin
PCIMGR	8	0x3E	0x3D	Minimum Grant
PCIMLR	8	0x3F	0x3C	Maximum Latency

Table 1-18. PCI Configuration Registers (cont)

Register	Size	PCI cfg Address	Local Offset	Description
HS_CSR	32	0xE4	0xE4	Hot Swap Control and Status Register
VPD_CSR	32	0xE8	0xE8	PCI Vital Product Control/Status Register
VPD_D	32	0xEC	0xEC	PCI Vital Product Data-Data Register

These registers are initialized with fixed reset values or with values stored in the I²C serial EEPROM and then used by the PCI HOST, mainly during the Power-On Self Tests (POST) for plug and play functionality or later by the operating system for enumeration.

The PCI Vendor ID equals 0x107E, and the PCI Device ID equals 0x90A0.

PowerSpan PCI Registers

These registers are used to define the parameters of the PCI-to-Local windows. They are mapped in the PCI memory space (base address defined in PCI configuration register 0x14 PCIBAR1) and in the local space for the local processor (base address 0xF0020000).

Table 1-19. PowerSpan PCI Registers

Offset	Register	Description
0x100	P1_TI0_CTL	PCI Target Image 0 Control Register
0x104	P1_TI0_TADDR	PCI Target Image 0 Translation Address Register
0x110	P1_TI1_CTL	PCI Target Image 1 Control Register
0x114	P1_TI1_TADDR	PCI Target Image 1 Translation Address Register
0x120	P1_TI2_CTL	PCI Target Image 2 Control Register
0x124	P1_TI2_TADDR	PCI Target Image 2 Translation Address Register
0x130	P1_TI3_CTL	PCI Target Image 3 Control Register
0x134	P1_TI3_TADDR	PCI Target Image 3 Translation Address Register
0x150	P1_ERRCS	PCI Bus error control and status register
0x154	P1_AERR	PCI Address error log register
0x160	P1_MISC_CSR	PCI Miscellaneous Control and Status Register
0x164	P1_ARB_CTRL	PCI Bus Arbiter Control Register

PowerSpan Processor Bus Registers

These registers are used to define the parameters of the local to PCI windows. They are mapped in the PCI memory space (base address defined in PCI configuration register 0x14 PCIBAR1) and in the local space for the local processor (base address 0xF0020000).

Table 1-20. PowerSpan Processor Bus Registers

Offset	Register	Description
0x200	PB_SI0_CTL	Processor Bus Slave Image 0 Control Register
0x204	PB_SI0_TADDR	Processor Bus Slave Image 0 Translation Address Register
0x208	PB_SI0_BADDR	Processor Bus Slave Image 0 Base Address Register
0x210	PB_SI1_CTL	Processor Bus Slave Image 1 Control Register
0x214	PB_SI1_TADDR	Processor Bus Slave Image 1 Translation Address Register
0x218	PB_SI1_BADDR	Processor Bus Slave Image 1 Base Address Register
0x220	PB_SI2_CTL	Processor Bus Slave Image 2 Control Register
0x224	PB_SI2_TADDR	Processor Bus Slave Image 2 Translation Address Register
0x228	PB_SI2_BADDR	Processor Bus Slave Image 2 Base Address Register
0x230	PB_SI3_CTL	Processor Bus Slave Image 3 Control Register
0x234	PB_SI3_TADDR	Processor Bus Slave Image 3 Translation Address Register
0x238	PB_SI3_BADDR	Processor Bus Slave Image 3 Base Address Register
...	...	...
0x270	PB_SI7_CTL	Processor Bus Slave Image 7 Control Register
0x274	PB_SI7_TADDR	Processor Bus Slave Image 7 Translation Address Register
0x278	PB_SI7_BADDR	Processor Bus Slave Image 7 Base Address Register
0x280	PB_REG_BADDR	Processor Bus Register Image Base Address Register
0x290	PB_CONF_INFO	Processor Bus PCI Configuration Cycle Information Register
0x294	PB_CONF_DATA	Processor Bus PCI Configuration Cycle Data Register
0x2A0	PB_P1_IACK	Processor Bus to PCI Interrupt Acknowledge Cycle Register
0x2B0	PB_ERRCS	Processor Bus Error Control and Status Register
0x2B4	PB_AERR	Processor Bus Address Error Log Register
0x2C0	PB_MISC_CSR	Processor Bus Miscellaneous Control and Status Register
0x2D0	PB_ARB_CTRL	Processor Bus Arbiter Control Register

PowerSpan DMA Registers

These registers are used to control the four bidirectional DMA engines provided in the PowerSpan. They are mapped in the PCI memory space (base address defined in PCI configuration register 0x14 PCIBAR1) and in the local space for the local processor (base address 0xF0020000).

Table 1-21. PowerSpan DMA Registers

Offset	Register	Description
0x304	DMA0_SRC_ADDR	DMA0 Source Address Register
0x30C	DMA0_DST_ADDR	DMA0 Destination Address Register
0x314	DMA0_TCR	DMA0 Transfer Control Register
0x31C	DMA0_CPP	DMA0 Command Packet Pointer Register
0x320	DMA0_ GCSR	DMA0 General Control Register
0x324	DMA0_ATTR	DMA0 Attributes Register
0x334	DMA1_SRC_ADDR	DMA1 Source Address Register
0x33C	DMA1_DST_ADDR	DMA1 Destination Address Register
0x344	DMA1_TCR	DMA1 Transfer Control Register
0x34C	DMA1_CPP	DMA1 Command Packet Pointer Register
0x350	DMA1_ GCSR	DMA1 General Control Register
0x354	DMA1_ATTR	DMA1 Attributes Register
0x364	DMA2_SRC_ADDR	DMA2 Source Address Register
0x36C	DMA2_DST_ADDR	DMA2 Destination Address Register
0x374	DMA2_TCR	DMA2 Transfer Control Register
0x37C	DMA2_CPP	DMA2 Command Packet Pointer Register
0x380	DMA2_ GCSR	DMA2 General Control Register
0x384	DMA2_ATTR	DMA2 Attributes Register
0x394	DMA3_SRC_ADDR	DMA3 Source Address Register
0x39C	DMA3_DST_ADDR	DMA3 Destination Address Register
0x3A4	DMA3_TCR	DMA3 Transfer Control Register
0x3AC	DMA3_CPP	DMA3 Command Packet Pointer Register
0x3B0	DMA3_ GCSR	DMA3 General Control Register
0x3B4	DMA3_ATTR	DMA3 Attributes Register

PowerSpan Miscellaneous Registers

This group of registers includes several configuration registers for the interrupt functions, as well as various runtime registers: mailboxes, doorbells, interrupt control/status, and semaphores. They are mapped in the PCI memory space (base address defined in PCI configuration register 0x14 PCIBAR1) and in the local space for the local processor (base address 0xF0020000).

Table 1-22. PowerSpan Miscellaneous Registers

Offset	Register	Description
0x400	MISC_CSR	Miscellaneous Control/Status Register
0x404	CLOCK_CTL	Clock Control Register
0x408	I²C_CSR	I²C Interface Control and Status Register
0x40C	RST_CSR	Reset Control and Status Register
0x410	ISR0	Interrupt Status Register 0
0x414	ISR1	Interrupt Status Register 1
0x418	IER0	Interrupt Enable Register 0
0x41C	IER1	Interrupt Enable Register 1
0x420	IMR_MBOX	Interrupt Map Register: Mailbox
0x424	IMR_Db	Interrupt Map Register: Doorbell
0x428	IMR_DMA	Interrupt Map Register: DMA
0x42C	IMR_HW	Interrupt Map Register: Hardware
0x430	IMR_P1	Interrupt Map Register: PCI
0x438	IMR_PB	Interrupt Map Register: Processor Bus
0x43C	IMR_PB2	Interrupt Map Register 2: Processor Bus
0x440	IMR_MISC	Interrupt Map Register: Miscellaneous
0x444	IDR	Interrupt Direction Register
0x450–0x46C	MBOX0 – MBOX7	Mailbox 0 to 7 Registers
0x470	SEMA0	Semaphore 0 Register
0x474	SEMA1	Semaphore 1 Register

PowerSpan I²O Registers

The PowerSpan includes I²O messaging queues controlled by several registers. These registers are mapped in two places in the PCI memory space: at the base address defined in PCI configuration register 0x10 PCIBAR0 and in the PowerSpan internal register space (base address defined in PCI configuration register 0x14 PCIBAR1). They are also mapped in the local space for the local processor (base address 0xF0020000).

Table 1-23. PowerSpan I²O Registers

Offset	Register	Description
0x500	PCI_TI2O_CTL	PCI I²O Target Image Control Register
0x504	PCI_TI2O_TADDR	PCI I²O Target Image Translation Address Register
0x508	I2O_CSR	I2O Control and Status Register
0x50C	I2O_QUEUE_BS	I²O Queue Base Address
0x510	IFL_BOT	I²O Inbound Free List Bottom Pointer Register
0x514	IFL_TOP	I²O Inbound Free List Top Pointer Register
0x518	IFL_TOP_INC	I²O Inbound Free List Top Pointer Increment Register
0x51C	IPL_BOT	I²O Inbound Post List Bottom Pointer Register
0x520	IPL_BOT_INC	I²O Inbound Post List Bottom Pointer Increment Register
0x524	IPL_TOP	I²O Inbound Post List Top Pointer Register
0x528	OFL_BOT	I²O Outbound Free List Bottom Pointer Register
0x52C	OFL_BOT_INC	I²O Inbound Free List Bottom Pointer Increment Register
0x530	OFL_TOP	I²O Outbound Free List Top Pointer Register
0x534	OPL_BOT	I²O Outbound Post List Bottom Pointer Register
0x538	OPL_TOP	I²O Outbound Post List Top Pointer Register
0x53C	OPL_TOP_INC	I²O Outbound Post List Top Pointer Increment Register
0x540	HOST_OIO	I²O Host Outbound Index Offset Register
0x544	HOST_OIA	I²O Host Outbound Index Alias Register
0x548	IOP_OI	I²O IOP Outbound Index Register
0x54C	IOP_OI_INC	I²O IOP Outbound Index Increment Register

Interrupt Pins and Doorbells Usage

The PowerSpan provides one interrupt pin on the PCI side (–INTA) and six other interrupt pins (–INT0 to –INT5) on the local side. On the 4539, only –INTA and –INT0 are used for true interrupt functions. The five other pins are used as I/O pins to control several signals.

The PowerSpan offers the ability to map any interrupt source to any interrupt pin. This capability is used to divert interrupts –INT1 to –INT5 from a pure interrupt function usage.

Interrupt pins –INT1 to –INT4 are configured as output ports and conventionally associated with doorbell bits DB3 to DB6 in the PowerSpan. Each doorbell bit, when set, will activate its corresponding interrupt pin (level = 0), and when reset will deactivate it (level =1).

Interrupt pin –INT5 is used as an input. Its state can be read in the PowerSpan Interrupt status register. As an interrupt source, it was decided not to map it to any interrupt output, so it will not generate interrupts. As an interrupt output pin, it was decided not to associate it to any interrupt source.

Interrupt pins –INTA and –INT0, used for true interrupt functions, have several other interrupt sources, such as Mailboxes interrupts, DMA interrupts, I²O interrupts, and PCI bus or local bus error interrupts. They are conventionally associated with a doorbell bit for software activation capability.

Table 1-24. PowerSpan Interrupt Pin Usage

PowerSpan Pin	Doorbell	Dir	Signal Name	Usage
–INTA	DB0	O	–INTA	Interrupt from the 4539 to the PCI Host controlled by software by the PowerQUICC II.
–INT0	DB2	O	–INTPSP	Interrupt from the PowerSpan to the PowerQUICC II interrupt input –IRQ1
–INT1	DB3	O	–INTRST	Flash mode: When this output is set to 0, the PowerQUICC II is maintained in reset, its busses are tri-stated, the 60x bus is parked on the PowerSpan and the special address translation mode on the FLASH memory is enabled.
–INT2	DB4	O	–PSP_INT2/ –HRESET/ ISPDI	When this pin is set to 0, the PowerQUICC II hardware reset signal is activated. For ISP EPLD programming, this pin serves also as "Serial Data In" signal.
–INT3	DB5	O	–PSP_INT3/ –SRESET/ ISPMODE	When this pin is set to 0, the PowerQUICC II soft reset signal is activated. For ISP EPLD programming, this pin serves also as "Mode" signal.
–INT4	DB6	O	–PSP_INT4/ ISPCK	For ISP EPLD programming, this pin serves as "Serial Clock" signal.
–INT5	–	I	–PSP_INT5/ ISPDO	For ISP EPLD programming, this pin serves as "Serial Data Out" signal.

PCI to Local Interrupt (ATN)

PowerSpan Interrupt pin –INT0 is used to control the PCI-to-Local Interrupt (renamed ATN (Attention) in the software examples).

Local to PCI Interrupt (–INTA)

The PowerQUICC II can generate an interrupt toward the PCI Host by setting a doorbell bit. Conventionally, doorbell bit 0 has been dedicated to this task, and has been associated with the PCI interrupt pin –INTA in the PowerSpan Interrupt Map registers.

Hardware and Software Resets Through the PowerSpan

PowerSpan interrupt pins –INT2 and –INT3 are used as output ports to control the MPC8260 hardware reset signal –HRESET and software reset signal –SRESET respectively. The PowerSpan Interrupt Map registers must have previously been correctly initialized.

During a power-up sequence, –HRESET and –SRESET are first activated and then deactivated once the PCI bus reset signal is deactivated. This allows the PowerQUICC II to boot without any host intervention, just after the end of the PCI reset.

For a normal utilization, the card should be reset by the PCI host (if needed) using only the –SRESET signal. The –HRESET signal is used for special cases, such as FLASH memory reprogramming through PCI.

Local Space Access From PCI Memory Space

The PowerSpan provides four memory windows from the PCI memory space to the Local memory space. Each window can map a programmable size of the local memory space into the PCI memory space. The size of the windows and their enabling is set in PowerSpan registers P1_TIx_CTL, and preset at power-up by the serial EEPROM.

In the 4539 communications controller, only two windows are enabled. They have been set to a relatively small size (2 MB and 512 KB), in order to comply with high availability operating system requirements. These operating systems are able to do dynamic PCI re-configuration during hot swap, only if the total memory size requested by the board is not too big.

The PCI base address of each window is defined in a PCI configuration register. Window 0 base address is set in P1_BAR2, Window 1 base address is set in P1_BAR3, etc. Each window can be moved on the local memory space, using a PowerSpan translation register (P1_TIx_TADDR), so that even a small window can allow access to any part of the 4 GB of local memory space.

During a PCI host access to the local space, the high-order address bits of the local bus must be generated by the PowerSpan (as defined in the PowerSpan P1_TI0_TADDR register) and the low-order address bits of the local bus come from the PCI address. This mode is called “Address Translation” in the PowerSpan Manual.

NOTE

A PowerSpan PCI-to-Local window must have been enabled in the I²C serial EEPROM, in order to allow the CompactPCI host to detect it at system power-on or after the “Hot Swap insertion” of the board and to map it in the PCI space. The corresponding PowerSpan register “PCI Target Image Control Register” must also have been initialized with the “Image Enable” bit set (IMG_EN=1) and the address translation mechanism enabled (TA_EN=1).

Figure 1-4 on page 28 illustrates the PCI-to-Local window mechanism.

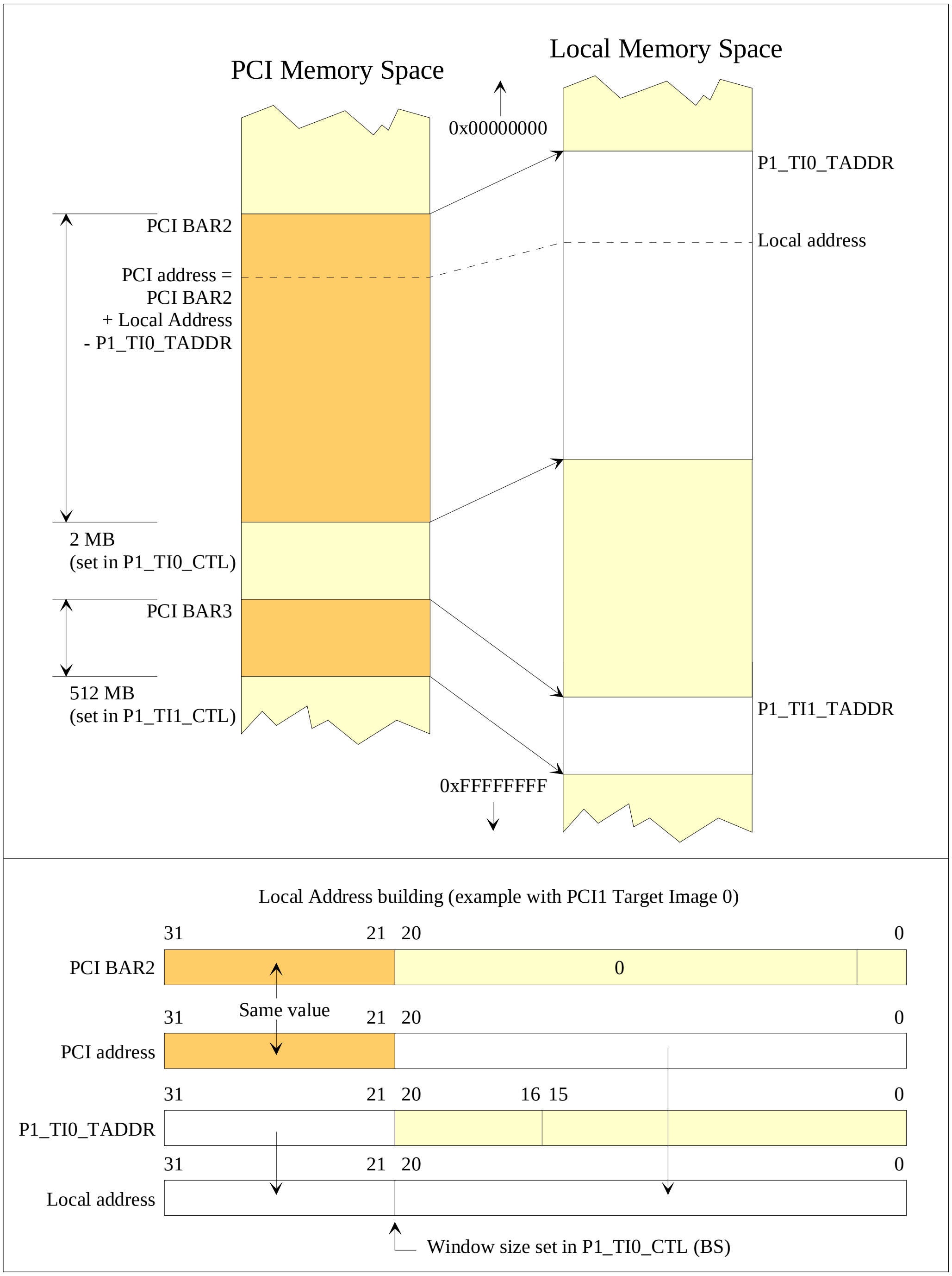


Figure 1-4. Local Space Access From PCI Memory Space

When the processor is running, the PCI bus can access all the elements connected to the local bus, except the FLASH boot memory. The accessible elements are the main SDRAM memory (the processor's SDRAM memory controller must be initialized), the Connection SDRAM memory (the processor's SDRAM memory controller must be initialized), the CAM memory, the processor Dual-Port RAM (DPRAM), the QuadFALC framers, the IMA device, etc. (the processor must have its chip selects programmed). The local space mapping is the same as when accessed by the processor (see *Local Space Mapping* on page 5).

It is not possible to have access to the entire FLASH device when the processor is running, because the FLASH device is an 8-bit data bus device connected to the 64-bit-only local bus of the PowerSpan. Only bytes modulo 8 are reachable.

This problem has been neutralized for the other non-64-bit peripherals, by tying their peripheral address bits 0 to N to local address bits 3 to N+3 respectively, so that all their registers can be accessed on byte lane 0, at consecutive modulo 8 addresses.

When the processor is in the reset state, its memory controllers and chip-select signals are reset, so nothing can be accessed, except the FLASH memory, for which a special mechanism has been implemented.

NOTE

It is possible to write from the PCI bus through a PowerSpan memory window to the MPC8260 internal registers but it is not possible to read them. When the PowerSpan performs a read on the 60x processor bus, it always generates a full 64-bit read. Because most of the MPC8260 internal registers only respond to byte or word read cycles, the returned value is 0xFFFFFFFF.

Access to the FLASH EEPROM Through the CompactPCI Bus

For FLASH in-situ re-programming through the CompactPCI bus, there is a special FLASH mode. In this mode, the PowerQUICC II is reset and logic generates a FLASH chip-select and works around the problem of an 8-bit device connected to a 64-bit-only PowerSpan.

The specific FLASH mode is enabled by one of the PowerSpan interrupt pins (–INT1) used as an output port. When –INT1 is set to 0, the PowerQUICC II is maintained in Hard Reset state (–HRESET=0), its pins are tri-stated, the 60x bus is parked on the PowerSpan, and the following address bus remap is implemented: the FLASH device's low order address bits A(2:0) are driven by the PowerSpan address bits A(24:22). This remap allows full access to the FLASH content through byte lane 0 of the 64-bit 60x bus, provided that some address translation is done by the software.

For more information on FLASH EEPROM device, see *The FLASH EEPROM Boot Memory* on page 34.

PCI Memory Space and I/O Space Access From the PowerQUICC II

The PowerSpan provides eight memory windows from the Local Memory space to the PCI memory space or PCI I/O space. Each window can map a programmable size of the PCI memory or I/O space into the PCI memory space. The size of the windows and their enabling is set in PowerSpan registers PB_SIx_CTL, and preset at power-up: the first window is preset by the serial EEPROM and the seven others are preset as disabled.

On the 4539 board, the serial EEPROM content disables the windows. By default, no Local to PCI window is enabled. It is not recommended using these windows for transfers from or to the PCI local space, because this mechanism can result in bad performance, depending on the other PCI devices tied to the PCI bus.

The local base address of each window is defined in PowerSpan internal register PB_SIx_BADDR. Note that the window must be mapped in the local space between 0xC0000 0000 and 0xCFFF FFFF, in order to comply with the card local space usage. Each window can be moved on the PCI memory or I/O space, using a PowerSpan translation register (PB_SIx_TADDR), so that even a small window can allow access to any part of the PCI space.

During a PowerQUICC II access to the PCI space, the high-order address bits on the PCI bus are generated by the PowerSpan (as defined in the PowerSpan PB_SIx_BADDR register) and the low-order address bits on the PCI bus come from the local address. This mode is called “Address Translation” in the PowerSpan Manual.

NOTE

A PowerSpan Local-to-PCI window must be enabled in the PB_SIx_CTL register. Bits IMG_EN (“Image Enable”) and TA_EN (“address translation enable”) must be set.

Figure 1-5 on page 31 illustrates the Local-to-PCI window mechanism:

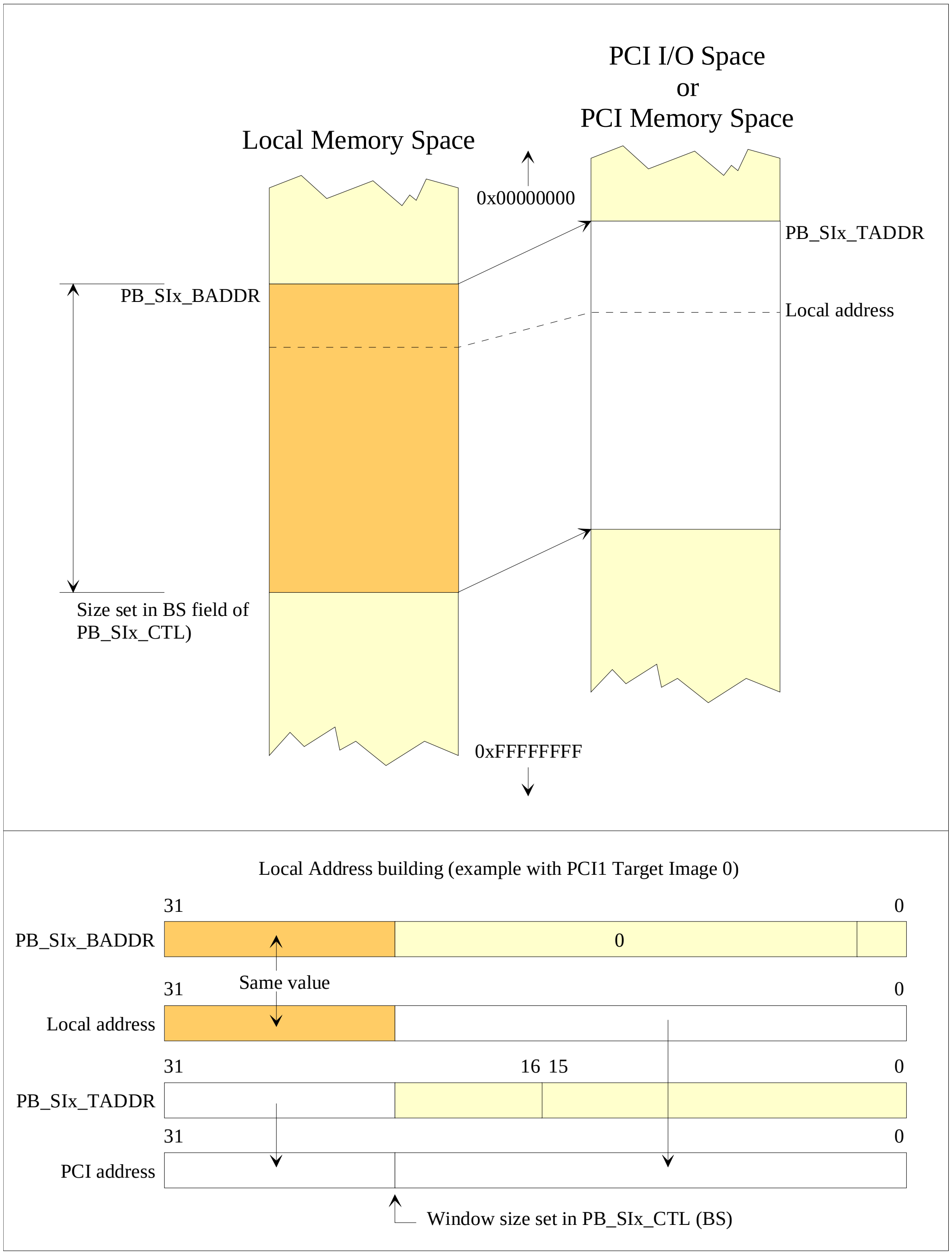


Figure 1-5. PCI I/O or Memory Space Access from Local Space

In-situ EPLDs Programming

Some glue logic is implemented in some EPLDs that can be programmed in-situ through the PCI interface.

These devices keep their programming during power off. So the EPLD should normally be already programmed and the normal user should not be aware of its programming.

The EPLDs are in a daisy-chain configuration, which enables all of them to be programmed at once. They can be programmed in-situ by the PCI host, using some PowerSpan interrupts as I/O pins. A jumper must be placed on board location JP1 to enable the programming (when present, this jumper sets the ISP signal –ISPEN to its active state 0).

Serial EEPROM Connected to the PowerSpan

A 256 bytes I²C serial EEPROM is connected to the PowerSpan. It is used to store the power-up PowerSpan registers initialization values, the "Board Equipment Register", PCI Vital Product Data (VPD) and other Interphase-specific data. There is still some room for other custom data. Table 1-25 shows the memory mapping for the EEPROM:

Table 1-25. Serial EEPROM Mapping

Address	Size	Description
0x00 – 0x3F	64 bytes	PowerSpan registers initial load
0x40 – 0x43	4 bytes	Board Equipment Register
0x44 – 0x8F	76 bytes	VPD and/or Custom data
0x90 – 0xAF	32 bytes	Interphase-Specific production data
0xB0 – 0xFF	80 bytes	Boot Monitor parameters

PowerSPAN Registers Initial Load

Details about the PowerSpan register initial load are described in Table 2-1 on page 72.

Board Equipment Register

The "Board Equipment Register" is a 32-bit word that allows the software to precisely determine the board equipment. The first three bytes are common to several Interphase Boards, so many field values are not possible on the 4539. For instance the 4539 does not have Monarch capability, so the Monarch bit will always be set to 0.

Table 1-26. Board Equipment Register Layout

EEPROM Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x40	MPC_ID				FLASH_SIZE		LSDRAM_SIZE	
0x41	SDRAM_SIZE			CAM_SIZE			0	MONARCH
0x42	BUS_FREQ			0	0	0	0	0
0x43	0	0	0	0	IMA_PRSNT	IMA_SRAM		IMA_DEV

Table 1-27. Hardware Configuration Register Field Descriptions

Field	Description
MPC_ID	Microprocessor identifier. 0000: MPC8260ZU200, 200/133/66 MHz, rev A.1 0001: MPC8260ZU133, 133/133/66 MHz, rev A.1 0010: MPC8260ZU200, 200/133/66 MHz, rev B.3 0011-1111: Reserved for future processor versions
FLASH_SIZE	Flash EEPROM size 00: 1 Mbyte 01: 4 Mbytes 10: 8 Mbytes 11: Reserved for future use
LSDRAM_SIZE	Local SDRAM size. 00: No memory device 01: 8 Mbytes 10: 16Mbytes 11: Reserved for future use
SDRAM_SIZE	Main SDRAM size. 000: 16 Mbytes (not possible on the 4539) 001: 32 Mbytes 010: 64 Mbytes 011: 128 Mbytes 100-111: Reserved for future use
CAM_SIZE	CAM size. 000: No CAM device 001: 4 K x 64 010: 8 K x 64 (not possible on the 4539) 011: 16 K x 64 100: 32 K x 64 (not possible on the 4539) 101-111: Reserved for future use
MONARCH	Monarch capability. 0: Not monarch capable 1: Monarch capable (not possible on the 4539)
BUS_FREQ	Local Bus frequency 000: 50.000 MHz 001: 65.536 MHz (the only frequency available up to now on the 4539) 010: 66.000 MHz 011-111: Reserved for future use
IMA_PRSNT	IMA device presence. 0 : Absent. 1: Present
IMA_SRAM	IMA SRAM size. 00: 64 Kbytes 01: 128 Kbytes 10: 256 KBytes 11: 512 KBytes

Table 1-27. Hardware Configuration Register Field Descriptions (cont)

Field	Description
IMA_DEV	IMA device type. 0: Octal MT90220. 1: Quad MT90221

Vital Product Data (VPD)

No VPD has been defined yet for the 4539.

Interphase-specific Production Data and Boot Monitor Parameters

Additional information concerning Interphase-specific Production Data and Boot Monitor parameters are provided in the *4539 Board Installation and Maintenance Manual* (UG04539-000).

The FLASH EEPROM Boot Memory

The boot memory is a 4Mx8 AMD 29LV033 FLASH EEPROM device, placed in the 60x bus byte lane 0. This non-volatile memory device contains the Reset Configuration Word required by the PowerQUICC II during the power-up phase, the 4539 Interphase Boot Firmware Code, and optionally, your own complete operational code. The FLASH memory is always mapped at address 0xFF800000.

Depending on the FLASH memory size, the mapping of the boot firmware will be different. There are three requirements:

- The Reset configuration must be mapped at the beginning of the FLASH memory.
- The initial vector table must be mapped at address 0xFFF00000. This address is never in the FLASH memory, but it will wrap onto its last MB.
- The FLASH Memory is organized in sectors. The reset configuration word and the vector table must be preserved; therefore their entire sectors will be reserved.

The various elements are/must then be mapped as follows (the FLASH addresses are obtained by masking the local address with the flash size: for a 4 MB flash device it is 0x003FFFFF). The table show two mappings, because the 4 MB Flash memory is aliased twice in the last 8 MB space.

Table 1-28. FLASH EEPROM Mapping

FLASH Addr	1st MAP	2nd MAP	Size	Description
0x000000	0xFF800000	0xFFC00000	0x00000100	The first 64-kbyte sector of the FLASH contains the Hardware Configuration word (at addresses 0xFF800000, 0xFF800008, 0xFF800010, 0xFF800018) and the remaining space is unused. See **Note**.
0x000100	0xFF800100	0xFFC00100	0x0000FE00	Unused byte space.

Table 1-28. FLASH EEPROM Mapping (cont)

FLASH Addr	1st MAP	2nd MAP	Size	Description
0x010000	0xFF810000	0xFFC10000	0x002F0000	Free space for Operational Firmware (47*64 KB)
0x300000	0xFFB00000	0xFFF00000	0x00000100	Unused byte space.
0x300100	0xFFB00100	0xFFF00100	0x000358E0*	4539 Boot Firmware ROM code
0x3359E0*	0xFFB359E0*	0xFFF359E0*		Unused byte space.

Note: Values depend on Boot Firmware size which varies from one version to another – use the MONITOR INFO command to display the actual size.

The FLASH device is normally controlled by the PowerQUICC II memory controller unit using chip-select signal CS0. The PowerQUICC II can read and re-program the FLASH using the AMD algorithms.

The FLASH device is not intended to be accessed through the CompactPCI bus. Because the FLASH device has an 8-bit data bus, and the PowerSpan supports only 64-bit wide devices, its byte lane can only be accessed by the CompactPCI host for addresses that are multiples of 8.

For more information, see *Access to the FLASH EEPROM Through the CompactPCI Bus* on page 29.

QuadFALC T1/E1/J1 Framer

The 4539 Communication Controller includes one QuadFALC device which controls four independent T1/E1/J1 interfaces. For each interface, the QuadFALC includes a framer and an LIU with data and clock recovery, a frame aligner with two frame elastic buffers for receive clock wander and jitter compensation, a signalling controller with an HDLC controller and 64-byte deep FIFOs, and an 8-bit microprocessor interface.

Each line can be independently configured for E1, T1, or J1. The pulse shape for CEPT E1 applications is programmed according to ITU-T G.703:

- Data Coding: HDB3
- Voltage of nominal pulse: 3 V (CCITT G703)
- Return Loss Transmitter: –12 dB (CCITT G703)
- Line Impedance: 120 Ohm

The pulse shape for T1 applications is programmed according to ANSI T1.403:

- Data Coding: B8ZS
- Voltage of nominal pulse: 3 V
- Return Loss Transmitter: –3.5 dB
- Line Impedance: 100 Ohm

The pulse shape for J1 applications is programmed according to ITU-T JT G.703:

- Data Coding: B8ZS
- Voltage of nominal pulse: 3 V (TBV)
- Return Loss Transmitter: –3.5 dB (TBV)
- Line Impedance: 110 Ohm

The QuadFALC includes a flexible clock unit that uses a clock supplied on its MCLK pin.

The QuadFALC MCLK input is connected to a 12.500 MHz +/-20ppm fixed frequency (CPM BRG4) used by the internal DPLL. As a result, the GCM registers must be programmed with the following values:

Table 1-29. GCM Register Programming (MCLK=12.500 MHz)

Register	Value
GCM1	0x2B
GCM2	0x5D
GCM3	0xAC
GCM4	0x89
GCM5	0x07
GCM6	0x15

The QuadFALC has an integrated short-haul and long-haul line interface, comprising a receive equalization network, noise filtering, and programmable Line Build-Outs (LBOs). It implements an integrated Channel Service Unit (CSU) in T1 mode. For each type of LBO, the shape of the transmit pulse must be adjusted through its registers LIM0, LIM2, XPM0, XPM1, and XPM2 in order to comply with FCC 68 or ANSI T1.403. Table 1-30 provides the values in T1 mode for the 4539 hardware (in E1 mode, default values are suitable).

Table 1-30. Transmit Pulse Shape Programming for 6435 RTMs

Line Build-Out	LIM0:EQON	LIM2:LBO2–1	XPM0	XPM1	XPM2
E1 Short Haul	0	–	0x5A	0x03	0x00
E1 Long Haul	1	–	0x5A	0x03	0x00
T1 Short Haul (no CSU)	0	00	0x9E	0xA3	0x01
T1 Long Haul, 0 dB	1	00	0x9E	0xA3	0x01
T1 Long Haul, –7.5 dB	1	01	0xF6	0x02	0x00
T1 Long Haul, –15 dB	1	10	0xB6	0x01	0x00
T1 Long Haul –22.5 dB	1	11	0xB4	0x01	0x00

suitable).

Table 1-31. Transmit Pulse Shape Programming for 6335 RTMs

Line Build-Out	LIM0:EQON	LIM2:LBO2–1	XPM0	XPM1	XPM2
E1 Short Haul	0	–	0xBE	0xA3	0x00
E1 Long Haul	1	–	0xBE	0xA3	0x01

For each line x, the QuadFALC provides four transmit multifunction ports (XPA_x, XPB_x, XPC_x and XPD_x) and four receive multifunction ports (RPA_x, RPB_x, RPC_x and RPD_x). Two of the receive multifunction ports are tied to both sides of a bi-color LED on the front panel. The tables below indicate how they are used on the 4539:

Table 1-32. QuadFALC Multifunction Port Usage

QuadFALC port	Dir	Function	Usage
XPA_x	Input	SYPX	TDM bus Frame synchronization pulse
XPB_x	–	–	(unused)
XPC_x	–	–	(unused)
XPD_x	–	–	(unused)
RPA_x	Input/Output	SYPR/RFM	TDM bus Frame synchronization pulse (the function used depends on the TDM bus configuration. See *TDM Bus Configurations* on page 40.)
RPB_x	–	–	(unused)
RPC_x	Output	$\overline{\text{RFSP}}$ or RMFB	Bi-color LED control (one side)*
RPD_x	Output	$\overline{\text{RFSP}}$ or RMFB	Bi-color LED control (other side)*

*The bi-color LED is green when RPC_x = $\overline{\text{RFSP}}$ and RPD_x = RMFB. The LED is red when RPC_x = RMFB and RPD_x = $\overline{\text{RFSP}}$. The LED is off when RPC_x = $\overline{\text{RFSP}}$ and RPD_x = $\overline{\text{RFSP}}$.

The four bi-color LEDs, LED1, LED2, LED3, and LED4 on the front panel, are connected respectively between RPC_1 and RPD_1, RPC_2 and RPD_2, RPC_3 and RPD_3, RPC_4 and RPD_4. The LEDs are controlled by the software by configuring these output pins as $\overline{\text{RFSP}}$ and RMFB. The LED is green when RPC = $\overline{\text{RFSP}}$ and RPD = RMFB. The LED is red when RPC = RMFB and RPD = $\overline{\text{RFSP}}$. The LED is off when RPC = $\overline{\text{RFSP}}$ and RPD = $\overline{\text{RFSP}}$.

The local processor and the PCI host see the QuadFALC as an 8-bit peripheral including a set of 1024 directly-addressable registers. These registers are placed at contiguous modulo 8 addresses, starting at addresses 0xF008 0000. The QuadFALC controls its own interrupt line to the local processor.

The QuadFALC reset input is controlled by PowerQUICC II CPM I/O port PC(26), (0=reset active).

The QuadFALC controls its own interrupt line to the local processor.

Each line of the QuadFALC framers can be configured independently in Line Termination mode (LT) or in Network Termination mode (NT). In LT mode, the QuadFALC is in slave mode and synchronizes on the lines. In NT mode, the QuadFALC is in master mode and synchronizes on a reference signal provided through connector P4, or P3, or on a free-running internal frequency.

Additional details about the Infineon PEB22554/QuadFALC can be found at Infineon's web site at address:

http://www.infineon.com/products/commics/qfalc.htm

ATM Inverse MUX Device

The 4539 ATM version is equipped with an MT90220 or MT90221 ATM inverse MUX device. The MT90220 controls up to eight T1/E1/J1 lines, the MT90221 controls up to four T1/E1/J1 lines.

The MT90220/1 supports the following two major modes of operation: IMA mode (as defined by the ATM Forum IMA Specification) and User Network Interface (UNI) mode. Up to four IMA groups can be implemented. Any of the eight TDM Interfaces can be assigned dynamically to any of these IMA groups. A different UTOPIA PHY address is assigned to each of the IMA groups. The UNI mode is used to transfer the cells from the UTOPIA interface to a TDM port without any overhead. Up to eight UTOPIA PHY addresses can be supported in UNI mode. The MT90220 also supports a mixed mode where the TDM Interfaces not assigned to an IMA group can be used in UNI mode. The device allows for bandwidth scaleability through the use of the UTOPIA MPHY, Level 2 specification at 25 Mhz.

The implementation of the IMA as per AF-PHY-0086.00 *Inverse Multiplexing for ATM (IMA) Specification Version 1.0* is divided into hardware and software functions. Hardware functions are implemented in the MT90220/1 device and software functions are implemented by the user. Additional hardware functions are included to assist in the collection of statistical information to support MIB implementation.

Hardware functions that are implemented in the MT90220/1 device are:

- UTOPIA Level 2 PHY interface
- Incoming HEC verification and correction (optional)
- Generation of a new HEC byte
- Format outgoing bytes into multi-vendor PCM formats
- Retrieve ATM cells from the incoming multi-vendor PCM format
- Perform cell delineation
- Provide various counters to assist in performance monitoring

Hardware functions implemented by the IMA processor in the MT90220/1 device are:

- Transmit scheduler (one per IMA group)

- Generation of the TX IMA Data Cell Rate
- Generation and insertion of ICP cells called Filler
- Cells and Stuff Cells in IMA mode and Idle Cells in UNI (non-IMA) mode; the ICP cells are programmed by the user and the Filler and Idle cells are pre-defined
- Retrieve and process ICP cells in IMA Mode
- Perform IMA Frame synchronization
- Management of Rx links to be part of the internal re-sequencer when active
- Extraction of Rx IMA Data Cell Rate
- Verification of delays between links
- Perform re-sequencing of ATM cells using external asynchronous Static RAM
- Can accommodate link differential delays with external SRAM memory
- Provide structured interrupt scheme to report various events

The MT90220/1 uses an external memory to temporarily store the received cells until they can be correctly re-ordered for output. The size of the memory defines the maximum delay allowed between the links. When a 512 Kbyte SRAM is used, it allows 276 ms delay for T1/J1 links and 220 ms for E1 links.

The MT90220/1 UTOPIA bus is connected to the CPM FCC2 ATM controller. It is seen as up to eight PHY interfaces by the ATM controller.

The first four TDM lines of the MT90220/1 are connected to the four QuadFALC TDM busses. When the MT90220 is equipped, its last four TDM busses are connected to PMC P4 connector.

The MT90220/1 reset input is controlled by PowerQUICC II CPM I/O port PC(28), (0=reset active).

The MT90220/1 is mapped in the local memory space. The local processor and the PCI host see it as an 8-bit peripheral including a set of 2048 directly addressable registers. These registers are placed at contiguous modulo 8 addresses, starting at address 0xF00C 0000.

The MT90220/1 controls its own interrupt line to the local processor (–IRQ7).

The Ethernet Transceiver

The Intel LXT971A is an IEEE compliant Fast Ethernet transceiver for 100-Base-TX and 10-Base-T applications. It is connected to the PowerQUICC II through a Media-Independent Interface (MII). It features:

- 10-Base-T and 100-Base-TX
- Auto-Negotiation and Parallel Detection
- MII interface with extended register capability
- Robust baseline wander correction performance
- Standard CSMA/CD or Full-Duplex operation
- MDIO management interface

Its management interface is controlled by PowerQUICC ports PC(23) (MDC) and PC(24) (MDIO).

The LXT971A controls its own interrupt line to the local processor (–IRQ2).

The LXT971A reset input is controlled by PowerQUICC II CPM I/O port PD(23) (0=reset).

The LXT971A also includes three programmable LED drivers, which are used to control the LEDs on the faceplate.

Table 1-33. Ethernet LEDs

LXT971A Output	Description
LED/CFG1	Faceplate LED 7 (green)
LED/CFG2	Faceplate LED 8 (green)
LED/CFG3	Faceplate LED 6 (green)

TDM Bus Configurations

General

The 4539 includes a TDM bus multiplex, which allows different TDM bus configurations, obtained by combining four TDM bus topologies and two network configurations.

The four TDM bus topologies are “Direct”, “Switched”, “Pass Through”, and "IMA".

In “**Direct**” mode, data is exchanged between the QuadFALC and the CPM. Two variants exist:

- "**Multiplex Direct**" Mode with one multiplexed TDM bus for the four framers. In that case the four framers have the same rhythm.
- "**Independent Direct**" Mode with one independent TDM bus per framer. Each framer can have its own rhythm which is the same for transmit and receive.

In "**Switched**" mode, on one hand, data is exchanged between the QuadFALC and the first TDM bus on P4, and on the other hand data is exchanged between the CPM TDMa1 bus and the second TDM bus on P4. These two data paths have no connection between each other on the 4539 board. They are routed to the motherboard through PMC connector P4. It is then up to the motherboard to switch the time slots from one TDM bus to the other using, for instance, a H110 switch.

In “**Pass through**” mode, data is exchanged on the one hand between the two first framers and a first CPM TDM bus, and on the other hand between the last two framers and a second CPM TDM bus. One framer of each group acts as a line termination while the other acts as a network termination. This allows applications like line snooping or concurrent treatment by several 4539 boards applied on one line.

In "**IMA**" mode, data is exchanged between the QuadFALC and the MT90220/1 IMA/UNI device. Each TDM bus of the QuadFALC is tied to a TDM bus of the MT90220/1 and has its own independent clocks.

The two network configurations are Line Termination (LT) and Network Termination (NT).

In the LT configuration, the network synchronization comes from the lines: the QuadFALC derives its clocks from the receive rhythm and provides (directly in Direct mode or indirectly in Switched mode) synchronization for the TDM busses.

In the NT configuration, the card, considered as being part of the network, is master of the line rhythm. A network reference synchronization signal must be provided through PMC connector P4 or P3 in order to control the lines rhythm in accordance to the network. If this reference signal is not provided, or is temporarily failing, the card automatically provides a fixed frequency reference.

Figure 1-6 and Figure 1-7 show the general organization of the different clocks. The use and the source of each clock are described in the different modes description. The framer description shown in this chapter is a partial description, refer to the PEB22554 Data Sheet for a full description.

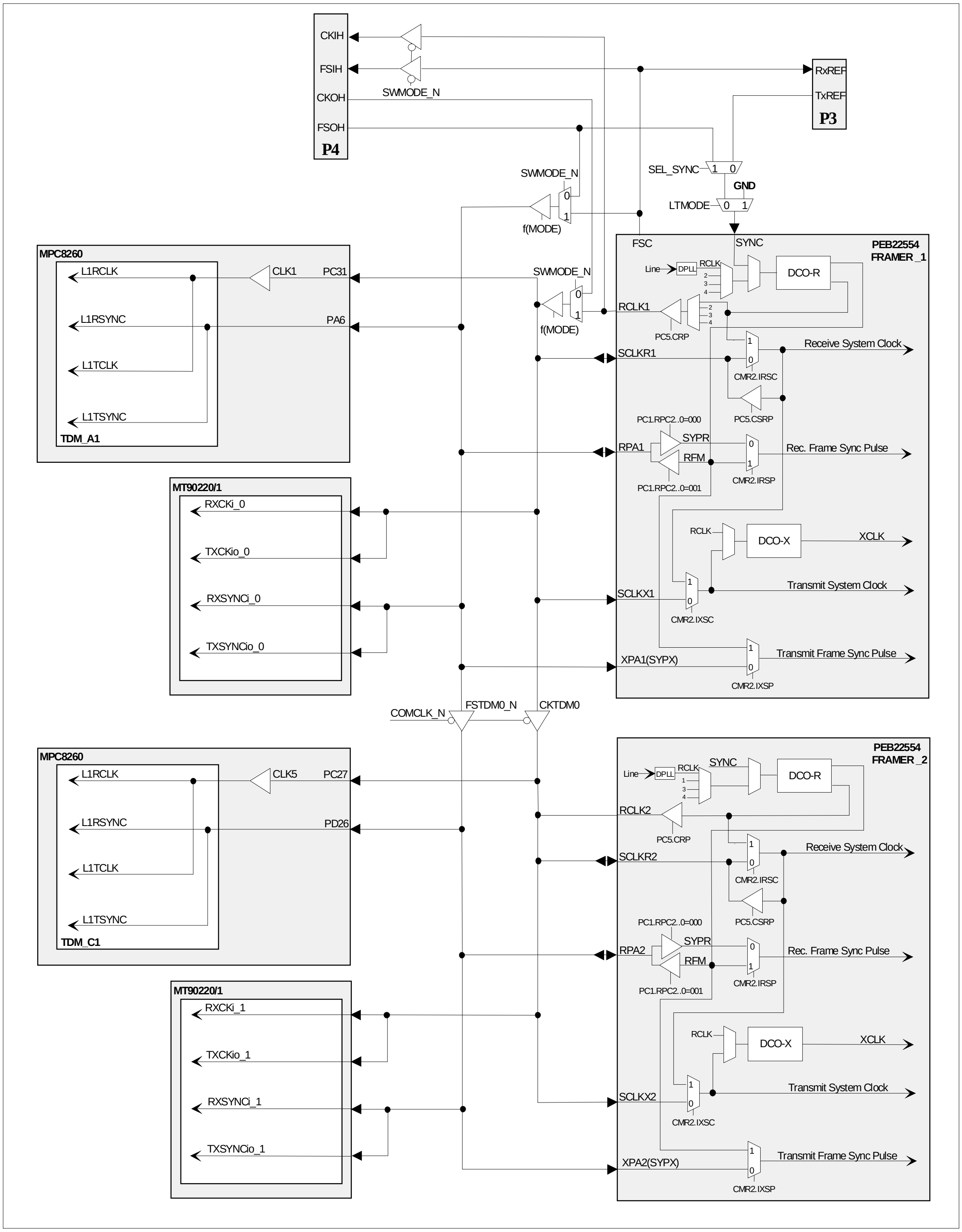


Figure 1-6. General Clock Structure (Framer 1 and 2)

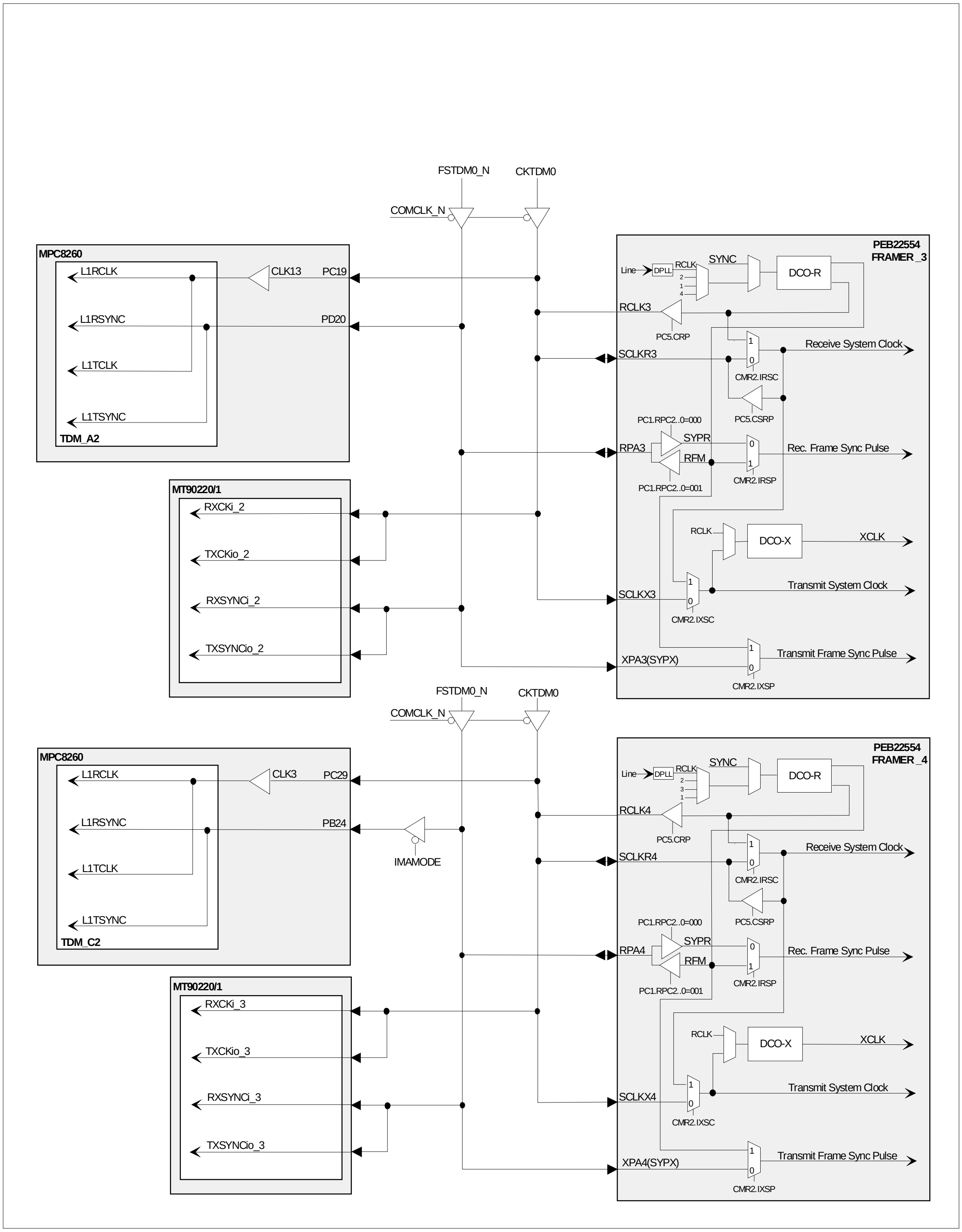


Figure 1-7. General Clock Structure (Framer 3 and 4)

Selection Signals

The PowerQUICC I I controls the TDM configuration through the use of six pins from the CPM I/O. Table 1-34 describes the possible configurations.

Table 1-34. Mode Selection

PC(0) COMCLK_N	PD(6) IMAMODE	PD(12) PTMODE_N	PD(13) SWMODE_N	PD(14) PT4TO3	PD(15) PT2TO1	Configuration
1	0	1	1	0	0	Multiplex Direct Mode.
1	0	1	1	1	0	Independent Direct Mode.
1	0	1	0	0	0	Switched Mode
0	0	0	1	0	0	Framer 1->2 and Framer 3->4 Pass-through Mode
0	0	0	1	0	1	Framer 2->1 and Framer 3->4 Pass-through Mode
0	0	0	1	1	0	Framer 1->2 and Framer 4->3 Pass-through Mode
0	0	0	1	1	1	Framer 2->1 and Framer 4->3 Pass-through Mode
1	1	1	1	0	0	IMA/UNI mode

The SEL_SYNC and LTMODE signals on ports PC(8) and PC(9) are used to select the signal applied to the QuadFALC SYNC input. All selections are possible in each mode.

Table 1-35. Reference Frequency Selection

SEL_SYNC PC(8)	LTMODE PC(9)	QuadFALC SYNC input	Description
0	0	TxREF	SYNC input is tied to the PMC P3 8 KHz UTOPIA reference clock.
1	0	FSOH	SYNC input is tied to the PMC P4 8 KHz network reference clock.
X	1	VDD	The SYNC input is tied to VDD.

Multiplex Direct Mode

According to Table 1-34, PC(0) = 1, PD(6) = 0, PD(12:15) = 1100.

In multiplex direct mode, the QuadFALC system interface is in multiplex mode and the four framers have the same rhythm. The first QuadFALC TDM bus is tied directly to the CPM TDM bus TDMa1. The TDM bus clock and the frame synchronization signal are provided by the QuadFALC. In NT mode, the QuadFALC can synchronize on an external network reference clock provided on connector P4 or P3. **In this mode, the MT90220/1 DSTO_0-3 pins must be tristated.**

Figure 1-8, Figure 1-9, and Figure 1-10 show the specific implementation of this mode. Grey lines indicate unused connections.

Table 1-36. TDM and Synchronization Signals in Multiplex Direct Mode

Output	Input(s)	Description
RDO_1 (QuadFALC)	TDMa1_RX	8 Mb/s received data from the four E1/T1 lines. The QuadFALC system interface is in multiplex mode.
TDMa1_TX (CPM)	XDI_1	8 Mb/s transmit data to the four E1/T1 lines. The QuadFALC system interface is in multiplex mode.
FSC (QuadFALC)	RPA1, XPA1, TDMa1_L1RSYNC RxREF	8 kHz synchronization pulse generated by one of the four DCO-Rs used for the TDM frame synchronization clock. RPA1 input is configured as $\overline{\text{SYPR}}$ and used for the Receive Frame Synchronous Pulse (CMR2.IRSP=0). XPA1 is configured as $\overline{\text{SYPX}}$ and used for the Transmit Frame Synchronous Pulse (CMR2.IXSP=0).
RCLK1 (QuadFALC)	SCLKR1 SCLKX1 TDMa1_L1RCLK	8.192 MHz dejittered clock generated by one of the four DCO-R circuits, output on RCLK1 (PC5.CRP=1) and used for the TDM bus clock. SCLKR1 input is used for the Receive System Clock (CMR2.IRSC=0). SCLKX1 input is used for the Transmit System Clock (CMR2.IXSC=0) and provides the transmit rhythm to the DCO-X circuits. In LT mode, the DCO-R synchronizes on one of the four recovered line clocks. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
FSOH (P4) or TxREF (P3) or GND	SYNC	SYNC input is tied either to the PMC P4 8 kHz network reference clock or to the PMC P3 8 kHz UTOPIA reference clock. The SYNC input signal is selected by SEL_SYNC (PC(8)) and LTMODE (PC(9)) signals. When no signal is provided, SYNC is tied to GND.

NOTE

TDMc1, TDMa2 and TDMc2 signals are not used and must be tristated.

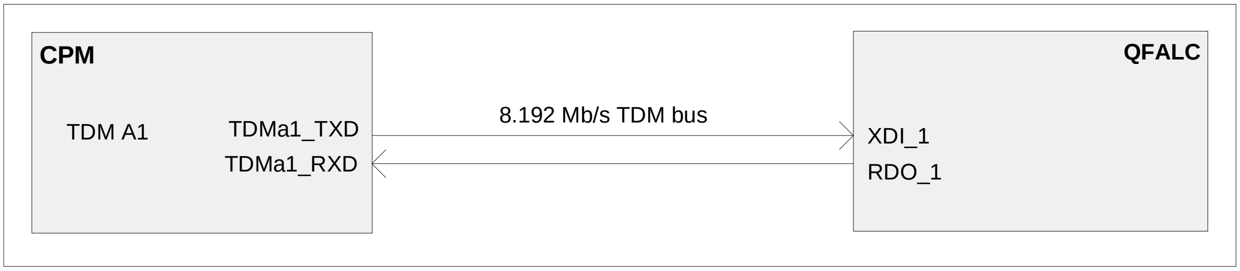


Figure 1-8. TDM Bus in Multiplex Direct Mode

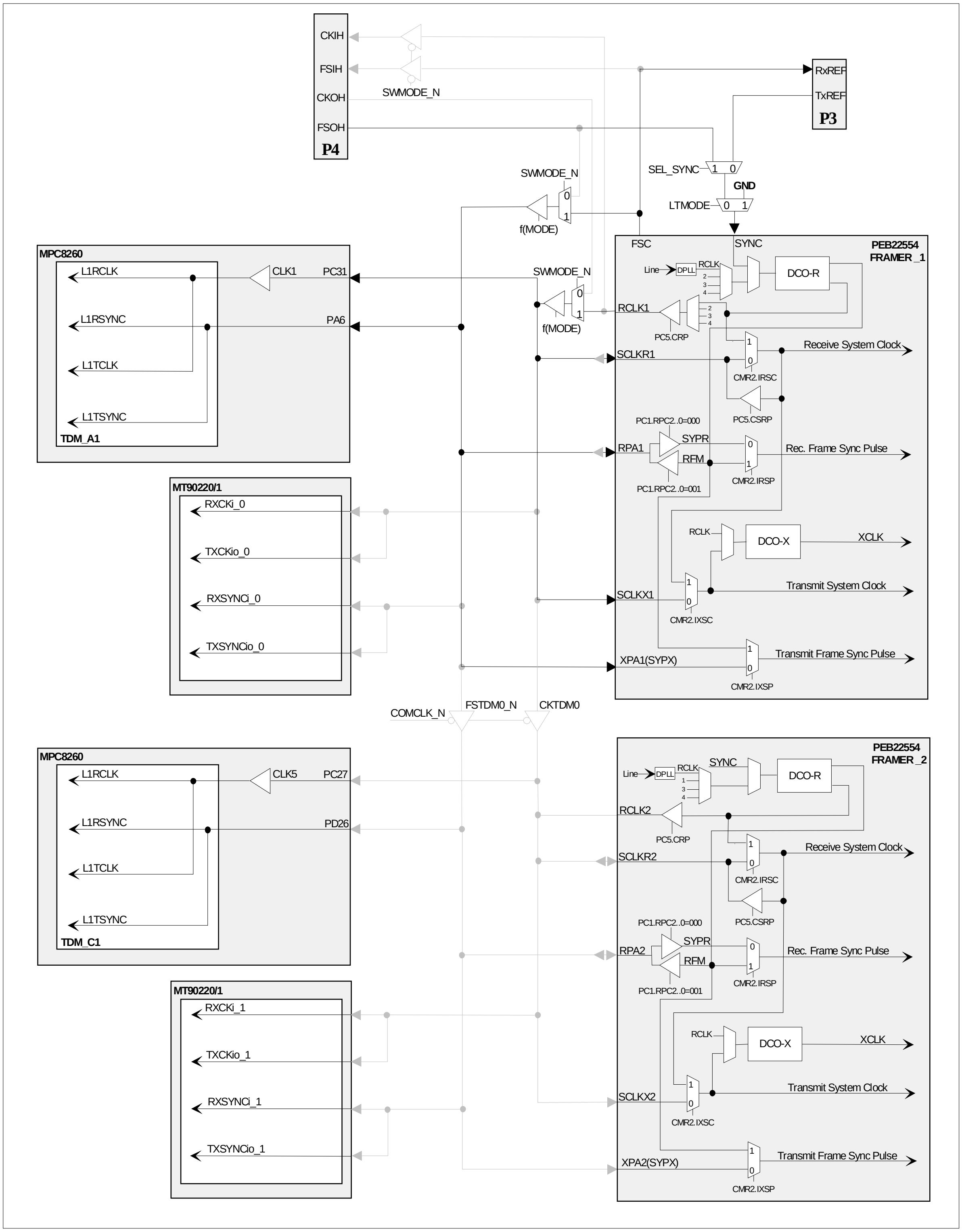


Figure 1-9. Clocks in Multiplex Direct Mode (Framer 1 and 2)

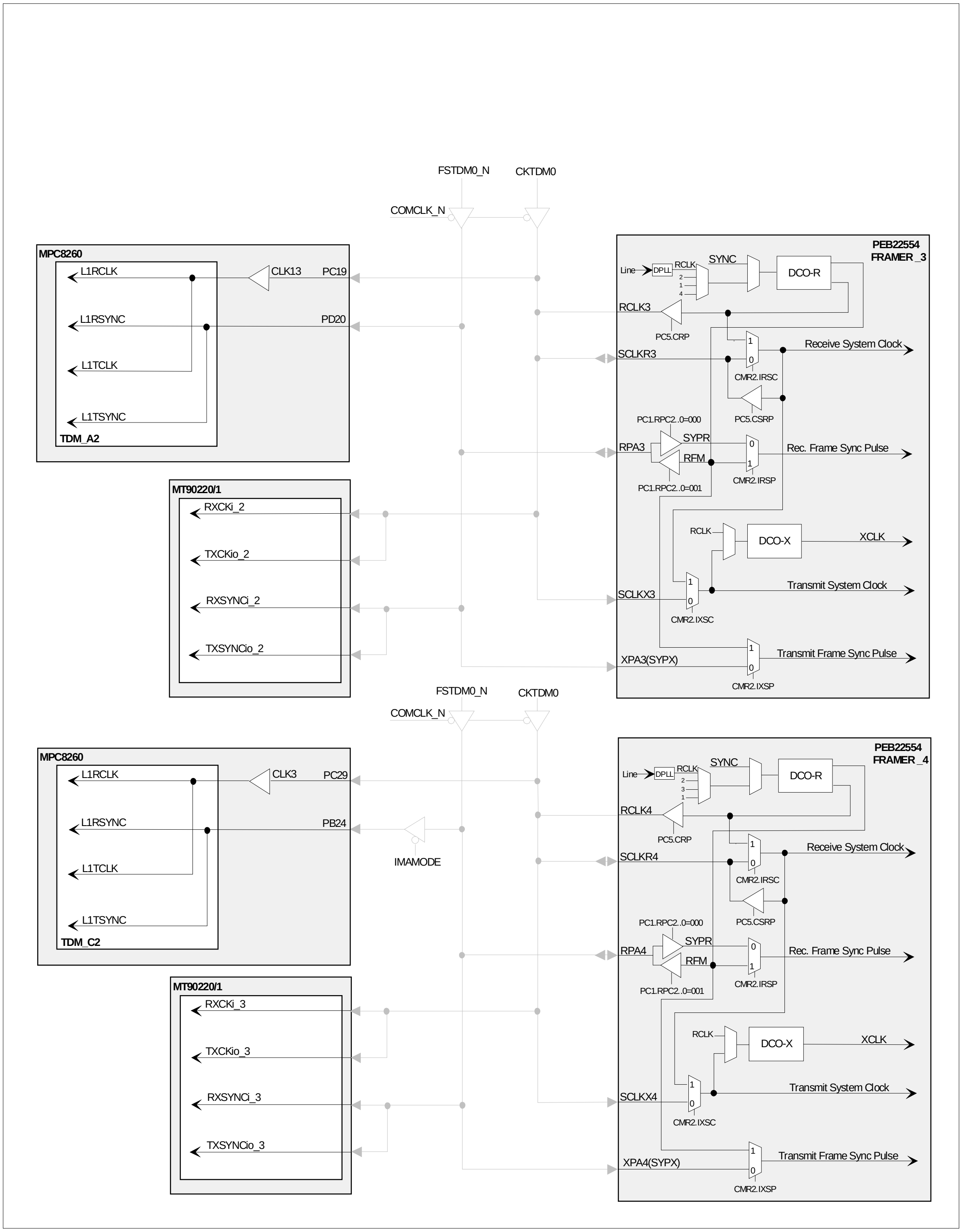


Figure 1-10. Clocks in Multiplex Direct Mode (Framer 3 and 4)

Independent Direct Mode

According to Table 1-34, PC(0) = 1, PD(6) = 0, PD(12:15) = 1110.

In independent direct mode, each framer can have its own rhythm. Each QuadFALC TDM bus is tied directly to a CPM TDM bus and has its own clock and frame synchronization signal provided by the QuadFALC. In NT mode, each framer can synchronize on an external network reference clock provided on connector P4 or P3. **In this mode, the MT90220/1 DSTO_0-3 pins must be tristated.**

Figure 1-11, Figure 1-12, and Figure 1-13 show the specific implementation of this mode. Grey lines indicate unused connections.

Table 1-37. TDM and Synchronization Signals in Independent Direct Mode

Output	Input(s)	Description
RDO_1 (QuadFALC)	TDMa1_RX	2 Mb/s received data from the E1/T1 line.
TDMa1_TX (CPM)	XDI_1	2Mb/s transmit data for the E1/T1 line.
RPA1 (QuadFALC)	TDMa1_L1RSYNC	8 kHz frame synchronization pulse generated by the DCO-R and output on RPA1 configured as RFM. The Receive Frame Synchronous Pulse and the Transmit Frame Synchronous Pulse are internally generated, (CMR2.IRSP=1, CMR2.IXSP=1).
SCLKR1 (QuadFALC)	TDMa1_L1RCLK	2.048 MHz dejittered Receive System Clock (CMR2.IRSC=1) generated by the DCO-R circuit, output on SCLKR1 (PC5.CSRP=1) and used for the TDM bus clock. The Transmit System Clock is sourced by the internal Receive System Clock (CMR2.IXSC=1) and provides the transmit rhythm to the DCO-X circuit. In LT mode, the input of the DCO-R is the recovered line clock. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
RDO_2 (QuadFALC)	TDMc1_RX	2 Mb/s received data from the E1/T1 line.
TDMc1_TX (CPM)	XDI_2	2 Mb/s transmit data for the E1/T1 line.
RPA2 (QuadFALC)	TDMc1_L1RSYNC	8 kHz frame synchronization pulse generated by the DCO-R and output on RPA2 configured as RFM. The Receive Frame Synchronous Pulse and the Transmit Frame Synchronous Pulse are internally generated, (CMR2.IRSP=1, CMR2.IXSP=1).

Table 1-37. TDM and Synchronization Signals in Independent Direct Mode

Output	Input(s)	Description
SCLKR2 (QuadFALC)	TDMc1_L1RCLK	2.048 MHz dejittered Receive System Clock (CMR2.IRSC=1) generated by the DCO-R circuit, output on SCLKR2 (PC5.CSRP=1) and used for the TDM bus clock. The Transmit System Clock is sourced by the internal Receive System Clock (CMR2.IXSC=1) and provides the transmit rhythm to the DCO-X circuit. In LT mode, the input of the DCO-R is the recovered line clock. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
RDO_3 (QuadFALC)	TDMa2_RX	2 Mb/s received data from the E1/T1 line.
TDMa2_TX (CPM)	XDI_3	2 Mb/s transmit data for the E1/T1 line.
RPA3 (QuadFALC)	TDMa2_L1RSYNC	8 kHz frame synchronization pulse generated by the DCO-R and output on RPA3 configured as RFM. The Receive Frame Synchronous Pulse and the Transmit Frame Synchronous Pulse are internally generated (CMR2.IRSP=1, CMR2.IXSP=1).
SCLKR3 (QuadFALC)	TDMa2_L1RCLK	2.048 MHz dejittered Receive System Clock (CMR2.IRSC=1) generated by the DCO-R circuit, output on SCLKR3 (PC5.CSRP=1) and used for the TDM bus clock. The Transmit System Clock is sourced by the internal Receive System Clock (CMR2.IXSC=1) and provides the transmit rhythm to the DCO-X circuit. In LT mode, the input of the DCO-R is the recovered line clock. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
RDO_4 (QuadFALC)	TDMc2_RX	2 Mb/s received data from the E1/T1 line.
TDMc2_TX (CPM)	XDI_4	2 Mb/s transmit data for the E1/T1 line.
RPA4 (QuadFALC)	TDMc2_L1RSYNC	8 kHz frame synchronization pulse generated by the DCO-R and output on RPA4 configured as RFM. The Receive Frame Synchronous Pulse and the Transmit Frame Synchronous Pulse are internally generated (CMR2.IRSP=1, CMR2.IXSP=1).

Table 1-37. TDM and Synchronization Signals in Independent Direct Mode

Output	Input(s)	Description
SCLKR4 (QuadFALC)	TDMc2_L1RCLK	2.048 MHz dejittered Receive System Clock (CMR2.IRSC=1) generated by the DCO-R circuit, output on SCLKR4 (PC5.CSRP=1) and used for the TDM bus clock. The Transmit System Clock is sourced by the internal Receive System Clock (CMR2.IXSC=1) and provides the transmit rhythm to the DCO-X circuit. In LT mode, the input of the DCO-R is the recovered line clock. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
FSOH (P4) or TxREF (P3) or GND	SYNC	SYNC input is tied either to the PMC P4 8 kHz network reference clock or to the PMC P3 8 kHz UTOPIA reference clock. The SYNC input signal is selected by SEL_SYNC (PC(8)) and LTMODE (PC(9)) signals. When no signal is provided, SYNC is tied to GND.

NOTE

RCLK2, RCLK3 and RCLK4 must be configured as inputs (PC5.CRP=0). XPA2, XPA3 and XPA4 shall be configured as SYPX (they must not be configured as outputs).

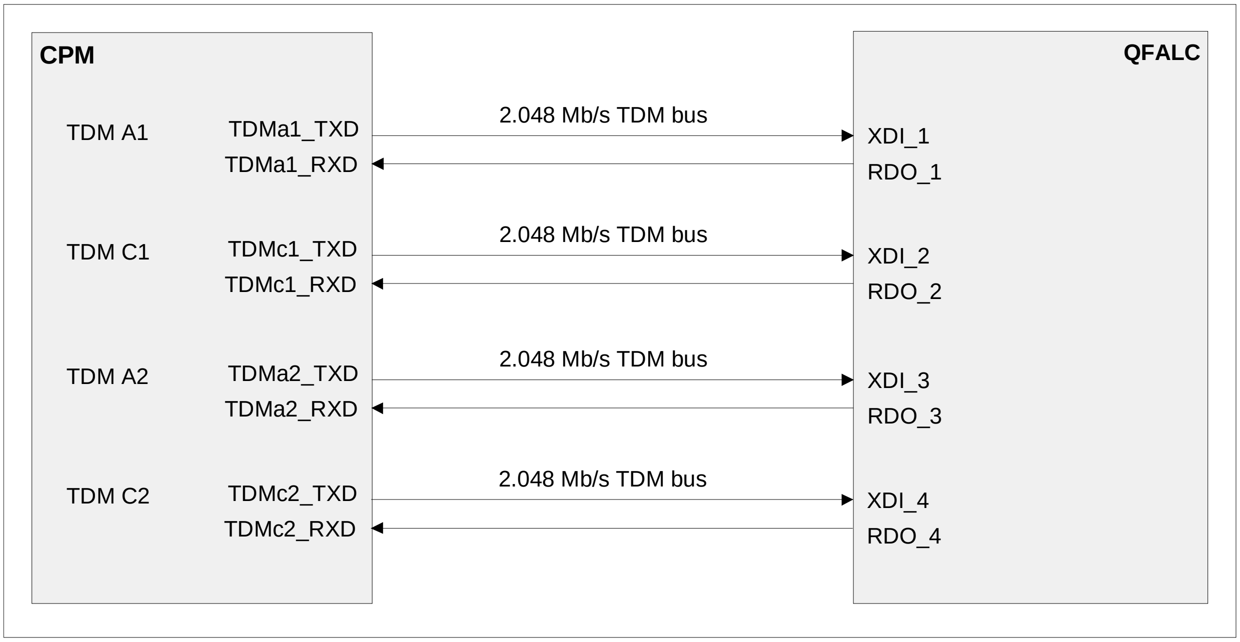


Figure 1-11. TDM Busses in Independent Direct Mode

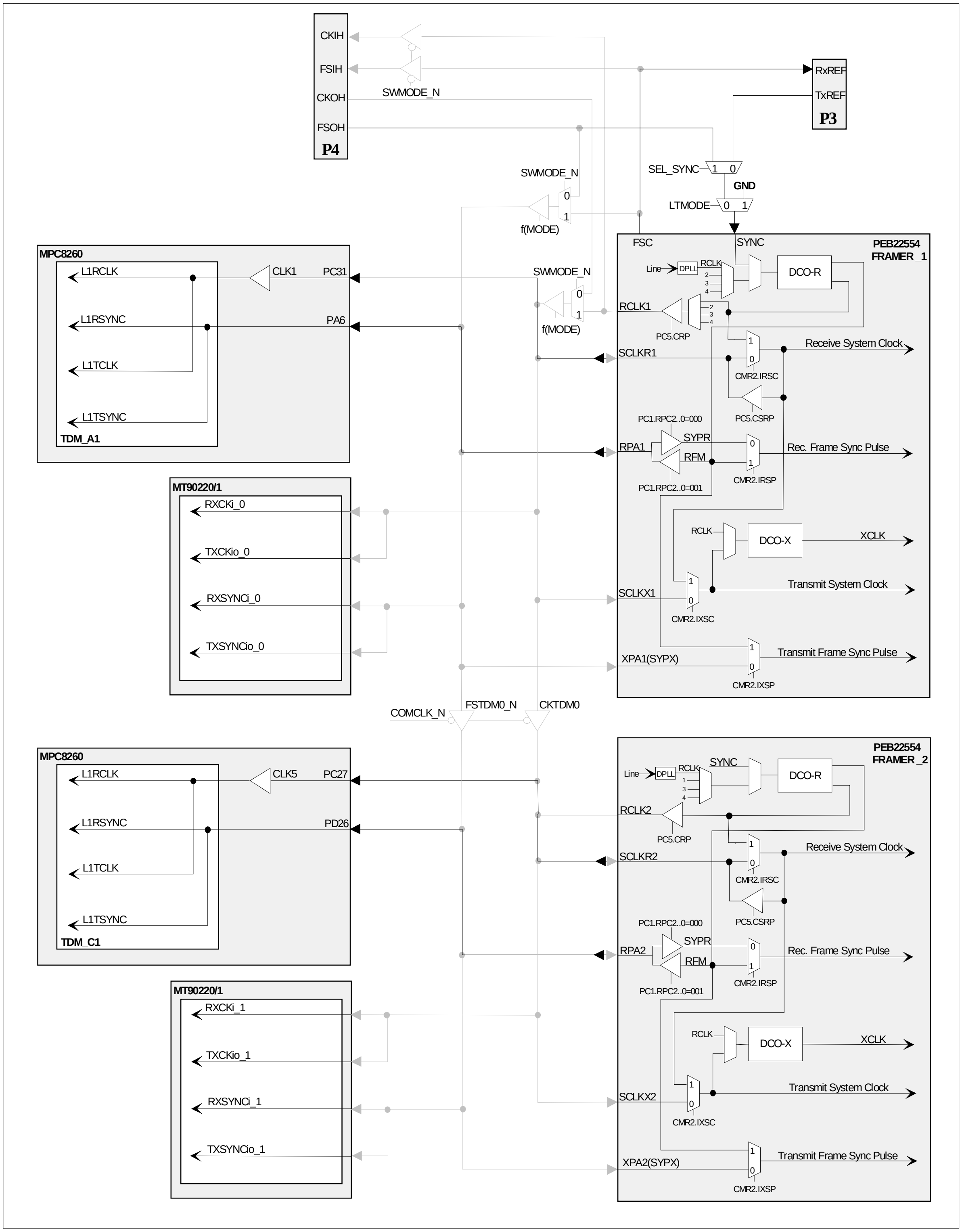


Figure 1-12. Clocks in Independent Direct Mode (Framer 1 and 2)

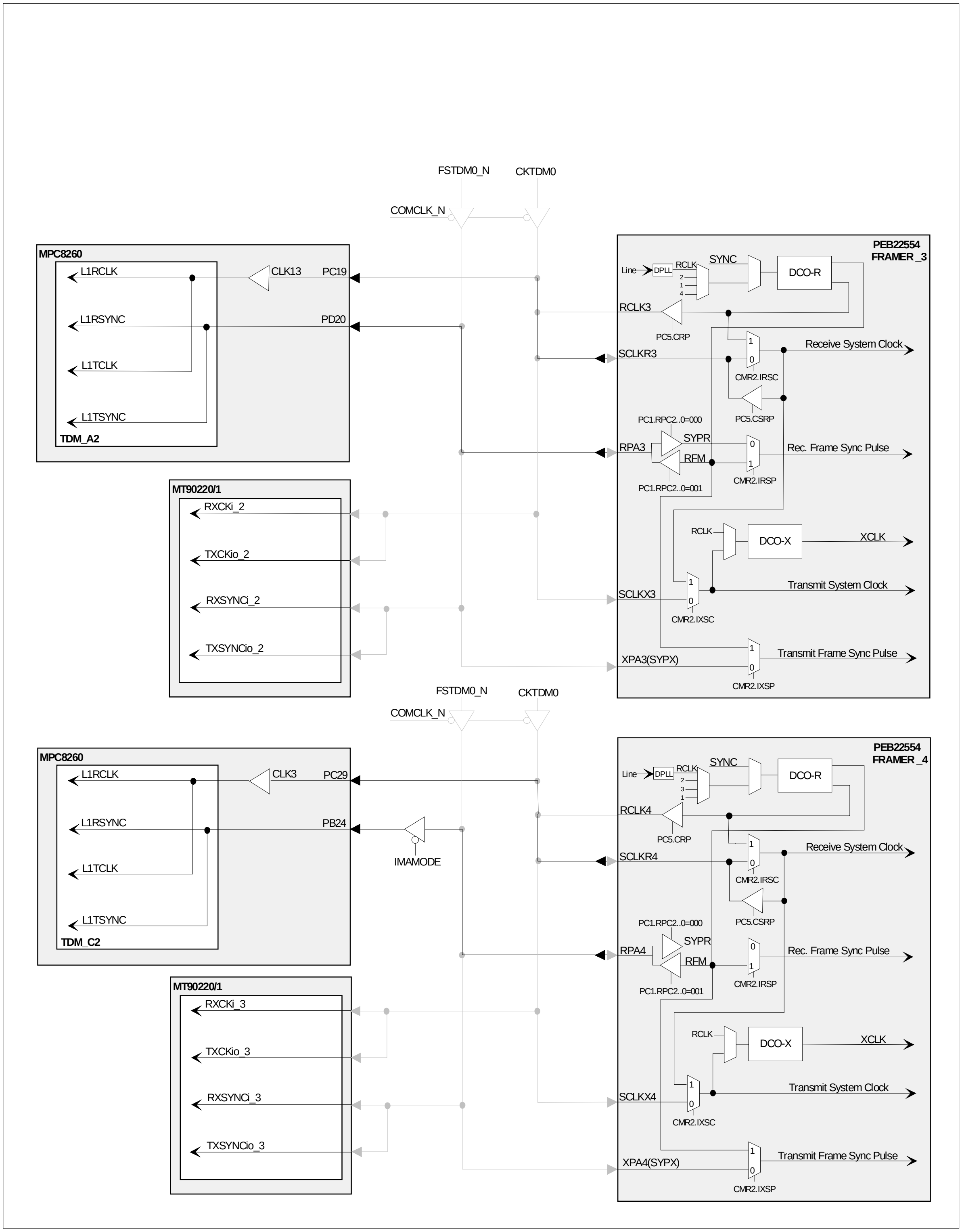


Figure 1-13. Clocks in Independent Direct Mode (Framer 3 and 4)

Switched Mode

According to Table 1-34, PC(0) = 1, PD(6) = 0, PD(12:15) = 1000.

In switched mode, the QuadFALC multiplexed TDM bus is tied to the first TDM bus on P4. The second TDM bus on P4 is tied to the MPC8260. The TDM busses clock and frame synchronization signals are provided by connector P4. In NT mode, the QuadFALC can synchronize on an external network reference clock provided on P4. **In this mode, the MT90220/1 DSTO_0-3 pins must be tristated.**

Figure 1-14, Figure 1-15, and Figure 1-16 show the specific implementation of this mode. Grey lines indicate unused connections.

Table 1-38. TDM and Synchronization Signals in Switched Mode

Output	Input(s)	Description
RDO_1 (QuadFALC)	DIH0	8 Mb/s received data from the four E1/T1 lines and sent to the first P4 TDM bus. The QuadFALC system interface is in multiplex mode.
DOH0 (P4)	XDI_1	8 Mb/s transmit data for the four E1/T1 lines from the first P4 TDM bus. QuadFALC system interface is in multiplex mode.
TDMa1_TX (CPM)	DIH1	8 Mb/s transmit data from the CPM TDMa1 bus to the second P4 TDM bus.
DOH1 (P4)	TDMa1_RX	8 Mb/s received data from the second P4 TDM bus to the CPM TDMa1 bus.
FSOH (P4)	TDMa1_L1RSYNC, RPA1, XPA1	8 kHz frame synchronization pulse provided by P4 for both TDM busses. RPA1 input is configured as $\overline{\text{SYPR}}$ and used for the Receive Frame Synchronous Pulse (CMR2.IRSP=0). XPA1 is configured as $\overline{\text{SYPX}}$ and used for the Transmit Frame Synchronous Pulse (CMR2.IXSP=0). In LT mode, FSOH should be externally synchronized to the lines rhythm.
CKOH (P4)	TDMa1_L1RCLK, SCLKR1, SCLKX1	8.192 MHz clock provided by P4 for both TDM busses. SCLKR1 input is used for the Receive System Clock (CMR2.IRSC=0). SCLKX1 input is used for the Transmit System Clock (CMR2.IXSC=0) and provides the transmit rhythm to the DCO-X circuits. In LT mode, CKOH should be externally synchronized to the lines rhythm. In NT mode, CKOH provides the line rhythm to the DCO-X circuit via SCLKX1.
FSC (QuadFALC)	FSIH, RxREF	8 kHz synchronization pulse generated by the internal DCO1-R, synchronized to the lines and provided to P4 and P3.
RCLK1 (QuadFALC)	CKIH	Dejittered clock generated by the internal DCO-R circuit, synchronized to the lines and provided to P4.

Table 1-38. TDM and Synchronization Signals in Switched Mode (cont)

Output	Input(s)	Description
GND	SYNC	SYNC input tied to GND. The SYNC input signal is selected by SEL_SYNC (PC(8)) and LTMODE (PC(9)) signals.

NOTE

TDMc1, TDMa2 and TDMc2 signals are not used and must be tristated.

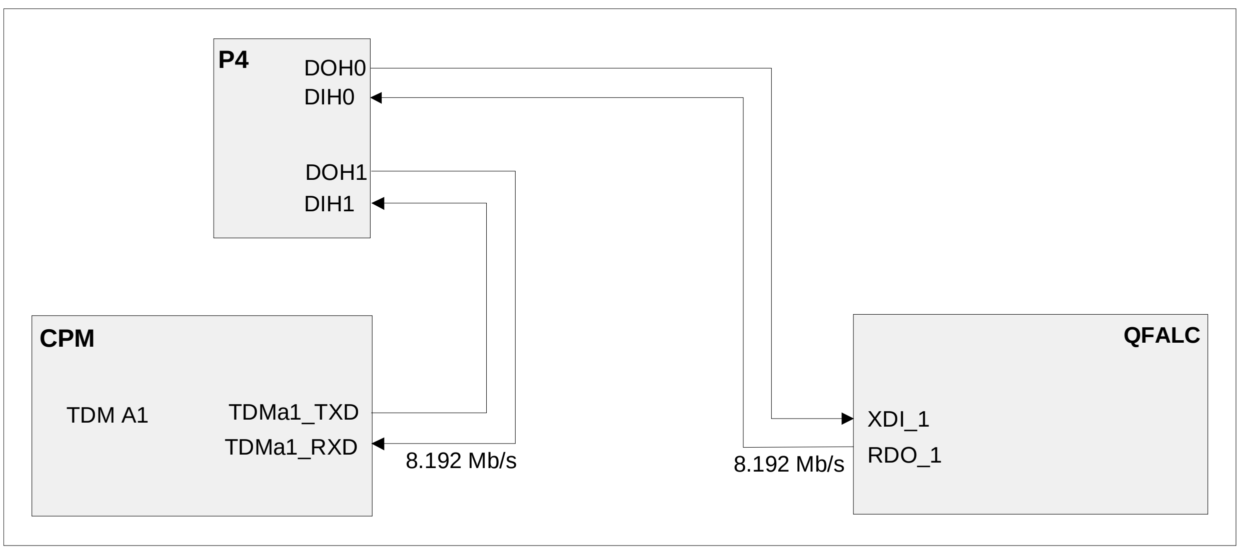


Figure 1-14. TDM Busses in Switched Mode

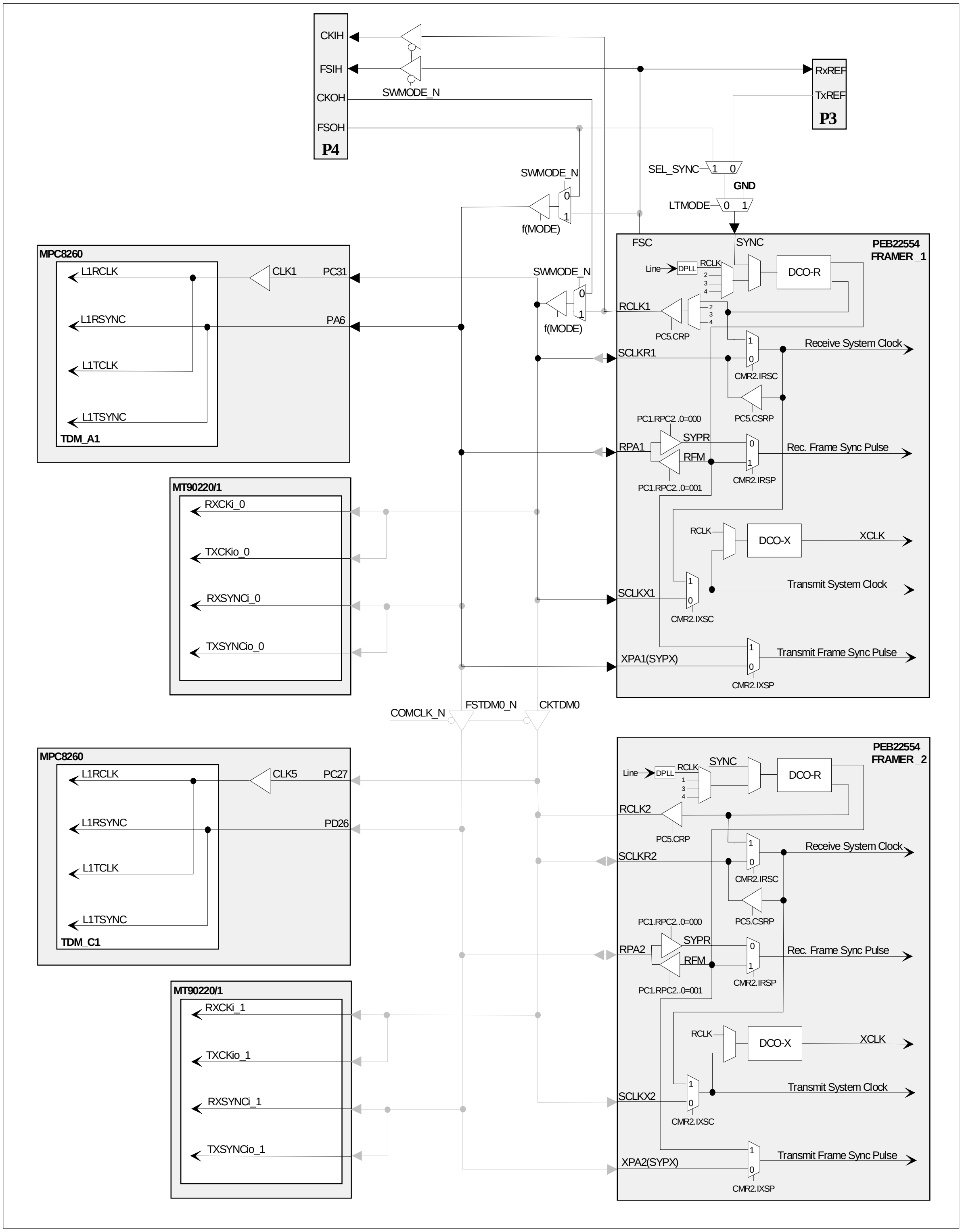


Figure 1-15. Clocks in Switched Mode (Framer 1 and 2)

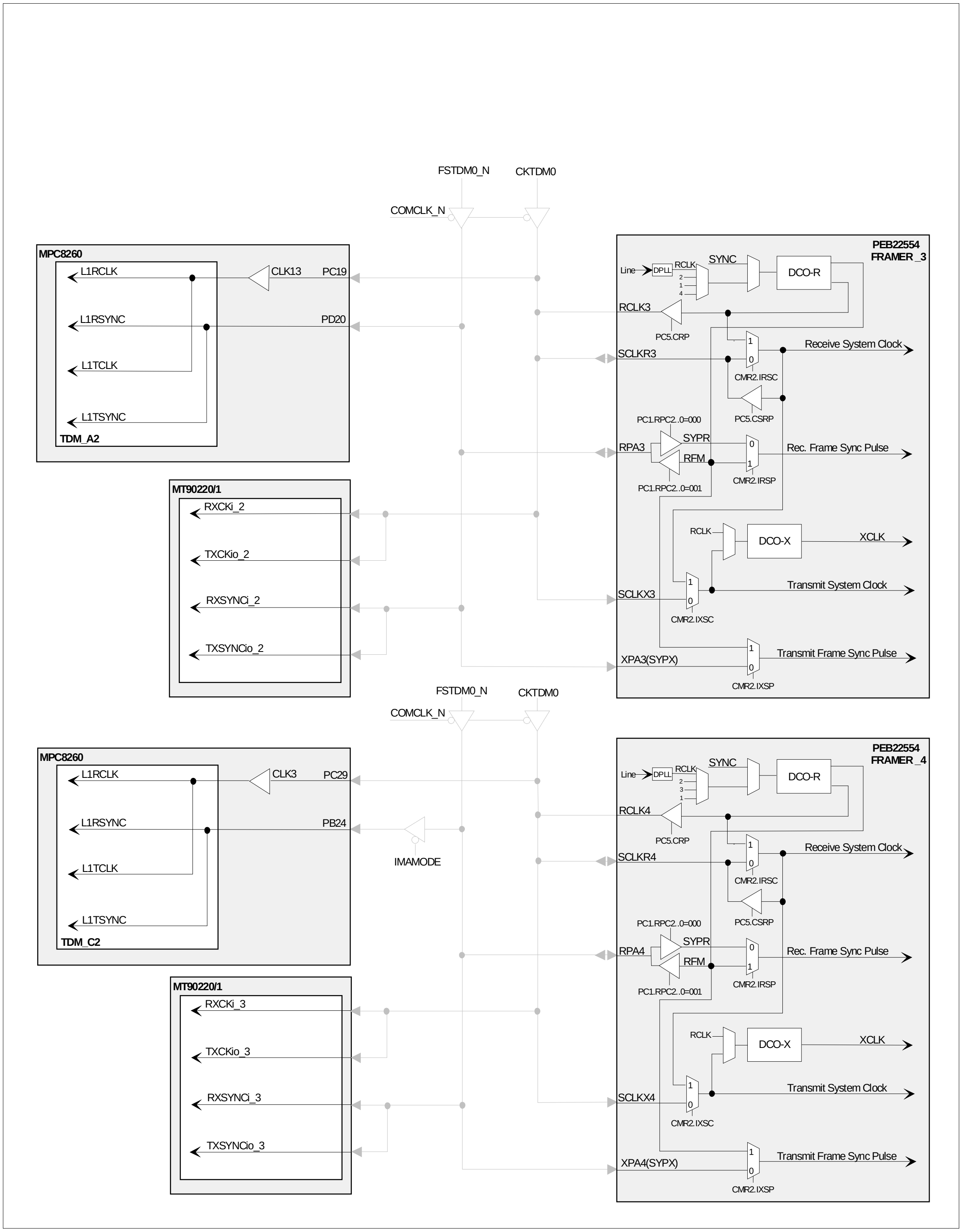


Figure 1-16. Clocks in Switched Mode (Framer 3 and 4)

Pass-Through Mode

According to Table 1-34, PC(0) = 0, PD(6) = 0, PD(12:13) = 01. PD(14) and PD(15) determine the pass-through direction.

Pass through is possible from framer 1 to framer 2 and vice versa and from framer 3 to framer 4 and vice versa. The four framers have the same clock rhythm (COMCLK_N = 0**)**.

In framer 1 to framer 2 pass-through mode, the first framer is tied to the network in LT mode. Data received from this framer goes to TDMa1 and to the second framer, which is in NT mode. Thus it can be connected to another adapter configured as a line Termination (LT) circuit. Data received from framer 2 is "anded" with Tx data from TDMa1 and is transmitted by framer 1. Framer 2 to framer 1 pass-through description is symmetrical. The same description applies to framer 3 and framer 4.

Figure 1-17, Figure 1-18, Figure 1-19 and Figure 1-20 show the specific implementation of this mode. Grey lines indicate unused connections. In Table 1-39, only framer 1 to framer 2 and framer 3 to framer 4 pass through is described.

In this mode, the MT90220/1 DSTO_0-3 pins must be tristated.

Table 1-39. TDM and Synchronization Signals in Pass Through Mode

Output	Input(s)	Description
RDO_1 (QuadFALC)	TDMa1_RX, XDI_2	2 Mb/s received data from the first E1/T1 line and sent to the second line and to the CPM.
RDO_3 (QuadFALC)	TDMa2_RX, XDI_4	2 Mb/s received data from the third E1/T1 line and sent to the fourth line and to the CPM.
TDMa1_TX (CPM) AND RDO_2 (QuadFALC)	XDI_1	2 Mb/s transmit data for the first E1/T1 line. Data from TDMa1_TX and RDO_2 are anded.
TDMa2_TX (CPM) AND RDO_4 (QuadFALC)	XDI_3	2 Mb/s transmit data for the third E1/T1 line. Data from TDMa2_TX and RDO_4 are anded.
FSC (QuadFALC)	RPA1, XPA1, RPA2, XPA2, RPA3, XPA3, RPA4, XPA4, TDMa1_L1RSYNC TDMc1_L1RSYNC TDMa2_L1RSYNC TDMc2_L1RSYNC, RxREF	8 kHz synchronization pulse generated by the DCO1-R used for the TDM frame synchronization clocks. RPA1,2,3,4 inputs are configured as $\overline{\text{SYPR}}$ and used for the Receive Frame Synchronous Pulse (CMR2.IRSP=0). XPA1,2,3,4 are configured as $\overline{\text{SYPX}}$ and used for the Transmit Frame Synchronous Pulse (CMR2.IXSP=0).

Table 1-39. TDM and Synchronization Signals in Pass Through Mode (cont)

Output	Input(s)	Description
RCLK1 (QuadFALC)	SCLKR1, SCLKX1, SCLKR2, SCLKX2, SCLKR3, SCLKX3, SCLKR4, SCLKX4, TDMa1_L1RCLK, TDMc1_L1RCLK, TDMa2_L1RCLK, TDMc2_L1RCLK	2.048 MHz dejittered receive clock generated by one of the four DCO-R circuits, output on RCLK1 (PC5.CRP=1) and used for the TDM bus clocks. SCLKR1,2,3,4 inputs are used for the Receive System Clock **(CMR2.IRSC=0)**. SCLKX1,2,3,4 inputs are used for the Transmit System Clock (CMR2.IXSC=0) and provides the transmit rhythm to the DCO-X circuits. The input of the DCO1-R is one of the four recovered line clocks.
GND	SYNC	SYNC input tied to GND. The SYNC input signal is selected by SEL_SYNC (PC(8)) and LTMODE (PC(9)) signals.

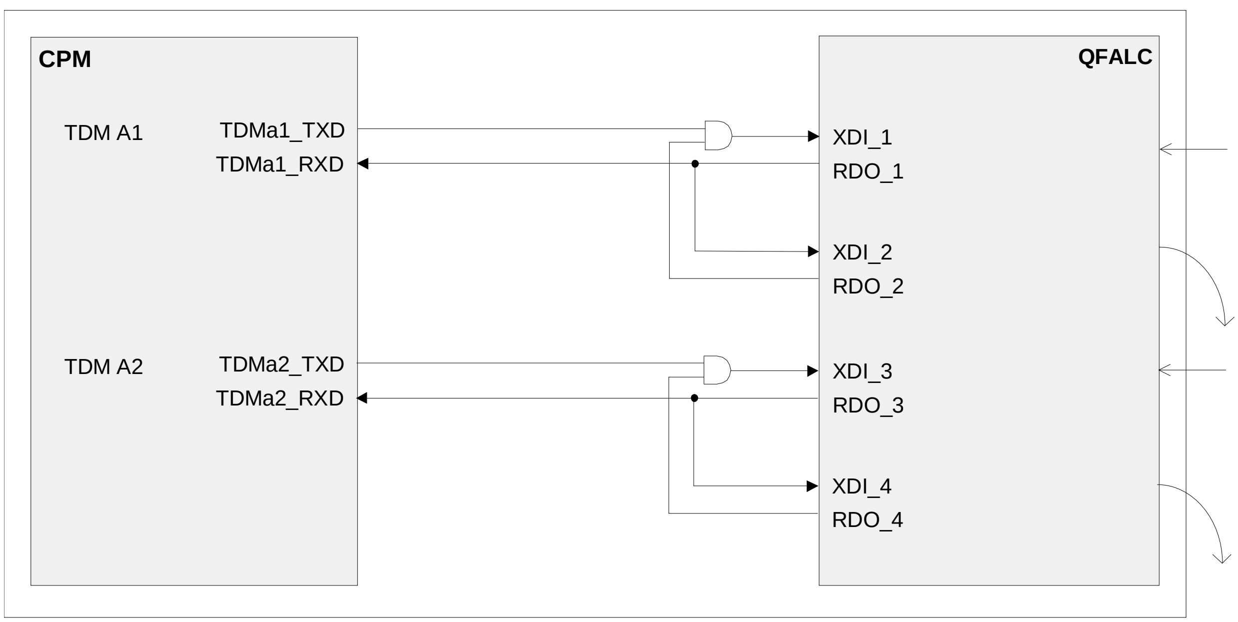


Figure 1-17. TDM Busses in Pass-Through Mode (1->2 and 3->4)

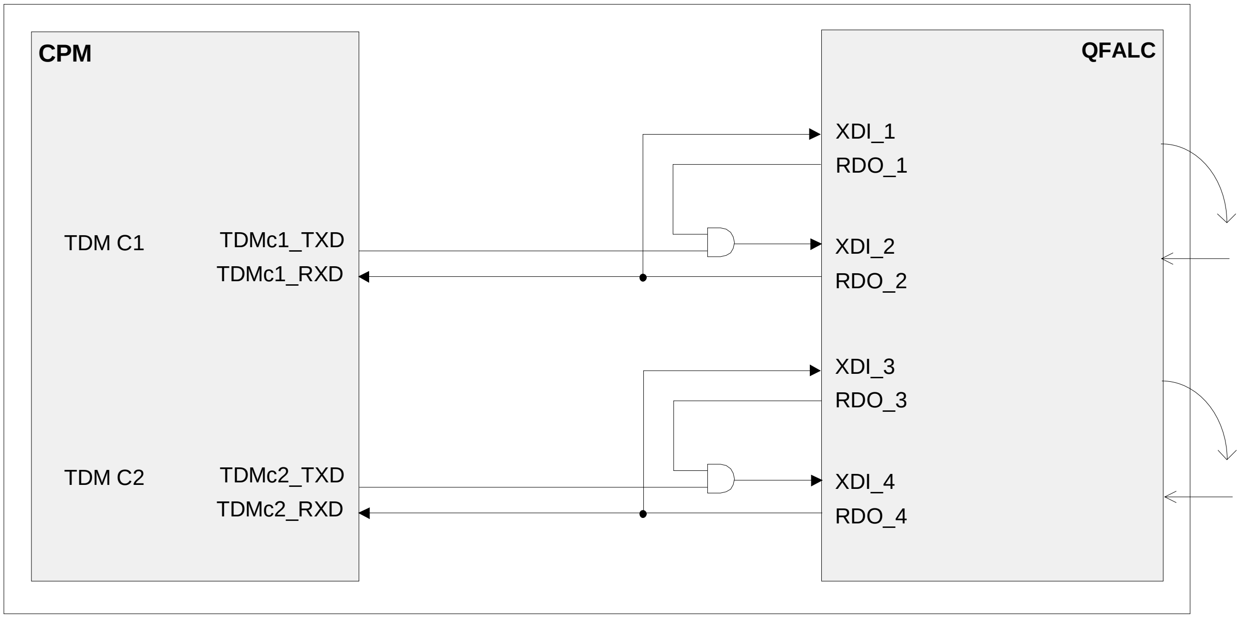


Figure 1-18. TDM busses in Pass-Through Mode (2->1 and 4->3)

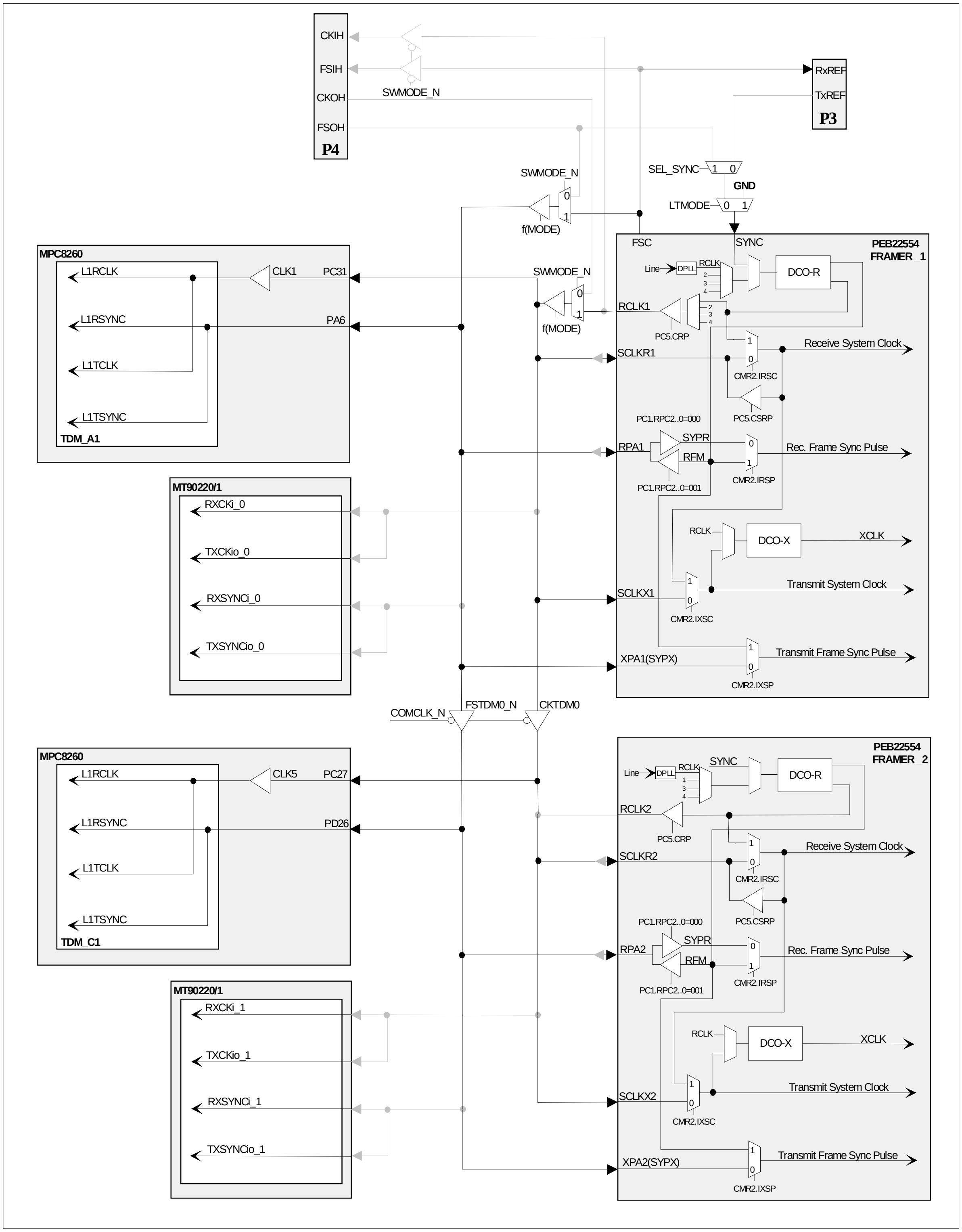


Figure 1-19. Clocks in Pass-Through Mode (Framer 1 and 2)

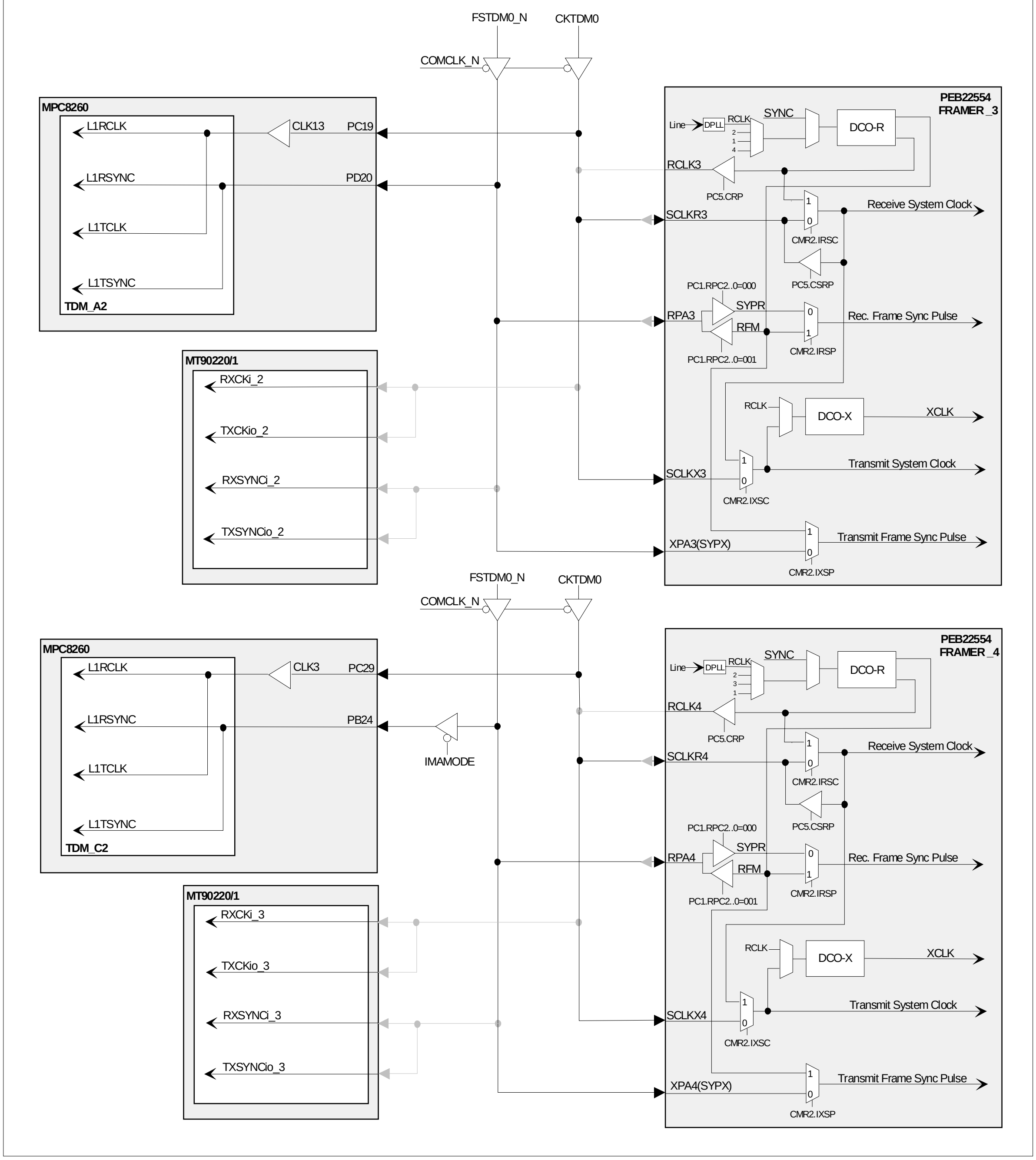


Figure 1-20. Clocks in Pass-Through Mode (Framer 3 and 4)

IMA/UNI Mode

According to Table 1-34, PC(0) = 1, PD(6) = 1, PD(12:15) = 1100.

In IMA/UNI mode, each QuadFALC TDM bus is directly tied to a MT90220/1 TDM bus and has its own independent clock and frame synchronization signal provided by the QuadFALC itself. In NT mode, each framer can synchronize on an external network reference clock provided on connector P4 or P3.

Figure 1-21, Figure 1-22 and Figure 1-23 show the specific implementation of this mode. Grey lines indicate unused connections.

Table 1-40. TDM and Synchronization Signals in IMA/UNI Mode

Output	Input(s)	Description
RDO_1 (QuadFALC)	DSTI_0 (MT90220/1)	2 Mb/s received data from the E1/T1 line.
DSTO_0 (MT90220/1)	XDI_1 (QuadFALC)	2Mb/s transmit data for the E1/T1 line.
RPA1 (QuadFALC)	RXSYNCi_0 TXSYNCio_0	8 KHz frame synchronization pulse generated by the DCO-R and output on RPA1 configured as RFM. The Receive Frame Synchronous Pulse and the Transmit Frame Synchronous Pulse are internally generated, (CMR2.IRSP=1, CMR2.IXSP=1).
SCLKR1 (QuadFALC)	RXCKi_0 TXCKio_0	2.048 MHz dejittered Receive System Clock (CMR2.IRSC=1) generated by the DCO-R circuit, output on SCLKR1 (PC5.CSRP=1) and used for the TDM bus clock. The Transmit System Clock is sourced by the internal Receive System Clock (CMR2.IXSC=1) and provides the transmit rhythm to the DCO-X circuit. In LT mode, the input of the DCO-R is the recovered line clock. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
RDO_2 (QuadFALC)	DSTI_1	2 Mb/s received data from the E1/T1 line.
DSTO_1 (MT90220/1)	XDI_2	2 Mb/s transmit data for the E1/T1 line.
RPA2 (QuadFALC)	RXSYNCi_1 TXSYNCio_1	8 KHz frame synchronization pulse generated by the DCO-R and output on RPA2 configured as RFM. The Receive Frame Synchronous Pulse and the Transmit Frame Synchronous Pulse are internally generated, (CMR2.IRSP=1, CMR2.IXSP=1).

Table 1-40. TDM and Synchronization Signals in IMA/UNI Mode

Output	Input(s)	Description
SCLKR2 (QuadFALC)	RXCKi_1 TXCKio_1	Internal 2.048 MHz dejittered Receive System Clock (CMR2.IRSC=1) generated by the DCO-R circuit, output on SCLKR2 (PC5.CSRP=1) and used for the TDM bus clock. The Transmit System Clock is sourced by the internal Receive System Clock (CMR2.IXSC=1) and provides the transmit rhythm to the DCO-X circuit. In LT mode, the input of the DCO-R is the recovered line clock. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
RDO_3 (QuadFALC)	DSTI_2	2Mb/s received data from the E1/T1 line.
DSTO_2 (MT90220/1)	XDI_3	2Mb/s transmit data for the E1/T1 line.
RPA3 (QuadFALC)	RXSYNCi_2 TXSYNCio_2	8 KHz frame synchronization pulse generated by the DCO-R and output on RPA3 configured as RFM. The Receive Frame Synchronous Pulse and the Transmit Frame Synchronous Pulse are internally generated, (CMR2.IRSP=1, CMR2.IXSP=1).
SCLKR3 (QuadFALC)	RXCKi_2 TXCKio_2	Internal 2.048 MHz dejittered Receive System Clock (CMR2.IRSC=1) generated by the DCO-R circuit, output on SCLKR3 (PC5.CSRP=1) and used for the TDM bus clock. The Transmit System Clock is sourced by the internal Receive System Clock (CMR2.IXSC=1) and provides the transmit rhythm to the DCO-X circuit. In LT mode, the input of the DCO-R is the recovered line clock. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
RDO_4 (QuadFALC)	DSTI_3	2 Mb/s received data from the E1/T1 line.
DSTO_3 (MT90220/1)	XDI_4	2 Mb/s transmit data for the E1/T1 line.
RPA4 (QuadFALC)	RXSYNCi_3 TXSYNCio_3	8 KHz frame synchronization pulse generated by the DCO-R and output on RPA4 configured as RFM. The Receive Frame Synchronous Pulse and the Transmit Frame Synchronous Pulse are internally generated, (CMR2.IRSP=1, CMR2.IXSP=1).

Table 1-40. TDM and Synchronization Signals in IMA/UNI Mode

Output	Input(s)	Description
SCLKR4 (QuadFALC)	RXCKi_3 TXCKio_3	2.048 MHz dejittered Receive System Clock (CMR2.IRSC=1) generated by the DCO-R circuit, output on SCLKR4 (PC5.CSRP=1) and used for the TDM bus clock. The Transmit System Clock is sourced by the internal Receive System Clock (CMR2.IXSC=1) and provides the transmit rhythm to the DCO-X circuit. In LT mode, the input of the DCO-R is the recovered line clock. In NT mode, the DCO-R synchronizes on the external SYNC signal. When no reference clock is provided on SYNC, DCO-R is in free running mode.
FSOH (P4) or TxREF (P3) or GND	SYNC	SYNC input is tied either to the PMC P4 8 kHz network reference clock or to the PMC P3 8 kHz UTOPIA reference clock. The SYNC input signal is selected by SEL_SYNC (PC(8)) and LTMODE (PC(9)) signals. When no signal is provided, SYNC is tied to GND.

NOTE

TDMa1, TDMc1 and TDMa2 must be tristated. To avoid signal conflict, set the IMA/UNI mode, PD(6) = 1, before configuring the FCC2 UTOPIA bus on port B.

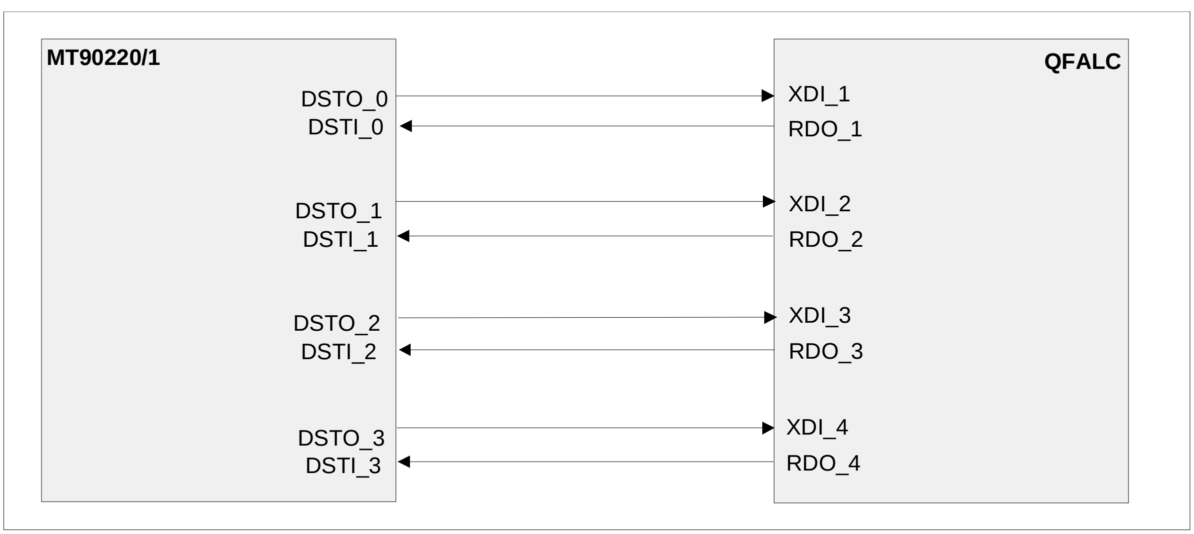


Figure 1-21. TDM Busses in IMA/UNI Mode

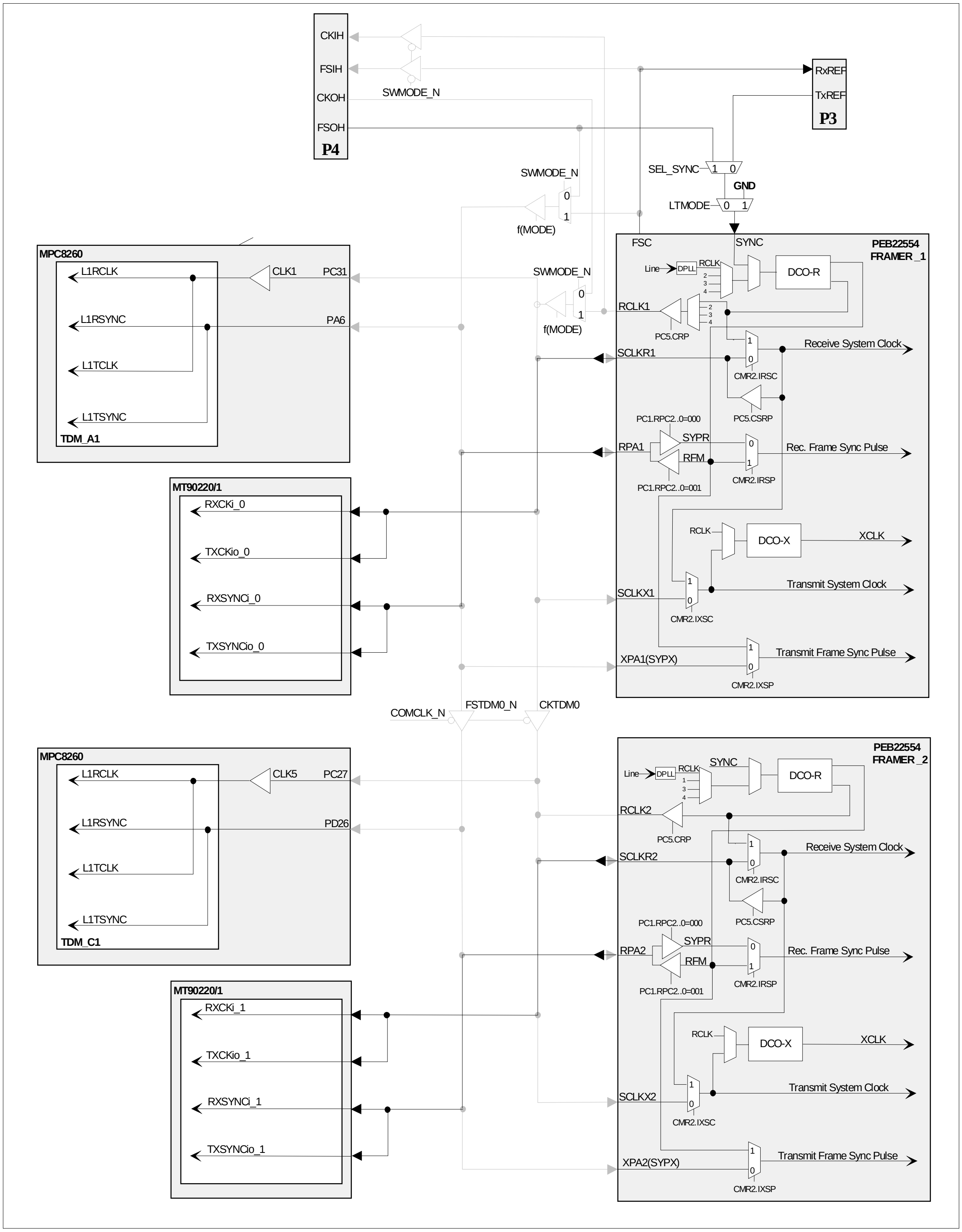


Figure 1-22. Clocks in IMA/UNI Mode

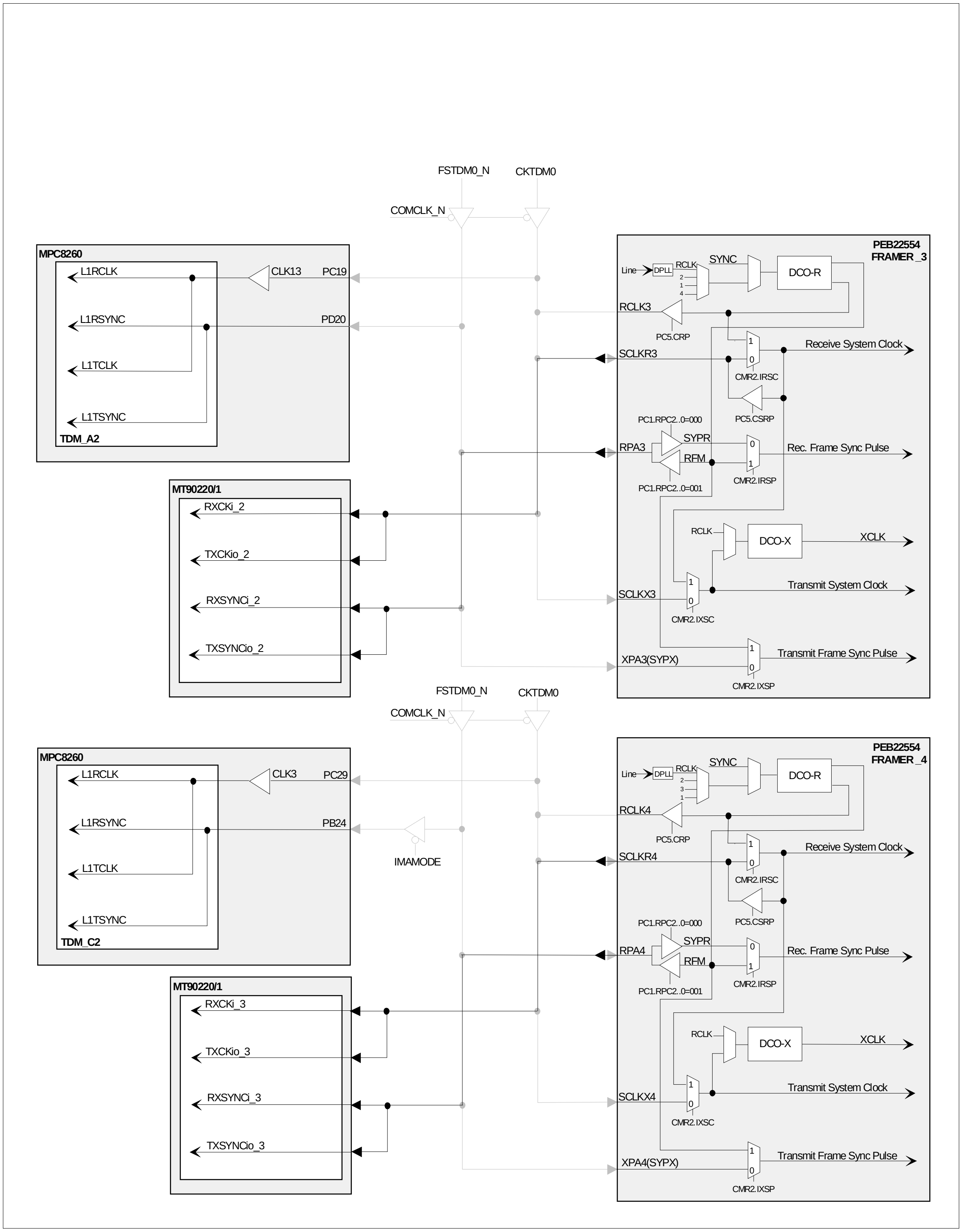


Figure 1-23. Clocks in IMA/UNI Mode

IMA/UNI Via Connector P4

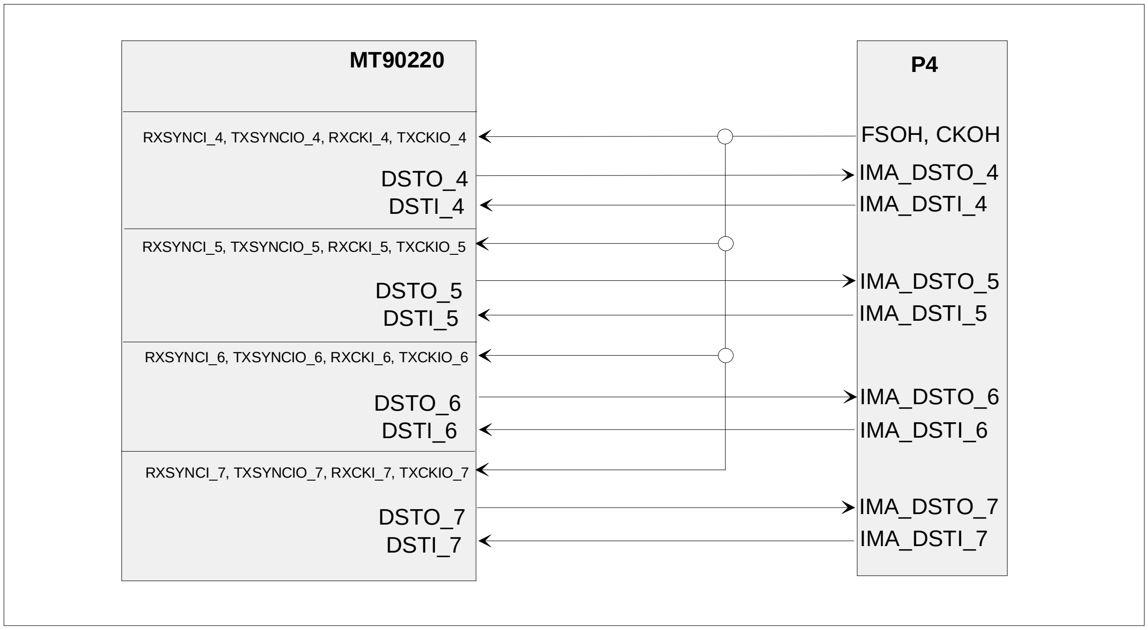


Figure 1-24. IMA/UNI Via Connector P4

When the MT90220 IMA chip is equipped, its last four TDM busses are connected to P4. The TDM bus clock and the frame synchronization signal are provided by P4.

Table 1-41. TDM and Synchronization Signals for IMA/UNI Via P4

Output	Input(s)	Description
FSOH (PMC P4)	TXSYNCIO_4-7, RXSYNCI_4-7	Synchronization pulse, used for TDM frame synchronization.
CKOH (PMC P4)	TXCKIO_4-7, RXCKI_4-7	TDM bus clocks
IMA_DSTI_4-7 (PMC P4)	DSTI_4-7	Received data from connector P4.
DSTO_4-7 (MT90220)	IMA_DSTO_4-7	Transmit data to connector P4.

Mixed Modes and Other Modes

Some modes can be mixed. For example, when the board is in IMA/UNI mode, the first three framers can be used in Independent mode.

4539 Power-Up Initialization

2

Overview

After power-up, the STARTUP code is executed. This code is written entirely in assembly language and is the entry point after a power-up or a reset exception. STARTUP configures the PowerQUICC II and several other critical hardware elements such as the SDRAM memories. Once STARTUP is executed, code written in a high-level language such as “C” is executed.

This chapter describes this STARTUP initialization.

NOTE

The STARTUP source code is provided in `APP/ASM/STARTUP.ASM` file.

PowerSpan Initialization

The PowerSpan is initialized by several different mechanisms.

During the power-up phase, the PowerSpan uses some of its I/O pins to determine its Hardware Configuration Word (also called Power-up options in PowerSpan documentation).

Then the PowerSpan initializes several of its internal registers by loading their values from the I^{2}C serial EEPROM. Among these registers, the PCI vendor and device identification, the size of the PCI-to-Local windows, and the size and position of the first Local to PCI window are initialized by the serial EEPROM.

The addresses in the PCI space of the PCI-to-Local windows are chosen by the PCI host during its boot and programmed in the PCI configuration registers.

All the other initializations must be done either by the PowerSpan during its boot or by the PCI host.

PowerSpan Hardware Configuration Word

On the board this configuration is defined as follows:

- PB_ARB_EN=0: Disable PowerSpan arbiter for 60x local bus
- P1_ARB_EN=0: Disable PowerSpan arbiter for PCI 1 bus
- P2_ARB_EN=0: Disable PowerSpan arbiter for PCI 2 bus (no PCI 2 bus)
- PWRUP_PRI_PCI=0: PCI 1 is primary PCI bus
- P1_R64_EN=0: Disable PCI 1 REQ64

- PWRUP_BOOT=0: The PowerQUICC II boots locally (not through PCI)
- PWRUP_DEBUG_EN=0: Disable debug mode
- PWRUP_BYPASS_EN=0: Disable PLL bypass

PowerSpan Register Initialization Through the I²C Serial EEPROM

Table 2-1 provides the PowerSpan Register initialization values stored in the Serial EEPROM. Refer to PowerSpan documentation, section *EEPROM Loading* for detailed mapping between EEPROM addresses and PowerSpan registers.

.

Table 2-1. PowerSpan Register Initialization Values in the Serial EEPROM

EEPROM Address	Initialization Value	Description
0	0x02	Long EEPROM load
5	0x0C	PCI Bus Master Enable and PCI memory Space Enable
6	0x00	PCI target image prefetch indicators (all 0)
7–8	0x90A0	PCI subsystem device ID
0x09–0x0A	0x107E	PCI subsystem vendor ID
0xB	0x02	Interrupt pin, INTA# is used on the PCI bus.
0xC	0x1C	PCI Base Address configuration registers enabled for the PowerSpan registers and target Image 0 and 1 (two PCI-to-local windows). Target images 1, 2, and 3 not enabled.
0xD	0x53	PCI target Image 0 = 2 MB, PCI target image 1 = 512 KB
0xE	0x00	PCI target images 2 and 3: size not defined (not enabled)
0xF	0x00	PCI Vital Product Data disabled
0x10	0x00	No PCI Lockout
0x11	0x5F	Interrupt pins direction: all outputs except –INT5.
0x12	0x00	PCI I²O target image disabled
0x13–0x1F	0x00	(Reserved)
0x20–0x21	0x90A0	PCI device ID
0x22–0x23	0x107E	PCI vendor ID
0x24	0x02	PCI Base Class Code: Network Controller
0x25	0x80	PCI Sub Class Code: Other
0x26	0x00	PCI programming Interface
0x27	0x01	Revision register
0x28–0x30	0x00	PB slave image 0 disabled (local to PCI window)
0x31–0x33	0xF00200	PB slave register image base address = 0xF0020000

Table 2-1. PowerSpan Register Initialization Values in the Serial EEPROM (cont)

EEPROM Address	Initialization Value	Description
0x34–0x3F	0x00	(Reserved)

Other PowerSpan Initializations

It is necessary to initialize the PowerSpan Interrupt Map registers in a specific way, in order to use the interrupt pins as specified for the 4539. This can be done by the local processor during its boot and/or by the PCI host.

The following C code is an example of interrupt pins initialization.

Example 2-1. PowerSpan Interrupt Map Registers Initialization Code

```
ConfigurePspanInterrupts( void)
{
   RegWrite32( T_IER0,     0x00000000); // No interrupt source enabled
   RegWrite32( T_IER1,     0x00000000); // No interrupt source enabled

   /* Map Interrupt sources to Interrupt pins (INT0-INT5 for I/O usage) */

   RegWrite32( T_IDR,      0x5F000000); // Interrupt dir: -INT5 only as input
   RegWrite32( T_IMR_MBOX, 0x44440000); // MBOX0-3 to -INTA, MBOX4-7 to -INT0
   RegWrite32( T_IMR_DB,   0xECA86420); // DB0 to -INTA, DB2:7 to -INT0:5
   RegWrite32( T_IMR_DMA,  0x00004400); // DMAX0-1 to -INTA, DMA2-3 to -INT0
   RegWrite32( T_IMR_HW,   0x02ECA864); // No map from one int pin to an
other
   RegWrite32( T_IMR_P1,   0x00000000); // PCI 1 errors on PCI 1 -INTA
   RegWrite32( T_IMR_PB,   0x22222222); // PB errors on -INT0
   RegWrite32( T_IMR2_PB,  0x22200000); // PB errors on -INT0
   RegWrite32( T_IMR_MISC, 0x20000000); // I2O host to -INTA, I2O IOP to -
INT0
   RegWrite32( T_IER0,     0x001F0000); // INT0-4 HW interrupts enabled
}
```

PowerQUICC II Hard Reset Configuration Word

When the PowerQUICC II hardware reset signal is de-asserted, the PowerQUICC II generates 64-bit reads into its boot memory (the FLASH) with addresses starting at 0 and incremented by 8. The first eight bytes set its Hard Reset Configuration.

For the 4539, the PowerQUICC II Hard Reset Configuration is (must be):

- EARB = 0: Internal bus arbitration
- EXMC = 0: The internal memory controller is used
- CDIS = 0: The core is active
- EBM = 1: 60x-compatible bus mode
- BPS = 01: 8-bit boot port size
- –CIP = 0: Initial vector table base address is 0xFFF0 0000
- ISPS = 0: Responds as 64-bit slave to 64-bit masters
- L2CPC = 10: L2 cache pins configured as BADDR
- DPPC = 00: Data parity pins used for interrupt signals IRQ1–7
- ISB = 110: Internal Memory Mapped Register base address is 0xFF00 0000
- BMS = 0: Boot memory space is 0xFE00 0000
- BBD = 0: Bus Busy pins are enabled

- MMR =11: External bus requests are masked (PQ2 is the boot master)
- LBPC = 00: Local bus enabled
- APPC = 10: Address parity pins used for bank select
- CS10PC =01: –CS10/–BCTL1 used as –BCTL1
- MODCK_H =0101: PLL multiplication factors: with MODCK[1:3]=111, Bus@66, CPM@133, Core@200 MHz; with MODCK[1:3]=101, Bus@66, CPM@133, Core@133 MHz

PowerQUICC II Initializations

After a power-up or a reset exception, the PowerQUICC II must initialize itself and adapt its System Interface Unit (SIU) to the 4539 hardware. It must set up its memory controllers and Chip Selects. Then it must also initialize the SDRAM devices, before using them as its system memory.

PowerQUICC II System Interface Unit (SIU) Initialization

The PowerQUICC II SIU includes the following elements:

- System configuration and protection
- System reset monitoring and generation
- Clock synthesizer
- Power management
- 60x bus interface
- Memory Control Units

Several registers of the SIU need to be initialized during boot time for proper operation.

Internal Memory Map Register (IMMR)

The PowerQUICC II IMMR register is normally properly set in the Reset Configuration Word to map the PowerQUICC II Internal registers to address 0xFF010000.

Bus Configuration Register (BCR)

Some fields of the BCR register are initialized by the Reset Configuration Word. Several other fields however, need to be initialized:

- EBM = 1: 60x bus mode
- APD = 010: Wait two cycles for ARTRY
- L2C = 0: No secondary cache
- L2D = 000: L2 cache hit delay (don’t care)
- PLDP = 0: Pipeline depth = 1
- EAV = 1: Drive full address on 60x bus
- ETM = 1: Enable Extended Transfer Mode

- LETM = 1: Enable Local Extended Transfer Mode
- NPQM = 000: PowerQUICC II master connected
- EXDD = 0: External Master Delay not disabled
- ISPS = 0: Internal Space Port Size = 64 bits

The resulting register value is BCR=0xA01C0000.

System Protection Control Register (SYPCR)

This register controls the software watchdog. It can be read at any time but can be written only once after system reset. During the first phases of a development, it may be simpler to disable the watchdog by setting SWE to 0 in this register just after reset.

The resulting register value is SYPCR=0xFFFFFFC0.

60x Bus Arbiter Registers (PPC_ACR, PPC_ALRH, and PPC_ALRL)

In the PPC_ACR register, the following fields must be initialized:

- DBGD = 1: Assert –DBG after –TS (needed if bus is parked on the PowerSpan)
- EARB = 0: Internal Arbiter used
- PRKM = 0110: Bus parked on internal PowerPC core

Registers PPC_ALRH and PPC_ALRL define the priorities of the various bus masters. On the 60x bus the recommended priority order is as follows (from the highest to the lowest):

- CPM high priority: highest
- CPM middle priority
- CPM low priority
- External Master (the PowerSpan)
- PowerPC core

The resulting registers values are: PPC_ACR = 0x26, PPC_ALRH = 0x01276345, and PPC_ALRL = 0x89ABCDEF.

Local Bus Arbiter Registers (LCL_ACR, LCL_ALRH, and LCL_ALRL)

In the LCL_ACR register, the following fields must be initialized:

- DBGD = 0: No delay for –DBG
- PRKM = 0000: Bus parked on CPM high priority

Registers LCL_ALRH and LCL_ALRL define the priorities of the various bus masters. On the local bus, the recommended priority order is as follows (from the highest to the lowest):

- CPM high priority: highest
- CPM middle priority
- CPM low priority

- PowerPC core
- Other masters are not important

The resulting registers values are LCL_ACR = 0x00, LCL_ALRH = 0x01234567, and LCL_ALRL = 0x89ABCDEF

SIU Module Configuration Register (SIUMCR)

The SIUMCR register configures various features in the SIU module, among them the configuration of several multifunction pins. Its fields must be set as follows:

- BBD = 0: –ABB and –DBB enabled
- ESE = 1: –GBL/–IRQ1 pin used as –GBL
- PBSE = 0: –PPBS/PGPL4 used as PGPL4
- CDIS = 0: Core is enabled
- DPPC = 00: –IRQ/DP pins used as –IRQ
- L2CPC = 10: L2 cache pins configured as BADDR
- LBPC = 00: Local bus pins used as local bus
- APPC = 00: Address Parity pins used as local bus
- CS10PC = 01: –CS10/–BCTL1 used as –BCTL1
- BCTLC = 01: –BCTL0 used as R/–W and –BCTL1 used as –OE
- MMR = 11: External bus requests initially masked at boot, then
  MMR = 00: No bus request masking once booted
- LPBSE = 0: LBPS/LGPL4 functions as LGPL4

The resulting register value is SIUMCR=0x4205C000.

Bus Transfer Error Registers (TESCR1 and L_TESCR1)

Since there is no parity checking on the 4539, data errors must be disabled (field DMD=1 in registers TESCR1 and L_TESCR1).

Memory Controllers

The PowerQUICC II includes sophisticated memory controller units: a General Purpose Chip-select Machine (GPCM), three User Programmable Machines (UPMs) and two SDRAM control machines. These units are used on the 4539 to control all the external devices, except the PowerSpan, which is directly a 60x bus compatible device.

The memory controller unit to be used is defined bank per bank. Each bank is defined by its Base Register (BRx) and its Option Register (ORx). The memory machine selection is done in the Option register.

Table 2-2. PowerQUICC II Memory Controller Machine Usage

Element Accessed	Bank	Memory Controller	ORx Value	BRx Value
FLASH EEPROM	0	60x bus GPCM	0xFF80 0882	0xFF80 0801
60x bus Main memory	1	60x bus SDRAM machine	0xFC00 2CD0	0x00000041
Local bus connection memory	2	Local bus SDRAM machine	0xFF80 3480	0x0800 1861
CAM Control Port	3	UPMB	0xFFFF 8100	0xF004 10A1
CAM Match Port	5	UPMA	0xFFFF 9100	0xF006 1885
IMA device (MT90220/1)	6	60x bus GPCM	0xFFFF 0C86	0xF00C 0801
Quad T1/E1/J1 framer (QuadFALC)	7	60x bus GPCM	0xFFFF 0C86	0xF008 0801

SDRAM Controller and SDRAM Device Initialization

For the SDRAM controller, a specific PowerQUICC II register (PSDMR for the 60x SDRAM controller and LSDMR for the local SDRAM controller) is used to configure operations pertaining to the SDRAM. This register includes several configuration fields and one Operation field (OP). This Operation field must be used to generate all the special accesses needed to initialize the SDRAM, such as the precharges, the refreshes, and the SDRAM internal Mode register write. This will be useful for generating the complete SDRAM initialization sequence.

To generate a special access, one must first set the OP field in the xSDMR register, and then generate a dummy access to the SDRAM memory.

The sequence for SDRAM device initialization is as follows:

- Precharge all banks (OP=101)
- Refresh the SDRAM eight times (OP=001)
- Write the SDRAM Mode register (OP=011). For the main SDRAM placed on the 60x bus, the row/column address multiplexing is done externally, so the mode register value must be coded in the column address of the dummy access following the PSDMR programming.
- Reset the xDMR register OP field for normal operation (OP=000).

The refresh periods for the SDRAM devices are defined by one common Memory Refresh Timer Prescaler Register (MPTPR) and by two individual SDRAM Refresh Timer Registers (PSTR for the 60x bus and LSTR for the local bus).

GPCM Controller Initialization

The initialization of a GPCM controller is done entirely in the bank Option Register (ORx).

UPM Controller Programming

User Programmable Machine A (UPMA) is used to control accesses to the CAM match port. For each access, there is a write to the match port, a small delay, and then a read to the same match port.

The whole sequence is coded in a RAM array on which each line describes the pattern of the chip select signal, the five GPLx signals, and the byte select signals during one clock period.

This RAM array must be initialized with the correct values through registers MDR (Memory Data Register) and MxMR (Machine x Mode Register).

Table 2-3. PowerQUICC II UPM Signal Usage for CAM Match Port

UPMA Signal	CAM Memory Signal
–CS5	–LHCAM
LGPL5 (through an external inverter)	–GCAM

User Programmable Machine B (UPMB) is used to control accesses to the CAM control port, although no special UPMB signal is used.

MPC603e Core Initialization

For full description of the MPC603e registers, read Motorola documents: *MPC603e a EC606e RISC Microprocessors User's Manual* (ref MPC603EUM/AD) and *PowerPC Microprocessor Family: The Programmer's Reference Guide* (ref MPRPPCPRG–01).

MMU Initialization

The 4539 local memory mapping is organized in such a way that the Block Address Translation (BAT) mechanism can be used rather than the more complicated Segments and Translation Look-aside Buffers (TLB) mechanism.

In the Boot Firmware, the MMU is initialized using the BAT mechanism. The cachable areas are defined in the BAT blocks. Once the IBATx and DBATx special purpose registers initialized, Address Translation is enabled for instruction and data in the Machine State Register (MSR).

Cache Initialization

The data and instruction caches are automatically invalidated after a power-up or after a hard reset, but not after a soft reset. The content of the instruction and data caches are easily invalidated, using the Instruction Cache FLASH Invalidate (ICFI) and the Data Cache FLASH Invalidate (DCFI) control bits in the HID0 register. Each bit must be set and cleared in two consecutive moves to SPR (mtspr) operations to the HID0 register.

The instruction and data caches are enabled through bits ICE and DCE of register HID0 respectively. The setting of ICE bit must be preceded by an **isync** instruction. The setting of DCE bit must be preceded by a **sync** instruction.

Communication Processor Module Initialization

I/O Port Initialization

The CPM I/O ports have to be configured according to their usage (see *Communication Processor Module (CPM) I/O Ports* on page 8). In the Interphase boot firmware, a first minimal initialization is done during the early phase of the boot (in `startup.asm`). Then, later, the ports are re-initialized with their full usage configuration.

Each CPM port is set by five registers in the Internal Register Area: PDIRx, PPARx, PSORx, PODRx, and PDATx

CPM RCCR Reset

At boot, it is important to reset the RISC Controller Configuration Register (RCCR) in order to disable any previously loaded CPM microcode and start with the default known CPM microcode.

4539 Startup Code

The startup code is written in assembly language. As long as the system memory is not initialized, no stack or variable can be used. It includes the following steps:

1. PowerQUICC II Hardware Configuration Word.
2. Reset and other exceptions entry points.
3. PowerQUICC II SIU initialization.
4. PowerQUICC II memory controllers UPM and GPCM initialization.
5. SDRAM memory initialization sequence.
6. PowerQUICC II MMU initialization.
7. PowerQUICC II caches initialization.
8. CPM ports initialization.
9. CPM RCCR reset.
10. Delay before accessing the PowerSpan to allow its serial EEPROM load to complete.
11. Various PowerSpan initializations (arbiter, address translation mode, mailboxes).
12. PowerSpan interrupt pin configuration.
13. System SDRAM and local SDRAM memory tests.

14. Stack initialization for C-language code.

15. Jump to the rest of the code in C-language.

Programming the Peripherals 3

Overview

This chapter provides information specific to the 4539 board for peripheral programming. Its initial purpose is not to detail how to program the peripherals themselves, for which the developers should refer to the manufacturers data sheets. However, for tricky peripherals, such as T1/E1/J1 framers, some important register programming is detailed. For more details, refer to the 4539 Boot Firmware sources provided with the CD-ROM and referenced (in *italics*) in this chapter. See also the *4539 Built-In Self Test and Monitor Users Manual* (UG04539-004).

NOTE

The 4539 Boot Firmware initializes systems registers (see `app\asm\startup.asm` source file), but resets the CPM, and asserts the reset pins of the peripherals (see `app\c\main.c`, function `gvMonFWPreInit`) before launching an Operational Firmware. Developers of Operational Firmware should not rely on the 4539 Boot Firmware for peripherals initialization.

TDM Bus Configurations

Developers can find descriptions of the different TDM Bus Configurations in *TDM Bus Configurations* on page 40. In the 4539 Boot Firmware, these configurations are called 'Operating Mode' and developers can scan the sources and look for `OPER_MODE_`... for special treatments (operating modes are defined in file `sys\h\parports.h` - see Table 3-1). Routine `gbSetBoardMode` in the `sys\c\parports.c` file can be used to set the board in one of these operating modes – afterwards, during QuadFALC initialization, the operating mode is tested (see `c\tst\qfalc.c`) to configure the registers as they require.

.

Table 3-1. Operating Modes

Mode	Description
OPER_MODE_IDLE	This mode is not an operating mode – its purpose is to configure the board in order to prevent a conflict with external devices that could be connected on PMC P3 and P4. When setting a board in an operational mode, first set the board in this IDLE mode.
OPER_MODE_MD	Multiplexed Direct mode
OPER_MODE_SD	Switched mode

Table 3-1. Operating Modes (cont)

Mode	Description
OPER_MODE_IMAUNI	IMA/UNI mode. Use this mode to perform ATM on T1/E1/J1 ports. Note: The 4539 Boot Firmware does not use the UTOPIA/MPHY feature and therefore sets the UTOPIA addressing pins as general purpose outputs – for an operational code, this should not be used.
OPER_MODE_P1234	Pass-through mode; port 0 to 1, port 2 to 3
OPER_MODE_P1243	Pass-through mode; port 0 to 1, port 3 to 2
OPER_MODE_P2134	Pass-through mode; port 1 to 0, port 2 to 3
OPER_MODE_P2143	Pass-through mode; port 1 to 0, port 3 to 2
OPER_MODE_ID	Independent Direct mode. Use this mode to perform HDLC (by using MCCs) on T1/E1/J1 ports.
OPER_MODE_ID_IMA3	This mode is used to perform HDLC on ports 0, 1, 2 and ATM on port 3.

PowerQUICC II CPM Initialization

The different functions on the CPM are used as follows:

- MCC1 connected to SI1, using TSA1 (128 time slots)
- MCC2 connected to SI2, using TSA2 (128 time slots)
- FCC1 connected to UTOPIA on P3 connector
- FCC2 connected to UTOPIA towards IMA chip
- FCC3 connected to MII interface for Fast Ethernet
- SMC1 used for TTY interface

Parallel Ports Initialization

See Boot Firmware sources: sys\h\4539.h.

Procedure **gvParPortsInit** in the c\sys\parports.c file, initializes the parallel ports in a default configuration. Use this procedure to pre-initialize the board and then use the **gbSetBoardMode** (see *TDM Bus Configurations* on page 83) to set the board in a an operating mode.

Example:

To configure the board to run HDLC on T1/E1/J1 ports:

```
gvParPortsInit()

gbSetBoardMode(OPER_MODE_IDLE)

gbSetBoardMode(OPER_MODE_ID)
```

(Then Initialize the QuadFALC chip, the MCCs, etc.)

To change the configuration to run ATM on T1/E1/J1 ports:

`gbSetBoardMode(OPER_MODE_IDLE)`

`gbSetBoardMode(OPER_MODE_IMAUNI)`

(Then initialize the QuadFALC chip, IMA chip, FCC2, etc.)

Serial Interfaces and Time-Slot Assigner Initialization

In the CPM, the Time-Slot Assigners (TSAs) are parts of the Serial Interfaces (SIs).

Most TSA programming is done in two 256x16bits SIx RAMs per SI: one for receive and one for transmit. These SIx RAMs are in the PowerQUICC II internal register area, they are not a part of the PowerQUICC II internal Dual-Port RAM (DPRAM). The programming of each entry in the SIx RAM determines the routing of a group of serial bits.

Modes Others Than Pass-Through

To configure TDMa1 and TDMa2, SI1AMR and SI2AMR registers must be set as follows:

- Reserved = 0: This bit should be cleared
- SADx = 000: Starting bank address is first bank, first 32 entries
- SDMx = 00: Normal operation (no loopback or echo)
- RFSDx = 00: No receive frame sync delay
- DSCx = 0: Clock rate = data rate
- CTRx = 1: Common receive and transmit pin clocks
- SLx = 1: Active synchronization level is low
- CEx = 1: Data sent on falling edge and received on rising edge
- FEx = 0: Frame Sync edge is falling edge
- GMx = 0: Grant mode not used
- TFSDx = 00: No transmit frame delay

Final result of the SI1AMR and SI2AMR registers is 0x0070.

To configure TDMc1 and TDMc2, SI1CMR and SI2CMR registers must be set as follows:

- Reserved = 0: This bit should be cleared
- SADx = 100: Starting bank address is third bank, first 32 entries
- SDMx = 00: Normal operation (no loopback or echo)
- RFSDx = 00: No receive frame sync delay
- DSCx = 0: Clock rate = data rate
- CTRx = 1: Common receive and transmit pin clocks
- SLx = 1: Active synchronization level is low
- CEx = 1: Data sent on falling edge and received on rising edge
- FEx = 0: Frame Sync edge is for falling edge

- GMx = 0: Grant mode not used
- TFSDx = 00: No transmit frame delay

Final result of the SI1CMR and SI2CMR registers is 0x4070.

Pass-Through Mode

To configure TDMa1 and TDMa2, SI1AMR and SI2AMR registers must be set as follows:

- Reserved = 0: This bit should be cleared
- SADx = 000: Starting bank address is first bank, first 32 entries
- SDMx = 00: Normal operation (no loopback or echo)
- RFSDx = 00: No receive frame sync delay
- DSCx = 0: Clock rate = data rate
- CTRx = 1: Common receive and transmit pin clocks
- SLx = 1: Active synchronization level is low
- CEx = 0: Data sent on rising edge and received on falling edge
- FEx = 1: Frame Sync edge is rising edge
- GMx = 0: Grant mode not used
- TFSDx = 00: No transmit frame delay

Final result of the SI1AMR and SI2AMR registers is 0x0068.

To configure TDMc1 and TDMc2, SI1CMR and SI2CMR registers must be set as follows:

- Reserved = 0: This bit should be cleared
- SADx = 100: Starting bank address is third bank, first 32 entries
- SDMx = 00: Normal operation (no loop-back or echo)
- RFSDx = 00: No receive frame sync delay
- DSCx = 0: Clock rate = data rate
- CTRx = 1: Common receive and transmit pin clocks
- SLx = 1: Active synchronization level is low
- CEx = 0: Data sent on rising edge and received on falling edge
- FEx = 1: Frame Sync edge is rising edge
- GMx = 0: Grant mode not used
- TFSDx = 00: No transmit frame delay

Final result of the SI1CMR and SI2CMR registers is 0x4068.

Since TDMbx, and TDMdx TDMs are not used, it is not necessary to initialize the corresponding SIxMR register.
See Boot Firmware sources: tst\c\pqtdm.c - *Function vPQTDM_SI_Init_PQII.*

Clocks and Baud-Rate Generators

Introduction

The CPM contains eight independent, identical, Baud-Rate Generators (BRGs) that can be used with the FCCs, SCCs, and SMCs. The clocks produced by the BRGs are sent to the bank-of-clocks selection logic, where they can be routed to the controllers. In addition, the output of a BRG can be routed to a pin to be used externally.

BRGCLK

The BRGCLK is an internal signal generated in the MPC8260 clock synthesizer specifically for the BRGs, the SPI, and the I^2C internal BRG. BRGCLK is itself sourced from VCO_OUT (twice the CPM clock) which is at 266.144 KHz. The DFBRG field of SCCR must be programmed to 01, so that BRGCLK equals VCO_OUT/16 (16384 KHz). For more information on SCCR and DFBRG fields, see the MPC8260 PowerQUICC II Users Manual.
See Boot Firmware sources: app\c\main.c.

BRG1 – TTY Baud-Rate Generator

The TTY interface is controlled by SMC1.

SMC1 baud-rate generator is BRG1.

Configure the CMXSMR register as follows:

- SMC1 = 0: SMC1 is not connected to TSA
- Reserved = 0: This bit should be cleared
- SMC1CS = 00: SMC1 transmit and receive clocks are BRG1
- SMC2 = 0: SMC2 is not connected to TSA (don't care)
- Reserved = 0: This bit should be cleared
- SMC2CS = 00: SMC2 transmit and receive clocks are BRG2 (don't care)

For more information on CMXSMR fields, see the MPC8260 PowerQUICC II Users Manual.

The DIV16 field of BRGC1 register must be set to 0, so the first BRG1 divider will divide the received BRGCLK clock by 1 and will use the 16384 KHz clock.

To provide the proper baud-rate value (2400, 4800, 9600 baud), the SMC1 clock source must be 16 times the rate of the line (see BRGC1 register).
See Boot Firmware sources: app\c\montty.c - *Function gwMonTTYOpen*.

CLK15 as FCC2/ATM Rx and Tx Clocks

To use FCC2 NMSI pins and assign CLK15 as clock for FCC2 (ATM controller) Rx and Tx clocks on FCC2 NMSIx pins, configure FCC2 fields in the CMXFCR register as follows:

- FC2 = 0: FCC2 is not connected to the TSA but NMSIx pins
- RF2CS = 110: FCC2 receive clock is CLK15
- TF2CS = 110: FCC2 transmit clock is CLK15

See Boot Firmware sources: tst\c\pqatm.c - Function FCC_init.

CLK10/CLK12 as FCC1/ATM Rx/Tx Clocks

To use FCC1 NMSI pins and assign CLK10 as Rx clock and CLK12 as Tx clock for FCC1 (ATM controller), configure FCC1 fields in the CMXFCR register as follows:

- FC1 = 0: FCC1 is not connected to the TSA but NMSIx pins
- RF1CS = 101: FCC1 receive clock is CLK10
- TF1CS = 111: FCC1 transmit clock is CLK12

See Boot Firmware sources: tst\c\pqatm.c - Function FCC_init.

CLK15 /BRG4 – T1/E1/J1 Framers Chip (QuadFALC) Master Clock

The QuadFALC MCLK master clock is a 12.5 MHz clock sourced from BRG4 (as output from the PowerQUICC II) derived from CLK15 25 MHz PowerQUICC II input.

To configure CLK15 as the source clock (divided by 2) for BRG4, configure BRGC4 register fields as follows:

- Bits 0-13 are reserved. Set to 0
- RST (Reset BRG) = 0 = Enable the BRG
- EN (Enable BRG Count) = 1 = Enable clocks to the BRG
- EXTC (External Clock Source) = 10 = Source is CLK15
- ATB (AutoBaud) = 0 = Normal operation of the BRG
- CD (Clock Divider) = 0x001 = Divide by 1 to set BRG4 at 12.5 MHz
- DIV16 (Divide-by-16) = 0 = Divide by 1

See Boot Firmware sources: tst\c\qfalc.c - Function gwQFalcInit.

BRG5 When Using ATM Transmit Internal Rate Mode

BRG5 may be used for ATM Transmit Internal Rate Mode. The user must be aware that this mode limits the number of ATM PHY addresses to four – a 4539 board equipped with a MITEL MT90220 controller is able to handle up to eight ATM PHY interfaces. However, if you want to use BRG5 for ATM Transmit Internal Rate, do the following:

1. Program the BRGC5 register. BRG5 is sourced from BRGCLK (EXTC bits) – set the source predivider (DIV16) and the 12-bit prescaler (CD bits). DIV16 and CD should be set according to the intended maximum rate on UTOPIA PHY devices.
2. Define BRG5 as Internal Rate Mode BRG CLK on FCC2 by setting F2IRB in CMXUAR to 00.

See Boot Firmware sources: tst\c\pqatm.c - Function FCC_Init.

BRG2 as FCC1/ATM Master Clock Source

When running FCC1/ATM as UTOPIA master, BRG2 must be used as clock source – BRG2 must be itself programmed as sourced from BRGCLK. The maximum frequency of BRG2 is that of BRGCLK. Current documents initialize SCCR to 0x01 providing a BRGCLK at 16384 KHz – for a greater BRG2 rate, SCCR and other BRGCx registers programming should be changed.

To configure BRG2 at 16384 KHz (presuming SCCR=0x01), configure BRGC2 register fields as follows:

- Bits 0-13 are reserved. Set to 0
- RST (Reset BRG) = 0 = Enable the BRG
- EN (Enable BRG Count) = 1 = Enable clocks to the BRG
- EXTC (External Clock Source) = 00 = Source is BRGCLK
- ATB (AutoBaud) = 0 = Normal operation of the BRG
- CD (Clock Divider) = 0x000 = Divide by 1 to set BRG2 at BRGCLK
- DIV16 (Divide-by-16) = 0 = Divide by 1

Final result of the BRGC2 register is 0x00010000.

See Boot Firmware sources: tst\c\pqatm.c - Function qwPQATMInit.

NOTE

If FCC1/ATM is UTOPIA slave, BRGC2 output should remain in high impedance by setting RST bit to 1 in BRGC2 register; otherwise, a short-circuit would occur between PD(17) (BRG2 output) and UTOPIA clocks inputs on PMC P3 connector.

NOTE

For FCC1/ATM UTOPIA operation, PC(30) must be set to 0 in master mode and 1 in slave mode. For Rx master and Tx slave configuration set PC(30) to 1. Tx master and Rx slave is not a possible operational configuration (this configuration is used for UEL test in 4539 Boot Firmware - see *4539 BIST and Monitor Manual* (UG04539-004)).

MCC Initialization

MCC channels are assigned to PowerQUICC TDMs by programming MCCF1 and MCCF2 registers. This programming depends on the board operation mode (independent direct mode, switched mode, etc.). The following MCCFx initializations are provided for the most common applications.

Multiplexed Direct and Switched Modes

Only TDMa1 is used – channels in range [0,127] are assigned to TDMa1. Since no TDMx2 is used, MCCF2 programming has no importance.

MCCF1 register initialization:

- Group 1 = 00: Group 1 (MCC channels 0-31) is used by TDMa1
- Group 2 = 00: Group 2 (MCC channels 32-63) is used by TDMa1
- Group 3 = 00: Group 3 (MCC channels 64-95) is used by TDMa1
- Group 4 = 00: Group 4 (MCC channels 96-127) is used by TDMa1

Final Result of MCCF1 register is 0x00.

Other Modes

Only TDMa1, TDMa2, TDMc1 and TDMc2 are used.

MCCF1 register initialization:

- Group 1 = 00: Group 1 (MCC channels 0-31) is used by TDMa1
- Group 2 = 00: Group 2 (MCC channels 32-63) is used by TDMa1
- Group 3 = 10: Group 3 (MCC channels 64-95) is used by TDMc1
- Group 4 = 10: Group 4 (MCC channels 96-127) is used by TDMc1

Final Result of MCCF1 register is 0x0A.

Note: 4539 Boot Firmware uses an alternate initialization – MCCF1 = 0x22:

- Group 1 = 00: Group 1 (MCC channels 128-159) is used by TDMa2
- Group 2 = 10: Group 2 (MCC channels 160-191) is used by TDMc2
- Group 3 = 00: Group 3 (MCC channels 192-223) is used by TDMa2
- Group 4 = 10: Group 4 (MCC channels 224-255) is used by TDMc2

MCCF2 register initialization:

- Group 1 = 00: Group 1 (MCC channels 128-159) is used by TDMa2
- Group 2 = 00: Group 2 (MCC channels 160-191) is used by TDMa2
- Group 3 = 10: Group 3 (MCC channels 192-223) is used by TDMc2
- Group 4 = 10: Group 4 (MCC channels 224-255) is used by TDMc2

Final Result of MCCF2 register is 0x0A.

Note: 4539 Boot Firmware uses an alternate initialization – MCCF2 = 0x22:

- Group 1 = 00: Group 1 (MCC channels 128-159) is used by TDMa2
- Group 2 = 10: Group 2 (MCC channels 160-191) is used by TDMc2
- Group 3 = 00: Group 3 (MCC channels 192-223) is used by TDMa2
- Group 4 = 10: Group 4 (MCC channels 224-255) is used by TDMc2

For details on MCC Initialization, *see Boot Firmware sources: tst\c\pqtdm.c Function vPQTDM_MCC_Init_PQII.*

NOTE

The MCCs must be initialized before connecting to them in the SIRAM, otherwise unpredictable errors, such as underruns will occur.

FCC2/ATM Framer Initialization

Setting ATM Mode for FCC2

FCC2 is used for ATM; this is configured in GFMR2 register by setting the MODE field to 1010. Other fields should be set to 0, except ENR and ENT bits that have to be set to 1, to activate reception and transmission respectively. For test purposes, the DIAG field allows configuring the FCC2 in loopback.

ATM UTOPIA Bus Setting

The ATM bus, which ties FCC2 (the ATM layer) to the MITEL MT90220/1 IMA chip (the PHY device), must be configured as a UTOPIA level 2 MPHY interface with 8-bit wide data busses.

The MITEL MT92220/1 provides one TxClav, one RxClav signal, and five addressing signals per direction (RxAddr[4:0], TxAddr[4:0]). The FCC2s UTOPIA bus must therefore be configured in Master Multiplexed Polling Mode. On the IMA chip, RxAddr[4] and TxAddr[4] are not used and connected to 0V (Logic value 0).

The IMA chip does not support the UTOPIA parity signals (RxPrty, TxPrty): this feature must be disabled on the FCC (see FPSMR register).
See Boot Firmware sources: tst\c\pqatm.c - Function FCC_Init.

An operational software should configure FPSMR register fields as follows (the fields for which the initialization should not change in an operational software are shown in **bold**):

- TEHS = 0000: Transmit extra header size. Generate one bit of extra header.
- REHS = 0000: Receive extra header size. Expect one bit of extra header.
- ICD = 0: Idle cells discard (to reduce the number of cells exchanged on UTOPIA reception, idle cells should be filtered by the IMA chip).
- **TUMS = 0: Transmit UTOPIA master mode is selected.**
- **RUMS = 0: Receive UTOPIA master mode is selected.**
- **LAST PHY = 00011: Address of last UTOPIA address in Multi-PHY master mode (assume that UTOPIA addresses 0 to 3 have been defined in the IMA chip to handle the four TDMs of a MITEL MT90221 - on a board equipped with a MITEL**

MT90220, this field should be initialized to 00111 to handle up to the eight TDMs).

- **Reserved = 000:** **These three bits should be cleared.**
- TUDC = 0: Transmit user-defined cells. 0 means regular 53-byte cells.
- RUDC = 0: Receive user-defined cells. 0 means regular 53-byte cells.
- **RXP = 1:** **Receive parity check. 1 for not checking receive parity.**
- **TUMP = 1:** **Transmit UTOPIA multiple PHY mode. 1 for multiple mode selected.**
- **Reserved = 0:** **This bit should be cleared.**
- **TSIZE = 0:** **Transmit UTOPIA data bus size. 0 means 8-bit data bus size.**
- **RSIZE = 0:** **Receive UTOPIA data bus size. 0 means 8-bit data bus size.**
- UPRM = 0: UTOPIA priority mode. 0 means round robin.
- **UPLM = 0:** **UTOPIA polling mode. 0 means multiplex polling.**
- **RUMP = 1:** **Receive UTOPIA multiple PHY mode. 1 for multiple PHY mode selected.**
- HECI = 0: HEC included. Used in UDC mode only.
- **Reserved = 00:** **These two bits should be cleared.**

Final Result of FPSMR register is 0x00030608 (0x00070608 if board is equipped with an MT90220).

NOTE

4539 Boot Firmware ATM tests do not use the MPHY feature and therefore FPSMR register is initialized to 0x00000400 (*See Boot Firmware sources: tst\c\pqatm.c-Function FCC_init*); for an operational software, the MPHY feature should be used and FPSMR initialized to 0x00030608 (0x00070608 for a MT90220).

CAM Initialization for ATM

All information needed to initialize the CAM memory is present on Boot Firmware file `/tst/c/cam.c`.

When using CAM for ATM, the user should set the EXT_CAM_BASE field of *MPC8260 ATM Parameter RAM* to the Match Port address.

The initialization of the CAMs must be performed through the Control Ports. The insertion/removal of an entry must also be done through the Control Ports.

The ATM controller (FCC2) uses the Match Port to search the Channel Code of an ATM channel (an ATM PHY, GFC+VPI, VCI combination). The Channel Code is used by the ATM controller as an index to its Receive Connection Table.

An operation on the CAM Match Port consists of one write followed by one read. The ATM controller writes the ATM PHY-GFC+VPI-VCI combination and reads the Channel Code. Between these two accesses, no other access must occur, otherwise the result of the read is corrupted. Therefore, Bank Register BR5 used for this port must be configured in “Read after Write ATOMIC operation” mode.
See Boot Firmware sources: APP\ASM\STARTUP.ASM (search BR5).

For details on CAM Initialization, *see Boot Firmware sources: tst\c\cam.c.*

T1/E1/J1 Framers Initialization

Introduction

This section details the QuadFALC registers initialization, assuming that for non-specified registers, the initialization is the default value (which is generally 0x00). 4539 Boot Firmware sources provide routines to initialize the framers in T1, E1, or J1 mode. Developers should refer to them (*see Boot Firmware sources: tst\c\qfalc.c*).

NOTE

At the end of a QuadFALC port configuration register initialization, it is recommended that you reset the transmitter and receiver by setting XRES and RRES bits in the CMDR register. *See Boot Firmware sources: tst\c\qfalc.c - Function vFalcWriteCMDR.*

Master Clock Initialization

The Master Clock provided on the MCLK pin of the QuadFALC devices is at 12.5 MHz (see *CLK15 /BRG4 – T1/E1/J1 Framers Chip (QuadFALC) Master Clock* on page 88).
See Boot Firmware sources: tst\c\qfalc.c.

Table 3-2. GCM Register Programming

Register	Value
GCM1	0x2B
GCM2	0x5D
GCM3	0xAC
GCM4	0x89
GCM5	0x07
GCM6	0x15

System Interface in Multiplexed Direct Mode

In multiplexed direct mode, the QuadFALC is connected to PowerQUICC II TDMa1 through an 8 MHz stream. This stream is the concatenation of four 2-MHz streams, corresponding to the four T1/E1/J1 lines. These four streams are mapped into this 8 MHz stream in an interleaved manner.

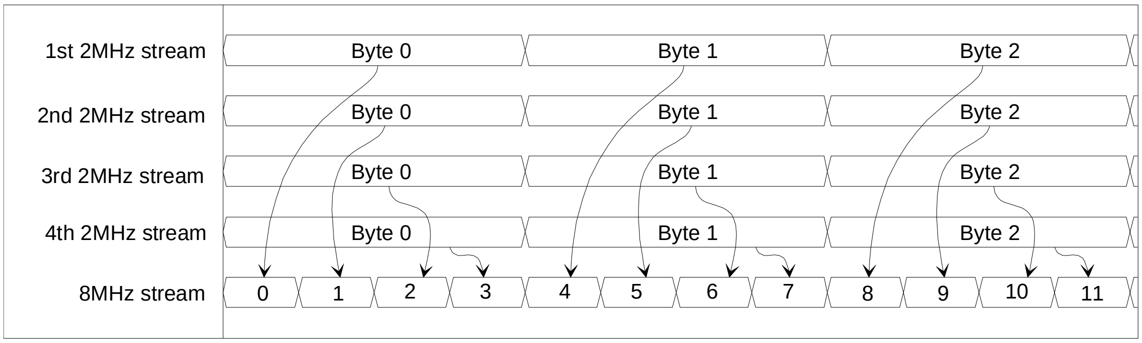


Figure 3-1. Mapping of Four 2 MHz Streams into an 8 MHz Stream

See Boot Firmware sources: tst\c\qfalc.c - Functions gvQFalcInitT1J1 and *gvQFalcInitE1*.

NOTE

TDMa1 time slots numbered 0, 1, 2, and 3 are not data time slots. They represent the framers framing and alarm information overwritten by the framers in the transmit direction. No MCC channels should be mapped on them.

The system multiplex mode must be enabled (GPC1.SMM = 1) with byte interleaved format (SIC1.BIM=0), clocking rate at 8.192 MHz (SIC1.SCC1/0=10) and data rate at 8.192 MBit/s (SIC1.SDD1=1, FMR1.SDD0=0). Time-slot offset programming was obtained by actual practice: XC0 = 0x00, XC1 = 0x04, RC0 = 0x00, RC1 = 0x04. The receive buffer size must be set to two frames (SIC1.RBS1/0 = 00). The transmit buffer size must be set to two frames (SIC1.XBS1/0 = 10). SIC3.RESX must be set to 1 (Synchronous Pulse Transmit (–SYPX) is latched on first clock rising edge), and SIC3.RESR must be set to 0 (Synchronous Pulse Receive (–SYPR) is clocked on first clock falling edge). –SYPX and –SYPR inputs are mapped to XPA1 and RPA1 pins respectively by setting the PC1 register to 0. SCLKX1 and SCLKR1 must be configured as inputs by setting PC5.CSXP and PC5.CSRP bits to 0. RCLK1 must be configured as an output by setting PC5.CRP bit to 1. All these initializations must be performed on each channel.

The multiplexed data stream is internally logically ored. Therefore, the selection of the active channel phase has to be configured differently for each single channel (1–4). Programming is done with SIC2.SICS2...0 bit as shown in Table 3-3.

NOTE

For T1/J1 applications, the mapping of the receive 24 line time slots over the 32 available on the system interface is configurable with FMR1.CTM bit. In 4539 Boot firmware, the choice is to select 'Channel translation mode 1', by setting FRM1.CTM bit to 1: on reception, the 24 line time slots are contiguously mapped before they are interleaved on the system bus. The same mapping occurs on transmission.

Table 3-3. Channel Phase Programming in Multiplexed System Data Streams

Channel	SIC2.SICS2...0
1	000
2	001
3	010
4	011

RCLK1 Signal

QuadFALC RCLK1 signal is connected to SCLKR1, SCLKX1 (QuadFALC pins configured as inputs) and PowerQUICC II TDMa1 L1RCLK input signal (on PC(31)). RCLK1 is configured as an active output by setting PC5.CRP bit to 1 on channel 0 (RCLK2, RCLK3, and RCLK4 pins have to be configured as inputs and so for channels 1, 2, and 3 leave the PC5.CRP bits at 0).

RCLK1 must be programmed as one of the four QuadFALC channels' DCO-R (a channel is a FALC within a QuadFALC) outputs: the channel selection is set with GPC1.R1S1 and GPC1.R1S0 bits – an active channel should be selected. RCLK1 must be programmed at 8.192 MHz: program CMR1.RS1=1, CMR1.RS0=1, SIC2.SSC2=0 to have channels' RCLK set to 8.192 MHz and also RLCK1 that derives from one of them. Program CMR1.DRSS1 and CMR.DRSS0 bits as shown in Table 3-4: each channel's DCO-R is synchronized by the clock recovered from the line on the same channel.

Table 3-4. QuadFALC Recovered Clock Source for Each DCO-R

Channel	CMR1.DRSS1	CMR1.DRSS0
1	0	0
2	0	1
3	1	0
4	1	1

SEC/FSC Signal

QuadFALC SEC/FSC signal is connected to RPA1 (QuadFALC pin configured as –SYSPR input) and PowerQUICC II TDMa1 L1RSYNC input signal (on PA(6)). SEC/FSC signal is configured as an active FSC output low by setting GPC1.CSFP1...0 to 11.

FSC source must match an active channel as for RCLK1: the source is selected with GPC1.FSS1 and GPC1.FSS0 bits.

System Interface in Switched Mode

QuadFALC initialization for switched mode should be identical to multiplexed direct mode (see *System Interface in Multiplexed Direct Mode* on page 94). The only difference depends on the PMC P4 CKIOH and FSOH signals shape (or phase) compared to QuadFALC signals RCLK1 and FSC respectively. If there are differences, the following register fields should be adapted: SIC3.RESR, RC0.RCx, RC1.RCx, SIC3.RESX, XC0.XCx, and XC1.XCx

System Interfaces in Independent Direct Mode

In independent direct mode, the QuadFALC framers' system busses are connected to PowerQUICC II TDMa1, TDMc1, TDMa2, and TDMc2 through 2 MHz streams.

NOTE

TDMa1, TDMc1, TDMa2, and TDMc2 time slots numbered 0 are not data time slots. They represent the framers framing and alarm information overwritten by the framers in the transmit direction. No MCC channels should be mapped on them.

The system multiplex mode must be disabled (GPC1.SMM = 0), clocking rate at 2.048 MHz (SIC1.SCC1/0=00) and data rate at 2.048 MBit/s (SIC1.SDD1=0, FMR1.SDD0=0). Time slot offset programming was obtained by actual practice: XC0 = 0x00, XC1 = 0x03, RC0 = 0x00, RC1 = 0x04. The receive buffer size must be set to two frames (SIC1.RBS1/0 = 00). The transmit buffer size must be set to two frames (SIC1.XBS1/0 = 10).

Set CMR2.IRSP and CMR2.IXSP bits to 1, to have the frame synchronization pulses for the receive and transmit system interfaces internally sourced by the DCO-R circuitry. Set PC1=0x10 to output the receive frame synchronization clocks on RPAx.

Set CMR2.IRSC and CMR2.IXSC bits to 1, to have the system bit clocks for the receive and transmit system interfaces internally sourced by the DCO-R circuitry. Set PC5.CSRP bit to 1 to output the receive system bit clocks on SCLKRx pin outputs.

SIC3.RESX must be set to 1 (Synchronous Pulse Transmit (–SYPX) is latched on the first clock rising edge), and SIC3.RESR must be set to 0 (Synchronous Pulse Receive (–SYPR) is clocked on the first clock falling edge).

All these initializations must be performed on each channel.

SEC/FSC Signal

QuadFALC SEC/FSC signal is output on RxREF P3 connector pin and can be used by an external device as a synchronization source. The SEC/FSC signal can be configured as an active FSC output low by setting GPC1.CSFP1...0 to 11. FSC source should match an active channel: the source is selected with GPC1.FSS1 and GPC1.FSS0 bits.

System Interfaces in Pass-Through Mode

In pass-through mode, the QuadFALC framers' system busses are connected to the PowerQUICC II in a special way (see *QuadFALC T1/E1/J1 Framer* on page 35) to two of the TDMs: TDMa1, TDMc1, TDMa2, and TDMc2 (depending on the type of pass-through sub-mode: [0->1 and 2->3] or [0->1 and 3->2] ...) through 2 MHz streams.

NOTE

TDMa1, TDMc1, TDMa2, and TDMc2 time slots numbered 0 are not data time slots. They represent the framers framing and alarm information overwritten by the framers in the transmit direction. No MCC channels should be mapped on them.

The system multiplex mode must be disabled (GPC1.SMM = 0), clocking rate at 2.048 MHz (SIC1.SCC1/0=00) and data rate at 2.048 MBit/s (SIC1.SDD1=0, FMR1.SDD0=0). Time slot offset programming was obtained by actual practice: XC0 = 0x00, XC1 = 0x04, RC0 = 0x00, RC1 = 0x04. The receive buffer size must be set to two frames (SIC1.RBS1/0 = 00). The transmit buffer size must be set to two frames (SIC1.XBS1/0 = 10). SIC3.RESX must be set to 0 (Synchronous Pulse Transmit (–SYPX) is latched on first clock falling edge), and SIC3.RESR must be set to 1 (Synchronous Pulse Receive (–SYPR) is clocked on first clock rising edge). –SYPX and –SYPR inputs are mapped to XPA1 and RPA1 pins respectively by setting the PC1 register to 0. SCLKX1 and SCLKR1 must be configured as inputs by setting PC5.CSXP and PC5.CSRP bits to 0. RCLK1 must be configured as output by setting PC5.CRP bit to 1. All these initializations must be performed on each channel.

SEC/FSC Signal

The QuadFALC SEC/FSC signal is output on RxREF P3 connector pin and can be used by an external device as a synchronization source. The SEC/FSC signal can be configured as an active FSC output low by setting GPC1.CSFP1...0 to 11. FSC source should match an active channel: the source is selected with GPC1.FSS1 and GPC1.FSS0 bits.

System Interfaces in IMA/UNI Mode

In IMA/UNI mode, the QuadFALC framer system busses are connected to IMA chip TDMs [0, 3].

The system multiplex mode must be disabled (GPC1.SMM = 0), clocking rate at 2.048 MHz (SIC1.SCC1/0=00) and data rate at 2.048 MBit/s (SIC1.SDD1=0, FMR1.SDD0=0). Time slot offset programming was obtained by actual practice: XC0 = 0x00, XC1 = 0x04 in E1 mode and 0x0C in T1/J1, RC0 = 0x00, RC1 = 0x04 in E1 mode and 0x0C in T1/J1. The receive buffer size must be set to two frames (SIC1.RBS1/0 = 00). The transmit buffer size must be set to two frames (SIC1.XBS1/0 = 10).

Set CMR2.IRSP and CMR2.IXSP bits to 1, to have the frame synchronization pulses for the receive and transmit system interfaces internally sourced by the DCO-R circuitry. Set PC1=0x10 to output the receive frame synchronization clocks on RPAx.

Set CMR2.IRSC and CMR2.IXSC bits to 1, to have the system bit clocks for the receive and transmit system interfaces internally sourced by the DCO-R circuitry. Set PC5.CSRP bit to 1 to output the receive system bit clocks on SCLKRx pin outputs.

SIC3.RESX must be set to 0 (Synchronous Pulse Transmit (–SYPX) is latched on first clock falling edge), and SIC3.RESR must be set to 1 (Synchronous Pulse Receive (–SYPR) is clocked on the first clock rising edge).

All these initializations must be performed on each channel.

NOTE

RC1 and XC1 programming values depend on the framer modes (E1 or T1/J1): this is due to the fact that the IMA chip (MITEL MT90220/1) starts mapping data (ATM cells) from time-slot 1 in E1 mode and 0 in T1/J1 modes. In T1/J1 modes, a shift of 8 bits must be performed between IMA chip TDMs and QuadFALC framer system TDMs.

SEC/FSC Signal

The QuadFALC SEC/FSC signal is output on RxREF P3 connector pin and can be used by an external device as a synchronization source. The SEC/FSC signal can be configured as an active FSC output low by setting GPC1.CSFP1...0 to 11. FSC source should match an active channel: the source is selected with GPC1.FSS1 and GPC1.FSS0 bits.

Framing and Line Coding Initialization

Common Initialization

Table 3-5. Common T1/E1/E1-CRC4 Initialization

Register Bit	Value	Comment
LIM1.DRS	0	The ternary interface is selected.
LIM2.SLT1...0	10	Receive slicer threshold = 50%
LIM1.RIL2...0	010	Line interface receive input threshold equals 0.6 V
PCD,PCR	0x0A,0x15	LOS is declared after 176 pulse positions without transitions and 0x16 pulses are required within 176 ms to clear an LOS alarm (fulfills G.775).

T1/J1 Specific Initialization

Table 3-6. T1/J1 Specific Initialization

Register Bit	Value	Comment
FMR1.PMOD	1	T1 mode
FMR4.FM1...0	2	24-frame multi-frame format (ESF)
FMR1.CRC	1	CRC6 check/generation enabled
FMR0.RC1...0	11	B8ZS serial line code for the receiver
FMR0.XC1...0	11	B8ZS serial line code for the transmitter
FRM4.AUTO	1	Automatic re-synchronization is enabled

Table 3-6. T1/J1 Specific Initialization (cont)

Register Bit	Value	Comment
FMR1.EDL	1	EDL bit is set to 1 to force FDL channel to 0 due to XDL1 and XDL2 registers defaulting to 0 – otherwise in some circumstances, false RAI alarms might be generated on the Tx stream (if EDL is not set to 1, FDL channel is fed by the backplane in high impedance on that channel).

J1 Specific Initialization

Table 3-7. J1 Specific Initialization

Register Bit	Value	Comment
FMR2.SSP	1	Terminal frame and multiframe synchronization are separated.
RC0.SJR	1	CRC6 calculation including FS/DL bits.

E1/E1-CRC4 Common Initialization

Table 3-8. E1/E1–CRC4 Common Initialization

Register Bit	Value	Comment
FMR1.PMOD	0	E1 mode
FMR0.RC1...0	11	HDB3 serial line code for the receiver
FMR0.XC1...0	11	HDB3 serial line code for the transmitter
XSW.XSIS	1	Spare bit for International use: not used
XSW.XY0...4	1111	Spare bits for National use: not used
XSP.XSIF	1	Transmit spare bit for international use (FAS Word) - not used
XSP.AXS	1	Automatic transmission of E-bits
XSP.EBP	1	E-bit polarity in asynchronous state set to 1
XSP.XS13	1	Transmit Spare bit frame 13 - not used
XSP.XS15	1	Transmit Spare bit frame 15 - not used

E1 non CRC4 Specific Initialization

Table 3-9. E1 Non CRC4 Specific Initialization

Register Bit	Value	Comments
FMR1.XFS	0	Transmit double-frame format.
FMR2.RFS1...0	00	Receive double-frame format.

E1-CRC4 Specific Initialization

Table 3-10. E1-CRC4 Specific Initialization.

Register Bit	Value	Comments
FMR1.XFS	1	Transmit CRC4 – multiframe format
FMR1.AFR	1	Automatic Force Re synchronization
FMR2.RFS1...0	10	Receive CRC4 – multiframe format.

Clock Synchronization Initialization

Slave Mode

Table 3-11. Slave Mode Initialization

Register Bit	Value	Comments
LIM0.MAS	0	Slave mode
LIM2.ELT	1	Enable Loop-Timed. Transmit clock is generated from the clock supplied by MCLK, which is synchronized to the extracted receive route clock.
FMR5.XTM	1	Disconnects the control of the transmit system interface from the transmitter.
CMR1.DXSS	0	The DCO-X circuitry synchronizes to the internal reference clock which is sourced by RCLK (assuming LIM1.RL=0 and LIM2.ELT = 1).

Master Mode

Table 3-12. Master Mode Initialization

Register Bit	Value	Comments
LIM0.MAS	1	Master mode.
LIM2.ELT	0	Normal operation – Loop-Timed disabled.
FMR5.XTM (T1) XSW.XTM (E1)	0	–SYSPX defines the frame beginning on the transmit system highway.
IPC.SSYF	0	Reference clock at port SYNC is 2.048 MHz.
LIM1.DCOC (T1)	1	For T1 only – 2.048 MHz reference clock for the DCO-R circuitry provided on pin SYNC.

Transmit Pulse Shape

For each type of Line Build-Out (LBO), the shape of the transmit pulse must be adjusted through QuadFALC registers LIM0, LIM2, XPM0, XPM1, and XPM2 in order to comply with FCC 68 or ANSI T1.403 (see Table 1-30 on page 36).

Line LED Control

For each T1/E1/J1 framer, there is one bi-color LED controlled by corresponding RPCi and RPDi pins configurations on the QuadFALC (see *LED Descriptions* on page 116). RPCi and RPDi pins of port i are configured through PC3 and PC4 registers of the same port i.

To switch off the LED: PC3 = PC4 = 0x70

To switch on the LED (Red): PC3= 0x20; PC4 = 0x70

To switch on the LED (Green): PC3= 0x70; PC4 = 0x20

Configuring a T1/E1/J1 Port In Internal Loopback

To configure a T1/E1/J1 port in internal loopback (so that HDLC data from MCC or ATM cells from FCC2 are echoed), just initialize the port normally and set LIM0.LL bit to 1 (*see 4539 Boot Firmware, File tst\c\falc.c, Function gvQFalcSetLLB*).

The IMA Device

NOTE

This section provides the 4539 board-dependant IMA register initializations only. User can find in the 4539 Boot Firmware the code necessary to initialize the IMA chip in UNI mode *(see Boot Firmware: tst\c\ima.c - Function vMT_UNIInit*). For IMA mode, refer to the device documentation.

The 4539 board-dependent IMA register field initializations are shown in bold in the following tables:

IMA TX PCM Control Register 1 (088 - 08B):

Table 3-13. IMA TX PCM Control Register 1 (088 - 08B) for T1/J1

Bit	Value	Description
7	0/1	Disable/Enable PCM outputs
6:5	**00**	**Select T1 generic mode**
4	**1**	**Select 2.048 MHz clock**
3	**1**	**Select grouped time slot**
2	0/1	0 to disable the use of the signalling channel (TS 24), 1 to enable
1	**0**	**Tx clock polarity = falling**
0	**0**	**Tx frame pulse polarity = negative**

Table 3-14. IMA TX PCM Control Register 1 (088 - 08B) for E1

Bit	Value	Description
7	0/1	Disable/Enable PCM outputs
6:5	**10**	**Select E1 generic mode**
4:2	000	Not used in E1
1	**0**	**Tx clock polarity = falling**

Table 3-14. IMA TX PCM Control Register 1 (088 - 08B) for E1 (cont)

Bit	Value	Description
0	**0**	**Tx frame pulse polarity = negative**

IMA TX PCM Control Register 2 (080 - 083):

Table 3-15. IMA TX PCM Control Register 2 (080 - 083)

Bit	Value	Description
7:5	000	Unused
4	**1**	**TXCK and TXSYNC: inputs**
3:0	**link**	**TXCK source for PCM link:** **link=0000 (RXCK0) for PCM link 0** **link=0001 (RXCK1) for PCM link 1** **link=0010 (RXCK2) for PCM link 2** **link=0011 (RXCK3) for PCM link 3**

IMA RX PCM Control Register 1 (090 - 093):

Table 3-16. IMA RX PCM Control Register (090 - 093) for T1/J1

Bit	Value	Description
7	0/1	Disconnect/Connect the PCM to the cell delineation block
6	**0**	**Select T1 format**
5	**0**	**Select Generic PCM mode**
4	**1**	**Select a 2.048 MHz clock**
3	**1**	**Select grouped time slots**
2	0/1	Disable/Enable the use of the signalling channel (TS 24)
1	**0**	**Sample on rising edge**
0	**0**	**Frame synchronize on negative pulse**

Table 3-17. IMA RX PCM Control Register (090 - 093) for E1

Bit	Value	Description
7	0/1	Disconnect/Connect the PCM to the cell delineation block
6	**1**	**Select E1 format**
5	**0**	**Select Generic PCM mode**
4	**0**	**Significant in T1 only**
3	**0**	**Significant in T1 only**
2	**0**	**Significant in T1 only**

Table 3-17. IMA RX PCM Control Register (090 - 093) for E1 (cont)

Bit	Value	Description
1	**0**	**Sample on rising edge**
0	**0**	**Frame synchronize on negative pulse**

The Ethernet Port Initialization

The Ethernet Line Interface Unit (LIU) is a Level One LXT971.

The LIU is connected to FCC3 through a Media Independent Interface (MII). The LIU internal registers are initialized through MDC and MDIO Management pins. These pins have to be manually manipulated through PC(23) and PC(24) pins. The LIU PHY address is set to 0 (address pins are cabled to 0V).
See Boot Firmware: eth\c\lxtinit.c.

The TTY Framer Initialization

SMC1 baud-rate generator is BRG1 - see *BRG1 – TTY Baud-Rate Generator* on page 87.

For a simple SMC1 controller example in polling mode:
See Boot Firmware: app\c\montty.c

Accessing the 4539 on the PCI Side

4

PowerSpan Configuration by the PCI Host

Several elements of the PowerSpan are automatically configured at power-up by the hardware, or by the PowerQUICC II. However, some PCI-specific settings have to be done by the PCI host.

PCI Configuration

The card is identified through its Interphase Vendor ID (0x107E) and its PCI device ID (0x90A0). Its PCI configuration is set up by the PCI host at its power-on or by the “high availability” operating system if the 4539 has been hot inserted.

Interrupt Pin Configuration

The set up of the PowerSpan Interrupt Map registers is normally done by the PowerQUICC II when it boots, so they should not need to be reconfigured, except if the card has not yet received a valid boot firmware.

PCI-to-Local Window Configuration

When accessing through a PowerSpan PCI-to-local window, this window must have been enabled in the I^2C serial EEPROM, in order to allow the CompactPCI host to detect it at system power-on or after hot insertion of the board, and map it in the PCI space.

The corresponding PowerSpan register PCI Target Image Control Register must also have been initialized with the Image Enable bit set (IMG_EN=1) and the address translation mechanism enabled (TA_EN=1).

Hot Swap Management

The CompactPCI Hot Swap Specification defines all the processes for installing and removing communications controller boards while the system is running, without adversely affecting other elements in the system. Two different levels of Hot Swap support are defined for an communications controller: Basic Hot Swap support and Full Hot Swap support. The 4539 supports the Full Hot Swap level.

Hot extraction and hot insertion events are generally managed by the host operating system, which also performs the task of removing or installing the card in its PCI spaces and notifying the event to the relevant card driver.

Controlling 4539 Hardware and Software Resets

PowerSpan interrupt pins –INT2 and –INT3 are used as output ports to control the MPC8260 hardware reset signal –HRESET and software reset signal –SRESET respectively. They are conventionally associated with doorbell bits 4 and 5 respectively. The PowerSpan Interrupt Map registers must have been correctly initialized before (see *PowerSpan Configuration by the PCI Host* on page 107).

During a power-up sequence, –HRESET and –SRESET are first activated and then deactivated once the PCI bus reset signal is deactivated. This allows the PowerQUICC II to boot without any host intervention, just after the end of the PCI reset.

For normal use, the card should be reset by the PCI host (if needed) using only the –SRESET signal. The –HRESET signal is used for special cases, such as FLASH memory re-programming through PCI. Example 4-1 is an example of C code routines to reset and run the board from the PCI side.

Example 4-1. Reset and Run Command Routines

```
#define T_HRESET        0x1000// DB4 controls -INT2 as output for -HRESET
#define T_SRESET        0x2000// DB5 controls -INT3 as output for -SRESET

void ResetCard( void)
{
    RegWrite32( T_IER0, RegRead32( T_IER0) | T_SRESET);
}

void RunCard( void)
{
    RegWrite32( T_ISR0, T_HRESET | T_SRESET);
}
```

Controlling the PCI-to-Local Interrupt

PowerSpan Interrupt pin –INT0 is used to control the PCI-to-Local interrupt (renamed ATN in the software examples: “Attention to the PowerQUICC II”). It is associated by convention with doorbell register 2. The PowerSpan Interrupt Map registers must have been previously correctly initialized. This interrupt controls the –IRQ1/DP1/–EXT_BG2 input pin of the PowerQUICC II.

Example 4-2 is an example of C code routines to set and reset the PCI-to-Local interrupt and to read the status of this interrupt from the PCI side.

Example 4-2. PCI to Local Interrupt Routines (From the PCI Side)

```
#define T_ATN            0x400    // DB2 controls -INT0 as output for -ATN
void SetAtn( void)
{
```

```
    RegWrite32( T_IER0, T_ATN);
}
unsigned char AtnState( void)
{
    return( (unsigned char) ((RegRead32( T_ISR0) & T_ATN)? 1 : 0) );
}
```

Local to PCI Interrupt (–INTA)

The PowerQUICC II can generate an interrupt toward the PCI Host by setting a doorbell bit. Conventionally, doorbell bit 0 has been dedicated to this task, and has been associated with PCI interrupt pin –INTA in the PowerSpan Interrupt Map registers.

Example 4-3 is an example of C code routines to reset the PCI-to-Local interrupt and to read the status of this interrupt from the local side.

Example 4-3. Routines Related to Local-to-PCI Interrupt

```
#define T_INTA 0x100    // DB0 controls -INTA PCI interrupt
void ResetIntA( void)
{
    RegWrite32( T_IER0, RegRead32( T_IER0) | T_INTA);
}
unsigned char PciIntState( void)
{
    return( (unsigned char) ((RegRead32( T_ISR0) & T_INTA)? 1 : 0) );
}
```

Local Space Access From PCI Memory Space

The PowerSpan provides four memory windows from the PCI memory space to the Local memory space. In the 4539 design, the default setting in the PowerSpan serial EEPROM enables two windows. The first one is set with a size of 2 MB and is intended for “operational” exchanges. The second one is set with a size of 512 KB and is intended to be used for “dumps”.

During a PCI host access to local space, the high-order address bits of the local bus must be generated by the PowerSpan (as defined in the PowerSpan P1_TI0_ADDR register), the low-order address bits of the local bus come from the PCI address. This mode is called “Address Translation” in the PowerSpan manual.

NOTE

When accessing through a PowerSpan PCI-to-local window, this window must have been enabled in the I^{2}C serial EEPROM, in order to allow the CompactPCI host to detect it at system power-on or after hot insertion of the board, and map it in the PCI space. The corresponding PowerSpan register “PCI Target Image Control Register” must also have been initialized with the “Image Enable” bit set (IMG_EN=1) and the address translation mechanism enabled (TA_EN=1).

When the processor is running, the PCI bus has access to all the elements connected to the local bus, except the FLASH boot memory: the main SDRAM memory (the processor’s SDRAM memory controller must be initialized), the Connection SDRAM memory (the processor’s SDRAM memory controller must be initialized), the CAM memory, the QuadFALC framers, the H110 switch, the IMA device, etc. (the processor must have its chip selects programmed). Local space mapping is the same as when accessed by the processor.

It is not possible to have access to the entire FLASH device when the processor is running, because the FLASH device is an 8-bit data bus device connected to the 64-bit-only local bus of the PowerSpan. Only bytes modulo 8 are reachable.

This problem has been neutralized for the other non-64-bit peripherals, by tying their peripheral address bits 0 to N to the local address bits 3 to N+3 respectively, so that all their registers can be accessed on byte lane 0, at consecutive modulo 8 addresses.

When the processor is in the reset state, its memory controllers and chip-select signals are reset, so nothing can be accessed.

Access to the FLASH EEPROM Through the PCI Bus

For FLASH in-situ re-programming by the PCI host, a special FLASH mode provides access. In this mode, the PowerQUICC II is reset and logic generates a FLASH chip-select and overcomes the problem of an 8-bit device connected to a 64-bit-only PowerSpan.

The specific FLASH mode is enabled by one of the PowerSpan interrupt pins (–INT1) used as an output port. When –INT1 is set to 0, the PowerQUICC II is maintained in hard reset state (–HRESET=0), its pins are tri-stated, the 60x bus is parked for the PowerSpan, and the following address bus remap is implemented: the device’s low order address bit A (2:0) is driven by PowerSpan address bit A (24:22). This remap allows full access of the FLASH content through byte lane 0 of the 64-bit 60x bus, provided that some address translation is done by the software.

Example 4-4. Set and Reset FLASH Mode Routine (From PCI Side)

```
#define T_FLASH_EN      0x800   // DB3 controls -INT1 as output for -FLASH_EN
#define T_HRESET        0x1000   // DB4 controls -INT2 as output for -HRESET
```

```
#define T_SRESET        0x2000    // DB5 controls -INT3 as output for -SRESET

void SetFLASHMode( void)
{
   RegWrite32( T_IER0, RegRead32( T_IER0) | T_SRESET | T_HRESET); // Reset
   RegWrite32( T_IER0, RegRead32( T_IER0) | T_FLASH_EN); // then FLASH mode
}

void ResetFLASHMode( void)
{
   RegWrite32( T_ISR0, T_FLASH_EN); // Reset FLASH mode
   RegWrite32( T_ISR0, T_HRESET | T_SRESET); // then run
}
```

Example 4-5. FLASH Read and Write Routines (From PCI Side)

```
unsigned char FLASHRead( unsigned long addr)
{
   unsigned char data, *ptr;

   /*----- Move the Memory window over the FLASH area -----*/
   RegWrite32( T_P1_TI0_ADDR, FLASH.BaseAddr);

   /*----- Put address bit A(2:0) at address bits A(24:22) -----*/
   addr = ((addr & 7)<<22) | (addr & 0x3FFFF8);

   /*----- Do the read -------*/
   ptr  = (unsigned char*)(WinBase+addr);
   data = *ptr;

   /*----- Move back Memory window at 0 -----*/
   RegWrite32( T_P1_TI0_ADDR, 0);
   return data;
}

void FLASHWrite( unsigned long addr, unsigned char data)
{
   unsigned char* ptr;

   /*----- Move the Memory window over the FLASH area -----*/
   RegWrite32( T_P1_TI0_ADDR, FLASH.BaseAddr);
```

```
    /*----- Put address bit A(2:0) at address bits A(24:22) -----*/
    addr = ((addr & 7)<<22) | (addr & 0x3FFFF8);

    /*----- Do the write -------*/
    ptr  = (unsigned char*)(WinBase+addr);
    *ptr = data;

    /*----- Move back Memory window at 0 -----*/
    RegWrite32( T_P1_TI0_ADDR, 0);
}
```

FLASH EEPROM Programming Algorithms

The boot memory is a 4Mx8 AMD 29LV033 FLASH device. To reprogram the AMD FLASH device, special programming algorithms are defined by AMD, which combine reads and writes with special address patterns. The algorithm descriptions can be found on the AMD web site. You can also look or start from the source provided in the BDK (file `app\c\amdflash.c`).

Serial EEPROM Connected to the PowerSpan

An I²C serial EEPROM is connected to the PowerSpan. It is used to store certain PowerSpan register initialization values and the PCI Vital Product Data (VPD). Other Interphase-specific data is stored there, and there is still some room for other custom data. See *Serial EEPROM Connected to the PowerSpan* on page 32.

Table 2-1 on page 72 provides the PowerSpan Register initialization values stored in the Serial EEPROM.

The I²C Serial EEPROM can be easily accessed from the PCI side or from the local processor side, by using dedicated PowerSpan Register I2C_CSR.

Example 4-6 is an example of C code read and write routines.

Example 4-6. I²C Serial EEPROM Read and Write Routines (From PCI Side)

```
unsigned char EepromByteRead( unsigned char a)
{
    unsigned long v;
    v = ((unsigned long) a)<<24 | 0xA000;
    while( RegRead32( T_I2C_CSR) & 0x80);  // Wait ACT=0
    RegWrite32( T_I2C_CSR, v);
    while( (v=RegRead32( T_I2C_CSR) & 0x80));  // Wait ACT=0
    if (v&0x40) printf("error");
    v>>=16;
    return (unsigned char) v;
```

```
}

void EepromByteWrite( unsigned char a, unsigned char d)
{
    unsigned long v,s;
    v = ((unsigned long) a)<<24 |((unsigned long) d)<<16 | 0xA100;
    while((s=RegRead32( T_I2C_CSR)) & 0x80);  // Wait ACT=0
    if (s&0x40) printf("error");
    RegWrite32( T_I2C_CSR, v);
}
```

In-Situ EPLD Programming

Glue logic is implemented in some EPLDs that can be programmed in the field using the PCI interface.

The EPLDs are in a daisy-chain configuration, which enables all of them to be programmed at once. They can be programmed in-situ by the PCI host, using PowerSpan interrupts as I/O pins. A jumper must be placed on board location JP1 to enable the programming (when present, this jumper sets the ISP signal –ISPEN to its active state 0).

These devices are initialized by Interphase and keep their programming during power off. The normal user should not need to reprogram them.

Optimizing the PCI Bus Utilization

The PCI maximum throughput of 266 MB/s is very difficult to reach. The actual throughput can be very disappointing if certain principles are not followed. These principles are:

Avoid the reads. Prefer the writes. Writes can be very efficient, because they are posted in the FIFOs included in the various PCI bridges. A read needs completion of the entire data transfer from its origin to its destination, before being considered as finished. Because of the arbitrations on the various local busses and because of resynchronizations occurring each time there are different bus clocks, a single read can take approximately 1 µs.

Prefer the bursts. During a burst, the duration of the transfers after the first one can be very efficient and last only one PCI cycle. On the 4539, only the PowerSpan DMAs can generate efficient bursts, because they do transfers to incremental addresses.

Prefer DMA transfers. For data transfers between the PCI space and the local 60x memory, the PowerSpan DMAs are more efficient than the local processor. They can use bursts on both the local 60x side and on the PCI side. They use FIFOs to de-couple the PCI bandwidth and the 60x bandwidth occupancies.

Effective Ordering of the PCI Accesses

The PowerSpan includes FIFOs between the PCI bus and the 60x bus in each direction. When a write is done by the PCI host into the local memory, the PowerSpan can acknowledge this write as soon as there is a place in the FIFO, but the effective write into the local memory can be delayed, due to previous writes still waiting in the FIFO, or due to the local 60x bus being used by the processor.

If the PCI Host makes an access to a PowerSpan register, just after this write to the local memory, the effective completion of this register access may occur before the effective write into the local memory. This can lead to unexpected behavior.

The order in which the PCI makes successive writes and reads into the local memory may also not be respected on the local side. Suppose that the host makes several writes followed by one read. Because the FIFO in the write direction may take some time to get emptied on the local side, the effective read on the local side may happen before the last write.

Example: The PCI host sets the DMA buffer descriptors into the local memory, and then it runs the DMA (a write into a PowerSpan register). The DMA starts before the effective completion of the buffer descriptors writes into the local 60x memory, so it loads a bad addresses, a bad byte count, etc., and accomplishes the transfer with this bad data.

In this case, the PCI host must ensure that its latest write into the memory is effectively finished locally, before starting the DMA.

PCI Deadlock Situations

Several deadlock situations can occur on the PCI bus. These situations will statistically rarely occur, but they need to be treated by exception handlers, in order not to lock the system.

Connectors and Front Panel 5

Connector Placement

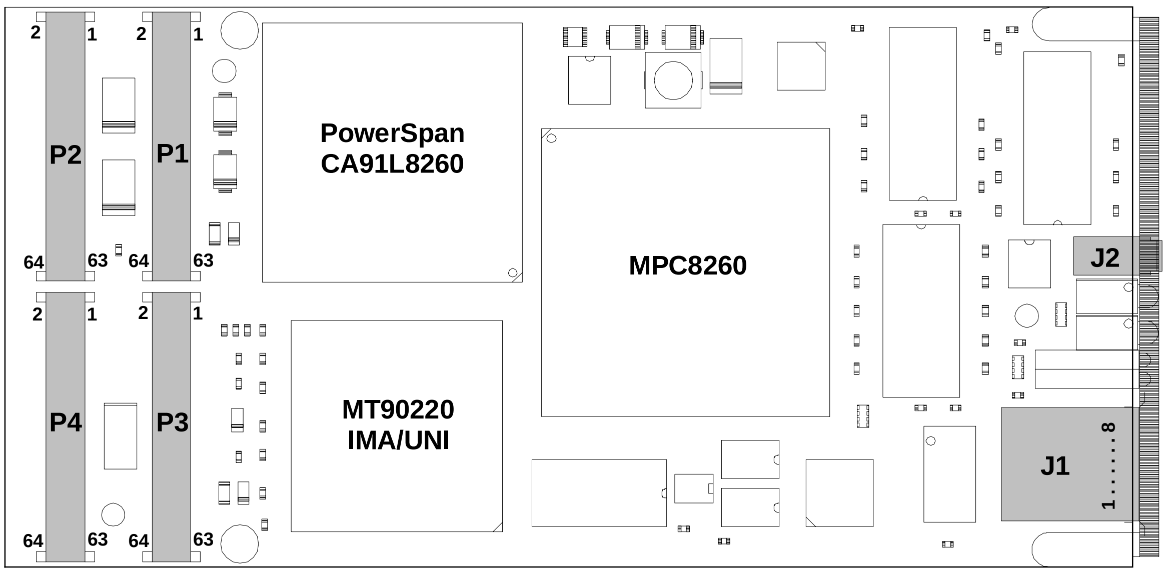


Figure 5-1. Connectors on Component Side

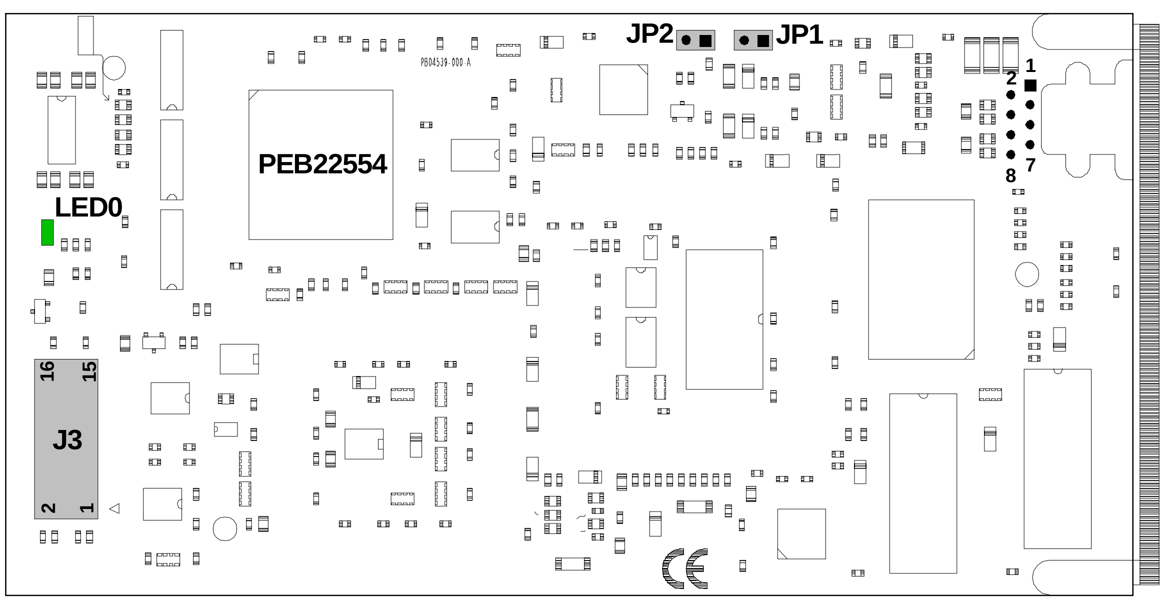


Figure 5-2. Connectors and LED on Solder Side

Front Panel

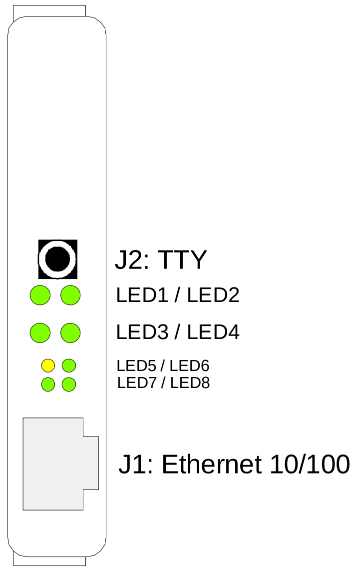


Figure 5-3. Connectors and LEDs on Front Panel

LED Descriptions

LED0: User-programmable green LED controlled by PD(25) (on the board)

LED1: Bi-color red-green LED dedicated to 1st E1/T1/J1 line

LED2: Bi-color red-green LED dedicated to 2nd E1/T1/J1 line

LED3: Bi-color red-green LED dedicated to 3rd E1/T1/J1 line

LED4: Bi-color red-green LED dedicated to 4th E1/T1/J1 line

LED5: User-programmable yellow LED controlled by PD(24)

LED6: Ethernet green LED controlled by LXT971A LED driver 3

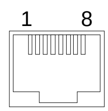
LED7: Ethernet green LED controlled by LXT971A LED driver 1

LED8: Ethernet green LED controlled by LXT971A LED driver 2

Ethernet 10/100 RJ45 Connector J1

On the front panel there is one shielded 8-pin modular jack connector for the Ethernet 10/100 interface (Twisted Pairs). The Ethernet signals are isolated from the board through a 1500V transformer.

Table 5-1. Ethernet 10/100 RJ45 Connector

	**Signal**
1	OUT+
2	OUT-
3	IN+
4	N/U
5	N/U
6	IN-
7	N/U
8	N/U

TTY Serial Port J2

A 2.5mm stereo jack connector provides a connection to an asynchronous serial device such as a TTY console. Signals on this connector have EIA-232-D electrical levels (RS232) for direct connection to a console.

Table 5-2. J2 RS232 TTY Connector

Pin	Signal
Tip	TxD
Ring	Ground
Sleeve	RxD

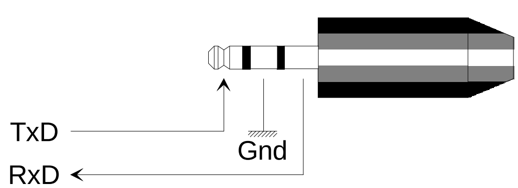


Figure 5-4. TTY Connector: 2.5 mm Stereo Jack Plug

PowerQUICC II Debug Port J3

On the 4539, a 2x8-pin connector can be provided to provide access to the BDM (Background Debug Mode) bus: the PowerQUICC II debug bus. Signals on this connector have 3.3V TTL electrical levels. Pin 1 is identified by a triangle on the PCB.

Table 5-3. J3 Debug Port

Pin	Signal	Signal	Pin
1	TDO	10 kΩ Pull-up to +3.3 V	2
3	TDI	TRST_N	4
5	QREQ_N	+3.3V through a 1 kΩ resistor	6
7	TCK		8
9	TMS		10
11	SRESET_N	GND	12
13	HRESET_N		14
15	10 kΩ Pull-up to +3.3 V	GND	16

ISP Enable Jumper JP1

The 4539 includes a location for a jumper at JP1. This jumper is used during production to enable the programming of the card’s EPLD programmable devices “in-situ”. This jumper should never be used by the normal user.

Blank Card Jumper JP2

Location for a jumper at JP2 is needed for production when the PowerSPAN serial EEPROM is not yet programmed, in order to prevent the card from locking the system. This jumper should never be installed by the normal user.

PMC Connectors P1 and P2

PMC connectors P1 and P2 support the 32-bit PCI bus as defined by the PMC standard. Signal levels are classified in the “Very Low Voltage Directory” by IEC 950 safety standard

Table 5-4. PMC Connector P1

Pin No.	Pin Name	Pin Type	Description
1			Not connected
2			Not connected

Table 5-4. PMC Connector P1 (cont)

Pin No.	Pin Name	Pin Type	Description
3	GND	Supply	Ground
4	INTA#	Output open drain	PCI interrupt A: Interrupt from the 4539 to the PCI Host controlled by software by the PowerSpan
5			Not connected
6			Not connected
7	BUSMODE1#	Output	Board Presence Indication: Indicates the presence and the PCI protocol capability of the board in response to the BUSMODE[4:2]# signals
8	+5V	Supply	+5 V Supply: Not used on this board
9			Not connected
10			Not connected
11	GND	Supply	Ground
12			Not connected
13	PCI_CLK	Input	PCI Clock: Clock input for the PCI interface. The frequency should be between 25 MHz and 33 MHz
14	GND	Supply	Ground
15	GND	Supply	Ground
16	GNT#	Input	PCI Grant: Input because an external arbiter is used.
17	REQ#	Output	PCI Bus Request: Output because an external arbiter is used
18	+5V	Supply	+5 V Supply: Not used on this board
19	VIO	Supply	VIO Supply: 3.3 V or 5.0 V
20	PAD(31)	Tristate bidirectional	PCI Address/Data
21	PAD(28)	Tristate bidirectional	PCI Address/Data
22	PAD(27)	Tristate bidirectional	PCI Address/Data
23	PAD(25)	Tristate bidirectional	PCI Address/Data
24	GND	Supply	Ground
25	GND	Supply	Ground
26	CBE3#	Tristate bidirectional	PCI Bus Command and Byte Enable
27	PAD(22)	Tristate bidirectional	PCI Address/Data

Table 5-4. PMC Connector P1 (cont)

Pin No.	Pin Name	Pin Type	Description
28	PAD(21)	Tristate bidirectional	PCI Address/Data
29	PAD(19)	Tristate bidirectional	PCI Address/Data
30	+5V	Supply	+5 V Supply: Not used on this board
31	VIO	Supply	VIO Supply: 3.3 V or 5.0 V
32	PAD(17)	Tristate bidirectional	PCI Address/Data
33	FRAME#	Tristate bidirectional	PCI Cycle Frame
34	GND	Supply	Ground
35	GND	Supply	Ground
36	IRDY#	Tristate bidirectional	PCI Initiator Ready
37	DEVSEL#	Tristate bidirectional	PCI Device Select
38	+5V	Supply	+5V Supply: Not used on this board
39	GND	Supply	Ground
40			Not connected
41			Not connected
42			Not connected
43	PAR	Tristate bidirectional	PCI Parity
44	GND	Supply	Ground
45	VIO	Supply	VIO Supply: 3.3 V or 5.0 V
46	PAD(15)	Tristate bidirectional	PCI Address/Data
47	PAD(12)	Tristate bidirectional	PCI Address/Data
48	PAD(11)	Tristate bidirectional	PCI Address/Data
49	PAD(9)	Tristate bidirectional	PCI Address/Data
50	+5V	Supply	+5 V Supply: Not used on this board
51	GND	Supply	Ground
52	CBE0#	Tristate bidirectional	PCI Bus Command and Byte Enable

Table 5-4. PMC Connector P1 (cont)

Pin No.	Pin Name	Pin Type	Description
53	PAD(6)	Tristate bidirectional	PCI Address/Data
54	PAD(5)	Tristate bidirectional	PCI Address/Data
55	PAD(4)	Tristate bidirectional	PCI Address/Data
56	GND	Supply	Ground
57	VIO	Supply	VIO Supply: 3.3 V or 5.0 V
58	PAD(3)	Tristate bidirectional	PCI Address/Data
59	PAD(2)	Tristate bidirectional	PCI Address/Data
60	PAD(1)	Tristate bidirectional	PCI Address/Data
61	PAD(0)	Tristate bidirectional	PCI Address/Data
62	+5V	Supply	+5 V Supply: Not used on this board.
63	GND	Supply	Ground
64			Not connected

Table 5-5. PMC Connector P2

Pin No.	Pin Name	Pin Type	Description
1			Not connected
2			Not connected
3			Not connected
4	PCITDO	Output	JTAG Test Output: Because the board doesn’t support the IEEE Standard 1149.1 interface, PCITDO and PCITDI pins are hard wired.
5	PCITDI	Input	JTAG Test Input: Because the board doesn’t support the IEEE Standard 1149.1 interface, PCITDO and PCITDI pins are hard wired.
6	GND	Supply	Ground
7	GND	Supply	Ground
8			Not connected
9			Not connected
10			Not connected
11	BUSMODE2#	Input	BUSMODE2# Signal: Used with BUSMODE3# and BUSMODE4# to determine the presence and protocol capability of the board. The result is output on BUSMODE1#.
12	+3.3V	Supply	+3.3 V Supply
13	PCI_RST#	Input	PCI Reset
14	BUSMODE3#	Input	BUSMODE3# Signal: Used with BUSMODE2# and BUSMODE4# to determine the presence and protocol capability of the board. The result is output on BUSMODE1#.
15	+3.3V	Supply	+3.3 V Supply.
16	BUSMODE4#	Input	BUSMODE4# Signal: Used with BUSMODE2# and BUSMODE3# to determine the presence and protocol capability of the board. The result is output on BUSMODE1#.
17			Not connected
18	GND	Supply	Ground
19	PAD(30)	Tristate bidirectional	PCI Address/Data
20	PAD(29)	Tristate bidirectional	PCI Address/Data
21	GND	Supply	Ground

Table 5-5. PMC Connector P2 (cont)

Pin No.	Pin Name	Pin Type	Description
22	PAD(26)	Tristate bidirectional	PCI Address/Data
23	PAD(24)	Tristate bidirectional	PCI Address/Data
24	+3.3V	Supply	+3.3 V Supply
25	IDSEL	Input	PCI Initialization Device Select
26	PAD(23)	Tristate bidirectional	PCI Address/Data
27	+3.3V	Supply	+3.3 V Supply
28	PAD(20)	Tristate bidirectional	PCI Address/Data
29	PAD(18)	Tristate bidirectional	PCI Address/Data
30	GND	Supply	Ground
31	PAD(16)	Tristate bidirectional	PCI Address/Data
32	CBE2#	Tristate bidirectional	PCI Bus Command and Byte Enable
33	GND	Supply	Ground
34			Not connected
35	TRDY#	Tristate bidirectional	PCI Target Ready
36	+3.3V	Supply	+3.3 V Supply
37	GND	Supply	Ground
38	STOP#	Tristate bidirectional	PCI Stop
39	PERR#	Tristate bidirectional	PCI Parity Error
40	GND	Supply	Ground
41	+3.3V	Supply	+3.3 V Supply
42	SERR#	Output open drain	PCI System Error
43	CBE1#	Tristate bidirectional	PCI Bus Command and Byte Enable
44	GND	Supply	Ground
45	PAD(14)	Tristate bidirectional	PCI Address/Data

Table 5-5. PMC Connector P2 (cont)

Pin No.	Pin Name	Pin Type	Description
46	PAD(13)	Tristate bidirectional	PCI Address/Data
47	GND	Supply	Ground
48	PAD(10)	Tristate bidirectional	PCI Address/Data
49	PAD(8)	Tristate bidirectional	PCI Address/Data
50	+3.3V	Supply	+3.3 V Supply
51	PAD(7)	Tristate bidirectional	PCI Address/Data
52			Not connected
53	+3.3V	Supply	+3.3 V Supply
54			Not connected
55			Not connected
56	GND	Supply	Ground
57			Not connected
58			Not connected
59	GND	Supply	Ground
60			Not connected
61			Not connected
62	+3.3V	Supply	+3.3 V Supply
63	GND	Supply	Ground
64			Not connected

PMC Connector P3

P3 and P4 are used in configuration 1 of the PICMG 2.15 “PCI Telecom Mezzanine Card” specification (PTID(2:0)=001).

PMC connector P3 supports the FCC1 UTOPIA bus (see Table 5-6). All input or output signals go through bus switches that isolate them from P3 when PTENB_N is deasserted. All the signals are enabled when PTENB_N is asserted. TxData always flows from the carrier to the board and RxData always flows from the board to the carrier. The other signal directions depend on the slave or master configuration.

Signal levels are classified in the “Very Low Voltage Directory” by IEC 950 safety standard.

Table 5-6. PMC Connector P3

Pin No.	Pin Name	Pin Type	Description
1	TXSOC	Input	Transmit Start of Cell: This is always an input because TXD(7:0) data always flows from the carrier to the board.
2	GND	Supply	Ground
3	GND	Supply	Ground
4			Not connected
5	TXCLAV	Input/Output	Transmit Cell Available: This is an output when the board is ATM PHY. This is an input when the board is ATM Controller.
6			Not connected
7	RXADR(3)	Input/Output	Receive Address 3: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
8	GND	Supply	Ground
9	PTID2	Not connected (Output)	PCI Telecom Identifier 2: PTID[2:0] = 001 allows the board to display its PTMC configuration 1 (not connected equals 0).
10	PTGNDZ3	Output	Ground Return 3: AC ground return for PTMC signals (tristatable).
11	PTGNDZ0	Output	Ground Return 0: AC ground return for PTMC signals.
12	RXREF	Output	Receive Reference Clock: 8 KHz reference clock provided by the QuadFALC FSC output to the carrier card.
13	TXREF	Input	Transmit Reference Clock: 8 KHz or 2.048 MHz clock provided by the carrier card to the QuadFALC SYNC input if selected.
14	GND	Supply	Ground
15	GND	Supply	Ground
16	RXENB_N	Input/Output	Receive Enable: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
17	TXADR(3)	Input/Output	Transmit Address 3: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
18	RXCLAV	Input/Output	Receive Cell Available: This is an output when the board is ATM PHY. This is an input when the board is ATM Controller.

Table 5-6. PMC Connector P3 (cont)

Pin No.	Pin Name	Pin Type	Description
19	TXADR(2)	Input/Output	Transmit Address 2: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
20	GND	Supply	Ground
21	PTID0	Output (VIO) (hard wired)	PCI Telecom Identifier 0: PTID[2:0] = 001 allows the board to advertise its PTMC configuration 1 (VIO equals 1).
22	TXENB_N	Input/Output	Transmit Enable: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
23	PTGNDZ1	Output	Ground Return 1: AC ground return for PTMC signals.
24	RXADR(2)	Input/Output	Receive Address 2: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
25	TXCLK	Input/Output	Transmit Clock: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
26	GND	Supply	Ground
27	GND	Supply	Ground
28	TXADR(1)	Input/Output	Transmit Address 1: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
29	TXADR(0)	Input/Output	Transmit Address 0: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
30	RXADR(1)	Input/Output	Receive Address 1: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
31			Not connected
32	GND	Supply	Ground
33	GND	Supply	Ground
34	RXADR(0)	Input/Output	Receive Address 0: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
35	TXD(7)	Input	Transmit Data 7: Data from the carrier card to the board.
36			Not connected
37	TXD(6)	Input	Transmit Data 6: Data from the carrier card to the board.

Table 5-6. PMC Connector P3 (cont)

Pin No.	Pin Name	Pin Type	Description
38	GND	Supply	Ground.
39	PTENB_N	Input	PTMC Enable: The PTENB_N signal is driven by the carrier card and enables all the signals on P3 when it is asserted at 0. It can be overridden on the board.
40	RXD(7)	Output	Receive Data 7: Data from the board to the carrier.
41	PTGNDZ2	Output	Ground Return 2: AC ground return for PTMC signals.
42	RXD(6)	Output	Receive Data 6: Data from the board to the carrier.
43	RXCLK	Input/Output	Receive Clock: This is an input when the board is ATM PHY. This is an output when the board is ATM Controller.
44	GND	Supply	Ground
45	GND	Supply	Ground
46	RXD(5)	Output	Receive Data 5: Data from the board to the carrier.
47	TXD(5)	Input	Transmit Data 5: Data from the carrier card to the board.
48	RXD(4)	Output	Receive Data 4: Data from the board to the carrier.
49	TXD(4)	Input	Transmit Data 4: Data from the carrier card to the board.
50	GND	Supply	Ground
51	GND	Supply	Ground
52	RXD(3)	Output	Receive Data 3: Data from the board to the carrier.
53	TXD(3)	Input	Transmit Data 3: Data from the carrier card to the board
54	RXD(2)	Output	Receive Data 2: Data from the board to the carrier.
55	TXD(2)	Input	Transmit Data 2: Data from the carrier card to the board
56	GND	Supply	Ground
57	PTID1	Not connected (Output)	PCI Telecom Identifier 1: PTID[2:0] = 001 allows the board to display its PTMC configuration 1 (not connected equals 0).
58	RXD(1)	Output	Receive Data 1: Data from the board to the carrier.
59	TXD(1)	Input	Transmit Data 1: Data from the carrier card to the board.
60	RXD(0)	Output	Receive Data 0: Data from the board to the carrier.
61	TXD(0)	Input	Transmit Data 0: Data from the carrier card to the board.

Table 5-6. PMC Connector P3 (cont)

Pin No.	Pin Name	Pin Type	Description
62	GND	Supply	Ground
63	GND	Supply	Ground
64	RXSOC	Output	Receive Start of Cell: This is always an output because RXD(7:0) data always flow from the board to the carrier.

PMC Connector P4

P3 and P4 are used in configuration 1 of the PICMG 2.15 “PCI Telecom Mezzanine Card” specification (PTID(2:0)=001). In this configuration PMC P4 is “User I/O”: not defined by the standard.

On the 4539, PMC connector P4 supports the four T1/E1/J1 lines, the two TDM busses with clocks and synchronization signals and the four last TDM busses of the IMA/UNI device. Signal levels are classified in the “Very Low Voltage Directory” by IEC 950 safety standard. These signals are not isolated by switches.

The T1/E1/J1 lines numbering starts from 0.

Table 5-7. PMC Connector P4

Pin No.	Pin Name	Pin Type	Description
1	DOH0	Input	Data from External TDM bus 0: In "Switched Mode", 8.192 Mb/s transmit data from P4 to the QuadFALC transmitters.
2	DIH0	Tristate Output	Data to External TDM bus 0: In "Switched Mode", 8.192 Mb/s received data from the QuadFALC and sent to P4. This output is enabled only in "switched mode".
3	DOH1	Input	Data from External TDM bus 1: In "Switched Mode", 8.192 Mb/s data from P4 to the MPC8260 TDMa1 bus.
4	DIH1	Tristate Output	Data to External TDM bus 1: In "Switched Mode", 8.192 Mb/s data from the MPC8260 TDMa1 bus sent to P4. This output is enabled only in "switched mode".
5	FSOH	Input	External 8 KHz frame synchronization pulse: provided by P4 to the MPC8260 and the QuadFALC in "switched mode". Provided to the IMA/UNI device when it is used with P4. FSOH can also be used as a reference clock for the QuadFALC SYNC input.

Table 5-7. PMC Connector P4 (cont)

Pin No.	Pin Name	Pin Type	Description
6	FSIH	Tristate Output	8 KHz pulse: Generated by the QuadFALC and output to the motherboard. This output is enabled only in "switched mode".
7	CKOH	Input	External TDM bus clock: Provided by P4 to the MPC8260 and the QuadFALC. Provided to the IMA/UNI device when it is used with P4.
8	CKIH	Tristate Output	Internal clock: generated by the QuadFALC and output on P4. This output is enabled only in "switched mode".
9	GND	Supply	Ground
10	GND	Supply	Ground
11	IMA_DSTI4	Input	IMA/UNI Serial Data Input 4: This input works with FSOH and CKOH clocks. Available only if an MT90220 is equipped.
12	IMA_DSTO4	Tristate Output	IMA/UNI Serial Data Output 4: This output works with FSOH and CKOH clocks. It can be tristated by the IMA/UNI device. Available only if an MT90220 is equipped.
13	IMA_DSTI5	Input	IMA/UNI Serial Data Input 5: This input works with FSOH and CKOH clocks. Available only if an MT90220 is equipped.
14	IMA_DSTO5	Tristate Output	IMA/UNI Serial Data Output 5: This output works with FSOH and CKOH clocks. It can be tristated by the IMA/UNI device. Available only if an MT90220 is equipped.
15	IMA_DSTI6	Input	IMA/UNI Serial Data Input 6: This input works with FSOH and CKOH clocks. Available only if an MT90220 is equipped.
16	IMA_DSTO6	Tristate Output	IMA/UNI Serial Data Output 6: This output works with FSOH and CKOH clocks. It can be tristated by the IMA/UNI device. Available only if an MT90220 is equipped.
17	IMA_DSTI7	Input	IMA/UNI Serial Data Input 7: This input works with FSOH and CKOH clocks. Available only if an MT90220 is equipped.
18	IMA_DSTO7	Tristate Output	IMA/UNI Serial Data Output 7: This output works with FSOH and CKOH clocks. It can be tristated by the IMA/UNI device. Available only if an MT90220 is equipped.
19			Not connected
20			Not connected

Table 5-7. PMC Connector P4 (cont)

Pin No.	Pin Name	Pin Type	Description
21			Not connected
22			Not connected
23			Not connected
24			Not connected
25			Not connected
26			Not connected
27			Not connected
28			Not connected
29			Not connected
30			Not connected
31			Not connected
32			Not connected
33	RL1_0	Analog Input	Line 0 Analog Input 1: The receive line presents a typical 100 W differential impedance.
34	XL1_0	Analog Output	Line 0 Analog Output 1.
35	RL2_0	Analog Input	Line 0 Analog Input 2: The receive line presents a typical 100 W differential impedance.
36	XL2_0	Analog Output	Line 0 Analog Output 2.
37	RL1_1	Analog Input	Line 1 Analog Input 1: The receive line presents a typical 100 W differential impedance.
38	XL1_1	Analog Output	Line 1 Analog Output 1.
39	RL2_1	Analog Input	Line 1 Analog Input 2: The receive line presents a typical 100 W differential impedance.
40	XL2_1	Analog Output	Line 1 Analog Output 2.
41	RL1_2	Analog Input	Line 2 Analog Input 1: The receive line presents a typical 100 W differential impedance.
42	XL1_2	Analog Output	Line 2 Analog Output 1.
43	RL2_2	Analog Input	Line 2 Analog Input 2: The receive line presents a typical 100 W differential impedance.
44	XL2_2	Analog Output	Line 2 Analog Output 2.
45	RL1_3	Analog Input	Line 3 Analog Input 1: The receive line presents a typical 100 W differential impedance.
46	XL1_3	Analog Output	Line 3 Analog Output 1.
47	RL2_3	Analog Input	Line 3 Analog Input 2: The receive line presents a typical 100 W differential impedance.
48	XL2_3	Analog Output	Line 3 Analog Output 2.

Table 5-7. PMC Connector P4 (cont)

Pin No.	Pin Name	Pin Type	Description
49			Not connected
50			Not connected
51			Not connected
52			Not connected
53			Not connected
54			Not connected
55			Not connected
56			Not connected
57			Not connected
58			Not connected
59			Not connected
60			Not connected
61			Not connected
62			Not connected
63			Not connected
64			Not connected

Operating environment

6

Recommended Operating Conditions

Table 6-1. Recommended Operating Conditions

Rating	Value		Unit
	Min	Max	
5 V Supply	not used		Volts
3.3 V Supply	3.135	3.465	Volts
Ambient Temperature	0	55	°C
	32	131	°F

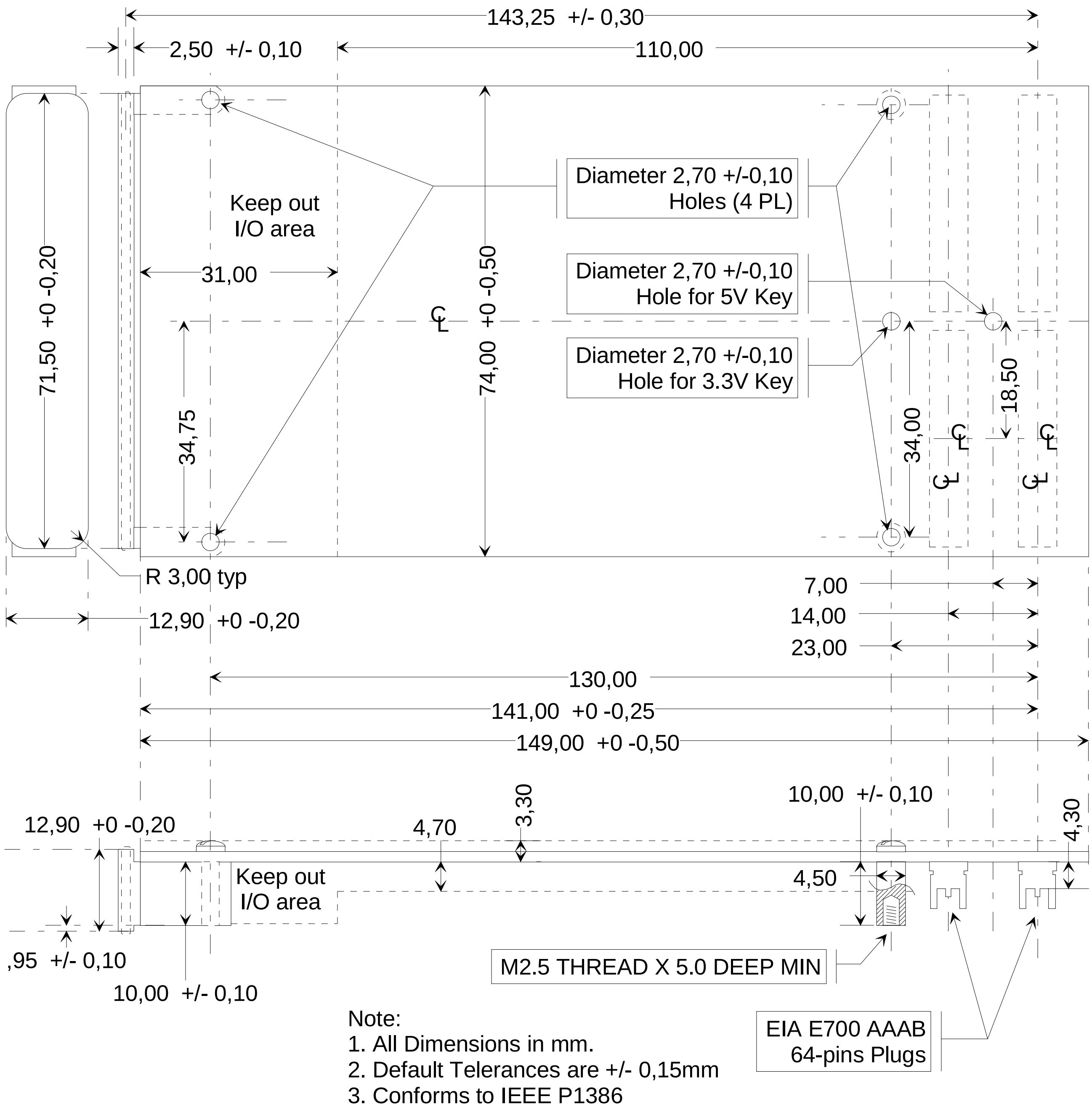
Operating Characteristics

Table 6-2. Operating Characteristics

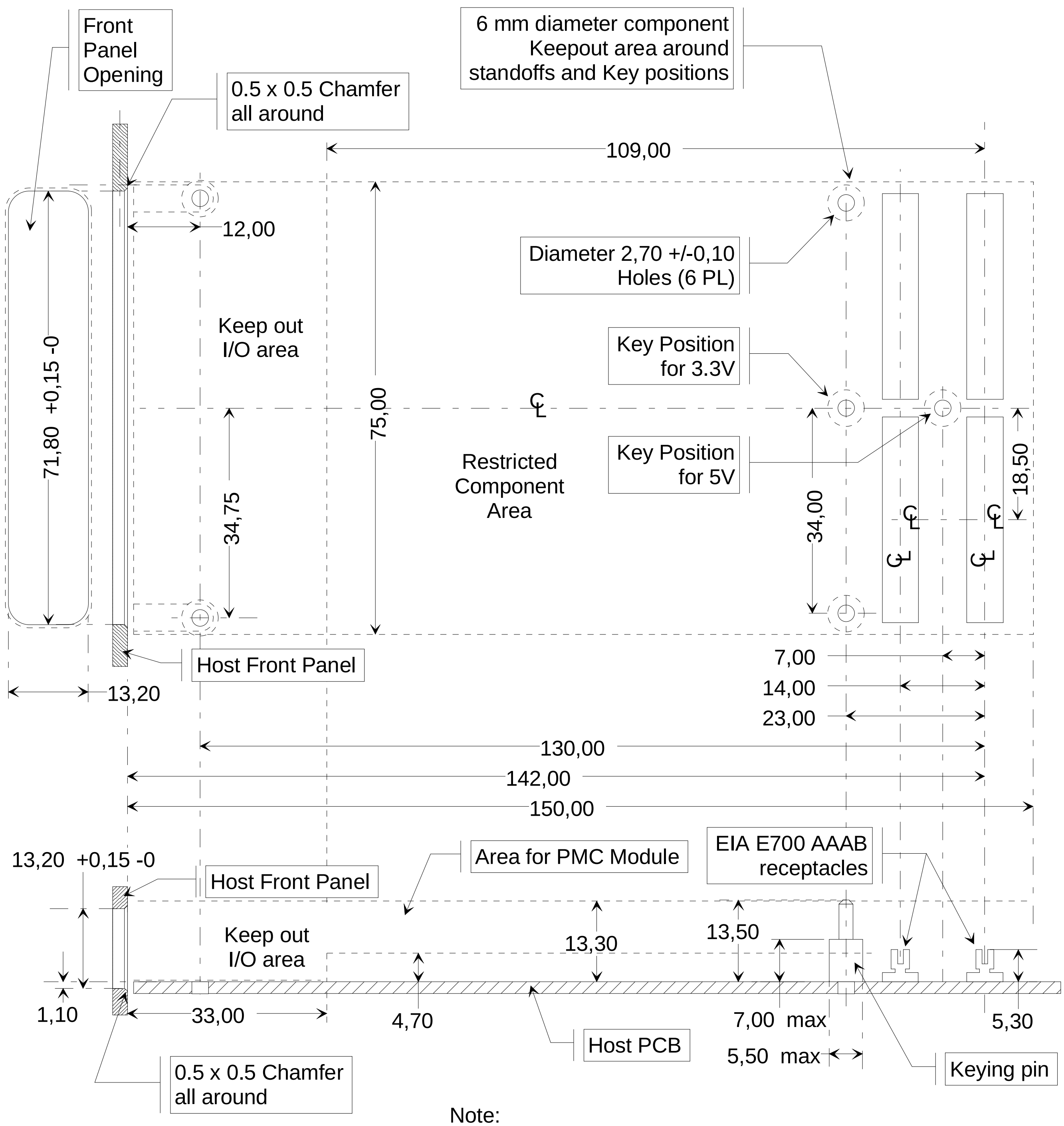
Rating	Value	Unit
5 V Operating Current	0	Amperes
3.3 V Operating Current	2.3	Amperes
Power Dissipation	8	Watts

Mechanical Information A

PMC Card Dimensions



Carrier Card Dimension Requirements



Note:
1. All Dimensions in mm.
2. Default Telerances are +/- 0,15mm
3. Conforms to IEEE P1386

Bibliography B

Industry Standards

EIA-232-D: Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange

Electronic Industries Alliance
2500 Wilson Boulevard
Arlington, VA 22201-3834
Telephone: (703) 907-7500
Web: http://www.eia.org

ECTF H.110 Hardware Compatibility Specification: CT Bus, Revision 1.0

Enterprise Computer Telephony Forum
39355 California Street, Suite 307
Fremont CA 94538
Telephone: (510) 608-5915
Fax: (510) 608-5917
Web: http://www.ectf.org

IEEE Std 802.3, 2000, IEEE Standards for Local and Metropolitan Area Networks: Media Access Control (MAC) Parameters, Physical Layer, Medium Attachment Units, and Repeater for 100 Mb/s Operation, Type 100BASE-T

Institute of Electrical and Electronics Engineers, Inc.,
445 Hoes Lane, PO Box 1331
Piscataway, NJ 08855-1331
Telephone: (732) 981-0060
Web: http://www.ieee.com

PCI-SIG Peripheral Component Interconnect (PCI) Local Bus Specification, Revision 2.2
PCI Special Interest Group
5200 N.E. Elam Young Parkway,
Hillsboro, OR 97124-6497
Telephone: 1-800-433-5177 or (503) 693-6232
FAX: (503) 693-8344
Web: http://www.pcisig.com

PICMG 2.0 CompactPCI Specification
PICMG 2.5 CompactPCI Computer Telephony Specification
PICMG 2.1 CompactPCI Hot Swap Specification

PCI Industrial Computer Manufacturers Group
401 Edgewater Place, Suite 500

Wakefield, MA 01880
Telephone: (781) 246-9318
Web: http:/www.picmg.org

PowerPC TM Microprocessor Common Hardware Reference Platform:
System Architecture (CHRP), Version 1.0
Literature Distribution Center for Motorola
Telephone: 1-800-441-2447
FAX: (602) 994-6430 or (303) 675-2150
E-mail: ldcformotorola@hibbertco.com
Publication reference: TB338/D

or
IBM
1580 Route 52, Bldg. 504
Hopewell Junction, NY 12533-6531
Telephone: 1-800-PowerPC
Publication reference: MPRP-CHRP-01

or
Morgan Kaufmann Publishers, Inc.
340 Pine Street, Sixth Floor
San Francisco, CA 94104-3205, USA
Telephone: (415) 392-2665 FAX: (415) 982-2665
Publication reference: ISBN 1-55860-394-8

PowerPC Reference Platform (PRP) Specification,
Third Edition, Version 1.0, Volumes I and II
International Business Machines Corporation
Power Personal Systems Architecture
11400 Burnet Rd.
Austin, TX 78758-3493
Document/Specification Ordering
Telephone: 1-800-PowerPC or 1-800-769-3772 or (708) 296-9332
Publication reference: MPR-PPC-RPU-02

Telecommunication Standards

ANSI T1.403-1995: Network-to-customer Installation - DS1 Metallic interface
ANSI T1.107-1995: Digital Hierarchy - Formats Specifications.
ANSI T1.102-1993: Digital Hierarchy - Electrical Interfaces.
ANSI T1.627-1993: Broadband ISDN - ATM Layer Functionality and Specification.
ANSI T1.107-1995: Digital Hierarchy - Formats Specifications.
ANSI T1.646-1995: Broadband ISDN - Physical Layer Specification for User-Network Interfaces Including DS1/ATM.

American National Standard for Telecommunications
ANSI American National Standard Institute
11 West 42nd Street
New York, New York 10036

T1 committee Web: http://www.t1.org

ATM Forum - ATM User-Network Interface Specification, V3.1, October, 1995.
ATM Forum - UTOPIA, An ATM PHY Interface Specification, Level 2, Version 1 June, 1995.

ATM forum
2570 West El Camino Real, Suite 304
Mountain View, CA 94040-1313
Telephone (650) 949-6700
Fax: (650) 949-6705
http://www.atmforum.com

AT&T TR54016: Requirements for Interfacing Digital Terminal Equipment to Services Employing the Extended Superframe Format, Sept 89
AT&T TR62411: ACCUNET® T1.5 Service Description and Interface Specification, Dec. 90

AT&T Customer Information CenterHoward Press
1026 West Elizabeth Avenue
Linden, NJ 07036
Telephone: (888) 387-8852 (US order), (908) 523-2257 (international order)
Fax: (908)-862-6722
Web: http://www.att.com/cpetesting/trs.html

Bell Communications Research, TA-TSY-000773 - “Local Access System Generic Requirements, Objectives, and Interface in Support of Switched Multi-megabit Data Service” Issue 2, March 1990 and Supplement 1, December 1990.

Telcordia
Morris Corporate Center
tel : (973) 829-2000
Morristown, NJ
Web : http://www.telcordia.com

ETSI TBR 012 Ed1 (1993-12) Business Telecommunications (BT); Open Network Provision (ONP) technical requirements; 2 048 kbit/s digital unstructured leased line (D2048U) Attachment requirements for terminal equipment interface.
ETSI TBR 013 (1996-01) Business TeleCommunications (BTC); 2 048 kbit/s digital structured leased lines (D2048S); Attachment requirements for terminal equipment interface.
ETSI ETS 300 269 Draft Standard T/NA(91)17 - Metropolitan Area Network Physical Layer Convergence Procedure for 2.048 Mbit/s”, April 1994.
ETSI ETS 300 011 - Integrated Services Digital Network (ISDN); Primary rate user-network interface Layer 1 specification and test principles.
ETSI ETS 300 166 - Transmission and Multiplexing (TM); Physical and electrical characteristics of hierarchical digital interfaces for equipment using the 2 048 kbit/s - based

plesiochronous or synchronous digital hierarchies.
ETSI ETS 300 233 - Integrated Services Digital Network (ISDN); Access digital section for ISDN primary rate.

European Telecommunications Standards Institute (ETSI)
Sophia Antipolis
http://www.etsi.org/

ITU-T G.703 - Physical/Electrical Characteristics of Hierarchical Digital Interfaces, 1991 (includes 2.048Mbps E1 and 1.56Mbps T1 definitions)
ITU-T G.704 - Terminal Equipment Synchronous Frame Structures Used At 1544, 6312, 2048, 8488 and 44 736 kbit/s Hierarchical Levels", July, 1995.
ITU-T I.431 - Primary rate user-network interface - Layer 1 specification, 1993
ITU-T I.432 - B-ISDN User-Network Interface - Physical Layer Specification, 1993
ITU-T G.804 - ATM Cell Mapping into Plesiochronous Digital Hierarchy (PDH), 1993.
ITU-T G.832 - Transport of SDH Elements on PDH Networks: Frame and Multiplexing Structures", 1993.
ITU-T Q.921 - ISDN User-Network Interface - Data Link Layer Specification, March, 1993.
Other ITU-T references: G.705, G.706, G.732, G.735, G.736, G.737, G.738, G.739, G.751, G775, G.823, G.824, O.151

International Telecommunications Union (ITU)
Place des Nations
CH-1211 Geneva 20
Switzerland
Telephone: +41-22-730-5111
Web: http://www.itu.int/ITU-T

Japanese references: JT- G.703, JT G.704, JT G.706

Manufacturers' Documents

PowerQUICC II Literature
MPC8260 PowerQUICC II User's Manual
Literature Distribution Center for Motorola
Telephone: 1-800-441-2447
FAX: (602) 994-6430 or (303) 675-2150
E-mail: ldcformotorola@hibbertco.com
http://www.mot-sps.com/

Tundra PowerSpan PCI Bridge
PowerSpan™
(CA91L8200/CA91L8260)
PowerPC/PCI Bridge
User Manual
80A1000_MA001_08
http://www.tundra.com

Infineon PEB22554 / QuadFALC T1/E1/J1 framer
Quad Framing and Line Interface Component for E1 / T1 / J1
QuadFALC TM
PEB 22554 Version 1.3
Data Sheet 07.2000
Infineon Technologies AG,
P.O. Box 800949,
81609 Munich, Germany
Web: http://www.infineon.com/products/commics/QuadFALC.htm

Mitel MT90220 Octal IMA/UNI PHY Device
Mitel MT90220 Octal IMA/UNI PHY Device
DS5036 ISSUE 4 December 1999
Web: http://www.mitelsemi.com

Lucent T8105 H.110 TDM bus switch
T8100A, T8102, T8105 Advance Data sheet
November 1999
Web: http://www.lucent.com/micro/dti/cti/ctidoc.cgi

Motorola Content Addressable Memory (CAM) MCM69C232
Data Sheet MCM69C232/D Rev-5
Web: http://mot-sps.com/products/ (select memory, then CAMs)

Level One LXT971 - 3.3V Dual-Speed Fast Ethernet Transceiver
Level One
Web: http://www.level1.com

Glossary

AAL u **ATM Adaptation Layer** Service-dependent sublayer of the data link layer. The AAL accepts data from different applications and presents it to the *ATM* layer in the form of 48-byte ATM payload segments. AALs consist of two sublayers: *CS* and *SAR*. AALs differ on the basis of the source-destination timing used, whether they use *CBR* or *VBR*, and whether they are used for connection-oriented or connectionless mode data transfer. At present, the four types of AAL recommended by the ITU-T are *AAL1*, *AAL2*, *AAL3/4*, and *AAL5*.

AAL1 u **ATM Adaptation Layer 1**One of four AALs recommended by the ITU-T. AAL1 is used for connection-oriented, delay-sensitive services requiring constant bit rates, such as uncompressed video and other isochronous traffic.

AAL2 u **ATM Adaptation Layer 2** One of four AALs recommended by the ITU-T. AAL2 is used for connection-oriented services that support a variable bit rate, such as some isochronous video and voice traffic.

AAL3/4 u **ATM Adaptation Layer 3/4** One of four AALs (merged from two initially distinct adaptation layers) recommended by the ITU-T. AAL3/4 supports both connectionless and connection-oriented links, but is primarily used for the transmission of *SMDS* packets over *ATM* networks.

AAL5 u **ATM Adaptation Layer 5** One of four AALs recommended by the ITU-T. AAL5 supports connection-oriented *VBR* services and is used predominantly for the transfer of classical *IP* over *ATM* and *LANE* traffic. AAL5 uses *SEAL* and is the least complex of the current AAL recommendations. It offers low bandwidth overhead and simpler processing requirements in exchange for reduced bandwidth capacity and error-recovery capability.

AIN u **Advanced Intelligent Network** In *SS7*, an expanded set of network services made available to the user, and under user control, that requires improvement in network switch architecture, signaling capabilities, and peripherals.

AMI u **Alternate Mark Inversion** Line-code type used on *T1* and *E1* circuits. In AMI, zeros are represented by 01 during each bit cell, and ones are represented by 11 or 00, alternately, during each bit cell. AMI requires that the sending device maintain ones density. Ones density is not maintained independently of the data stream. Sometimes called binary coded alternate mark inversion.

API u **Application Programming Interface** (1) The interface to a library of language-specific subroutines (such as a graphics library) that implement higher-level functions. (2) A set of calling conventions defining how a service is invoked through a software package.

ASCII u **American Standard Code for Information Interchange** The standard binary encoding of alphabetical characters, numbers, and other keyboard symbols.

ATM u **Asynchronous Transfer Mode** International standard for cell relay in which multiple service types (such as voice, video, or data) are conveyed in fixed-length (53-byte) cells. Fixed-length cells allow cell processing to occur in hardware, thereby reducing transit delays. ATM is designed to take advantage of high-speed transmission media such as *E3*, *SONET*, and *T3*.

B8ZS u **Binary 8-Zero Substitution** Line-code type, used on *T1* and *E1* circuits, in which a special code is substituted whenever 8 consecutive zeros are sent over the link. This code is then interpreted at the remote end of the connection. This technique guarantees ones density independent of the data stream. Sometimes called bipolar 8-zero substitution.

B Channel u **Bearer Channel** In ISDN, a full-duplex, 64-kbps channel used to send user data.

BIOS u **Basic Input/Output System** The built-in program that controls the basic functions of communications between the processor and the Input/Output (I/O) devices of a computer.

BISDN u **Broadband ISDN** *ITU-T* communication standards designed to handle high-bandwidth applications such as video. BISDN currently uses *ATM* technology over *SONET*-based transmission circuits to provide data rates from 155 to 622 Mbps and beyond.

bootROM u **boot Read-Only Memory** Chip mounted on the printed circuit board used to provide executable boot instructions to a computer device.

BRI u **Basic Rate Interface** *ISDN* interface composed of two *B Channel*s and one *D Channel* for circuit-switched communication of voice, video, and data.

BSP u **Board Support Package** A board support package consists of documentation and software used to configure and install a specific operating system on a specific product.

BUS u **Broadcast and Unknown Server** Multicast server used in *ELAN*s that is used to flood traffic addressed to an unknown destination and to forward multicast and broadcast traffic to the appropriate clients.

CAM u **Content Addressable Memory** Memory that is accessed based on its contents, not on its memory address.

CBR u **Constant Bit Rate** *QoS* class defined by the *ATM* Forum for ATM networks. CBR is used for connections that depend on precise clocking to ensure undistorted delivery.

CCS u **Common Channel Signaling** Signaling system used in telephone networks that separates signaling information from user data. A specified channel is exclusively designated to carry signaling information for all other channels in the system.

COM u **Communication or Communications**

CompactPCI u CompactPCI is an adaptation of the Peripheral Component Interconnect (*PCI*) Specification for industrial and/or embedded applications requiring a more robust mechanical form factor than desktop PCI. CompactPCI uses industry standard mechanical components and high performance connector technologies to provide an optimized system intended for rugged applications. CompactPCI provides a system that is electrically compatible with the PCI Specification, allowing low cost PCI components to be utilized in a mechanical form factor suited for rugged environments. CompactPCI is an open specification supported by the PICMG (PCI Industrial Computer Manufacturers Group), which is a consortium of companies involved in utilizing PCI for embedded applications.

CPCS u **Common Part Convergence Sublayer** An abstract *ATM* protocol *API* defined by the *ATM* Forum. It forms the boundary interface between the purely software implemented higher layer ATM protocols and the segmentation and reassembly process controlled by hardware.

CPM u **Communication Processing Module**

CRC4 u **Cyclic Redundancy Check**. Error-checking technique in which the frame recipient calculates a remainder by dividing frame contents by a prime binary divisor and compares the calculated remainder to a value stored in the frame by the sending node.

CS u **Convergence Sublayer** One of the two sublayers of the *AAL CPCS*, which is responsible for padding and error checking. *PDU*s passed from the *SSCS* are appended with an 8-byte trailer (for error checking and other control information) and padded, if necessary, so that the length of the resulting PDU is divisible by 48. These PDUs are then passed to the *SAR* sublayer of the CPCS for further processing.

CSU u **Channel Service Unit** A component that terminates a digital circuit, such as *T1*. A CSU assures compliance to FCC regulations and preforms some line-conditioning functions.

D Channel u **Data Channel** Full-duplex, 16-kbps (*BRI*) or 64-kbps (*PRI*) *ISDN* channel.

DCE u 1. **Data Communications Equipment** (EIA expansion). 2. **Data Circuit-terminating Equipment** (ITU-T expansion). Devices and connections of a communications network that comprise the network end of the user-to-network interface. The DCE provides a physical connection to the network, forwards traffic, and provides a clocking signal used to synchronize data transmission between DCE and DTE devices. Modems and interface cards are examples of DCE.

DLCI u **Data-Link Connection Identifier** Value that specifies a *PVC* or *SVC* in a *Frame Relay* network. In the basic Frame Relay specification, DLCIs are locally significant (connected devices might use different values to specify the same connection). In the *LMI* extended specification, DLCIs are globally significant (DLCIs specify individual end devices).

DMA u **Direct Memory Access** The transfer of data directly into memory without supervision of the processor. The data is passed on the bus directly between the memory and another device.

DPRAM u **Dual Port Random Access Memory**

DS1 u **Digital Signal level 1** Framing specification used in transmitting digital signals at 1.544-Mbps on a *T1* facility (in the United States) or at 2.108-Mbps on an *E1* facility (in Europe).

DS3 u **Digital Signal level 3** Framing specification used for transmitting digital signals at 44.736 Mbps on a *T3* facility.

DSX1 u Cross-connection point for *DS1* signals.

DTE u **Data Terminal Equipment** Device at the user end of a user-network interface that serves as a data source, destination, or both. DTE connects to a data network through a *DCE* device (for example, a modem) and typically uses clocking signals generated by the DCE. DTE includes such devices as computers, protocol translators, and multiplexers.

E1 u Wide-area digital transmission scheme used predominantly in Europe that carries data at a rate of 2.048 Mbps. E1 lines can be leased for private use from common carriers.

E3 u Wide-area digital transmission scheme used predominantly in Europe that carries data at a rate of 34.368 Mbps. E3 lines can be leased for private use from common carriers.

EEPROM u **Electrically Erasable Programmable Read-Only Memory** A nonvolatile *PROM* that can be written as well as read form. Usually used to hold information about the current system configuration, alternate boot paths, etc.

ELAN u **Emulated LAN** *ATM* network in which an Ethernet or Token Ring *LAN* is emulated using a client-server model. ELANs are composed of an *LEC*, an *LES*, a *BUS*, and an *LECS*. Multiple ELANs can exist simultaneously on a single ATM network. ELANs are defined by the *LANE* specification.

END u **Enhanced Network Driver**

EPLD u **Electrically Programmable Logic Device**

ES u **End System** Generally, an end-user device on a network.

ESF u **Extended Superframe Format** Framing type used on *T1* circuits that consists of 24 frames of 192 bits each, with the 193rd bit providing timing and other functions.

Ethernet u Baseband *LAN* specification invented by Xerox Corporation and developed jointly by Xerox, Intel, and Digital Equipment Corporation.

FCC u **Federal Communications Commission** The Government agency responsible for regulating telecommunications in the United States.

FCC u **Fast serial Communication Controllers** Used to control the fast Ethernet port.

FDDI u **Fiber Distributed Data Interface** *LAN* standard, defined by ANSI X3T9.5, specifying a 100-Mbps token-passing network using fiber-optic cable, with transmission distances of up to 2 km. FDDI uses a dual-ring architecture to provide redundancy.

Flash u Nonvolatile storage that can be electrically erased and reprogrammed so that software images can be stored, booted, and rewritten as necessary.

Frame Relay u Industry-standard, switched data link layer protocol that handles multiple virtual circuits using *HDLC* encapsulation between connected devices. Frame Relay is more efficient than *X.25*, the protocol for which it is generally considered a replacement.

FTP u **File Transfer Protocol** Application protocol, part of the *TCP/IP* protocol stack, used for transferring files between network nodes.

GB u **GigaBytes** 10^9 bytes per second.

Gbps u **Gigabits per second** 10^9 bits per second.

HDLC u **High-Level Data Link Control** Bit-oriented synchronous data link layer protocol developed by *ISO*. Derived from *SDLC*, *HDLC* specifies a data encapsulation method on synchronous serial links using frame characters and checksums.

IMA u **Inverse Multiplexing over ATM** Standard protocol defined by the ATM Forum in 1997.

IMMR u **Internal Memory Map Register**

IP u **Internet Protocol** Network layer protocol in the *TCP/IP* stack offering a connectionless internetwork service. IP provides features for addressing, type-of-service specification, fragmentation and reassembly, and security.

ISDN u **Integrated Services Digital Network** Communication protocol, offered by telephone companies, that permits telephone networks to carry data, voice, and other source traffic.

ISO u **International Organization for Standardization** International organization that is responsible for a wide range of standards, including those relevant to networking. ISO developed the *OSI* reference model, a popular networking reference model.

ITU-T u **International Telecommunication Union Telecommunication Standardization Sector** International body that develops worldwide standards for telecommunications technologies. The ITU-T carries out the functions of the former CCITT.

J1 u Japanese transmission standard

LAN u **Local-Area Network** High-speed, low-error data network covering a relatively small geographic area (up to a few thousand meters). LANs connect workstations, peripherals, terminals, and other devices in a single building or other geographically limited area. LAN standards specify cabling and signaling at the physical and data link layers of the *OSI* model. *Ethernet*, *FDDI*, and *Token Ring* are widely used LAN technologies.

LANE u **LAN Emulation** Technology that allows an *ATM* network to function as a *LAN* backbone. The ATM network must provide multicast and broadcast support, address mapping (*MAC Address*-to-ATM), *SVC* management, and a usable packet format. LANE also defines *Ethernet* and *Token Ring ELAN*s.

LAPB u **Link Access Procedure, Balanced**. Data link layer protocol in the *X.25* protocol stack. LAPB is a bit-oriented protocol derived from *HDLC*.

LEC u **LAN Emulation Client** Entity in an end system that performs data forwarding, address resolution, and other control functions for a single *ES* within a single *ELAN*. An LEC also provides a standard *LAN* service interface to any higher-layer entity that interfaces to the LEC. Each LEC is identified by a unique *ATM* address, and is associated with one or more *MAC Address*es reachable through that ATM address.

LECS u **LAN Emulation Configuration Server** Entity that assigns individual *LANE* clients to particular *ELAN*s by directing them to the *LES* that corresponds to the ELAN. There is logically one LECS per administrative domain, and this serves all ELANs within that domain.

LED u **Light Emitting Diode** A semiconductor device used to provide visual indications, used in place of an incandescent light. Also a semiconductor device used to transmit light into a fiber.

LES u **LAN Emulation Server** Entity that implements the control function for a particular *ELAN*. There is only one logical LES per ELAN, and it is identified by a unique *ATM* address.

LMI u **Local Management Interface** Set of enhancements to the basic *Frame Relay* specification. LMI includes support for a keepalive mechanism, which verifies that data is flowing; a multicast mechanism, which provides the network server with its local *DLCI* and the multicast DLCI; global addressing, which gives DLCIs global rather than local significance in Frame Relay networks; and a status mechanism, which provides an on-going status report on the DLCIs known to the switch. Known as LMT in ANSI terminology.

MAC Address u Standardized data link layer address that is required for every port or device that connects to a *LAN*. Other devices in the network use these addresses to locate specific ports in the network and to create and update routing tables and data structures. MAC addresses are 6 bytes long and are controlled by the IEEE. Also known as a hardware address, MAC-layer address, and physical address.

MCC u **Multichannel Communication Controller**

MiniDIN u Miniature multi-pin connector.

MUX u **Multiplexer** Combines multiple signals for transmission over a single line. The signals are demultiplexed, or separated, at the receiving end

NT1 u **Network Termination 1** A device that provides the interface between customer premises equipment and central office switching equipment.

NVRAM u **Nonvolatile RAM** *RAM* that retains its contents when a unit is powered off.

OC3 u **Optical Carrier 3** Physical protocol defined for *SONET* optical signal transmissions. OC3 signal levels put *STS* frames onto multimode fiber-optic line at 155.52 Mbps.

OSI u **Open System Interconnection** International standardization program created by *ISO* and *ITU-T* to develop standards for data networking that facilitate multivendor equipment interoperability.

PCI u **Peripheral Component Interconnect** A high-performance multiplexed address and data bus. Supporting 32-bit with optional 64-bit data transfers, the PCI bus is intended to be an interconnect between peripheral controllers, peripheral add-in boards, and processor/memory systems. The PCI bus operates at up to 33 MHz, providing burst transfer rates up to 132 MBps 32 bits wide, or up to 264 MBps 64 bits wide.

PDN u **Public Data Network** Network operated either by a government (as in Europe) or by a private concern to provide computer communications to the public, usually for a fee. PDNs enable small organizations to create a *WAN* without all the equipment costs of long-distance circuits.

PDU u **Protocol Data Unit** A message of a given protocol comprising payload and protocol-specific control information, typically contained in a header.

PLP u **Packet Level Protocol** Network layer protocol in the *X.25* protocol stack. Sometimes called X.25 Level 3 and X.25 Protocol.

PMC u **PCI Mezzanine Card** *PCI* "daughter" card designed to mount on a "mother card".

POST u **Power-On-Self-Test** Test that automatically runs whenever the power is applied to the card.

PRI u **Primary Rate Interface** *ISDN* interface to primary rate access. Primary rate access consists of a single 64-Kbps *D Channel* plus 23 (*T1*) or 30 (*E1*) *B Channel*s for voice or data.

PROM u **Programmable Read-Only Memory** *ROM* that can be programmed using special equipment. PROMs can be programmed only once.

PVC u **Permanent Virtual Circuit or Connection** Virtual circuit that is permanently established. PVCs save bandwidth associated with circuit establishment and tear down in situations where certain virtual circuits must exist all the time. In *ATM* terminology, called a permanent virtual connection.

QoS u **Quality of Service** Measure of performance for a transmission system that reflects its transmission quality and service availability.

RAM u **Random-Access Memory** Volatile memory that can be read and written by a microprocessor.

RISC u **Reduced Instruction Set Computing**

ROM u **Read-Only Memory** Nonvolatile memory that can be read, but not written, by the microprocessor.

RTM u **Rear Transition Module** A module that provides network connections from the rear of a system.

Rx u **Receive or Receiver**

SAR u **Segmentation And Reassembly** One of the two sublayers of the *AAL CPCS*, responsible for dividing (at the source) and reassembling (at the destination) the *PDU*s passed from the *CS*. The SAR sublayer takes the PDUs processed by the CS and, after dividing them into 48-byte pieces of payload data, passes them to the *ATM* layer for further processing.

SCC u **Serial Communication Controller**

SDH u **Synchronous Digital Hierarchy** European standard that defines a set of rate and format standards that are transmitted using optical signals over fiber. SDH is similar to *SONET*, with a basic SDH rate of 155.52 Mbps, designated at *STM-1*.

SDLC u **Synchronous Data Link Control** *SNA* data link layer communications protocol. SDLC is a bit-oriented, full-duplex serial protocol that has spawned numerous similar protocols, including *HDLC* and *LAPB*.

SDU u **Service Data Unit** A unit of interface information whose identity is preserved from one end of a layer connection to the other.

SDRAM u **Synchronous Digital Random Access Memory**

SEAL u **Simple And Efficient AAL** Scheme used by *AAL5* in which the *SAR* sublayer segments *CS PDU*s without adding additional fields.

SIU u **Serial Interface Unit**

SMC u **Serial Management Controller**

SMDS u **Switched Multimegabit Data Service** High-speed, packet-switched, datagram-based *WAN* networking technology offered by the telephone companies.

SNA u **Systems Network Architecture** Large, complex, feature-rich network architecture developed in the 1970s by IBM.

SONET u **Synchronous Optical Network** High-speed (up to 2.5 Gbps) synchronous network specification developed by Bellcore and designed to run on optical fiber. *STS1* is the basic building block of SONET. Approved as an international standard in 1988.

SS7 u **Signaling System 7** Standard *CCS* system used with *BISDN* and *ISDN*.

SSCS u **Service Specific Convergence Sublayer** One of the two sublayers of any *AAL*. SSCS, which is service dependent, offers assured data transmission. The SSCS can be null as well, in classical *IP* over *ATM* or *LAN* emulation implementations.

STM-1 u **Synchronous Transport Module level 1** One of a number of *SDH* formats that specifies the frame structure for the 155.52-Mbps lines used to carry *ATM* cells.

STS u **Synchronous Transport Signal**

STS1 u **Synchronous Transport Signal level 1** Basic building block signal of *SONET*, operating at 51.84 Mbps. Faster SONET rates are defined as STS-n, where n is a multiple of 51.84 Mbps.

SVC u **Switched Virtual Circuit** Virtual circuit that is dynamically established on demand and is torn down when transmission is complete. SVCs are used in situations where data transmission is sporadic. Called a switched virtual connection in *ATM* terminology.

T1 u T1 transmits *DS1*-formatted data at 1.544 Mbps through the telephone-switching network, using *AMI* or *B8ZS* coding.

T3 u Digital *WAN* carrier facility. T3 transmits *DS3*-formatted data at 44.736 Mbps through the telephone switching network.

TCP u **Transmission Control Protocol** Connection-oriented transport layer protocol that provides reliable full-duplex data transmission. TCP is part of the *TCP/IP* protocol stack.

TCP/IP u **Transmission Control Protocol/Internet Protocol** Common name for the suite of protocols developed by the U.S. DoD in the 1970s to support the construction of worldwide internetworks. *TCP* and *IP* are the two best-known protocols in the suite.

TFTP u **Trivial File Transfer Protocol** Simplified version of *FTP* that allows files to be transferred from one computer to another over a network.

Token Ring u Token-passing *LAN* developed and supported by IBM. Token Ring runs at 4 or 16 Mbps over a ring topology. Similar to IEEE 802.5.

TTY u **Teletypewriter** General term for an input device.

Tx u **Transmit or Transmitter**

USRBUF u A driver structure describing the use of a specific buffer containing payload data to be transferred using *ATM*. They can be linked together to allow non-contiguous areas of memory to be sent as one unit.

X.25 u *ITU-T* standard that defines how connections between *DTE* and *DCE* are maintained for remote terminal access and computer communications in *PDN*s. X.25 specifies *LAPB*, a data link layer protocol, and *PLP*, a network layer protocol. *Frame Relay* has to some degree superseded X.25.

VBR u **Variable Bit Rate** *QoS* class defined by the *ATM* Forum for ATM networks. VBR is subdivided into a Real Time (RT) class and Non-Real Time (NRT) class. VBR (RT) is used for connections in which there is a fixed timing relationship between samples. VBR (NRT) is used for connections in which there is no fixed timing relationship between samples, but that still need a guaranteed QoS.

VCC u **Virtual Channel Connection** Can be a Permanent Virtual Connection (*PVC*) or a Switched Virtual Connection (*SVC*). Any *ATM* connection between two nodes.

VCI u **Virtual Channel Identifier**16-bit field in the header of an *ATM* cell. The VCI, together with the *VPI*, is used to identify the next destination of a cell as it passes through a series of ATM switches on its way to its destination. ATM switches use the VPI/VCI fields to identify the next network VCL that a cell needs to transit on its way to its final destination. The function of the VCI is similar to that of the *DLCI* in *Frame Relay*.

VCL u **Virtual Channel Link** Connection between two *ATM* devices. A *VCC* is made up of one or more VCLs.

VPI u **Virtual Path Identifier** 8-bit field in the header of an *ATM* cell. The VPI, together with the *VCI*, is used to identify the next destination of a cell as it passes through a series of ATM switches on its way to its destination. ATM switches use the VPI/VCI fields to identify the next VCL that a cell needs to transit on its way to its final destination. The function of the VPI is similar to that of the *DLCI* in *Frame Relay*.

WAN u **Wide-Area Network** Data communications network that serves users across a broad geographic area and often uses transmission devices provided by common carriers. *Frame Relay*, *SMDS*, and *X.25* are examples of WANs.

Index

When using this index, keep in mind that a page number indicates only where referenced material begins. It may extend to the page or pages following the page referenced.