

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*



PMC-8260/DS1

Reference Guide

P / N 222192 Revision AA
January 2004

Copyright

The information in this publication is subject to change without notice. Force Computers, GmbH reserves the right to make changes without notice to this, or any of its products, to improve reliability, performance, or design.

Force Computers, GmbH shall not be liable for technical or editorial errors or omissions contained herein, nor for indirect, special, incidental, or consequential damages resulting from the furnishing, performance, or use of this material. This information is provided “as is” and Force Computers, GmbH expressly disclaims any and all warranties, express, implied, statutory, or otherwise, including without limitation, any express, statutory, or implied warranty of merchantability, fitness for a particular purpose, or non-infringement.

This publication contains information protected by copyright. This publication shall not be reproduced, transmitted, or stored in a retrieval system, nor its contents used for any purpose, without the prior written consent of Force Computers, GmbH.

Force Computers, GmbH assumes no responsibility for the use of any circuitry other than circuitry that is part of a product of Force Computers, GmbH. Force Computers, GmbH does not convey to the purchaser of the product described herein any license under the patent rights of Force Computers, GmbH nor the rights of others.

Copyright© 2004 by Force Computers, GmbH. All rights reserved.

The Force logo is a trademark of Force Computers, GmbH. SENTINEL is a registered trademark of Force Computers, GmbH

IEEE is a registered trademark of the Institute for Electrical and Electronics Engineers, Inc.
PICMG, CompactPCI, and the CompactPCI logo are registered trademarks and the PICMG logo is a trademark of the PCI Industrial Computer Manufacturer’s Group.
MS-DOS, Windows95, Windows98, Windows2000 and Windows NT are registered trademarks and the logos are a trademark of the Microsoft Corporation.
Intel and Pentium are registered trademarks and the Intel logo is a trademark of the Intel Corporation.
SPARC is a registered trademark, the SPARC logo is a trademark and UltraSPARC is a registered trademark of SPARC International, Inc.
PowerPC is a registered trademark and the PowerPC logo is a trademark of International Business Machines Corporation.
AltiVec is a registered trademark and the AltiVec logo is a trademark of Motorola, Inc.
Sun, Sun Microsystems, the Sun logo, SPARCengine Ultra, Solaris, Open Boot, SunVTS are trademarks or registered trademarks of SUN Microsystems, Inc.
The Linux Kernel is Copyright© Linus B. Torvalds under the terms of the General Public License (GPL).
GoAhead ia a registered trademark of GoAhead Software, Inc. and SelfReliant and Self Availability are trademarks of GoAhead Software, Inc.
LynxOS and BlueCat are registered trademarks of LynuxWorks, Inc.
Tornado, VxWorks, Wind, WindNavigator, Wind River Systems, Wind River Systems and design, WindView, WinRouter and Xmath are registered trademarks or service marks of Wind River Systems, Inc.
Envoy, the Tornado logo, Wind River, and Zinc are trademarks or service marks of Wind River Systems, Inc.
Sony is a registered trademark of Sony Corporation, Japan
Ethernet is a trademark and Xerox is a registered trademark of Xerox Corporation

Other product names mentioned herein may be trademarks and / or registered trademarks of their respective companies.



World Wide Web: www.fci.com
24-hour access to on-line manuals, driver updates, and application notes is provided via SMART, our SolutionsPLUS customer support program that provides current technical and services information.

Headquarters

The Americas

Force Computers Inc.
4211 Starboard Drive
Fremont CA 94538

Tel.: +1 (510) 624-5300
Fax: +1 (510) 624-5301
Email: support@fci.com

Europe

Force Computers GmbH
Lilienthalstr. 15
D-85579 Neubiberg / München

Tel.: +49 (89) 608 14-0
Fax: +49 (89) 609 77 93
Email: support-de@fci.com

Asia

Force Computers Japan K.K.
Shibadaimon MF Bldg. 4F
Shiba Daimon 2-1-16
Minato-ku, Tokyo 105-0012

Tel.: +81 (03) 3437-3948
Fax: +81 (03) 3437-3968
Email: support-de@fci.com

222192 420 000 AA

Contents

Using This Guide

Other Sources of Information

Safety Notes

Sicherheitshinweise

1 Introduction

Features 1-3

Variants 1-5

PMC-8260/DS1-SC-F 1-5
Variant Features 1-5
Restrictions 1-5

PMC-8260/DS1-H110-F and -RLH 1-6
Variant Features 1-6
Restrictions 1-6

Standard Compliances 1-7

Ordering Information 1-8
Product Nomenclature 1-8
Order Numbers 1-9

2 Installation

Action Plan 2-3

Requirements 2-4
Long Haul Requirements 2-4
Carrier Board Signaling Level Requirements 2-4
Environmental Requirements 2-4
Power Requirements 2-5
FCC Requirements for PMC-8260/DS1 2-6

Switch Settings 2-7

Allowed Product Combinations 2-8

Installing a PMC Module 2-10
Installation Procedure 2-11
Removal Procedure 2-13

Software Upgrades and Accessories 2-14

3 Indicators and Connectors

Front Panel 3-3
Connectors 3-4
Serial Connector 3-4
Line Interface Connector 3-5
LEDs 3-5

On-Board Connectors 3-7
PMC I/O 3-8
JTAG/Debug 3-11

4 Firmware

Introduction 4-3

Features 4-3

Resources Used by the Firmware 4-3

Power-Up Sequence 4-4

Loading Application Software 4-8

Requirements for Both Loading Methods 4-8

Preparing a ROM Image 4-9

SYPCR Register 4-9

ROM Image Configuration 4-9

Argument Passed to ROM Image 4-11

Execution of ROM Image 4-11

System State at ROM Image Execution 4-12

Controlling Boot Process 4-13

Monitoring Boot Process 4-14

Primary Booter 4-15

Resources Used by the Primary Booter 4-15

Control/Status Register Section 4-16

Primary Booter Commands 4-18

Starting the Primary Booter 4-20

Reading and Writing 4-22

Programming a Boot Image into Boot Flash 4-23

Obtaining Results from the Power-On Self-Test 4-24

Requirements 4-24

POST Result Storage Area 4-24

Global Results 4-25

Checksum 4-26

POST Progress Information 4-26

Device-Specific Results 4-26

Code Examples 4-30

Data Definitions 4-30

Resetting the PMC-8260/DS1 4-32

Starting the Primary Booter 4-33

Waiting for Command Completion 4-35

Data Transfer and Flash Programming 4-36

Erasing the ROM Image 4-37

Checking Alive State of Target 4-37
Starting Code Execution 4-38
Reading POST Results 4-38

5 Memory Map and Devices

Block Diagram 5-3

Memory Map 5-4

Interrupts 5-5

Motorola PowerQUICC II 5-6

Bus Monitor 5-6

Communication Processor Module 5-6
Serial Communication Channels 5-6
Serial E^2PROM 5-7
Serial Interface with Time Slot Assigner and Multi-Channel Controller 5-7

PowerSPAN II PCI Bus Bridge 5-9

60x Bus Arbiter 5-9

E^2PROM Contents 5-9

PowerSPAN II Configuration 5-11
PCI Lockout 5-11
Opening More Windows in the PCI Target Channel 5-12
Accessible 60x Bus Resources 5-13
Opening Windows in the 60x Bus Slave Channel 5-14

SDRAM and its Performance 5-15

SDRAM Machine Memory Controller 5-15

SSRAM Information for Support Package Programmers 5-16

I/O-Bus 5-18

Boot Flash PROM 5-18
Programming the Boot Flash when PowerQUICC II is in Reset 5-19
Memory Map when PowerQUICC II is in Reset 5-20
Displaying the Boot Flash Programming Progress via LEDs 5-22

Agere Ambassador T8105 5-22
PMC-Sierra PM4351 5-23
LED Control Register 5-24

Resetting the Devices 5-26
PowerQUICC II 5-26
Local I/O Devices from the PCI Bus 5-27

6 TDM Channel Configuration

About this Chapter 6-3

Synchronizing the Clock 6-4
Configuring the PMC-8260/DS1 as Clock Master 6-4
Configuring the PMC-8260/DS1 as Clock Slave 6-6

Routing Time Division Multiplex Timeslots 6-7

PowerQUICC II Port Functions on PMC-8260/DS1 6-11

A Troubleshooting

Index

Product Error Report

Tables

Introduction

Table 1 Standard Compliances 1-7
Table 2 Ordering Information 1-9

Installation

Table 3 Environmental Requirements of the PMC-8260/DS1 2-5
Table 4 Current and Power Requirements 2-6
Table 5 Default Switch Settings for SW1 2-7
Table 6 PMC-8260/DS1 on Reference Platforms 2-8
Table 7 Revisions Supported by Reference Platform 2-9

Indicators and Connectors

Table 8 Availablility of Serial I/O Channels 3-4
Table 9 LED States During Power Up 3-5

Firmware

Table 10 Steps Performed During Power-Up 4-6
Table 11 ROM Image Configuration Section 4-10
Table 12 PCR Register Settings 4-13
Table 13 Primary Booter CSR Section 4-17
Table 14 Primary Booter Commands and Parameters 4-18
Table 15 POST Result Storage Area 4-25
Table 16 POST Result and Status Bits 4-25
Table 17 SDRAM Test 4-27
Table 18 Synchronous SRAM Test 4-27

Table 19 PowerSpan II Tests 4-28
Table 20 Framer Tests 4-28
Table 21 T8105 Tests 4-29

Memory Map and Devices

Table 22 Memory Map 5-4
Table 23 Interrupt Inputs of the PowerQUICC II 5-5
Table 24 PowerSPAN II Access from 60x Bus 5-9
Table 25 PowerSPAN II Serial E^2PROM Contents 5-10
Table 26 PCI Target Image#0 Programming Example 5-13
Table 27 60x Bus Slave Image#0 Programming Example 5-14
Table 28 SDRAM Access on 60x Bus (PowerPC Bus) 5-15
Table 29 SSRAM Timing Performance at 66 MHz (T=15.15 nsec) 5-16
Table 30 SSRAM Address on 60x-Bus/Local Bus 5-17
Table 31 Flash PROM Access on 60x Bus 5-18
Table 32 Memory Map when PowerQUICC II is in Reset 5-21
Table 33 Flash PROM Access from PCI Bus with PowerQUICC II Held in Reset 5-21
Table 34 T8105 Access on 60x-Bus 5-22
Table 35 Framer Access on 60x Bus 5-23
Table 36 LED Control Register Access on 60x-Bus 5-24
Table 37 LED Control Register 5-24

TDM Channel Configuration

Table 38 Used Port Pins of SI1 6-11
Table 39 Used Port Pins of SI2 6-12

Figures

Introduction

Figure 1 Function Blocks 1-4

Installation

Figure 2 Location of Switch 2-7
Figure 3 Installation (Example with PowerCoreCPCI-6750 Rev. 2.x) 2-12

Indicators and Connectors

Figure 4 Front Panels 3-3
Figure 5 Front Panel Serial I/O Port Connector Pinout 3-4
Figure 6 Front Panel RJ-45 Interface Connector Pinout 3-5
Figure 7 PMC I/O and JTAG/Debug Connector 3-7
Figure 8 PMC I/O Connector Pinout for PMC-8260/DS1-SC-F 3-8
Figure 9 PMC I/O Connector Pinout for PMC-8260/DS1-H110-F 3-9
Figure 10 PMC I/O Connector Pinout for PMC-8260/DS1-H.110-RLH 3-10
Figure 11 JTAG/Debug Connector 3-11
Figure 12 JTAG/Debug Connector Pinout 3-12

Firmware

Figure 13 Flowchart of the PMC-8260/DS1 Firmware Power-Up Sequence 4-5
Figure 14 Memory Layout of the Primary Booter 4-16

Memory Map and Devices

Figure 15 Block Diagram 5-3
Figure 16 I/O Bus 5-18
Figure 17 Byte Address Generation for Flash Programming from PCI Bus via PowerSPAN II . . . 5-20

TDM Channel Configuration

Figure 18 Local TDM Clocking Structure 6-5
Figure 19 Connection of TDM Channels 6-7
Figure 20 Switching of TDM Channels 6-8
Figure 21 Connection of Local TDM Streams 6-9
Figure 22 Connection of Data Lines in Inter-TDM Link 6-10

Using This Guide

This Reference Guide is intended for users qualified in electronics or electrical engineering. Users must have a working understanding of Peripheral Component Interconnect (PCI), Compact Peripheral Component Interconnect (CPCI), and telecommunications.

Conventions

Notation	Description
1234	All numbers are decimal numbers except when used with the notations described below
00000000_{16}	Typical notation for hexadecimal numbers (digits are 0 through F), e.g. used for addresses and offsets
0000_2	Same for binary numbers (digits are 0 and 1)
x	Generic use of a letter
n	Generic use of numbers
Bold	Character format used to emphasize a word
`Courier`	Character format used for on-screen output
`Courier+Bold`	Character format used to characterize user input
Italics	Character format for references, table, and figure descriptions
<text>	Typical notation used for variables and keys
[text]	Typical notation for buttons
...	Repeated item
..	Ranges
:	Extents
Note:	No danger encountered. Pay attention to important information marked using this layout
Caution 	Possibly dangerous situation: slight injuries to people or damage to objects possible

Abbreviations

AACK	Address Acknowledge
BD	Buffer Descriptor
BIB	Board Information Block
BSP	Board Support Package
BST	Base Translation
CAS	Column Address Strobe
CE	Consumer Electronics
CI	Cache Inhibit
COP	Common On-Chip Processor
CP	Communication Processor
CompactPCI	Compact Peripheral Component Interconnect
CPM	Communication Processor Module
CPU	Central Processing Unit
CS	Chip Select
CSR	Control/Status Register
CSU	Channel Service Unit
DMA	Direct Memory Access
DRAM	Dynamic Random Access Memory
EMC	Electromagnetic Compatibility
ESD	Electrostatic Discharge
FAE	Field Applications Engineer
FCC	Federal Communications Commission
GND	Ground
GPCM	General-Purpose Chip-Select Module
GPIO	General Purpose Input/Output
IC	Integrated Circuit
ICT	In-Circuit Tester
IDMA	Independent DMA

IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IMMR	Internal Memory Map Registers
ISB	Internal Space Base
ITU	International Telecommunication Union
JTAG	Joint Test Action Group
LAPD	Link Access Procedure-D
LDI	Local Data Input
LDO	Local Data Output
LED	Light-Emitting Diode
LFM	Linear Feet per Minute
LSB	Least Significant Bit
MAC	Media Access Control
MMU	Memory Management Unit
MPTPR	Memory Periodic Timer Prescaler Register
MSB	Most Significant Bit
MSR	Machine State register
MTP	Message Transfer Protocol
PB	Processor Bus
PCA	Printed Circuit Assembly
PCI	Peripheral Component Interconnect
PCM	Pulse Code Modulation
PCR	Power-Up Control Register
PMC	PCI Mezzanine Card
POST	Power-On Self Test
PRC	POST Result Code
PROM	Programmable Read-Only Memory
PSDVAL	Partial Data Valid
PSR	Power-up Status Register
RAM	Random Access Memory

RDY	Ready
ROMEX	ROM Execution
RTB	Rear Transition Board
RTU	Right to use
SCC	Serial Communication Controller
SCL	Serial Clock
SDA	Serial Data
SDMA	Serial DMA
SDRAM	Synchronous Dynamic Random Access Memory
SI	Serial Interface
SMC	Serial Management Controller
SMD	Surface Mounted Device
SSRAM	Synchronous Static Random Access Memory
SYPCR	System Protection Control register
TA	Transfer Acknowledge
TDI	Test Data In
TDM	Time Division Multiplex
TDO	Test Data Out
TNV	Telecommunication Network Voltage
TSA	Time Slot Assigner
TSR	Test Status Register
UPM	User-programmable Machine
VME	Versa Module Eurocard
VPD	PCI Vital Product Data

Revision History

Order No.	Revision	Date	Description
212125	AA	April 2000	Preliminary Installation Guide
212125	AB	November 2000	Updated the “Order Numbers” section on page 1-8 Added the “FCC Requirements for PMC-8260/DS1” section on page 2-6 Updated the “Firmware” chapter Added the “Code Examples” section on page 4-26 Added time delay information in the “Programming the Boot Flash when Power-QUICC II is in Reset” section on page 5-19 Added the “Troubleshooting” chapter Editorial changes
214293	AA	January 2001	Added Table 7 “Revisions Supported by Reference Platform” on page 2-9; corrected Figure 22 “Interrupt Structure” on page 6-9
214293	AB	April 2001	Added note to the “Routing Time Division Multiplex Timeslots” section on page 6-7; Editorial changes
215165	AA	August 2001	Corrected information in the “Front Panel” section on page 1-3; Changed Table 2 “Ordering Information” on page 1-8; Removed information on PMC-8260/DS1-N variant; Added note to the “Action Plan” section on page 2-3; Corrected setting of SW1-4 in Table 5 “Default Switch Settings for SW1” on page 2-7; Corrected Figure 3 “Front Panel Location Diagrams” on page 2-12; Corrected Figure 10 “PMC I/O Connector Pinout for PMC-8260/DS1-H.110-RLH” on page 3-10; Added note to the “JTAG/Debug” section on page 3-11; Removed Troubleshooting chapter; Added the “Appendix” chapter;
215165	AA	August 2001	Added the the “Index” section on page I-1 Editorial changes;

Order No.	Revision	Date	Description
215165	AB	November 2001	Editorial changes: removed picture from title page, corrected document number
216053	AA	March 2002	Modified chapters Safety Notes and Sicherheitshinweise; revised the “Features” section on page 1-3, the “Standard Compliances” section on page 1-6 and the “Ordering Information” section on page 1-7; moved figure “Block Diagram” to the “Memory Map and Devices” chapter; updated order numbers in the “Ordering Information” section on page 1-8; deleted all references to the 133 MHz CPU core frequency variant; revised and moved section “Variants of the PMC-8260/DS1” to chapter “Introduction”; replaced power values of 133 MHz variant by values of 200 MHz variant in the “Power Requirements” section on page 2-5; revised the “Allowed Product Combinations” section on page 2-8; removed section “Location Overview”; modified the “Installing a PMC Module” section on page 2-10; removed Figure 11 “Inter-TDM Link Signals on PMC I/O Connector” on page 3-6 and integrated information in Figure 10 “PMC I/O Connector Pinout for PMC-8260/DS1-H.110-RLH” on page 3-10; modified Figure 8 “PMC I/O Connector Pinout for PMC-8260/DS1-SC-F” on page 3-8 and Figure 9 “PMC I/O Connector Pinout for PMC-8260/DS1-H110-F” on page 3-9; renamed figure 10 to “PMC I/O Connector Pinout for PMC-8260/DS1-H.110-RLH”; implemented information on COP signals into Figure 12 “JTAG/Debug Connector Pinout” on page 3-12; deleted figure “Local TDM Streams Structure on PMC-8260/DS1-N-(R/F) Variant”; completely revised the “Firmware” chapter; completely revised the “Memory Map and Devices” chapter; completely revised the former “Structures” chapter and renamed to “TDM Channel Configuration

Order No.	Revision	Date	Description
217254	AA	April 2003	Updated “Other Sources of Information” page xxii; updated section “Features” on page 1-3, updated Table 2 “Ordering Information” on page 1-9; changed max. power supply ramp-up time from 5 to 20 msec page 2-5; updated Table 4 “Current and Power Requirements” on page 2-6; added the “Software Upgrades and Accessories” section on page 2-14; modified H110-RLH front panel in Figure 4 “Front Panels” on page 3-3; modified and added information to the “Preparing a ROM Image” section on page 4-9; replaced PowerSpan with PowerSpan II throughout the document; replaced PowerQUICC II frequency of 200 MHz with 300 MHz throughout the document; added descriptions for bits 3..0 in Table 37 “LED Control Register” on page 5-24; added the new on-board 8 kHz clock to list of possible clock sources in the “Configuring the PMC-8260 / DS1 as Clock Master” section on page 6-4, editorial changes
222192	AA	January 2004	Changed manual type from “Installation Guide” to “Reference Guide”; updated “Other Sources of Information” page xxii; added PCI interface data to the “Features” section on page 1-3; modified Figure 1 “Function Blocks” on page 1-4; added the “Carrier Board Signaling Level Requirements” section on page 2-4; added PCI bus and signaling level information to the “Installing a PMC Module” section on page 2-10; modified Figure 15 “Block Diagram” on page 5-3

Other Sources of Information

For further information, refer to the following documents or data sheets of the devices given in the table below.

Company	www.	Document/Device
Agere Systems	agere.com	Agere T8105 Advance
AMD	amd.com	AMD AM29LV017B
Atmel	atmel.com	AT24C02 Serial I^2C EEPROM
Force Computers	fci.com	ACC / RTB8260 / DS1 Installation Guide
		MTP-2 Rel. 5.x Programmer's Giude
		MTP-2 Rel. 4.2.0 for Solaris Installation Guide
		MTP-2 Rel. 5.0.5 for VxWorks Installation Guide
		MTP-2 Rel. 4.2.0 for Linux Installation Guide
		StackWare Rel. 4.2.0 Release Notes
		Distributed MTP-3 Rel. 5.x Programmer's Guide
		Distributed MTP-3 Rel. 5.0.1 Release Notes
		DMTP-3 Rel. 5.0.1 for MontaVista Linux Carrier Grade Edition Installation Guide
IEEE	ieee.org	IEEE Standard Physical and Environmental Layers for PCI Mezzanine Cards (PMC) - IEEE1386.1-2001
Micron	micron.com	MT48LC4M16A2 and MT48LC8M16A2 (SDRAM)
		MT58L256L36F-S-10 (SSRAM)
Motorola	motorola.com	MPC8260 PowerQUICC II User's Manual
		MPC8260 PowerQUICC II User's Manual: MPC8260A (HiP4) Supplement
		MPC826xA (HiP4) Family Hardware Specification
PMC-Sierra	pmc-sierra.com	PMC-Sierra PM4351 (COMET)

Company	www.	Document/Device
Tundra	tundra.com	PowerSpan II ™ PowerPC-to-PCI Bus Switch User Manual
		PowerSpan II and PowerSpan Differences Summary



Safety Notes

This section provides safety precautions to follow when installing, operating, and maintaining the PMC-8260/DS1.

We intend to provide all necessary information to install and handle the PMC-8260/DS1 in this Reference Guide. However, as the product is complex and its usage manifold, we do not guarantee that the given information is complete. If you need additional information, ask your Force Computers representative.

The PMC-8260/DS1 has been designed to meet the standard industrial safety requirements. It must not be used except in its specific area of office telecommunication industry.

Only personnel trained by Force Computers or persons qualified in electronics or electrical engineering are authorized to install, maintain, and operate the PMC-8260/DS1. The information given in this manual is meant to complete the knowledge of a specialist and must not be taken as replacement for qualified personnel.

EMC

The module has been tested in a standard Force Computers system and found to comply with the limits for a Class A digital device in this system, pursuant to part 15 of the FCC Rules respectively EN 55022 Class A. These limits are designed to provide reasonable protection against harmful interference when the system is operated in a commercial, business or industrial environment.

The module generates and uses radio frequency energy and, if not installed properly and used in accordance with this Reference Guide, may cause harmful interference to radio communications. Operating the system in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.



Installation

Be sure to mount only allowed combinations of PMC-8260/DS1 variants (the “Allowed Product Combinations” section page 2-8). Otherwise damage to PMC module, carrier board and equipment attached to the rear transition board may occur.

The PMC-8260/DS1 may only be connected to TNV-1 circuits (Telecommunication Network Voltage-1 circuits) defined in EN 60950. Otherwise, the PMC module will be damaged.

Electrostatic discharge and incorrect module installation and removal can damage circuits or shorten their life. Therefore:

- **Before installing or removing the PMC module, read the “Installation” chapter and the Installation Guide of the carrier board under consideration.**
- **Before installing the PMC module on or removing it from a carrier board not supporting hot swap, proceed as follows:**
  - **Check all installed boards for steps that you have to take before turning off the power.**
  - **Take those steps.**
  - **Finally turn off the power.**
- **Before touching modules or electronic components, make sure that you are working in an ESD-safe environment.**
- **The 3.3V and 5V power supplies of the PMC module carrier board must ramp up simultaneously in about max. 20 msec. Otherwise, the PMC module may be damaged.**
- **The power supply circuits on VME bus carrier boards may be overloaded if more than one PMC-8260/DS1 is assembled. This results in permanent damage to the carrier board. Therefore, make sure that the VMEbus carrier board’s +3.3 Volt supply supports the power requirements as described in the “Power Requirements” section page 2-5.**
- **The PMC specification (P1386.1) specifies a maximum total power consumption of 7.5W per single PMC module. The PMC-8260/DS1 in its standard configuration may exceed this value. Therefore, it is neccessary to synchronize the carrier board’s power supply with the PMC-8260/DS1 power requirements. Due to the high power dissipation forced air cooling is necessary (see also forced airflow environmental requirement on page 2-5).**



Operation

While operating the module ensure that the power and environmental requirements as given in the "Requirements" section on page 2-4 are met.

When operating the carrier board and the PMC module in areas of electromagnetic radiation ensure that the carrier board is bolted on the VME or CompactPCI system and is shielded by enclosure.

Make sure that contacts and cables of the module cannot be touched while it is operating. Touching contacts can cause contact oxidation and reduces conductivity and touching power pins is dangerous.

Do not operate the product outside the specified environmental limits. High humidity and condensation may cause short circuits. Make sure the product is completely dry and there is no condensation of water on any parts of the board's surface before applying power. Do not operate the product below 0°C.

RJ-45 Connector

Connecting a telephone or Ethernet to the RJ-45 connector of the PMC-8260/DS1-H110-F and SC-F variants causes damage to the telephone, the PMC module and to attached devices. Therefore, only connect E1/T1/J1 lines to the RJ-45 connector.

Environment

Always dispose of old modules according to your country's legislation, if possible in an environmentally acceptable way.





Sicherheitshinweise

Dieser Abschnitt enthält Sicherheitshinweise, die bei der Installation, dem Betrieb und der Wartung des PMC-8260/DS1 zu beachten sind.

Wir sind darauf bedacht, alle notwendigen Informationen zur Installation und zum Betrieb mit dem PMC-8260/DS1 in diesem Handbuch bereit zu stellen. Da es sich jedoch bei dem PMC-8260/DS1 um ein komplexes Produkt mit vielfältigen Einsatzmöglichkeiten handelt, können wir die Vollständigkeit der im Handbuch enthaltenen Informationen nicht garantieren. Falls Sie weitere Informationen benötigen, wenden Sie sich bitte an die für Sie zuständige Geschäftsstelle von Force Computers.

Das PMC-8260/DS1 erfüllt die für die Industrie geforderten Sicherheitsvorschriften und darf ausschließlich für Anwendungen in der Telekommunikationsindustrie verwendet werden.

Installation, Wartung und Betrieb dürfen nur von durch Force Computers ausgebildetem oder im Bereich Elektronik oder Elektrotechnik qualifiziertem Personal durchgeführt werden. Die in diesem Handbuch enthaltenen Informationen dienen ausschließlich dazu, das Wissen von Fachpersonal zu ergänzen, können dieses jedoch nicht ersetzen.

EMV

Das PMC Modul wurde in einem Force Computers Standardsystem getestet. Es erfüllt die für digitale Geräte der Klasse A gültigen Grenzwerte in einem solchen System gemäß den FCC-Richtlinien Abschnitt 15 bzw. EN 55022 Klasse A. Diese Grenzwerte sollen einen angemessenen Schutz vor Störstrahlung beim Betrieb des PMC Moduls in Geschäfts-, Gewerbe- sowie Industriebereichen gewährleisten.

Das PMC Modul arbeitet im Hochfrequenzbereich und erzeugt Störstrahlung. Bei unsachgemäßer Installation und anderem als in diesem Handbuch beschriebenen Einsatz können Störungen im Hochfrequenzbereich auftreten.

Warnung!
Dies ist eine Einrichtung der Klasse A. Diese Einrichtung kann im Wohnbereich Funkstörungen verursachen. In diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen durchzuführen.



Installation

Es dürfen nur bestimmte Varianten des PMC-8260/DS1 kombiniert werden (siehe Abschnitt “Allowed Product Combinations” auf Seite 2-8). Sonst können das PMC Modul, die Trägerkarte und über das RTB angeschlossene Komponenten beschädigt werden.

Das PMC-8260/DS1 darf ausschließlich an TNV-1 Stromkreise (Telecommunication Network Voltage-1 Stromkreise), wie in EN 60950 definiert, angeschlossen werden, da sonst das PMC Modul beschädigt wird.

Elektrostatische Entladung und unsachgemäßer Ein- und Ausbau des PMC Moduls kann Schaltkreise beschädigen oder ihre Lebensdauer verkürzen. Beachten Sie deshalb die folgenden Punkte:

- **Lesen Sie vor Ein- oder Ausbau des PMC Moduls das Kapitel “Installation” und das Installationshandbuch der betreffenden Trägerkarte.**
- **Führen Sie vor Ein- und Ausbau des PMC Moduls auf eine oder von einer CompactPCI oder VME Trägerkarte, die kein Hot Swap unterstützt, folgende Schritte durch:**
  - **Prüfen Sie, welche Schritte bei den bereits im System installierten Boards vor dem Abschalten des Stroms durchgeführt werden müssen.**
  - **Führen Sie die Schritte aus.**
  - **Schalten Sie den Strom ab.**
- **Ist mehr als ein PMC-8260/DS1 auf einem VME Board installiert, so kann dies zu einer Überlastung des Konverters führen, der die 3,3 Volt Spannungsversorgung auf dem VME Board generiert, und das Board wird beschädigt. Stellen Sie deshalb sicher, daß die 3,3 Volt Versorgung des VME Boards die in Table 4 “Current and Power Requirements” on page 2-6 beschriebenen Anforderungen einhält.**
- **Die PMC Spezifikation (P1386.1) sieht eine maximale Leistungsaufnahme von 7,5W für jedes PMC Modul vor. In der Standardkonfiguration kann das PMC-8260/DS1 diesen Wert überschreiten. Es ist deshalb notwendig, die Stromversorgung der Trägerkarte mit den Anforderungen des PMC-8260/DS1 abzustimmen. Aufgrund der hohen Leistungsaufnahme ist eine Zwangskühlung notwendig (siehe Tabelle 6 “Environmental Requirements of the PMC-8260/DS1” auf Seite 2-10).**



- **Bevor Sie Boards oder elektronische Komponenten berühren, vergewissern Sie sich, dass Sie in einem ESD-geschützten Bereich arbeiten.**
- **Stellen Sie sicher, dass die 3.3V and 5V Stromversorgungseinheiten auf der Trägerkarte gleichzeitig in höchstens 20 msec. betriebsbereit sind. Ansonsten wird das PMC Modul beschädigt.**

Betrieb

Achten Sie darauf, dass die Umgebungs- und die Leistungsanforderungen während des Betriebs eingehalten werden.

Wenn Sie das PMC Modul in Umgebungen mit elektromagnetischer Störstrahlung betreiben, stellen Sie sicher, dass die Trägerkarte mit dem CompactPCI oder VME System verschraubt ist und das System durch ein Gehäuse abgeschirmt wird.

Stellen Sie sicher, dass Anschlüsse und Kabel des PMC Moduls während des Betriebs nicht berührt werden können. Das Berühren von Kontakten kann zur Oxidation und verminderten Leitfähigkeit der Kontakte führen. Das Berühren von Spannungskontakten auf der Backplane ist gefährlich.

Betreiben Sie das PMC-8260/DS1 nur innerhalb der angegebenen Grenzwerte für die relative Luftfeuchtigkeit und Temperatur, da durch hohe Luftfeuchtigkeit und Kondensation Kurzschlüsse entstehen können. Stellen Sie vor dem Einschalten des Stroms sicher, dass sich auf dem PMC-8260/DS1 kein Kondensat befindet, und betreiben Sie das PMC-8260/DS1 nicht unter 0°C.

RJ-45 Stecker

Wenn Sie ein Telefon oder Ethernet an den RJ-45 Stecker der PMC-8260/DS1-H110-F und SC-F Varianten anschließen, wird das Telefon, das PMC Modul und das angeschlossene Gerät beschädigt. Schließen Sie deshalb nur E1/T1/J1 Netzwerke an den RJ-45 Stecker an.

Umweltschutz

Entsorgen Sie alte PMC Module gemäß der in Ihrem Land gültigen Gesetzgebung, wenn möglich umweltfreundlich.



1

Introduction

Features

The PMC-8260/DS1 is a single-size PMC (PCI mezzanine card) module based on the PowerQUICC II (MPC8260), a versatile microprocessor and peripheral controller for communication and networking applications.

Combined with a CompactPCI or VME CPU carrier board, the PMC-8260/DS1 is an ideal platform for applications in areas such as Wireless Infrastructure, Unified Messaging Servers, Enterprise Computer Telephony, Intelligent Peripherals, and Advanced Intelligent Network.

To reduce your software development efforts when enabling communication with a telecommunication management network, Force Computers offers MTP-2, LAPD, Frame Relay protocol stacks for the operating systems Solaris, VxWorks and Linux.

Revision 1.3 of the PMC-8260/DS1 provides the following improvements compared to Revision 1.2:

- New HiP4 derivate of PowerQUICC II is used which provides an increased core frequency of 300 MHz and an increased Communication Processor Module (CPM) frequency of 200 MHz.
- New bridge: PowerSPAN II (CA91L8260B) with a 32-/64-bit, 33-/66-MHz PCI interface
- RLH variant provides two additional user-programmable LEDs to show the status of all four line interfaces.
- Additional on-board 8 kHz clock source is connected to the T8105. Now, switching from a line recovered clock source to another clock source, i.e. this 8 kHz clock, is possible.
- Increased SDRAM capacity of 64 MByte

The PMC-8260/DS1 function blocks are shown in the figure below.

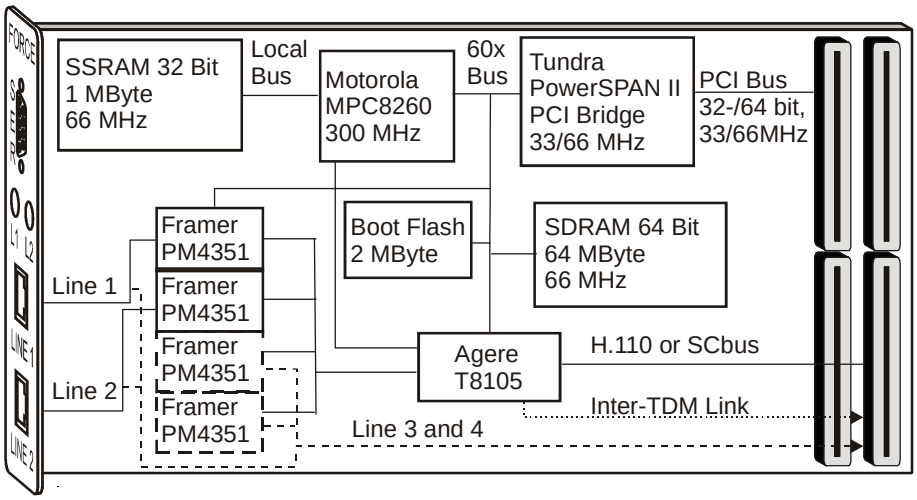


Figure 1: Function Blocks

Variants

There are three different variants of the PMC-8260/DS1:

- PMC-8260/DS1-SC-F
- PMC-8260/DS1-H110-F
- PMC-8260/DS1-H110-RLH

The letters “F” and “R” refer to the line interfaces. “F” stands for two line interfaces on the front panel, “R” stands for four line interfaces available via the rear transition board. “LH” stands for long haul support.

PMC-8260/DS1-SC-F

The PMC-8260/DS1-SC-F is tailored to be mounted on the following Force Computers carrier boards:

- PowerCore-660x (VMEbus)
- PowerCore-6750 (VMEbus)
- MFIO-120 (CompactPCI bus)

Variant Features

The PMC-8260/DS1-SC-F has the following features:

- Two line interfaces on the front panel
- SCbus specification-compliant termination and clocking
- Backwards-compatible hardware to Force Computers PMC/860...E1/T1 module

Restrictions

The PMC-8260/DS1-SC-F has the following restrictions:

- Can only be used in PMC slot 1 (due to SCbus load restrictions)
- No SCbus ID signals available
- No inter-TDM link connection

PMC-8260/DS1-H110-F and -RLH

The PMC-8260 / DS1-H110-F and -RLH are tailored to be mounted on the following Force Computers CompactPCI bus carrier boards:

- PowerCoreCPCI-6750 (Rev. 2.0 and later)
- MFIO-120 (CompactPCI bus)

Variant Features

The PMC-8260 / DS1-H110-F and -RLH modules can be used in PMC slots 1 and 2 and have the following features:

- Interfaces:
  - PMC-8260 / DS1-H110-F: One serial interface and two E1 / T1 / J1 line interfaces on front panel
  - PMC-8260 / DS1-H110-RLH: Four E1 / T1 / J1 line interfaces with FCC Part 68 approval for long haul T1 (LH) available on the rear transition board ACC / RTB-8260 / DS1. If a second -RLH module is installed on the carrier board, the ACC / RTB-8260 / DS1 offers eight E1 / T1 / J1 line interfaces.
- H.110 bus (CTbus) specification-compliant termination and redundant clocking
- PMC slot1 TDM streams may be routed to PMC slot 2 via the inter-TDM link connection or via the PCI bus.
- CT-Management bus (CT_MC) is availabe via the PowerQUICC II serial interface SCC3.

Restrictions

The PMC-8260 / DS1-H110-F and -RLH variants have the following restrictions:

- Front panel line interfaces of PMC-8260 / DS1-H110-F variant only support short haul.
- On the carrier boards, only PMC slot 2 is connected to the H.110 bus (CompactPCI P4 connector).

Standard Compliances

The PMC-8260/DS1 meets the following standard requirements:

Table 1: *Standard Compliances*

Standard	Description
EN 55024 EN 55022 FCC Part 15	EMC requirements on system level
EN 60950 UL 60950	Legal safety requirements
FCC Part 68 (T1-100 Ohm) ITU G.703/704, G.823 TBR4 (E1-120 Ohm) Industry of Canada CS-03 (PMC-8260/DS1-H110-RLH variant)	Telecom requirements
IEC 68-2-1/2/3/13/14	Climatic environmental requirements. The PMC-8260/DS1 can only be used in a restricted temperature range (see the "Environmental Requirements" section on page 2-4 for details).
IEC 68-2-6/27/32	Mechanical environmental requirements
ANSI/IPC-A-610 Rev.C Class 2, ANSI/IPC-7711, ANSI/IPC-7721, ANSI-J-001...003	Manufacturing Requirements

Ordering Information

The PMC-8260/DS1 has been designed to cost-effectively support a wide range of hardware requirements. Therefore, a number of factory options are available:

- Line interfaces:
  - Two line interfaces on front panel
  - Four line interfaces via rear transition board (RTB) ACC/RTB-8260/DS1
- SC bus or H.110 bus support

Product Nomenclature

In the following you find the key for the product name extensions.

PMC-8260/DS1-uuuu-vvv/ccc-xx-y	
uuuu	Bus support: H.110: H.110 bus backplane TDM bus SC: SCbus backplane TDM bus
vvv	Interface location: F: Two line interfaces on front panel RLH: Four line interfaces via PMC-IO on rear transition board and long-haul support
ccc	CPU clock frequency in MHz: 300: 300 MHz core frequency and 200 MHz Communication Processor Module (CPM) frequency
xx	SDRAM capacity in MByte: 64: 64 MByte SDRAM
y	SSRAM capacity in MByte: 1: 1 MByte SSRAM

Order Numbers

The variant of the PMC-8260/DS1 to be used depends on the carrier board (see the "Allowed Product Combinations" section on page 2-8). Consult your local sales representative to confirm availability of specific combinations.

Table 2: *Ordering Information*

Order No.	PMC-8260/DS1-...	Description
110695	...SC-F/300-64-1	PowerQUICC-II based PMC module, 300 MHz, two software-selectable E1/T1/J1 interfaces on PMC front panel, SCbus interface, 64 MByte SDRAM, 1 MByte SSRAM, 2 MByte user flash, for selected VME/CompactPCI carriers use only (incl. VME PowerCore-6603/4 (lower slot only))
110690	...H110-F/300-64-1	PowerQUICC-II based PMC module, 300 MHz, two software-selectable E1/T1/J1 interfaces on PMC front panel, H.110 bus interface, 64 MByte SDRAM, 1 MByte SSRAM, 2 MByte user flash, for selected CompactPCI carriers use only (incl. Force CPCI-6750)
110694	...H110-RLH/300-64-1	PowerQUICC-II based PMC module, 300 MHz, four software-selectable E1/T1/J1 interfaces on PMC user I/O, H.110 bus interface, 64 MByte SDRAM, 1 MByte SSRAM, 2 MByte user flash, for selected CompactPCI carriers use only (incl. Force CPCI-6750)
Hardware Accessories PMC-8260/DS1		
107338	ACC/RTB-8260/DS1	RTB with eight E1/T1/J1 interfaces and debug port for Force CompactPCI carrier boards equipped with Force PMC-8260/DS1-H110-RLH
102189	ACC/CABLE/MD9-D9/2	Serial cable for 9-pin micro D-Sub to 9-pin D-Sub
Software Accessories PMC-8260/DS1		
107405	Tornado 2.0/BSP PMC-8260	Tornado 2.0 Board Support Package (BSP) for PMC-8260/DS1

Table 2: *Ordering Information (cont.)*

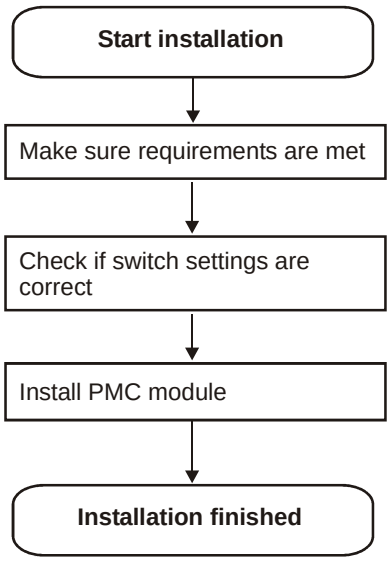
Order No.	PMC-8260/DS1-...	Description
106881	SW / STACK / MTP2 / R4	Force MTP-2 protocol software package including control, configuration and test functions, Solaris 2.6, 2.7 and 8 (64-bit mode), VxWorks 5.4 / Tornado 2.0 and Linux 2.4.x driver, RTU license
106446	SW / STACK / FR / R4	Force Frame Relay protocol software package including control, configuration and test functions, Solaris 2.6, 2.7 and 8 (64-bit mode), VxWorks 5.4 / Tornado 2.0 and Linux 2.4.x driver, RTU license
106449	SW / STACK / LAPD / R4	Force LAPD protocol software package including control, configuration and test functions, Solaris 2.6, 2.7 and 8 (64-bit mode), VxWorks 5.4 / Tornado 2.0 and Linux 2.4.x driver, RTU license

2

Installation

Action Plan

In order to install the PMC module, the following steps are necessary and are described in detail in the sections of this chapter.



Requirements

The installation requires:

- Backplane according to used carrier board:
  - CompactPCI carrier board: CompactPCI (optional CompactPCI-H.110) backplane system with power supply
  - VMEbus carrier board: VMEbus (optional VMEbus-SCbus) backplane system with power supply
- Corresponding carrier board with at least one PMC slot

Long Haul Requirements

The two front-panel line interfaces in T1 mode on the SC-F and H.110-F variants are DSX-1 type interfaces which means that in T1 mode, the interfaces do not support long haul required for public networks. To be able to operate the PMC-8260/DS1 in a public network you need a CSU (channel service unit).

The line interfaces available on the backplane via PMC-IO on the H.110-RLH variant can be configured via the framers to support long haul T1 mode if the ACC/RTB8260/DS1 is used. With this special RTB external CSUs are no longer required. For information on how to configure the framers for long haul T1, refer to the PMC-Sierra PM4351 documentation.

Carrier Board Signaling Level Requirements

The PMC-8260/DS1 supports signaling levels 3.3V and 5V and therefore is a universal PMC module. Make sure that your carrier board has a signaling level of 3.3V or 5V at the PMC interface.

Environmental Requirements

The environmental values given in the table below must be tested and proven in the used system configuration. Operating temperatures refer to the temperature of the air circulating around the board and not to the actual component temperature.

Note: To ensure that the operating conditions are met, forced air cooling is required within the chassis environment.

Caution



Do not operate the product outside the specified environmental limits. High humidity and condensation may cause short circuits. Make sure the product is completely dry and there is no condensation of water on any parts of the board's surface before applying power. Do not operate the product below 0°C.

Table 3: *Environmental Requirements of the PMC-8260/DS1*

Feature	Operating	Non-Operating
Temperature	0°C to +55°C	- 40°C to +85°C
Temp. Change	+/-0.5°C per minute	+/-1.0°C per minute
Forced Airflow	300 LFM (linear feet per minute)	-
Rel. Humidity	5% to 95% at +40°C	5% to 95% at +40°C
Altitude	-300 m to +3,000 m	-300 m to +13,000 m
Vibration 10 to 15 Hz 15 to 150 Hz	 2 mm amplitude 2 g	 5 mm amplitude 5 g
Shock	5 g/11 ms halfsine	15 g/11 ms halfsine
Free Fall	100 mm/3 axis	1200 mm/all edges and corners (packed state)

Power Requirements

The power supply must meet the power requirements of the carrier board and the PMC-8260/DS1 module. For the power requirements of the carrier board refer to the respective Installation Guide.

Caution



- **The PMC module carrier board must guarantee that the 3.3V and 5V power supplies ramp up simultaneously in about max. 20 msec. Otherwise, the PMC module may be damaged.**
- **The power supply circuits on VMEbus carrier boards may be overloaded if more than one PMC-8260/DS1 is assembled. This results in permanent damage to the carrier board. Therefore, make sure that the VMEbus carrier board's +3.3 Volt supply supports the power requirements described in this section.**

- **The PMC specification (P1386.1) specifies a maximum total power consumption of 7.5W per PMC module. The PMC-8260/DS1 in its standard configuration may exceed this value. Therefore, it is neccessary to synchronize the carrier board's power supply with the PMC-8260/DS1 power requirements. Due to the high power dissipation forced air cooling is necessary (see also forced airflow requirement in Table 3 "Environmental Requirements of the PMC-8260/DS1" on page 2-5).**

The current and power values for one PMC module can be seen in the table below.

Table 4: *Current and Power Requirements*

	+5V	+3.3V	Power Dissipation
Typical	0.6A	1.1A	6.63W
Maximum	0.8A	1.3A	8.3W

FCC Requirements for PMC-8260/DS1

This equipment complies with Part 68 of the FCC rules. On the equipment is a label that contains, among other information, the FCC registration for this equipment. If requested, provide information to your telephone company.

If your PMC-8260/DS1 causes harm to the telephone network, the telephone company may discontinue your service temporarily. If possible, they will notify you in advance. But if advance notice is not practical, you will be notified as soon as possible. You will be advised of your right to file a complaint with the FCC.

Your telephone company may make changes in its facilities, equipment, operations, or procedures that could affect the proper operation of your equipment. If they do, you will be given advance notice so as to give you an opportunity to maintain uninterrupted service.

If you experience trouble with the PMC-8260/DS1, contact your local Force Computers representative for repair and warranty information. The telephone company may ask you to disconnect this equipment from the network until the problem has been corrected or you are sure that the equipment is not malfunctioning.

Never repair the PMC-8260/DS1 yourself. Otherwise, you may lose the FCC approval.

Switch Settings

The PMC-8260/DS1 is configurable via switch SW1.

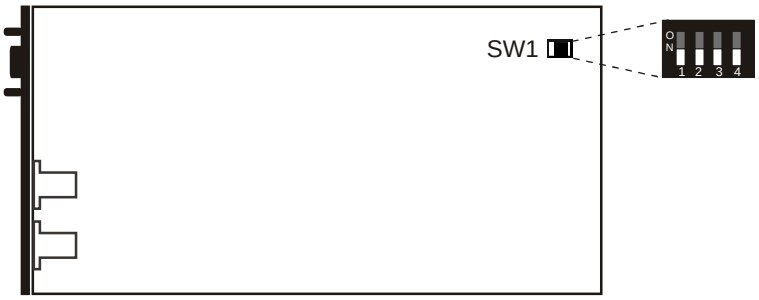


Figure 2: *Location of Switch*

Note:

- **In normal operation mode SW1-1 must be in OFF position. If switch 1-1 is ON, the PMC module does not boot.**
- **Always put SW1-2 in OFF position after programming a firmware update into the flash from PCI bus. Otherwise, the board does not boot.**

Table 5: *Default Switch Settings for SW1*

Switch	Number	Description
SW1 	1	PowerSpan power-up option boot from 60x/PCI: OFF (default): Normal operation ON: Boot from PCI (Lockout bit cleared)
	2	Flash programming via PCI bus: OFF (default): Normal operation ON: Enable boot Flash programming via PCI bus
	3	**Reserved, must be OFF**
	4	Enable MPC8260 COP interface (connector P4) OFF (default): Normal operation ON: COP interface emulator can be attached to P4 (see the "JTAG/Debug" section on page 3-11)

Allowed Product Combinations

The following tables summarize the allowed product combinations. The PMC module names are shortened, for example “SC-F” instead of “PMC-8260/DS1-SC-F”.

Table 6: *PMC-8260/DS1 on Reference Platforms*

Carrier Board	Slot 1	Slot 2	Backplane-TDM Bus (PMC Slot)	Line Inter-faces	Hard-ware TDM-Mux (T8105)	Inter-TDM Link
Power-Core-660x	SC-F	-	H.110/SCbus interoperability mode (#1)	2	Slot 1: yes Slot 2: no	No
Power-Core-6750	SC-F	-	H.110/SCbus interoperability mode (#1)	2	Slot 1: yes Slot 2: no	No
Power-CoreCPCI 6750 (Rev 2.0 and later)[1)]	H.110-F and -RLH	H.110-F and -RLH	H.110 bus (#2)	2 - 4	Slot 1: yes Slot 2: yes	Yes
MFIO-120	H.110-F and -RLH	H.110-F and -RLH	H.110 bus (#2)	2 - 4	Slot 1: yes Slot 2: yes	Yes

1) PowerCoreCPCI-6750 Rev. 2.x supports two PMC modules only if it has a certain PCB revision. For further information, see Table 7 “Revisions Supported by Reference Platform” on page 2-9

Note: As described in the table above, the reference platform PowerCoreCPCI-6750 Rev.2.x supports two PMC-8260/DS1 modules, depending on the hardware revision. The supported hardware revisions are listed in the following table. Hardware revisions earlier than those listed below support only one PMC-8260/DS1. Therefore, if you want to install two PMC modules, make sure to order the CPCI-6750 Rev. 2.x with the minimum PCA revision level given in the table.

Table 7: *Revisions Supported by Reference Platform*

PCA Revision	PCA Product Number and Name	Official Order Number
Rev. L	106347 PCA PowercoreCPCI-6750/0-233-L512-4/R.2	106476 PPC/PowerCoreCPCI-6750/64-233-L512-4 R2
Rev. M	106349 PCA PowercoreCPCI-6750/0-400-L1024-8R.2	106477 PPC/PowerCoreCPCI-6750/128S-400-L1M-8 R2
		106478 PPC/PowerCoreCPCI-6750/64S-400-L1M-8 R2
Rev. H	107398 PCA PowercoreCPCI-6750/0-500-L1M-8/2.0	107344 PPC/PowerCoreCPCI-6750/128S-500-L1M-8R2

Installing a PMC Module

The PMC-8260/DS1 supports a data bus width of 32/64 bit with a frequency of 33 or 66 MHz. The supported signaling levels are 3.3V and 5V, therefore, the PMC-8260/DS1 can be installed on carrier boards with a signaling level of 3.3V or 5V at their PMC interface.

The front panel of the carrier board provides one or more cutouts to enable the installation of PMC modules. If the board is shipped without modules installed, the front panel cutouts are covered by blind panels to ensure proper EMC shielding.

Caution



- **Be sure to mount only allowed combinations of PMC-8260/DS1 variants (see the "Allowed Product Combinations" section on page 2-8). Otherwise damage to PMC module, carrier board and to equipment attached to the rear transition board may occur.**
- **The PMC-8260/DS1 may only be connected to TNV-1 circuits (Telecommunication Network Voltage-1 circuits) defined in EN 60950. Otherwise, the PMC module will be damaged.**
- **Before touching modules or electronic components, make sure that you are working in an ESD-safe environment.**

Installation Procedure

You can install the PMC-8260/DS1 on carrier boards with or without hot-swap support. If you install it on a hot-swappable carrier board, start installation procedure with step 4. If you install it on a non-hot-swap carrier board, start with step 1.

Note: Before installing the PMC module, check which PMC slot(s) is/are supported on the used carrier board (see see the "Allowed Product Combinations" section on page 2-8).

1. Check all boards installed in system for steps to be taken before turning off power.
2. Take those steps
3. Turn off power
4. Remove carrier board from system according to carrier board's Installation Guide if it is already installed in system
5. Remove blind panel of respective PMC slot from carrier board's front panel
6. Store blind panel in a safe place
7. Plug PMC module into connectors

8. Check whether standoffs of module cover mounting holes of carrier board

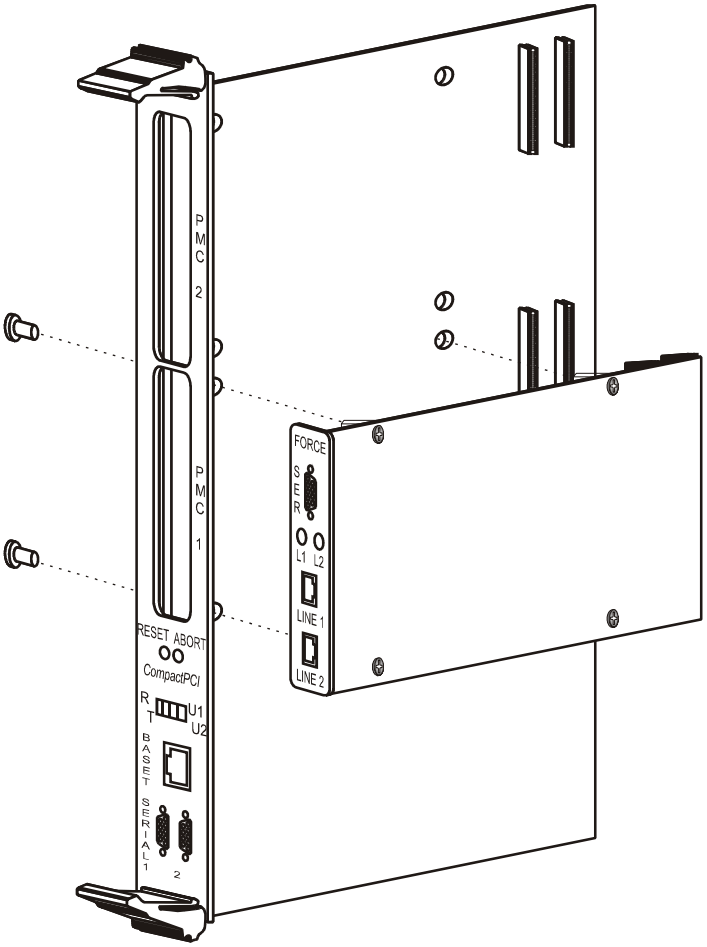


Figure 3: *Installation (Example with PowerCoreCPCI-6750 Rev. 2.x)*

9. Fasten four screws delivered with PMC module
10. Install carrier board according to carrier board's Installation Guide
11. Attach interface cables
12. If not installing a hot-swappable carrier board, switch power on

Removal Procedure

If you remove the PMC module from a hot-swappable carrier board, start installation procedure with step 4. If you remove it from a non-hot-swap carrier board, start procedure with step 1.

1. Check all boards installed in system for steps to be taken before turning off power
2. Take those steps
3. Switch off system
4. Remove carier board from system according to carrier board's Installation Guide
5. Remove screws holding respective PMC module
6. Disconnect PMC module carefully from slot
7. Close front panel gap at free slot with blind panel

Software Upgrades and Accessories

Force Computers provides software packages with control, configuration and test functions for the layer-2 protocols MTP-2, LAPD, and Frame Relay. Support is available for the operating systems Solaris 2.6, 2.7 and 8, VxWorks 5.4/Tornado 2.0 and Linux Kernel 2.4.x.

For possible combinations of protocol, operating system and carrier board, ask your local Force Computers sales representative.

For installation information, refer to the respective Installation Guide and for using the protocol functions, refer to the respective Programmer's Guide.

3

Indicators and Connectors

Front Panel

The interface connectors and LEDs available on the different variants can be seen in the following figure.

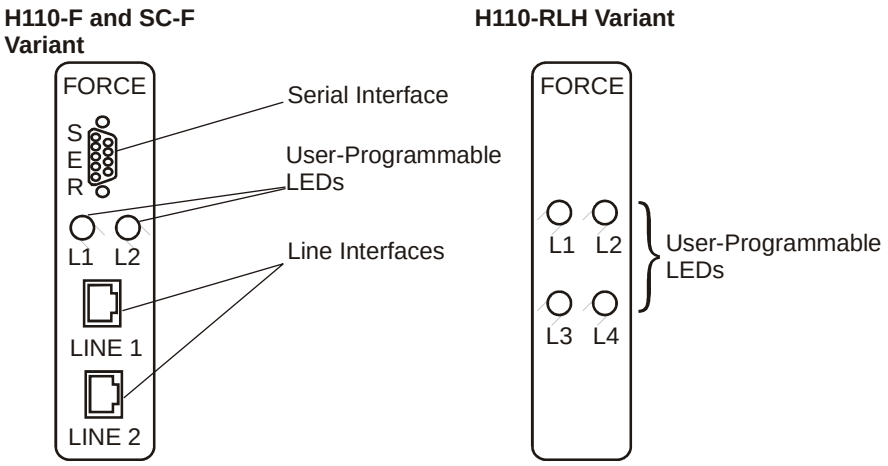


Figure 4: *Front Panels*

Connectors

The front panel of the H110_F and SC-F variant provides one serial connector and two line interface connectors.

Serial Connector

The PMC-8260/DS1-H110-F and PMC-8260/DS1-SC-F variants offer a serial interface on the front panel. The following connector pinout may be useful to give information on which signal is assigned to which pin.

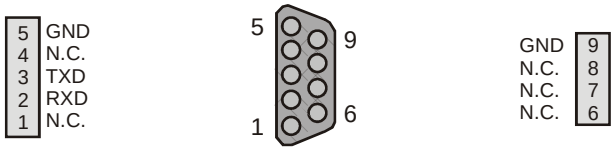


Figure 5: *Front Panel Serial I/O Port Connector Pinout*

The following table shows the availability of serial I/O channels.

Table 8: *Availablility of Serial I/O Channels*

I/O Channel	MicroD-Sub on Front Panel	PMC User I/O	SMD JTAG/Debug (P4)
PowerQUICC II SCC_1 (UART)	RS-232 (only on H110-F and SC-F variants)	RS-232 (only on H110-RLH variant; assembly option for H110-F and SC-F variants)	-
PowerQUICC II SMC_1 (UART)	-	-	RS-232
PowerQUICC II SCC_2 (ETH)	SCC2-Ethernet mode (assembly option instead of RS-232) Needs external 10BaseT-PHY	-	-
PowerQUICC II SCC_3 (HDLC)	Only available on board and used for CT-Bus Management Bus (CT_MC) SCC3 is clocked by T8105 clock output L_SC<0>=T8105_MC_CLK signal=2.048 MHz		
PowerQUICC II SCC_4 and SMC_2	SCC_4 and SMC_2 are not connected to any external devices.		

Line Interface Connector

Caution



Connecting a telephone or Ethernet to the RJ-45 connector of the PMC-8260/DS1-H110-F and SC-F variants causes damage to the telephone, the PMC module and to attached devices. Therefore, only connect E1/T1/J1 lines to the RJ-45 connector.

The following connector pinout may be useful to give information on which signal is assigned to which pin.

1	Rx- (Ring/Input)
2	Rx+ (Tip/Input)
3	Reserved
4	Tx- (Ring/Output)
5	Tx+ (Tip/Output)
6	Reserved
7	Reserved
8	Reserved



Figure 6: *Front Panel RJ-45 Interface Connector Pinout*

LEDs

All variants offer two user-programmable LEDs L1 and L2 (for location see Figure 4 “Front Panels” on page 3-3) which can take the states “off”, “red”, “green” and “yellow”. At power up, they show the state of the power-up sequence. LEDs 3 and 4 on the RLH variant do not indicate anything during power-up. Furthermore, you can control the status of the LEDs with the LED Control Register (see the “LED Control Register” section on page 5-24).

The following table shows the LED settings during power up:

Table 9: *LED States During Power Up*

State	LED 1	LED 2
Power up / reset (hardware setting)	Yellow	Yellow
Before memory controller setup	Red	Red
After memory controller setup	Yellow	Off

Table 9: *LED States During Power Up (cont.)*

State		LED 1	LED 2
POST running	SDRAM test running	Yellow flashing	Off
	SSRAM test running	Off	Yellow flashing
	PowerSpan test running	Yellow	Off
	Framer #1 test running	Off	Yellow
	Framer #2 test running	Yellow	Off
	Framer #3 test running	Off	Yellow
	Framer #4 test running	Yellow	Off
	T8105 test running	Off	Yellow
	POST complete	Yellow	Green
Executing ROM image (end of boot sequence)		Green	Off
Primary booter	Idle loop	Red	Green (dimmed flashing)
	Flash erase / write operation ongoing	Red	Yellow (flashing)
	Before execution of uploaded image	Green	Off

On-Board Connectors

Apart from the three standard PMC connectors, the PMC-8260/DS1 offers the PMC I/O and the JTAG/Debug connector. The figure below shows the location of these two connectors.

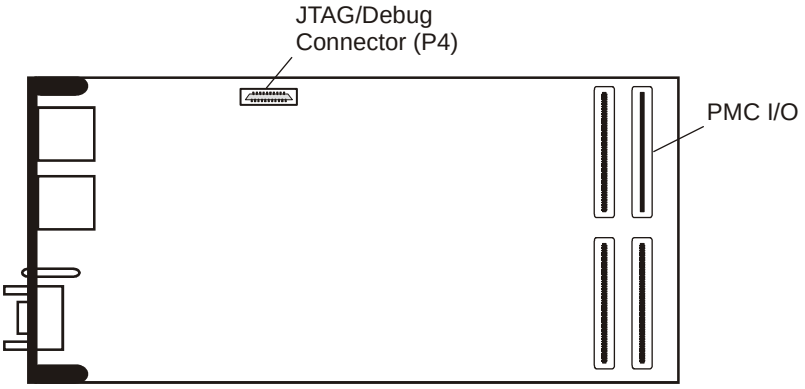


Figure 7: *PMC I/O and JTAG/Debug Connector*

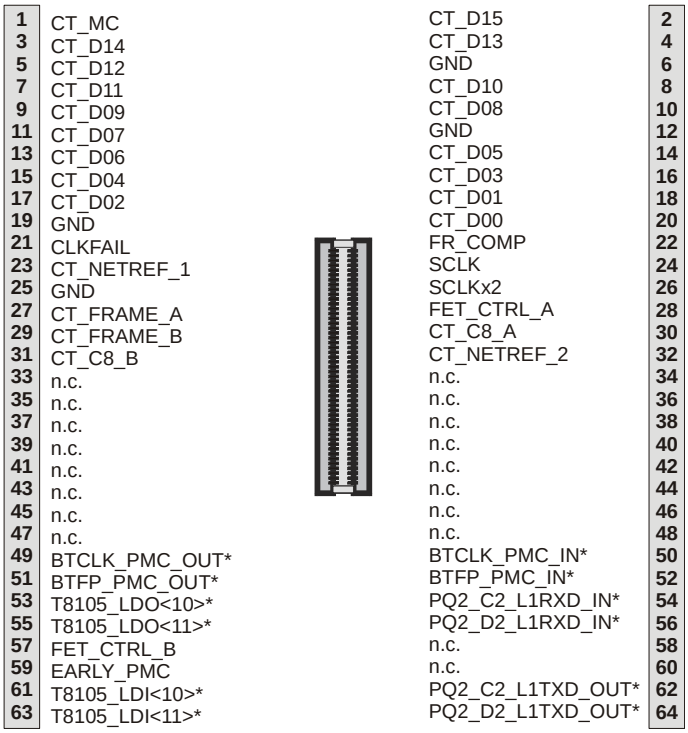
PMC I/O

The following figures show the PMC I/O connector pinouts which differ from variant to variant.

1	SC_MC	SC_D15	2
3	SC_D14	SC_D13	4
5	SC_D12	GND	6
7	SC_D11	SC_D10	8
9	SC_D09	SC_D08	10
11	SC_D07	GND	12
13	SC_D06	SC_D05	14
15	SC_D04	SC_D03	16
17	SC_D02	SC_D01	18
19	GND	SC_D00	20
21	SC_CLKFAIL	SC_FR_COMP	22
23	SC_REF8K	SCLK	24
25	GND	SCLKx2	26
27	n.c.	n.c.	28
29	n.c.	n.c.	30
31	n.c.	n.c.	32
33	n.c.	n.c.	34
35	n.c.	n.c.	36
37	n.c.	n.c.	38
39	n.c.	n.c.	40
41	n.c.	n.c.	42
43	n.c.	n.c.	44
45	n.c.	n.c.	46
47	n.c.	n.c.	48
49	n.c.	n.c.	50
51	n.c.	n.c.	52
53	n.c.	n.c.	54
55	n.c.	n.c.	56
57	n.c.	n.c.	58
59	n.c.	n.c.	60
61	n.c.	n.c.	62
63	n.c.	n.c.	64

Figure 8: *PMC I/O Connector Pinout for PMC-8260/DS1-SC-F*

Note: Pins 27, 29, 30, 31, and 32 available on the PMC/860...E1/T1 (SCBus ID pins) are not available on the PMC-8260/DS1. Otherwise pin compatibility is given between PMC/860...E1/T1 and PMC-8260/DS1.



1	CT_MC	CT_D15	2
3	CT_D14	CT_D13	4
5	CT_D12	GND	6
7	CT_D11	CT_D10	8
9	CT_D09	CT_D08	10
11	CT_D07	GND	12
13	CT_D06	CT_D05	14
15	CT_D04	CT_D03	16
17	CT_D02	CT_D01	18
19	GND	CT_D00	20
21	CLKFAIL	FR_COMP	22
23	CT_NETREF_1	SCLK	24
25	GND	SCLKx2	26
27	CT_FRAME_A	FET_CTRL_A	28
29	CT_FRAME_B	CT_C8_A	30
31	CT_C8_B	CT_NETREF_2	32
33	n.c.	n.c.	34
35	n.c.	n.c.	36
37	n.c.	n.c.	38
39	n.c.	n.c.	40
41	n.c.	n.c.	42
43	n.c.	n.c.	44
45	n.c.	n.c.	46
47	n.c.	n.c.	48
49	BTCLK_PMC_OUT*	BTCLK_PMC_IN*	50
51	BTFP_PMC_OUT*	BTFP_PMC_IN*	52
53	T8105_LDO<10>*	PQ2_C2_L1RXD_IN*	54
55	T8105_LDO<11>*	PQ2_D2_L1RXD_IN*	56
57	FET_CTRL_B	n.c.	58
59	EARLY_PMC	n.c.	60
61	T8105_LDI<10>*	PQ2_C2_L1TXD_OUT*	62
63	T8105_LDI<11>*	PQ2_D2_L1TXD_OUT*	64

Figure 9: *PMC I/O Connector Pinout for PMC-8260/DS1-H110-F*

Note: The signals marked by an asterisk are inter-TDM link signals.

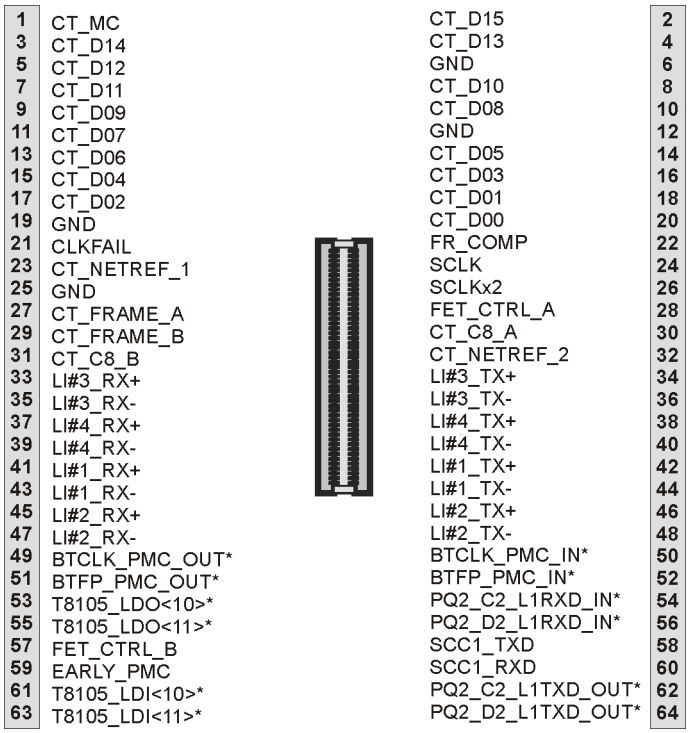


Figure 10: *PMC I/O Connector Pinout for PMC-8260/DS1-H.110-RLH*

Note: On the MFIO-120 and the PPC/PowerCoreCPCI-6750 Rev 2.0 and later, the signals on pins 49, 51, 53, 55, 61, 63 marked by an asterisk are cross-connected to pins 50, 52, 54,56, 62, 64. This cross-connection realizes the inter-TDM link between PMC slots 1 and 2.

JTAG/Debug

The following figure shows the location of the JTAG / Debug connector (P4).

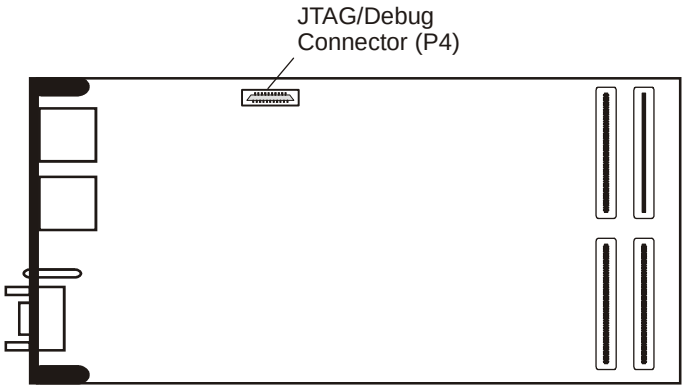


Figure 11: *JTAG/Debug Connector*

Note: The COP interface is not available on revisions prior to Revision 1.2.

The JTAG / Debug connector provides the signals of the PowerQUICC II COP interface port which allows easy connection of a Standard Motorola Third Party COP-Interface Emulator. Additionally, it provides a second RS-232 interface for debugging convenience (PowerQUICC II SMC1-interface) and the necessary supply voltages.

The COP-interface port can be used for:

- Program flow tracking
- Internal watch point and breakpoint generation
- Emulation system control over PowerQUICC II

The RXD / TXD pins of the PowerQUICC II SMC_1 are connected to the development port as well as the debug mode connector (Pins 8 and 9) via RS-232 drivers.

Note: To enable the COP interface, SW1-4 must be set to ON position.

The JTAG / Debug connector is a two-row 21-pin SMD connector (vertical receptible of the type BERG CONAN™ 91931-WXY21) with 1 mm pin-pitch located on the bottom side of the PMC-8260 / DS1 module (see Figure 11 “JTAG / Debug Connector” on page 3-11).

The following figure shows the pinout of the JTAG / Debug mode connector.

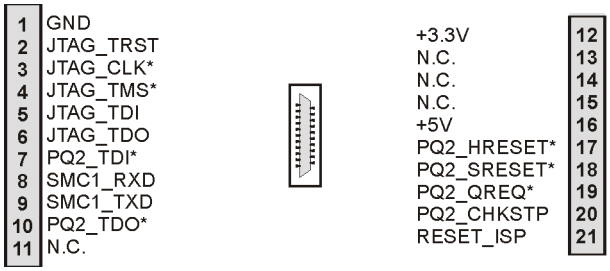


Figure 12: *JTAG/Debug Connector Pinout*

Note: The signals marked by an asterisk are COP signals.

4

Firmware

Introduction

Force Computers provides a firmware with the PMC-8260/DS1 which is executed immediately after power up. It performs a power-on self-test (POST) and a basic hardware initialization so that all devices are accessible. If you have created a driver for the host board with the help of this chapter, it furthermore enables communication between the PMC module and the carrier board.

Code examples are given in the "Code Examples" section on page 4-30.

Features

The firmware offers a number of features:

- Execute POST and store POST results in the POST result storage area
- Execute application software from boot flash device
- Start primary booter to:
  - Load application software via PCI bus
  - Program application software into the boot flash device
  - Read from and write to all devices on the PMC-8260/DS1
  - Obtain detailed results from the power-on self-test to evaluate error conditions
- Controlling and monitoring the boot process

Resources Used by the Firmware

The PMC8260/DS1 firmware uses the following resources:

- Firmware is stored in the boot flash at offset 100_{16}, and occupies the first 256 KByte. This leaves 1.75 MByte for application usage.
- Mailbox registers 5, 6 and 7 of the PowerSpan II PCI bridge as control and status registers.

Note: In order to avoid firmware malfunction, do not use the mailbox registers 5, 6 and 7 of the PowerSpan II since they are used by the firmware.

Power-Up Sequence

At power up, the CPU starts firmware code execution at address 100_{16} and the watchdog timer and the bus monitor are enabled to prevent a reset or a system hang. One of the first things it does is to map the boot flash device to address $FE000000_{16}$ and to continue code execution there. The following flow chart shows an overview of the steps performed by the firmware and also the steps which can be controlled by a driver.

Note: The PMC8260/DS1 firmware does not display any banner or prompt on a terminal connected to the serial debug ports. The firmware has no shell, therefore it cannot accept commands from a terminal. Only the front panel LEDs show the current status of the boot process, the power-on self-test and the primary booter (see Table 9 “LED States During Power Up” page 3-5).

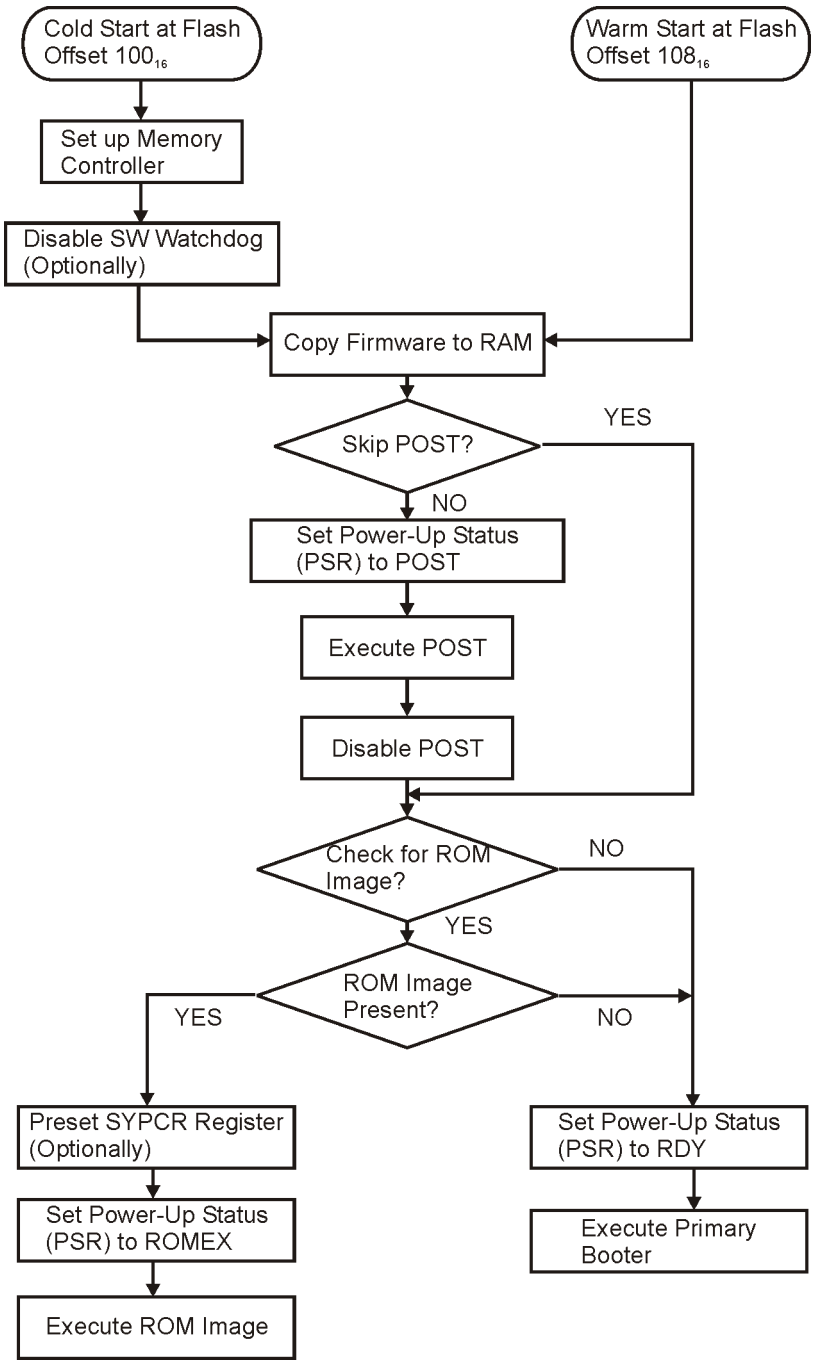


Figure 13: *Flowchart of the PMC-8260/DS1 Firmware Power-Up Sequence*

The steps shown in the figure above are described in the table below.

Table 10: *Steps Performed During Power-Up*

Step	Description
Cold start at flash offset 100_{16}	The firmware starts code execution at 100_{16} after a hardware power up or a reset. At this point the hardware is not initialized.
Set up memory controller	The firmware initializes the PowerQUICC II memory controller and maps the devices to the local addresses as shown in see Table 22 “Memory Map” page 5-4.
Disable SW Watchdog (optionally)	When the firmware starts execution, the watchdog is enabled. It can be disabled by setting the WDDIS bit in the power-up control register (PCR). For further information, see the “Controlling Boot Process” section on page 4-13.
Warm start at flash offset 108_{16}	The firmware starts code execution at 108_{16} after a software reset. Since the hardware was already initialized after the cold start, the previous two steps are omitted.
Skip POST?	The POST can be enabled / disabled by setting the PSKIP bit in the PCR register. Another way to influence the POST execution is to program the byte at flash offset $000400F1_{16}$ (see the “Controlling Boot Process” section on page 4-13).
Set power-up status (PSR) to POST	Before the power-on self-test is started, the value 1 is written into the power-up status register (PSR). This status bit can be used to poll the power-up status.
Execute POST	The power-on self-test (POST) provides basic tests for SSRAM, SDRAM, PCI bridge and the I/O devices (framers, H.110 interface). The tests ensure proper connectivity of devices on the board. However, no extensive functional tests are performed. Since the POST does not halt if it detects errors your application software should evaluate the POST results written into mailbox register 5 of the PowerSpan II as soon as possible (see the “Obtaining Results from the Power-On Self-Test” section on page 4-24).
Disable POST	If the POST is completed, the firmware sets the PSKIP bit in the PCR register so that the POST is not executed again after a warm start.

Table 10: *Steps Performed During Power-Up*

Step	Description
Check for ROM image?	The firmware can be configured to check for a ROM image in the boot flash device or to start the primary booter. This is done by setting the ROMSKIP bit in the PCR register (see Table 12 “PCR Register Settings” page 4-13).
ROM image present?	The firmware checks whether a ROM image is present at address $FE040108_{16}$.
Preset SYPCR register (optionally)	At power up, the watchdog and the bus monitor are enabled. You can configure your ROM image to make the firmware change the System Protection Control register (SYPCR) settings and to disable the watchdog and the bus monitor. This is done by programming the ROM image accordingly. For further information, see section “SYPCR Register” on page 4-9.
Set power-up status (PSR) to ROMEX	Before the ROM image is executed, the value 3 is written into the PSR. This information can be used to poll the power-up status.
Set power-up status (PSR) to RDY	Before the primary booter is started, the value 2 is written into the PSR. This status bit can be used to poll the power-up status.

Loading Application Software

You can either load software from:

a) PMC module's boot flash device by programming a ROM image of max 1.75 MByte. The ROM image is copied into the boot flash device via the primary booter.

b) Carrier board via the PCI bus and primary booter

Requirements for Both Loading Methods

Both loading procedures require the following:

- Host system with operating system
- Host driver which enables communication between the PMC module and the carrier board. The driver has to include programming routines at least for the following:
  - Controlling the boot process
  - Monitoring the boot process
  - Starting the primary booter
    Necessary to enable communication between PMC module and host when loading an image from PCI bus, programming a ROM image into boot flash, performing read / write accesses to I / O devices (e.g. reading the boot flash contents or POST results).
  - Programming the boot image into boot flash (not necessary if you want to load an image via PCI bus)
  - Reading and writing to I / O devices
  - Optional: Reading POST results

If you use the Tornado 2.0 BSP for the PMC-8260 / DS1 such a driver is already implemented. For more information on the BSP, refer to the BSP documentation.

In the following sections we give you the information required to create such a driver. However, we cannot give you information on how to implement the driver into your operating system because the procedure varies depending on the used operating system. At the end of this chapter, code examples are given to show how the routines can be programmed.

Preparing a ROM Image

The ROM image of max. 1.75 MByte must be programmed into the boot flash device beginning at an offset of 256 KByte into the boot flash device which corresponds to an effective address of $FE040000_{16}$. The firmware considers a ROM image to be valid and executes it only if one of the following conditions apply:

- Checksum information was programmed into the ROM configuration section and the checksum is valid. For further information, see descriptions for CSUM_ID, IMG_SIZE, and IMG_CSUM in Table 11 “ROM Image Configuration Section” on page 4-10.
- No checksum information is present and the 32-bit value at ROM image offset 108_{16} differs from $FFFF.FFFF_{16}$.

SYPCR Register

By default, the firmware will not alter the contents of the System Protection and Control (SYPCR) register of the PowerQUICCC II at power up, i.e. leaving the bus monitor and the software watchdog enabled. A problem arises when the firmware detects and executes a ROM image which does not provide support for the software watchdog, i.e. the watchdog is not triggered and the board is then reset.

To be able to install ROM images without watchdog support, the firmware provides a mechanism to preset the SYPCR register for a ROM image’s purpose before executing it. The firmware presets the SYPCR register if you program the fields SYPCR_WR and SYPCR_VAL of the ROM configuration section. For further information, see SYPCR_WR and SYPCR_VAL descriptions in Table 11 “ROM Image Configuration Section” on page 4-10.

ROM Image Configuration

The first 256 bytes of a ROM image are considered as control section where you can define how the firmware must handle the image. You can, for example, define the SYPCR register value, POST execution speed and whether POST is executed or not. The following table shows the offsets and values to be programmed.

Table 11: *ROM Image Configuration Section*

Offset	Bytes	Mnemonic	Description
$E0_{16}$	1	CSUM_ID	If this byte is set to $C5_{16}$ the checksum information in the subsequent fields IMG_SIZE and IMG_CSUM is considered valid. Otherwise, no checksum test is performed.
$E4_{16}$	4	IMG_SIZE	The size of the ROM image in bytes (without the 256 bytes configuration section). Values of $FFFF.FFFF_{16}$ and 0 are considered invalid. This parameter must be set for checksum calculation and the shadow feature.
$E8_{16}$	4	IMG_CSUM	Checksum of the ROM image The valid 32-bit checksum of a ROM image is calculated by adding IMG_SIZE bytes beginning at ROM image offset 100_{16}. To perform the checksum test, CSUM_ID must be set to $C5_{16}$.
$F0_{16}$	1	SYPCR_WR	If set to 0, the firmware will preset the SYPCR register of the PowerQUICC II with SYPCR_VAL (see below) before executing the ROM image. Otherwise the SYPCR register is left in its reset state.
$F1_{16}$	1	POST_SKIP	If set to 0, no POST will be executed by the firmware. If set to FF_{16} POST execution depends on the setting of the PCR[PSKIP] bit (see Table 12 “PCR Register Settings” on page 4-13).
$F2_{16}$	1	FLAGS	Various flags Bit / Mnemonic / Description 0 / FPOST / POST execution 0: Normal POST 1: Fast POST 1 / SHADOW / RAM shadow 0: No shadow to RAM is executed 1: Shadow to RAM is executed 2 / SMODE / Shadow incl. configuration section 0: Shadow without configuration section 1: Shadow with configuration section

Table 11: *ROM Image Configuration Section (cont.)*

Offset	Bytes	Mnemonic	Description
$F4_{16}$	4	SYPCR_VAL	Enter the value to be programmed in the SYPCR register of the PowerQUICC II before the ROM image is executed. For this purpose, SYPCR_WR must be set to 0. For register settings, see the *MPC8260 PowerQUICC II User's Manual*.
$F8_{16}$	4	RAM_ADRS	Address to which the ROM image is copied to, if FLAGS:SHADOW is set to 0.
FC_{16}	4	RAM_EXEC	Address at which the firmware has to begin code execution.

Argument Passed to ROM Image

The firmware passes a single argument to the ROM image in register r3. The value depends on the start type:

- When performing a cold start at power up, the firmware passes a value of 2 to the ROM image.
- When performing a warm start, the firmware preserves the register r3 and uses it as argument to the ROM image.

Execution of ROM Image

The firmware executes a ROM image either directly from the ROM or after copying it to the RAM. The following example code outlines how this procedure is performed:

```
If FLAGS:SHADOW is 0
   If SYPCR_WR is 0
      Write SYPCR_VAL to SYPCR register
   Set LEDs to GREEN/OFF
   Jump to ROM image offset 108₁₆ (address fe04.0108₁₆)

If FLAGS:SHADOW is 1
   If FLAGS:SMODE is 0
      Copy block [fe04.0100₁₆:fe20.0000₁₆[ to RAM_ADRS
   else
      Copy block [fe04.0000₁₆:fe20.0000₁₆[ to RAM_ADRS

   If SYPCR_WR is 0
      Write SYPCR_VAL to SYPCR register
   Set LEDs to GREEN/OFF
   If RAM_EXEC is within the address range of the copied image
      Jump to RAM address RAM_EXEC
   else
      Jump to RAM_ADRS+108₁₆
```

System State at ROM Image Execution

For an application it is important to know how the CPU and peripherals are configured when control is passed to the ROM image.

CPU Registers

The following CPU registers are initialized during cold boot:

- Machine State register (MSR) is set to 0
- SPRG registers are set to 0
- Segment registers are set to 0
- All D and IBAT registers are cleared.
- The translation lookaside buffers (TLBs) are invalidated

Caches

Both the instruction and the data cache are disabled. The contents of the caches are not explicitely cleared by the firmware. If shadowing a ROM image to RAM for execution, the instruction cache will be enabled.

MMU

The Memory Management Unit (MMU) and block address translation mechanism is not used by the firmware.

Memory Controller

The memory controller is programmed according to table 21 “Memory Map” on page 5-3.

PowerSpan II

The firmware uses mailbox registers 5, 6 and 7 as control and status registers. Furthermore, it programs the address translation of PCI base address register 2 so that it is mapped to the local DRAM.

Port Pin Configuration

The firmware does not program the pin assignment registers for ports A-C. If you have purchased the Force Computers Tornado 2.0 BSP for the PMC-8260/DS1 or one of the Force stackware packages, you do not need to program the PowerQUICC II port pins. If not, the “PowerQUICC II Port Functions on PMC-8260/DS1” section page 6-11 gives information which port functions must be programmed.

Dual-Ported RAM

Parts of the dual-ported RAM are used by the power-on self test:

- The first 268 bytes are overwritten by the memory test.
- The address range from $B800_{16}$ to $B82F_{16}$ is used to store the POST results.

Other PowerQUICC II Settings

The firmware does not change any settings in the PowerQUICC II register set except the registers for the memory controller and the SYPCR register, if SYPCR_VAL and SYPCR_WR are programmed in the ROM image configuration section.

Controlling Boot Process

The execution of the power-up sequence can be controlled by the host via the driver by setting configuration bits in the mailbox register #7 of the PowerSpan II PCI-to-60x bus bridge (offset $46C_{16}$ relative to BAR0). In this context, this register is referred to as power-up control register (PCR).

Table 12: *PCR Register Settings*

Offset: $46C_{16}$

Bit	Signal	Description	Access
31...4	Reserved		
3	INI	1: Firmware is allowed to perform write accesses to PowerSpan II PCI bridge. In particular, it will provide progress information on the boot process and enable access to the 60x bus from the PCI bus.	r/w
2	WDDIS	0: Software watchdog timer remains enabled. 1: Software watchdog timer will be disabled. It is not possible to enable the software watchdog timer once it has been disabled except by resetting PMC-8260/DS1. In any case, the firmware will periodically trigger the software watchdog to prevent it from expiring.	r/w
1	ROMSKIP	0: Firmware will look for a ROM image and execute it if one is found. 1: Firmware will always execute primary booter and not look for ROM image.	r/w
0 (lsb)	PSKIP	0: Power-on self-test will be executed. 1: Power-on self-test will not be executed. This bit is set by the firmware before the power-on self-test is started to prevent the POST from being executed in future.	r/w

Note: Another way to control whether the firmware executes POST is to program the byte at flash offset $000400F1_{16}$: If programmed to 0, the firmware will never execute the POST. If it is programmed to FF_{16} and the PCR[PSKIP] bit is set to 1, the POST will not be executed either. If $000400F1_{16}$ is programmed to FF_{16} and the PCR[PSKIP] bit is set to 0, the POST will be executed.

Monitoring Boot Process

When PCR[INI] is set, the firmware sets status information in the mailbox register 6 of the PowerSpan II PCI-to-60x bus bridge (offset 468_{16}) where it can be read by software. In this context, this register is referred to as power-up status register (PSR).

Before resetting the board, the host should set this register to its reset value of zero. After the module has been started, host software must poll the PSR for a non-zero value before accessing local resources on the 60x bus. Otherwise, the PMC module would hang because a non-zero value means the power-up procedure is not completed. The following values and their mnemonics are defined:

- 1 (POST)
  This value is set immediately before the power-on self-test is started.
- 2 (RDY)
  This value is set after all initialization steps and optionally the POST are completed and the primary booter is about to be executed.
- 3 (ROMEX)
  This value is set after all initialization steps and optionally the POST are completed and a ROM image has been detected and is about to be executed.

Primary Booter

The primary booter contains tools for:

- Loading software via PCI bus
- Programming the flash
- Accessing on-board resources
- Reporting POST result information to the host

For this purpose a simple communication protocol is implemented. On the following pages this communication protocol with its commands, return codes, and the I/O buffer are described. The end of this section provides instructions on how to program routines for using the primary booter.

Resources Used by the Primary Booter

As can be seen in the following figure, the primary booter code is located within the main memory (SDRAM). 256 byte of main memory are used as control and status register section (CSR) which is used for the communication protocol. The CSR section is accessible both from the PCI bus (via the default base translation register provided by the PCI bridge) and the local PowerQUICC II.

The BST0 register of the PowerSpan II PCI bridge is configured to 4 MByte by hardware. The primary booter maps this window to the last 4 MByte of local SDRAM. The PCI memory address is assigned by the PCI bus enumerator of the host.

The following figure shows the memory layout as seen from the local CPU (right) and the PCI bus (left). The SDRAM and SSRAM regions marked by the dashed lines are not directly accessible from PCI bus. In order to read from / write to these regions, the contents must first be copied into the I/O buffer area (see shaded area in the following figure).

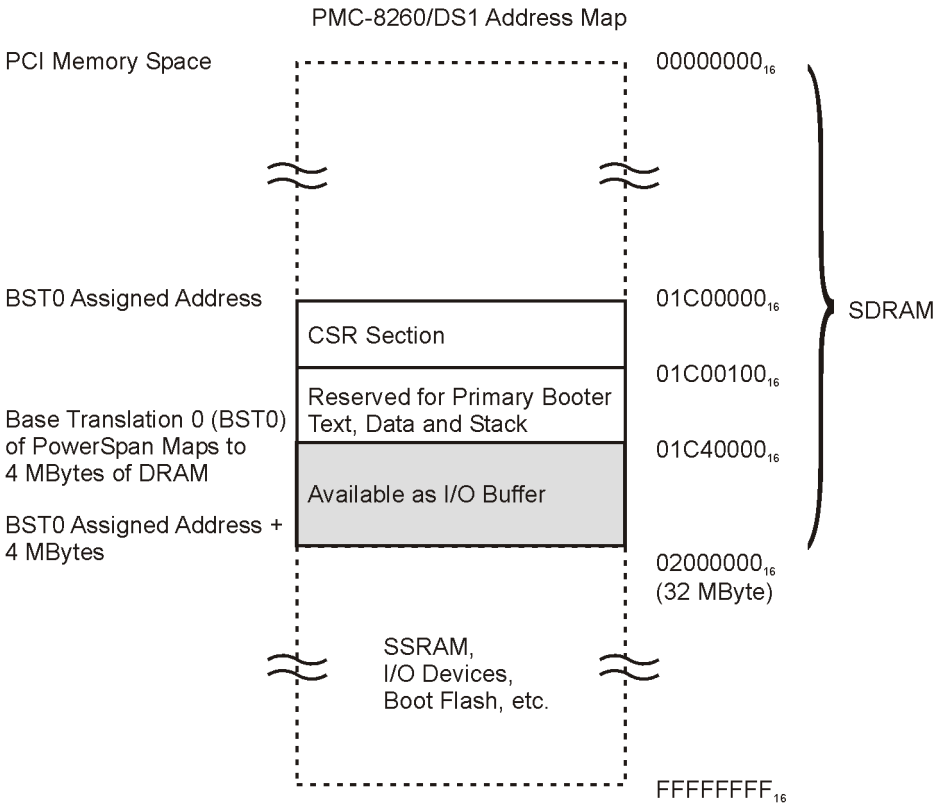


Figure 14: *Memory Layout of the Primary Booter*

Control/Status Register Section

The CSR section ($01C00000_{16}$ - $01C00100_{16}$) is used by the primary booter to report various information, to issue commands from the host to the primary booter, and to report status codes back to the host.

Table 13: *Primary Booter CSR Section*

Offset	Size (bytes)	Name and Description
0_{16}	4	SYNC A 4-byte ASCII string containing command and status codes for communication between host and PMC module. For a list of available commands, see Table 14 “Primary Booter Commands and Parameters” page 4-18.
4_{16}	4	ADRS Argument for SYNC This parameter is used for the .go! command to indicate at which local memory address to start program execution.
40_{16}	16	CPU Zero-terminated string containing the name of the CPU: “PPCEC603”
50_{16}	32	VEND Vendor identifier string: “Force Computers GmbH“. The string expands into the next field (TARG).
70_{16}	32	TARG Zero-terminated string containing the name of the PMC / 8260 module: “PMC8260 / DS1”
90_{16}	16	FVER 16-byte ASCII string containing the version of the primary booter in the format V<x>.<y><MMM><DD><YYYY> where <y> and <y> are the major and minor version numbers in the range of 0 to 9. <MMM>, <DD>, and <YYYY> are the firmware’s date code.
$A0_{16}$	1	CSPC Source / destination address space, must remain 0
$A1_{16}$	1	CWID Source / Destination access width
$A2_{16}$	2	Reserved
$A4_{16}$	4	COFF Address of I / O buffer (offset relative to $01C00000_{16}$ in memory window)
$A8_{16}$	4	Reserved
AC_{16}	4	CADR Source or destination address

Table 13: *Primary Booter CSR Section (cont.)*

Offset	Size (bytes)	Name and Description
$B0_{16}$	4	CSIZ Number of bytes to be transferred
$C0_{16}$	1	PSTAT POST Result: 0: POST passed 1: POST failed
$C1_{16}$	3	Reserved
$C4_{16}$	4	PINFO see Table 16 “POST Result and Status Bits” page 4-25

Primary Booter Commands

The following table shows the commands which can be entered into the SYNC field, their functions and the necessary parameters.

Table 14: *Primary Booter Commands and Parameters*

Command	Description	Necessary Parameters
eras	Erases flash device at the address space denoted by <CSPC>.	CSPC: Destination address space (must be 0)
flsh	Initiates flash programming of <CSIZ> bytes of data stored at offset <COFF> into memory window at the address space denoted by <CSPC>, beginning at flash device offset <CADR>.	COFF: Address of I/O buffer (offset must be used relative to $01C00000_{16}$) CSIZ: Number of bytes to be transferred CADR: Destination address CSPC: Destination address space (must be 0)
.go!	Starts code execution at local memory address <ADRS>. This command terminates the primary booter, i.e. all memory resources used by the primary booter can be overwritten. For a code example, see the “Starting Code Execution” section on page 4-38.	ADRS: Local memory address

Table 14: *Primary Booter Commands and Parameters (cont.)*

Command	Description	Necessary Parameters
ping	Checks alive state of primary booter. Unless the primary booter is busy with the previous operation, it will set the .OK. status immediately as it sees PING. For a code example, see the "Checking Alive State of Target" section on page 4-37.	
read	Copies bytes set by parameter <CSIZ> from memory window offset set by parameter <COFF>.	COFF: Address of I/O buffer (offset must be used relative to $01C00000_{16}$) CWID: Source access width (must be 0) CADR: Source address CSIZ: Number of bytes to be transferred CSPC: Destination address space (must be 0)
writ	Copies bytes set by <CSIZ> from local address <CADR> to memory window offset <COFF>.	COFF: Address of I/O buffer (offset must be used relative to $01C00000_{16}$) CWID: Destination access width (must be 0) CADR: Destination address CSIZ: Number of bytes to be transferred CSPC: Destination address space (must be 0)

Basic Command Programming Procedure

Program all commands according to the following instruction.

1. Set necessary command parameters (see Table 14 "Primary Booter Commands and Parameters" page 4-18)
2. Issue command by writing it to SYNC word
3. Poll SYNC word to obtain information on command execution status
   For possible return codes, see the "Return Codes" section on page 4-20.

Return Codes

Possible return codes written by the firmware into the SYNC word are:

- BUSY
  Operation is ongoing.
- .OK.
  Command has been completed successfully.
- !ERR
  Error occurred. The ADRS field might give further error information. A value other than 0 can be interpreted as offset relative to the beginning of the CSR section where a zero-terminated error string is stored.
- <Written SYNC word>
  Primary booter either does not support the command or some error prevents the booter from reacting. The latter case can be detected by checking the alive state (see Table 14 “Primary Booter Commands and Parameters” page 4-18).

Starting the Primary Booter

The programming routine for starting the primary booter is necessary for:

- Loading an image from PCI bus
- Programming a ROM image into boot flash
- Perfoming read / write accesses to I / O devices e.g. when reading the boot flash contents or POST results.

The primary booter will start automatically if there is no ROM image present in the boot flash device or if the ROMSKIP bit in the PCR register is set. Communication is possible as soon as the LEDs indicate it (L1 is red, L2 is flashing green).

Program your software according to the following description[1)] to start the primary booter. A code example for this routine is given on page 4-33.

1. Configure PowerSpan II local interrupt INT5 as output

Note: Mailbox registers 5, 6, and 7 are used by the firmware. Using these PowerSpan II registers will cause firmware malfunction. Therefore, do not use the mailbox registers 5, 6 and 7.

---

2. Map one of PowerSpan II’s mailbox or doorbell registers (except mailbox registers 5, 6, and 7) to interrupt pin 5

1) This sequence assumes that PCI resources have already been assigned to the PowerSpan II.

3. Enable mailbox / doorbell interrupt
4. Program access to respective mailbox / doorbell register
5. Set bits INI and ROMSKIP in PCR register
6. Optional: Set bits PSKIP to skip POST and WDDIS to disable software watchdog
7. Set PSR register to zero
8. Deassert reset from local CPU by clearing pending mailbox interrupt
   After resetting the PMC-8260 / DS1 it should not take more than one second for the PSR to be set either to RDY or POST. When POST is set, it should not take more than 16 seconds for the self-test to complete and RDY to be set (the actual time is usually much less, depending on the assembly options of the PMC-8260 / DS1). If it takes longer than 16 seconds for RDY to be set, a firmware or hardware error occured and the PMC module has to be sent back to Force Computers.
9. Poll PSR register until RDY value appears
   After the RDY status is read, it should not take more than one second for the primary booter to assert either ".OK." or "FAIL" status. If it takes longer than one second to get ".OK." or "FAIL", a firmware or hardware error occured and the PMC module has to be sent back to Force Computers.
10. Poll SYNC field at offset 0 of PCI memory window
    The following codes could be returned by the firmware:
    .OK.
    Primary booter is ready for communication. It is recommended to issue a PING command (see Table 14 "Primary Booter Commands and Parameters" page 4-18).
    INIT
    Primary booter is currently initializing itself.
    FAIL
    Fatal error occurred which prevents the primary booter from running. The ADRS field contains an offset relative to the beginning of the CSR section which contains a zero-terminated string with further error information.

Reading and Writing

Only part of the address space of the PMC-8260/DS1 is directly accessible from the PCI bus (see the "Resources Used by the Primary Booter" section on page 4-15). The other parts of the address map (e.g. boot flash area) can be accessed by using the read and write command sequence. Program your software according to the following description[1)]:

1. Define I/O buffer in area available for this purpose ($01C40000_{16}$ - 02000000_{16})
   This is accomplished by writing the offset of the selected I/O buffer relative to the beginning of the memory window into the COFF field (in big endian data format).
2. When accessing hardware devices with a specific port size it might be necessary to set the CWID field to 0
3. Write source/destination address (in the PMC-8260/DS1 address space) of read/write operation into CADR field
4. Write number of bytes to be transferred into CSIZ field
5. For write operations, store data to be written into I/O buffer
6. Start read or write operation by writing command "read" or "writ" into SYNC field respectively
7. Wait for completion (see "Basic Command Programming Procedure" on page 4-19 and "Return Codes" on page 4-20).
   In case of a read operation, data is now visible in I/O buffer

1) Using the primary booter for reading and writing operations requires also a programming routinge for starting the primary booter.

Programming a Boot Image into Boot Flash

This operation is very similar to the write operation. Program your driver according to the following description[1]. A code example for this routine is given on page 4-36.

1. Erase current content of boot image area by writing “eras” into SYNC word
2. Wait for completion
3. Write offset of I/O buffer relative to beginning of memory window into COFF field (in big endian data format) to define I/O buffer in area available as I/O buffer ($01C40000_{16}$ - 02000000_{16})
4. Write programming offset into CADR field
   This offset is relative to the boot image area in the boot flash device, not relative to the beginning of the boot flash device itself.
5. Write number of bytes to be programmed into CSIZ field
6. Store data to be written into I/O buffer
7. Start operation by writing command “flsh” into SYNC field
8. Wait for completion (see the “Return Codes” section on page 4-20)

1) Using the primary booter for programming the boot flash requires also a programming routinge for starting the primary booter.

Obtaining Results from the Power-On Self-Test

While the POST is running, the firmware stores the results of each test in the dual-ported RAM of the PowerQUICC II. Progress is indicated via:

- Front panel LEDs (see the “LED States During Power Up” section page 3-5)
- Mailbox register 5 of the PowerSpan II (offset 464_{16}, see section “POST Result Storage Area” on page 4-24)

Since the POST does not halt if it detects errors your application software should evaluate the POST results written into mailbox register 5 of the PowerSpan II as soon as possible. Otherwise, a fatal error while the POST is running will cause the software watchdog timer to expire and reset the board.

Requirements

POST results are obtained via the primary booter. In order to evaluate the POST results, your driver has to include at least the programming routines for:

- Starting the pimary booter
  Program your software according to the description on page 4-33.
- Reading and writing
  Program your software according to the description on page 4-22.
- Reading POST results
  For a code example, see page 4-38.

For information on the primary booter’s communication window, commands, parameters, and command programming conventions, refer to section “Primary Booter” on page 4-15.

POST Result Storage Area

The POST stores information on its progress and the results in bank 12 of the internal dual-ported RAM of the PowerQUICC II. The offset relative to the IMMR is $B800_{16}$, resulting in the absolute address of $F000B800_{16}$ in the memory map of the PMC-8260 / DS1.

The POST information is stored in the following twelve 32-bit words:

Table 15: *POST Result Storage Area*

Offset Relative to Dual-Ported RAM: $B800_{16}$	
Offset (hex)	**Content**
00	POST result code (PRC)
04	Checksum
08	SDRAM test results
0C	SSRAM test results
10	PowerSpan II test results
14	Framer #1 test results
18	Framer #2 test results
1C	Framer #3 test results
20	Framer #4 test results
24	Lucent T8105 test results
28	Reserved (zero)
2C	Reserved (zero)

Global Results

The POST result code (PRC) is set and each bit represents one of the eight tests being executed, plus an additional “global POST failure” bit (0).

Table 16: *POST Result and Status Bits*

Offset: 00_{16}	
Bit	**Description**
0 (lsb)	POST Failure. Set to 1 if at least one device failed the tests
1..3	Not used
4	Set to 1 if framer #1 passed all tests
5	Set to 1 if framer #2 passed all tests
6	Set to 1 if framer #3 passed all tests
7	Set to 1 if framer #4 passed all tests
8	Set to 1 if the T8105 passed all tests
9	Set to 1 if the PowerSpan II PCI bridge passed all tests

Table 16: *POST Result and Status Bits (cont.)*

Offset: 00_{16}	
Bit	**Description**
10	Set to 1 if the SDRAM passed all tests
11	Set to 1 if the SSRAM passed all tests

Checksum

The checksum word contains the sum of all other POST result words. Before evaluating the POST results the software must calculate this checksum and verify it against the checksum word. If the checksum does not match, no POST has been executed.

POST Progress Information

Provided that the PCR[INI] bit is set (see the "Controlling Boot Process" section on page 4-13), the POST uses mailbox register 5 of the PowerSpan II (offset 464_{16}) to pass information to the host which test is currently being executed. In this context this register is referred to as the test status register (TSR).

When a particular test is ongoing the corresponding bit as defined in table 16 "POST Result and Status Bits" on page 4-25 is set. All other bits are cleared. A value of zero indicates that the POST is either complete or a test has been completed and the next test is about to be started.

The TSR can be used to monitor the progress of the POST and to see what the last test was that has been executed in case of a fatal error.

Device-Specific Results

More detailed information on the tests performed on each device are stored in the test result words. Each bit set in one of these words indicates a specific operation which failed.

In general, a test result word of $FFFFFFFF_{16}$ indicates a device that is either not assembled or could not be accessed. Only the latter case is indicated as error in the PRC.

Table 17: *SDRAM Test*

Offset: 08_{16}	
Status Bit	**Description**
0 (lsb)	The assembled SDRAM size as read from the port C configuration pins 30 and 31 is not supported.
1	The SDRAM size determined by probing differs from the SDRAM size reported by the configuration pins.
4...30	Memory pattern test failed. The index of the long-WORD which failed the test is stored in bits 4...30 (i.e. shifted by four), beginning with 1 for the first address.
31	Fatal error during test execution

Table 18: *Synchronous SRAM Test*

Offset: $0C_{16}$	
Status Bit	**Description**
0 (lsb)	No device was detected at the expected address.
1	The assembled SSRAM size as determined by probing is not supported.
4...30	Memory pattern test failed. The index of the long-WORD which failed the test is stored in bits 4...31 (i.e. shifted by four), beginning with 1 for the first address.
31	Fatal error during test execution

Table 19: *PowerSpan II Tests*

Offset: 10_{16}	
Status Bit	**Description**
0 (lsb)	No device was detected at the expected address.
1	Reading the ID register does not return $826010E3_{16}$.
2	Reading the Subsystem Vendor ID does not return 1146_{16}. This indicates a problem with the serial configuration PROM of the PowerSPAN II.
3	Writing and verifying value 55555555_{16} to the DMA0 source address register failed.
4	Writing and verifying value $AAAAAAAA_{16}$ to the DMA0 source address register failed.
5	Writing and verifying value 0 to the DMA0 source address register failed.
31	Fatal error during test execution

Table 20: *Framer Tests*

Offset: $14\text{-}20_{16}$	
Status Bit	**Description**
0 (lsb)	No device was detected at the expected address.
1	TYPE section in the ID register is not 001_2.
2	Data mismatch when accessing register at offset 0
3	Data mismatch when accessing register at offset FF_{16}
4	Data mismatch when writing 55_{16} to register FE_{16}
5	Data mismatch when writing AA_{16} to register FE_{16}
31	Fatal error during test execution

Table 21: *T8105 Tests*

Offset: 24_{16}	
Status Bit	**Description**
0 (lsb)	No device was detected at the expected address.
1	Wrong device ID (CSR register at offset FE_{16})
2	Data mismatch when writing 55_{16} to the first byte of CAM Data Memory 1
3	Data mismatch when writing AA_{16} to the first byte of CAM Data Memory 1
31	Fatal error during test execution

Code Examples

This section contains C code examples which can be implemented into your driver software to enable communication between the host and the PMC-8260/DS1 firmware via the PCI bus.

It assumes that the PowerSpan II PCI bridge has been fully initialized by the host's PCI enumerator and is accessible. The code printed in bold at the beginning of the data definitions may require some modification to your host environment.

Data Definitions

```
/* some compatibility macros (e.g. for vxWorks) */
#define UINT32          unsigned int
#define sleep(s)        taskDelay(s*sysAuxClkRateGet())

/* Set this if host is big endian */
#define BIG_ENDIAN_HOST

#define LLSB(x)  ((x) & 0xff)
#define LNLSB(x) (((x) >> 8) & 0xff)
#define LNMSB(x) (((x) >> 16) & 0xff)
#define LMSB(x)  (((x) >> 24) & 0xff)

#ifdef BIG_ENDIAN_HOST

#define LONGSWAP(x) ((LLSB(x) << 24) | \
                     (LNLSB(x) << 16)| \
                     (LNMSB(x) << 8) | \
                     (LMSB(x)))
#define NOLONGSWAP(x)   (x)

#else

#define LONGSWAP(x)     (x)
#define NOLONGSWAP(x) ((LLSB(x) << 24) | \
                     (LNLSB(x) << 16)| \
                     (LNMSB(x) << 8) | \
                     (LMSB(x)))
#endif

/* Required PowerSpan Registers */
#define PSPAN_MBOX(num)         (0x00000450 + num*4)
#define PSPAN_ISR0              (0x00000410)
#define PSPAN_IER0              (0x00000418)
#define PSPAN_INT_MBOX0         (0x00000001)
```

```
#define PSPAN_IMR_MBOX          (0x00000420)
#define PSPAN_IMR_MBOX0         (0x0000000e)
#define PSPAN_IMR_MBOX0_S       (1)
#define PSPAN_INT_INTA1         0
#define PSPAN_INT_INTA2         1
#define PSPAN_INT_INT0          2
#define PSPAN_INT_INT1          3
#define PSPAN_INT_INT2          4
#define PSPAN_INT_INT3          5
#define PSPAN_INT_INT4          6
#define PSPAN_INT_INT5          7

#define RREAD(off,val)          {val=(*(volatile
UINT32*)(csrBase+off));\
                                 val = LONGSWAP(val);}
#define RWRITE(off,val)         *(volatile UINT32*)(csrBase+off) =\
                                LONGSWAP(val)

/* Firmware status and control registers in PowerSpan */
#define PMC8260_FW_PCR          (PSPAN_MBOX(7))
#define PMC8260_FW_PSR          (PSPAN_MBOX(6))
#define PMC8260_FW_PCR_PSKIP    1       /* skip POST flag */
#define PMC8260_FW_PCR_ROMSKIP  2       /* skip ROM image flag */
#define PMC8260_FW_PCR_WDDIS    4       /* Disable WD at startup */
#define PMC8260_FW_PCR_INI      8       /* May write to PSpan? */

#define PMC8260_FW_PSR_POST     1       /* POST is executing */
#define PMC8260_FW_PSR_RDY      2       /* Primary booter runs */
#define PMC8260_FW_PSR_ROMEX    3       /* ROM image runs */

/* Primary Booter CSR section */
#define PB_HDR_SYNC     (0x00000000)    /* Synchronization word */
#define PB_HDR_ADRS     (0x00000004)    /* Adrs or error offset */
#define PB_HDR_VERS     (0x00000029)    /* CSR Version */
#define PB_HDR_CPU      (0x00000040)    /* Name of CPU */
#define PB_HDR_VEND     (0x00000050)    /* FORCE COMPUTERS ... */
#define PB_HDR_TARG     (0x00000070)    /* Target name */
#define PB_HDR_FVER     (0x00000090)    /* Firmware version */

#define PB_HDR_CP_CSPC  (0x000000a0)    /* Source/dest. space */
#define PB_HDR_CP_CWID  (0x000000a1)    /* Access width */
#define PB_HDR_CP_COFF  (0x000000a4)    /* Offset into window */
#define PB_HDR_CP_CADR  (0x000000a8)    /* Src./dst. address */
#define PB_HDR_CP_CADRH (0x000000a8)    /* -> high */
#define PB_HDR_CP_CADRL (0x000000ac)    /* -> low*/
#define PB_HDR_CP_CSIZ  (0x000000b0)    /* # bytes to move */

#define PBSTR(off)        (char*)(memBase+off)
#define PBSYNC            PBSTR(PB_HDR_SYNC)
#define PBREADL(off,val)  {val = (*(volatile
UINT32*)(memBase+off));\
                           val = NOLONGSWAP(val);}
```

```
#define PBWRITEL(off,val) *(volatile UINT32*)(memBase+off) =\
                           NOLONGSWAP(val)

#define PBREADB(off,val)  val = (*(volatile char*)(memBase+off));
#define PBWRITEB(off,val) *(volatile char*)(memBase+off) = val
```

Resetting the PMC-8260/DS1

```
/* Reset/start PMC8260 */
void pmc8260_reset_control (
        char *csrBase,  /* CSR Base address */
        int reset       /* 0 = clear reset, 1 = assert reset */
        )
{
        UINT32  dummy;

        if (!reset)
        {
                /* Deassert PowerQuicc-II reset by clearing mailbox
                 * 0 interrupt */
                RWRITE(PSPAN_ISR0, PSPAN_INT_MBOX0);
                return;
        }

        /* Set PowerQuicc-II reset by asserting local interrupt 5
         */

        /* map mailbox 0 to LINT 5 */
        RREAD(PSPAN_IMR_MBOX, dummy);
        dummy = (dummy & ~PSPAN_IMR_MBOX0) |
          (PSPAN_INT_INT5 << PSPAN_IMR_MBOX0_S);
        RWRITE (PSPAN_IMR_MBOX, dummy);

        /* Enable mailbox 0 interrupt */
        RREAD(PSPAN_IER0, dummy);
        RWRITE(PSPAN_IER0, PSPAN_INT_MBOX0);

        /* Access mailbox 0 to assert interrupt/RESET */
        RWRITE(PSPAN_MBOX(0), 0);

        /* wait one second */
        sleep (1);
}
```

Starting the Primary Booter

```
/* Reset module and start firmware.
 * Returns 0 on error, 1 on success
 */
int pmc8260_fw_start (
        char *csrBase,          /* CSR Base address */
        char *memBase           /* PCI window base address */
        )
{
        int seconds = 0;
        int done = 0;
        UINT32  dummy;

        /* reset the module */
        printf ("resetting module ..\n");
        pmc8260_reset_control (csrBase, 1);

        /* Clear PSR and set INI and ROMSKIP in PCR */
        RWRITE(PMC8260_FW_PSR, 0);
        RWRITE(PMC8260_FW_PCR, PMC8260_FW_PCR_INI |
                                        PMC8260_FW_PCR_ROMSKIP);

        /* Re-start the module */
        pmc8260_reset_control(csrBase, 0);

        /* Wait for firmware to come up */
        printf ("waiting for firmware ");
        while (!done)
        {
                RREAD(PMC8260_FW_PSR, dummy);
                switch (dummy)
                {
                case 0: printf (".”); break;
                case PMC8260_FW_PSR_POST:
                        printf ("POST "); break;
                case PMC8260_FW_PSR_ROMEX:
                        printf ("ROM Image execution\n");
                        return 0;
                case PMC8260_FW_PSR_RDY:
                        printf ("OK\n");
                        done = 1;
                }
                sleep(1);
                if (seconds++ > 20)
                {
                        printf("Timeout\n");
                        return 0;
                }
        }
```

```
        /* Wait for OK status */
        printf (“waiting for OK status “);
        for (seconds = 0; seconds < 5; seconds++)
        {
                if (!strncmp(PBSYNC, “INIT”, 4))
                {
                        printf (“.”);
                }
                if (!strncmp(PBSYNC, “FAIL”, 4))
                {
                        printf (“Primary booter failure\n”);
                        return 0;
                }
                if (!strncmp(PBSYNC, “.OK.”, 4))
                {
                        printf (“OK\n”);
                        printf (“Board Name:    %s\n”,
                                PBSTR(PB_HDR_TARG));
                        printf (“Firmware Ver.: %s\n”,
                                PBSTR(PB_HDR_FVER));
                        return 1;
                }
                sleep(1);
        }
        printf (“timeout\n”);
        return 0;
}
```

Waiting for Command Completion

```
/* Wait for completion of current command
 * Returns 0 on error, 1 on success
 */
int pmc8260_wait (
         char *memBase  /* PCI window address */
         )
{
    int seconds = 0;
    printf (“waiting “);

    while (1)
    {
        if (!strncmp (PBSYNC, “.OK.”, 4))
        {
            printf (“OK\n”);
            return 1;
        }
        if (!strncmp (PBSYNC, “FAIL”, 4))
        {
            printf (“FAILURE\n”);
            return 0;
        }
        if (strncmp (PBSYNC, “BUSY”, 4))
        {
            if (seconds > 2)
            {
                printf (“command acknowledge timeout\n”);
                return 0;
            }
        }
        printf (“.”);
        sleep(1);
        if (seconds++ > 40)
        {
            printf (“Still BUSY after 40 seconds ...\n”);
            return 0;
        }
    }
}
```

Data Transfer and Flash Programming

```
/* read/write to/from 60x bus or program a ROM image.
 * Uses offset 1MB-2MB into PMC8260 PCI memory window as I/O
 * buffer.
 * Returns 0 on error, 1 on success.
 */
int pmc8260_rdwr (
        char    *memBase,       /* PCI window address */
        int     write,          /* 0==read, 1==write, 2==flash */
        char    *buffer,        /* read/write buffer on host */
        int     nbyte,          /* # of bytes to transfer */
        UINT32  targAdrs        /* PMC8260 60x bus adrs */
        )
{
    char        *ioBuff = memBase+0x100000;
    int         remaining = nbyte;
    int         size;

    /* set copy arguments (dfl. size, space and I/O buffer offset)
     */
    PBWRITEB(PB_HDR_CP_CSPC, 0);
    PBWRITEB(PB_HDR_CP_CWID, 0);
    PBWRITEL(PB_HDR_CP_COFF, 0x100000);

    while (remaining > 0)
    {
        size = (remaining > 0x100000) ? 0x100000 : remaining;
        PBWRITEL(PB_HDR_CP_CADRL, targAdrs);
        PBWRITEL(PB_HDR_CP_CSIZ, size);

        printf (“transferring %d bytes from/to target or “
                “flash adrs. 0x%x\n”,
                size, targAdrs);

        if ((write == 1) || (write == 2))
            bcopy(buffer, ioBuff, size);

        /* issue command */
        switch (write)
        {
            case 0:     strcpy(PBSYNC, “read”); break;
            case 1:     strcpy(PBSYNC, “writ”); break;
            case 2:     strcpy(PBSYNC, “flsh”); break;
        }

        /* wait for completion */
        if (!pmc8260_wait(memBase))
            return 0;

        if (write == 0)
```

```
            bcopy(ioBuff, buffer, size);

        buffer += size;
        targAdrs += size;
        remaining -= size;
    }
    return 1;
}
```

Erasing the ROM Image

```
/* Erase ROM image area
 * Returns 0 on error, 1 on success.
 */
int pmc8260_rom_erase (
        char    *memBase        /* PCI window address */
        )
{
    printf (“erasing ROM image “);

    strcpy(PBSYNC, “eras”);

    if (!pmc8260_wait(memBase))
        return 0;

    return 1;
}
```

Checking Alive State of Target

```
/* Check alive status of Primary Booter
 * Returns 0 on error, 1 on success.
 */
int pmc8260_ping (
        char    *memBase        /* PCI window address */
        )
{
    strcpy(PBSYNC, “PING”);

    if (!pmc8260_wait(memBase))
    {
        printf (“No answer\n”);
        return 0;
    }

    printf (“Target %s@0x%08x is alive\n”,
            PBSTR(PB_HDR_TARG), memBase);
    return 1;
}
```

Starting Code Execution

```
/* Start code execution on target. This terminates the Primary
 * Booter.
 */
void pmc8260_exec (
        char    *memBase,       /* PCI window address */
        UINT32  execAdrs        /* execution target address */
        )
{
    printf ("starting code execution at 60x-Bus address 0x%08x\n",
            execAdrs);

    PBWRITEL(PB_HDR_ADRS, execAdrs);
    strcpy(PBSYNC, ".go!");
}
```

Reading POST Results

```
/* Read and check POST results.
 * Returns 1 on success, 0 on failure.
 */
int pmc8260_post_get (
        char    *memBase,       /* PCI window address */
        UINT32  *postArray      /* POST result codes */
        )
{
    int         i;
    UINT32      chksum;

    if (!pmc8260_rdwr(memBase, 0, postArray, 0x30, 0xf000b800))
        return 0;

    for (i = 0; i < 12; i++)
        postArray[i] = NOLONGSWAP(postArray[i]);

    chksum = postArray[0];
    for (i = 2; i < 12; i++)
        chksum += postArray[i];

    if ((chksum != postArray[1]) || (postArray[0] == 0))
    {
        printf ("POST checksum error\n");
        return 0;
    }

    printf ("Framer 1:  %s [0x%08x]\n",
            (postArray[0] & (1<<4)) ? "pass":"fail",
            postArray[5]);
    printf ("Framer 2:  %s [0x%08x]\n",
```

```
                (postArray[0] & (1<<5)) ? “pass”:”fail”,
                postArray[6]);
        printf (“Framer 3:  %s [0x%08x]\n”,
                (postArray[0] & (1<<6)) ? “pass”:”fail”,
                postArray[7]);
        printf (“Framer 4:  %s [0x%08x]\n”,
                (postArray[0] & (1<<7)) ? “pass”:”fail”,
                postArray[8]);
        printf (“T8105:     %s [0x%08x]\n”,
                (postArray[0] & (1<<8)) ? “pass”:”fail”,
                postArray[9]);
        printf (“PowerSpan: %s [0x%08x]\n”,
                (postArray[0] & (1<<9)) ? “pass”:”fail”,
                postArray[4]);
        printf (“DRAM:      %s [0x%08x]\n”,
                (postArray[0] & (1<<10)) ? “pass”:”fail”,
                postArray[2]);
        printf (“SRAM:      %s [0x%08x]\n”,
                (postArray[0] & (1<<11)) ? “pass”:”fail”,
                postArray[3]);
        return 1;
    }
```

5

Memory Map and Devices

Block Diagram

The block diagram gives an overview on how the PMC-8260/DS1 devices are connected.

Note:

- **The framers and their connections drawn with a dashed line are only available on the PMC-8260/DS1-H.110-RLH variant.**
- **The inter-TMD link represented as a dotted line is only available on the H.110-F and H.110-RLH variants.**

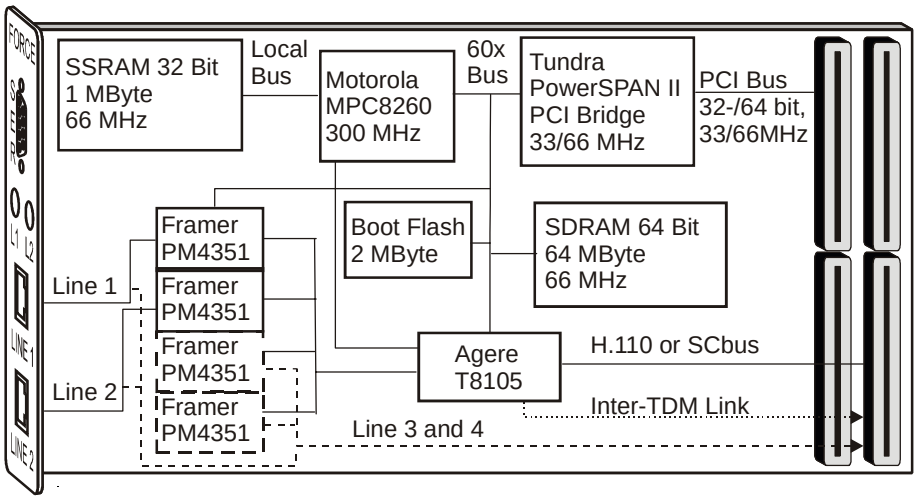


Figure 15: *Block Diagram*

Memory Map

The following table shows how important devices are mapped on the PMC-8260/DS1 by the firmware.

Table 22: *Memory Map*

Device	Address Range	Port Size	Chip-select	Comment
SDRAM (64 MByte)	00000000_{16} - $03FFFFFF_{16}$	64	1	Mapped to Boot Flash if PowerQUICC II is in reset for reprogramming
SSRAM	18000000_{16} - $180FFFFF_{16}$ (1 MByte)	32	2 - 5	Local bus
PowerSPAN II Register	30000000_{16} - 30001000_{16}	32	n.a.	Decoded by PowerSPAN II (4 KByte space PB_REG_BADDR is one of four speciality processor bus slave images)
PowerQUICC II	$F0000000_{16}$ - $F001FFFF_{16}$	32	n.a.	128 KByte register and internal RAM (Internal Memory Map register (IMMR) power-up option Internal Space Base (ISB))
Framer #1	$F0040000_{16}$ - $F00401FF_{16}$	8	6	
Framer #2	$F0040200_{16}$ - $F00403FF_{16}$	8	6	
Framer #3	$F0040400_{16}$ - $F00405FF_{16}$	8	6	
Framer #4	$F0040600_{16}$ - $F00407FF_{16}$	8	6	
Agere T8105	$F0080000_{16}$ - $F0087FFF_{16}$	8	7	
LED_CTRL register	$F00A0000_{16}$ - $F00A7FFF_{16}$	8	8	
Boot Flash	$FE000000_{16}$ - $FE1FFFFF_{16}$ (2 MByte)	8	0	Reset value: 00000000_{16}

Interrupts

Interrupts enforce the PowerQUICC II to temporarily suspend normal processing and to transfer control to an interrupt handling routine. On the PMC-8260/DS1 the PowerSPAN II, framers, T8105, and PowerQUICC II have interrupt capability. The devices are connected to the PowerQUICC II according to the table below.

Table 23: *Interrupt Inputs of the PowerQUICC II*

Device	PowerQUICC II IRQ Input
PowerSpan II INT0	IRQ1
PowerSpan II INT3	IRQ6
Framers 1-4	IRQ2
Timeslot Interchanger T8105	IRQ4

For further information, refer to the *MPC8260 PowerQUICC II User's Manual* and to the *PowerSpan II User Manual*.

Motorola PowerQUICC II

The PowerQUICC II (MPC8260 HiP4) is a versatile microprocessor and peripheral controller for communication and networking applications. It has a core frequency of 300 MHz and a Communication Processor Module (CPM) frequency of 200 MHz. If you need additional information to the information given in this section, refer to the *MPC8260 PowerQUICC II User's Manual* and the *MPC8260 PowerQUICC II User's Manual MPC8260A (HiP4) Supplement*.

Bus Monitor

The PowerQUICC II has two bus monitors, one for the 60x bus and one for the local bus. They monitor the accesses to the memory and the I/O devices and generate a machine check or reset interrupt if the time-out value is reached.

By default, the bus monitor on the PMC-8260/DS1 is active upon booting and the time-out value is set to 30.9 µsec. If you want to disable the bus monitor because the short time-out value could create an undesired machine check or reset interrupt, this can be programmed into the ROM image. The firmware then sets the SYPCR register accordingly. For further information, refer to the "ROM Image Configuration" section on page 4-9.

Communication Processor Module

The CPM contains features that allow the PowerQUICC II to excel in a variety of applications targeted mainly for networking and telecommunication markets.

Serial Communication Channels

The PowerQUICC II has six channels for serial communication: two SMCs (Serial Management Controller) called SMC_1 and SMC_2 and four SCCs (Serial Communication Controller) called SCC_1…SCC_4. They are connected to the PMC-8260/DS1 interfaces as follows:

- SCC_1 is connected to the 9-pin MicroD-Sub connector on the front panel via an RS-232 driver.
- SMC_1 is connected to the debug mode connector via RS-232 drivers.

- SCC_2 is used in a factory option for 10BaseT Ethernet on the front panel 9-pin MicroD-Sub (external PHY is needed).
- SCC_3 is used as CT-Bus Management Bus (CT_MC).
- SMC_2 and SCC_4 are not connected to any external devices.

To be able to use the serial interfaces, you have to program the respective serial controllers in your driver.

For the availability of serial I/O channels refer to Table 8 “Availablility of Serial I/O Channels” page 3-4.

Serial E^2PROM

A serial E^2PROM (24C02) is connected to the PowerQUICC II I^2C controller via a two-wire interface (SDA and SCL lines). The serial E^2PROM has a capacity of 2048 bit, and is internally organized as 256 x 8 bit. In the first 128 bytes the Board Information Block (BIB) of the PMC-8260/DS1 is contained.

Note: Do not overwrite the first 128 byte of the serial E^2PROM, otherwise the PMC-8260/DS1 does not boot. Therefore, use only the last 128 byte for your data.

The serial E^2PROM’s address pins are connected to GND which means that the I^2C device’s address is 0_{16}. For details on the device operations refer to the *Serial E^2PROM Data Sheet*.

Serial Interface with Time Slot Assigner and Multi-Channel Controller

Two Serial Interface (SI) blocks in the PowerQUICC II (SI1 and SI2) can be programmed to handle eight TDM lines concurrently. TDM channels on SI1 are referred to as TDMa1, TDMb1, TDMc1, TDMd1; TDM channels on SI2 are TDMa2, TDMb2, TDMc2, TDMd2.

Each SI has the following features:

- Connect to four independent TDM channels. Each TDM can be one T1 or E1 line.
- Independent, programmable transmit and receive routing paths
- Internal routing and strobe selection can be dynamically programmed.
- Supports automatic echo and loopback mode for each TDM.

The PowerQUICC II's two multi-channel controllers (MCC1 and MCC2) each handle up to 128 serial, full-duplex data channels. The 128 channels are divided into four subgroups of 32 channels each. One or more subgroups can be multiplexed through corresponding SIx TDM channels; MCC1 connects through SI1, and MCC2 uses SI2. Each channel can be programmed separately either to perform HDLC formatting/ deformatting or to act as a transparent channel. The MCCs provide independent mapping for receive/transmit and up to 256 DMA channels with independent buffer descriptor (BD) tables. For a detailed description of the TDM stream structure on the PMC-8260/DS1 see the TDM Channel Configuration chapter.

PowerSPAN II PCI Bus Bridge

The Tundra semiconductor PowerSPAN II chip is the bridge between the PCI bus and the PowerQUICC II Processor Bus (PowerPC Bus or 60x bus).

The main features of the PowerSPAN II are:

- Processor Bus (PB) interface (also called 60x bus)
- PCI interface
- Four DMA controllers
- Mailboxes and doorbell capabilities

The following table shows the PowerSPAN II access from 60x and PCI bus.

Table 24: *PowerSPAN II Access from 60x Bus*

Device	Size	Width	60x Address	Note
PowerSPAN II registers	4 KByte	64 bit	Programmable	Decoded by Power-SPAN II
PCI bus	Programmable	64 bit	Programmable	Decoded by Power-SPAN II

60x Bus Arbiter

On the PMC-8260/DS1 the PowerQUICC II internal arbiter is used as 60x-bus arbiter. Therefore, if you want to change the access priorities of devices, for example, you have to program the PowerQUICC II internal arbiter because the PowerSPAN II 60x bus arbiter is not used.

E^2PROM Contents

The PowerSPAN II's I^2C controller has a two-wire interface (Serial Data and Serial Clock lines) which connects to a serial E^2PROM.

Certain registers of the PowerSPAN II can be programmed at system reset by data stored in the E^2PROM. This allows, for example, to configure interrupts as input or output and to set unique identifiers for cards on the PCI bus at reset and various image parameters and addresses. Configuring the PowerSPAN II with the E^2PROM allows PowerSPAN II to boot up as a Plug and Play-compatible device. PowerSPAN II supports reads from and writes to the E^2PROM. The PowerSPAN II has a master-only I^2C bus-compatible interface that is used for the initialization of registers and for read-

ing and writing PCI Vital Product Data (VPD). The PowerSPAN II also provides a mechanism for the user to perform master read and write operations to the E^2PROM.

The PowerSPAN II reads the first twelve bytes of the serial E^2PROM during PCI bus reset to initialize certain registers. This is, for example, the PCI_SID register containing the Subsystem Vendor ID, the Subsystem ID or the PBTI0_CTL or PBTI1_CTL registers containing the block size of the PCI target images and others.

The serial E^2PROM is not write-protected and its contents can be modified via PowerSPAN II register EEPROM_CS.

The default contents of the E^2PROM loaded by the PowerSPAN II are listed in the table below. For further information on the PowerSPAN II Serial E^2PROM contents refer to the *PowerSPAN II User Manual*.

Note: The settings printed in bold in the following table must not be modified. Otherwise, the proper function of the firmware delivered together with the PMC-8260/DS1 cannot be guaranteed.

Table 25: *PowerSPAN II Serial E^2PROM Contents*

Byte Offset	Contents	Affected PowerSPAN II Registers
0	01_{16}	Defines short E^2PROM data format
1...4	00_{16}	Reserved
5	$0c_{16}$	P1_CSR[BM]: Enable PCI bus master P1_CSR[MS]: Enable PCI memory space
6	08_{16}	P1_BST0[PRFTCH]: Target image 0 prefetch enable
7...8	**1146_{16}**	**P1_SID[SID]: FORCE Computers Vendor ID**
9...10	**8260_{16}**	**P1_SID[SVID]: PMC8260/DS1 assigned subsystem ID**
11	02_{16}	P1_MISC1[INT_PIN[0]]: Interrupt pin enable
12	**18_{16}**	**P1_MISC_CSR[BSREG_BAR_EN]: Register image BAR enable** **P1_TI0_CTL[BAR_EN]: Target image 0 BAR enable**
13	**60_{16}**	**P1_TI0_CTL[BS]: Target image 0 block size is 4 MByte**
14	00_{16}	P1_TI2_CTL, P1_TI3_CTL: Target image 2 and 3 block size set to 0
15	00_{16}	PCI Vital Product Data disabled

Table 25: *PowerSPAN II Serial E^2PROM Contents (cont.)*

Byte Offset	Contents	Affected PowerSPAN II Registers
16	80_{16}	MISC_CSR[P1_LOCKOUT]: Clear PCI Lockout after load
17	FF_{16}	IDR: All interrupts configured as output
18	00_{16}	PCI_I2O_CTL: I_2O target image BAR disabled
19...31	00_{16}	Unused
32...63	00_{16}	Data elements for long load format (unused)
64...255	00_{16}	Unused

PowerSPAN II Configuration

This section describes the basic configuration that needs to be applied by applications to gain access to the 60x bus from PCI and vice versa.

PCI Lockout

In a PCI system various configuration options are set both by the PMC-8260/DS1 itself and by the PCI host independently. For some configuration options it must be ensured that they have been configured properly on the PMC-8260/DS1 before the PCI host gains access to the module.

The PCI lockout mechanism ensures that any access from the PCI bus to the PMC-8260/DS1 is retried until all basic resources of the PowerSPAN II have been configured. The data access is completed normally if the lockout state is left.

Note:

- **It must be ensured that the PCI lockout state is ended as soon as possible after power up. Most PCI systems have a retry limit which may cause unpredictable results on the host, if reached.**
- **After ending the PCI lockout state, it does not make sense to set it again for configuring the PowerSPAN II.**

In its factory configuration, the basic settings for the PowerSPAN II are programmed into the serial E^2PROM which is loaded at power up. After the serial download has been completed, the PCI lockout state is left.

Opening More Windows in the PCI Target Channel

The PCI target channel needs to be programmed to make the resources on the 60x bus side visible on the PCI bus i.e. to enable the PCI bus to access devices on the 60x bus. By default, one window of four possible windows is activated by the default settings in the E^2PROM. To activate more windows, program the other PCI target images (P1_TI1_CTL to P1_TI3_CTL) in the firmware of the carrier board. Program the registers of the PCI target images according to the instructions below.

P1_TIx_CTL

Define basic attributes and sizes for the base address registers. These settings must be accomplished while PCI lockout is active. The following settings in P1_TIx_CTL must be programmed:

1. Set bit BAR_EN to 1 to enable BAR in PCI configuration header
2. Set bit MODE to 0 to map BAR to PCI memory space, or to 1 for PCI I/O space
3. Set bit BS to define the size of translation image
4. Optional: Set bit P1_BSTx[PRFTCH] to enable read prefetching

P1_BSTx

Configure the PCI base address by programming P1_BSTx. This setting is usually performed by the PCI bus enumerator of the host which assigns PCI resources to all PCI devices.

P1_TIx_TADDR

Configure the 60x bus base address by programming P1_TIx_TADDR. For further information, see the *PowerSpan II User Manual.*

P1_TIx_CTL

Set the following additional attribute bits and the size of the target image in P1_TIx_CTL:

- Image and translation enable / disable
- Caching attributes
  Setting the bit GBL enforces cache coherency on the 60x bus. Cache Inhibit (CI) is not supported on the PMC-8260 / DS1.
- Read prefetching attributes
  Defines whether to keep or purge prefetched data and the amount of data prefetched

The following table shows an example of how to program the PCI Target Image#0. For further information, refer to the *PowerSPAN II User Manual.*

Table 26: *PCI Target Image#0 Programming Example*

PowerSPAN II Register	Register Contents	Comment
P1_TI0_CTL	$E60A02C0_{16}$	IMG_EN = 1: Image enabled TA_EN = 1: Translation enabled BAR_EN = 1: BAR enabled[1)] MD_EN = 0: No master decode BS = 616: 4 MByte size[2)] MODE = 0: PCI memory space RTT = $0A_{16}$ GBL = 0: Ensure cache coherency CI = 0: No cache inhibit WTT = 02_{16} PRKEEP = 1: Keep prefetched data END = 10_2: Big endian MRA = 0: No aliasing RD_AMT = 0: 8 byte prefetched.
P1_BST0	$D0000008_{16}$	PCI memory space address: $D0000000_{16}$ Read prefetching enabled[3)]
P1_TI0_TADDR	$000000FE_{16}$	60x bus base address: 00000000_{16} Mx = 1: Claim transactions from all masters
P1_CSR	$02B00006_{16}$	MS = 1: Memory space enable BM = 1: Bus master enable

1) Set via configuration E^2PROM.
2) Set via configuration E^2PROM.
3) Set via configuration E^2PROM.

With these settings, the PCI bus address range $D0000000_{16}$ to $D0400000_{16}$ corresponds to the PMC-8260/DS1 60x bus address range 00000000_{16} to 00400000_{16}, which is the first 4 MByte of the SDRAM.

Accessible 60x Bus Resources

The PowerSPAN II can only perform 60x bus transfers with a port size of 64 bits. For this reason only the SDRAM, the SSRAM and the dual-ported RAM of the PowerQUICC II are accessible. All other devices on the 8-bit bus (E1/T1/J1 framers, T8105) cannot be accessed.

In order to update the firmware, it is possible to access the boot flash device and the LED register. Refer to the “Programming the Boot Flash when PowerQUICC II is in Reset” section on page 5-19.

Opening Windows in the 60x Bus Slave Channel

The PowerSPAN II provides eight 60x bus slave images to map 60x bus cycles to the PCI bus. No window is opened by the default settings in the E^2PROM. To enable the PMC module to access the PCI bus and memory of the carrier board, the following basic programming steps are necessary to set up a 60x bus slave image:

1. Program PB_SIx_BADDR register with 60x bus base address
2. Program PB_SIx_TADDR register with PCI base address
3. Set translation attributes and enable translation in PB_SIx_CTL register
4. Ensure that P1_CSR[BM] bit is set to allow PCI mastership to PowerSPAN II
   This is usually done by the PCI enumerator

The following programming example for 60x bus slave image 1 shows the register contents necessary to map 256 MBytes of 60x bus space, beginning at address 20000000_{16} to PCI memory space address 0.

Table 27: *60x Bus Slave Image#0 Programming Example*

PowerSPAN II Register	Register Contents	Comment
PB_SI0_BADDR	20000000_{16}	60x bus address: 20000000_{16}
PB_SI0_TADDR	00000008_{16}	PCI memory space address: 00000000_{16} Mx = 1: Claim transactions from all masters.
PB_SI0_CTL	$D0000040_{16}$	IMG_EN = 1: Image enabled TA_EN = 1: Translation enabled MD_EN = 0: Master decode disabled BS = 10000_2: Block size 256 MByte MODE = 0: Map to PCI memory space PRKEEP = 0: Do not keep prefetched data. END = 10_2: Big endian data format RD_AMT = 0: 8 byte prefetch
P1_CSR	$02B00006_{16}$	MS = 1: Memory space enable BM = 1: Bus master enable

SDRAM and its Performance

The PMC-8260/DS1 module provides 64 MByte on-board Synchronous DRAM (SDRAM) which is 64-bit wide. The SDRAM is clocked with the 60x bus system clock of 66 MHz.

The SDRAM interface supports page-based bank interleaving which allows highest efficiency of the four-bank SDRAM architecture. The burst performance of the SDRAM interface is 3:1:1:1 accesses for 32-byte burst reads and 2:1:1:1 accesses for 32-byte burst writes (CAS latency = 2). Parity generation and check is not supported. This results in high performance main memory access bandwidth for the PowerQUICC II processor.

The SDRAM is accessible from the PowerQUICC II or the PCI bus via the PowerSPAN II PCI bus bridge.

Table 28: *SDRAM Access on 60x Bus (PowerPC Bus)*

Device	Size	Width	60x-bus Address	Controlled by PowerQUICC II
SDRAM	64 MByte	64 bit	00000000_{16} - $03FFFFFF_{16}$ (programmable)	SDRAM machine

SDRAM Machine Memory Controller

The SDRAM interface supports back-to-back page mode. A page remains open as long as back-to-back accesses that hit the page are generated on the bus. The page is closed once the bus becomes idle. The SDRAM interface uses SDRAM devices with four internal banks and supports bank interleaving on a page-based boundary. This method uses low address bits as the bank-select for the SDRAM, thus allowing interleaving on every page boundary. This means that if a missed page is in a different SDRAM bank from the page currently open, the SDRAM machine issues an “activate” command hiding the precharge time of the already open bank.

The SDRAM interface supports single-beat reads/writes up to DWORD (64-bit, 8 byte) size and burst reads/writes of four DWORDs (32-byte).

SSRAM Information for Support Package Programmers

The PMC-8260/DS1 module provides 1 MByte on-board Synchronous Burst SRAM (SSRAM) which is 32-bit wide and clocked at 66 MHz. This SSRAM is connected to the local bus (32-bit wide data, 18-bit wide address) of the PowerQUICC II processor. The 18-bit wide address local bus has an accessible memory space of 256 KByte. Therefore, to realize the access to 1 MByte SSRAM four chip select signals are consumed, each accessing 256 KByte of memory space. The two missing high order address bits are generated out of the four chip select signals by external glue logic on the PMC-8260/DS1.

The access time of the SSRAM is 10 ns. The SSRAM supports single-read, single-write, burst-read, and burst-write access (parity generation and check is supported as a boot option).

Note: The local bus does not burst when accessed from the PowerQUICC II core or from the external master (PCI bridge PowerSPAN II). Therefore, regions that are accessed across the 60x bridge to local bus must be marked as cache-inhibited in the PowerQUICC II core MMU. Otherwise these burst accesses will be terminated with a machine check exception. The local bus uses bursts of 32 byte for certain CPM Serial DMA (SDMA) and Independent DMA (IDMA) accesses. As the local bus is only 32-bits wide, it performs an eight-beat burst.

Table 29: *SSRAM Timing Performance at 66 MHz (T=15.15 nsec)*

Read/Write	Without Parity (Default Setting)	With Parity (Boot Option)
4-Byte Single Read	3 clocks	4 clocks
4-Byte Single Write	2 clocks	2 clocks
32-Byte Burst Read	3:1:3:1:3:1:3:1 (16 clocks)	4:2:2:2:2:2:2:2 (18 clocks)
32-Byte Burst Write	2:1:1:1:2:1:1:1 (10 clocks)	2:1:1:1:2:1:1:1 (10 clocks)

The SSRAM is accessible from the PowerQUICC II or the PCI bus via PowerSPAN II.

Note: The local bus is a separated secondary bus of the PowerQUICC II processor. Accesses to the 60x-bus and the local bus are possible at the same time. From the address mapping the local bus is still mapped into the 32-bit/4 GByte 60x-bus address space.

Table 30: *SSRAM Address on 60x-Bus/Local Bus*

Device	Size	Width	60x-Bus (Local Bus) Address	CS Lines
SSRAM	1 MByte	32 bit	18000000_{16} ...$180FFFFF_{16}$ programmable	CS2,CS3,CS4,CS5

I/O-Bus

As you can see in the following figure, the 8-bit I/O devices of the PMC-8260/DS1 (boot flash, four framers, Agere T8105, and LED register) are connected to the 64-bit wide PowerPC 60x bus. This is done via a bi-directional data buffer (not shown below) which isolates these slow devices from the high-speed SDRAM on the 60x bus.

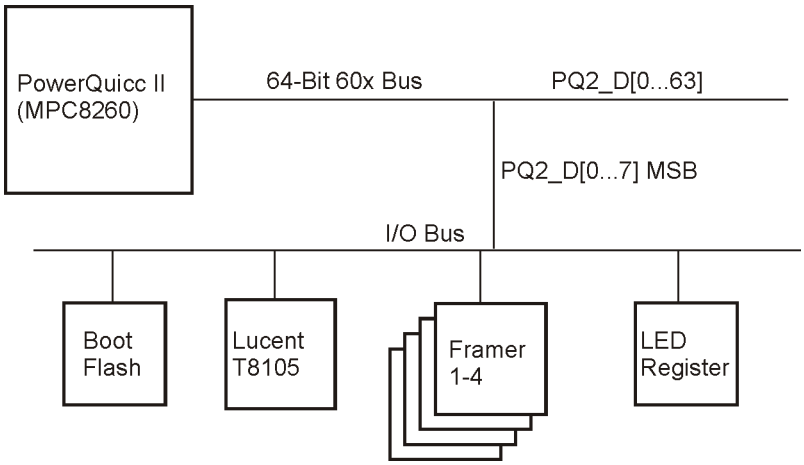


Figure 16: *I/O Bus*

Boot Flash PROM

The PMC-8260/DS1 module provides a 2 MByte on-board boot flash PROM which is 8 bit wide. The access time of the flash PROM is 120 ns.

Table 31: *Flash PROM Access on 60x Bus*

Device	Size	Width	60x Bus Address	Controlled by Line
Flash PROM	2 MByte	8 bit	Programmable (default: $FE000000_{16}$)	CS0

Programming the Boot Flash when PowerQUICC II is in Reset

Note: The following instructions only serve to update the firmware. Programming another firmware than delivered by Force Computers into the boot flash is not supported.

The firmware can be updated by programming a firmware update into the boot flash PROM from the PCI bus side when the PowerQUICC II is held in reset. To put the PowerQUICC II in reset for the boot flash programming procedure the following steps are necessary:

1. Switch off system
2. If carrier board is already installed in system, remove carrier board from system according to carrier board's Installation Guide
3. Set SW1-2 to ON

Note: Switch SW1-2 must be in ON position during the whole procedure of reprogramming the boot flash from the PCI bus. Be sure to put SW1-2 in OFF position after reprogramming the boot flash, otherwise the PMC-8260/DS1 will not power up.

4. Install carrier board into system according to carrier board's Installation Guide
5. Switch power on
6. On carrier board configure interrupt INT5 as output

Note: Mailbox registers 5, 6, and 7 are used by the firmware. Using these PowerSpan II registers will cause firmware malfunction. Therefore, do not use the mailbox registers 5, 6 and 7.

7. Set up one of PowerSpan II's mailbox registers (except mailbox registers 5, 6, and 7 because they are used by the firmware) to trigger PSPAN_INT5 for a minimum of 5 msec

8. Program access to respective mailbox register
   PowerQUICC II is then set into the reset state (Powerup_RESET). On-board glue-logic parks the 60x-bus to the PowerSPAN II and asserts Address Acknowledge (AACK) together with Partial Data Valid (PSDVAL) and Transfer Acknowledge (TA) to terminate the PowerSPAN II's access cycles to the boot flash. No address pipelining is supported by hardware during this time.
9. Program fixed time delay (min. 1 msec) to expire until PowerSPAN II is allowed to program flash
   This time delay is necessary to allow the PowerQUICC II to tristate all signals and buses, so that the PowerSPAN II can act as the only 60x bus master.

As the PowerSPAN II supports only 64-bit port size devices on the 60x bus side, high order 60x-addresses connected to an external multiplexer are used to access the 8-bit boot flash device. See the following figure for a symbolic block diagram of the byte address generation.

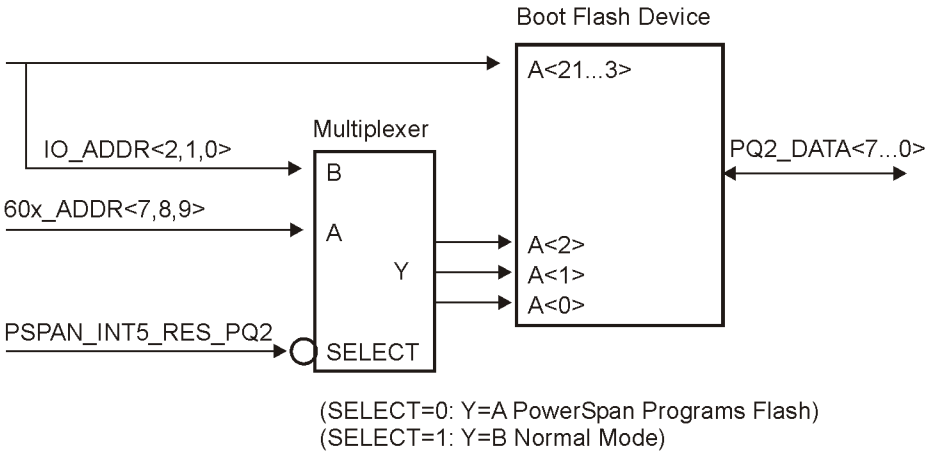


Figure 17: *Byte Address Generation for Flash Programming from PCI Bus via PowerSPAN II*

Memory Map when PowerQUICC II is in Reset

To accommodate the flash programming from the PowerSPAN II a multiplexer is used to generate the byte addresses for the flash device out of high order 60x bus addresses 60x_ADDR<7,8,9>.

Because the PowerSPAN II PCI bus bridge supports only 64-bit slaves and the 8-bit wide boot flash device is connected to the most significant 60x bus byte lane PQ2_DATA[0...7], accesses to the 8-bit boot flash device must always be targeted to the 60x-bus least significant byte addresses

60x_ADDR<29,30,31>=000 (which correspond to byte lane PQ2_DATA[0...7], see Figure 16 “I/O Bus” page 5-18).

For the flash device the byte addresses are generated by connecting the high order 60x-bus addresses 60x_ADDR<7,8,9> to the flash low addresses ADDR<2,1,0> while PQ2 is held in RESET. This is also accomplished with the help of the low-asserted PowerSPAN_INT<5> signal as can be seen in figure 17 “Byte Address Generation for Flash Programming from PCI Bus via PowerSPAN II” on page 5-20.

The resulting memory map is shown in the following table.

Table 32: *Memory Map when PowerQUICC II is in Reset*

60x Address A<0...31>	60x Address <7,8,9>	Flash Byte Address <2,1,0>	Size (MByte)	Addressable Flash Byte Addresses
xx000000 xx3FFFFF	000	000	4	xxxxxxx0 xxxxxxx8
xx400000 xx7FFFFF	001	001	4	xxxxxxx1 xxxxxxx9
xx800000 xxBFFFFF	010	010	4	xxxxxxx2 xxxxxxxA
xxC00000 xxFFFFFF	011	011	4	xxxxxxx3 xxxxxxxB
x1000000 x13FFFFF	100	100	4	xxxxxxx4 xxxxxxxC
x1400000 x17FFFFF	101	101	4	xxxxxxx5 xxxxxxxD
x1800000 x1BFFFFF	110	110	4	xxxxxxx6 xxxxxxxE
x1C00000 x1FFFFFF	111	111	4	xxxxxxx7 xxxxxxxF

The boot flash device’s base address on the 60x bus must be programmed to $0XX00000_{16}$ inside the PowerSPAN II.

Table 33: *Flash PROM Access from PCI Bus with PowerQUICC II Held in Reset*

Device	Size	Width	60x Bus Address
Flash PROM	2 MByte	8 bit	00000000_{16}

Displaying the Boot Flash Programming Progress via LEDs

It is also possible to set values for the four front panel LEDs while PowerQUICC II is in reset (SW1-2 in ON position and PSPAN_INT5 asserted low). This is accomplished using the PSPAN_INT3 signal, which is a default input pulled high. By driving this pin low and executing a dummy write access to the flash memory space with the desired LED value on PQ2_D0...D7, it is possible to set the front panel LEDs to the desired value. After this dummy write access, the PSPAN_INT3 signal has to be negated again, otherwise successive writes to the flash device will modify the LED register contents. Since the LED register is connected to PQ2_D[0...7], the dummy write access must be addressed to byte address 0_{16}.

Note: After any reset, the LED register (see the "LED Control Register" section on page 5-24) also gets reset, resulting in the two front panel LEDs lighting up yellow.

Agere Ambassador T8105

The H.110 bus/SCbus controller Agere Ambassador T8105 provides a complete interface between the H.110 backplane bus and four E1/T1/J1 framers and the PowerQUICC II Time Slot Assigner.

One of the features is the programmable connections to any of the 32 TDM lines (4096 time slots) on the H.110 bus. On the PMC-8260/DS1 only 16 TDM lines CT_D[0...15] of the H.110 bus are supported.

For further information on the T8105, see the *Ambassador T8105 H.100/H.110 Interfaces and Time-Slot Interchangers Advance Data Sheet*.

Table 34: *T8105 Access on 60x-Bus*

Device	Size	Width	60x Bus Address	Controlled by Line
T8105	512 Byte	8 bit	Programmable (default: $F0080000_{16}$)	CS7

PMC-Sierra PM4351

On the PMC-8260/DS1-H110-F and PMC-8260/DS1-SC-F variants there are two and on the PMC-8260/DS1-H110-RLH there are four E1/T1/J1 framers assembled. They can be accessed from the 60x bus.

Table 35: *Framer Access on 60x Bus*

Device	Size	Width	60x Bus Address	Controlled by Line
Framer #1	512 Byte	8 bit	Programmable (default: $F0040000_{16}$)	CS6 (additional glue logic)
Framer #2	512 Byte	8 bit	Programmable (default: $F0040200_{16}$)	CS6 (additional glue logic)
Framer #3	512 Byte	8 bit	Programmable (default: $F0040400_{16}$)	CS6 (additional glue logic)
Framer #4	512 Byte	8 bit	Programmable (default: $F0040600_{16}$)	CS6 (additional glue logic)

Note: The framers operating in E1 or T1 mode are compliant to the ITU G.703 standard. The E1 or T1 line interface transmit signal strongly depends on the cable length and the resulting signal attenuation. When using a long cable, the pulse must be stronger than when using short cables. The pulse shape is programmable via framer registers and can easily be adapted to the user's application. For a detailed description refer to the PMC-Sierra PM4351 data sheets.

On the PMC-8260/DS1-H110-F and PMC-8260/DS1-SC-F variants, the E1/T1/J1 signal lines are routed to the two front panel connectors. The transformers for galvanic decoupling and suppressor diodes for primary and secondary line interface protection are implemented. The front panel line interfaces do not support long haul T1. For the front panel connector pinout see Figure 6 "Front Panel RJ-45 Interface Connector Pinout" page 3-5.

The PMC-8260/DS1-H110-RLH variant has four E1/T1/J1 line interfaces which are routed to the PMC I/O connector. These module variants connect the E1/T1/J1 framer signals RX+/RX- and TX+/TX- directly to the PMC I/O connector without any additional circuitry neither for impedance matching and galvanic decoupling nor for primary/secondary line interface protection. This circuitry must be implemented on the rear transition board.

Force Computers provides a rear transition board called ACC/RTB-8260/DS1. The ACC/RTB-8260/DS1 has implemented all the additional circuitry such as:

- Transformers for impedance matching
- Galvanic decoupling
- Primary/secondary line interface protection.

It also supports long haul T1 mode. When used together with a PPC/PowerCoreCPCI-6750 Rev. 2.0 the ACC/RTB-8260/DS1 provides the eight E1/T1/J1 line interfaces of two PMC-8260/DS1-H110-RLH modules. For detailed information refer to the *ACC/RTB-8260/DS1 Installation Guide.*

LED Control Register

The PMC-8260/DS1 provides an LED Control register to control the front panel LEDs.

Table 36: *LED Control Register Access on 60x-Bus*

Device	Size	Width	60x Bus Address	Controlled by Line
LED control	512 Byte	8 bit	Programmable (default: $F00A0000_{16}$)	CS8

Note:

- **If an LED is set to red and green at the same time, the respective LED lights up yellow.**
- **After any reset, the LED register is also reset, resulting in the front panel LEDs lighting up yellow.**

Table 37: *LED Control Register*

Address: $F00A0000_{16}$

Bit	Signal	Description	Access
7 (LSB)	L1 red	LED 1 lights red 0: LED on (default) 1: LED off	w
6	L1 green	LED 1 lights green 0: LED on (default) 1: LED off	w

Table 37: *LED Control Register (cont.)*

Address: F00A0000$_{16}$

Bit	Signal	Description	Access
5	L2 red	LED 2 lights red 0: LED on (default) 1: LED off	w
4	L2 green	LED 2 lights green 0: LED on (default) 1: LED off	w
3	L3 red	L3 lights red[1)] 0: LED on (default) 1: LED off	w
2	L3 green	L3 lights green[1)] 0: LED on (default) 1: LED off	w
1	L4 red	L4 lights red[1)] 0: LED on (default) 1: LED off	w
0 (MSB)	L4 green	L4 lights green[1)] 0: LED on (default) 1: LED off	w

1) The settings only have an effect on the H110-RLH variant.

Resetting the Devices

The PMC-8260/DS1 provides the possibility to reset the following devices in case of a software hang:

- PowerQUICC II
- I/O devices:
  - Framers
  - Agere T8105
  - LED register

A reset of the PowerQUICC II does not reset the I/O devices and vice versa.

PowerQUICC II

To enforce a PowerQUICC II reset after a firmware or application software hang, program the PowerSPAN II INT<5> signal from the PCI bus as follows:

1. On carrier board configure interrupt INT5 as output

Note: Mailbox registers 5, 6, and 7 are used by the firmware. Using these PowerSpan II registers will cause firmware malfunction. Therefore, do not use the mailbox registers 5, 6 and 7.

---

2. Set up one of PowerSpan II's mailbox registers (except mailbox registers 5, 6, and 7 because they are used by the firmware) to trigger PSPAN_INT5 for a minimum of 5 msec
3. Program access to respective mailbox register
   The PowerQUICC II is then set into the RESET state (Powerup_RESET).

Local I/O Devices from the PCI Bus

It is possible to reset the local I/O devices (framers, T8105, and LED register) from the PCI bus side in the two operating modes:

- PowerQUICC II in reset (with SW1-2 in ON position and PSPAN_INT5 asserted)
- PowerQUICC II in normal operating mode

Program a reset of the I/O devices as follows:

1. Configure interrupt INT4 as output

Note: Mailbox registers 5, 6, and 7 are used by the firmware. Using these PowerSpan II registers will cause firmware malfunction. Therefore, do not use the mailbox registers 5, 6 and 7.

---

2. Set up one of PowerSpan II's mailbox registers (except mailbox registers 5, 6, and 7) to trigger PSPAN_INT4 for a minimum time of 5 msec
3. Program access to respective mailbox register

6

TDM Channel Configuration

About this Chapter

This chapter will enable you to program your communications protocol driver by giving you the hardware-relevant information. If you have purchased one of the Force Computers stackware packages (LAPD, MTP-2 or Frame Relay), you do not need this information because the settings are already implemented in the software.

The following information is given:

- Bit clock and frame pulse pins used on the PMC-8260/DS1 necessary to program a synchronous network.
- How the Time Division Multiplex (TDM) channels of the PowerQUICC II are connected to the framers and the H.110/SCbus via the T8105 timeslot interchanger to enable timeslots routing.
- PowerQUICC II port pin functions used on the PMC-8260/DS1 to enable communication.

Synchronizing the Clock

If the devices in a telecommunication network use different clocks, you will not be able to operate your network. To be able to create a synchronuous network in which all devices use the same clock, the Agere T8105 switch includes a clock multiplexer to select among the possible sources. It is possible to configure the PMC-8260/DS1 to be the clock slave or the clock master.

There may be more ways to configure the clocks than described in this section. For further information, refer to the T8105 documentation.

Configuring the PMC-8260/DS1 as Clock Master

If the PMC-8260/DS1 is to be configured as clock master, the following on-board sources can be used to provide the master clock:

- Line recovered network reference of one of the framers which are provided to the T8105 on its L_REF[0:3] pins. Figure 18 “Local TDM Clocking Structure” on page 6-5 shows which framer is connected to which L_REF pin.
- On-board 8 kHz clock (+/- 50ppm) connected to L_REF_5 of T8105
- On-board quarz crystal oscillator clock 16.384 MHz (+/- 10ppm) which is provided via the T8105 XTALIN/XTALOUT pins.
- CTBus network reference clocks CT_NETREF_1 and CT_NETREF_2 (see Figure 18 “Local TDM Clocking Structure” on page 6-5).

To use one of these clocks, select one of the L_REF input pins, the quarz oscillator input pin, or the CT_NETREF_x input pins inside the T8105 switch via T8105 registers. The selected clocks are then distributed via frame pulse clock line L_SC_1 and bit clock line TCLK_OUT and a clock driver to the clock lines of the framers (BTCLK, BTFP) and the PowerQUICC II (L1RCLK and L1RSYNC). In this way, all on-board devices use the same clock. Figure 18 “Local TDM Clocking Structure” on page 6-5 shows the clock driver and the clock lines.

The PMC-8260/DS1 can be configured as primary or secondary H.110 bus clock master. In order to do this, program CT_C8_A and CT_FRAME_A or CT_C8_B and CT_FRAME_B inside the T8105 as output.

When operating as H.110 bus clock master and driving CT_C8_A and CT_FRAME_A or CT_C8_B and CT_FRAME_B according to the H.110 bus specification, clock masters must shortcut the 33 Ohm series termination resistors available in the signals CT_C8_A and CT_FRAME_A or CT_C8_B

and CT_FRAME_B. On the PMC-8260/DS1, this can be accomplished via the T8105 general purpose I/O pins GP(0)=FET_CTRL_A and GP(0)=FET_CTRL_B. Normally, these GPIO pins are configured as inputs, so that you have to program CT_C8_A, CT_FRAME_A, CT_C8_B and CT_FRAME_B as output and set FET_CTRL_A and FET_CTRL_B to 0.

For information on the T8105 and the registers needed for programming, refer to the T8105 data sheets.

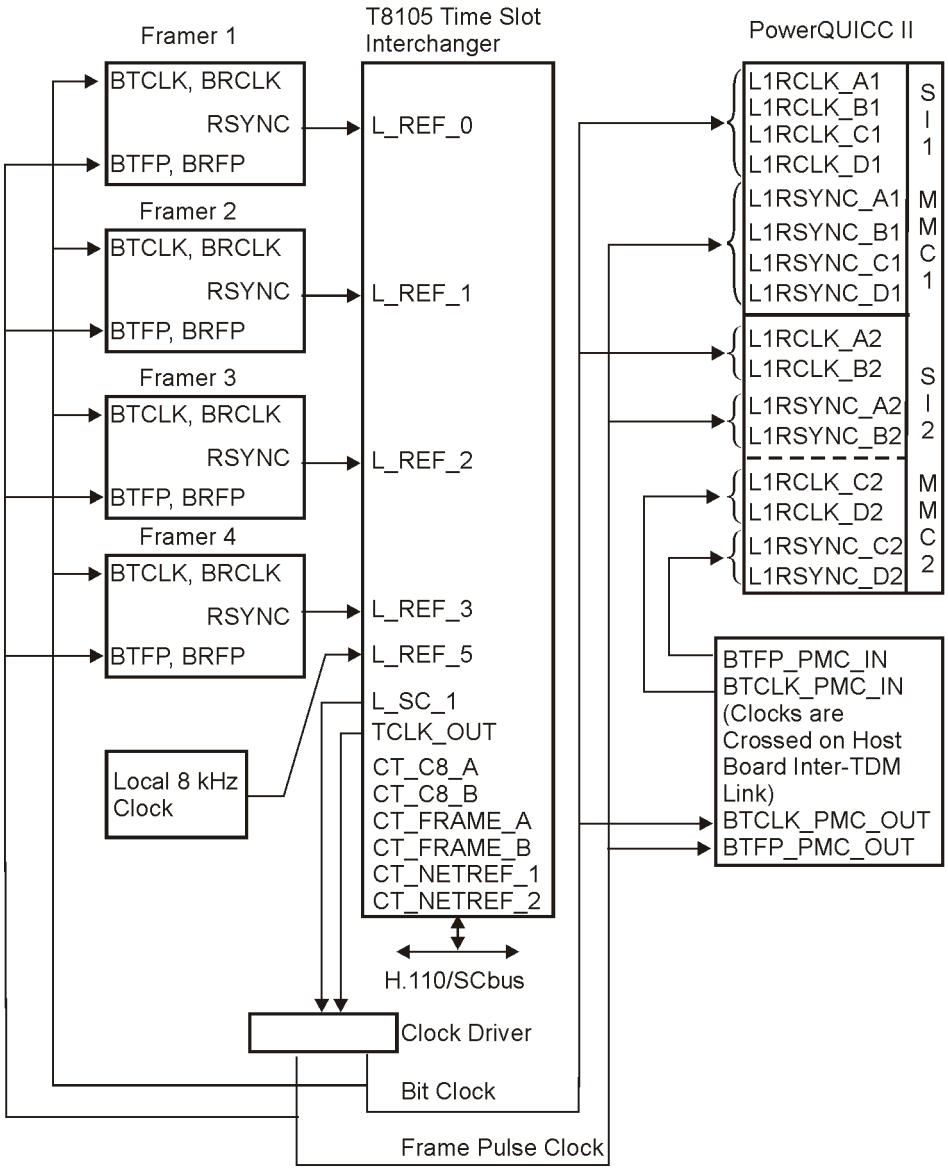


Figure 18: *Local TDM Clocking Structure*

Configuring the PMC-8260/DS1 as Clock Slave

Depending on which clock line (A or B) is the H.110 bus master clock line, program CT_C8_A or CT_C8_B and CT_FRAME_A or CT_FRAME_B inside the T8105 as input. The clocks are then distributed via frame pulse clock line L_SC_1 and bit clock line TCLK_OUT and a clock driver to the clock lines of the framers (BTCLK, BTFP) and the PowerQUICC II (L1RCLK and L1RSYNC). Figure 18 “Local TDM Clocking Structure” on page 6-5 shows these clock lines.

The T8105 general purpose I/O pins FET_CTRL_A and FET_CTRL_B must be configured to activate the 33 Ohm series resistors of line A and B. This is accomplished by setting the bits FET_CTRL_A and FET_CTRL_B to 1 (default).

For information on the T8105 and the registers needed for programming, refer to the T8105 data sheets.

Routing Time Division Multiplex Timeslots

The Time Division Multiplex (TDM) channels of the PowerQUICC II are connected to the framers and the H.110/SCbus via the T8105 timeslot interchanger. The schematic figure shows these connections of the H.110-F and SC-F variants.

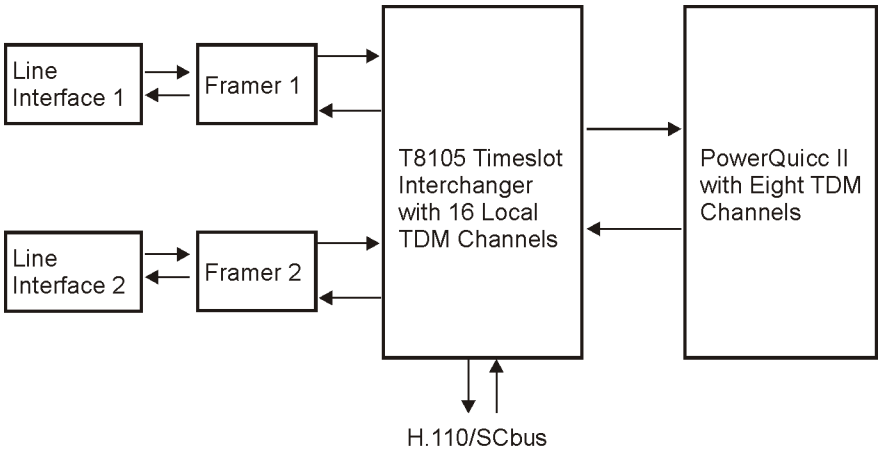


Figure 19: *Connection of TDM Channels*

The timeslot interchanger has 16 local TDM channels, each supporting 32 timeslots. The framers support 24 timeslots if configured for T1/J1 or 32 timeslots if configured for E1. To be able to build your specific network each of the mentioned timeslots can be switched between:

- Local TDM channels (between the PowerQUICC II and the framers)
- H.110/SCbus TDM channels
- Local TDM channels and H.110/SCbus TDM channels

This means that any local timeslot can be switched from the PowerQUICC II or one of the framers to any timeslot on the H.110/SCbus or vice versa. In the same way, any local timeslot can be switched between the framers or between the framers and the PowerQUICC II. This can be seen in the following figure.

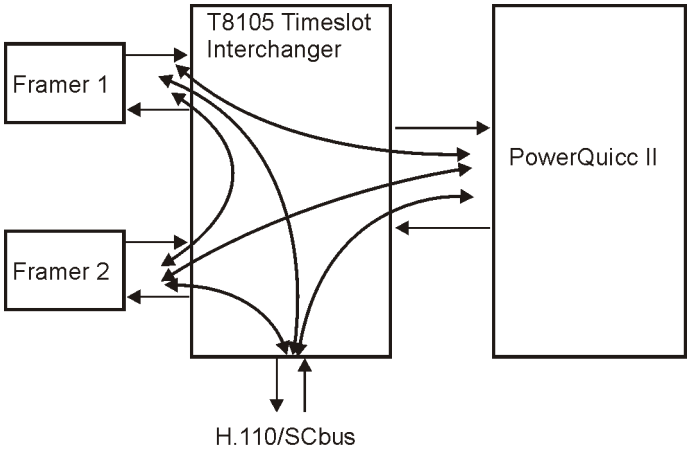


Figure 20: *Switching of TDM Channels*

Routing the timeslots in the described way is done by programming the PowerQUICC II and T8105 registers in your communications protocol driver. Figure 21 “Connection of Local TDM Streams” on the following page shows which T8105 local TDM channel data lines LDI0...LDI15 and LDO0...LDO15 are connected to the framers’ backplane receive / transmit PCM (BRPCM and BTPCM) data lines and which T8105 local TDM channel data lines are connected to the PowerQUICC II SI1 and SI2 TDM lines. If, for example, a timeslot is to be routed from framer 2 to PowerQUICC II TDM channel A2, it has to be routed between LDI_1 and LDO_8 in the T8105.

Note:

- **The local TDM channel data lines LDO_2/3 and LDI_2/3 and the framers 3 and 4 shown in the following figure are only available on the PMC-8260/DS1-H110-RLH variant.**
- **The connection of data lines of the inter-TDM link is shown in a separate figure (Figure 22 “Connection of Data Lines in Inter-TDM Link” page 6-10).**
- **The local TDM channel data lines LDO_12...15 and LDI_12...15 can be used for subrate switching to be able to transmit more calls as is required by mobile communication.**

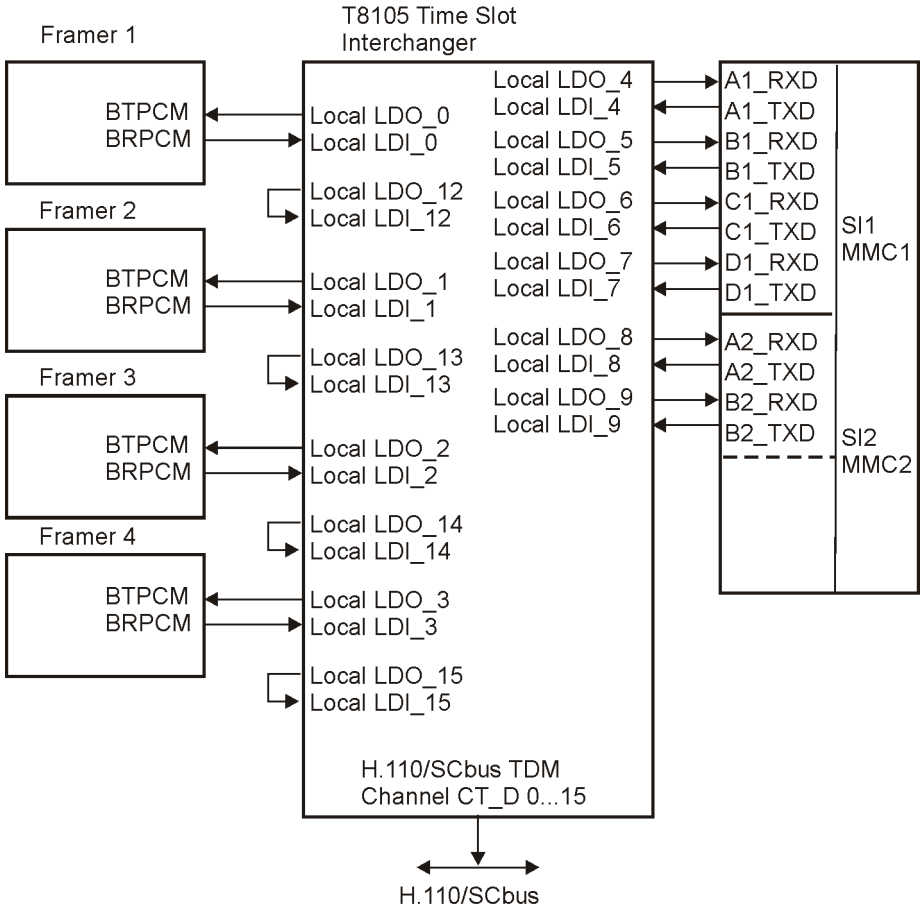


Figure 21: *Connection of Local TDM Streams*

When using the H.110-F or H.110-RLH variant together with a PPC/PowerCoreCPCI-6750 Rev. 2.x or an MFIO-120, additional data lines are provided via the inter-TDM link.

The reason is that the capacitive load restriction of the H.110 bus allows to connect only PMC slot 2 to the H.110 backplane bus connector P4. Therefore, PMC-8260/DS1 modules mounted in PMC slot 1 of the reference CompactPCI carrier boards do not have a physical H.110 bus connection.

To allow sharing and exchanging TDM streams between the two PMC slots, the Force Computers reference CompactPCI carrier boards CPCI-6750 Rev. 2.x and MFIO-120 implement an on-board inter-TDM link connection. This link is a simple cross-connection of four TDM streams and one TDM clock signal and one frame pulse signal.

As is shown in the following figure the four output TDM streams C2 / D2_TXD and LDO_10 / 11 of the PMC-8260 / DS1-H110 in the upper PMC slot 2 position are cross-routed to the four TDM input streams C2 / D2_RXD and LDI_10 / 11 of the PMC-8260 / DS1-H110 in the lower PMC slot 1. The same applies to the clock output and the frame pulse output signals.

In the other direction (from the lower PMC slot 1 to the upper PMC slot 2) four additional TDM output streams and two clock signals are cross-routed via the carrier board to the four TDM input streams and two clock inputs of the upper PMC slot 2.

Note: The inter-TDM link is only available on H.110 variants but not on SC-F variants.

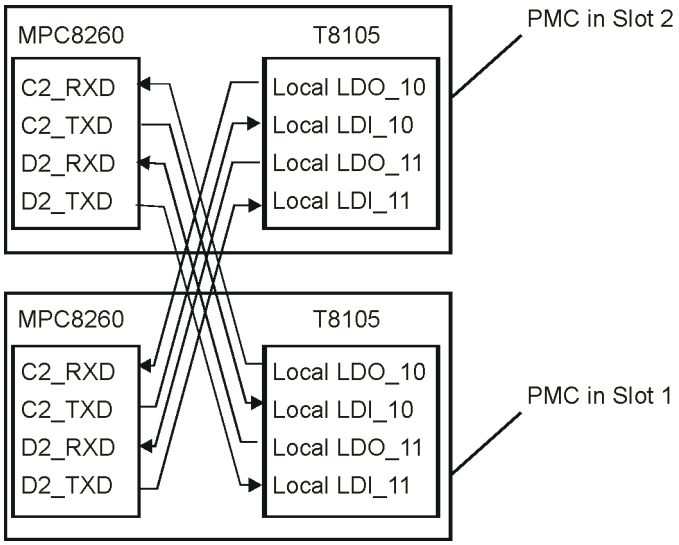


Figure 22: *Connection of Data Lines in Inter-TDM Link*

PowerQUICC II Port Functions on PMC-8260/DS1

The general purpose input/output (GPIO) pins of the PowerQUICC II can have several functions. The following table shows the functions used by the PMC-8260/DS1. Program these pins in your communications protocol driver according to the table below by setting the PowerQUICC II port registers. For further information, see the *MPC8260 PowerQUICC II User's Manual.*

Note: Since only PowerQUICC II receive clock pins are connected to the clock driver, PowerQUICC II TDM interfaces have to be configured to use common clock pins for RX/TX CLK and SYNC. These common clocks are fed into RXCLK and RSYNC.

Table 38: *Used Port Pins of SI1*

PowerQUICC II Port Pin	Used Function
PA08	L1RXD_A1
PA09	L1TXD_A1
PC31	L1RCLK_A1
PA06	L1RSYNC_A1
PD12	L1RXD_B1
PD13	L1TXD_B1
PC29	L1RCLK_B1
PD10	L1RSYNC_B1
PB14	L1RXD_C1
PD28	L1TXD_C1
PC27	L1RCLK_C1
PB12	L1RSYNC_C1
PB10	L1RXD_D1
PB11	L1TXD_D1
PC25	L1RCLK_D1
PD23	L1RSYNC_D1

Table 39: *Used Port Pins of SI2*

PowerQUICC II Port Pin	Used Function
PB06	L1RXD_A2
PB07	L1TXD_A2
PC19	L1RCLK_A2
PB04	L1RSYNC_A2
PB30	L1RXD_B2
PB31	L1TXD_B2
PC17	L1RCLK_B2
PB29	L1RSYNC_B2
PB26	L1RXD_C2
PB27	L1TXD_C2
PB17	L1RCLK_C2
PB24	L1RSYNC_C2
PB22	L1RXD_D2
PB23	L1TXD_D2
PA03	L1RCLK_D2
PB20	L1RSYNC_D2

A

Troubleshooting

Dear Customer,

the PMC-8260/DS1 is a highly sophisticated product. This chapter can be taken as a hint list for detecting erroneous system configurations and strange behaviors. However, it cannot replace a serious and sophisticated pre and post sales support during application development.

If it is not possible to fix a problem with the help of this chapter, contact your local sales representative or FAE for further support.

Problem	Possible Reason	Solution
During Power Up		
Primary Booter does not start	The ROM image area in the boot flash device is not empty (LEDs switch to green/off after POST).	Enforce primary booter execution via PCR register (see "Controlling Boot Process" page 4-13). Erase the ROM image
Application in ROM image area does not start / LEDs switch to yellow/yellow after green/off.	A software reset occured while executing the ROM image.	Configure ROM image so that firmware disables the watchdog before executing it (see "SYPCR Register" page 4-9). Add software watchdog support to application in ROM image area.
After Power Up		
LEDs remain yellow/yellow after power-up	Wrong switch settings	Verify switch settings (all switches must be in OFF position).
During Operation		
PCI error when trying to communicate with the firmware via the 4 MByte PCI window	The address translation of the PCI window is not enabled.	Allow firmware to enable address translation by setting bit 3 (value 8_{16}) in mailbox register 6 (offset 468_{16}) of PowerSpan. This can be done either before resetting the module or while the primary booter is running. Enable address translation by setting bits 31 and 30 (hex. value $C0000000_{16}$) in target image 1 control register (offset 100_{16}) of PowerSpan.

Index

A

Allowing firmware to access PowerSpan II ..4-13
Arbiters 5-9

B

Bus monitor 4-9, 5-6

C

CPM frequency 1-3
CPU frequency 1-3

D

Disabling POST 4-14
Disabling watchdog 4-13

E

Enabling POST 4-14
Enabling watchdog 4-13
Executing primary booter 4-13

F

FCC rules 2-6

I

I/O buffer 4-16
Inter-TDM link connections 6-10

P

PCI interface 1-3
PowerQUICC II frequency 1-3
PowerQUICC II Port Functions 6-11
PowerQUICC II version 1-3
PowerQUICCII serial I/O channel usage .. 3-4
PowerSpan 4-3
Primary booter 4-15
Protocol software packages 1-3

R

Resources used by primary booter 4-16
ROM image configuration section 4-10
ROM image execution 4-9, 4-13

S

SCbus specification 1-5
SDRAM performance 5-15
Serial EEPROM Contents 5-10
Skipping POST 4-13
SSRAM performance 5-16
Starting the primary booter 4-20
SYPCR register 4-9, 5-6
System Protection and Control 4-9

T

TDM channel connection 6-9

W

Watchdog 4-9

Product Error Report

Product:	Serial No.:
Date Of Purchase:	Originator:
Company:	Point Of Contact:
Tel.:	Ext.:
Address:	
Present Date:	
Affected Product: ❑ Hardware ❑ Software ❑ Systems	Affected Documentation: ❑ Hardware ❑ Software ❑ Systems
Error Description:	

This Area to Be Completed by Force Computers:

Date:

PR#:

Responsible Dept.: ❑ Marketing ❑ Production

Engineering ➠ ❑ Board ❑ Systems

☞ Send this report to the nearest Force Computers headquarter listed on the address page.

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP