MPC8360EA MDS Processor Board

User Manual

Rev. 0.1
12/2006

freescale™
semiconductor

Contents

Chapter 6
Working with the PIB

Chapter 7
Replacing Devices

Chapter 1
General Information

1.1 Introduction

This document describes the MPC8360EA MDS Processor Board; in its stand-alone operating mode, in its operating mode via a PCI slot in a PC, and in its operating mode on the “PowerQUICC MDS Platform I/O Board (PIB)”.

The MPC8360EA MDS Processor Board is an application devlopment system that provides a complete debugging environment for engineers developing applications for the MPC8360 series of Freescale processors.

The MPC8360EA is a cost-effective, highly integrated communications processor that addresses the needs of the networking, wireless infrastructure and telecommunications markets. Target applications include next generation DSLAMs, network interface cards for 3G base stations (Node Bs), routers, media gateways and high end IADs. The 8360EA extends current PowerQUICC II offerings, adding higher CPU performance, additional functionality, faster interfaces and interworking between ATM and Ethernet based protocols while addressing the requirements related to time-to-market, price, power, and package size. The MPC8360EA can be used for the control plane along with data plane functionality.

The MPC8360EA MDS Processor Board includes various peripherals, such as data input/output devices (GETH, USB, DUART), memories (DDR, SDRAM (optional), Serial EEPROM, FLASH and BCSR registers), and control switches and LED indicators.

Using its on-board resources and debugging devices, a developer is able to upload code, run the code, set breakpoints, display memory & registers and connect his own proprietary hardware to be incorporated into a target system that uses the MPC8360EA as a processor.

The software application developed for the MPC8360EA can be run in a "bare bones" operation (with only the MPC8360EA processor), or with various input or output data streams, such as from the GETH connection, PCI or the USB connections. Results can be analyzed using the *Code Warrior*® debugger in addition to using other methods for directly analyzing the input or output data stream. The BSP is built using the Linux OS.

This board can also be used as a demonstration tool for the developer. For instance, the developer's application software may be programmed into its Flash memory and run in exhibitions.

1.2 Working Configurations

1.2.1 Stand-Alone Mode (Host)

The MPC8360EA MDS Processor Board can be run in a stand-alone mode, like other application development systems, with direct connections to debuggers (via a JTAG/COP connector and JTAG/Parallel Port command converter), power supply, and the GETH, MiniAB USB and Dual RS-232 (DUART) connections. In this mode, the MPC8360EA MDS Processor Board acts as a Host.

1.2.2 PIB Combined Mode (as Host)

The MPC8360EA MDS Processor Board can be connected to the PIB - the Platform I/O Board, which allows it to be used in a back plane, and provides room and connections for additional modules. There are many modules, some of which are TDM or ATM modules. There can also be additional Processor Boards from the MPC83xx family, acting as Agents. This capability allows the MPC8360EA processor on the MPC8360EA MDS Processor Board to act as a Host for Agent processors in the MPC83xx family.

Power for the MPC8360EA MDS Processor Board in this case is provided via the PIB. The PIB also provides an additional 2x4 twisted pair for GETH signals to be connected via the back plane (if used). Optical signals via 2x SFP connectors for GETH on the front plane side of the PIB are also provided. The MPC8360EA MDS Processor Board can be connected to a PC in this configuration (via a parallel port connector), without needing an external command converter.

1.2.3 Agent Mode (in PC or PIB)

Using its PCI edge connector, the MPC8360EA MDS Processor Board can be inserted into a PC, or onto the PIB. Power and debugging are supplied from the PC (no command converter necessary). Other external connections are the same as in the Stand-Alone Mode. In this mode, the MPC8360EA MDS Processor Board acts as an Agent.

1.3 MPC8360EA MDS Processor Board

1.3.1 Features

- Supports MPC8360EA running up to 533MHz at 1.2V Core voltage.
- DDR 72-bit on SODIMM, at a rate up to 400MHz - *or* - 2 x DDR 40bit units on two SODIMM, each at a rate up to 400MHz.
- PCI edge connector interfaces with 32bit PCI bus (used when inserted in a PC, or as an agent on the PIB).
- Two 10/100/1000Mb/sec Ethernet Phys on GETH ports.
- USB 1.1 Transceiver.
- Dual RS232 transceiver on one DUART port.
- Local Bus interface:
  - 133MHz SDRAM memory (implemented using three units), 64Mbyte size with parity.
  - One 32Mbyte (expandable) Flash with 16bit port size in socket.
  - Address Latch and Buffers to support slow devices on the PIB Board.
  - Mictor Logic Analyzer Connectors on mux bus for evaluation only.
- Three Hi-speed Riser Connectors to enable connection to the PIB Board.
- Debug port access via dedicated 16-pin connector (COP), via PCI port or from parallel port interface on the PIB.
- One I2C port for EEPROM 256Kbyte, Real Time Clock (RTC) and SODIMM SPD EEPROM - A second I2C port is used to connect to the Board Revision Detect 256Kbyte EEPROM.

- Can function in one of three configurations:
  - Stand-alone.
  - As a PCI add-in card for a standard PC computer, or as an agent on the PIB.
  - PIB combined mode - development platform with Processor Board (as a Host) and PIB connected together.
- Board Control and Status Register (BCSR) implemented in Xilinx FPGA.
- Three power options:
  - Main 5V power is fed from external power supply for stand-alone mode.
  - Power from PC supply when acting as a PCI add-in card.
  - Power from the PIB when PIB and Processor Boards are combined.
- PCI add-in card form factor dimensions: 285mm x 106mm.

1.3.2 External Connections

The MPC8360EA MDS Processor Board interconnects with external devices via the following set of connectors:

- P1 - MiniAB USB connector.
- P2 - RJ45 (10-pin) for DUART signals.
- P3 - 32bit PCI Edge Connector.
- P4, P5, P6 - Three Logic Analyzer MICTOR Connectors.
- P7 - SMB RF Connector for external pulse generator.
- P8 - 16-pin COP/JTAG Connector.
- P9 - 16-pin header for FPGA In-System Programming.
- P10 - Voltage Input
- P11,P12,P13 - 300-pin FCI Expansion Connectors.
- J1,J2 - RJ45 8-pin Gigabit Ethernet Connectors.

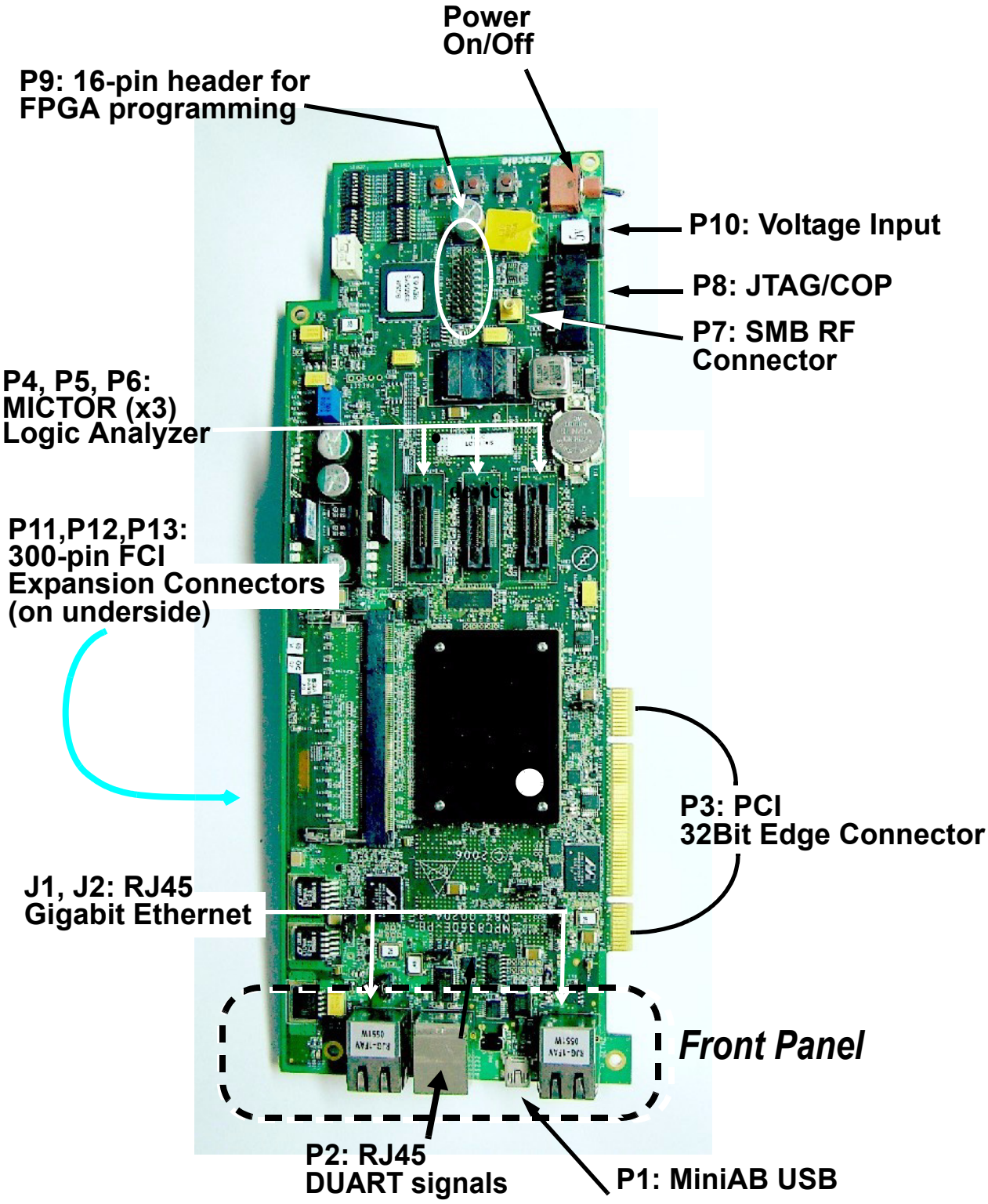


Figure 1-1. MPC8360EA MDS Processor Board External Connections

1.3.3 Block Diagram

The block diagram of the MPC8360EA MDS Processor Board is shown below in Figure 1-2.

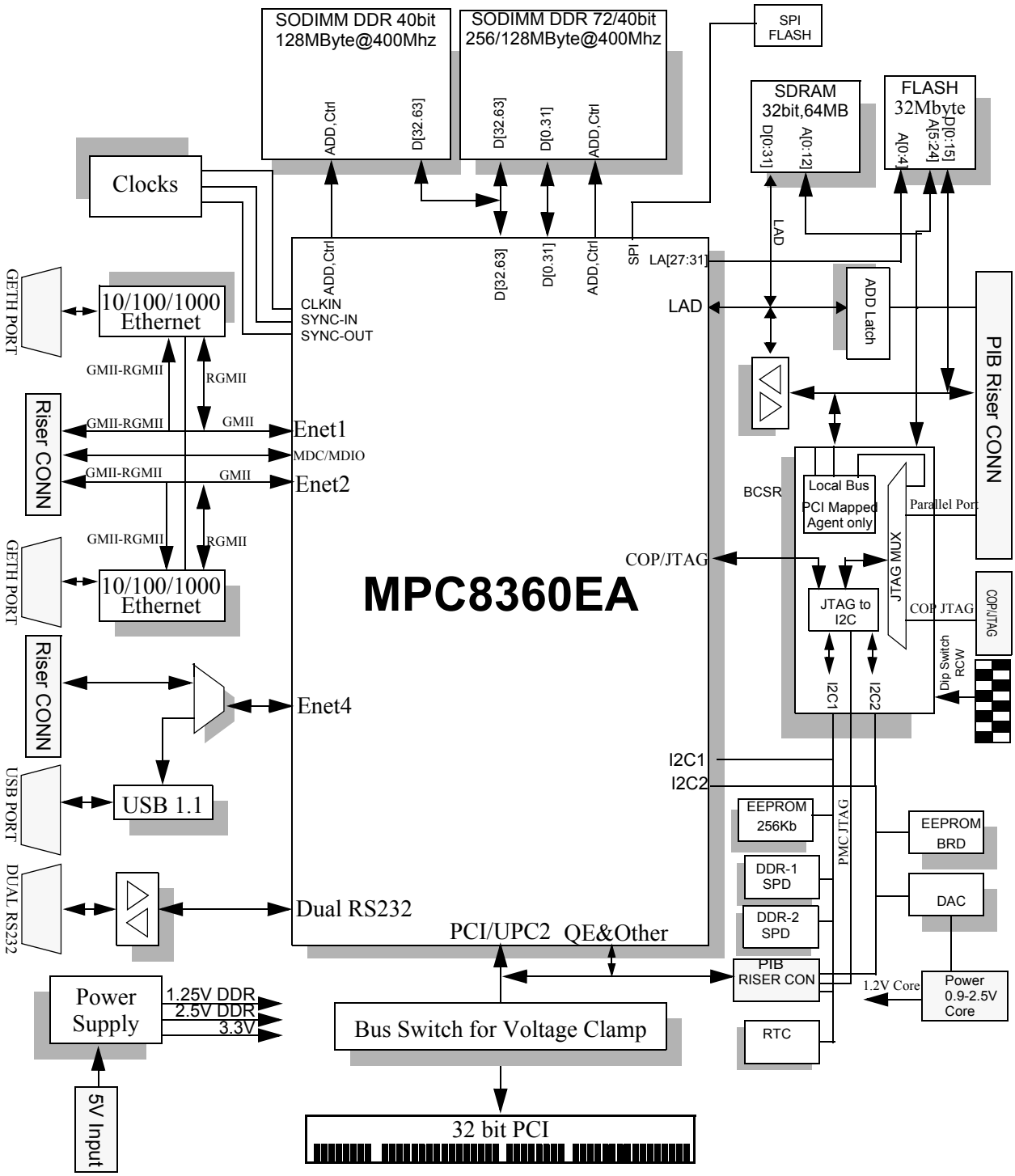


Figure 1-2. MPC8360EA MDS Processor Board Block Diagram

1.4 Definitions, Acronyms, and Abbreviations

ADS	Application Development System
BCSR	Board Control and Status Register
BRD	Board Revision Detect (I2C EEPROM)
BSP	Board Support Package
CCR	COP Control Register (FPGA)
COP	Common On-chip Processor (JTAG Debug Port)
CS	Chip Select
CW	*Code Warrior*® IDE for PowerPC
DAC	Digital-to-Analog Converter
DDR	Double Data Rate
DIP	Dual-In-Line Package.
DMA	Direct Memory Access
DUART	Dual UART
EEPROM	Electrical Erasable Programmable Memory
FCFG	Flash Configuration Select
FCI	Type of Riser Connector
FLASH	Non volatile reprogrammable memory.
FPGA	Field-Programmable Gate Array
GbE	Gigabit Ethernet
GETH	Gigabit Ethernet
GPCM	General Purpose Chip-select Machine
GPL	General Purpose Line
I2C	Philips Semi Serial Bus
LBIU	Local Bus Interface Unit
LED	Light Emitting Diode
lsb	least significant bit
MII	Media Independent Interface
GMII	General Media Independent Interface
JTAG	Joint Test Access Group
OTG	On-the-Go
PC	IBM-compatible Personal Computer
PCI	Peripheral Components Interconnect

Phy	Physical Layer
PIB	Platform I/O Board - expands the ADS functionality.
PSRAM	Pseudo-Static Random Access Memory
PSU	Power Supply Unit
RCW(L,H)	Reset Configuration Word (Low/High)
RGMII	Reduced General Media Independent Interface
RTC	Real Time Clock
SDRAM	Synchronous Dynamic Random Access Memory
SMB	Type of Mini-RF connector
SODIMM	Mini DIMM Form Factor
SPD	Serial Present Detect
TSEC	Triple Speed Ethernet Controller
ULPI	UTMI+ Low Pin Interface
UPM	User Programmable Machine
USB	Universal Serial Bus
ZD	Zero Delay clock buffer, with internal PLL for skew elimination

1.5 Related Documentation

- MPC8360EA HW Specification
- MPC8360EA User's Manual
- PowerQUICC MDS Platform I/O Board User's Manual
- MPC8360EA Hardware Getting Started

1.6 Specifications

The MPC8360EA MDS Processor Board specifications are given in Table 1-1.

Table 1-1. MPC8360EA MDS Processor Board specifications

CHARACTERISTICS	SPECIFICATIONS
Power requirements	Stand-Alone Mode: 5V @ 3A external DC power supply PIB Combined Mode: Power supplied by PIB Working in PC: Power supplied by PC

Table 1-1. MPC8360EA MDS Processor Board specifications

CHARACTERISTICS	SPECIFICATIONS
MPC8360EA processor	Internal clock runs at 533MHz @ 1.2V (note that the board supports any working frequency that the MPC8360EA supports. To change the working frequency, see Section 5.3)
Memory: Two DDR busses	2 x 128MB space 40bit wide (32bits data, 8 bits ECC) in two SODIMM-200's . Data rate 400MHz.
Local Bus: SDRAM (Optional)	64MB space 32bit wide + 4bit parity implemented in three SDRAM parts. 133MHz clock.
Buffered Memory (Flash on socket):	32MB space 16bits wide.
BCSR on FPGA	16-registers, 8bits wide.
Expansion	Four banks with 16bit- Address bus, 16bit- Data bus connected to riser connectors
Operating temperature	$0^{O}C$ - $70^{O}C$
Storage temperature	$-25^{O}C$ to $85^{O}C$
Relative humidity	5% to 90% (non-condensing)
Dimensions (according to PCI 64-bit Add-in-card form factor): Length Width Height	 285 mm 106 mm 16 mm

Chapter 2
Hardware Preparation and Installation

This chapter provides unpacking instructions, hardware preparation, and installation instructions for the MPC8360EA MDS Processor Board, including all three configurations: Stand-Alone, PIB Combined Mode, and Agent Mode (either on the PIB, or inserted in a PC). For more details on hardware preparation, see the "Hardware Getting Started" document for the MPC8360EA MDS Processor Board.

2.1 Unpacking Instructions

NOTE

If the shipping carton is damaged upon receipt, request carrier's agent to be present during unpacking and inspection of equipment.

CAUTION

AVOID TOUCHING AREAS OF INTEGRATED CIRCUITRY; STATIC DISCHARGE CAN DAMAGE CIRCUITS.

1. Unpack equipment from shipping carton.
2. Refer to packing list and verify that all items are present.
3. Save packing material for storing and reshipping of equipment.

2.2 Installation Instructions

Do the following ***in the order indicated*** to install the MPC8360EA MDS Processor Board properly:

1. Verify that Jumpers and Switches are in default positions. For default positions, see the "Hardware Getting Started Guide" document for the MPC8360EA MDS Processor Board.
2. Determine in which working configuration you will operate the MPC8360EA MDS Processor Board:
   - Stand-Alone - continue from Section 2.2.1
   - PIB Combined Mode, with the PIB Board - continue from Section 2.2.2
   - Agent Mode - continue from Section 2.2.3

2.2.1 For Stand-Alone Mode

1. *For Stand-Alone Mode only*: Connect the four plastic spacers. See Figure 2-1.
2. Connect external cables in accordance with your development needs.
3. Connect PSU (to P10), and turn the power on-off switch to ON.
4. Verify that LD1 and LD2 turn on and turn off (see Figure 2-2 for location). They should be on for only a few moments. This indicates that the board has successfully completed the boot-up sequence.

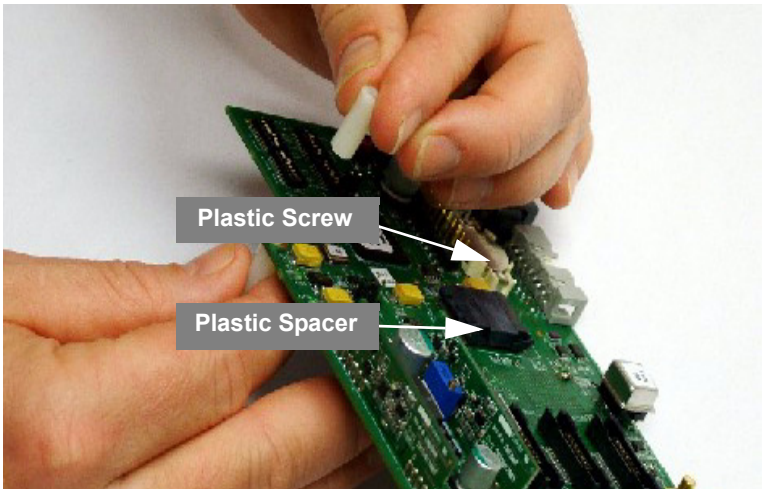


Figure 2-1. Connecting Plastic Spacers

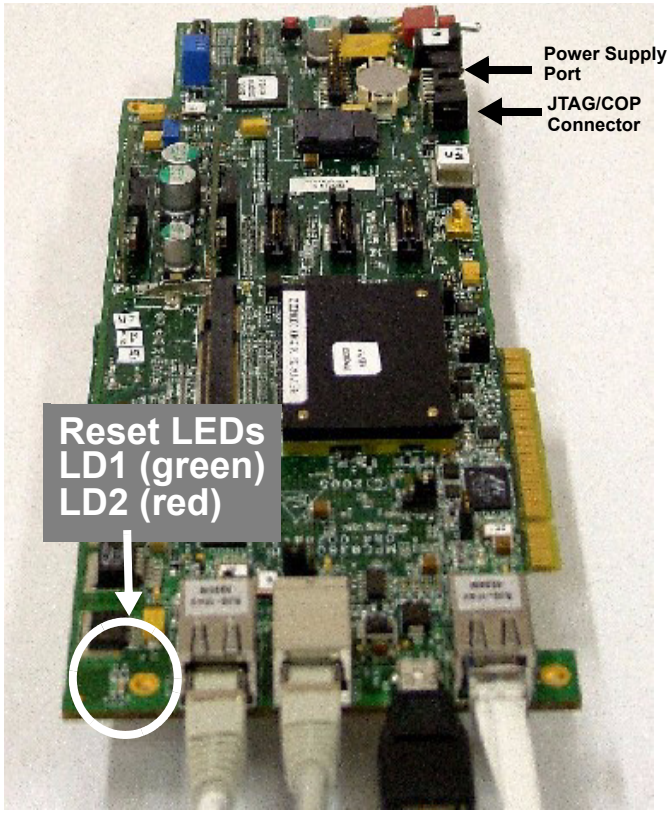


Figure 2-2. Boot-Up sequence: LD1 and LD2 (turn on, then off)

2.2.2 For PIB Combined Mode

1. Remove protective covers from the 300-pin connectors on the bottom side of the processor board (See Figure 2-3.).
2. Remove protective covers from the 300-pin connectors on the PIB board (see Figure 2-4).

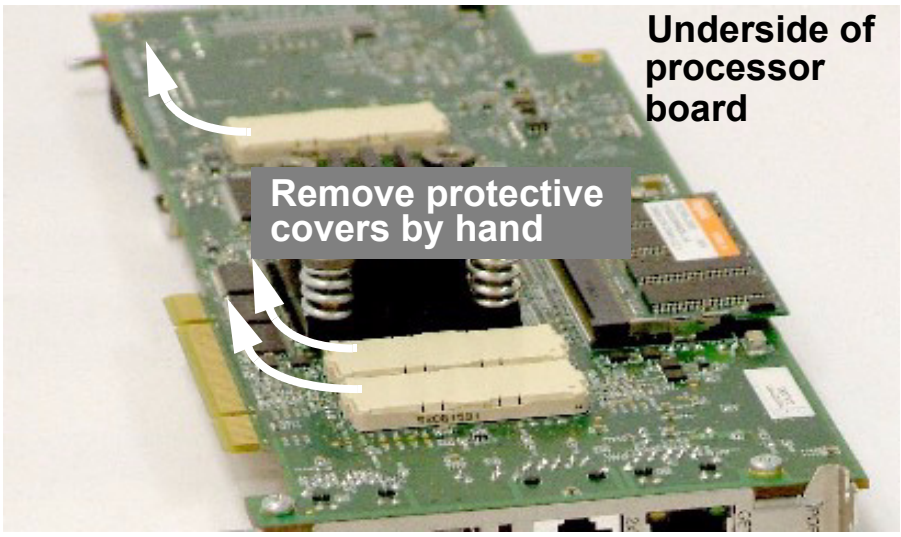


Figure 2-3. Remove Protective Covers from 300-pin connectors (underside of MPC8360EA MDS Processor Board shown)

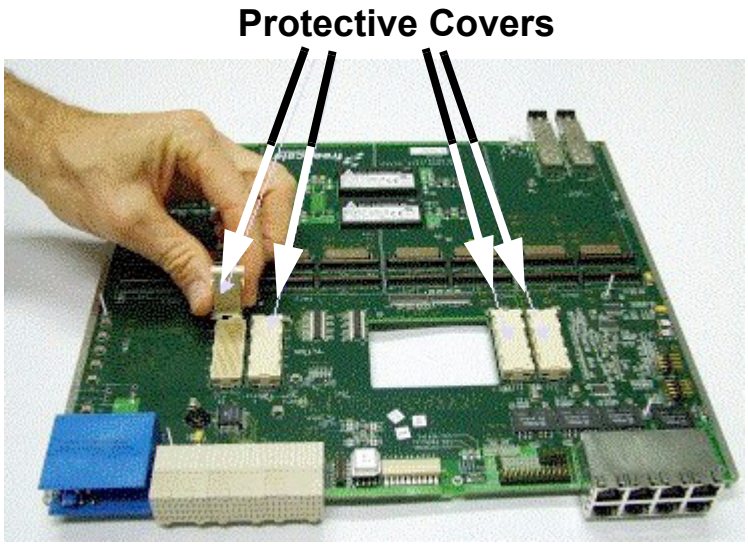


Figure 2-4. Remove Protective Covers from 300-pin connectors

3. Connect processor board to PIB board as shown in Figure 2-5.
4. Ensure a tight fit by pressing down on the processor board **by hand only** until the pins engage (see Figure 2-5.)
5. Manually fasten the four screws as shown in Figure 2-6.

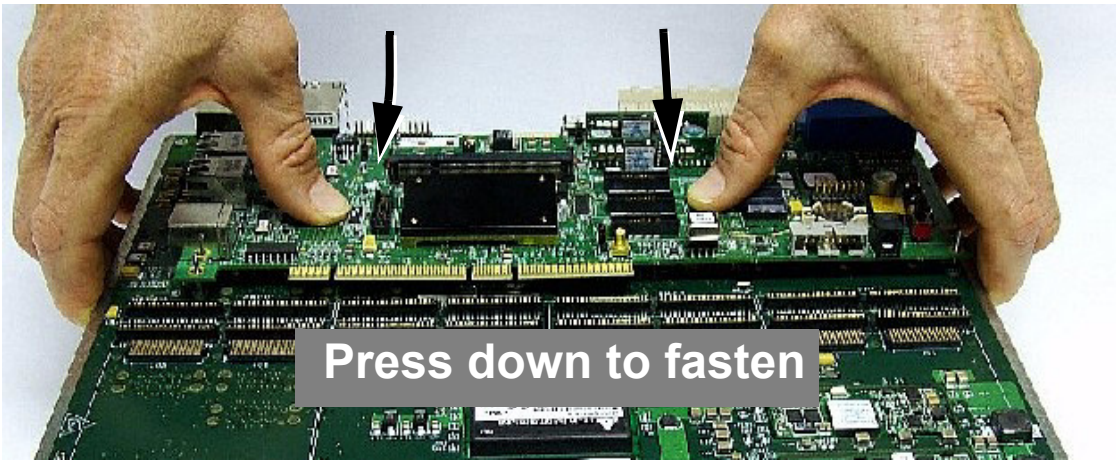


Figure 2-5. Connect Processor board to PIB and press down with fingers [1]

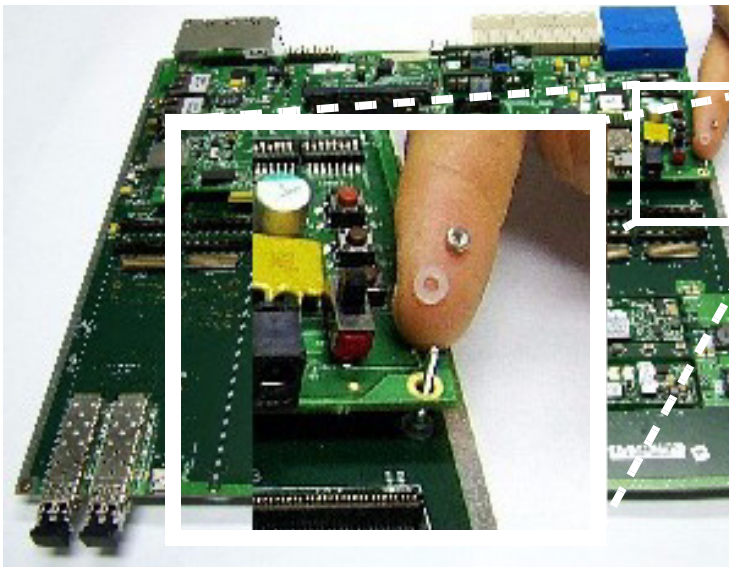


Figure 2-6. Fasten the four tightening screws

1. Although an MPC8349 Processor Board is shown, this step is the same for the MPC8360EA

6. If you will be working with a back plane, and wish GETH signals to traverse either the back plane connection, or the front plane optical connection, connect the two GETH sockets on the MPC8360EA MDS Processor Board with sockets on the PIB board as shown in Figure 2-7 and Figure 2-8.

   Note that if you do not do this, you can still connect GETH cables directly to the Processor board's sockets, if they are accessible in your laboratory configuration.

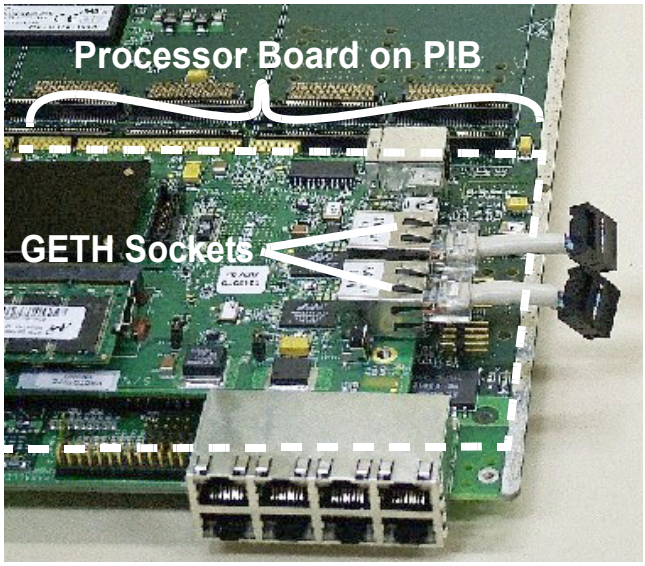


Figure 2-7. Insert GETH interconnecting cables to GETH sockets on Processor board

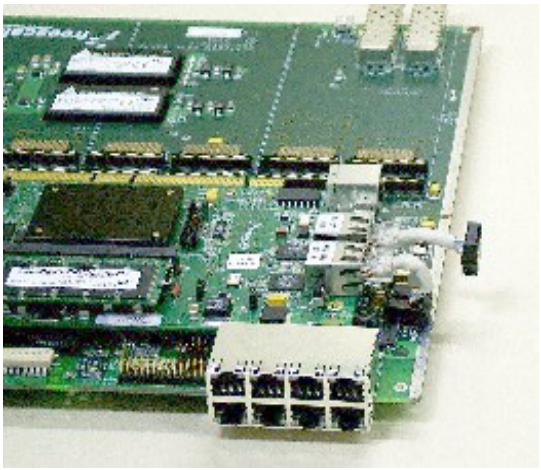


Figure 2-8. Connect GETH interconnecting cables to sockets on PIB

7. Connect the power supply to the voltage input as shown in Figure 2-9.

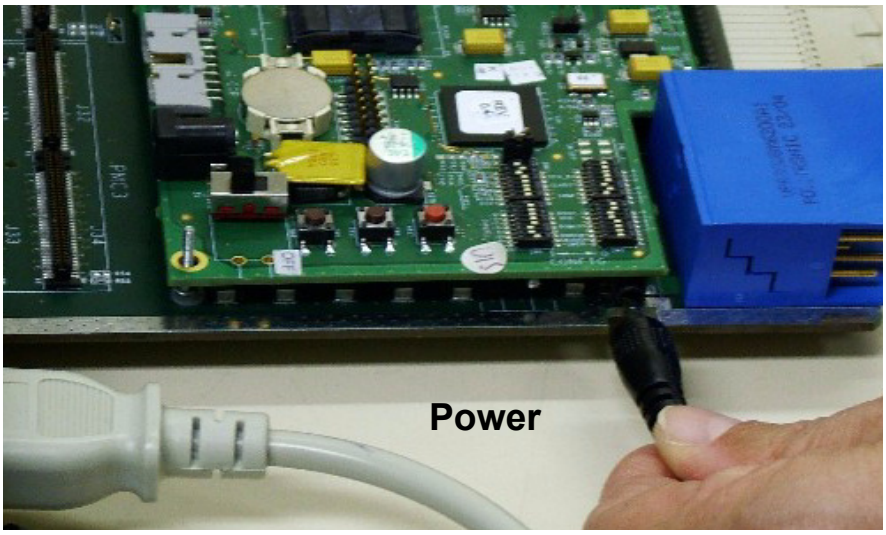


Figure 2-9. Connecting Power input to the PIB

8. If you wish to work with a module connected to a PCI adaptor, follow the illustrations in Figure 2-10, Figure 2-11, and Figure 2-12 to connect these items to the PIB.

   Note that the ATM card can only be inserted in the PMC0 and PMC1 (PMC0 shown). The PCI adaptors can be inserted in any section, for up to 4 PCI adaptors, if space allows.

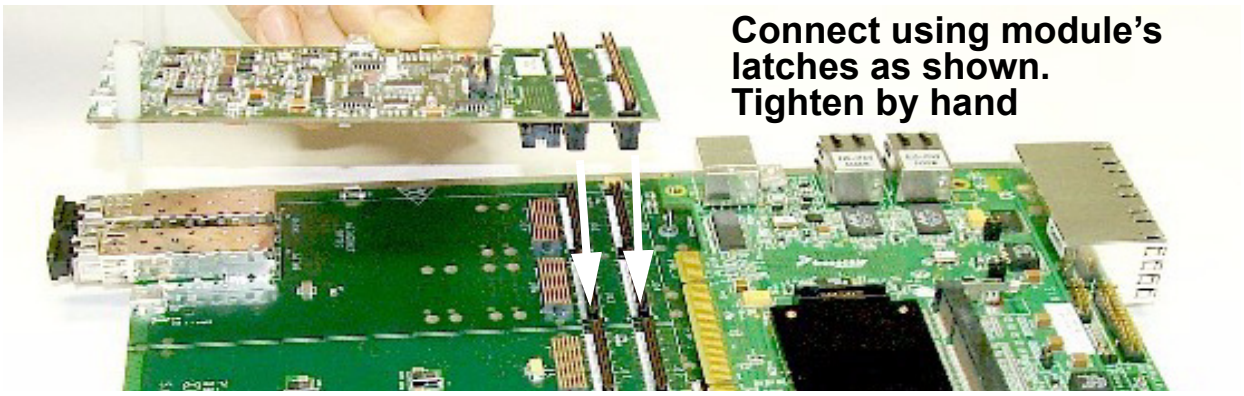


Figure 2-10. Connecting module card to PIB

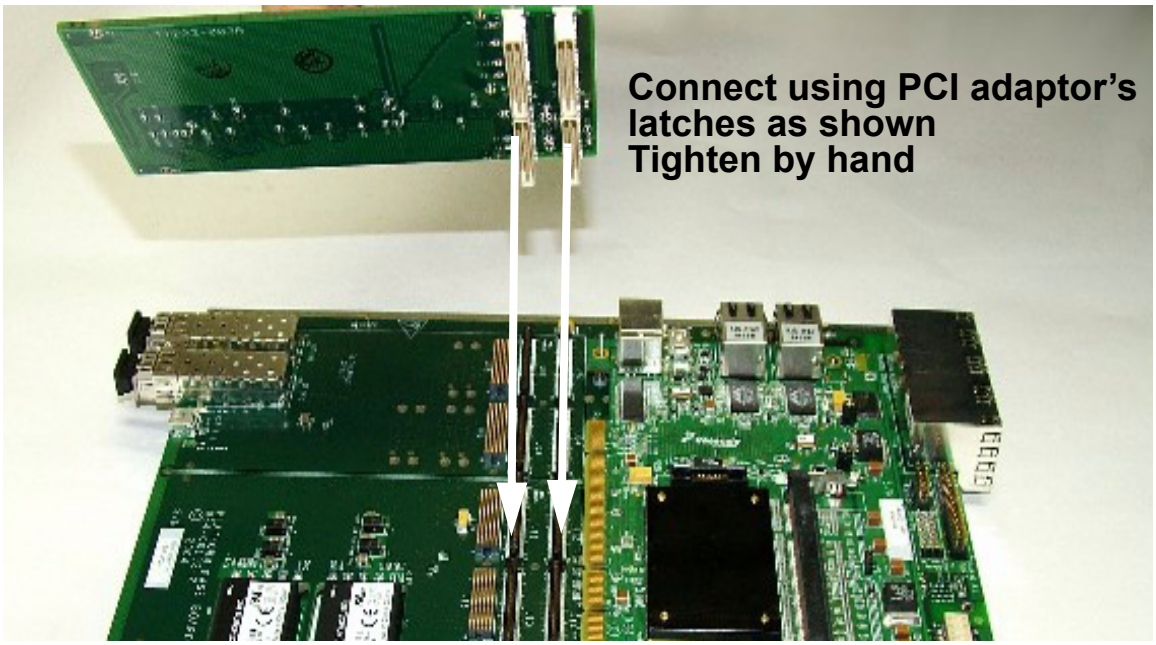


Figure 2-11. Connecting PCI adaptor to PIB

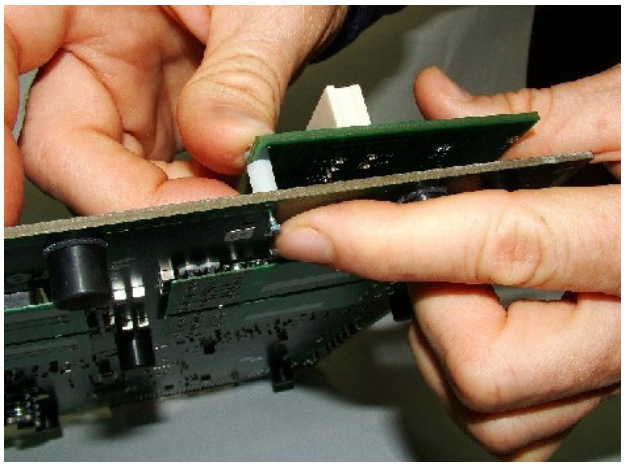


Figure 2-12. Inserting spacers between PCI card and PIB

9. The fully assembled PIB-Processor board is shown in Figure 2-13, which also shows the PIB external connections relevant when the MPC8360EA is used.

   All external connections of the Processor board are active when the Processor board is installed on the PIB, except the voltage input (receives power from the PIB power input, or the back plane only), and the JTAG/COP connection (P8), which is replaced by the parallel port connection to a PC.

   In Figure 2-13, three PCI adaptors and one additional module are shown installed on the PIB. The PCI cards are ready to receive any PCI- compatible board, including an 83xx Processor board, installed in this case in the same manner as they are in a PC. Using this system, these boards (up to three) function as agents, while the Processor board already installed functions as a host. This allows you to take advantage of the parallel processing capabilities of the 83xx line of products.
10. Operate *Code Warrior*® to verify that the processor board has been installed properly.
11. Connect power cable to the PIB, and external cables in accordance with your development needs.
12. Verify that LD1 and LD2 turn on and then turn off (see Figure 2-2 for location). They should be on for only a few moments. This indicates that the board has successfully undergone the boot-up sequence.

.

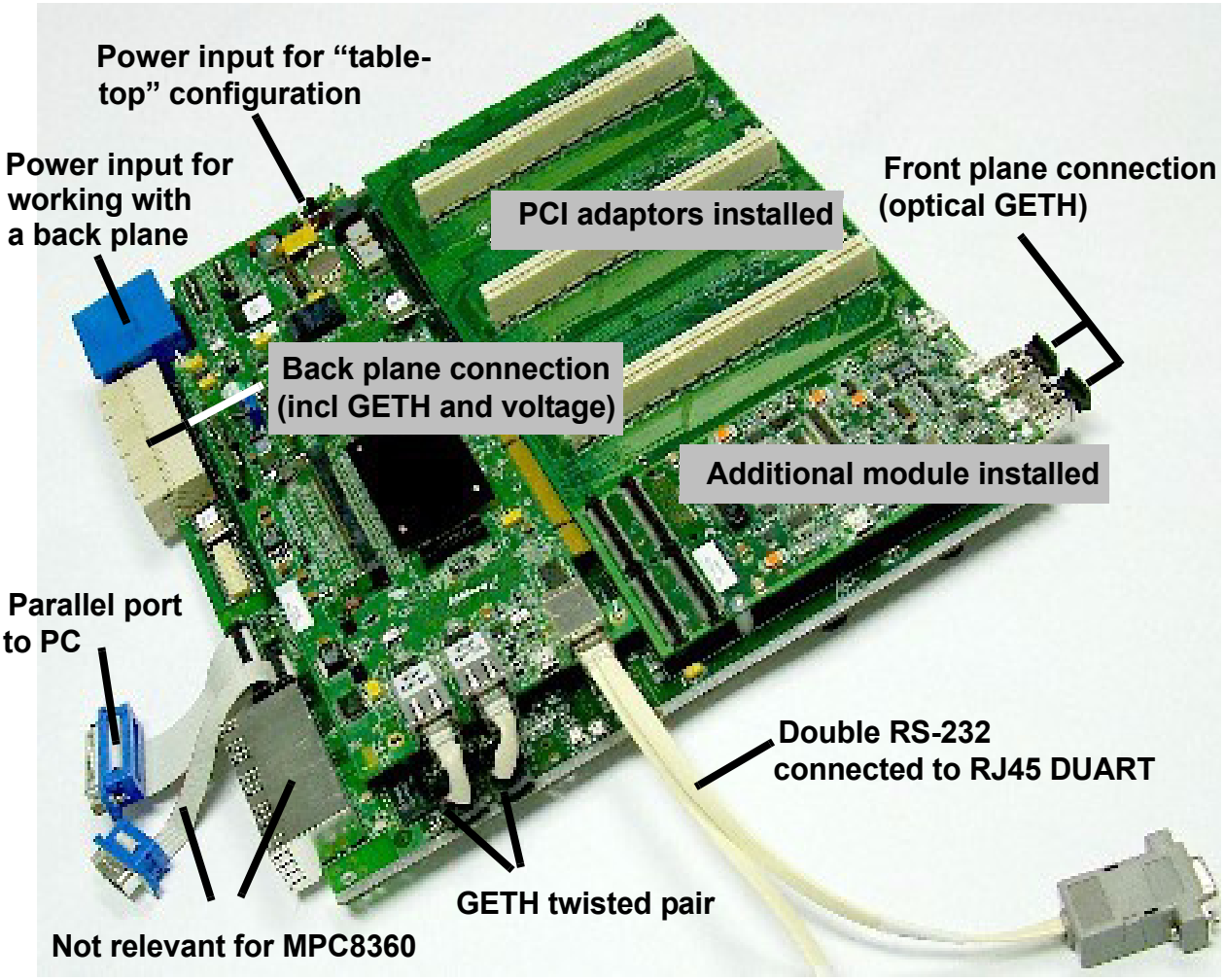


Figure 2-13. Fully Assembled Combined system: PIB, Processor Board, additional module, and PCI cards

2.2.3 For Agent Mode

1. **Installed in a PC**: Insert the MPC8360EA MDS Processor Board into a PC, using its PCI edge connector.

   As an agent on the PIB: Insert PCI adaptor or adaptors into the PIB as shown in Figure 2-11 and Figure 2-12, then insert the Processor Board into an adaptor, using its PCI edge connector, as shown in Figure 2-13.

2. Operate *Code Warrior*® to verify that the processor board has been installed properly.
3. Connect external cables in accordance with your development needs.
4. Verify that LD1 and LD2 turn on and then turn off (see Figure 2-2 for location). They should be on for only a few moments. This indicates that the board has successfully undergone the boot-up sequence.

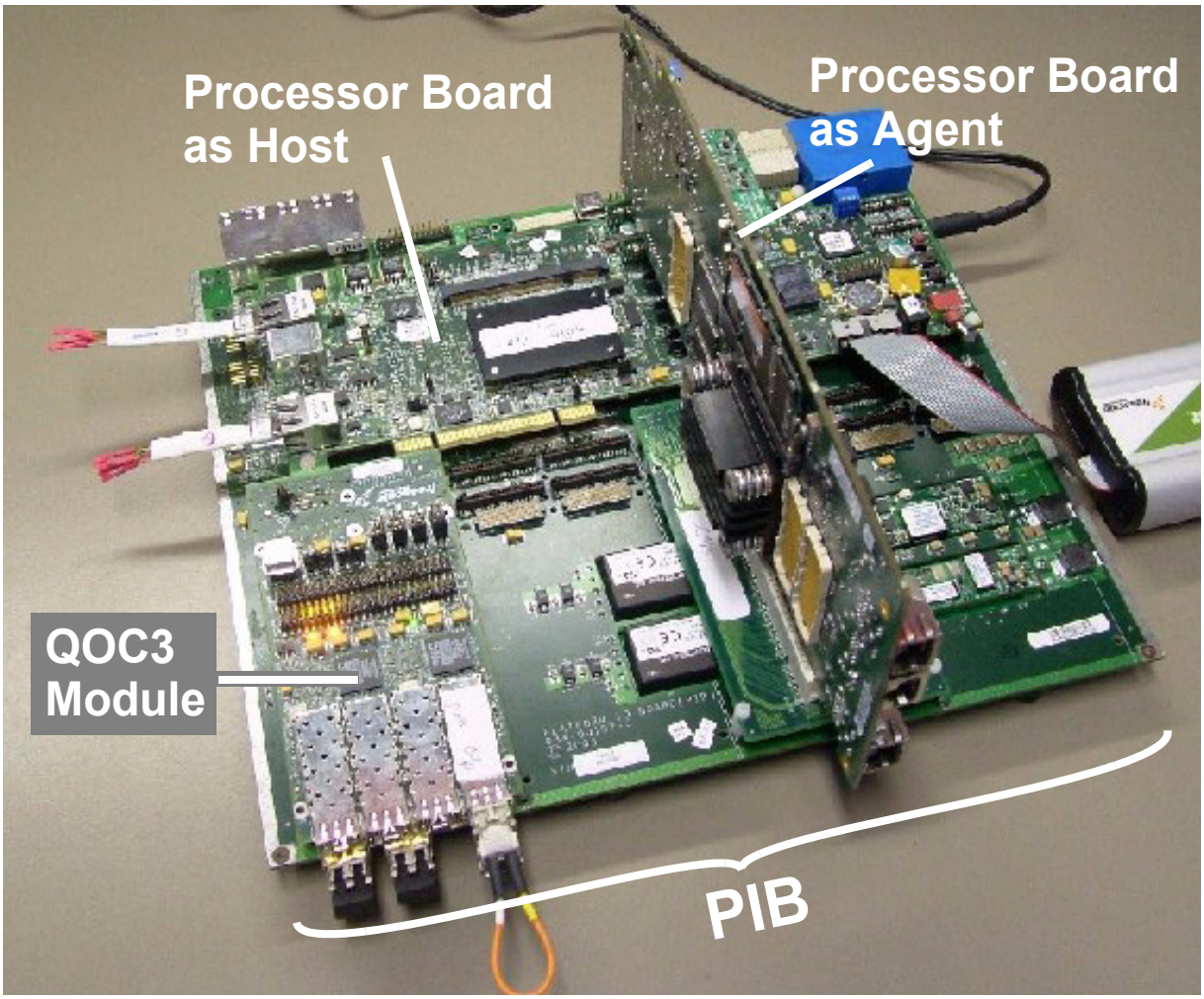


Figure 2-14. PIB with one Processor Board as Host, another as Agent, and a QOC3 module

Chapter 3
Memory Map

3.1 MPC8360EA MDS Processor Board Mapping

The MPC8360EA Memory Controller governs all access to the processor memory slaves. Consequently the memory map may be reprogrammed according to user needs. The memory map defined in Table 3-1 is only a recommendation. The user can choose to work with alternative memory mapping. It should be noted that the described mode is supported by Metrowerks' *Code Warrior*® debug tool.

After performing Hard Reset, the debug host may initialize the memory controller via the JTAG/COP connector so this allows additional access to bus addressable peripherals. The DDR72, 2xDDR36, SDRAM and FLASH memory respond to all types of memory access - program/data and Direct Memory Access (DMA).

The following table (Table 3-1) presents the Memory Map of the MPC8360EA MDS Processor Board. It includes an address range for each target that utilizes memory, the "target" itself on the board, the specific device that is used to implement the target's function, the amount of memory allotted to the target (volume in bytes), and its port size (in bits). Note the information on the DDR SDRAM. There are two main options: using one 72-bit device, or one or two 36-bit devices. The one 72-bit device can be used to implement a memory capacity of 256MB, or 1GB. Alternatively, a memory capacity of 256MB (128 + 128) can be implemented using two 36-bit devices.

Table 3-1. MPC8360EA MDS Processor Board Memory Map

No.	Address Range	Target	Device Name	Volume in Bytes	Port Size in Bits
0	00000000 - 0FFFFFFF (for 256MB volume) (00000000 - 3FFFFFFF) (for 1GB volume)	**DDR SDRAM :** Main SODIMM MEMC1CS0, MEMC1CS1 72-bit	WV3HG32M72EEU403PD4GG with ECC (for 256MB space) ***- or -*** WV3HG128M72EEU-PD4 with ECC (for 1GB space) (optional, not supplied with board)	256MB (1GB)	64+8 ECC
1	00000000 - 0FFFFFFF (for 128MB volume)	Main SODIMM MEMC1CS0, MEMC1CS1 40-bit	WV3HG32M72EEU403PD4GG w/o ECC	128MB	32+8 ECC
2	00000000 - 0FFFFFFF (for additional 128MB volume)	Second SODIMM MEMC2CS0,MEMC2CS1 40-bit	WV3HG32M40SEU-PD4	128MB	32+8 ECC
3	10000000 - 7FFFFFFF	Empty Space	-	3GB	-
4	80000000 - 9FFFFFFF	PCI1	Inbound/Outbound window	512MB	32
5	A0000000 - DFFFFFFF	Empty Space	-	1GB	-

Table 3-1. MPC8360EA MDS Processor Board Memory Map (continued)

No.	Address Range	Target	Device Name	Volume in Bytes	Port Size in Bits
6	E0000000 - E01FFFFF	8360	Internal Memory Register Space	2MB	32
7	E0200000 - EFFFFFFF	Empty Space	-	~256MB	-
8	F0000000 - F3FFFFFF or (F0000000 - F7FFFFFF)	**Local Bus SDRAM:** (optional) on CS2	MT48LC16M16A2TG-6A x 1 (with MT48LC32M16A2TG-7E for parity) or MT48LC16M16A2TG-6A x 2	64MB or (128MB)	32+8 parity
9	F4000000 - F7FFFFFF	Empty Space	in case SDRAM is only 64MB	64MB	-
10	F8000000 - F8007FFF	BCSR on CS1	Xilinx FPGA	32KB	8
11	F8008000 - F800FFFF	PIB CS4	(see PIB documentation)	32KB	
12	F8010000 - F8017FFF	PIB CS5	(see PIB documentation)	32KB	
13	F8018000 - F801FFFF	PIB CS3	(see PIB documentation)	32KB	
14	F8020000 - FDFFFFFF	Empty Space	-	~100MB	-
15	FE000000 - FEFFFFFF	FLASH on CS0	S29GL256M	16MB	16
16	FE800000 - FFFFFFFF	Empty Space	-	24MB	-

3.2 Configuration Registers Mapping

The table below shows how to initialize the registers in the MPC8360EA. It shows the register name, its address, and what value must be written to that register for proper initialization. There are several instances that require repeat actions (such as writing FF to LSDMR several times). These must be carried out as described.

Table 3-2. MPC8360EA MDS Processor Board Configuration Registers

Register Name	Register Address	Register Value	Description
IMMRBAR	0xff400000	0xE0000000	
Local Access Window FLASH			
LBLAWBAR0	0xE0000020	0xFE000000	
LBLAWBAR0	0xE0000024	0x80000017	16M
Local Access Window BCSR			
LBLAWBAR1	0xE0000028	0xF8000000	
LBLAWBAR1	0xE000002C	0x8000000E	32K
Local Access Window SDRAM			

Table 3-2. MPC8360EA MDS Processor Board Configuration Registers (continued)

Register Name	Register Address	Register Value	Description
LBLAWBAR2	0xE0000030	0xF0000000	
LBLAWBAR2	0xE0000034	0x80000019	64M
LBLAWBAR2	0xE0000034	0x8000001A	128M
Local Access Window PIB			
LBLAWBAR3	0xE0000038	0xF8008000	
LBLAWBAR3	0xE000003c	0x8000000F	64K
PIB (CS4)			
OR4	0xE0005024	0xFFFFE9F7	32K
BR4	0xE0005020	0xF8008801	Port-size=8bit, MSEL=GPCM
PIB (CS5)			
OR5	0xE000502C	0xFFFFE9F7	32K
BR5	0xE0005028	0xF8010801	Port-size=8bit, MSEL=GPCM
Local Access Window PCI			
PCILAWBAR0	0xE0000060	0x80000000	
PCILAWBAR0	0xE0000064	0x8000001C	512M
QE Secondary Bus Access Windows			
LBMCSAR	0xE0001800	0x000F0000	
LBMCEAR	0xE0001840	0x000F3FFF	
LBMCAR	0xE0001880	0x00000001	
LBIU registers			
LCRR	0xE00050D4	0x00000004	
Flash (CS0)			
BR0	0xE0005000	0xFE001001	Port-size=16bit, MSEL=GPCM
OR0	0xE0005004	0xFF006FF7	16M
BCSR (CS1)			
BR1	0xE0005008	0xF8000801	Port-size=8bit, MSEL=GPCM
OR1	0xE000500C	0xFFFFE9F7	32K
SDRAM (CS2)			
OR2	0xE0005014	0xFC006901	64M
BR2	0xE0005010	0xF0001861	Port-size=32-bit, MSEL=SDRAM

Table 3-2. MPC8360EA MDS Processor Board Configuration Registers (continued)

Register Name	Register Address	Register Value	Description
LSRT	0xE00050A4	0x3F000000	LSRT - SDRAM refresh, timer period.
MRTPR	0xE0005084	0x20000000	
LSDMR	0xE0005094	0x0063B723	
SDRAM			
LSDMR	0xE0005094	0x2863B723	
	0xF0000000	0x000000FF	
LSDMR	0xE0005094	0x0863B723	
	0xF0000000	0x000000FF	
	0xF0000000	0x000000FF	
	0xF0000000	0x000000FF	
	0xF0000000	0x000000FF	
	0xF0000000	0x000000FF	
	0xF0000000	0x000000FF	
	0xF0000000	0x000000FF	
	0xF0000000	0x000000FF	
	0xF0000000	0x000000FF	
LSDMR	0xE0005094	0x1863B723	
	0xF00000cc	0x000000FF	
LSDMR	0xE0005094	0x4063B723	
#LSRT	0xE00050A4	0x20000000	LSRT - SDRAM refresh, timer period
MRTPR	0xE0005084	0x20000000	MRTPR - Refresh timers prescaler - <Period of the refresh timers input clk> = <sys_clk> / <PTP=0x20>

Table 3-3. Init DDR values for DDR 72-bit

Register Name	Register Address	Register Value	Description
DDRLAWBAR0	0xE00000A0	0x00000000	
DDRLAWAR0	0xE00000A4	0x8000001B	256M
First DDR Controller Registers SODIMM initializations			
CLK_CNTL	0xE0002130	0x82000000	SS_EN=1 ; CLK_ADJST = 2
#CS0_BNDS	0xE0002000	0x00000007	first 128MB
#CS0_CONFIG	0xE0002080	0x80000101	

Table 3-3. Init DDR values for DDR 72-bit (continued)

#CS1_BNDS	0xE0002008	0x0008000f	last 128MB
#CS1_CONFIG	0xE0002084	0x80000101	
#TIMING_CFG_1	0xE0002108	0x37343321	
#TIMING_CFG_2	0xE000210C	0x00000800	
#DDR_SDRAM_CFG	0xE0002110	0x42008000	32_BE=0 => 64-bit bus is used, 2T_EN=1
#DDR_SDRAM_MODE	0xE0002118	0x20000162	
#DDR_SDRAM_INTERVAL	0xE0002124	0x045B0100	
delay before enable			
DDR_SDRAM_CFG	0xE0002110	0xC2008000	

Table 3-4. Init DDR values for 2 x 36-bit

Register Name	Register Address	Register Value	Description
Local Access Window DDR SDRAM			
DDRLAWBAR0	0xE00000A0	0x00000000	
DDRLAWAR0	0xE00000A4	0x8000001A	first 128M
Local Access Window Secondary DDR SDRAM			
SDDRLAWBAR0	0xE00000E0	0x08000000	
SDDRLAWAR0	0xE00000E4	0x8000001A	first 128M
QE Secondary Bus Access Windows			
SDMCSAR	0xE0001804	0x00008000	
SDMCEAR	0xE0001844	0x0000FFFF	
SDMCAR	0xE0001884	0x00000001	
First DDR Controller Registers SODIMM initializations			SWISSBIT Half SODIMM
CLK_CNTL	0xE0002130	0x82000000	SS_EN=1 ; CLK_ADJST = 2
CS0_BNDS	0xE0002000	0x00000007	first 128MB
CS0_CONFIG	0xE0002080	0x80000102	13 row bits, 10 columns bits
TIMING_CFG_1	0xE0002108	0x37343321	
TIMING_CFG_2	0xE000210C	0x00000800	
DDR_SDRAM_CFG	0xE0002110	0x42008000	32_BE=0 => 64-bit bus is used, 2T_EN=1

Table 3-4. Init DDR values for 2 x 36-bit (continued)

DDR_SDRAM_MODE	0xE0002118	0x20000163	
DDR_SDRAM_INTERVAL	0xE0002124	0x045B0100	
delay before enable			
DDR_SDRAM_CFG	0xE000D130	0xC2008000	
Second DDR Controller Registers			
CAUTION CAUTION!!!!SDDRIOE (SECONDARY DDR IO ENABLE) MUST BE SET TO 1 !!! # USE A CONFIGURATION WORD WHICH INCLUDES SDDRIOE=1 OR SET SDDRIOE=1 THROUGH BCSR7[3] (AND CNFLOCK(BCSR13[3]) AND PORE-SET(BCSR11[4]))			
CLK_CNTL	0xE000D130	0x82000000	
CS0_BNDS	0xE000D000	0x0008000B	first 64MB
CS0_CONFIG	0xE000D080	0x80000101	13 row bits, 9 columns bits
CS1_BNDS	0xE000D008	0x000C000F	last 64MB
CS1_CONFIG	0xE000D084	0x80000101	13 row bits, 9 columns bits
#TIMING_CFG_1	0xE000D108	0x37343321	
#TIMING_CFG_2	0xE000D10C	0x00000800	
#DDR_SDRAM_CFG	0xE000D110	0x420C8000	
#DDR_SDRAM_MODE	0xE000D118	0x20000163	
#DDR_SDRAM_INTERVAL	0xE000D124	0x045B0100	
delay before enable			
DDR_SDRAM_CFG	0xE000D110	0xC20C8000	

Chapter 4
Controls and Indicators

This chapter describes controls and indicators of the MPC8360EA MDS Processor Board, which includes switches, jumpers, LEDs, and push buttons.

4.1 Switches

Figure 4-1 below shows the locations of the DIP Switches. Note that when “ON”, the value of the switch is zero.

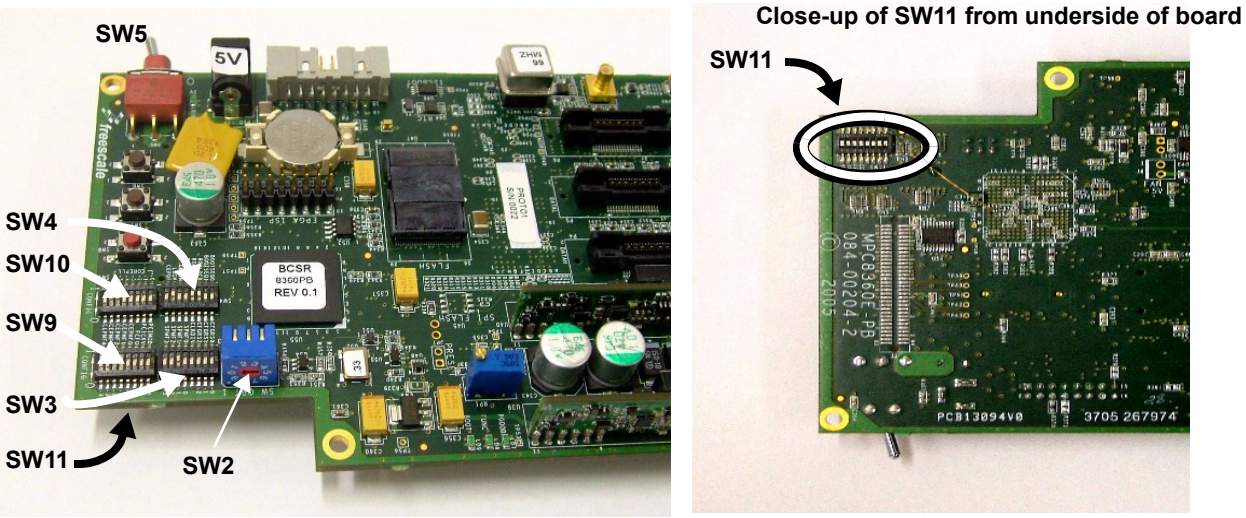
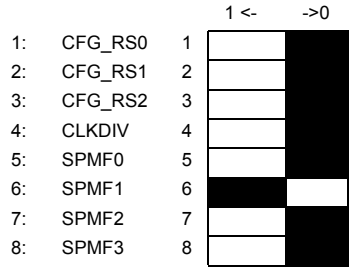


Figure 4-1. MPC8360EA MDS Processor Board Switches Locations

Descriptions of settings for the DIP switches are described below. For more detailed descriptions of the bits and fields, see the *MPC8360 User Manual*:

SW3



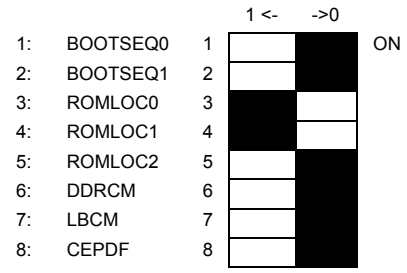
The "On" DIP Switch position corresponds to a signal value of “zero”.

SW3.1-SW3.3
CFG_RS sets the Reset Configuration Words Source
Default setting: '000' = RCW is retrieved from the local bus (see also Section 5.1.1)

SW3.4
CLKDIV selects the relationship between CLKIN and PCI_SYNC_OUT
If MPC8360EA is configured as a PCI Agent (factory setting) then CLK_DIV is low. If MPC8360EA is host, CLK_DIV should be high.
Default setting: Low

SW3.5-SW3.8
SPMF selects the System PLL Multiplication Factor
Default setting: '0100'
clock ratio: csb_clk/CLKIN = 4 (csb_clk = 266MHz) or csb_clk/PCI_CLK = 4

SW4



SW4.1-SW4.2: Configuration Boot Sequencer
Boot sequencer loads configuration data from the serial ROM
Default setting: '00' = disables access to I2C ROM

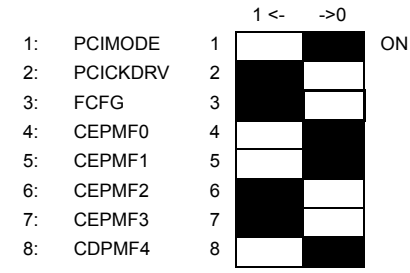
SW4.3-SW4.5: Boot ROM location
Default setting: '110'; provides Flash boot on local bus

SW4.6 DDR: Clock Mode
Default setting: '0' = DDR_Controller_Clock: csb_clk ratio is 1:1
'1' = DDR_Controller_Clock: csb_clk ratio is 2:1

SW4.7: Local Bus Clock Mode
Default setting: '0'; the local bus and Secondary DDR memory controller will operate with a frequency equal to csb_clk

SW4.8: CEPDF
Default setting: '0'; QE_clk = (primary clock input) X (CEPMF)

SW9

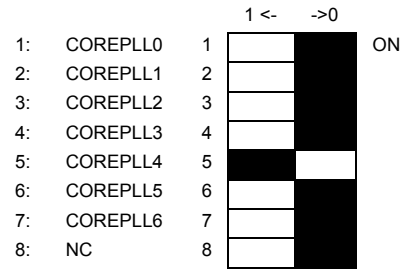
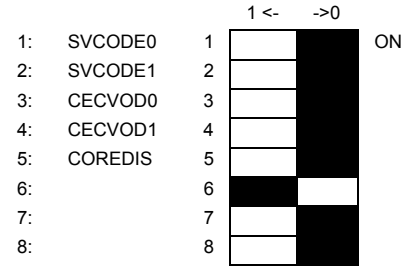
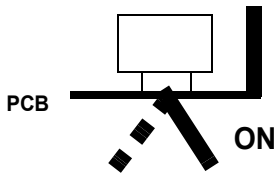


SW9.1: PCI_MODE.
Default setting: '0' (this switch should be only in '0' state - see BCSR10:4)

SW9.2: PCICKDRV
Default setting: '1' = PCI clock output buffers are enabled

SW9.3 FCFG: sets RCW source on local bus
'0': BCSR source; settings from DIP-switches SW3.1-SW3.3
'1': Flash source
Default setting: '1'; Flash

SW9.4 - SW9.8: CEPMF
QE PLL multiplication factor
Default setting: '6'; '00110'
If CLKDIV=0 then for 400Mhz CEPMF should be 6.

SW10  	**SW10.1-SW10.7: core PLL setting** Sets the ratio between the e300 core clock and the internal csb_clk Default setting: '00001000' for fcore = 533MHz set '00000110' for fcore = 500MHz
SW11  	**SW11.1 - SW11.2: SVCODE** Sets SVCODE to '0' Default setting: 00; **SW11.3 - SW11.4: CEVCODE** Set CECVOD to 0; Default setting: '00'; **SW11.5 - COREDIS** ‘1’: e300 core is prevented from fetching the boot code until configuration by an external master is complete. ‘0’: e300 core runs normally Default setting: ‘0’
SW5 Power Switch  	**SW5: power switch** ON: power from an external 5V power supply via the P10 power jack combined mode: powered from +5V on PIB power supply through riser connectors (regardless of SW5 position) board plugged as a PCI add-in card: PC internal power supply will provide 5V via PCI edge connector

4.2 Jumpers

Figure 4-2 below shows the locations of the jumpers and LEDs.

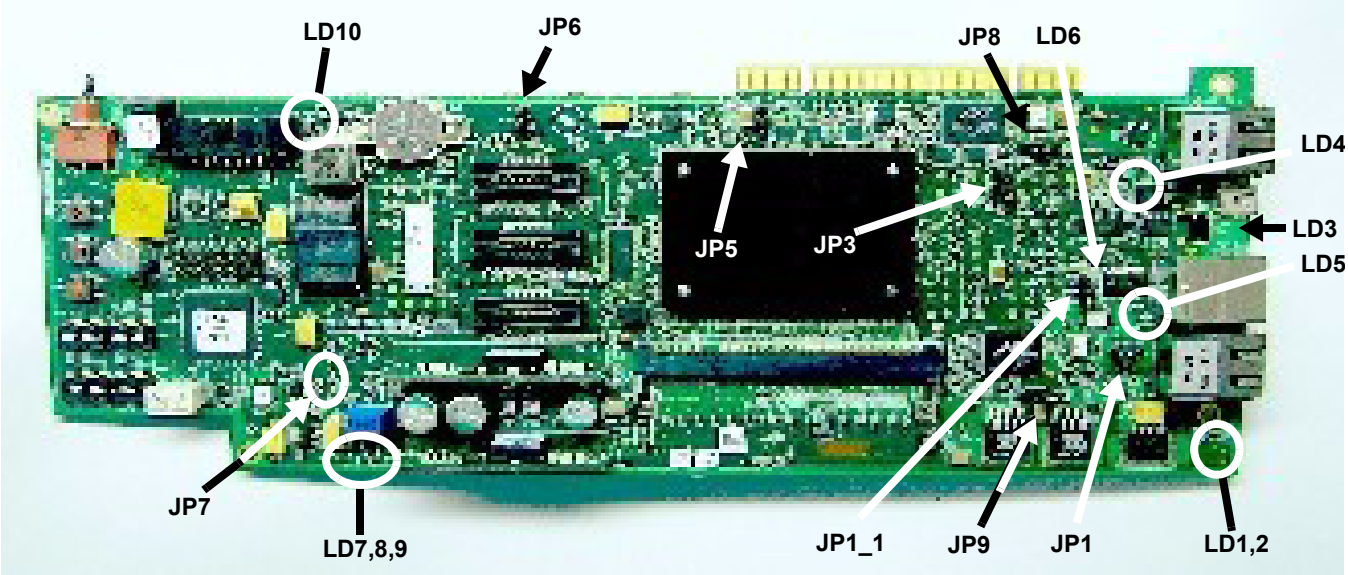


Figure 4-2. MPC8360EA MDS Processor Board Jumpers and LEDs Locations

MPC8360EA MDS Processor Board jumpers settings are described in Table 4-1, below.

Table 4-1. Jumper Settings

JP1: 3.3V/2.5V	GETH Voltage For 2.5V connect 1-2 For 3V JP1 connect 3-2
JP1_1: ATM/USB	JP1_1 Select between ATM function of USB function. For USB, connect 1-2 For ATM connect 3-2
JP3: TDMATXCLK/M2GTX125	For GMII2, connect 1-2 (M2GTX125 for GTETH2). For TDMA when using PIB, connect 3-2.
JP5: LVDD2 3.3V/2.5V	8360 LVDD2 Power For LVDD2 3.3V connect 2-3. For LVDD2 2.5V connect 1-2.
JP6: Clock source PIB(onboard)/External	JP6 Clock Connected 1-2 to configure for an external clock.
JP7: Reset.	JP7 reset not connected.
JP8, JP9:	JP8, JP9: not in use

4.3 LEDs

The MPC8360EA MDS Processor Board has the following LEDs (locations shown in Figure 4-2 above):

LD1, LD2 - Signaling LEDs

LED's, LD1 (**green**) and LD2 (**red**), are program controlled. They are used for extra visibility on the running utility. They are lit up by setting bits BCSR0.5-6 respectively.

LD3 - USB Power

When lit (**green**), the USB Vbus is powered.

LD4, LD5 - GETH Enable

The **green** LED, LD4,5, indicates enable for GETH Transceivers U6,U5, respectively.

LD6 - DUART Enable

A **green** LED, LD6, indicates enable for the RS232 Dual Transceiver.

LD7 - Power GOOD

A **green** LED, LD7, indicates that the MPC8360EA MDS Processor Board power is operating normally.

LD8 - Not in Use

LD9 - BOOT Indicator

The LD9 indicates MPC8360 boot processing.

LD10 - 5V Power Indicator

The **green** LED, LD10, indicates a 5V power level on the MPC8360EA MDS Processor Board.

A 5V power supply is plugged into the P10 Power Connector on the board's front side for the Stand-Alone Mode. The MPC8360EA MDS Processor Board is powered by the 5V external power supply when the SW5 Power Switch is turned to the "ON" (up) position.

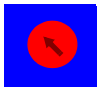
When the MPC8360EA MDS Processor Board is plugged into an PC via the PCI edge connector it is powered from the edge connector's 5V power rail (Agent Mode). In the PIB Combined Mode, 5V power is supplied from the PIB's power supply via risers connectors. Note that if working in either of these two modes, the position of SW1 is ignored.

4.4 Other Controls and Indicators

4.4.1 Push Buttons

Table 4-2 below describes the functionality of the board's push buttons. See Figure 4-3 for the locations of these push buttons.

Table 4-2. The MPC8360EA MDS Processor Board Push Buttons

SW1 Power-on-Reset	 PRESET	Pressing button SW1 results in Power-On-Reset for all components on the MPC8360EA MDS Processor Board. Use this reset button when the MPC8360EA MDS Processor Board is installed in a PC.
SW2 Software Option	 SW OPT	Rotary Switch SW2 allows the user to change the program flow according to eight available cases. Not available when installed in a PC.
SW6 Hard Reset	 HRESET	Pressing button SW6 results in a Hard Reset for the MPC8360EA. Not available when installed in a PC.
SW7 Soft Reset	 SRESET	Pressing button SW7 results in a Soft Reset for the MPC8360EA. Despite the reset, clock and chip-select data as well as SDRAM (if installed) contents are retained. Not available when installed in a PC.
SW8 NMI (Abort)	 NMI	Pressing button SW8 results in aborting program execution by issuing a level 0 interrupt to the MPC8360EA. The ABORT switch signal is de-bounced.

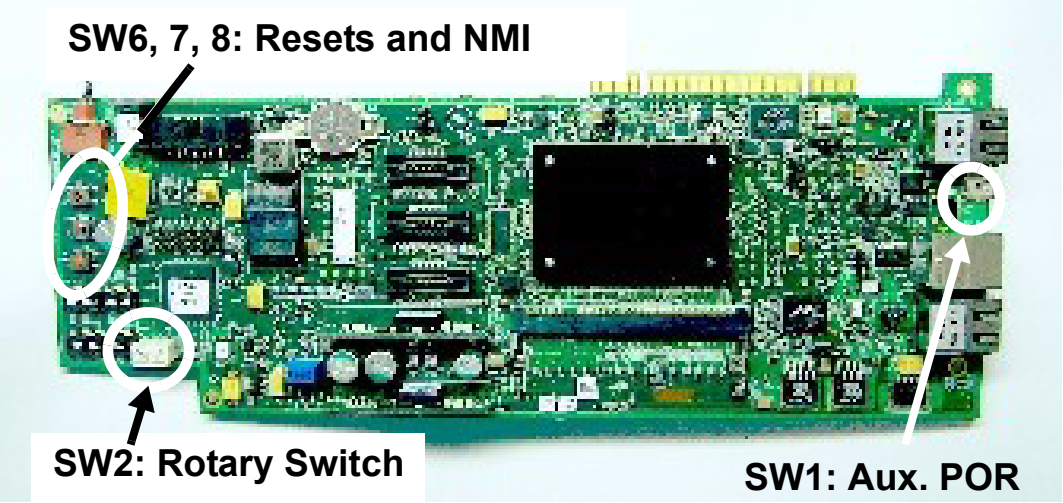


Figure 4-3. MPC8360EA MDS Processor Board Push Buttons and Auxiliary POR

4.4.2 Test Points

TP46, shown below in Figure 4-4, is used in conjunction with the voltage trimmer for adjusting the core frequency. To change the core frequency to 500MHz, use the screw on the voltage trimmer to adjust the Vdd voltage to 1.3 V. Measure this voltage using Test Point 46, as shown. To return the core frequency to 533MHz, return this voltage to 1.2V.

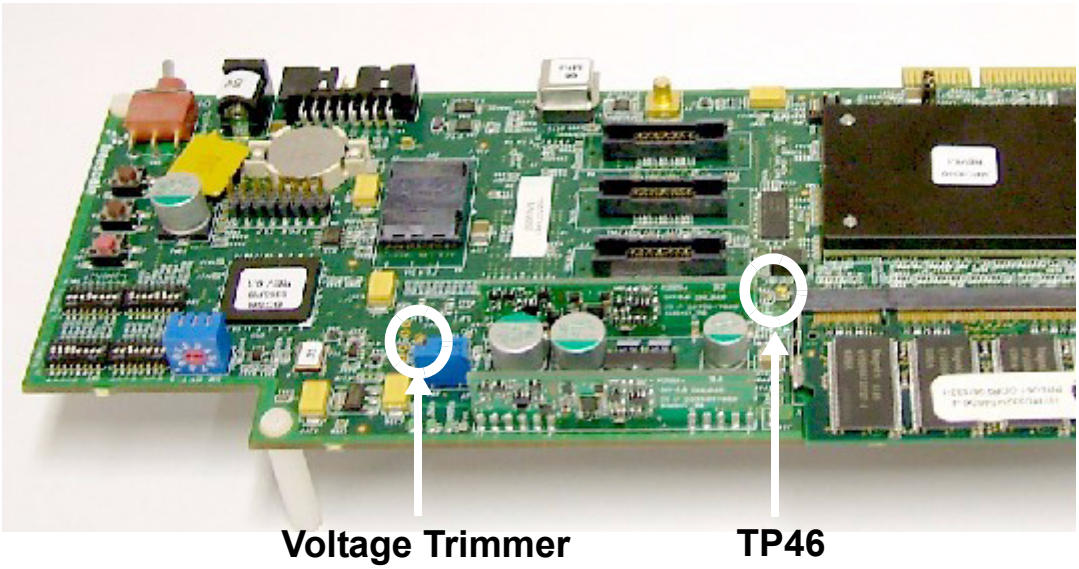


Figure 4-4. Voltage Trimmer and Test Point

Chapter 5
Functional Description

In this chapter the design details of various modules of the MPC8360EA MDS Processor Board are described. This includes memory map details and software initialization of the board.

5.1 Reset & Reset - Configuration

5.1.1 Reset Clocking and Configuration Initialization

The MPC8360EA samples four configuration pins at Power-on-Reset negation. Three of these pins (CFG_RESET[0-2]) allow setting up to eight variant reset configuration sources such as I2C EEPROM, parallel EPROM (Flash) or various hard coded options (see the *MPC8360 User Manual* for more details). See Table 5-1 for a list of the options. The fourth pin, CFG_CLKIN_DIV, determines whether or not CLKIN is divided by 2. Figure 5-1 below shows a schematic diagram of the reset circuit, including the various signals and their sources.

Table 5-1. Reset Configuration Words Source

Reset Configuration Signal Name	Value (Binary) SW3(1-3)	Meaning
CFG_RESET[0-2]	000	Reset configuration word is loaded from a local bus EEPROM, or the BCSR (and DIP switches), depending on the value of SW9[3]
	001	Reset configuration word is loaded from an I^2C EEPROM, PCI_CLK/PCI_SYNC_IN is in the range of 25-44 Mhz.
	010	Reset configuration word is loaded from an I^2C EEPROM, PCI_CLK/PCI_SYNC_IN is in the range of 44-66.666 Mhz.
	011	Hard coded option #0. Reset configuration word is not loaded.
	100	Hard coded option #1. Reset configuration word is not loaded.
	101	Hard coded option #2. Reset configuration word is not loaded.
	110	Hard coded option #3. Reset configuration word is not loaded.
	111	Hard coded option #4. Reset configuration word is not loaded.

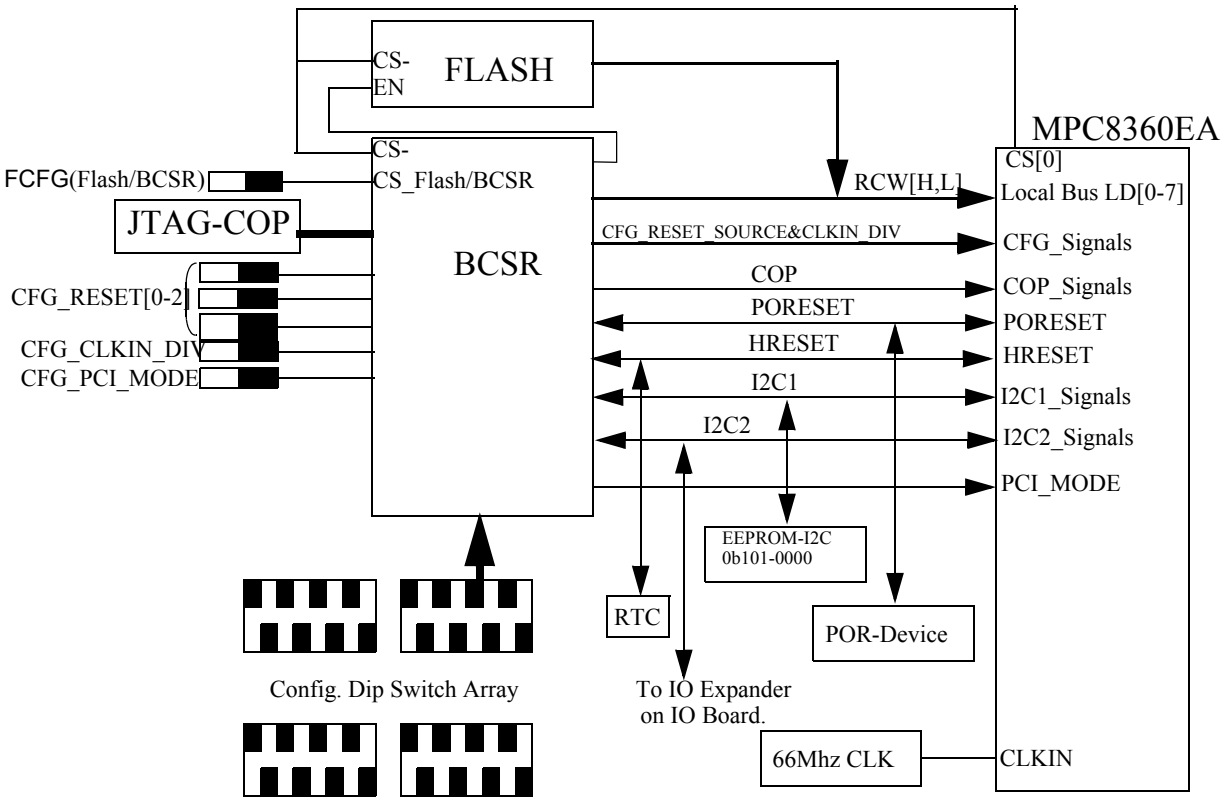


Figure 5-1. Reset Circuit Block Diagram

Once the PORESET signal is negated, the MPC8360EA starts to load the reset configuration word (RCW) bits. These bits are latched from the DIP-switches into appropriate FPGA registers (the BCSR's).

Excluding the hard-coded options, there are three ways to drive the RCW:

- Via BCSR: CFG_RESET_SOURCE[0-2] = 000, & FCFG = 0 (SW9[3])
- Via FLASH: CFG_RESET_SOURCE[0-2] = 000 & FCFG = 1 (SW9[3])
- Via I2C - 1 bus: CFG_RESET_SOURCE[0-2] is 001 or 010, depending on the value desired for PCI_CLK. The setting for FCFG has no effect in this case.

All the RCW bits can be changed from their initial settings using either the FPGA BCSR through the local bus or using the LLD (low level debugger). The BCSR must then drive HRESET/ PORESET to load a new configuration word to the device. It is possible to read the value of the RCW from the BCSRs.

Figure 5-2 below shows the timing for the reset sequence.

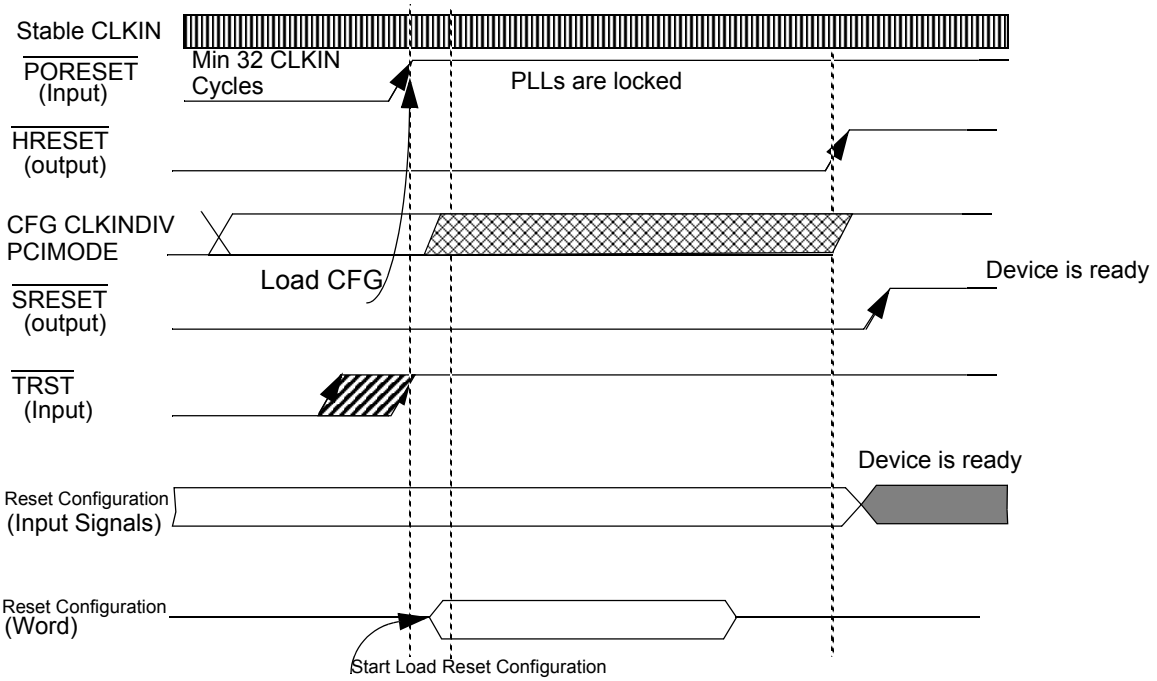


Figure 5-2. Reset Timing Diagram

5.1.2 Reset Circuit

The Reset Circuit of the MPC8360EA MDS Processor Board has the following features:

- Reset controller (MIC2774N-23BM5 from MICREL) - drives the PORESET signal.
- Low power detect device - drives PORESET as follows:
  - if the detected input power is less than 4.5V, PORESET is driven at 5V
  - if the detected input power is less than 3V, PORESET is driven at 3.3V
  - if the detected input power is less than 2V, PORESET is driven at 2.2V
  - if the detected input power is less than 1.6V, PORESET is driven at 1.8V
- JTAG COP - can drive HRESET or SRESET, depending on the command given from the JTAG device.
- Push button for PORESET, HRESET and SRESET.
- FLASH memory is protected during PORESET and after that until the system is enabled
- PCI RST signal is connected to PORESET signal.
- The BCSR resets the GETH PHYs.

5.1.3 MPC8360EA MDS Processor Board Reset Principles

Upon power on: The device MIC2774N-23BM5 drives PORESET low for 300msec to the MPC8360EA, BCSR, FLASH and PIB (if connected).

The BCSR performs initialization procedures, and drives the $\overline{\text{BCSR Done}}$ signal to logical 1, in order to keep PORESET low. After finishing initialization the $\overline{\text{BCSR Done}}$ signal is logical 0.

The BCSR drives a RESET signal to the GETH - PHYs.

In Agent mode: When the MPC8360EA MDS Processor Board is in Agent Mode, PORESET can be driven by PCI RST signal.

The PCI edge connector PIN RST is connected to the PORESET in order to reset the MPC8360EA MDS Processor Board.

There are several reset sources on the MPC8360EA MDS Processor Board:

- Power On Reset
- Manual Hard Reset
- Manual Soft Reset
- MPC8360 device

HRESET & SRESET can be driven through JTAG COP connector, by BCSR, by RTC when it reaches its count value, or by push buttons.

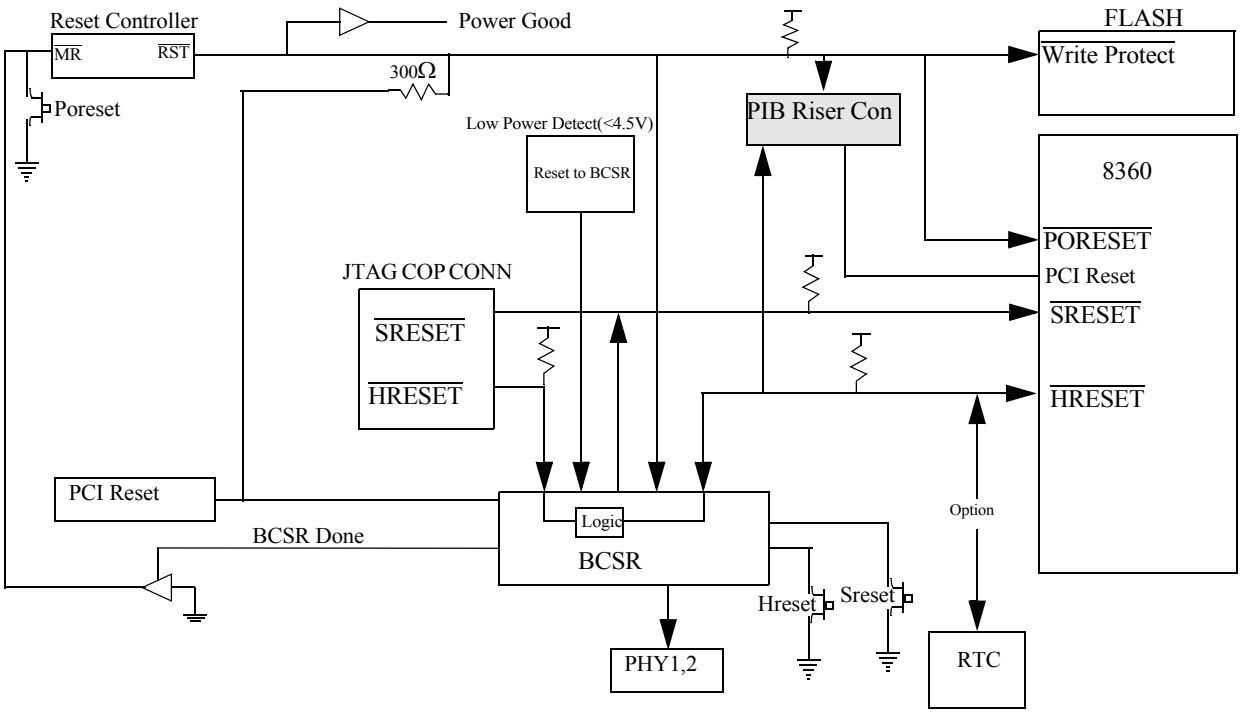


Figure 5-3. Reset Block Scheme

5.1.4 Power - On Reset

The power on reset to the MPC8360EA MDS Processor Board initializes the processor's state after power up. A dedicated logic unit asserts PORESET input for a period long enough to cover the MPC8360EA core

voltage stabilization. A Power-On-Reset may be generated manually as well by an on-board dedicated push-button (SW1).

In addition, a power on reset for the MPC8360EA can be done by toggling bit #4 in BCSR11.

5.1.5 Hard Reset

Hard Reset may be generated on the MPC8360EA MDS Processor Board by any one of the following sources:

- COP/JTAG Port (in Stand-Alone Mode only)
- Manual Hard Reset.
- Board Internal sources.

Hard Reset, when generated, causes the MPC8360EA to reset all its internal hardware except for PLL logic and re-acquires the Hard Reset configuration from its current source. Since Hard Reset also resets the refresh logic for dynamic RAMs, their content is lost as well.

CAUTION

HRESET is an open-drain signal and must be driven with an open-drain gate by whatever external source is driving it. Otherwise, contention will occur over that line, and that might cause permanent damage to either board logic and/or to the MPC8360EA.

5.1.6 COP/JTAG Port Hard - Reset (stand-alone only)

To provide convenient Hard Reset capability for a COP/JTAG controller, an HRESET line has been connected to the COP/JTAG port connector. The COP/JTAG controller may directly generate a Hard Reset by asserting (low) this line. The HRST signal from the COP/JTAG is then driven to the FPGA and from the FPGA to the MPC8360EA. This technique is used in order to cooperate with the low level debugger, although it can operate properly without connecting it to the FPGA.

5.1.7 Manual Hard Reset

To allow a run-time Hard Reset, a manual Hard Reset is facilitated, via SW6. Note that this cannot be done when the MPC8360EA MDS Processor Board is connected in a PC (Agent Mode), but instead SW1 (PORESET) can be used.

In addition, a manual Hreset for the MPC8360EA can be done by toggling bit #4 in the CCR (Address F800000F) register in the FPGA.

5.1.8 Manual Soft Reset

To allow a run-time Soft Reset, manual Soft Reset is facilitated, via SW8. Note that this cannot be done when the MPC8360EA MDS Processor Board is connected in a PC (Agent Mode).

In addition, a manual Sreset for the MPC8360EA can be done by toggling bit #5 in the CCR (Address F800000F) register.

5.2 Default Settings

The default settings for the MPC8360EA MDS Processor Board are as follows:

- Clock in (primary clock) = 66Mhz
- Core freq = 533Mhz
- CCB = 266Mhz
- DDR = 400Mhz
- QE = 400Mhz
  [QE Clock = (Primary clock) x (CEPMF)], CEPMF set by SW9[4-8]
- Local Bus 66Mhz LBIU.LCRR = 0x4
- Primary DDR = DDR2
- CFG_RS[0-2], BootSeq, RomLoc = Read configuration word from Local Bus (BCSR)
- PCI_MODE = Enable PCI Clock drive

5.3 Clocking

A block diagram showing internal details for the clocks of the MPC8360EA MDS Processor Board is shown below in Figure 5-5. As can be seen from the diagram, all input signals (those that are on the left of the block), except CFG_CLKIN_DIV and CLKIN are from the configuration word, and should be set according to information in the *MPC8360EA Reference Manual.*

A diagram showing certain electrical aspects of the clocking connections is shown below in Figure 5-4.

There are two modes of clock operations:

- Agent
- Host

In **Agent mode**, the MPC8360EA MDS Processor Board is inserted in a PC or in an adaptor on the PIB. Its edge connector (P3), which is inserted in the PC's or PIB's PCI slot, causes an open circuit between PCI_SYNC_OUT and PCI_SYNC_IN. The PC or the host processor on the PIB therefore supplies the input clock signal via the PCI_SYNC_IN pin. In this case, the CLKIN pin is connected to GND.

In **Host mode**, the edge connector (P3) is not connected. As a result, CLKIN is not connected to ground, and supplies a clock signal to PCI_SYNC_OUT via a 5" loop trace (which covers the PCI clock routing on the PIB), which in turn is connected to PCI_SYNC_IN, thus supplying a clock to the MPC8360E device.

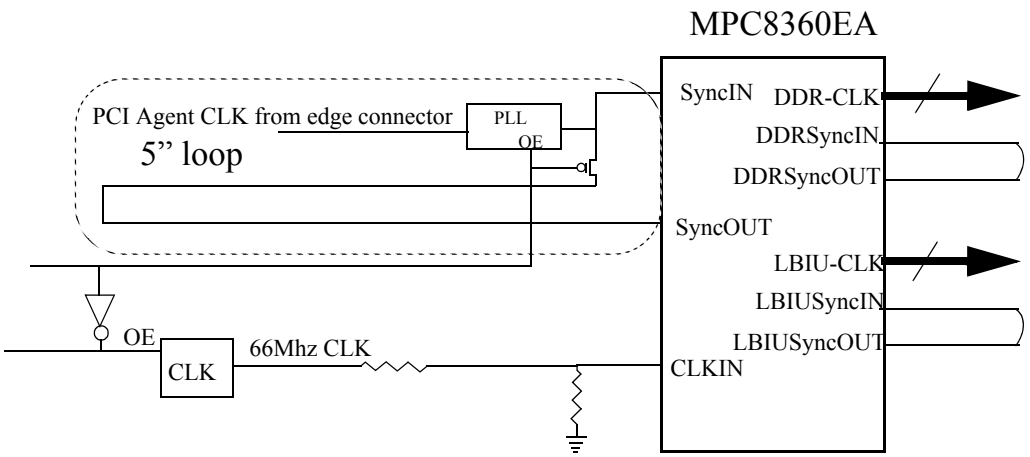


Figure 5-4. Clocks - Showing External Connection Scheme, and 5” Loop

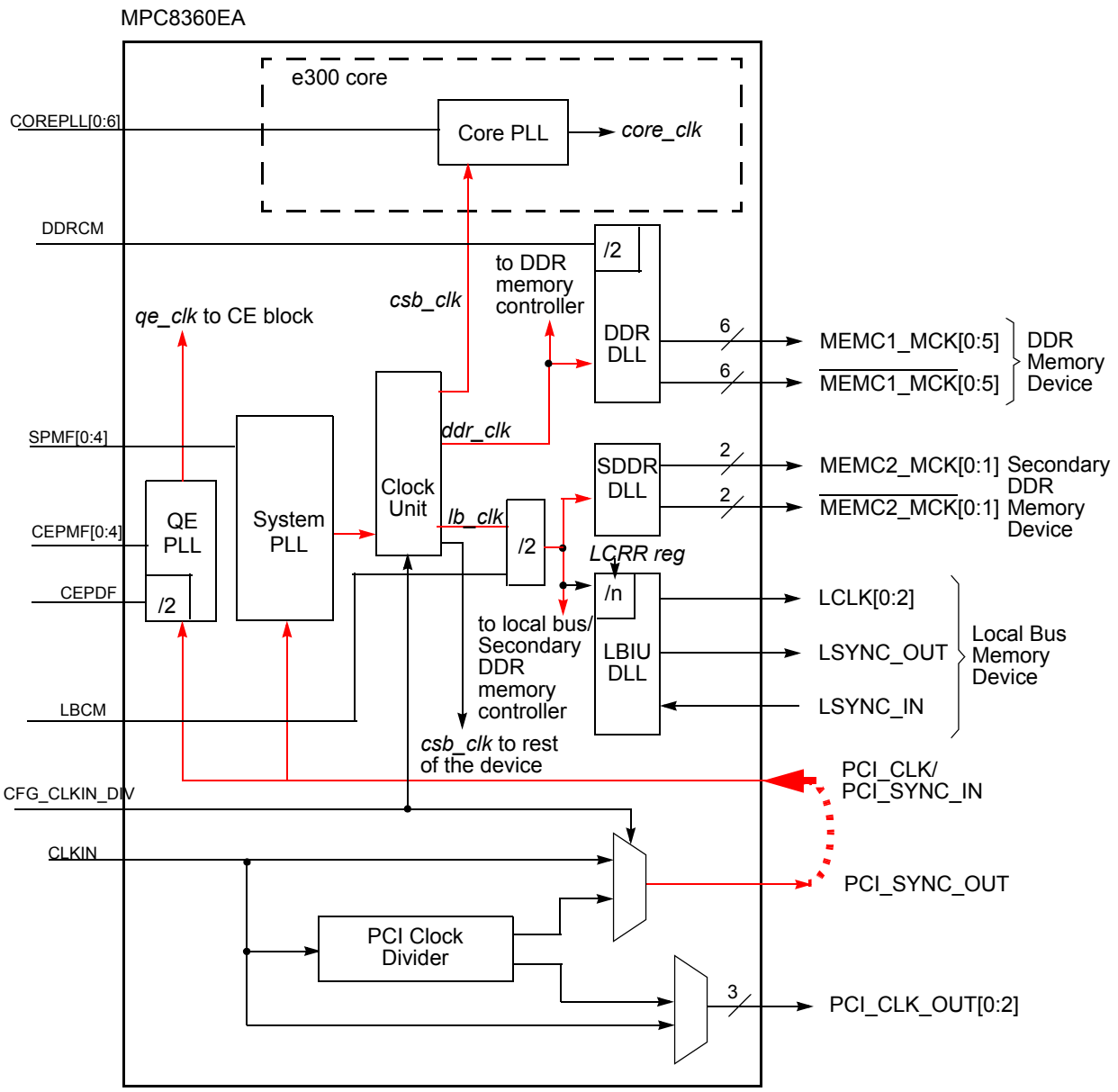


Figure 5-5. Clocks - Internal Details

Working with ATM signals:

It is important to note that if working with ATM/UPC2 signals, the MPC8360EA must receive a 66MHz signal on the PCI_SYNC_IN pin. Since this is the frequency of the CLKIN signal, this is not a problem in Host mode. However, in Agent mode, when the clock signal comes from an outside source, and the CLKIN is connected to GND, the user must ensure that the ATM nevertheless uses this 66MHz signal. Do this as follows:

First, write to the following, **before** you set the PCIMODE bit (BCSR10[4]) bit to '1':

0X'EFE35010' to offset 0x'XX00148C'
and then
0X'FFE3A030' to offset 0x'XX00148C'

5.4 FPGA - Board Control & Status Registers (BCSR)

The FPGA contains the BCSR, which is an 8-bit wide read / write register file that controls or monitors most of the MPC8360EA MDS Processor Board hardware options. The BCSR's register may be accessed from the Local Bus. The BCSR includes up to 16 registers, some of which are optional.

BCSR registers are duplicated numerous times within a CS1 region. This is due to the CS region's 32KB minimum block size and the fact that only address lines A[27-31] are decoded for register selection by the BCSR. BCSR is implemented on a Xilinx FPGA device that provides register and logic functions over some MPC8360EA MDS Processor Board signals.

The BCSR controls or monitors the following functions:

- Power-on-Reset & Hardware configuration setting for the processor.
- Hardware Reset Configuration bits are stored in BCSR registers available from the Local Bus.
- Hard- Soft- Reset and NMI (IRQ) push buttons debounce function.
- Hardware Configuration for both GETH transceivers.
- Enable/Disable to:
  - Two GETH1,2 Transceivers.
  - Dual RS232 Transceiver.
  - LED off.
- Provides hardware write protection for FLASH and BRD I2C EEPROM .
- Two LEDs (one green, one red), which provide software signaling.
- Special CCR - COP register for JTAG port connectivity.
- Status registers include:
  - PCI Host Mode indicates if the Board is working in a Host Mode (Stand-Alone or PIB Combined) or the Agent Mode BCSR4[0]
  - Processor Low Power Mode (QUISCE) BCSR11[0]
  - BCSR Revision code BCSR12[0-3] REV BCSR12[4-7] SUBREV.

Sections of the BCSR slice control registers generally have **low** active notations. This means that a bit function is realized while the bit is zero. When a bit is set to **high** a related function is disabled. The default setting is assumed to be non-functional. The most significant bit is bit 0.

The BCSR (No.'s 0...7) registers reflect the values in the RCW High and Low registers of the MPC8360EA chip. The correspondence is indicated in the "CWL" or "CWH" column.

5.4.1 BCSR0 - Status Register

BCSR0 serves as a 8-bit control register on the board. BCSR0 may be read or written at any time. BCSR0 defaults are attributed immediately after a Power-On Reset or HRESET. BCSR0 fields are described below in Table 5-2

Table 5-2. BCSR0 Register Description

CWL	BIT	MNEMONIC	Function	DEF on PORESET
0	0	LBCCM	***Local Bus/Secondary DDR Controller.*** Selects local bus and secondary DDR memory controllers clock ratio. The bit is set by default by appropriate DIP switch. May be rewritten via LBIU.	SW4[7] Sampled at HRESET
1	1	DDRCM	***DDR SDRAM Clock Mode.*** If this bit set **high**, the DDR SDRAM memory controller will operate with frequency equal to twice the frequency of the *csb_clk*. If this bit is **low**, the DDR SDRAM memory controller will operate at the *csb_clk* frequency. A DIP-switch may change DDRCM bit setting. May be rewritten any time via LBIU.	SW4[6] Sampled at HRESET
2-3	2-3	SVCOD[0-1]	**VCO Division**. The two bits reflect SVCOD[2-3] signals logic level during Hard Reset Configuration sequence. The bits are set low by default. May be rewritten via LBIU.	SW11[1-2] Sampled at HRESET
4-7	4-7	SPMF	**System PLL Multiplication Factor**. The four bits reflect SPMF[0-3] signals logic level during Hard Reset Configuration sequence. The bits are set by default by appropriate DIP switch. May be rewritten via LBIU.	SW3[5-8] Sampled at PORESET neg.

5.4.2 BCSR1 - Status Register

On the board, BCSR1 acts as a control register. BCSR1, which may be read or written at any time, receives its defaults immediately after Power-On or PORESET. BCSR1 fields are described below in Table 5-3:

Table 5-3. BCSR1 Register Description

CWL	BIT	MNEMONIC	Function	DEF on PORESET
8	0	Reserved	Reserved	
9-15	1-7	COREPLL	**Core PLL Multiplication Factor.** The seven bits reflect COREPLL[0-6] signals logic level during Hard Reset Configuration sequence. The bits are set by appropriate DIP switch (default). May be rewritten via LBIU.	SW10[1-7] Sampled at HRESET

5.4.3 BCSR2 - Status Register

This register is reserved.

5.4.4 BCSR3 - Status Register

On the board, BCSR3 acts as a control register. BCSR3, which may be read or written at any time, receives its defaults immediately after the PORESET signal. BCSR3 fields are described below in Table 5-4:

Table 5-4. BCSR3 Register Description

CWL	BIT	MNEMONIC	Function	DEF on PORESET	ATT.
24-25	0-1	CEVCOD[0-1]	**CE PLL VCO**. The two bits reflect CEVCOD[0-1] signals logic level during Hard Reset Configuration sequence. The bits are set low by default. May be rewritten via LBIU.	SW11[3-4] Sampled at HRESET	R,W
26	2	CEPDF	**CE PLL Division Factor**. This bits reflect CEPDF signal logic level during Reset Configuration sequence. The bits are set by appropriate DIP switch. May be rewritten via LBIU.	SW4[8] Sampled at HRESET	R,W
27-31	3-7	CEPMF	***CE PLL Multiplication Factor.*** The five bits reflect CEPMF signals logic level during Reset Configuration sequence. The bits are set by appropriate DIP switch. May be rewritten via LBIU.	SW9[4-8] Sampled at HRESET	R,W

5.4.5 BCSR4 - Status Register

On the board, BCSR4 acts as a control register. BCSR4, which may be read or written at any time, receives its defaults immediately after PORESET signal. BCSR4 fields are described below in Table 5-5:

Table 5-5. BCSR4 Register Description

CWH	BIT	MNEMONIC	Function	DEF on PORESET
0	0	PCIHOST	**PCI Host Mode**. If **low**, the PCI processor is set to agent mode. This occurs when the Processor Board is not connected to the PIB riser connectors (stand-alone, inserted in PC, or inserted in PMC slot in PIB). When the Processor Board is combined with the PIB (connected to riser connections), this bit is **high**, the PCI processor is in host mode.	If P3 (Edge Connector) connected to PC or PIB: 0 If P3 not connected: 1
1	1	Reserved	Reserved	0
2	2	PCIARB	**PCI Internal Arbiter Mode**: ***PCI Arbiter.*** Set **low** for PCI add-in card configuration (in PC or on PMC slot in PIB) to provide external arbiter. When the Processor Board is combined with the PIB the bit is **high** to configure the PCI port with an internal arbiter.	If P3 (Edge Connector) connected to PC or PIB: 0 If P3 not connected: 1
3	3	PCICKDRV	**PCI Clock Output Drive**: The bit reflects the PCICKDRV signal logic level during Reset Configuration sequence. The bit is set by appropriate DIP switches. May be rewritten via LBIU.	SW9[2]

Table 5-5. BCSR4 Register Description (continued)

CWH	BIT	MNEMONIC	Function	DEF on PORESET
4	4	COREDIS	**Core Disable Mode**: When **high**, the e300 core is prevented from fetching the boot code until configuration by an external master is complete. If **low** the core runs normally. May be rewritten via LBIU.	SW11[5]
5	5	BMS	***Boot Memory Space.*** Sets lower 8MByte boot memory space location when the DDR is used as the PCI boot source. Otherwise (for LBIU boot source), this bit is **high** for upper boot memory space location. User may change boot source location by request.. May be rewritten any time via LBIU.	0
6-7	6-7	BOOTSEQ[0-1]	***Boot Sequencer Configuration.*** These two bits reflect BOOTSEQ[0-1] signals logic level during Reset Configuration sequence. The bits are set by appropriate DIP switch. May be rewritten via LBIU.	SW4[1-2]

5.4.6 BCSR5 - Status Register

On the board, BCSR5 acts as a control register. BCSR5, which may be read or written at any time, receives its defaults immediately after the PORESET signal. BCSR5 fields are described below in Table 5-6 :

Table 5-6. BCSR5 Register Description

CWH	BIT	MNEMONIC	Function	DEF on PORESET
8	0	SWEN	**Soft Ware Watchdog Enable**: This bit reflects the SWEN signals logic level during Reset Configuration sequence. The bit is set by default to **low**, to Software Watchdog Disable. May be rewritten via LBIU.	0
9-11	1-3	ROMLOC[0-2]	**Boot ROM Interface Location**. These bits reflect ROMLOC[0-2] signals logic level during Reset Configuration sequence. The bits are set by default by appropriate DIP switch. May be rewritten via LBIU.	SW4[3-5]
12-14	4-6	Reserved	Reserved	0
15	7	CEBOOT	Reserved	0

5.4.7 BCSR6 - Status Register

This register is reserved.

5.4.8 BCSR7 - Status Register

On the board, BCSR7 acts as a control register. BCSR7, which may be read or written at any time, receives its defaults immediately after PORESET signals. BCSR7 fields are described below in Table 5-7.

Table 5-7. BCSR7 Register Description

CWH	BIT	MNEMONIC	Function	DEF on PORESET	ATT.
24-26	0-2	Reserved	Reserved	0	R,W
27	3	SDDRIOE	**Secondary DDR IO Enable**: This bit reflects SDDRIOE signals logic level during Reset Configuration sequence. The bit is set by default to **low,** which means the secondary DDR is disabled. May be rewritten via LBIU.	0	R,W
28	4	TLE	**True Little Endian** : If **high**, true little endian is used. If **low**, big endian is used.	0	R,W
29	5	LALE	**LATE Timing**: If **high**, the LALE is negated 1/2 lbiu_controller_clk earlier than normal. If **low**, normal LALE timing is used.	0	R,W
30-31	6-7	Reserved	Reserved	0	R,W

5.4.9 BCSR8 - Status Register

On the board, BCSR8 acts as a control register. BCSR8, which may be read or written at any time, receives its defaults immediately after PORESET. BCSR8 fields are described below in Table 5-8

Table 5-8. BCSR8 Register Description

BIT	MNEMONIC	Function	DEF on PORESET	ATT.
0-1	TSEC1M	**G-ETH port 1 Config Mode.** These two bits select the communication protocol used by the G-ETH1 controller. For details on possible values and their meanings, See Table 5-9. May be rewritten any time via LBIU. **Note: If mounted on PIB should be only Reduced mode.**	10	R,W
2-3	TSEC2M	***G-ETH port 2 Config Mode.*** These two bits select the communication protocol used by the G-ETH2 controller. For details on possible values and their meanings, See Table 5-9. May be rewritten any time via LBIU. **Note: If mounted on PIB should be only Reduced mode.**	10	R,W
4	TSEC1MST	***GETH1 Master Mode.*** If **high** GETH1 transceiver configures in Master Mode otherwise when **low** GETH1 transceiver operates as Slave. May be rewritten any time via LBIU.	1	R,W
5	TSEC2MST	***GETH2 Master Mode.*** If **high** GETH2 transceiver configures in Master Mode otherwise when **low** GETH2 transceiver operates as Slave. May be rewritten any time via LBIU.	1	R,W
6	Reserved	Reserved	0	R,W

Table 5-8. BCSR8 Register Description (continued)

BIT	MNEMONIC	Function	DEF on PORESET	ATT.
7	FLEN	***FLASH Enable.*** **Low** enables Flash accesses. When **high** Flash operation is not available, PSRAM part may be enabled instead. May be rewritten any time via LBIU.	0	R,W

Table 5-9. G-ETH Port Mode

Setting Value TSEC1/2M	G-ETH Mode
00	The G-ETH controller operates in the RGMII protocol, using only four transmit data signals and four receive data signals.
01	The G-ETH controller operates in the RTBI protocol, using only four transmit data signals and four receive data signals.
10	The G-ETH controller operates in the GMII protocol, using eight transmit data signals and eight receive data signals.
11	The G-ETH controller operates in the TBI protocol, using ten transmit data signals and eight receive data signals.

5.4.10 BCSR9 - Status Register

The BCSR9 serves as an 8-bit control register. BCSR9 may be read or written at any time. BCSR9 defaults are attributed at the time of Power-On-Reset or HRESET. BCSR9 fields are described below in Table 5-10

Table 5-10. BCSR9 Register Description

BIT	MNEMONIC	Function	DEF on HRST	ATT.
0	GETH1EN	***GETH Transceiver 1 Enable.*** When **low,** enables the Geth PHY#1. When negated (**high**), GETH PHY#1 enters standby mode. May be rewritten via LBIU.	0	R,W
1	GETH2EN	***GETH Transceiver 2 Enable***. When **low,** enables Get PHY#2. When negated (**high**), GETH PHY#2 enters standby mode. May be rewritten via LBIU.	0	R,W
2	GETHRST	***GETH Transceiver Reset***. The GETH devices are reset when the GETHRST is asserted (**low**). The Board Hard Reset signal of the MPC836x will reset GETH phy devices. This bit may be rewritten via LBIU. When this bit is negated (**high**), that is, toggled from 0 to 1, the PHY configuration is taken from the current values in BCSR8[0-5]. After the PHY configuration is completed this bit is cleared (high).	1	R/W
3	RS232EN	***UART Ports Transceivers Enable***. Upon activation **(low),** the Dual RS232 Transceiver, using the UART ports of the MPC8360, is enabled. When negated (**high**), the RS232 Transceiver enters standby mode. May be rewritten via LBIU.	0	R,W

Table 5-10. BCSR9 Register Description (continued)

BIT	MNEMONIC	Function	DEF on HRST	ATT.
4	BOOTWP	***BOOT I2C EEPROM Protect***. When asserted (**low**) BOOT EEPROM functions normally, when negated (**high**) write operations are disabled. May be rewritten via LBIU.	1	R,W
5	SIGNAL0	***Signal LED 0***. A dedicated **Green** LED is illuminated when SIGNAL0 is active (**low**). The LED is unlit when in it's inactive (default) state **(high)**. During the Reset Configuration sequence the illuminated LED indicates the SRESET assertion. The user may utilize the LED for s/w Slave signalling purposes. May be rewritten via LBIU.	1	R,W
6	SIGNAL1	***Signal LED Slave 1***. A dedicated **Red** LED is illuminated when SIGNAL1 is active (**low**). The LED is unlit when in its inactive (default) state **(high)**. During the Reset Configuration sequence the illuminated LED indicates the HRESET assertion. May be rewritten via LBIU.	1	R,W
7	CEUARTEN	For CE UART on the Processor Board, this bit should be active (**low**). When this bit is negated (**high**), the UPC2-ATM is used.	0	R,W

5.4.11 BCSR10 - Status Register

On the board, BCSR10 acts as a control register. BCSR10, which may be read or written at any time, receives its defaults upon Power-On or PORESET. BCSR10 fields are described below in Table 5-11

Table 5-11. BCSR10 Description

BIT	MNEMONIC	Function	DEF on PORESET	ATT.
0	CFG_CLKIN_DIV	**CLKIN Division.** The bit reflects CFG_CLKIN_DIV signal logic level during the Power-On Reset Configuration sequence. If **low**, (agent mode), CLKIN is not connected to PCI_SYNC_OUT (which in turn, is connected to PCI_SYNC_IN). If **high**, PCI_SYNC_IN receives its signal from PCI_SYNC_OUT, which in turn, is connected to CLKIN (host mode). The bit is set by default by appropriate DIP switch SW3/4. May be rewritten via LBIU.	SW3[4]	R,W
1-3	CFG_RS[0-2]	**Reset Configuration Words Source**. These bits reflect CFG _RS[0-2] signals logic level during PORESET Configuration sequence. The bits are set by default by appropriate DIP switch SW3/1-3. May be rewritten via LBIU.	SW3[1-3]	R,W
4	PCI_MODE	**PCI MODE.** Select if the PCI or ATM is used. If **low**, PCI is used. If **high**, ATM is used (but see note below). This switch is driven constantly, and should be driven always to H/L after reset sequence. The PCIMODE switch on SW9[1] should not be set to '1' for ATM. In order to place the device in ATM mode, write to: 0x'XX001484' 0x'EFE35010'	0	R,W
5	FLASHPRT	**Flash Protected**. If this bit is **low**, the flash memory is protected, and cannot be written to or changed. If this bit is **high**, the flash memory can be written to.	1	R,W
6	LEDEN	**LEDs Enable.** Extinguish all LED when **high**. When **low** the LEDs are controlled normally. May be rewritten any time via LBIU.	0	R,W
7	JTAG2SEL	**JTAG Chain Select.** When **high**, this bit selects JTAG chain for external devices on PMC cards. **Low** provides JTAG normal configuration.	0	R,W

5.4.12 BCSR11 - Status Register

On the board, BCSR11 acts as a control register. BCSR11, which may be read or written at any time, receives its defaults upon Power-on or PORESET signals. BCSR11 fields are described below in Table 5-12

Table 5-12. BCSR11 Register Description

BIT	MNEMONIC	Function	DEF on PORESET	ATT.
0	QUISCE	**QUISCE Status**. Allow the processor power down mode to be determined (**low**) by reading via JTAG I/F. If this bit is **high**, the processor power down mode cannot be determined by reading from the JTAG I/F.	Processor signal	
1	BUFFEN	**Expansion Buffer Enable.** A value of **low** enables access to the PIB for the Combined Mode. A value of **high** sets the expansion buffer for working in the stand alone mode. May be rewritten any time via LBIU.	Setup defined at Power On	R,W
2	SHMOOEN	**SHMOO Test Enable.** Enables programming the Internal Core Power Supply and applying an external clock from the PIB Board (**low**). When the bit is **high**, the board operates in regular mode. May be rewritten any time via LBIU. .	1	R,W
3	BRDWP	**BRD Write Protect.** When **high** the BRD EEPROMs on the Processor Board cannot be written to. When **low,** the BRD's can be written to. May be rewritten any time via LBIU.	1	R,W
4	PORESET	**Power-On-Reset.** Toggling **low-high** within the proper time window will generate a PORESET negative pulse on the board. May be rewritten any time via LBIU.	1	R,W
5-7	SWOP	**Software Option** Three bits code the value read from the Rotary switch.	SW Rotary Coded	R

5.4.13 BCSR12 - Status Register

On the board, BCSR12 acts as a control register. BCSR12, which may be read or written at any time, receives its defaults upon Power-on signals. BCSR12 fields are described below in Table 5-13

Table 5-13. BCSR12 Register Description

BIT	MNEMONIC	Function	DEF on PORESET	ATT.
0-3	REV	**BCSR Revision**. Four bits revision coding	Programmed value	R
4-7	SUBREV	**BCSR SUB Revision**. Four bits revision coding	Programmed value	R

5.4.14 BCSR13 - Status Register

On the board, BCSR13 acts as a control register. BCSR13, which may be read or written at any time, receives its defaults upon Power-on signals. BCSR13 fields are described below in Table 5-14**.**

Table 5-14. BCSR13 Register Description

BIT	MNEMONIC	Function	DEF on PORESET	ATT.
0	PIBDRIVEPC10	**PIB Drive PC10**: If **high**, this bit instructs the PIB to drive the signal PC10. If **low**, this signal is not driven.	0	R,W
1	PIBDRIVEPC16	**PIB Drive PC16**: If **high**, this bit instructs the PIB to drive the signal PC16. If **low**, this signal is not driven.	0	R,W
2	FCFG	**FLASH Configuration**. When **high**, and configuration source set as Local Bus (CFG_RST_SOURCE) the RCW is loaded from FLASH, if **low** - from BCSR.	1	R
3	CNFLOCK	**Config Bit Lock**. When **low** BCSR contents don't update during PORESET. When **high**, BCSR contents do update during PORESET. Low provides the normal mode. ? Use for debug purpose.	1	R,W
4	USBEN	**USB1.1 Enable**: When **low**, the USB is enabled. When **high**, the USB is disabled.	0	R,W
5	USBSPEED	**USB1.1 SPEED**. When this bit is **high**, USB full Speed is selected. When this bit is **low**, USB low speed is selected.	1	R,W
6	USBMODE	**USB Mode**. Sets the USB as a host (**high**), or as a Device (**low**)	1	R,W
7	USBVCC	**USB Voltage Source**. When this bit is **low**, the MPC8360 supplies power to USB. When **high**, the USB gets power from the MINI USB connector.	1	R,W

5.4.15 BCSR14 - Board Status Register 14

On the board, BCSR14 acts as a control register. BCSR14, which may be read or written at any time, receives its defaults upon Power-on signals. BCSR14 fields are described below in Table 5-15

Table 5-15. BCSR14 Register Description

BIT	MNEMONIC	Function	DEF on PORESET	ATT.
0	MII1	**MII1 Enable**. When this bit is **high,** MII1 is enabled. When this bit is **low**, MII1 is disabled Note: in early versions, MII may not be supported, but is used to select the unused signals in RGMII/RTBI to be routed to the PIB.	0	R,W
1	MII2	**MII2 Enable**. When this bit is **high,** MII2 is enabled. When this bit is **low**, MII2 is disabled. Note- in early versions, MII is not supported, but it used to select the unused signals in RGMII/RTBI to be routed to the PIB.	0	R,W
3-7	Not Used		0	

5.4.16 CCR - COP Control Register

CCR - COP Control Register is a service register accessed from the Local Bus. It is a part of PCI2JTAG converter for the Agent Mode. The CCR fields are described below in Table 5-16

Table 5-16. CCR-COP Register Description

BIT	MNEMONIC	Function	Default upon PORESET	Attr.
0	TDI	TAP Data Input. Drives serial data into COP port.	Disabled	W
	TDO	TAP Data Output. Reads serial data from COP port.	Disabled	R
1	TCK	TAP Clock. When asserted **(low)**, TAP clock is enabled, and driven into the COP port. If negated **(high)**, TAP clock is disabled.	Disabled	W
2	TMS	TAP Mode Select. When asserted (**low)**, drives TMS signal into COP port.	Disabled	W
3	TRST	TAP Reset. When asserted (**low)**, resets TAP controller of COP port.	Disabled	W
4	HRESET	Hard Reset. **Low** provides short negative HRST pulse on the board.	Disabled	W
5	SRESET	Soft Reset. **Low** provides short negative SRST pulse on the board.	Disabled	W
6	CKSTPI	Check Stop. When asserted (**low)**, causes Machine Check Stop of the processor.	Disabled	W

Table 5-16. CCR-COP Register Description (continued)

BIT	MNEMONIC	Function	Default upon PORESET	Attr.
7	COPEN	CCR COP Enable. **Low** permits access to processor JTAG port via CCR register. **High** disables the CCR register.	1	W

5.5 External Connections

External connections locations are shown in Figure 5-6, below.

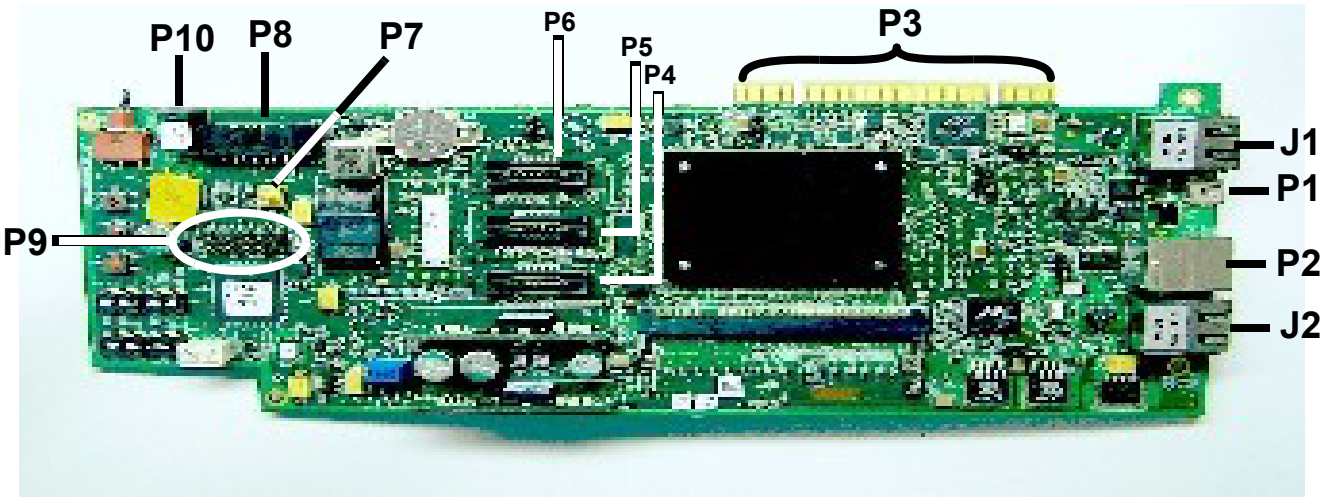


Figure 5-6. External Connections for the MPC8360EA MDS Processor Board

5.5.1 P1 - MiniAB USB Connector

The MiniAB USB connector pinout is shown in Table 5-17, below. This connector is used for connectivity to external devices USB1.1. It is accessible from the front panel of the board (see **Figure 1-1** for location).

Table 5-17. P1-MiniAB USB Connector

Pin No.	Signal Name	Description
1	Vbus	5V Power for USB - Power is generated internally if working in PCI mode or is supplied from a cable (in stand-alone mode) while the USB controller configures the device. Note also BCSR13[7]
2	DM	Differential Negative Data
3	DP	Differential Positive Data
4	ID	Identification Signal for Host/Device Mode Setting (PCI mode vs. stand-alone mode)

Table 5-17. P1-MiniAB USB Connector (continued)

Pin No.	Signal Name	Description
5	GND	Ground

5.5.2 P2 - DUART Port

The DUART port connector - P2 is implemented with a 90^{o}, 10-pin, RJ45 connector, signals of which are described in Table 5-18

Table 5-18. DUART Port Description

Pin No.	Signal Name	UART Port	Attr.	Description
1	CTS1	1	I	Clear To Send
2	RXD1		I	Receive Data
3	TXD1		O	Transmit Data
4	RTS1		O	Ready To Send
5,10	GND		P	Ground.
6	CTS2	2	I	Clear To Send
7	RXD2		I	Receive Data
8	TXD2		O	Transmit Data
9	RTS2		O	Ready To Send

For connection to regular D-Type-9 RS232 cable use special cable from MPC8360EA MDS Processor Board set.

5.5.3 P3 - 32-bit PCI Edge Connector

P3 is a PCI 32-bit edge connector, used when the MPC8360EA MDS Processor Board functions as an agent, in either a PC, or inserted in an adaptor on the PIB (see Figure 2-14).

5.5.4 P4, P5, P6 - Logic Analyzer Connectors

P4, P5, and P6 are 38-pin, SMT, high density, matched impedance connectors made by Tyco and used for Logic Analyzer measurements. They contain local bus MPC8360 signals.

5.5.5 P7 - SMB Connector

RF Subminiature Coaxial Connector P7 is used to connect an external clock to the MPC8360EA, which is enabled only when jumper JP6(2-3) is closed. Optional.

5.5.6 P8 - Debug COP Connector

P8 is a Freescale-standard JTAG/COP connector for the PowerPC. It is a 16 pin 90^{o} two row header connector with key. During debug, all processors connected by the JTAG chain may be accessed through connector P8. The pinout of P8 is shown in Table 5-19. "P8 - JTAG/COP Connector" below:

Table 5-19. P8 - JTAG/COP Connector

Pin No.	Signal Name	Description
1	TDOc	Transmit Data Output. This is the MPC8360EA JTAG serial data output driven by Falling edge of TCK.
2,10,12, 16	GND	Main GND plane.
3	TDIc	Transmit Data In. This is the JTAG serial data input of the MSC8101, sampled on the rising edge of TCK.
4	nTRSTc	Test port Reset. When this signal is active (Low), it resets the JTAG logic. This line is provides a pull-down on the ADS with a 4.7KΩ resistor, which provides a continuous reset of the JTAG logic, when connector is unplugged.
5	N.C.	Not Connected.
6	SENSE	Connect to 3.3V power supply bus via protection resistor. May be used for Command Convertor power.
7	TCKc	Test port Clock. This clock shifts in / out data to / from the JTAG logic. Data is driven on the falling edge of TCK and is sampled both internally and externally on its rising edge.
8	Check Stop Input	Machine Check Stop Input
9	TMSc	Test Mode Select. This input selects test mode and is sampled on the rising edge of TCK. This line is qualified with TCK in a same manner as TDI, and changes the state of the JTAG machines. This line is pulled up internally by the MPC8360EA.
11	nSRSTc	When asserted by an external H/W, generates Soft Reset sequence for the MPC8360EA. Pulled Up on the ADS using a 4.7KΩ resistor. When driven by an external tool, **MUST be driven with an Open Drain gate. Failure to do so might result in permanent damage to the processor and / or to ADS logic.**
13	nHRSTc	When asserted by an external H/W, generates Hard Reset sequence for the MPC8360EA. Pulled Up on the ADS using a 4.7KΩ resistor. When driven by an external tool, **MUST be driven with an Open Drain gate. Failure to do so might result in permanent damage to the processor and / or to ADS logic.**

Table 5-19. P8 - JTAG/COP Connector (continued)

Pin No.	Signal Name	Description
14	KEY	No pin in connector. Serves for correct plug insertion. See Figure 5-7 for location.
15	Check Stop Output	Machine Check Stop Output

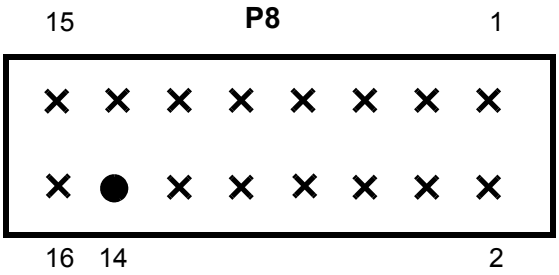


Figure 5-7. P8 COP connector front view

5.5.7 P9 - FPGA's In-System-Programming (ISP)

This is a 16 pin generic 0.100" pitch header connector, providing In System Programming capability for on board programmable logic devices by Xilinx FPGA (Spartan-2E). The pinout of P9 is shown in Table 5-20. "P9 - FPGA Programming ISP Connector" below:

Table 5-20. P9 - FPGA Programming ISP Connector

Pin No.	Signal Name	Attr.	Description
1	ISP_TDO	I	Transmit Data Output.
2,10,12, 16	GND	P	Main GND plane.
3	ISP_TDI	O	Transmit Data In.
4,5,8,11,1 3,14,15	N.C.	-	Not Connected.
6	SENSE	P	Connect to 3.3V power supply bus via protection resistor. Use for programmer powering.
7	ISP_TCK	O	Test port Clock.
9	ISP_TMS	O	Test Mode Select.

5.5.8 P10 - Power Connector

P10 is 2mm Power Jack RAPC722 which provides a connection to an external power supply +5DC@2.5A.

5.5.9 J1,J2 - GETH Port Connector

The GETH connectors on the MPC8360EA (J1,J2) are both Twisted-Pair (1000-Base-T) compatible connectors. They are implemented with a 90^{o}, 8-pin, RJ45 Combo connector with internal magnetics and two LEDs (indicating communication speed), signals of which are described in Table 5-21. "J1, J2 - GETH Port Interconnect Signals" below. These connections are on the front panel. For location, see Figure 1-1. Green LED indicates 1000Mbit Data rate, Yellow LED is lit when 100Mbit Data rate mode. .

Table 5-21. J1, J2 - GETH Port Interconnect Signals

Pin No.	Wire Color	10Base-T/100Base-T Signal	1000 Base-T Signal
1	White	Twisted-Pair Transmit Data positive **output**	BI-DA+
2	White-Orange	Twisted-Pair Transmit Data negative **output**.	BI-DA-
3	White-Green	Twisted-Pair Receive Data positive **input**.	BI-DB+n
4	Blue	Unused	BI-DC+
5	White-Blue	Unused	BI-DC-
6	Green	Twisted-Pair Receive Data negative **input**	BI-DB-
7	White-Brown	Unused	BI-DD+
8	Brown	Unused	BI-DD-

5.6 PCI

5.6.1 General

The MPC8360EA PCI interface allows the MPC8360EA MDS Processor Board to function as either a PCI host or as a PCI peripheral device (referred to as agent mode).

The PCI port is muxed with the ATM port. In order to enable PCI signals, and disable ATM signals, the value of PCI_MODE must be set to “0”. This can be set via DIP-Switches (SW9[1])or BCSR10[4] (this pin is driven constantly).

There are four MPC8360EA MDS Processor Board operation configurations, each of which uses the PCI in host, or agent mode (indicated):

- Stand alone - Host mode, but PCI interface not used
- Mounted on PIB through Riser Connectors - Host mode
- Mounted on PIB through PMC Connectors - Agent mode
- Installed on PC - Agent mode

The PCI controller mode of operation is determined at reset by values of the reset configuration word high (RCWH). These can be seen in BCSR4, bits PCIHOST, PCIARB, and PCICKDRV as shown in the following two sections.

5.6.2 PCI Setting when MPC8360EA MDS Processor Board is Host

If the MPC8360EA MDS Processor Board is in a stand-alone configuration, or on the riser connectors of the PIB (that is, not inserted in a PC, nor inserted on the PIB as an agent), the FPGA configures the RCW to PCI Host by setting BCSR4[0] to "1". The PCI port should be set to internal arbitration (BCSR4[2] = "1"), and the PCI Clock Output Drive should be set to high (BCSR4[3] = "1"). Also note that in Host Mode the CLKIN pin receives a 66Mhz clock from an external (to the chip) clock oscillator. This is located on the Processor Board.

The following table (Table 5-22) shows PCI settings that determine the communication settings between the Processor Board and the PMC slots when the MPC8360EA MDS Processor Board is mounted on the PIB as a Host.

The configuration setting is done using the I2C-2 bus, and the device addresses relate to I2C-2 addresses. The I2C device controls its setting using the PCA9555PW (an I/O expander). For more details on this, see the *PIB User's Manual*.

Table 5-22. PCI Settings for PIB

	Mode	device address	register address	value
	All modes:	0x23	6	0
	All modes:	0x26	6	0x0034
	All modes:	0x27	6	0
	All modes:	0x23	2	0xFFFF
choose one entry according to working mode	All modes:	0x27	2	0xFFEF
	PCI - Expansion Board 32 bit	0x26	2	0xF8FF
	PCI - PMC1 32-bit	0x26	2	0xFFFF
	PCI - PMC1 & PMC2 32-bit	0x26	2	0xF7FF
	PCI2 - PMC3 32-bit	0x26	2	0xFDFF
	PCI2 - PMC2 & PMC3 32-bit	0x26	2	0xF9FF

5.6.3 PCI Setting when MPC8360EA MDS Processor Board is Agent

If the MPC8360EA MDS Processor Board is inserted in a PC, or inserted on a PMC slot on the PIB, the FPGA configures the RCW to PCI Agent by setting BCSR4[0] to "0". The PCI port should be set to

external arbitration (BCSR4[2] = “0”), and the PCI Clock Output Drive should be set to low (BCSR4[3] = “0”).

Also note that in Agent Mode the CLKIN pin is neutralized, and the SYNCIN pin receives a clock from an external (to the board) clock oscillator via the P3 edge connector.

The MPC8360EA MDS Processor Board is compatible with PCI specification Revision 2.2. The PCI Interface is 3.3V/5V (32 bit). It uses a 32-bit multiplexed, address/data bus that can run from 25MHz up to 66MHz.

5.7 DDR

The DDR SDRAM Interface supports a 400MHz bus in both 72-bit or 2x 40bit widths.

- **72 bit** (64bit data, 8bit ECC): The MPC8360EA MDS Processor Board supports a 72-bit data width by using a DDR SODIMM.
- **2 x 40bit** (32bit data, 8bit ECC): Implementation of 2 x 40bit DDR interface is shown in Figure 5-8.

To implement one 72-bit bus or 2 x 40-bit buses, two SODIMM sockets are mounted on the board, (M1, M2), one of which (M2) has 72-bit capability that can be configured to work at either 40-bits HIGH or at 72-bits.

For 40-bit HIGH, the WV3HG32M72EEU403PD4GG from White Electronics is used in M1 (mounted on U33, see Figure 7-7). Note that it is the only SODIMM for which 40-bit HIGH only is populated.

For the 2 x 40-bit mode, the same WV3HG32M72EEU403PD4GG from White Electronics is used in M1 (mounted on U33, see Figure 7-7), and a standard 72-bit SODIMM module is configured for 40-bit LOW, and used in M2 (mounted on U69, see Figure 7-6).

For the 72-bit mode, only one SODIMM module should be used in M2 (mounted on U69, see Figure 7-6).

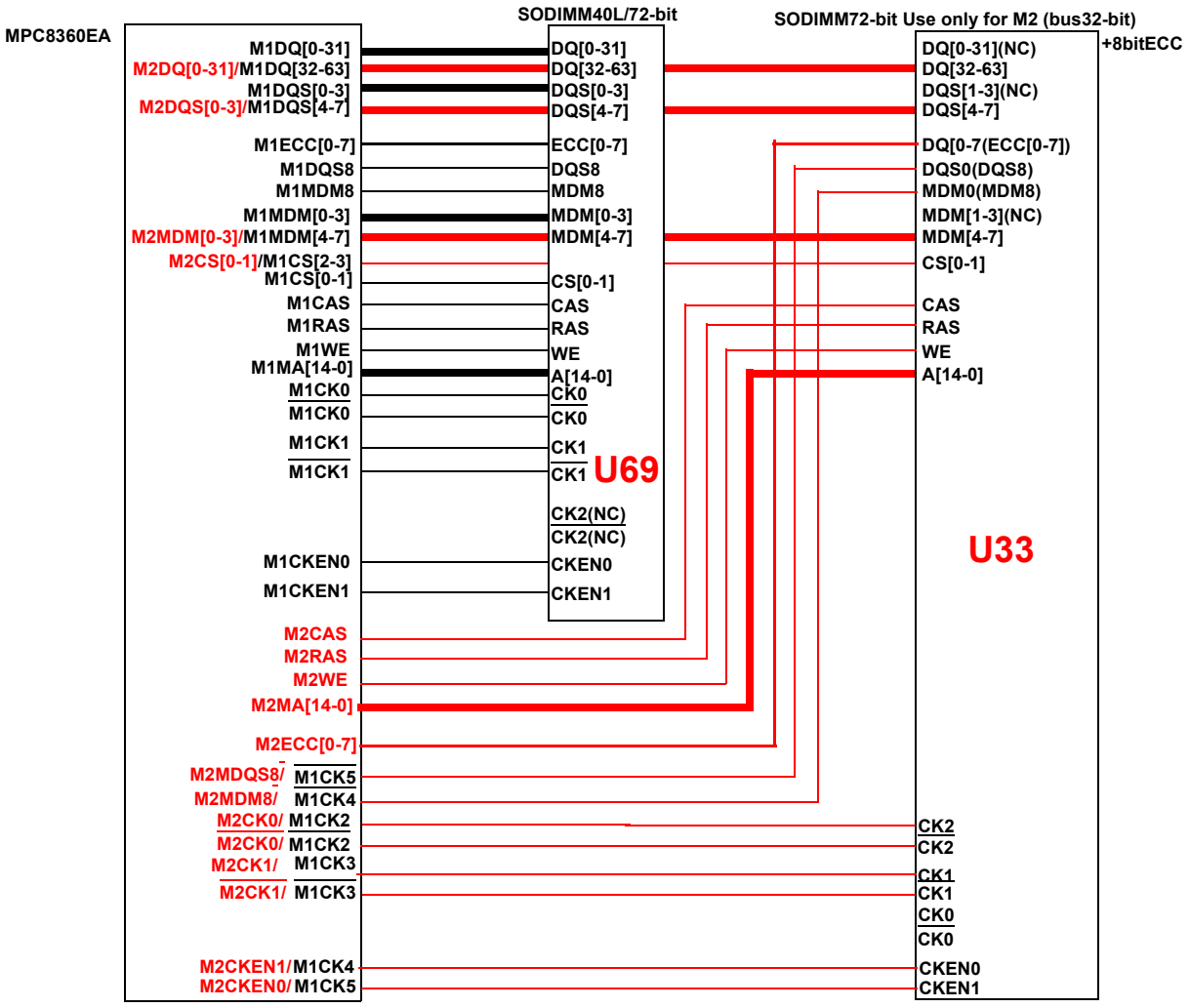


Figure 5-8. DDR Connections Block Diagram

5.8 Local Bus

This section describes devices that are connected to the local bus of the MPC8360EA MDS Processor Board.

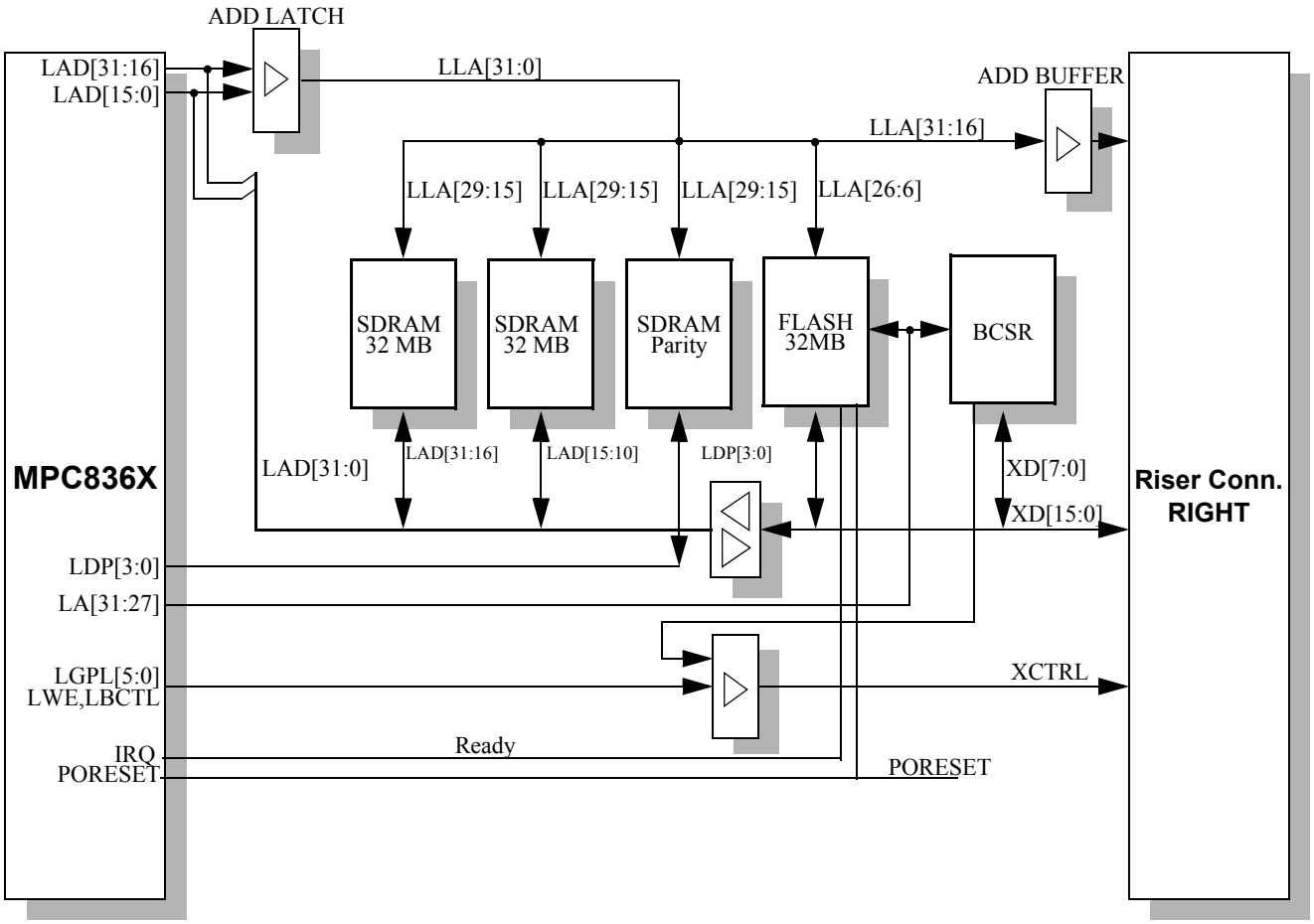


Figure 5-9. Local Bus Scheme

5.8.1 Address Latch/ Data Transceiver

The address latch buffer used is Texas Instruments' SN74ALVTH32373GKE. The address latch buffer latches the addresses and drives them to the fast bus (which includes SDRAM components). It also drives the address to an extra 16-bit buffer (TI's SN74ALVCH32244KR) for the slow bus, which connects to Flash and BCSR, in addition to PIB components (via riser connectors).

The data transceiver is On-Semi's MC74LCX245DT. One side of the buffer is connected to the MPC863X LAD[0:31]. Only Flash, BCSR and PIB Riser connectors are connected to the other side of the data transceiver.

5.8.2 SDRAM

The SDRAM memory is implemented using three of Micron's MT48LC16M16A2TG-6A units. They are organized as 4 Banks x 4M x 16bits, where all input and output levels are compatible with LVTTL. One is configured as a 64Mbyte bank. The other two devices are 32Mbyte, and are used in order to achieve a capacity of 64Mbyte plus 1 device for four bits parity. There is a total of 3 MT48LC16M16A2TG-6 devices. The device has 13 Rows, 9 Column and 2 bits for bank select.

Table 5-23 below describes the local bus address interface to the SDRAM.

The first row in the table (LB ADD) shows the local bus address from A29 to A6 (the logical address). This row also shows that the columns start at A21 and continue to A29. The bank select uses logical address A19 and A20 and the row address uses local bus A6 to A18.

The next row in the table (ROW FOLD) shows how the row address folds over the column address - for example, we can see that the bank select internal A19 and A20 will go out on A15 and A16.

The next row in the table (SDRAM ADD) shows the address pins from the SDRAM point of view. We can see that A10, operates with the command and comes with the rows so that command A10 comes from A8 (LSDA10 is connected to SDRAM A10). The bank selects signals are the MSB address bits in the SDRAM and are latched with the row addresses- that is, they will placed after the rows signals on A15 and A16. The signals connected to the SDRAM is LAD [15:29] while SDRAM A10 is connected to LSDA10.

During the first phase of memory access, the LALE will latch Local Bus Address [6:18] plus bank select A19:A20 on A15:A16. The local bus will then drive LRAS, and the SDRAM device will latch the row and bank select. During the second phase of memory access, the LALE will latch the column on A21:A29, then the local bus will drive LCAS and finally the SDRAM will latch the column.

The parity device is the same device as for D[0:31], as it has the same parameters. The Local Bus Data Parity LDP [0:3] is connected to the SDRAM data D[3:0], and the local bus LPBS signal is connected to the SDRAM DQM. The address pins of the parity device are connected to the same pins as the data devices.

Table 5-23. SDRAM Connection to Local BUS

	Row Connection 13 lines													Bank Select		Column Connection 9 lines								
LB ADD	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29 (LSB)
ROW FOLD										19	20	6	7	8 LSDA10	9	10	11	12	13	14	15	16	17	18
SDRAM ADD										BA1	BA0	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
Second ALE to LATCH																Column num of COLS 9 lines								
First ALE to LATCH										Bank Sel BSMA = LB (19-20)		ROW num of ROWS 13 lines												
												Address Multiplexed A(6:18) over A(21:29)												

5.8.3 Flash Memory

The Flash memory used (see also Section 7.1 on page 7-1) is the S29GL256M from Spansion. The standard unit has 256Mbit, but can be upgraded to 512Mbit by simply inserting the larger volume device in the socket. The Flash is connected to the slow bus. The slow bus is organized in such a way that the data is obtained from the data transceiver, and the address is obtained from the address latch buffer.

The Chip Select signal is connected to the CS0~ line for booting from the Flash. A special buffer for control signals is used in order to minimize the load on the local bus control signals, which are already used on the SDRAM. Local bus LOE~ control Flash OE and LBS [0-1] control Flash WE~ signals.

The Local Bus A7 is routed to the flash for optional expansion.

The flash output STS signal indicates that either a programming or erase function is being done - this signal is connected to MPC836x interrupt line.

In order to boot from FLASH SW9[3] should be set to '1'.

5.9 GETH

GETH ports features are as follows:

- Support for GMII, TBI, RGMII and RTBI
  - (The GETH ports are compatible with RGMII for 10/100-BaseT or GMII, RGMII, TBI and RTBI for 1000Base-T)
  - Default mode: GMII
- When working in PIB Combined Mode (attached to riser connections on PIB) GMII, TBI, RGMII and RTBI are supported on both the MPC8360EA MDS Processor Board and on the PIB.
- Two IEEE 802.3 compliant GETH ports with T.P. (10/100/1000-Base-TX) I/F
- Two PHYs from Marvel (88E1111), connected to ENET1 & ENET2 Pins.
- The two PHYs are controlled and configured via the Processor Board's BCSR8[0-5] in all modes
  - (It is also possible to configure to other PHY configuration modes via the PHY internal registers)

Note also the following:

Some of the PHY configuration parameters require a Soft Reset to activate the new configuration value. See PHY documentation (from Marvel) for a list of such parameters.

The PHY reset input is driven by either an assertion of HRST (Hard Reset), or by writing zero to BCSR9[2]. The registers BCSR9[0,1] are the enable bits for each of the PHYs. Note that whenever BCSR9[2] is changed from low to high, the value of BCSR8[0-5] is driven to the PHY to configure the hardware setting.

The PHYs reset any time an HRST (Hard Reset) sequence is taken. The PHY may execute a Soft Reset by asserting bit 15 (MSB) of the 88E1111 control register 0 via the MDC & MDIO signals.

When the PIB is used *only the reduced modes (RGMII and RTBI) are recommended*.

Section 5.9.1 to Section 5.9.6 describe in more detail each PHY mode.

5.9.1 GMII Interface

This interface is the default interface upon Power On.

This interface is recommended on this board for 1000Base-T speed. Note for 100Base-T and 10Base-T modes RGMII interface should be selected (see Section 5.9.3 on page 5-32).

Figure 5-10. below indicates the signal mapping of the 88E1111 device to GMII interface. The GMII interface supports GMII to copper or GMII to fiber connections in 1000Base-T speed. The GMII interface is selected by setting the 88E1111 HWCFG_MODE [3-0] to 0b1111. It can also be controlled by BCSR8[0-3] &BCSR9[0-2].

If using 1000Base-T speed, a 125MHz input to the MPC8360 is taken from the PHY. Each one of the PHYs is driving its own 125MHz clock to the appropriate UCC. The MPC8360EA GMII interface transmits a 125MHz clock to the PHY GTX_CLK pin.

> Note: The COL signal is not used in this board although it appears in the schematics.

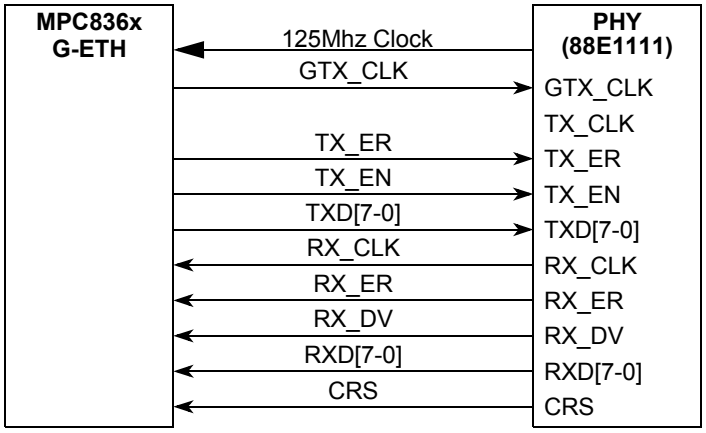


Figure 5-10. GMII Interconnections

5.9.2 Ten Bit Interface (TBI)

The TBI interface pin mapping is shown in Table 5-24. The TBI interface supports 1000Base-T mode of operation only. The TBI-to-copper interface is selected by software through the MDC and MDIO pins or by BCSR8[0-5]. For more details, see Section 5.4.9 on page 5-13.

Table 5-24. TBI Signals

TBI Signal Names	
TBI Signal Name	**PHY Signal Name**
TBI_TXCLK	GTX_CLK
RXCLK1	TX_CLK
TXD9	TX_ER
TXD8	TX_EN

Table 5-24. TBI Signals (continued)

TBI Signal Names	
TBI Signal Name	**PHY Signal Name**
TXD[7-0]	TXD[7-0]
RX_CLK0	RX_CLK
RXD9	RX_ER
RXD8	RX_DV
RXD[7-0]	RXD[7-0]
external PU	CRS

As shown in Figure 5-11 below, the TBI uses the same signals as the GMII interface.

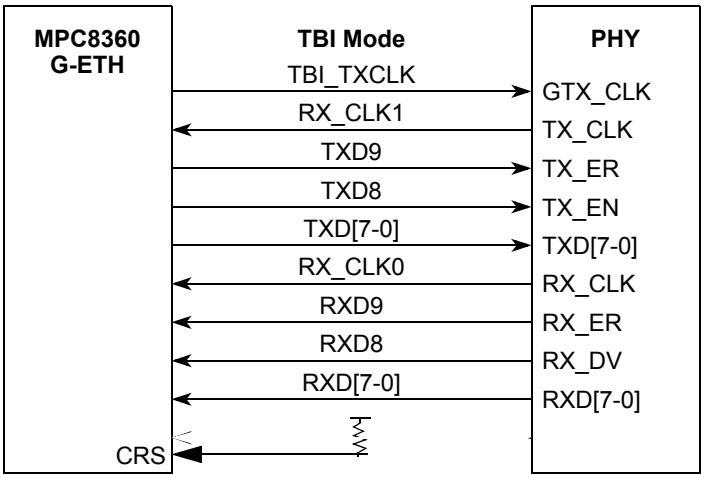


Figure 5-11. TBI Signal Diagram

5.9.3 Reduced Pin Count GMII (RGMII)

This option should be used if 10 or 100Base-T speed is desired. This interface reduces the number of pins between the PHY and the MPC836x device to 12 pins. The RGMII-to-copper interface is selected by software through the MDC and MDIO pins or by BCSR8[0-5]. See Section 5.4.9 on page 5-13 for more details. In this mode of operation, the data is transmitted and received on both clock edges. The signals used in this mode are shown in Table 5-25.

Table 5-25. RGMII Signals

RGMII Signal Name	PHY Signal Name
GTX_CLK	GTX_CLK
TX_EN	TX_EN

Table 5-25. RGMII Signals (continued)

RGMII Signal Name	PHY Signal Name
TXD[3-0]	TXD[3-0]
RX_CLK	RX_CLK
RX_CTL	RX_DV
RXD[3-0]	RXD[3-0]

Figure 5-12 below shows the signal mapping between the MPC8360EA device and PHY in RGMII mode.

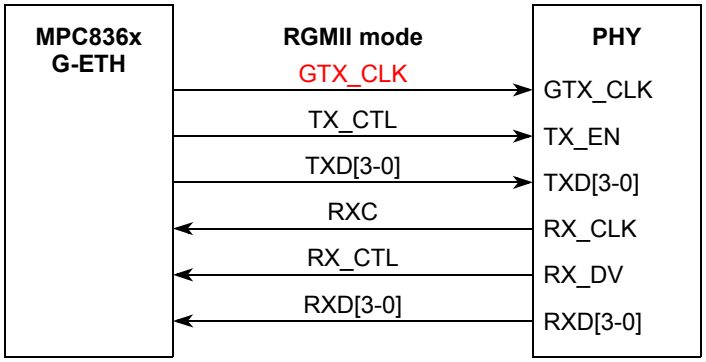


Figure 5-12. RGMII Signal Mapping

5.9.4 Reduced Ten Bit Interface (RTBI)

The RTBI interface pin mapping is shown in Table 5-26. The RTBI supports only 1000Base-T. This interface reduces the number of pins between the PHY and the MPC8360EA device to 12 pins. The RTBI-to-copper interface is selected by software through the MDC and MDIO pins or by BCSR8[0-5] and BCSR9[0-2].

Table 5-26. RTBI Signals

RTBI Signal Name	PHY Signal Name
GTX_CLK	GTX_CLK
TD4_TD9	TX_EN
TD [0-3]	TXD [3-0]
RCX	RXCLK
RD4_RD9	RX_DV
RD [3-0]	RXD [3-0]

Figure 5-13 shows the signal mapping between the MPC8360EA device and PHY in RTBI mode.

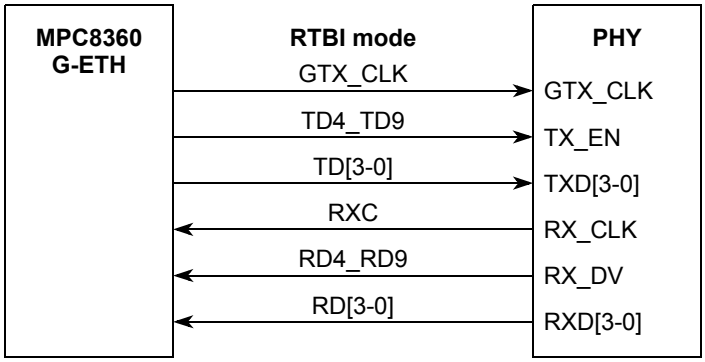


Figure 5-13. RTBI Signal Mapping

5.9.5 RGMII or RTBI via the PIB

In order to use the maximum number of pins for functions between the processor board and the PIB, configure UCC1/2 to use the RGMII or RTBI protocols, which use fewer pins than GMII or TBI. The table below shows how to do this.

Note that in this case, the GETH-PHY1 connection and the MII1 must be disabled.

Table 5-27. Selecting GMII/RGMII/TBI/RTBI or PIB

BCSR8[0,1]	BCSR9[0]	BCSR14[0]	
Select Geth1 Mode	**Enable/Disable Geth-PHY1**	**Enable/Disable MII1**	**Function Selected**
00 (RGMII)	0 - **Enable**	0 - **Disable**	RGMII
00 (RGMII)	0 - **Enable**	1 - **Enable**	MII
00(RGMII)	1 - **Disable**	0 - **Disable**	Ethernet signals use PIB connections
00 (RGMII)	1 - **Disable**	1 - **Enable**	MII
01 (RTBI)	0 - **Enable**	0 - **Disable**	RTBI
01 (RTBI)	0 - **Enable**	1 - **Enable**	MII
01 (RTBI)	1 - **Disable**	0 - **Disable**	Ethernet signals use PIB connections
01 (RTBI)	1 - **Disable**	1 -**Enable**	MII

5.9.6 RMII via the PIB

The PIB contains the RTL8208 PHY device, which supports an 8-port integrated physical layer and transceiver for 10Base-T and 100Base-TX. When used with the MPC8360EA MDS Processor Board, 6 of these 8 ports can be utilized, and are connected to UCC(3-8).

The input clock to the RTL8208 device is 50MHz, arriving from the clock oscillator. This input clock is split to 2x50MHx lines: one to the MPC8360EA Clock-16, and the other to Clock-7. Clock-16 is the only clock that can supply the EVEN UCC RMII or the ODD UCC. Clock-7 can be used for the Even-numbered UCCs.

Figure 5-14 below is a block diagram showing the connection scheme of the RMII PHY on the PIB to the MPC8360EA MDS Processor Board.

Note that ENET6 & ENET7 share the same pins with TDM & UPC1. Therefore a bus switch must be placed on the sharing signals in order to buffer the signals for a short trace. In order to do this, the following should be written to I2C (address 0x26): to Reg 6 write 0x7ffc, and then to Reg 2 0x7ffe. That is:

```
0x26 0x06 0x7ffc
0x26 0x02 0x7ffe
```

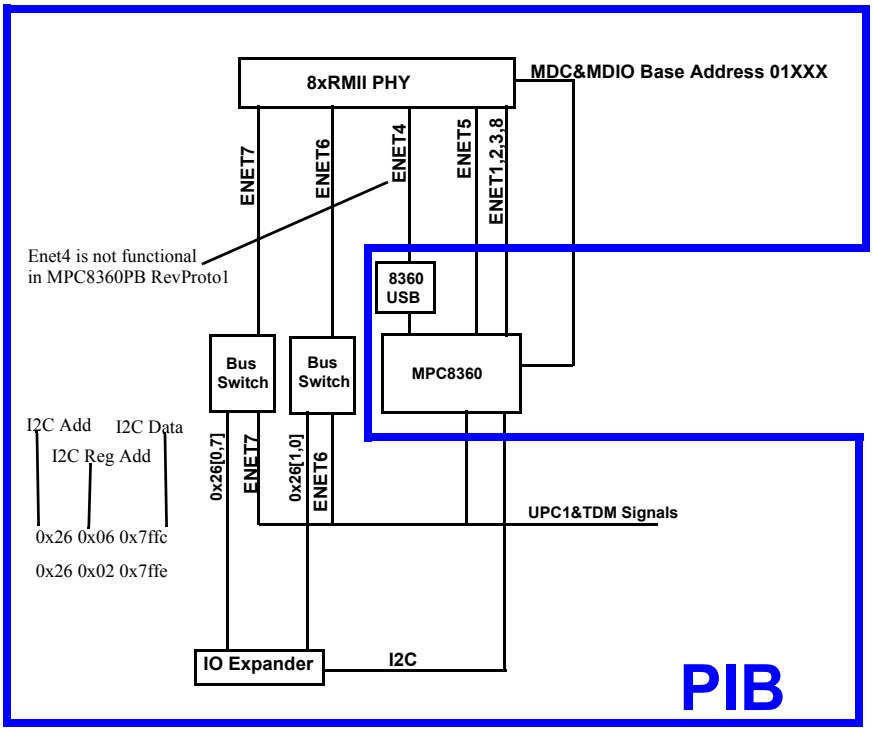


Figure 5-14. RMII as it traverses the PIB

Table 6-5 describes each one of the UCC(3-8) pins used for RMII in the PIB. It describes a complete trace of all signals from the MPC8360 device pins, to the riser connectors on the MPC8360EA MDS Processor Board, to the PMC connections on the PIB, and finally to the RMII functions named on the PIB.

5.9.7 Summary

Table 5-28 shows the MPC8360EA device G-ETH pins used for each of the above-mentioned interface modes.

Table 5-28. G-ETH Pin Use

G-ETH Pin Name	GMII	RGMII	TBI	RTBI
GT_CLK	GT_CLK	TXC	TBI_TXCLK	TXC
TX_CLK	—	—	RCLK1	—
TX_ER	TX_ER	—	TXD9	—
TX_EN	TX_EN	TX_CTL	TXD8	TD4_TD9
TXD [7-0]	TXD [7-0]	TD [3-0]	TXD [7-0]	TD [3-0]
RX_CLK	RX_CLK	RXC	RXCLK0	RXC
RX_ER	RX_ER	—	RXD9	—
RX_DV	RX_DV	RX_CTL	RXD8	—
RXD [7-0]	RXD [7-0]	RXD [3-0]	RXD [7-0]	RD [3-0]
CRS	CRS	—	--	—
COL		—	—	—

Table 5-29 below summarizes what values to write to which registers in order to set the G-ETH mode.

Table 5-29. Selecting GMII/RGMII/TBI/RTBI or PIB

BCSR8[0,1]	BCSR9[0]	BCSR14[0]	
GETH Mode	**Enable/Disable Geth-PHY1**	**Enable/Disable MII1**	**Function Selected**
00 (RGMII)	0 - **Enable**	0 - **Disable**	RGMII
00 (RGMII)	0 - **Enable**	1 - **Enable**	MII Signals not used for MII bus connected to PIB.
00(RGMII)	1 - **Disable**	0 - **Disable**	Relevant signals connected to PIB
00 (RGMII)	1 - **Disable**	1 - **Enable**	MII (not supported yet)
01 (RTBI)	0 - **Enable**	0 - **Disable**	RTBI
01 (RTBI)	0 - **Enable**	1 - **Enable**	MII (not supported yet)
01 (RTBI)	1 - **Disable**	0 - **Disable**	Relevant signals connected to PIB

Table 5-29. Selecting GMII/RGMII/TBI/RTBI or PIB (continued)

BCSR8[0,1]	BCSR9[0]	BCSR14[0]	
GETH Mode	**Enable/Disable Geth-PHY1**	**Enable/Disable MII1**	**Function Selected**
01 (RTBI)	1 - **Disable**	1 -**Enable**	MII (not supported yet) Signals not used for MII bus connected to PIB.
10 (GMII)	0 - **Enable**	0 - **Disable**	GMII
10 (GMII)	0 - **Enable**	1 - **Enable**	GMII
10(GMII)	1 - **Disable**	0 - **Disable**	All signals connected to PIB
10 (GMII)	1 - **Disable**	1 - **Enable**	MII (not supported yet) Signals not used for MII bus connected to PIB.
11 (TBI)	0 - **Enable**	0 - **Disable**	TBI
11 (TBI)	0 - **Enable**	1 - **Enable**	MII (not supported yet) Signals not used for MII bus connected to PIB.
11 (TBI)	1 - **Disable**	0 - **Disable**	All signals connected to PIB
11 (TBI)	1 - **Disable**	1 -**Enable**	MII (not supported yet) Signals not used for MII bus connected to PIB.

5.10 USB

The USB module is a standard Universal Serial Bus for implementing a USB interface in compliance with the USB 1.1 specification. It is able to act as a device, or host controller, and provides interfaces to negotiate the host or peripheral role on the bus. A dedicated USB transceiver - made by Fairchild USB1T11AMX is provided. A MiniAB connector (see Section 5.5.1 on page 5-20) provides functioning in both device and host modes.

To correctly support the two different speed modes (Full Speed - 12Mbit/s and Low Speed - 1.5Mbit/s) when being used either as a device or as a host, a detachable pull-up resistor is provided over the D+ line of the USB, controlled by the USB_SPEED bit of BCSR13.

When functioning as a host and when USB_SPEED is in low-speed level (low) D- is pulled-up while D+ remains floating. When the USB_SPEED bit is in high-speed level, D+ is pulled-up and D- floats. When functioning as a device, D+ and D- are both connected directly to the line.

The Vbus Voltage is monitored by Charge pumps that provide 5V power on Vbus and monitor the power fault on the MiniAB connector. A Vbus detection failure on the USB 1.1 Serial Mode is reported to the MPC836x by an IRQ signal.

A 48Mhz clock is input to the MPC8360EA pin PC20 (clock-21) and BRG15. The 48Mz is used for full speed and can be divided internally. To use this clock for Low speed, use BRG9.

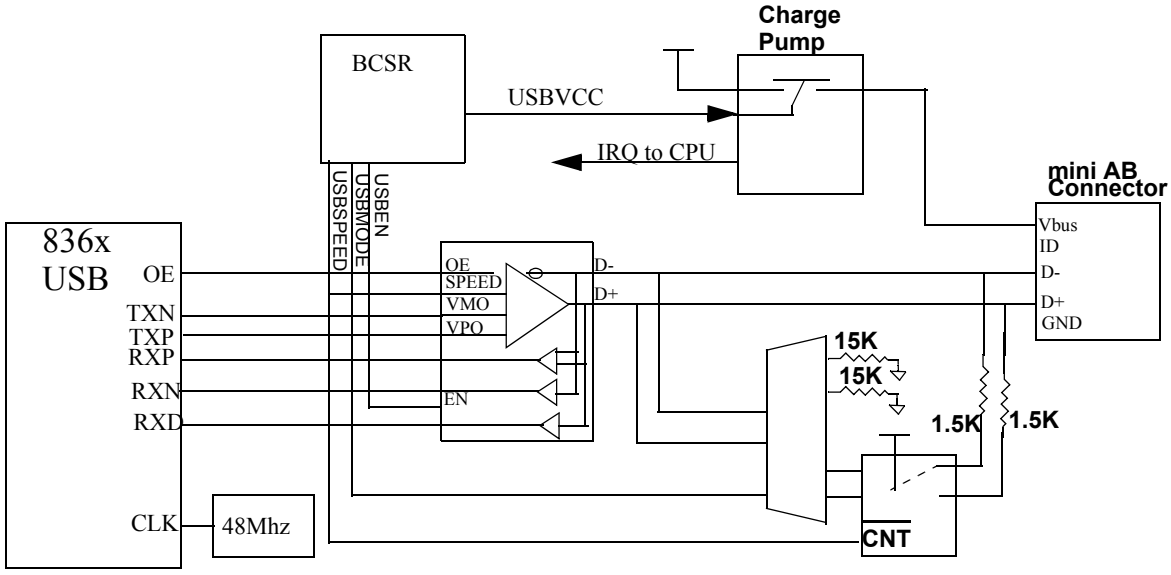


Figure 5-15. USB 1.1 Block

Table 5-30 below (also reproduced in Table 6-4) describes the USB signals with their other functions options. The table shows:

- The pin selection from the MPC8360EA,
- The location pin on the Riser connectors that connect the PIB to the MPC8360EA MDS Processor Board,
- The pin name used on the PIB,
- The pin on the PMC on PIB,
- The pin function for the GMII/TBI on the MPC8360EA MDS Processor Board UCC1 & UCC2 or TDMG,
- The pin function when the PIB uses UCC4, UPC1_TXEN2,
- And finally, the pin name used by the Universal Serial Bus (USB).

Table 5-30. USB Signals and their mix with other functions

MPC8360EA Pin Name	Riser Conn	Signals name PIB	PMC on PIB	Function2 RGMII	Function1 RMII	Function3
PB2	P12-D13	GPIO30	PMC1-J3-55	Enet2-TXD[4]	Enet4-RXD[1]/RXD[3]	USB_OE
PB3	P12-D14	GPIO31	PMC1-J3-25	Enet2-TXD[5]	Enet4-RX_DV/RX_CRS	USB_TP
PB8	P12-E12	GPIO36	PMC1-J3-31	Enet2-TXD[7]	Enet4-RXD[0]/RXD[2]	USB_TN

Table 5-30. USB Signals and their mix with other functions (continued)

PB9	P12-E14	GPIO37	PMC1-J3-35	Enet1-TXD[6]	Enet4-TXD[0]/TXD[2]	USB_RP
PB10	P12-E15	GPIO38	PMC1-J3-37	Enet1-TXD[7]	Enet4-TXD[1]/TXD[3]	USB_RXD
PB11	P12-E17	GPIO39	PMC1-J3-41	Enet2-RXD[7]	Enet4-TX_EN	USB_RN
PC20	P12-C3	GPIO76	PMC0-J1-17	TDMG_TXCLK	XUPC1_TXEN2	USB-CLK CLK21 BRG9

5.11 Debugging Applications

5.11.1 Stand-Alone and on PIB

Chip debugging is provided via the JTAG port. While the MPC8360EA MDS Processor Board functions as a host (on PIB riser connectors, or as a Stand-Alone), the standard 16-pin JTAG/COP connector (P8, see also Section 5.5.6) is used to connect a USB Tap to which a PC with *CodeWarrior®* is connected.

5.11.2 Inserted in PC

When the MPC8360EA MDS Processor Board functions as an agent (plugged into a PC's PCI slot), the access to JTAG/COP interface is available via the PCI bus and a special register implemented in the FPGA called the CCR (described below). In this case, the PC acts as a host, and debugging is carried out using *CodeWarrior®* (or a similar IDE) installed on the PC.

For debugging purposes, the PC host may download program code to an inbound memory window in the address space of the MPC8360EA MDS Processor Board (e.g. DDR, SDRAM) on its local bus. The access to the JTAG/COP debug port can be reached by using the COP control register (CCR). The CCR is mapped on the local bus memory space, which allows it to be "seen" from the PCI host. Manipulating bits in the CCR allows the PCI host to manage a serial access to the Processor Debug Port in the same manner as an on-board command converter. Figure 5-16 below illustrates this access concept.

Table 5-31 shows a list of CCR bits and their possible values and attributes. Figure 5-17 below illustrates the access path to the COP from the host via the PCI bus.

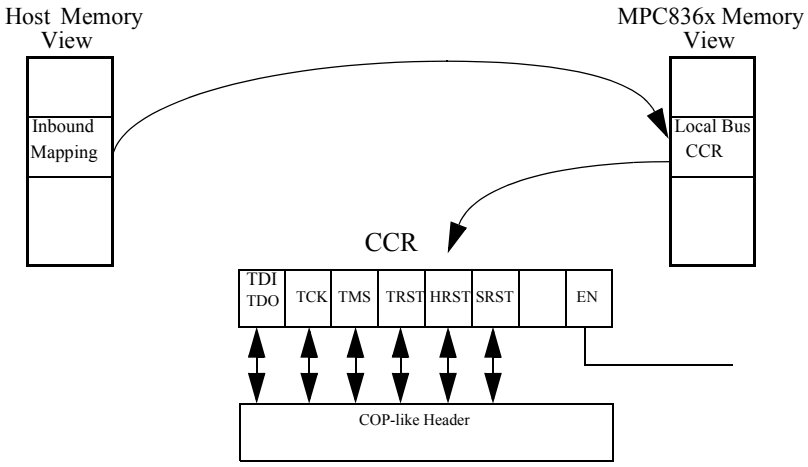


Figure 5-16. CCR PCI Local Bus Concept

The CCR bit fields are described below:

Table 5-31. CCR Description

Bit	Mnemonic	Function	Default POR Value	Attr.
0 (MSB)	TDI	Test Data In	Hi-Z	W/O
	TDO	Test Data Out	-	R/I
1	TCK	Test Clock	Hi-Z	W/O
2	TMS	Test Mode	Hi-Z	W/O
3	TRST	Test Reset	Hi-Z	W/O
4	HRESET[a]	Hard Reset	Hi-Z	W/O,OD
5	SRESET	Soft Reset	Hi-Z	W/O,OD
6	-	Reserved	-	-
7 (LSB)	nCOP_EN	High - Enables the CCR COP signals on the debug port Low - Allows a different COP debug source	Hi	R/W

[a] Hreset or Sreset bits assertion causes a reset to the processor's register. They receive the default value that causes a reset in the Local Bus Controller. To access CCR again, the prior Local Bus Controller values should be configured.

Note: R or W– Read or Write option
I or O– Presents as an input/output signal
OD– Open-Drain driver
Hi-Z - Three state

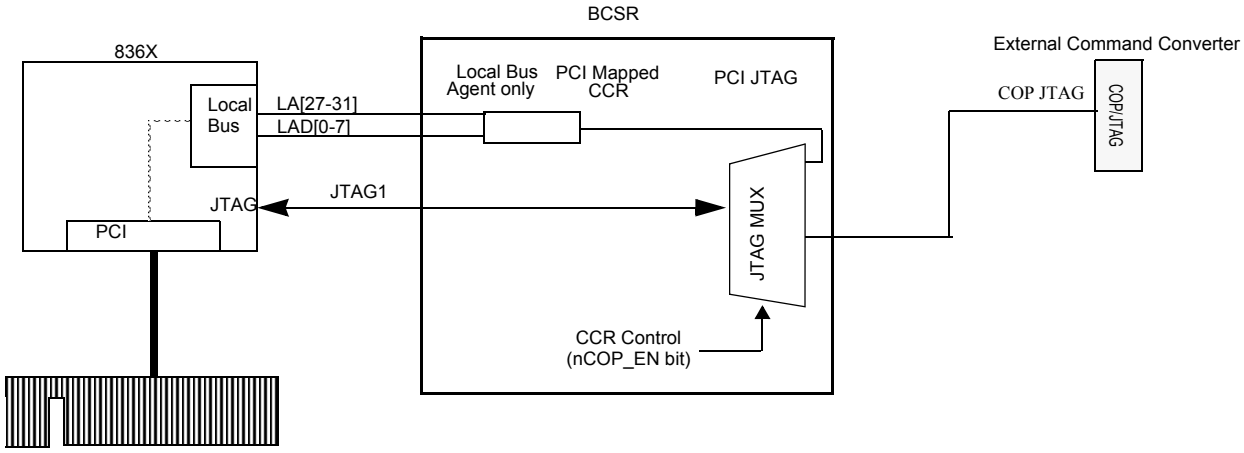


Figure 5-17. JTAG Block diagram

5.12 UART Ports

To assist with development of user's applications and to provide convenient communication channels with both a terminal and a host computer, two RS-232 transceivers are provided on the MPC8360EA MDS Processor Board. These transceivers are connected to the MPC8360EA device via dedicated UART ports (UART1 and UART2). The implementation is done by the HIN211CA part from Intersil, which internally generates the required RS-232 levels from a single 3.3V supply.

The transceivers are enabled by BCSR9[3]. As for the ports, the UART1 port is always enabled, while BCSR9[7] enables/disables the use of the UART2 port. If the UART2 port is not used, these pins can be used by UPC2 for other functions on the PIB. The UART also features hardware flow control. The RS-232 signals are presented on a single 10-pin RJ45 connector, due to the small amount of room available on the PCI panel (used when the MPC8360EA MDS Processor Board is inserted in a PC). A special cable (included with the MPC8360EA MDS Processor Board kit) was prepared to connect between the 10pin RJ45 to two 9pin D-Type female connectors. The connector may be directly connected (via standard serial cable) to any IBM-PC compatible RS-232 port.

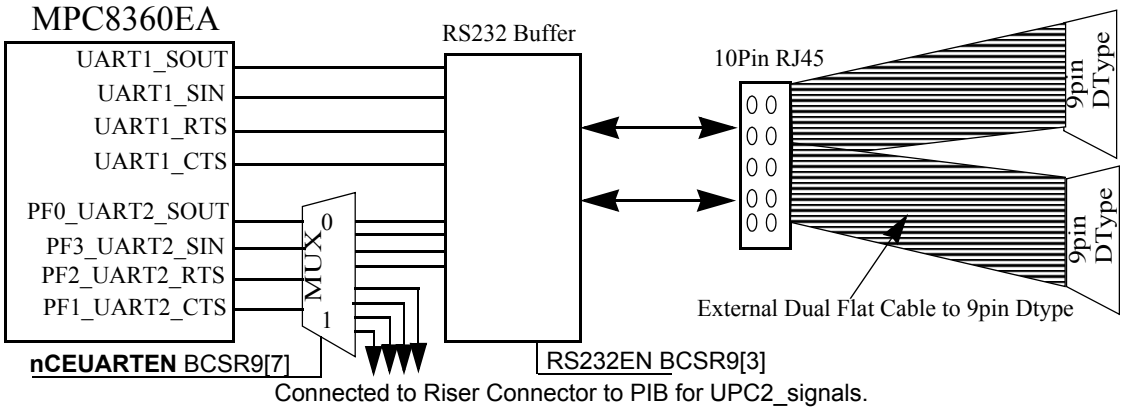


Figure 5-18. RS232 Block Diagram.

The table below show the signals used by UART2 and their alternate UPC2 function. The table below describes the complete use of the signals from the MPC8360EA to the PIB Riser Connectors, and from the Riser connectors to the PMC slots located on the PIB.

Table 5-32. UART2 and UPC2 shared signals

MPC8360 Pin	UPC2 Function (on PIB)	Riser Conn	PMC Conn (on PIB)	RS232 Function
PF3	UPC2-RxDATA[9]	P13-C29	PMC1-J2-54	UART2_SIN
PF0	UPC2-RxDATA[5]	P13-E29	PMC1-J2-9	UART2_SOUT
PF1	UPC2-RxDATA[1]	P13-K20	PMC1-J2-8	UART2_CTS
PF2	UPC2-RxDATA[0]	P13-K22	PMC1-J2-10	UART2_RTS

5.13 I2C (Dual) Port

The MPC8360EA has a dual I2C interfaces (I2C-1 & I2C-2) with multi-master support. Each I2C uses a two-wire interface that contains an SCL (Serial Clock) signal and an SDA (Serial Data) signal for data transfer. All devices that are connected to these two signals must have open-drain or open-collector outputs. A logical AND function is performed on both signals with external pull-up resistors located on the MPC8360EA MDS Processor Board. See Figure 5-19 for an illustration of the I2C connection scheme.

5.13.1 I2C-1

I2C -1 has six devices connected to it.

- The first is the Boot EEPROM (ST EEPROM M24256-B 256Kbit) which provides configuration settings. Its address is ‘0x50’, when CFG_RESET_SOURCE[0-2] = [001] or [010]. The Serial Presence Detect (SPD) function utilized on a dedicated EEPROM in the SODIMM allows retrieval of the configuration data of the integrated DIMM to a program DDR controller.
- The second device is the SPD EEPROM (for SODIMM-1 DDR). It is located at address ‘0x51’.

- The third device is SPD EEPROM (for SODIMM-2 DDR). It is located at address ‘0x52’.
- The fourth device is the Real Time Clock (RTC), implemented by Maxim DS1374. It is located at address ‘0x68’. The DS1374 uses an external 32.768kHz crystal. The DS1374 includes a 32-bit binary counter to continuously count time in seconds. Separate output pins are provided for an interrupt and a square wave at one of four selectable frequencies: 32.768kHz, 8.192kHz, 4.096kHz and 1Hz. This output provides a tick signal for RTC input to the MPC8360EA. The RTC device is fully programmed via serial bus.
- The fifth device is the BCSR, which can be used as the I2C Master or Slave. When the BCSR is a Master, the COP interface controlled via the BCSR, in addition to all I2C -1 bus devices. When the BCSR is a Slave, the COP interface or the MPC8360EA can write/read to/from the BCSR registers.
- The sixth device is the PIB.

5.13.2 I2C-2

I2C -2 has three devices connected to it.

- The first device is the EEPROM Board. This is a serial Atmel EEPROM AT24C01A 128KByte at address ‘0x50’. This device contains all Board history.
- The second device is the BCSR as Master/Slave, the same as on I2C - 1.
- The third device is the PIB.

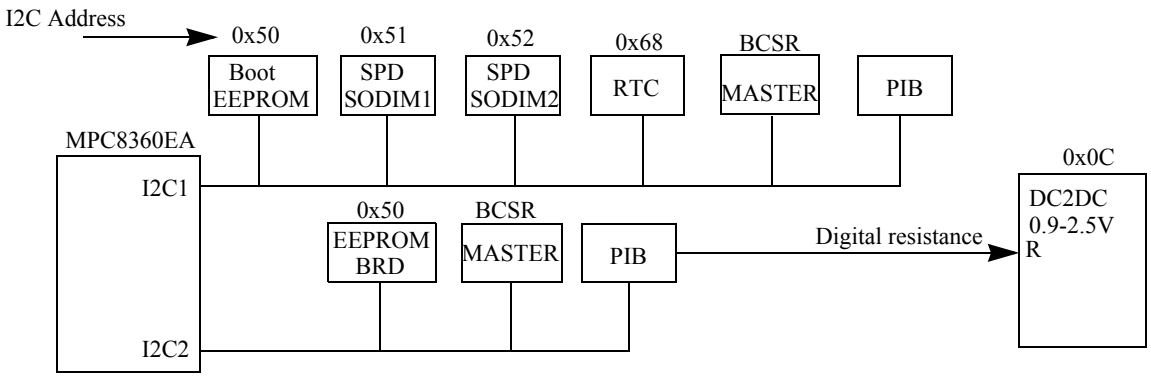


Figure 5-19. Dual I2C Block Scheme

5.14 External Interrupts

There are several external interrupts applied to the MPC8360EA via its interrupt controller:

- ABORT (NMI)
- PIB
- PCI interrupt Host or Agent.
- RTC
- FLASH
- COP CKSTPIn. CKSTPO.

- GETH1 & GETH2

5.14.1 ABORT Interrupt

The ABORT (NMI), is generated by a push-button (see Section 4.4.1 for location). When this button is depressed, the IRQ0 input to the MPC8360EA is asserted. The purpose of this type of interrupt is to support the use of resident debugger if any is made available to the board.

5.14.2 PIB Interrupt

The PIB has 4 interrupts IRQ4, IRQ5, IRQ6, IRQ7. In PCI Host mode they are called IRQW, IRQX, IRQY & IRQZ appropriately. Each PMC module can use all four of the above interrupts.

5.14.3 PCI Interrupt

Each PCI slot on the PIB can generate up to four interrupts, for a total of sixteen (4 slots x 4 interrupts each). Each PCI expansion board can generate an interrupt at any given time. When the MPC8360EA MDS Processor Board is in Agent Mode, only the INTA is used.

5.14.4 RTC Interrupt

The RTC (real time clock) device used is DS1374 from Maxim. It is connected to IRQ5, and can be programmed via the I2C - 1 bus.

5.14.5 FLASH Interrupt

The FLASH memory is connected to IRQ6. Invoking this interrupt indicates that the programming of the flash was done.

5.14.6 JTAG/COP Interrupt

The JTAG/COP Connector uses two interrupts: IRQ6 & IRQ7 for Check Stop In/Out. It is used when working with an external debugger.

5.14.7 GETH Interrupt

GETH1 has IRQ1 as an interrupt, and GETH2 has IRQ2 as an interrupt. Invoking any one of these interrupts indicates that data has been transferred via the specific GETH port.

5.15 Power Supply

The MPC8360EA MDS Processor Board power supply provides all necessary voltages for correct operation of the MPC8360EA device, the DDR, TSEC, Xilinx FPGA, and all on-board peripheral devices.

5.15.1 Primary Power Supply

There are 3 possible sources of power:

- External 5V Power Supply with On/Off Power switch.
- 5V Power Supplied via PCI Edge Connector.
- 5V Power Supplied via PIB Riser Connector.

The External 5V Power Supply is a standard power supply. Its parameters are:

- Vin = 100V - 240V AC
- Nin freq. = 50Hz - 60Hz
- Iin = 2A
- OUTPUT = 5V DC out±5% @ 5A

5.15.2 MPC8360EA MDS Processor Board Power Supply Structure

The MPC8360EA MDS Processor Board supplies power via the following:

- Power Module DC/DC converter DNS04S3R3R06PC made by Delta to produce MPC8360EA 1.2V voltage Core @ 6A.
- Power Module DC/DC converter DNS04S3R3R06PC made by Delta to produce 3.3V @ 6A.
- LDO regulators - LT1764EQ-2.5V, FAN1655 Fairchild Co. Produce DDR VDD, DDR bus required termination and reference voltages (V_{TT} = 1.25V @ 2.1A, V_{REF} = 1.25V)
- Set linear regulator and LDO regulator (MIC49150, LT1764-2.5 from Linear Tech) Provides all necessary TSEC PHY's VDDO, VDDOH, AVDD and DVDDL core voltage (2.5V DC @ 3A, 1V DC @1.5A.)
- Provides necessary visual indication, and power sequence functions.

5.15.3 Power Supply Operation

When a 5V Power source is connected to the board, the MPC8360EA 1.2V core voltage is generated first. The MPC8360EA OVDD should not reach 1V before MPC8360EA VDD reaches 1V. All the above voltages are derived from the 3.3V power supply using LDO regulators and Linear regulators. After 3.3V and 1.2V have been produced, the green LED (Power Good - LD10) indication is illuminated.

When the 5V PIB Power source is selected, a 3V PIB Indication is inserted into the FPGA.

The MPC8360EA LVDD voltage group is used for TSEC I/F. It is required to provide 3.3V or 2.5V to this voltage group, in order to test it in both voltages. Therefore a jumper (JP5) is mounted on board.

Each of the regulators used in the on-board power supply have embedded over-current, voltage, and temperature protection.

The ability to measure the amount of current consumed by the 1.2V is provided by measuring the voltage drop on the fixed series resistors (10mΩ) in the corresponding circuits.

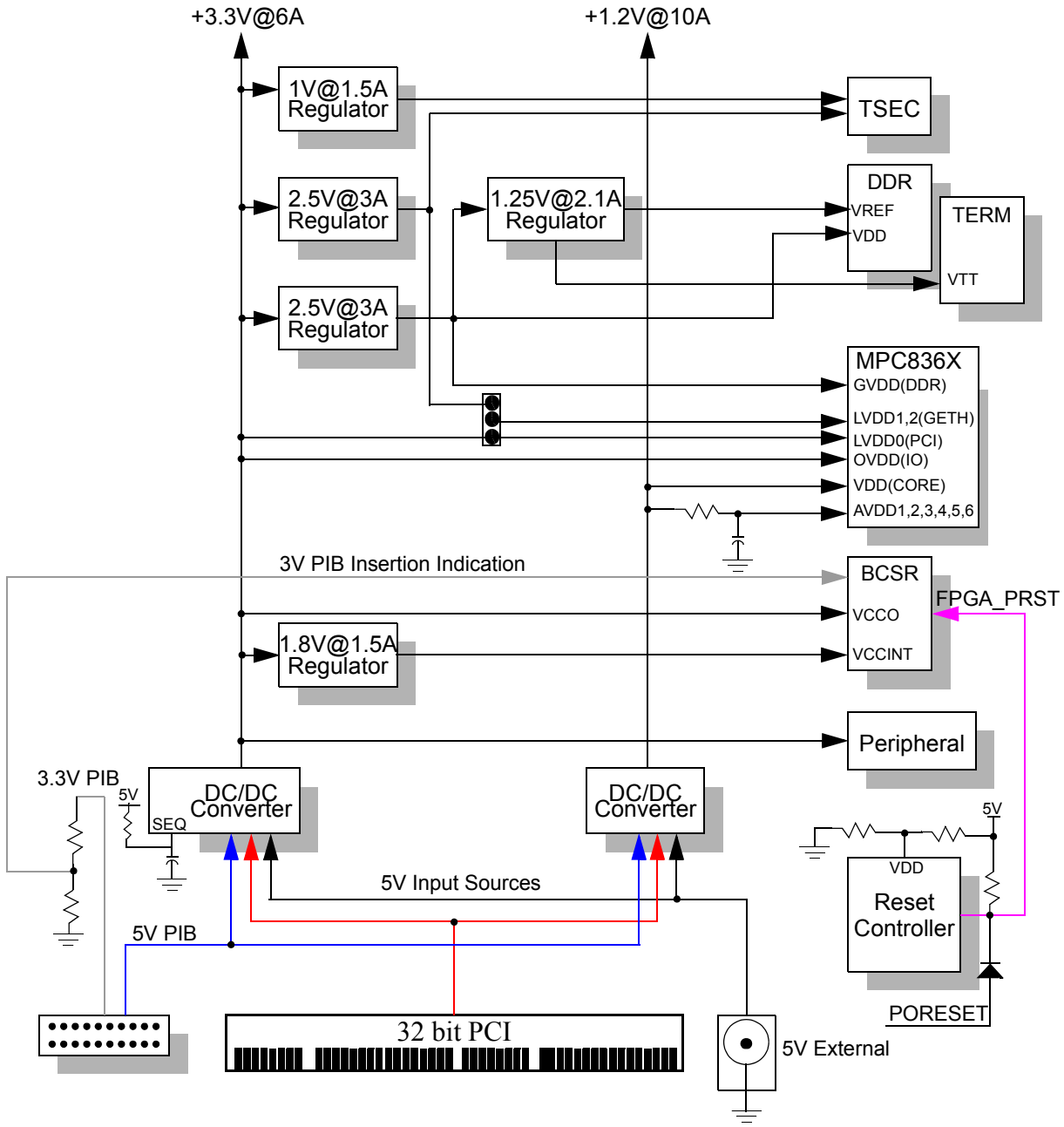


Figure 5-20. Power Distribution on the MPC8360EA MDS Processor Board

Chapter 6
Working with the PIB

6.1 Platform I/O Board Concept

The MPC8360EA MDS Processor Board, together with the MPC8Xxx MDS Processor Board, form the MPC8Xxx Modular Development System (MDS). The MDS enables software programmers to develop software for the 8Xxx architecture. A block diagram of the PIB with a Processor Board is shown in Figure 6-1 on page 6-2.

The PIB provides more capabilities for developing 8Xxx software than the MPC8Xxx Processor Board alone by allowing an MPC8Xxx Processor Board to be configured as a Host, with up to four PCI-compatible boards as Agents, connected to PCI slots (via PMC-PCI adaptors or via the Expansion adaptor) on the PIB motherboard.

The PIB also allows an MPC8Xxx Processor Board to be used in a back plane configuration, and provides room and connections for additional modules such as an ATM, Quad-OC3, or other modules.

Power is provided by the PIB, which also provides additional signal connections via the back plane (if used), and optical GETH connectors on the front plane side of the PIB. The MPC8360EA MDS Processor Board can be connected to a PC in this configuration (via a parallel port connector), without the need for an external command converter.

In summary, the PIB provides the following (list specific for the MPC8360EA MDS Processor Board):

- Support for the MPC8360EA as a PCI Host.
- Support for MPC83xx Agent boards connected to the PCI bus.
- Supports simultaneously operating the PCI bus on PMC1 & ATM on PMC0.
- Supports the operation of ATM on PMC0 for UPC1 and PMC1 for UPC2.
- Provides 6 RMII ports for UCC3:UCC8.
- Allows a view of all the QE signals through PMC0, PMC1

In the block diagram in Figure 6-1 below, note carefully the specific communication lines that are connected with each PMCx slot. For example, PCI1 connects to the PMC1 slot and the PMC2 slot, and PCI2 connects to the PMC2 and PMC3 slot. The specific connections dictate which modules can be connected to which PMC slot.

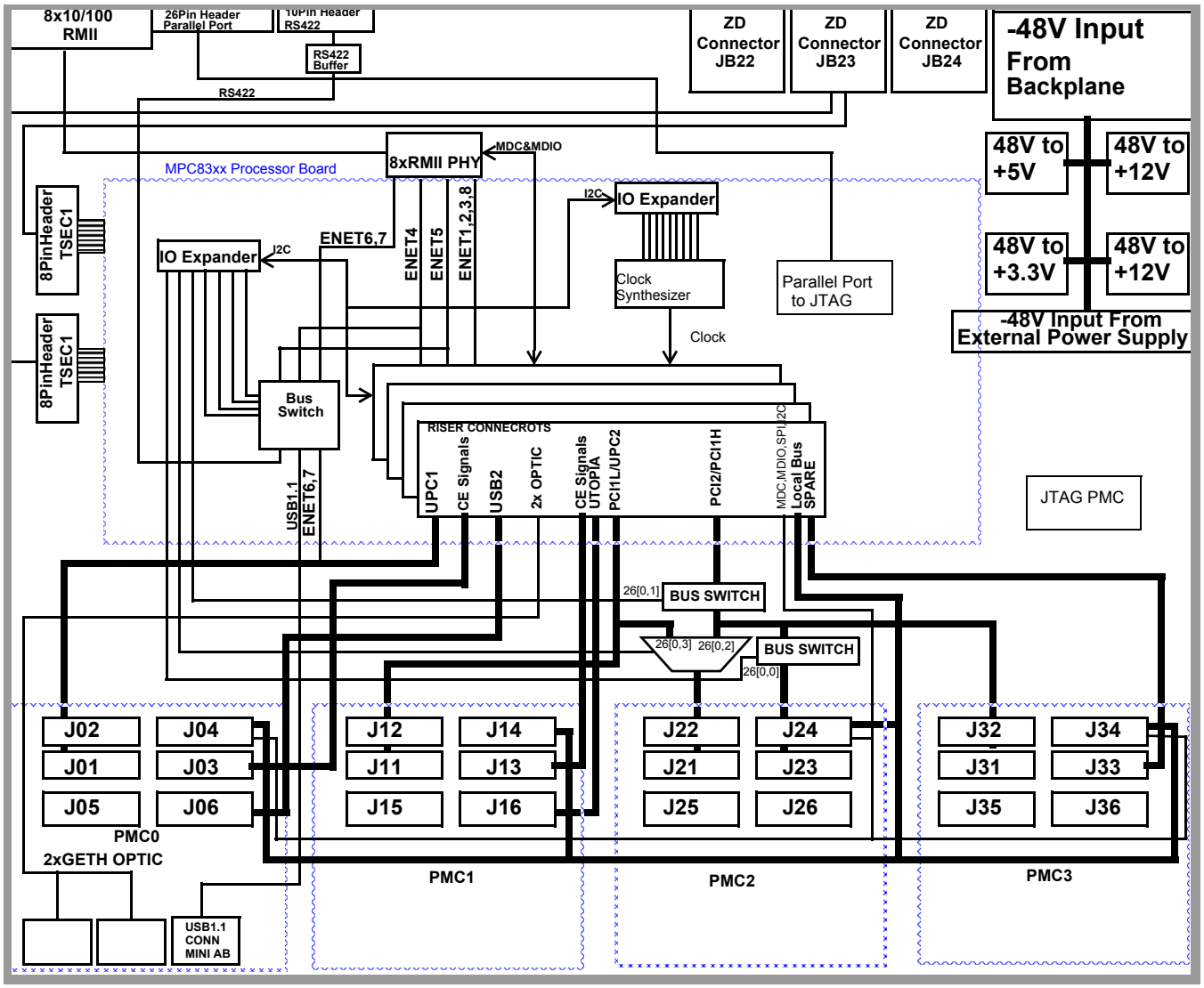


Figure 6-1. PIB Block Diagram, with Processor Board

6.2 MPC8360EA MDS Processor Board as Host

If the edge connector (P3) is ***not*** connected, the MPC8360EA MDS Processor Board configures itself as a Host, which means that its clock is supplied via a clock device on the Processor Board.

In addition, the Host Processor Board configures the PCI communication bus on the PIB, identifies the various agents and modules connected to the PIB, and allocates resources for them. A list of signals between the PIB and the Processor Board is supplied in Section 6.4, below.

It is important to note that an external debugger must be connected to the JTAG/COP connector on the board on which the processor to be debugged is found.

6.3 MPC8360EA MDS Processor Board as Agent

If the MPC8360EA MDS Processor Board is inserted in an adaptor on the PIB, the edge connector (P3) disconnects the clock generated on the agent Processor Board. Its clock is therefore supplied from the host Processor Board via the adaptor connector on the PIB.

A list of signals between the PIB and the Processor Board is supplied in Section 6.4, below.

It is important to note that an external debugger must be connected to the JTAG/COP connector on the board on which the processor to be debugged is found.

6.4 MPC8360EA MDS Processor Board - PIB Signals

The tables below show the correspondence between signals on the MPC8360 and those on the PIB. They are divided by subject (TDM signals, UPC, Ethernet, USB, etc.), and there may be some repeat entries.

- The PMC/XMC pin number is the number of the pin on the PMC slot of the PIB.
- The Riser Connector Pin is the number of the pin on the MPC8360EA MDS Processor Board that connects to the PIB.
- The MPC8360EA Pin name is the name of the pin on the MPC8360EA device. For a complete list of these pins, see the *MPC8360EA User's Manual*.
- The PIB signal name is the name of the signal on the PIB that uses the indicated pin. This is not necessarily the name of the signal on the MPC8360EA device, or on the specific module connected to the PMCx slot on the PIB.
- Function-1 and Function-2 indicate that the specific pin can have two different functions, depending on the configuration of the MPC8360EA device (see first row in Table 6-1, below).

Table 6-1. TDM, UPC, Ethernet Signals

PMC/XMC pin Number	Riser Connector Pin Number	MPC8360 PIN Name	PIB Signal Name	Function-1	Function-2	MPC8360 Clock Number & USB
				UTOPIA +UTOPIA/PCI GETH +8TDM + 4xRMII	**2xRGMII + 4 Devices UTOPIA/POS + 4RMII + UTOPIA/PCI**	
PMC0-J3-40 PMC1-J3-34 PMC2-J4-63 PMC3-J2-52	P11-G7	PA1		MDIO	MDIO	
PMC0-J3-24 PMC1-J3-30 PMC2-J4-58 PMC3-J2-54	P11-G5	PA2		MDC	MDC	
	P13-F18	PA0	PA0 (No GPIO Number)	GEnet1-RXCLK	GEnet1-RXCLK	
PMC0-J3-64	P12-A8	PA3	GPIO0	GEnet1-TXD[0]/TXD[4]	GEnet1-TXD[0]/TXD[4]	
PMC0-J3-54	P12-A10	PA4	GPIO1	GEnet1-TXD[1]/TXD[5]	GEnet1-TXD[1]/TXD[5]	
PMC1-J3-58	P12-A11	PA5	GPIO2	GEnet1-TXD[2]/TXD[6]	GEnet1-TXD[2]/TXD[6]	
PMC1-J3-60	P12-A13	PA6	GPIO3	GEnet1-TXD[3]/TXD[7]	GEnet1-TXD[3]/TXD[7]	
PMC0-J3-46	P12-A14	PA7	GPIO4	GEnet1-TX_EN/TX_ER	GEnet1-TX_EN/TX_ER	

Table 6-1. TDM, UPC, Ethernet Signals (continued)

PMC/XMC pin Number	Riser Connector Pin Number	MPC8360 PIN Name	PIB Signal Name	Function-1	Function-2	MPC8360 Clock Number & USB
PMC0-J3-48	P12-A17	PA9	GPIO6	GEnet1-RXD[0]/RXD[4]	GEnet1-RXD[0]/RXD[4]	
PMC0-J3-52	P12-A19	PA10	GPIO7	GEnet1-RXD[1]/RXD[5]	GEnet1-RXD[1]/RXD[5]	
PMC1-J3-54	P12-A20	PA11	GPIO8	GEnet1-RXD[2]/RXD[6]	GEnet1-RXD[2]/RXD[6]	
PMC0-J3-58	P12-A22	PA12	GPIO9	GEnet1-RXD[3]/RXD[7]	GEnet1-RXD[3]/RXD[7]	
PMC0-J3-60	P12-B14	PA15	GPIO12	GEnet1-RX_DV/RX_ER	GEnet1-RX_DV/_RX_ER	
PMC0-J1-40	P12-C30	PC9	GPO65	GEnet1-GTXCLK-CLKO	GEnet1-GTXCLK-CLKO	CLK10
PMC1-J3-52	P12-C28	PC8	GPIO64	GEnet1-Input-125M	GEnet1-Input-125M	CLK9
PMC0-J1-48	P12-A16	PA8	GPIO5	TDMA-RXD[0]	UPC1-RxCLAV[1]	
PMC0-J2-46	P12-A23	PA13	GPIO10	TDMA-TXD[0]	UPC1-TxCLAV[1]	
PMC0-J2-57	P12-A25	PA14	GPIO11	TDMA-RSYNC	UPC1-RxEN[1]	
PMC0-J2-19	P12-B15	PA16	GPIO13	TDMA-TSYNC	UPC1-TxEN[1]	
PMC0-J3-25	P12-A5	PC7	GPIO63 (GPIO58)	TDMA-RXCLK		CLK3
PMC0-J3-29	P12-A28	PC3	GPIO59	TDMA-TXCLK		CLK4
PMC0-J3-31	P12-C13	PA26	GPIO23	TDME-RSYNC	GEnet2-RXD[3]/RXD[7]	
PMC0-J3-35	P12-B17	PA17	GPIO14	TDME-TSYNC	GEnet2-TXD[0]/TXD[4]	
PMC0-J3-37	P12-D7	PA29	GPIO26	TDME-RXD[0]	GEnet2-RX_DV/_RX_ER	
PMC0-J3-41	P12-D8	PA30	GPIO27	TDME-TXD[0]		
PMC0-J2-58	P12-K26	PC16	GPI72	TDME-TXCLK	UPC1-TxCLAV[3]	CLK17
PMC0-J3-7	P13-A3	PC23	GPI79	TDME-RXCLK	UPC1-RMOD	CLK24
PMC0-J3-49	P12-B18	PA18	GPIO15	TDMD-TSYNC	GEnet2-TXD[1]/TXD[5]	
PMC0-J3-53	P12-B20	PA19	GPIO16	TDMD-TXD[0]	GEnet2-TXD[2]/TXD[6]	
PMC0-J3-55	P12-C10	PA24	GPIO21	TDMD-RSYNC	GEnet2-RXD[1]/RXD[5]	
PMC0-J3-59	P12-C12	PA25	GPIO22	TDMD-RXD[0]	GEnet2-RXD[2]/RXD[6]	
PMC0-J2-34	P12-G1	PC13	GPIO69	TDMD-RXCLK	UPC1-TxEN[3]	CLK14
PMC0-J3-61	P12-A3	PC1	GPIO57	TDMD-TXCLK		CLK2
PMC0-J3-4	P12-B21	PA20	GPIO17	TDMB-TSYNC	GEnet2-TXD[3]/TXD[7]	
PMC0-J3-6	P12-E1	PC10	GPO66	TDMB-TXCLK		CLK11
PMC0-J2-23	P12-B24	PA22	GPIO19	TDMB-TXD[0]	UPC1-RxCLAV[2]	
PMC0-J3-10	P12-A30	PC4	GPIO60	TDMB-RXCLK		CLK5
PMC0-J1-47	P12-C15	PA27	GPIO24	TDMB-RXD[0]	UPC1-TxCLAV[2]	
PMC0-J3-12	P12-C16	PA28	GPIO25	TDMB-RSYNC	UPC2-RxADDR[5]	
PMC0-J2-52	P12-K28	PC17	GPI73	TDMF-RXCLK	UPC1-RxCLAV[3]	CLK18
PMC0-J3-16	P12-B23	PA21	GPIO18	TDMF-TSYNC	GEnet2-TX_EN/TX_ER	
PMC0-J3-18	P12-B26	PA23	GPIO20	TDMF-RSYNC	GEnet2-RXD[0]/RXD[4]	
PMC0-J3-17	P12-J18	PE9	GPIO119	TDMF-RXD[0]	UPC1-STPA	
PMC0-J3-11	P12-J19	PE10	GPIO120	TDMF-TXD[0]	UPC1-RVAL	
PMC0-J3-22	P12-A7	PC18	GPI74	TDMF-TXCLK		CLK19
PMC0-J1-	P13-A11	PC29 (No GPIO Number)	PC29 (No GPIO Number)		GEnet1-TBIRXCLK1	
PMC0-J1-	P13-A10	PC28 (No GPIO Number)	PC28 (No GPIO Number)		GEnet2-TBIRXCLK1	
PMC1-J3-1	P12-D10	PB0	GPIO28	Enet3-TXD[0]/TXD[2]	Enet3-TXD[0]	
PMC1-J3-5	P12-D11	PB1	GPIO29	Enet3-TXD[1]/TXD[3]	Enet3-TXD[1]	

Table 6-1. TDM, UPC, Ethernet Signals (continued)

PMC/XMC pin Number	Riser Connector Pin Number	MPC8360 PIN Name	PIB Signal Name	Function-1	Function-2	MPC8360 Clock Number & USB
PMC1-J3-7	P12-D16	PB4	GPIO32	Enet3-TX_EN	Enet3-TX_EN	
PMC1-J3-11	P12-E9	PB6	GPIO34	Enet3-RXD[0]/RXD[2]	Enet3-RXD[0]	
PMC1-J3-13	P12-E11	PB7	GPIO35	Enet3-RXD[1]/RXD[3]	Enet3-RXD[1]	
PMC1-J3-17	P12-E18	PB12	GPIO40	Enet3-RX_DV/RX_CRS	Enet3-RX_DV	
PMC1-J3-19	P12-E20	PB13	GPIO41	Enet3-RX_ER	Enet3-RX_ER	
PMC1-J3-55	P12-D13	PB2	GPIO30	Enet4-RXD[1]/RXD[3]	Enet4-RXD[1]/RXD[3]	USB_OE
PMC1-J3-25	P12-D14	PB3	GPIO31	Enet4-RX_DV/RX_CRS	Enet4-RX_DV/RX_CRS	USB_TP
PMC1-J3-29	P12-E8	PB5	GPIO33	Enet4-RX_ER	Enet4-RX_ER	
PMC1-J3-31	P12-E12	PB8	GPIO36	Enet4-RXD[0]/RXD[2]	Enet4-RXD[0]/RXD[2]	USB_TN
PMC1-J3-35	P12-E14	PB9	GPIO37	Enet4-TXD[0]/TXD[2]	Enet4-TXD[0]/TXD[2]	USB_RP
PMC1-J3-37	P12-E15	PB10	GPIO38	Enet4-TXD[1]/TXD[3]	Enet4-TXD[1]/TXD[3]	USB_RXD
PMC1-J3-41	P12-E17	PB11	GPIO39	Enet4-TX_EN	Enet4-TX_EN	USB_RN
PMC0-J3-34	P12-H26	PE0	GPIO110	TDMH-TXD[0]	UPC1-RxADDR[5]	Enet7-TXD0
PMC0-J1-41	P12-J10	PE4	GPIO114	TDMH-TSYNC	UPC1-RxADDR[2]	Enet7-TXEN
PMC0-J3-30	P12-J13	PE6	GPIO116	TDMH-RSYNC	UPC1-TxADDR[5]	Enet7-RXD0
PMC0-J1-54	P12-J22	PE12	GPIO122	TDMH-RXD[0]	UPC1-TxADDR[2]	Enet7-RXDVCRS
PMC0-J1-20	P13-C1	PC21	GPIO77	TDMH-RXCLK	*UPC1-RxEN[2]	CLK22
PMC0-J3-1	P13-A1	PC22	GPIO78	TDMH-TXCLK	*UPC1-TMOD	CLK23
PMC0-J1-16	P12-J21	PE11	GPIO121	TDMC-RXD[0]	UPC1-RxADDR[3]	
PMC0-J2-55	P12-J14	PE7	GPIO117	TDMC-TXD[0]	UPC1-RxADDR[4]	Enet7-RXD0
PMC0-J2-51	P12-J24	PE13	GPIO123	TDMC-RSYNC	UPC1-TxADDR[4]	
PMC0-J1-10	P12-H27	PE1	GPIO111	TDMC-TSYNC	UPC1-TxADDR[3]	Enet7-TXD1
PMC0-J3-43	P12-E3	PC11	GPI67	TDMC-RXCLK		CLK12
PMC0-J3-47	P12-A1	PC0	GPIO56	TDMC-TXCLK		CLK1
PMC0-J1-4	P12-H29	PE2	GPIO112	TDMG-RSYNC	UPC1-RxADDR[0]	
PMC0-J1-58	P12-H30	PE3	GPIO113	TDMG-TSYNC	UPC1-TxADDR[0]	
PMC0-J2-13	P12-J11	PE5	GPIO115	TDMG-TXD[0]	UPC1-RxADDR[1]	
PMC0-J1-55	P12-J16	PE8	GPIO118	TDMG-RXD[0]	UPC1-TxADDR[1]	
PMC0-J2-61	P13-C5	PC19	GPIO75	TDMG-RXCLK	*UPC1-RxEN[3]	CLK20
PMC0-J1-17	P13-C3	PC20	GPIO76	TDMG-TXCLK	*UPC1-TxEN[2]	CLK21
PMC0-J1-32	P12-E21	PB14	GPIO42	UPC1-TxSOC	UPC1-TxSOC	
PMC0-J2-29	P12-F8	PB15	GPIO43	UPC1-*TxENB[0]	UPC1-*TxENB[0]	
PMC0-J1-29	P12-F9	PB16	GPIO44	UPC1-TxCLAV[0]	UPC1-TxCLAV[0]	
PMC0-J2-28	P12-F10	PB17	GPIO45	UPC1-TxDATA[15]	UPC1-TxDATA[15]	
PMC0-J1-28	P12-F12	PB18	GPIO46	UPC1-TxDATA[14]	UPC1-TxDATA[14]	
PMC0-J1-27	P12-F13	PB19	GPIO47	UPC1-TxDATA[13]	UPC1-TxDATA[13]	
PMC0-J2-26	P12-F15	PB20	GPIO48	UPC1-TxDATA[12]	UPC1-TxDATA[12]	
PMC0-J1-49	P12-F16	PB21	GPIO49	UPC1-TxDATA[11]	UPC1-TxDATA[11]	
PMC0-J1-23	P12-F18	PB22	GPIO50	UPC1-TxDATA[10]	UPC1-TxDATA[10]	
PMC0-J2-22	P12-F19	PB23	GPIO51	UPC1-TxDATA[9]	UPC1-TxDATA[9]	
PMC0-J1-22	P12-F21	PB24	GPIO52	UPC1-TxDATA[8]	UPC1-TxDATA[8]	
PMC0-J1-21	P12-F22	PB25	GPIO53	UPC1-TxDATA[7]	UPC1-TxDATA[7]	
PMC0-J2-20	P12-F24	PB26	GPIO54	UPC1-TxDATA[6]	UPC1-TxDATA[6]	
PMC0-J2-48	P12-F25	PB27	GPIO55	UPC1-TxDATA[5]	UPC1-TxDATA[5]	

Table 6-1. TDM, UPC, Ethernet Signals (continued)

PMC/XMC pin Number	Riser Connector Pin Number	MPC8360 PIN Name	PIB Signal Name	Function-1	Function-2	MPC8360 Clock Number & USB
PMC0-J1-13	P12-E5	PC12	GPIO68	UPC1-TXCLK	UPC1-TXCLK	CLK13
PMC0-J3-42	P12-G3	PC14	GPIO70	UPC1-RXCLK	UPC1RXCLK	CLK15
PMC0-J2-45	P12-F30	PD0	GPIO82	UPC1-TxDATA[4]	UPC1-TxDATA[4]	
PMC0-J1-52	P12-G7	PD1	GPIO83	UPC1-TxDATA[3]	UPC1-TxDATA[3]	
PMC0-J2-43	P12-G8	PD2	GPIO84	UPC1-TxDATA[2]	UPC1-TxDATA[2]	
PMC0-J2-32	P12-G10	PD3	GPIO85	UPC1-TxDATA[1]	UPC1-TxDATA[1]	
PMC0-J1-26	P12-G11	PD4	GPIO86	UPC1-TxDATA[0]	UPC1-TxDATA[0]	
PMC0-J1-43	P12-G13	PD5	GPIO87	UPC1-RxSOC	UPC1-RxSOC	
PMC0-J1-33	P12-G14	PD6	GPIO88	*UPC1-RxEN[0]	*UPC1-RxEN[0]	
PMC0-J2-35	P12-G16	PD7	GPIO89	UPC1-RxCLAV[0]	UPC1-RxCLAV[0]	
PMC0-J1-36	P12-G17	PD8	GPIO90	UPC1-RxDATA[15]	UPC1-RxDATA[15]	
PMC0-J2-38	P12-G19	PD9	GPIO91	UPC1-RxDATA[14]	UPC1-RxDATA[14]	
PMC0-J1-37	P12-G20	PD10	GPIO92	UPC1-RxDATA[13]	UPC1-RxDATA[13]	
PMC0-J2-25 PMC0-J4-5	P12-G22	PD11	GPIO93	UPC1-RxDATA[12]	UPC1-RxDATA[12]	
PMC0-J2-42	P12-G23	PD12	GPIO94	UPC1-RxDATA[11]	UPC1-RxDATA[11]	
PMC0-J2-39	P12-G25	PD13	GPIO95	UPC1-RxDATA[10]	UPC1-RxDATA[10]	
PMC0-J2-54	P12-G26	PD14	GPIO96	UPC1-RxDATA[9]	UPC1-RxDATA[9]	RMII6-TXD0
PMC0-J1-61	P12-G28	PD15	GPIO97	UPC1-RxDATA[8]	UPC1-RxDATA[8]	RMII6-TXD1
PMC0-J1-53	P12-H10	PD18	GPIO100	UPC1-RxDATA[7]	UPC1-RxDATA[7]	RMII6-TXEN
PMC0-J1-46	P12-H12	PD19	GPIO101	UPC1-RxPRTY	UPC1-RxPRTY	
PMC0-J2-49	P12-H14	PD20	GPIO102	UPC1-RxDATA[6]	UPC1-RxDATA[6]	RMII6-RXD0
PMC0-J2-9	P12-H15	PD21	GPIO103	UPC1-RxDATA[5]	UPC1-RxDATA[5]	RMII6-RXD1
PMC0-J2-31	P12-H17	PD22	GPIO104	UPC1-TxPRTY	UPC1-TxPRTY	
PMC0-J1-60	P12-H18	PD23	GPIO105	UPC1-RxDATA[4]	UPC1-RxDATA[4]	
PMC0-J1-59	P12-H20	PD24	GPIO106	UPC1-RxDATA[3]	UPC1-RxDATA[3]	
PMC0-J2-49	P12-H21	PD25	GPIO107	UPC1-RxDATA[2]	UPC1-RxDATA[2]	
PMC0-J2-8	P12-H23	PD26	GPIO108	UPC1-RxDATA[1]	UPC1-RxDATA[1]	RMII6-TXDVCRS
PMC0-J2-10	P12-H24	PD27	GPIO109	UPC1-RxDATA[0]	UPC1-RxDATA[0]	
PMC0-J3-5	P12-F27	PC24	GPIO80	UPC1-REOP	UPC1-REOP	
PMC0-J2-23	P12-F28	PC25	GPIO81	UPC1-TEOP	UPC1-TEOP	
PMC0-J3-19	P12-G29	PD16	GPIO98	UPC1-TERR	UPC1-TERR	
PMC0-J3-13	P12-H9	PD17	GPIO99	UPC1-RERR	UPC1-RERR	
PMC1-J3-47	**P12-G5**	**PC15**	**GPO71**	**RMII CLK-UCC(3,5,7) It can be used for UCC (3,4,5,6,7,8)**		**CLK16**
PMC1-J3-43	P12-C3	PC6	GPO62	**RMII CLK-UCC(4,6,8)**		CLK7
PMC1-J3-61	P12-K11	PE16	GPO126	Enet5-RXD[1]/RXD[3]	Enet5-RXD[1]/RXD[3]	
PMC1-J3-4	P12-K12	PE17	GPO127	Enet5-RX_DV/RX_CRS	Enet5-RX_DV/RX_CRS	
PMC1-J3-49	P12-K16	PE19	GPO129	Enet5-RX_ER	Enet5-RX_ER	
PMC1-J3-59	P12-K20	PE22	GPO132	Enet5-RXD[0]/RXD[2]	Enet5-RXD[0]/RXD[2]	
PMC1-J3-6	P12-K22	PE23	GPO133	Enet5-TXD[0]/TXD[2]	Enet5-TXD[0]/TXD[2]	
PMC1-J3-53	P12-K23	PE24	GPO134	Enet5-TXD[1]/TXD[3]	Enet5-TXD[1]/TXD[3]	
PMC1-J3-55	P12-K25	PE25	GPO135	Enet5-TX_EN	Enet5-TX_EN	
PMC1-J3-10	P12-J26	PE26	GPO136	Enet8-RX_DV/RX_CRS	Enet8-RX_DV/RX_CRS	

Table 6-1. TDM, UPC, Ethernet Signals (continued)

PMC/XMC pin Number	Riser Connector Pin Number	MPC8360 PIN Name	PIB Signal Name	Function-1	Function-2	MPC8360 Clock Number & USB
PMC1-J3-12	P12-E28	PE27	GPO137	Enet8-RX_ER	Enet8-RX_ER	
PMC1-J3-16	P12-J25	PE14	GPO124	Enet8-TXD[0]/TXD[2]	Enet8-TXD[0]/TXD[2]	
PMC1-J3-18	P12-K10	PE15	GPO125	Enet8-TXD[1]/TXD[3]	Enet8-TXD[1]/TXD[3]	
PMC1-J3-22	P12-K14	PE18	GPO128	Enet8-TX_EN	Enet8-TX_EN	
PMC1-J3-24	P12-K17	PE20	GPO130	Enet8-RXD[0]/RXD[2]	Enet8-RXD[0]/RXD[2]	
PMC1-J3-28	P12-K19	PE21	GPO131	Enet8-RXD[1]/RXD[3]	Enet8-RXD[1]/RXD[3]	
XMC0-J6-E13	P13-F15	PA31(no GPIO Number)	PA31(no GPIO Num-ber)			

Table 6-2. GMII1/RGMII1/TBI1/RTBI1 &GMII2/RGMII2/TBI2/RTBI2

MPC8360 Pin	Riser Conn	PIB GPIO Pin	MPC/XMC	MII/GMII[1]	MII/GMII[2]	action for GETH/PIB
PA0	**P13-F18**	**PA0 (No GPIO Num)**	**XMC0-J06-E15**	**GEnet1-RXCLK**		always used for GETH
PA3	P12-A8	GPIO0	PMC0-J3-64	GEnet1-TXD[0]		For RGMII/RTBI function BCSR9[0] should be 0. For Selecting GPIO function for PIB BCSR9[0] should be 1. Sig G1NARROW = 0
PA4	P12-A10	GPIO1	PMC0-J3-54	GEnet1-TXD[1]		
PA5	P12-A11	GPIO2	PMC1-J3-58	GEnet1-TXD[2]		
PA6	P12-A13	GPIO3	PMC1-J3-60	GEnet1-TXD[3]		
PA7	P12-A14	GPIO4	PMC0-J3-46	GEnet1-TX_EN		
PA9	P12-A17	GPIO6	PMC0-J3-48	GEnet1-RXD[0]		
PA10	P12-A19	GPIO7	PMC0-J3-52	GEnet1-RXD[1]		
PA11	P12-A20	GPIO8	PMC1-J3-54	GEnet1-RXD[2]		
PA12	P12-A22	GPIO9	PMC0-J3-58	GEnet1-RXD[3]		
PA15	P12-B14	GPIO12	PMC0-J3-60	GEnet1-RX_DV		
PC9	**P12-C30**	**GPIO65**	**PMC0-J1-40**	**GEnet1-GTX_CLK**		always used for GETH
PC8	**P12-C28**	**GPIO64**	**PMC1-J3-52**	**GEnet1-Input-125M**		always used for GETH
PC10	P12-E1	GPO66	PMC0-J3-6	Enet1-TXCLK(MII)		
PA8	P12-A16	GPIO5	PMC0-J1-48	GEnet1-TX_ER		For GMII/TBI function: BCSR9[0] should be 0. & BCSR8[0,1] should be [11] or [10] For Selecting GPIO function for PIB: BCSR9[0] should be 1 & BCSR8[0,1] should be [00] or [01 Sig nMII1EN = 0]
PA14	P12-A25	GPIO11	PMC0-J2-57	GEnet1-CRS		
PA16	P12-B15	GPIO13	PMC0-J2-19	GEnet1-RX_ER		

Table 6-2. GMII1/RGMII1/TBI1/RTBI1 &GMII2/RGMII2/TBI2/RTBI2 (continued)

MPC8360 Pin	Riser Conn	PIB GPIO Pin	MPC/XMC	MII/GMII[1]	MII/GMII[2]	action for GETH/PIB
PA13	P12-A23	GPIO10	PMC0-J2-46	GEnet1-RXD4/COL		For GMII/TBI function: BCSR9[0] should be 0. & BCSR8[0,1] should be [11] or [10] For Selecting GPIO function for PIB: BCSR9[0] should be 1 & BCSR8[0,1] should be [00] or [01] Sig G1WIDE = 0
PB0	P12-D10	GPO28	PMC1-J3-1	Enet1-RXD[6]		
PB1	P12-D11	GPO29	PMC1-J3-5	Enet1-RXD[5]		
PB4	P12-D16	GPO32	PMC1-J3-7	Enet1-RXD[7]		
PB6	P12-E9	GPO34	PMC1-J3-11	Enet1-TXD[4]		
PB7	P12-E11	GPO35	PMC1-J3-13	Enet1-TXD[5]		
PB9	P12-E14	GPO37	PMC1-J3-35	Enet1-TXD[6]		
PB10	P12-E15	GPO38	PMC1-J3-37	Enet1-TXD[7]		
PC3	P12-A28	GPIO59	PMC0-J3-29		**GEnet2-Input-125M**	always used for GETH
PA26	P12-C13	GPIO23	PMC0-J3-31		GEnet2-RXD[3]	For RGMII/RTBI function BCSR9[1] should be 0. For Selecting GPIO function for PIB BCSR9[1] should be 1. Sig G2NARROW = 0
PA17	P12-B17	GPIO14	PMC0-J3-35		GEnet2-TXD[0]	
PA29	P12-D7	GPIO26	PMC0-J3-37		GEnet2-RX_DV	
PA18	P12-B18	GPIO15	PMC0-J3-49		GEnet2-TXD[1]	
PA19	P12-B20	GPIO16	PMC0-J3-53		GEnet2-TXD[2]	
PA24	P12-C10	GPIO21	PMC0-J3-55		GEnet2-RXD[1]	
PA25	P12-C12	GPIO22	PMC0-J3-59		GEnet2-RXD[2]	
PA20	P12-B21	GPIO17	PMC0-J3-4		GEnet2-TXD[3]	
PA21	P12-B23	GPIO18	PMC0-J3-16		GEnet2-TX_EN	
PA23	P12-B26	GPIO20	PMC0-J3-18		GEnet2-RXD[0]	
PC28 (No GPIO Number)	**P13-A10**	**PC28 (No GPIO Number)**	**PMC0-J1-**		**GEnet2-TBIRXCLK1**	always used for GETH
PA31	**P13-F15**	**PA31(no GPIO Number)**	**XMC0-J06-E13**		**GEnet2-RXCLK**	always used for GETH
PC2	**P12-A13**	**GPIO58**	**XMC0-J06-E5**		**GEnet2-GTXCLK**	always used for GETH
PC16	P12-K26	GPI72	PMC0-J2-58		Enet2-TXCLK(MII)	Sig nMII2EN = 0
PA22	P12-B24	GPIO19	PMC0-J2-23		Enet2-TX_ER	For GMII/TBI function BCSR9[0] should be 0. & BCSR8[2,3] should be [11] or [10] For Selecting GPIO function for PIB BCSR9[1] should be 1 & BCSR8[2,3] should be [00] or [01] Sig nMII2EN = 0
PA30	P12-D8	GPIO27	PMC0-J3-41		GEnet2-RX_ER	
PA28	P12-C16	GPIO25	PMC0-J3-12		Enet2-CRS	

Table 6-2. GMII1/RGMII1/TBI1/RTBI1 &GMII2/RGMII2/TBI2/RTBI2 (continued)

MPC8360 Pin	Riser Conn	PIB GPIO Pin	MPC/XMC	MII/GMII[1]	MII/GMII[2]	action for GETH/PIB
PA27	P12-C15	GPIO24	PMC0-J1-47		Enet2-RXD4/COL	For GMII/TBI function BCSR9[0] should be 0. & BCSR8[2,3] should be [11] or [10] For Selecting GPIO function for PIB BCSR9[1] should be 1 & BCSR8[2,3] should be [00] or [01] Sig G2WIDE = 0
PB12	P12-E18	GPO40	PMC1-J3-17		Enet2-RXD[5]	
PB13	P12-E20	GPO41	PMC1-J3-19		Enet2-RXD[6]	
PB2	P12-D13	GPO30	PMC1-J3-55		Enet2-TXD[4]	
PB3	P12-D14	GPO31	PMC1-J3-25		Enet2-TXD[5]	
PB5	P12-E8	GPO33	PMC1-J3-29		Enet2-TXD[6]	
PB8	P12-E12	GPO36	PMC1-J3-31		Enet2-TXD[7]	
PB11	P12-E17	GPO39	PMC1-J3-41		Enet2-RXD[7]	

Table 6-3. PCI/UTOPIA Bus Signals

PIB-PMC	PB/PIB RISER CONN	8360 Pin	PCI Function	Utopia Function on PMC1 Only
PMC1-J3-42	P12-C1	PC5	GPIO61	UPC2_RXCLK/CLK_IN
PMC1,2,3-J4-58	P13-A26	PF5	INTA_B	UPC2-RxADDR[0]
PMC1,2,3-J2-13	P13-A16	PF6	PCI_RESET_OUT_B	UPC2_RxADDR[1]
PMC1,2,3-J1-61	P13-B26	PG0	PCI_AD0	UPC2_RxDATA[8]
PMC1,2,3-J1-60	P13-A22	PG1	PCI_AD1	UPC2_RxDATA[4]
PMC1,2,3-J1-59	P13-K19	PG2	PCI_AD2	UPC2_RxDATA[3]
PMC1,2,3-J1-58	P13-J22	PG3	PCI_AD3	UPC2_TxADDR[0]
PMC1,2,3-J1-55	P13-H27	PG4	PCI_AD4	UPC2_TxADDR[1]
PMC1,2,3-J1-54	P13-G25	PG5	PCI_AD5	UPC2_TxADDR[2]
PMC1,2,3-J1-53	P13-F27	PG6	PCI_AD6	UPC2_RxDATA[7]
PMC1,2,3-J2-51	P13-E24	PG7	PCI_AD7	UPC2_TxADDR[4]
PMC1,2,3-J2-49	P13-D22	PG8	PCI_AD8	UPC2_RxDATA[2]
PMC1,2,3-J1-49	P13-C24	PG9	PCI_AD9	UPC2_TxDATA[11]
PMC1,2,3-J2-48	P13-B24	PG10	PCI_AD10	UPC2_TxDATA[5]
PMC1,2,3-J1-48	P13-A20	PG11	PCI_AD11	UPC2_RxCLAV[1]
PMC1,2,3-J1-47	P13-K17	PG12	PCI_AD12	UPC2_TxCLAV[2]
PMC1,2,3-J2-46	P13-J21	PG13	PCI_AD13	UPC2_TxCLAV[1]
PMC1,2,3-J2-45	P13-H26	PG14	PCI_AD14	UPC2_TxDATA[4]
PMC1,2,3-J1-46	P13-G22	PG15	PCI_AD15	UPC2_RxPRTY
PMC1,2,3-J2-31	P13-F25	PG16	PCI_AD16	UPC2_TxPRTY

Table 6-3. PCI/UTOPIA Bus Signals (continued)

PIB-PMC	PB/PIB RISER CONN	8360 Pin	PCI Function	Utopia Function on PMC1 Only
PMC1,2,3-J1-32	P13-E23	PG17	PCI_AD17	UPC2_TxSOC
PMC1,2,3-J2-29	P13-D20	PG18	PCI_AD18	UPC2_TxEN[0]
PMC1,2,3-J1-29	P13-C22	PG19	PCI_AD19	UPC2_TxCLAV[0]
PMC1,2,3-J2-28	P13-B23	PG20	PCI_AD20	UPC2_TxDATA[15]
PMC1,2,3-J1-28	P13-A19	PG21	PCI_AD21	UPC2_TxDATA[14]
PMC1,2,3-J1-27	P13-K16	PG22	PCI_AD22	UPC2_TxDATA[13]
PMC1,2,3-J2-26	P13-J19	PG23	PCI_AD23	UPC2_TxDATA[12]
PMC1,2,3-J2-23	P13-H24	PG24	PCI_AD24	UPC2_RxCLAV[2]
PMC1,2,3-J1-23	P13-G23	PG25	PCI_AD25	UPC2_TxDATA[10]
PMC1,2,3-J2-22	P13-F24	PG26	PCI_AD26	UPC2_TxDATA[9]
PMC1,2,3-J1-22	P13-E21	PG27	PCI_AD27	UPC2_TxDATA[8]
PMC1,2,3-J1-21	P13-D19	PG28	PCI_AD28	UPC2_TxDATA[7]
PMC1,2,3-J2-20	P13-C18	PG29	PCI_AD29	UPC2_TxDATA[6]
PMC1,2,3-J2-19	P13-B21	PG30	PCI_AD30	UPC2_TxEN_B[1]
PMC1,2,3-J1-20	P13-A17	PG31	PCI_AD31	UPC2_RxEN_B[2]
PMC1,2,3-J1-52	P13-D23	PF7	PCI_C_BE_B0	UPC2-TxDATA[3]
PMC1,2,3-J2-43	P13-C26	PF8	PCI_C_BE_B1	UPC2-TxDATA[2]
PMC1,2,3-J2-32	P13-D25	PF9	PCI_C_BE_B2	UPC2-TxDATA[1]
PMC1,2,3-J1-26	P13-A23	PF10	PCI_C_BE_B3	UPC2-TxDATA[0]
PMC1,2,3-J1-43	P13-E26	PF11	PCI_PAR	UPC2-RxSOC
PMC1,2,3-J1-33	P13-F28	PF12	PCI_FRAME_B	UPC2-RxEN[0]
PMC1,2,3-J2-35	P13-G26	PF13	PCI_TRDY_B	UPC2-RxCLAV[0]
PMC1,2,3-J1-36	P13-H29	PF14	PCI_IRDY_B	UPC2-RxDATA[15]
PMC1,2,3-J2-38	P13-J24	PF15	PCI_STOP_B	UPC2-RxDATA[14]
PMC1,2,3-J1-37	P13-K23	PF16	PCI_DEVSEL_B	UPC2-RxDATA[13]
PMC1,2,3-J4-5	P13-J18	PF17	PCI_IDSEL	UPC2-RxDATA[12]
PMC1,2,3-J2-42	P13-C25	PF18	PCI_SERR_B	UPC2-RxDATA[11]
PMC1,2,3-J2-39	P13-E30	PF19	PCI_PERR_B	UPC2-RxDATA[10]
PMC1,2,3-J1-17	P13-G28	PF20	PCI_REQ_B[0]	UPC2_TxEN_B[2]
PMC1,2,3-J1-41	P13-F30	PF21	PCI_REQ_B[1]	UPC2_RxADDR[2]
PMC1,2,3-J1-10	P13-E27	PF22	PCI_REQ_B2	UPC2_TxADDR[3]
PMC1,2,3-J1-16	P13-J25	PF23	PCI_GNT_B[0]	UPC2_RxADDR[3]
PMC1,2,3-J2-55	P13-A25	PF24	PCI_GNT_B[1]	UPC2_RxADDR[4]

Table 6-3. PCI/UTOPIA Bus Signals (continued)

PIB-PMC	PB/PIB RISER CONN	8360 Pin	PCI Function	Utopia Function on PMC1 Only
PMC1,2,3-J2-57	P13-K25	PF25	PCI_GNT_B[2]	UPC2_RxEN_B[1]
PMC2-J1-13	P11-A28	PF26	PCI_CLK[0]	Not Used UPC2_RxEN_B[3]
PMC0-J3-36	P13-J13	PF27	PCI_CLK[1] Master PCI CLK for Host	
PMC1-J1-13	P13-A28	PF28	PCI_CLK[2]	UPC2-CLKO/TXCLK
PMC1-J2-47	P12-B12	PF4	M66EN	UPC2_RxDATA[6]
PMC1-J2-54	P13-C29	PF3	UART2_SIN	UPC2-RxDATA[9]
PMC1-J2-9	P13-E29	PF0	UART2_SOUT	UPC2-RxDATA[5]
PMC1-J2-8	P13-K20	PF1	UART2_CTS	UPC2-RxDATA[1]
PMC1-J2-10	P13-K22	PF2	UART2_RTS	UPC2-RxDATA[0]
PMC0-J3-36	P13-J13	Clock_B uff	XPCI_CLK1	
PMC2-J1-13	P13-A28	Clock_B uff	XPCI_CLK0	

Table 6-4. USB Signals

MPC8360 Pin Name	Riser Conn	Signals name PIB	PMC on PIB	Function2 RGMII	Function1 RMII	Function3
PB2	P12-D13	GPIO30	PMC1-J3-55	Enet2-TXD[4]	Enet4-RXD[1]/RXD[3]	USB_OE
PB3	P12-D14	GPIO31	PMC1-J3-25	Enet2-TXD[5]	Enet4-RX_DV/RX_CRS	USB_TP
PB8	P12-E12	GPIO36	PMC1-J3-31	Enet2-TXD[7]	Enet4-RXD[0]/RXD[2]	USB_TN
PB9	P12-E14	GPIO37	PMC1-J3-35	Enet1-TXD[6]	Enet4-TXD[0]/TXD[2]	USB_RP
PB10	P12-E15	GPIO38	PMC1-J3-37	Enet1-TXD[7]	Enet4-TXD[1]/TXD[3]	USB_RXD
PB11	P12-E17	GPIO39	PMC1-J3-41	Enet2-RXD[7]	Enet4-TX_EN	USB_RN
PC20	P12-C3	GPIO76	PMC0-J1-17	TDMG_TXCLK	XUPC1_TXEN2	USB-CLK CLK21 BRG9

Table 6-5. GETH Signals if RMII traverses the PIB

PMC-Conn	Riser Conn	MPC8360 Pin	PIB Text Name	Pin Function1	Pin Function2	Pin Function3
PMC1-J3-47	**P12-G5**	**PC15**	**GPO71**	**RMII CLK-UCC(3,5,7) It can be used for UCC (3,4,5,6,7,8)**	**CLK16**	
PMC1-J3-43	**P12-C3**	**PC6**	**GPO62**	**RMII CLK-UCC(4,6,8)**	**CLK7**	
PMC1-J3-61	P12-K11	PE16	GPO126	Enet5-RXD[1]/RXD[3]	Enet5-RXD[1]/RXD[3]	
PMC1-J3-4	P12-K12	PE17	GPO127	Enet5-RX_DV/RX_CRS	Enet5-RX_DV/RX_CRS	
PMC1-J3-49	P12-K16	PE19	GPO129	Enet5-RX_ER	Enet5-RX_ER	
PMC1-J3-59	P12-K20	PE22	GPO132	Enet5-RXD[0]/RXD[2]	Enet5-RXD[0]/RXD[2]	

Table 6-5. GETH Signals if RMII traverses the PIB (continued)

PMC1-J3-6	P12-K22	PE23	GPO133	Enet5-TXD[0]/TXD[2]	Enet5-TXD[0]/TXD[2]	
PMC1-J3-53	P12-K23	PE24	GPO134	Enet5-TXD[1]/TXD[3]	Enet5-TXD[1]/TXD[3]	
PMC1-J3-55	P12-K25	PE25	GPO135	Enet5-TX_EN	Enet5-TX_EN	
PMC1-J3-10	P12-J26	PE26	GPO136	Enet8-RX_DV/RX_CRS	Enet8-RX_DV/RX_CRS	
PMC1-J3-12	P12-E28	PE27	GPO137	Enet8-RX_ER	Enet8-RX_ER	
PMC1-J3-16	P12-J25	PE14	GPO124	Enet8-TXD[0]/TXD[2]	Enet8-TXD[0]/TXD[2]	
PMC1-J3-18	P12-K10	PE15	GPO125	Enet8-TXD[1]/TXD[3]	Enet8-TXD[1]/TXD[3]	
PMC1-J3-22	P12-K14	PE18	GPO128	Enet8-TX_EN	Enet8-TX_EN	
PMC1-J3-24	P12-K17	PE20	GPO130	Enet8-RXD[0]/RXD[2]	Enet8-RXD[0]/RXD[2]	
PMC1-J3-28	P12-K19	PE21	GPO131	Enet8-RXD[1]/RXD[3]	Enet8-RXD[1]/RXD[3]	
PMC1-J3-1	P12-D10	PB0	GPIO28	Enet3-TXD[0]/TXD[2]	Enet3-TXD[0]	
PMC1-J3-5	P12-D11	PB1	GPIO29	Enet3-TXD[1]/TXD[3]	Enet3-TXD[1]	
PMC1-J3-7	P12-D16	PB4	GPIO32	Enet3-TX_EN	Enet3-TX_EN	
PMC1-J3-11	P12-E9	PB6	GPIO34	Enet3-RXD[0]/RXD[2]	Enet3-RXD[0]	
PMC1-J3-13	P12-E11	PB7	GPIO35	Enet3-RXD[1]/RXD[3]	Enet3-RXD[1]	
PMC1-J3-17	P12-E18	PB12	GPIO40	Enet3-RX_DV/RX_CRS	Enet3-RX_DV	
PMC1-J3-19	P12-E20	PB13	GPIO41	Enet3-RX_ER	Enet3-RX_ER	
PMC1-J3-55	P12-D13	PB2	GPIO30	Enet4-RXD[1]/RXD[3]	USB_OE	Enet4 does not operate with board MPC8360PB Rev-Proto1
PMC1-J3-25	P12-D14	PB3	GPIO31	Enet4-RX_DV/RX_CRS	USB_TP	
PMC1-J3-29	P12-E8	PB5	GPIO33	Enet4-RX_ER		
PMC1-J3-31	P12-E12	PB8	GPIO36	Enet4-RXD[0]/RXD[2]	USB_TN	
PMC1-J3-35	P12-E14	PB9	GPIO37	Enet4-TXD[0]/TXD[2]	USB_RP	
PMC1-J3-37	P12-E15	PB10	GPIO38	Enet4-TXD[1]/TXD[3]	USB_RXD	
PMC1-J3-41	P12-E17	PB11	GPIO39	Enet4-TX_EN	USB_RN	
PMC0-J3-34	P12-H26	PE0	GPIO110	Enet7-TXD0	UPC1-RxADDR[5]	TDMH-TXD[0]
PMC0-J1-41	P12-J10	PE4	GPIO114	Enet7-TXEN	UPC1-RxADDR[2]	TDMH-TSYNC
PMC0-J3-30	P12-J13	PE6	GPIO116	Enet7-RXD0	UPC1-TxADDR[5]	TDMH-RSYNC
PMC0-J2-55	P12-J14	PE7	GPIO117	Enet7-RXD1	UPC1-RxADDR[4]	TDMC-TXD[0]
PMC0-J1-10	P12-H27	PE1	GPIO111	Enet7-TXD1	UPC1-TxADDR[3]	TDMC-TSYNC
PMC0-J1-54	P12-J22	PE12	GPIO122	Enet7I-RXDVCRS	UPC1-TxADDR[2]	TDMH-RXD[0]
PMC0-J2-54	P12-G26	PD14	GPIO96	Enet6-TXD0	UPC1-RxDATA[9]	
PMC0-J1-61	P12-G28	PD15	GPIO97	Enet6-TXD1	UPC1-RxDATA[8]	
PMC0-J1-53	P12-H10	PD18	GPIO100	Enet6-TXEN	UPC1-RxDATA[7]	
PMC0-J2-49	P12-H14	PD20	GPIO102	Enet6-RXD0	UPC1-RxDATA[6]	
PMC0-J2-9	P12-H15	PD21	GPIO103	Enet6-RXD1	UPC1-RxDATA[5]	
PMC0-J2-8	P12-H23	PD26	GPIO108	Enet6-TXDVCRS	UPC1-RxDATA[1]	

There are ethernet connections on the PIB itself. The table below shows how to configure the host Processor Board for using the PIB ethernet connections as opposed to its own ethernet connections.

Table 6-6. Selecting GMII/RGMII/TBI/RTBI or PIB

BCSR8[0,1]	BCSR9[0]	BCSR14[0]	
Select Geth1 Mode	**Enable/Disable Geth-PHY1**	**Enable/Disable MII1**	**Function Selected**
00 (RGMII)	0 - **Enable**	0 - **Disable**	RGMII
00 (RGMII)	0 - **Enable**	1 - **Enable**	MII
00(RGMII)	1 - **Disable**	0 - **Disable**	Relevant signals connected to PIB
00 (RGMII)	1 - **Disable**	1 - **Enable**	MII
01 (RTBI)	0 - **Enable**	0 - **Disable**	RTBI
01 (RTBI)	0 - **Enable**	1 - **Enable**	MII
01 (RTBI)	1 - **Disable**	0 - **Disable**	Relevant signals connected to PIB
01 (RTBI)	1 - **Disable**	1 -**Enable**	MII

Chapter 7
Replacing Devices

This chapter provides instructions on replacing various devices on the MPC8360EA MDS Processor Board.

7.1 Replacing Flash Memory

To remove the flash memory, follow the instructions below in Figure 7-1 to Figure 7-4 below (in that order). Note that the flash memory can be changed no more than 50 times.

To replace the flash memory, follow the instructions in reverse order (Figure 7-4 to Figure 7-1), then secure the casing as shown in Figure 7-5.

.

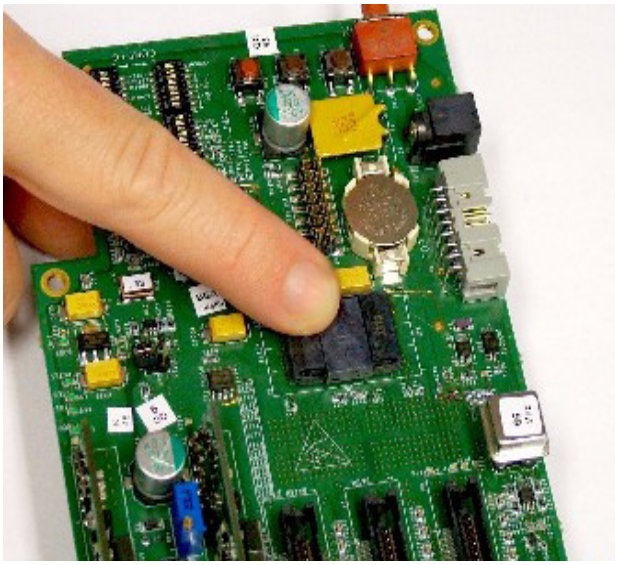


Figure 7-1. Flash Memory - push to dislodge casing

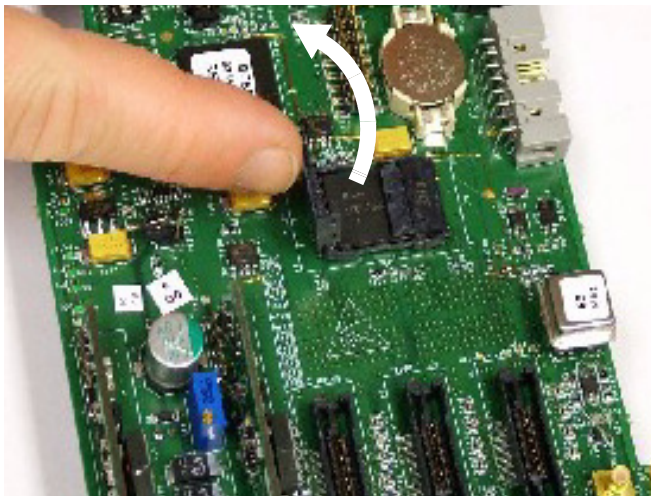


Figure 7-2. Flash Memory - open casing

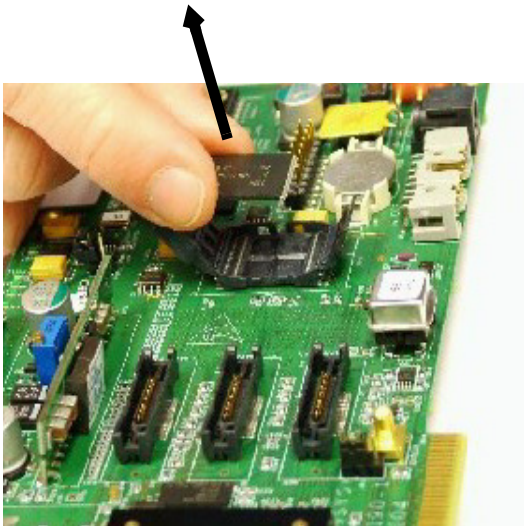


Figure 7-3. Flash Memory - remove memory unit

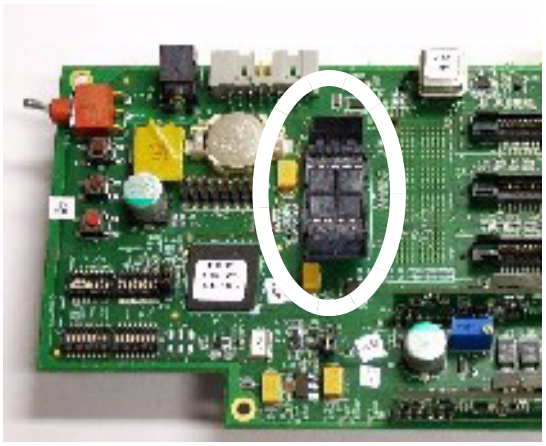


Figure 7-4. Flash Memory - unit removed

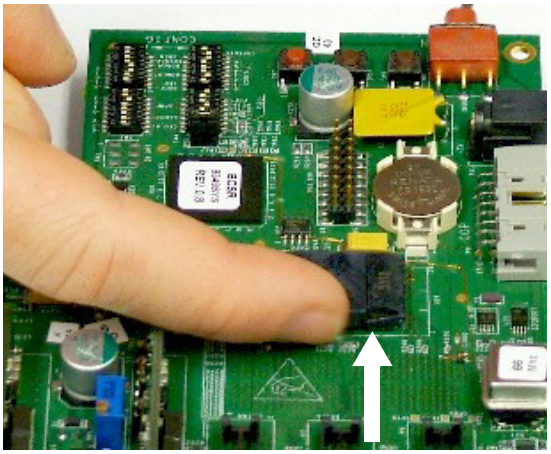


Figure 7-5. Flash Memory - replacing unit (push in until “click” is heard)

7.1.1 Cleaning Flash Memory

If there is some decrease in performance from the flash memory unit, the socket may need to be cleaned. Do this by dipping a tooth pick dipped in isopropyl alcohol, and gently removing any residual debris from the flash memory socket.

7.2 Replacing SODIMM units

To remove or replace the SODIMM units, follow the instructions in Figure 7-6 through Figure 7-10, in that order.

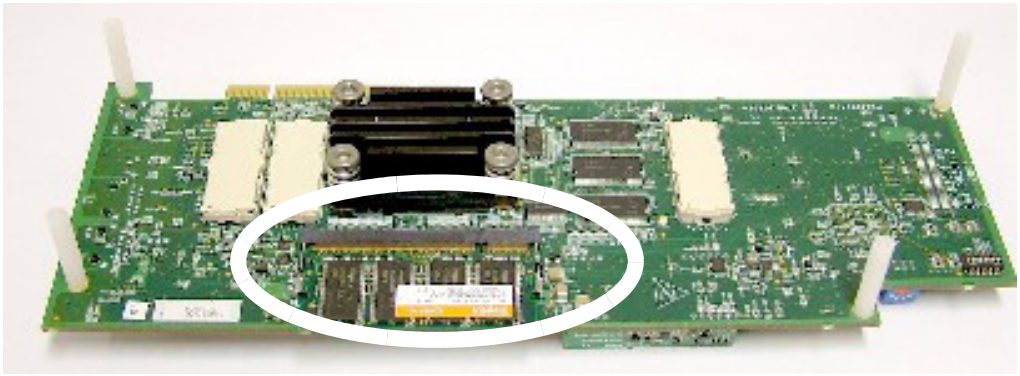


Figure 7-6. SODIMM Memory (for 72-bit, or 40-bit High - Socket U69 on underside of board)

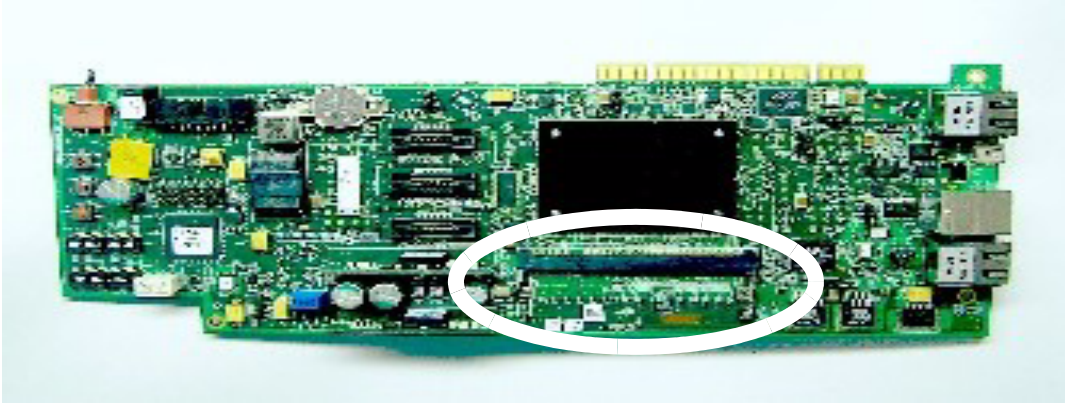


Figure 7-7. SODIMM Memory (for 40-bit - Socket U33 on top side of board)

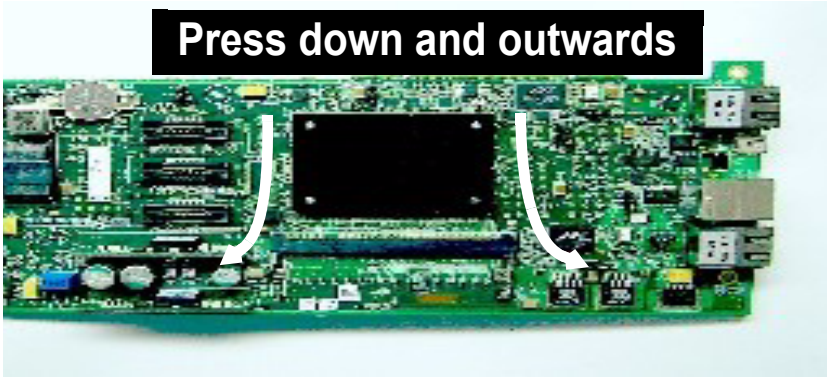


Figure 7-8. SODIMM Memory - release retaining clips

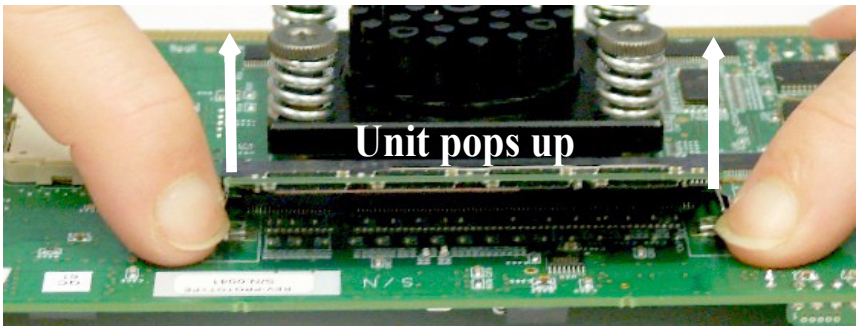


Figure 7-9. SODIMM Memory - release unit

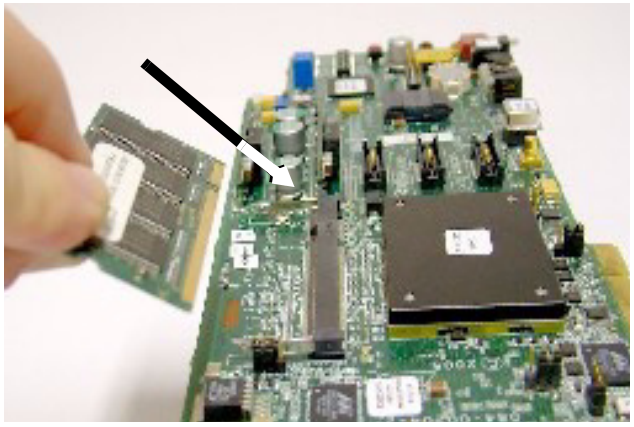


Figure 7-10. SODIMM Memory - remove/replace unit

7.3 Replacing MPC8360EA Processor

To remove the MPC8360EA processor, follow the instructions in Figure 7-11 to Figure 7-16 below, then remove the chip.

To replace the MPC8360EA processor, align the chip properly as shown in Figure 7-16, then follow the instructions in Figure 7-15 to Figure 7-11 below (in that order).

Note that the Allen wrench is provided in the tool kit.

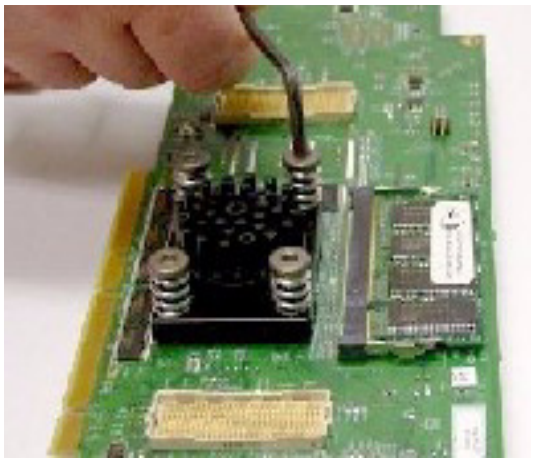


Figure 7-11. Loosen Allen screws

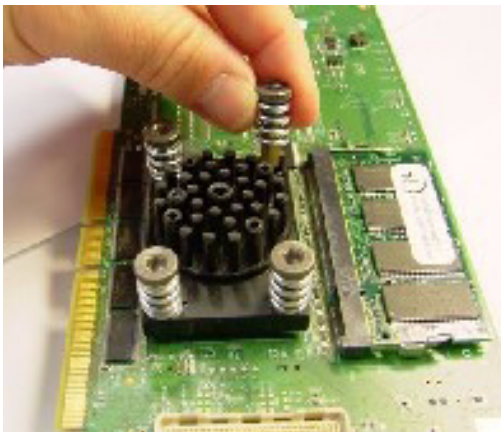


Figure 7-12. Remove Allen screws by hand

Figure 7-13. Allen screws removed

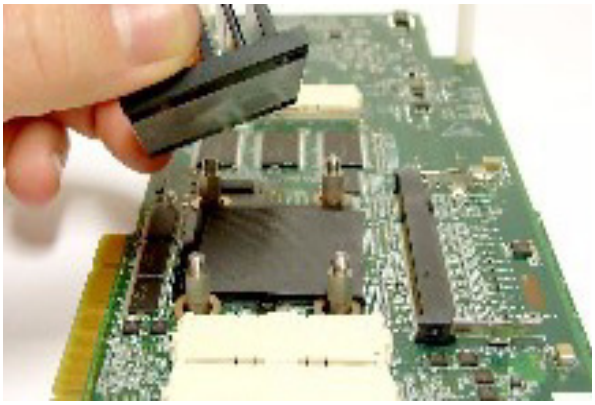


Figure 7-14. Remove heat sink

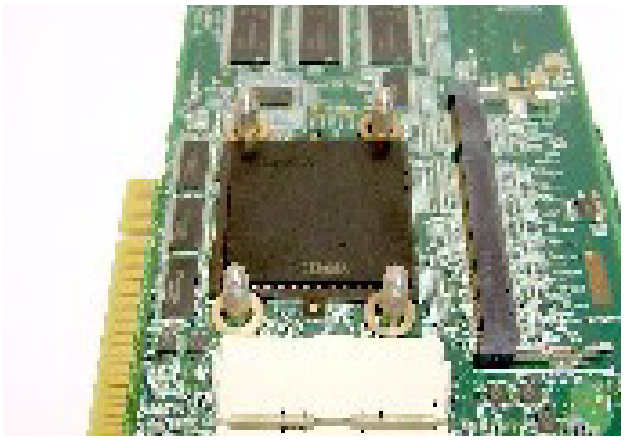


Figure 7-15. Heat sink removed

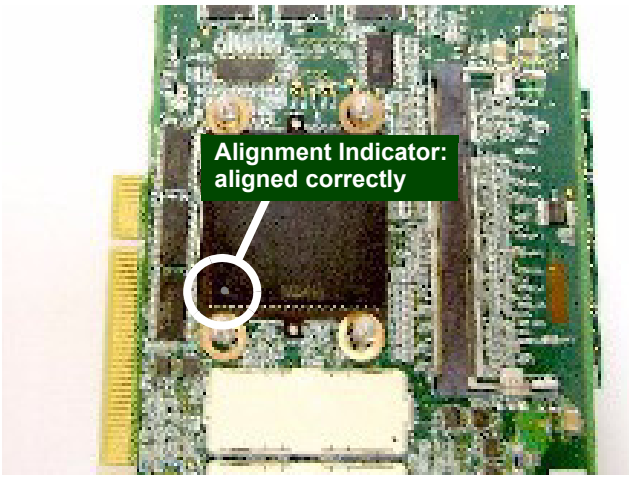


Figure 7-16. Chip alignment: Correct

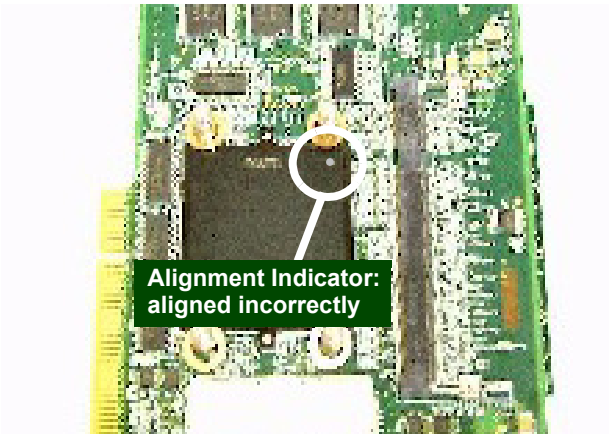


Figure 7-17. Chip alignment: Incorrect

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations Not Listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GmbH
Technical Information Center
Schatzbogen 7
81829 Muenchen, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T., Hong Kong
+800 2666 8080
support.asia@freescale.com

***For Literature Requests Only*:**
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-521-6274 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com