# PPC750FX® Evaluation Board

# User's Manual

SA14-2720-00

**Preliminary**

June 10, 2003

Printed in the United States of America June 2003



**Note:** This document contains information on products in the sampling and/or initial production phases of development. This information is subject to change without notice. Verify with your IBM field applications engineer that you have the latest version of this document before finalizing a design.

IBM Microelectronics Division
1580 Route 52, Bldg. 504
Hopewell Junction, NY 12533-6351

The IBM home page can be found at http://www.ibm.com

The IBM Microelectronics Division home page can be found at http://www.ibm.com/chips

June 10, 2003

# Contents

# Tables

# Figures

# About This Book

This manual describes an evaluation platform for the PPC750FX chip.

## Who Should Use This Book

This book is written to aid programmers and other technical personnel in the use of the PPC750FX Evaluation Board. In order to use the board and this document, the reader shouldbe familiar with the following:

- PowerPC Architecture™
- PCI bus
- Embedded microprocessor hardware
- IBM RISCWatch™ debugger

## How to Use This Book

This book describes the features and interfaces of the IBM PPC750FX Evaluation Board. This book contains the following sections:

- *Overview* provides a brief overview of the processor chip. Some chip aspects important to understanding the board design are discussed in greater detail.
- *Board Design* describes the architecture of the evaluation board.
- *Memory Map* describes the address space usage of the board. Tables are provided which define the access methods for all memory-mapped registers on the board.
- *Programming the System Controller* outlines the required programming to configure the MV64360 system controller for the board memory and peripherals.
- *Reset and Interrupts* lists the sources of resets and interrupts on the board, and provides information required to program the PPC750FX amd MV64360 interrupt controllers.
- *Switches* locates and describes the function of all switches on the board, and indicates their default settings.
- *Fuses, Batteries, Regulators, and Fans* locates and describes the function of fuses, batteries, and voltage regulator adjustments on the board.
- *Displays* locates and describes all displays on the board.
- *Jumpers* locates and describes all jumpers on the board, and indicates their default settings.
- *Connectors* locates and describes all connectors on the board, and identifies the pin usage.
- *CPLD Programming* provides the source code and timing information for the CPLDs on the board.
- *Bills of Materials* provides lists of materials and parts that are assembled on the board, parts shipped with the board but not assembled on the board, and other tools that are useful while using the board.

## Related Publications

The following publications contain related information:

- *PowerPC 750FX RISC Microprocessor Embedded Controller Data Sheet*
- *PowerPC 750FX RISC Microprocessor Embedded Controller Functional Specification*
- *PowerPC 750FX RISC Microprocessor Embedded Controller User's Manual*
- *PowerPC 750FX RISC Microprocessor Evaluation Design Kit User's Manual*
- *PowerPC 750FX RISC Microprocessor Evaluation Board Schematics*
- *PowerPC Architecture*
- *PowerPC Microprocessor Family: The Programming Environments*, MPRPPCFPE-01
- *PowerPC Embedded Processor Solutions*, SC09-3032, a CDROM which includes the *RISCWatch Debugger User's Guide*
- *PowerPC CoreConnect Bus (PLB) Specification*
- *PCI Local Bus Specification*
- *Marvell® MV64360/1/2 Data Sheet*

# 1. Overview

The PowerPC® 750FX Evaluation Board is an evaluation platform intended to support the needs of prospective users of the IBM PowerPC 750FX processor. The form factor of the board is a full-length PCI card. This board is suitable for software development, for benchmarking, and for detailed study of the hardware. This board contains two PPC750FX processors. A memory control and PCI bridge function provided on the board coordinates the operations of the two processors. The two processors may be used together or independently under program control. Both processors share the available memory and the PCI interface. The board can appear to the PCI interface as either a 64-bit adapter or a 64-bit host.

Please be aware that the circuitry on this board is sometimes more complex than would be required for a board design limited to a particular application. A customer who is developing his own design using this board design as a guide should simplify the design wherever his application allows.

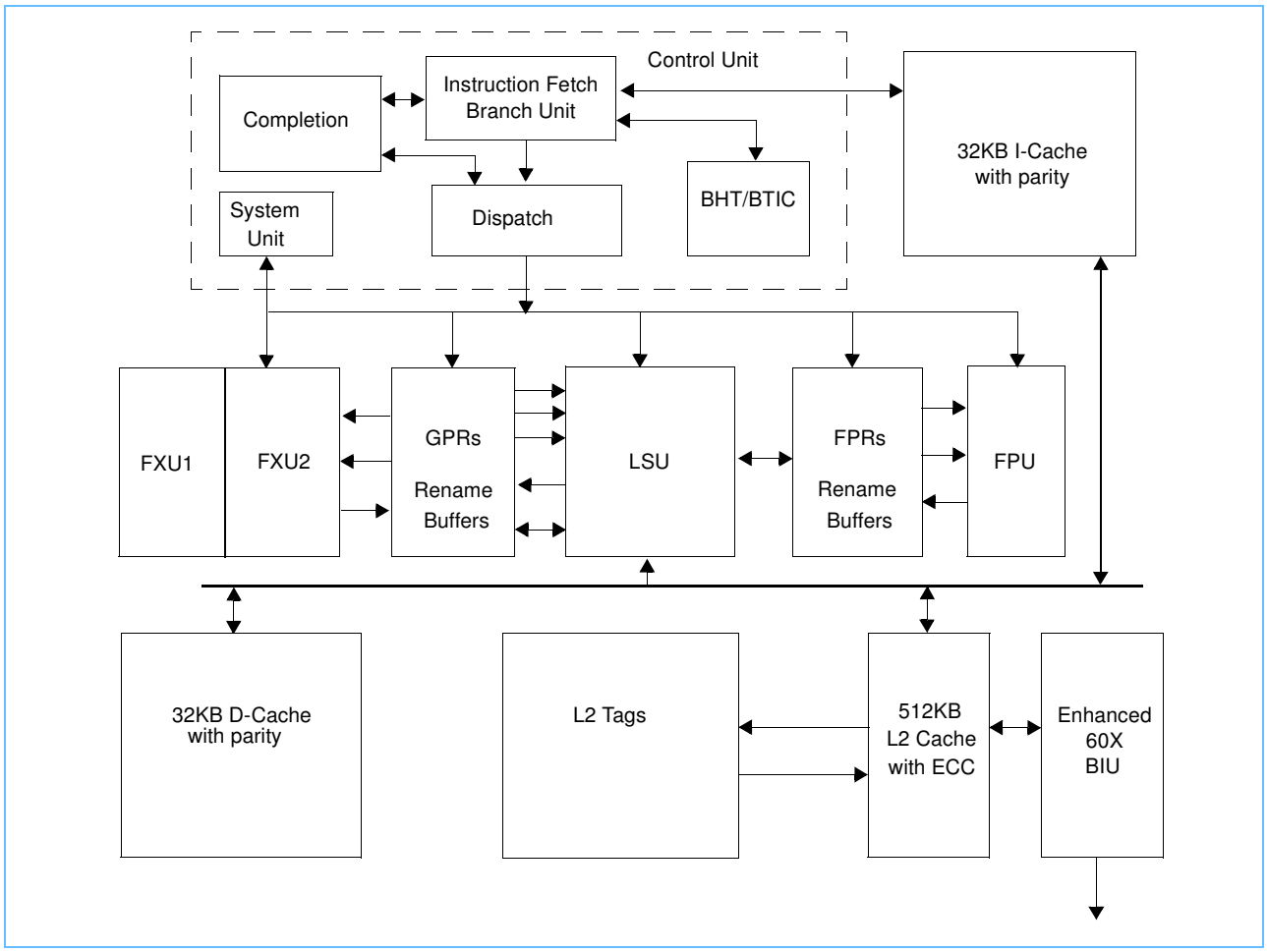
> **Warning:** IBM is not responsible for use of the circuit designs on this board or use of the design of the board itself in any other applications. Any functional, reliability, or safety issues resulting from the use of any part of this board design, including copying the board, are the responsibility of the user.

The following sections will highlight the PPC750FX processor, and will then briefly discuss the features available on the board.

## 1.1 PowerPC 750FX RISC Microprocessor Features

The IBM PowerPC 750FX RISC Microprocessor is a 32-bit implementation of the IBM PowerPC family of reduced instruction set computer (RISC) microprocessors. The PPC750FX is targeted for high performance, low power systems using a 60x bus. The 750FX also includes an internal 512KB L2 cache with on-board Error Correction Circuitry (ECC). A block diagram of the processor chip is provided in *Figure 1-1*.

*Figure 1-1. PPC750FX Block Diagram*




The PPC750FX processor has the following features:

- Branch processing unit
  - Four instructions fetched per clock.
  - One branch processed per cycle (plus resolving two speculations).
  - Up to one speculative stream in execution, one additional speculative stream in fetch.
  - 512-entry branch history table (BHT) for dynamic prediction.
  - 64-entry, 4-way set associative branch target instruction cache (BTIC) for eliminating branch delay slots.
- Dispatch unit
  - Full hardware detection of dependencies (resolved in the execution units).
  - Dispatch two instructions to six independent units (system, branch, load/store, fixed-point unit 1 (FXU1), fixed-point unit 2 (FXU2), or floating-point).
  - Four-stage pipeline: fetch, dispatch, execute, and complete.
  - Serialization control (predispatch, postdispatch, execution, serialization).

- Decode
  - Register file access.
  - Forwarding control.
  - Partial instruction decode.
- Load/Store unit
  - One cycle load or store cache access (byte, half word, word, double word).
  - Effective address generation.
  - Hits under misses (one outstanding miss).
  - Single-cycle misaligned access within double word boundary.
  - Alignment, zero padding, sign extend for integer register file.
  - Floating-point internal format conversion (alignment, normalization).
  - Sequencing for load/store multiples and string operations.
  - Store gathering.
  - Cache and translation look-aside buffer (TLB) instructions.
  - Big and little-endian byte addressing supported.
  - Misaligned little-endian support in hardware.
- Fixed-point units
  - FXU1: multiply, divide, shift, rotate, arithmetic, logical.
  - FXU2: shift, rotate, arithmetic, logical.
  - Single-cycle arithmetic, shift, rotate, logical.
  - Multiply and divide support (multi-cycle).
  - Early out multiply.
  - Thirty-two, 32-bit general purpose registers.
- Floating-point unit
  - Support for IEEE-754 standard single- and double-precision floating-point arithmetic.
  - Optimized for single-precision multiply/add.
  - Thirty-two, 64-bit floating point registers.
  - Enhanced reciprocal estimates.
  - Three-cycle latency, one-cycle throughput, single-precision multiply-add.
  - Three-cycle latency, one-cycle throughput, double-precision add.
  - Four-cycle latency, two-cycle throughput, double-precision multiply-add.
  - Hardware support for divide.
  - Hardware support for denormalized numbers.
  - Time deterministic non-IEEE mode.
- System unit
  - Executes CR logical instructions and miscellaneous system instructions.
  - Special register transfer instructions.
- Level 1 (L1) Cache structure
  - 32KB, 32-byte line, 8-way set associative instruction cache.
  - 32KB, 32-byte line, 8-way set associative data cache.
  - Single-cycle cache access.

  - Pseudo-LRU replacement.
  - Copy-back or write-through data cache (on a page per page basis).
  - Parity on L1 tags and arrays.
  - Three-state (MEI) memory coherency.
  - Hardware support for data coherency.
  - Non-blocking instruction cache (one outstanding miss).
  - Non-blocking data cache (two outstanding misses).
  - No snooping of instruction cache.

- Memory management unit
  - 64-entry, 2-way set associative instruction TLB (total 128).
  - 64-entry, 2-way set associative data TLB (total 128).
  - Hardware reload for TLB's.
  - Eight instruction BAT's and eight data BATs.
  - Virtual memory support for up to 4PB ($2^{52}$) virtual memory.
  - Real memory support for up to 4GB ($2^{32}$) of physical memory.
  - Support for big/little-endian addressing.

- Dual PLLs
  - Allows seamless frequency switching.

- Level 2 (L2) cache
  - Internal L2 cache controller and 4K-entry tags; 512KB data SRAMs.
  - Two-way set associative, supports locking by way.
  - Copy-back or write-through data cache on a page basis, or for all L2.
  - 64-byte sectored line size.
  - L2 frequency at core speed.
  - ECC protection on SRAM array.
  - Parity on L2 tags.
  - Supports up to 2 outstanding misses (1 data and 1 instruction or 2 data).

- Bus interface
  - 32-bit address bus.
  - 64-bit data bus (can be operated in 32-bit mode).
  - Core-to-bus frequency multipliers of 3.5x, 4x, 4.5x, 5x, 5.5x, 6x, 6.5x, 7x, 7.5x, 8x, 8.5x, 9x, 9.5x, 10x, 11x, 12x, 13x, 14x, 15x, 16x, 17x, 18x, 19x, and 20x supported.
  - Supports 1.8V, 2.5V, or 3.3V I/O modes.

- Power
  - Low power consumption with low voltage.
  - Dynamic power management.
  - Three static power saving modes: doze, nap, and sleep.
  - Thermal Assist Unit (TAU).

- Reliability and Serviceability
  - Parity checking on 60x busses.
  - ECC checking on L2 cache.
  - Parity on the L1 arrays.
  - Parity on the L1 and L2 tags.
- Testability
  - Level-sensitive scan design (LSSD).
  - Powerful diagnostic and test interface through Common On-Chip Processor (COP) and IEEE 1149.1 (JTAG) interface.

## 1.2 Board Features

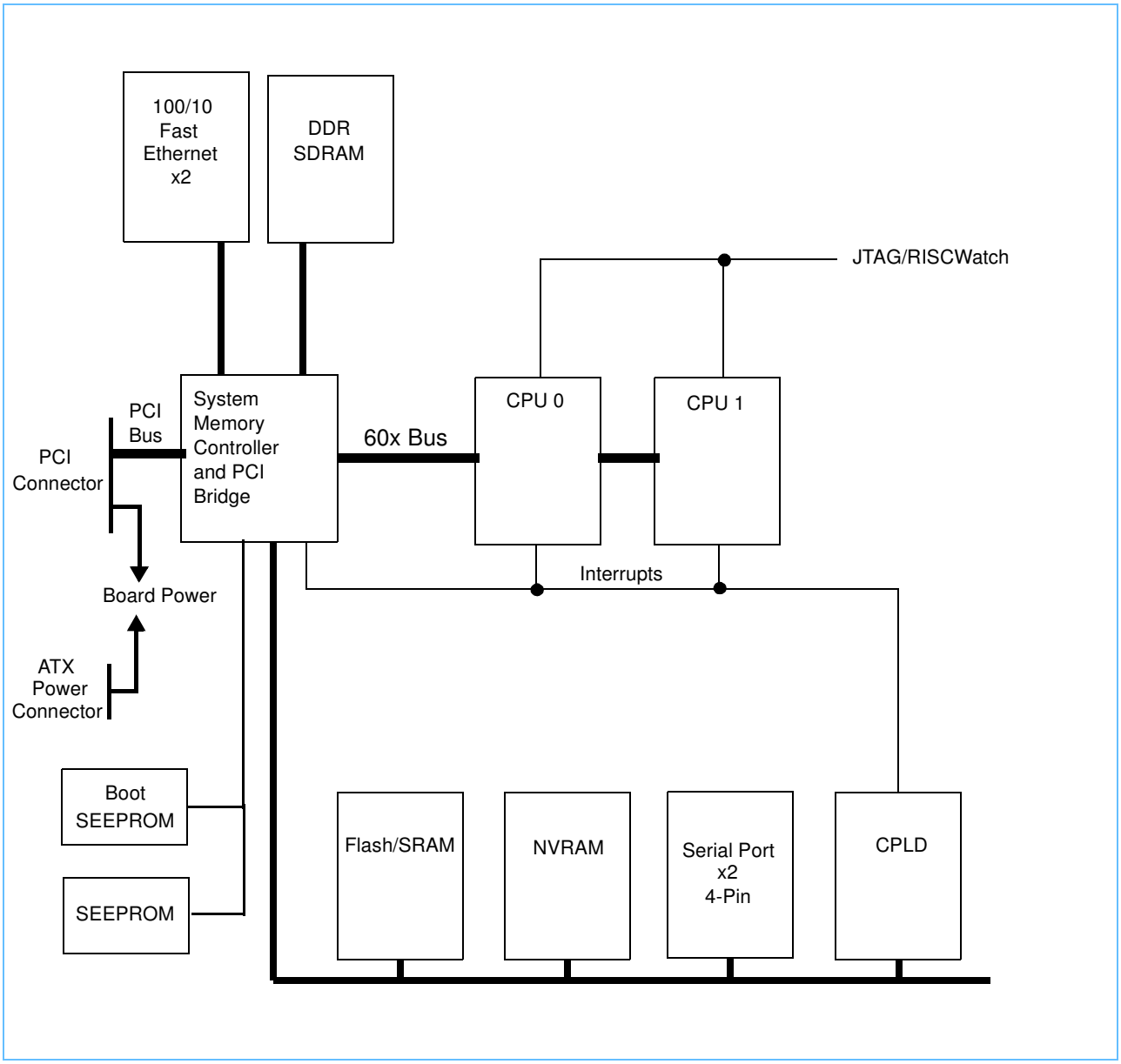
The features of the PPC750FX evaluation board are summarized briefly below. More detail may be found in *Section 2 Board Design* on page 19.

- PCI adapter form factor
- Two IBM PowerPC 750FX processors
- Marvell® MV64360 System Controller
- 256MB DDR SDRAM with ECC
- 1MB 8-bit wide socketed Flash (2 - 512KB devices)
- 1MB 8-bit wide SRAM
- 32MB 32-bit wide Flash
- 32KB Ferroelectric Nonvolatile RAM (FRAM)
- Two 100BASE-TX Ethernet ports
- Two 16550 compatible serial ports
- Two 64kb IIC Serial EEPROMs
- Single RISCWatch header for both processors
- Powered either externally or from PCI slot
- External system clock input
- External input for programming the on-board CPLD (FPGA)

# 2. Board Design

*Figure 2-1* illustrates the architecture of the PPC750FX evaluation board. Subsequent sections discuss aspects of *Figure 2-1* in more detail.

*Figure 2-1. PPC750FX Board Architecture*



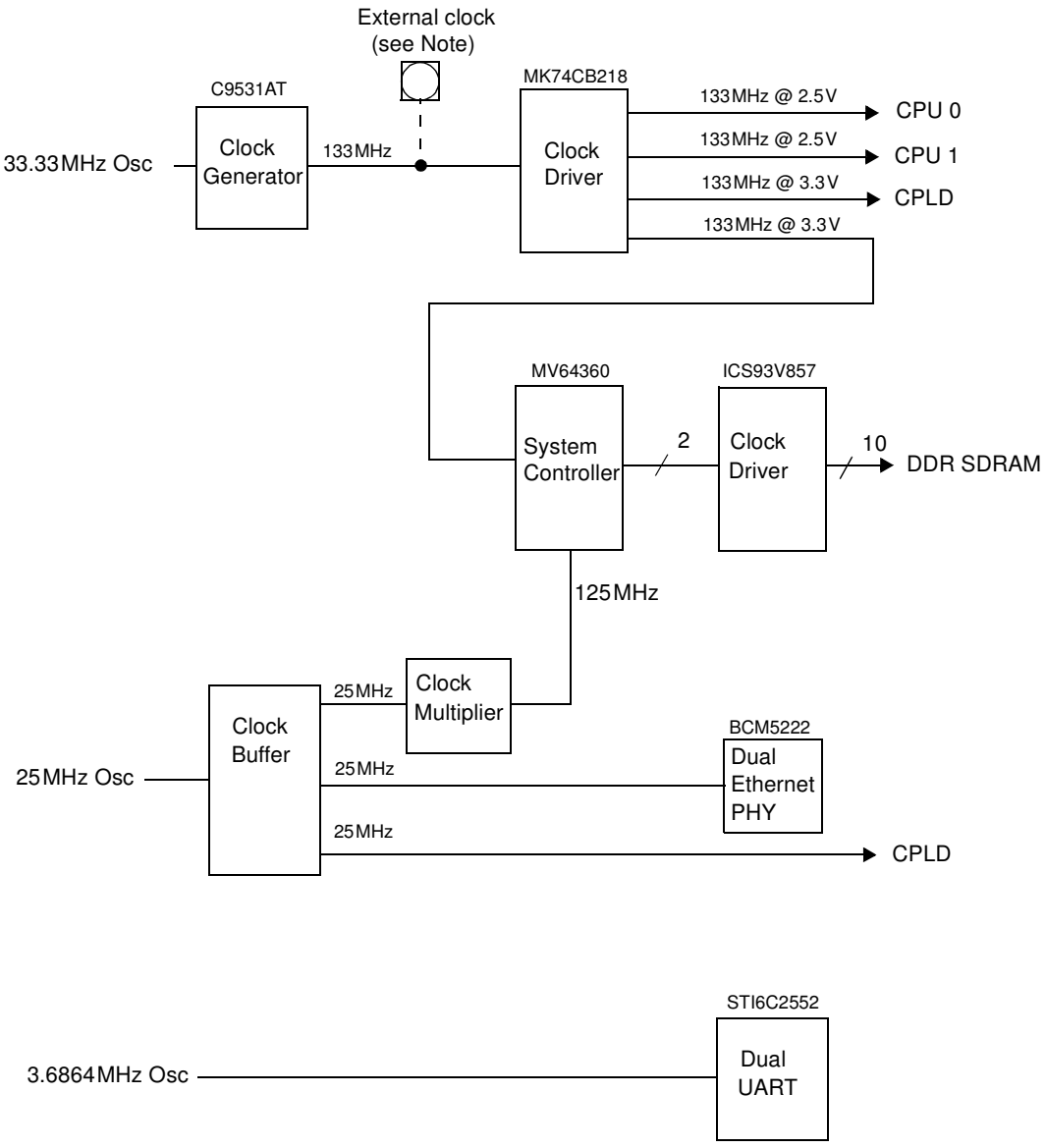

## 2.1 Processor

The PPC750FX evaluation board is based upon the PPC750FX processor. See *Section 1.1 PowerPC 750FX RISC Microprocessor Features* on page 13 for details. There are two PPC750FX processors on this board.

## 2.2 Board Clocking

The clock architecture of the PPC750FX board is illustrated in *Figure 2-2*.

*Figure 2-2. Clock Distribution on the PPC750FX Board*




**Note :** Rework to the board is required to use the external oscillator input.

## 2.3 Internal Processor Clocking

The PPC750FX requires a single system clock input SYSCLK. The frequency of this input determines the frequency of the PPC750FX bus interface. Internally, the PPC750FX uses a phase-lock loop (PLL) circuit to generate a master core clock that is frequency-multiplied and phase-locked to the SYSCLK input. The PLL in the PPC750FX is configured using seven pins PLL_CFG(0:4) and PLL_RANGE(0:1). On the PPC750FX evaluation board, the configuration of these pins is controlled by switch settings (see *Section 6 Switches* on page 39).

## 2.4 System Controller

The board contains a Marvell MV64360 system controller that connects to the 60x bus of the PPC750FX, and provides an interface to DDR SDRAM, the PCI bus, the integrated Ethernet MACs, the integrated SRAM, an interrupt controller, DMA engines, and an interface to attach external devices. Hereafter, in this document, this component is referred to as the system controller.

## 2.5 SDRAM Interface

This board provides 256MB of permanently mounted DDR SDRAM operating at 133.33MHz. The interface to the SDRAM is through the system controller, and is accessed using DRAM chip selects CS0 and CS1. The SDRAM on the board is 72 bits wide and allows the use of the SDRAM Error Checking and Correction (ECC) feature in the system controller if desired.

## 2.6 PCI Bus

This PPC750FX evaluation board is a full-length PCI card and is intended to be operated while plugged into a PCI slot in a personal computer or a PCI backplane. However, because an ATX power connector is provided, it can be operated without being plugged into a PCI slot. If this external mode of operation is used, the PCI bus will not be available.

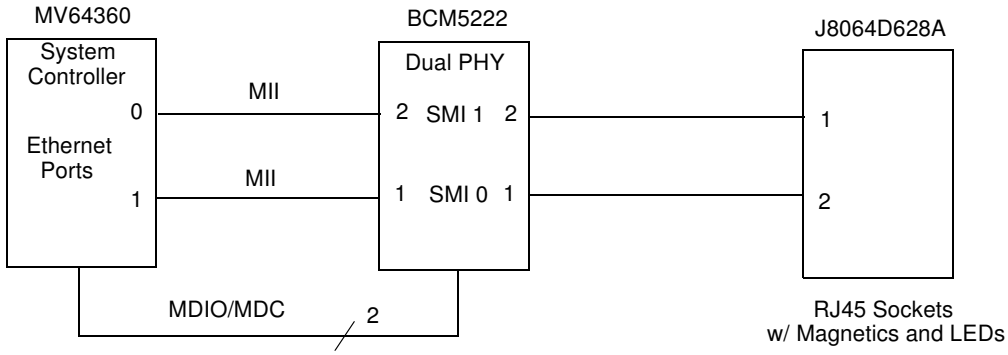
## 2.7 Ethernet

The board provides two 100BASE-TX Ethernet interfaces. The physical layer for both Ethernet ports is provided by the BCM5222 which contains two medium-independent interface (MII) PHYs. The five address pins of the BCM5222 are tied to ground making the addresses of the two PHYs 0 and 1. *Table 2-1* shows the relationship between the two Ethernet ports being used in the MV64360, the PHY to which each port is connected in the BCM5222, the address of each PHY, and finally the RJ45 connectors for each port (see J20 in *Figure 10-1* on page 54).

*Table 2-1. Ethernet Ports*

| MV64360 Ethernet Port No. | BCM522 PHY No. | Serial Manangement Interface PHY Address | RJ45 Connector |
|---|---|---|---|
| 0 | 2 | 1 | 1 |
| 1 | 1 | 0 | 2 |

*Figure 2-3. Board Ethernet Architecture*




The supported media is Category 5A Unshielded Twisted Pair cable (UTP), accessed via two RJ45 connectors on the board. The two RJ45 connectors are in a common housing with integrated magnetics and LEDS.

## 2.8 Flash Memory

The following describes how to access Flash memory directly on the board.

Eight-bit Flash memory is used on the PPC750FX board. No benchmarking impact is expected from the use of a narrow rather than a wide Flash array. For performance work, one should expect to replace the initial firmware support provided in the Flash with more optimized routines residing in DRAM.

The PPC750FX board contains 1MB of 8-bit wide Flash memory provided by two socketed 8b x 0.5Mb modules, and 32MB of 32-bit wide Flash memory provided by two 16b x 16Mb module (+3.3V only) for data or code storage. Additionally, 1MB of SRAM, provided as two 512KB modules, can be used in this memory space.

The board can be set to boot from 8-bit wide Flash with SRAM below it in memory or, alternatively, it can boot from the SRAM with the 8-bit wide Flash below it in memory. This setup uses system controller chip select BootCS.

When set up to use BootCS to boot from 8-bit wide Flash or from SRAM, the system controller chip select CS0 is used to select the 32-bit wide Flash. If desired, these chip selects can be swapped so that the board will boot from the 32-bit wide Flash using BootCS, then CS0 selects the 8-bit wide Flash/SRAM combination.

There are two switches used to control where the system controller chip selects are directed. See *System Controller Initialization* on page 42 for details on U17 switch 6 and U24 switch 7.

The 8-bit wide Flash is installed at the top of the address space. Immediately below that Flash in the address space is the SRAM. Switch #7 on switch U24 allows the exchange of the two blocks in the address space. The intent of this SRAM is to aid in the debug of ROM boot code, not for speed enhancement. Flash contents will be copied to SRAM, then SRAM will be placed at the top of the address space. ROM code can then be debugged from SRAM, allowing the placement of unlimited software break points.

**Note 1:** A jumper at J8 can be installed to prevent any 32-bit Flash write operations.

**Note 2:** Caching the 8- or 32-bit Flash memories is not supported.

*Table 2-2* and *Table 2-3* describe the switch settings and the resulting configurations.

*Table 2-2. Switch Settings*

| Configuration | U17 SW6 | U24 SW 7 | Description |
|---|---|---|---|
| 1 | ON | ON | 8-bit boot, Flash at higher address |
| 2 | ON | OFF | 8-bit boot, SRAM at higher address |
| 3 | OFF | ON | 32-bit boot, Flash at higher address |
| 4 | OFF | OFF | 32-bit boot, SRAM at higher address |

*Table 2-3. Flash Configurations*

| Configuration | Address Range | Module(s) selected |
|---|---|---|
| 1 | 0xFFF00000 to 0xFFFFFFFF | 8-bit Flash controlled by BootCS |
| | 0xFFE00000 to 0xFFEFFFFF | 8-bit SRAM controlled by BootCS |
| | 0xFC000000 to 0xFDFFFFFF | 32-bit Flash controlled by DevCS0 |
| 2 | 0xFFF00000 to 0xFFFFFFFF | 8-bit SRAM controlled by BootCS |
| | 0xFFE00000 to 0xFFEFFFFF | 8-bit Flash controlled by BootCS |
| | 0xFC000000 to 0xFDFFFFFF | 32-bit Flash controlled by DevCS0 |
| 3 | 0xFE000000 to 0xFFFFFFFF | 32-bit Flash controlled by BootCS |
| | 0xFC100000 to 0xFC1FFFFF | 8-bit Flash controlled by DevCS0 |
| | 0xFC000000 to 0xFC0FFFFF | 8-bit SRAM controlled by DevCS0.<br>**Note:** In configuration 3, SRAM is at the beginning of the DevCS0 memory range followed by 8-bit wide Flash. |
| 4 | 0xFE000000 to 0xFFFFFFFF | 32-bit Flash controlled by BootCS |
| | 0xFC100000 to 0xFC1FFFFF | 8-bit SRAM controlled by DevCS0 |
| | 0xFC000000 to 0xFC0FFFFF | 8-bit Flash controlled by DevCS0.<br>**Note:** In configuration 4, 8-bit wide Flash is at the beginning of the DevCS0 memory range followed by SRAM. |

**Notes: 1.** The reset vector of the PPC750FX is 0xFFF00100.

**2.** The base addresses of peripherals attached to the MV64360 system controller device are software dependent. The values in the table above are used by the PPC750FX Evaluation Kit Software. Other software environments may use different values for the peripheral base addresses.

## 2.9 NVRAM

The board provides 32KB of non-volatile RAM. This memory is attached to the system controller device interface, and is accessed using chip select DevCS3. This particular type of non-volatile ram uses magnetic core technology and is referred to in the board schematics as FRAM. It does not require any battery power to maintain its contents.

## 2.10 SRAM

The PPC750FX evaluation board provides 1MB of permanently mounted SRAM. This memory interfaces to the system controller. Access to this memory by the processors is through the system controller. Address space for SRAM is shared with Flash memory. See *Flash Memory* on page 22 for details on how the address space can be configured.

In addition to the SRAM on the board, there are 256KB of addressible SRAM integrated in the system controller module.

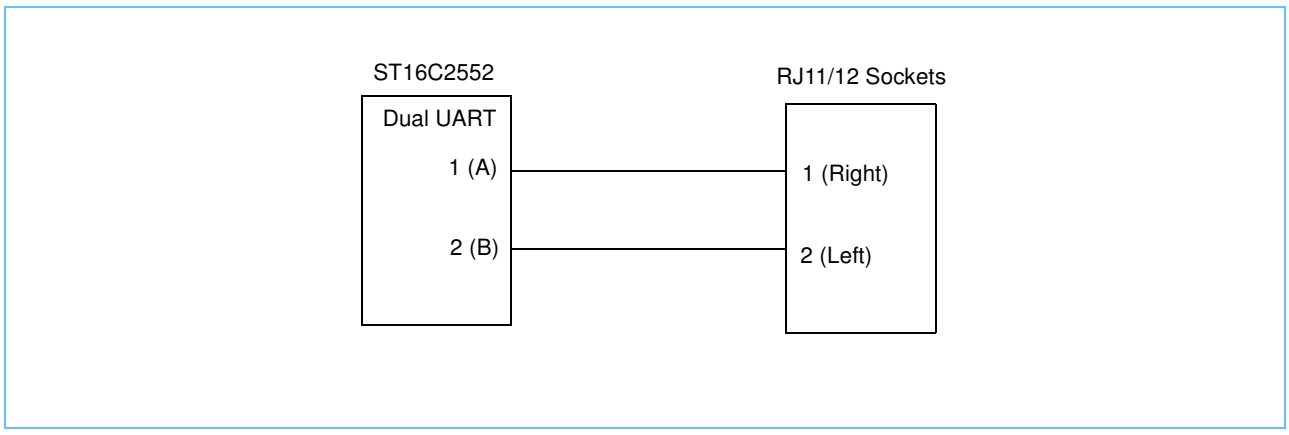
## 2.11 Serial Ports

The board utilizes an Exar ST16C2552 DUART to provide two 16550 compatible UARTs. The DUART is attached to the device interface of the system controller, and is accessed using chip select DevCS2. Each UART provides four interface signals (Tx, Rx, DSR, DTR) and is connected to an RJ11/12 connector.

Both serial ports are clocked by the same 3.68MHz oscillator provided on the board.

The Multi-Protocol Serial Controllers in the system controller are not supported on the PPC750FX evaluation board.

*Figure 2-4. Board Serial Port Architecture*



## 2.12 Logic Analyzer Connections

The system controller device bus is attached to Mictor logic analyzer connectors.

HP Logic analyzer connection to the PCI bus is accomplished using a FuturePlus PCI Local Bus Passive Analysis Probe, vendor part number FS2005. The customer may purchase this probe from the manufacturer. Probe hardware does not ship with the board.

## 2.13 Power Supply

The PPC750FX evaluation board obtains its power from the PCI slot connector into which it is plugged or from the on-board ATX power connector.

### 2.13.1 PCI Voltages

The voltages provided through the PCI slot connector are:
- +5V
- +3.3V

All of the voltages described in the following sections are developed from the +3.3V PCI voltage.

### 2.13.2 System Controller Voltages

The system controller requires four voltages:

- System controller I/O—+3.3V
- DRAM —+2.5V
- CPU I/O—+2.5V
- System controller logic and Ethernet—+1.8V

  **Note:** There are no external connection points for the +1.8V or +2.5V supplies. As a result variable voltage testing and current measurement capability for these supplies are not available.

### 2.13.3 PPC750FX Voltages

The PPC750FX chip requires two voltages:
- Logic and PLL analog circuits— +1.45V
- 60x bus I/O circuits— +2.5V

Both voltages are generated by on-board regulators from the +3.3V voltage. Current measurement points are available for both voltages. These measurement points can also be used to connect external voltage supplies.

**Note:** The 60x bus voltage supply to the PPC750FX can be +1.8V, +2.5V, or +3.3V. To use any voltage other than the +2.5V supplied by the board, the voltage must be supplied externally, and the 60x voltage selection signals to the PPC750FX must be programmed accordingly.

### 2.13.4 SDRAM Voltages

An on-board regulator supplies +2.5V to the DDR SDRAM. This voltage is also the supply for the DRAM interface in the system controller.There is also a +1.25V reference voltage provided to the SDRAM. There is no current measurement point provided for this voltage.

## 2.14 Form Factor

The PPC750FX board is a full-length PCI card intended to be plugged into and operated in a standard PCI slot on a personal computer or a PCI backplane. If a personal computer or PCI backplane are not available, it can operate stand-alone with an external ATX power supply connected at J34.

# 3. Memory Map

Table 3-1 provides a summary of the board address space usage. For details about address space usage relating to the processor registers, refer to the *PPC750FX Embedded Processor User's Manual*.

*Table 3-1. Board Address Space Usage*

| Peripheral | Start Address | End Address | Chip Select | Size |
|---|---|---|---|---|
| DDR SDRAM | 0x00000000 | 0x0FFFFFFF | SDRAM CS0 and CS1 | 256MB |
| MV64360 Integrated SRAM | 0x42000000 | 0x4203FFFF | n/a | 256KB |
| FRAM | 0xEF500000 | 0xEF507FFF | DevCS3 | 32KB |
| ST16C2552 UART Channel B | 0xEF600000 | 0xEF600007 | DevCS2 | 8B |
| ST16C2552 UART Channel A | 0xEF600008 | 0xEF60000F | DevCS2 | 8B |
| CPLD Registers | 0xEF700000 | 0xEF700004 | DevCS1 | 5B |
| MV64360 Registers | 0xF1000000 | 0xF100FFFF | n/a | 64KB |
| 32-bit Flash | 0xFC000000 | 0xFDFFFFFF | DevCS0 | 32MB |
| SRAM | 0xFFE00000 | 0xFFEFFFFF | BootCS | 1MB |
| 8-bit Flash | 0xFFF00000 | 0xFFFFFFFF | BootCS | 1MB |
| **Note:** The base addresses of peripherals attached to the MV64360 system controller device are software dependent. The values in the table above are used by the PPC750FX Evaluation Kit Software. Other software environments may use different values for the peripheral base addresses. | | | | |

## 3.1 CPLD Register Definitions

This section provides description by bit for each of the CPLD registers.

Each CPLD register is 8 bits wide. In the tables below, the most significant bit is bit 0, and the least significant bit is bit 7. The CPLD source code uses the reverse bit ordering.

*Table 3-2. Register0*

| Bit | Name | R/W | Description |
|---|---|---|---|
| 0:7 | CPLD Revision | R | Revision level of CPLD code |

*Table 3-3. Register1*

**Note:** This register should be written before reading in order to latch the most current status. Any value can be written to the register.

| Bit | Name | R/W | Description |
|---|---|---|---|
| 0 (msb) | na | na | Unused |
| 1 | na | na | Unused |
| 2 | ATX or PCI Power | R | 0 = Using an ATX power supply<br>1 = Power obtained from a PCI slot |
| 3 | Spare Switch B | R | 0 = U35 Switch 8 is ON<br>1 = U35 Switch 8 is OFF |
| 4 | Spare Switch A | R | 0 = U30 Switch 8 is ON<br>1 = U30 Switch 8 is OFF |
| 5 | PCI Adapter/Host select | R | 0 = PCI Host mode U24 Switch 6 is ON<br>1 = PCI Adapter mode U24 Switch 6 is OFF |
| 6 | 8-bit Flash/SRAM swap select | R | 0 = 8-bit Flash is at a higher address in memory, U24 Switch 6 is ON<br>1 = 8-bit SRAM is at a higher address in memory, U24 Switch 6 is ON |
| 7 (lsb) | BootFlash select | R | 0 = Booted from 8-bit flash or SRAM U17 Switch 6 is ON<br>1 = Booted from 32-bit flash U17 Switch 6 is OFF |

*Table 3-4. Register2*

| Bit | Name | R/W | Description |
|---|---|---|---|
| 0 (msb) | CPU1 MCP control | R/W | Asserts the Machine Check Pin (MCP) signal on CPU1<br>0 = CPU1 MCP signal not asserted<br>1 = CPU1 MCP signal asserted |
| 1 | CPU0 MCP control | R/W | Asserts the Machine Check Pin (MCP) signal on CPU0<br>0 = CPU0 MCP signal not asserted<br>1 = CPU0 MCP signal asserted |
| 2 | CPU TBEN control | R/W | Controls the state of the timebase enable (TBEN) signal of both CPUs<br>0 = timebase runs freely on both CPUs<br>1 = timebase frozen on both CPUs |
| 3 | CPU1 SMI control | R/W | Asserts the System Management Interrupt signal on CPU1<br>0 = CPU1 SMI signal not asserted<br>1 = CPU1 SMI signal asserted |
| 4 | CPU0 SMI control | R/W | Asserts the System Management Interrupt signal on CPU0<br>0 = CPU0 SMI signal not asserted<br>1 = CPU0 SMI signal asserted |
| 5 | DS4 LED control | R/W | 0 = DS4 LED is ON<br>1 = DS4 LED is OFF |
| 6 | DS2 LED control | R/W | 0 = DS2 LED is ON<br>1 = DS2 LED is OFF |
| 7 (lsb) | DS1 LED control | R/W | 0 = DS1 LED is ON<br>1 = DS1 LED is OFF |

*Table 3-5. Register3*

| Bit | Name | R/W | Description |
|---|---|---|---|
| 0 (msb) | Block MPP resets | R/W | Five MPP/GPP pins on the system controller can be used to control the SRESET and HRESET of pins of the processors, and an entire board reset. To give software a chance to configure the MPP/GPP pins 7, 8, 11, 12, and 24 properly, the signals are blocked by the CPLD until this bit is set to 1.<br>0 = MPP resets are blocked<br>1 = MPP resets are not blocked |
| 1:7 | unused | | |

*Table 3-6. Register4*

| Bit | Name | R/W | Description |
|---|---|---|---|
| 0:7 | Board Revision | R | Board revision level in binary (for example, 0x00000010 = Revision level 2). |

# 4. Programming the System Controller

This section provides guidance on programming the system controller to agree with the board design.

## 4.1 DDR SDRAM

The following are the characteristics of the DDR SDRAM memory on the PPC750FX evaluation board:

*Table 4-1. DDR SDRAM Characteristics*

| | |
|---|---|
| Memory Type | DDR SDRAM |
| Number of Row Addresses | 13 |
| Number of Column Addresses | 9 |
| Number of Module Banks | 2 |
| SDRAM Width, Primary | 16 bits |
| Error Checking SDRAM Width | 8 bits |
| Data Width | 72 bits |
| Number of SDRAM Banks | 4 |

### 4.1.1 SDRAM Controller Initialization

See Marvell MV64360/1/2 data sheet.

## 4.2 Device Controller Bank Register Settings

The following sections define the settings for the device controller bank registers in the system controller.

### 4.2.1 Device Bank 0 Parameters (32-bit Flash)

*Table 4-2. Device Bank 0 Parameters = 0x85A492BF*

| Field | Value (bin) | Comment |
|---|---|---|
| TurnOff | 111 | Number of Sysclk cycles that the system controller does not drive the address/data bus after completion of a device read |
| Acc2First | 0111 | Number of Sysclk cycles from the de-assertion of ALE to the cycle that the first read data is sampled |
| Acc2Next | 0101 | Number of Sysclk cycles in a burst read access between the cycle that samples data N to the cycle that samples data N+1 |
| ALE2Wr | 010 | Number of Sysclk cycles from ALE de-assertion to the assertion of Wr[0] |
| WrLow | 010 | Number of Sysclk cycles that Wr[0] is active |
| WrHigh | 010 | Number of Sysclk cycles between data beats of a burst write that Wr[0] is held in-active. BAdr and data are held valid for WrHigh-1 cycles |
| DevWidth | 10 | Device width of 32 bits |
| TurnOffExt | 0 | TurnOff extension (most significant bit) |
| Acc2FirstExt | 1 | Acc2First extension (most significant bit) |
| Acc2NextExt | 1 | Acc2Next extension (most significant bit) |
| ALE2WrExt | 0 | ALE2Wr extension (most significant bit) |
| WrLowExt | 1 | WrLow extension (most significant bit) |
| WrHighExt | 0 | WrHigh extension (most significant bit) |
| BadrSkew | 00 | Number of Sysclk cycles from when BAdr changes to the read of the data |
| DPEn | 0 | Parity Disabled |
| Reserved | 1 | |

### 4.2.2 Device Bank 1 Parameters (CPLD registers)

*Table 4-3. Device Bank 1 Parameters = 0x8004921A*

| Field | Value (bin) | Comment |
|---|---|---|
| TurnOff | 010 | Number of Sysclk cycles that the system controller does not drive the address/data bus after completion of a device read |
| Acc2First | 0011 | Number of Sysclk cycles from the de-assertion of ALE to the cycle that the first read data is sampled |
| Acc2Next | 0100 | Number of Sysclk cycles in a burst read access between the cycle that samples data N to the cycle that samples data N+1 |
| ALE2Wr | 010 | Number of Sysclk cycles from ALE de-assertion to the assertion of Wr[0] |
| WrLow | 010 | Number of Sysclk cycles that Wr[0] is active |
| WrHigh | 010 | Number of Sysclk cycles between data beats of a burst write that Wr[0] is held in-active. BAdr and data are held valid for WrHigh-1 cycles |
| DevWidth | 00 | Device width of 8 bits |
| TurnOffExt | 0 | TurnOff extension (most significant bit) |
| Acc2FirstExt | 0 | Acc2First extension (most significant bit) |
| Acc2NextExt | 0 | Acc2Next extension (most significant bit) |
| ALE2WrExt | 0 | ALE2Wr extension (most significant bit) |
| WrLowExt | 0 | WrLow extension (most significant bit) |
| WrHighExt | 0 | WrHigh extension (most significant bit) |
| BadrSkew | 00 | Number of Sysclk cycles from when BAdr changes to the read of the data |
| DPEn | 0 | Parity Disabled |
| Reserved | 1 | |

### 4.2.3 Device Bank 2 Parameters (UARTs)

*Table 4-4. Device Bank 2 Parameters = 0x8C002BD6*

| Field | Value (bin) | Comment |
|---|---|---|
| TurnOff | 110 | Number of Sysclk cycles that the system controller does not drive the address/data bus after completion of a device read |
| Acc2First | 1010 | Number of Sysclk cycles from the de-assertion of ALE to the cycle that the first read data is sampled |
| Acc2Next | 0111 | Number of Sysclk cycles in a burst read access between the cycle that samples data N to the cycle that samples data N+1 |
| ALE2Wr | 101 | Number of Sysclk cycles from ALE de-assertion to the assertion of Wr[0] |
| WrLow | 000 | Number of Sysclk cycles that Wr[0] is active |
| WrHigh | 000 | Number of Sysclk cycles between data beats of a burst write that Wr[0] is held in-active. BAdr and data are held valid for WrHigh-1 cycles |
| DevWidth | 00 | Device width of 8 bits |
| TurnOffExt | 0 | TurnOff extension (most significant bit) |
| Acc2FirstExt | 0 | Acc2First extension (most significant bit) |
| Acc2NextExt | 0 | Acc2Next extension (most significant bit) |
| ALE2WrExt | 0 | ALE2Wr extension (most significant bit) |
| WrLowExt | 1 | WrLow extension (most significant bit) |
| WrHighExt | 1 | WrHigh extension (most significant bit) |
| BadrSkew | 00 | Number of Sysclk cycles from when BAdr changes to the read of the data |
| DPEn | 0 | Parity Disabled |
| Reserved | 1 | |

### 4.2.4 Device Bank 3 Parameters (FRAM)

*Table 4-5. Device Bank 3 Parameters = 0x8D891445*

| Field | Value (bin) | Comment |
|---|---|---|
| TurnOff | 101 | Number of Sysclk cycles that the system controller does not drive the address/data bus after completion of a device read |
| Acc2First | 1000 | Number of Sysclk cycles from the de-assertion of ALE to the cycle that the first read data is sampled |
| Acc2Next | 010 | Number of Sysclk cycles in a burst read access between the cycle that samples data N to the cycle that samples data N+1 |
| ALE2Wr | 010 | Number of Sysclk cycles from ALE de-assertion to the assertion of Wr[0] |
| WrLow | 100 | Number of Sysclk cycles that Wr[0] is active |
| WrHigh | 100 | Number of Sysclk cycles between data beats of a burst write that Wr[0] is held in-active. BAdr and data are held valid for WrHigh-1 cycles |
| DevWidth | 00 | Device width of 8 bits |
| TurnOffExt | 0 | TurnOff extension (most significant bit) |
| Acc2FirstExt | 1 | Acc2First extension (most significant bit) |
| Acc2NextExt | 1 | Acc2Next extension (most significant bit) |
| ALE2WrExt | 0 | ALE2Wr extension (most significant bit) |
| WrLowExt | 1 | WrLow extension (most significant bit) |
| WrHighExt | 1 | WrHigh extension (most significant bit) |
| BadrSkew | 00 | Number of Sysclk cycles from when BAdr changes to the read of the data |
| DPEn | 0 | Parity Disabled |
| Reserved | 1 | |

### 4.2.5 Boot Device Parameters (8 bit flash and SRAM)

If booting from the 8-bit Flash or SRAM, the default value of the Boot Device Parameters register in the system controller is 0x8FCFFFFF. To improve the access time to the Flash contents, the register can be changed to the following:

*Table 4-6. Boot Device Bank Parameters = 0x8185D09E*

| Field | Value (bin) | Comment |
|---|---|---|
| TurnOff | 110 | Number of Sysclk cycles that the system controller does not drive the address/data bus after completion of a device read |
| Acc2First | 0011 | Number of Sysclk cycles from the de-assertion of ALE to the cycle that the first read data is sampled |
| Acc2Next | 010 | Number of Sysclk cycles in a burst read access between the cycle that samples data N to the cycle that samples data N+1 |
| ALE2Wr | 010 | Number of Sysclk cycles from ALE de-assertion to the assertion of Wr[0] |
| WrLow | 111 | Number of Sysclk cycles that Wr[0] is active |
| WrHigh | 010 | Number of Sysclk cycles between data beats of a burst write that Wr[0] is held in-active. BAdr and data are held valid for WrHigh-1 cycles |
| DevWidth | 00 | Device width of 8 bits |
| TurnOffExt | 0 | TurnOff extension (most significant bit) |
| Acc2FirstExt | 1 | Acc2First extension (most significant bit) |
| Acc2NextExt | 1 | Acc2Next extension (most significant bit) |
| ALE2WrExt | 0 | ALE2Wr extension (most significant bit) |
| WrLowExt | 0 | WrLow extension (most significant bit) |
| WrHighExt | 0 | WrHigh extension (most significant bit) |
| BadrSkew | 00 | Number of Sysclk cycles from when BAdr changes to the read of the data |
| DPEn | 0 | Parity Disabled |
| Reserved | 1 | |

# 5. Reset and Interrupts

The following sections provide details regarding the reset and interrupt operation of the board.

## 5.1 Resets

Reset to the PPC750FX is generated at power-on, by the reset pushbutton, by system-reset from the PPC750FX (usually in response to a command from the RISCWatch debugger), or by undervoltage on the +3.3 V supply.

Under software control, using registers in the CPLD, each processor can be reset individually, or the entire board can be reset.

## 5.2 Interrupts

The system controller contains an interrupt controller that handles interrupts from peripherals inside the system controller as well as external peripherals.

There are three external interrupt inputs to the PPC750FX (INT, MCP, and SMI). See *Table 5-1* for more detail.

*Figure 5-1. Interrupt Architecture*

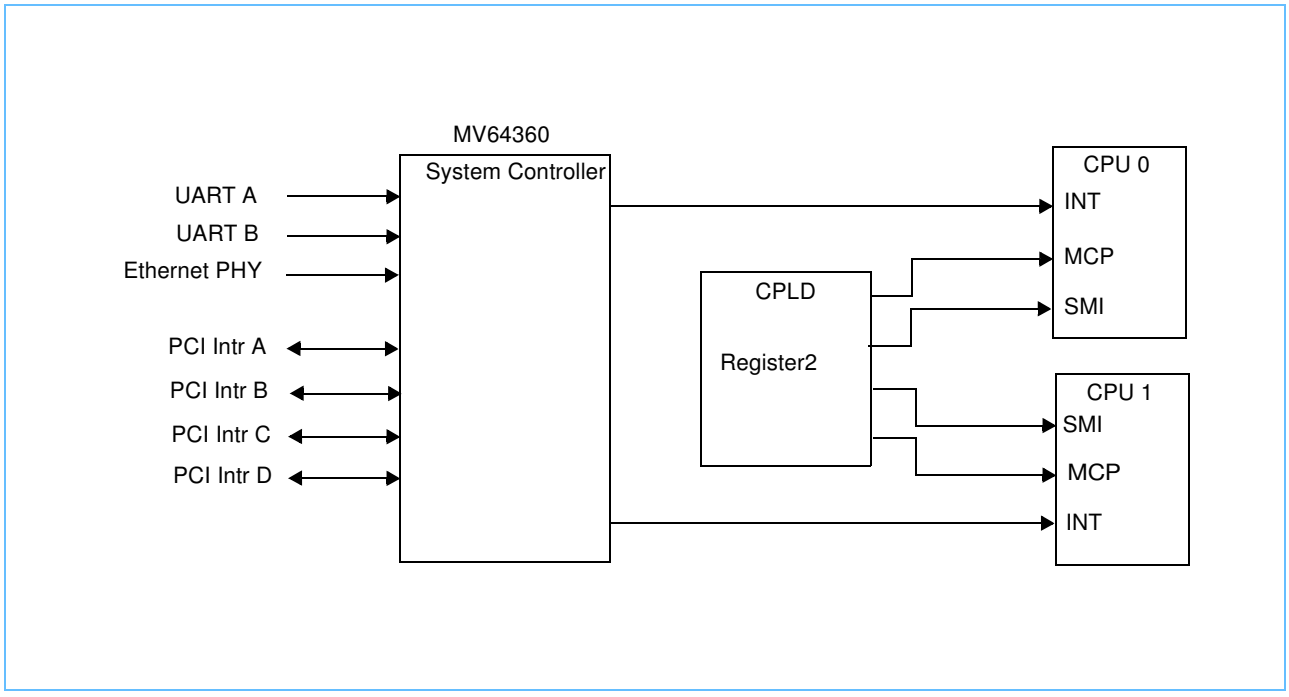


*Table 5-1. External Interrupts*

| MPP Controller Pin | +/- Active | Sensitivity | Description |
|---|---|---|---|
| 25 | + | level | UART Channel A |
| 26 | + | level | UART Channel B |
| 27 | – | level | Ethernet PHY |
| 28 | – | level | PCI Intr A |
| 29 | – | level | PCI Intr B |
| 30 | – | level | PCI Intr C |
| 31 | – | level | PCI Intr D |
| **Note:** The PCI interrupts are inputs when the board is operating as a PCI host. If operating as a PCI adapter, these pins should be configured as outputs. | | | |

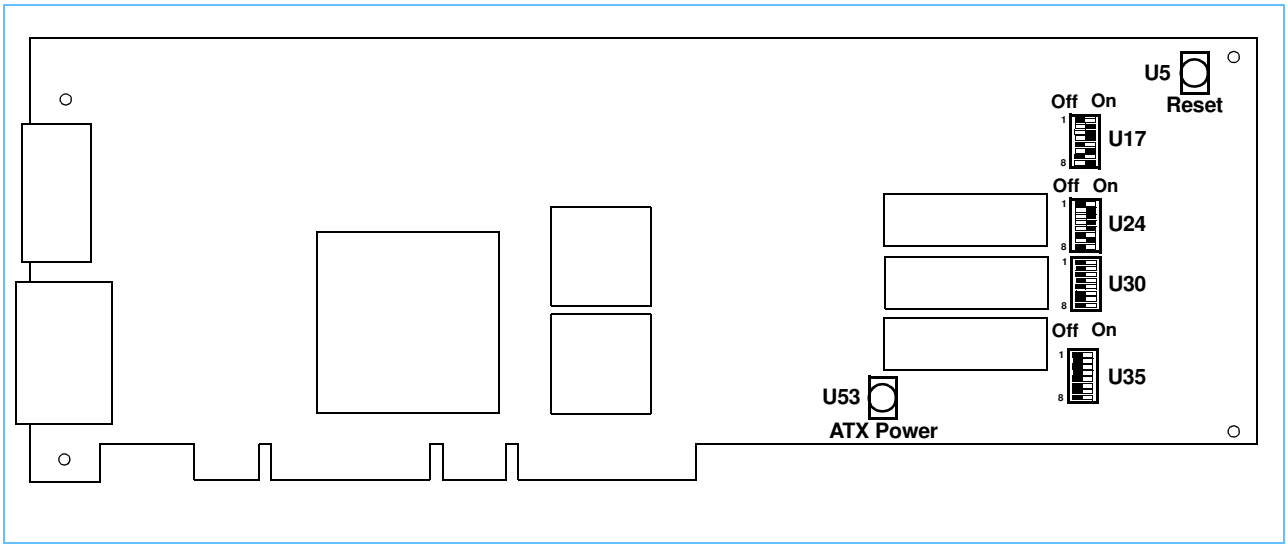
# 6. Switches

This section shows the location of all the switches on the board, and explains the function of each switch.

*Table 6-1. Switches*

| Location | Function | Page |
|---|---|---|
| U5 | Reset pushbutton | 39 |
| U17, U24 | System controller initialization | 42 |
| U30 | CPU 0 PLL configuration | 40 |
| U35 | CPU 1 PLL configuration | 41 |
| U53 | External ATX power | 40 |

*Figure 6-1. Switch Location Diagram*




## 6.1 Reset Pushbutton

When pressed, the pushbutton switch at U5 pulls the PWRGD signal to ground, causing a reset of the board.

*Table 6-2. Reset Pushbutton—U5*

| Signal | Description (0 = ON = close) |
|---|---|
| Pulls down PWRGD. | Main board reset. |

## 6.2 ATX Power-on Pushbutton

When pressed, the pushbutton switch at U53 generates a power-on signal to the external ATX power supply connected to J34. This pushbutton must be pressed *after the ATX power supply is connected to J34* in order to activate the power supply

**Note:** If the board is plugged into a PCI slot, the external ATX power supply will not activate under any conditions, and pressing U53 will have no effect.

*Table 6-3. Reset Pushbutton—U53*

| Signal | Description (0 = ON = close) |
|---|---|
| PS_ON | Generates power-on signal to the external ATX power connector. |

## 6.3 CPU 0 PLL Configuration

An 8-position DIP switch at location U30 configures the PLL for the first PPC750FX processor (U1).

*Table 6-4. CPU 0 PLL Configuration—U30*

| Switch No. | Signal | Default Setting | Description (0 = ON = closed, 1 = OFF = open) |
|---|---|---|---|
| 1 | PLL_CONFIG0 | ON | Refer to the latest version of the *PowerPC 750FX RISC Microprocessor Data Sheet* for details on the bit settings for PLL_CONFIG and PLL_RANGE. |
| 2 | PLL_CONFIG1 | OFF | |
| 3 | PLL_CONFIG2 | ON | |
| 4 | PLL_CONFIG3 | OFF | |
| 5 | PLL_CONFIG4 | OFF | |
| 6 | PLL_RANGE0 | ON | |
| 7 | PLL_RANGE1 | ON | |
| 8 | SPARESWITCH1 | ON | Input to CPLD F2 pin with pull-up to +3.3V. See CPLD Register1. |

## 6.4 CPU 1 PLL Configuration

An 8-position DIP switch at location U35 configures PLL for the second PPC750FX processor (U2).

*Table 6-5. CPU 1 PLL Configuration Switches—U35*

| Switch No. | Signal | Default Setting | Description (0 = ON = closed, 1= OFF = open) |
|---|---|---|---|
| 1 | PLL_CONFIG0 | ON | Refer to the latest version of the *PowerPC 750FX RISC Microprocessor Data Sheet* for details on the bit settings for PLL_CONFIG and PLL_RANGE. |
| 2 | PLL_CONFIG1 | OFF | |
| 3 | PLL_CONFIG2 | ON | |
| 4 | PLL_CONFIG3 | OFF | |
| 5 | PLL_CONFIG4 | OFF | |
| 6 | PLL_RANGE0 | ON | |
| 7 | PLL_RANGE1 | ON | |
| 8 | SPARESWITCH2 | ON | Connected to +3.3V pull-up. See CPLD Register1. |

## 6.5 System Controller Initialization

Two 8-position DIP switches at location U17 and U24 provide initilization settings for the system controller.

*Table 6-6. System Controller Initilization—U17*

| Switch No. | Signal | Default Setting | Description (0 = ON = closed, 1 = OFF = open) |
|---|---|---|---|
| 1 | DEV_AD0 | ON | ON = MV64360 Serial ROM initialization disabled<br>OFF = MV64360 Serial ROM initialization enabled |
| 2:3 | DEV_AD2:DEV_AD3 | ON:ON | Specifies the two least significant bits of the 7 bit IIC address of the Serial ROM the MV64360 can use for initialization.<br>ON:ON = address 0b1010000 (Serial EEPROM U36)<br>ON:OFF = reserved<br>OFF:ON = address 0b1010001 (Serial EEPROM U55)<br>OFF:OFF = reserved |
| 4 | DEV_AD5 | OFF | ON = MV64360 register base address is 0x14000000<br>OFF = MV64360 register base address is 0xF1000000 |
| 5 | DEV_AD8 | OFF | ON = MV64360 CPU Pads Calibration Disabled<br>OFF = MV64360 CPU Pads Calibration Enabled |
| 6 | DEV_AD15 | ON | ON = boot from 8 bit socketed Flash BOOTSMALL_N<br>OFF = boot from 32 bit Flash |
| 7 | DEV_AD16 | OFF | ON = MV64360 PCI retry disabled<br>OFF = MV64360 PCI retry enabled |
| 8 | DEV_AD18 | OFF | ON = DDR-SDRAM clock is running at a higher frequency than the MV64360 core clock<br>OFF = DDR-SDRAM clock is running at the same frequency as the MV64360 core clock |

*Table 6-7. System Controller Initilization—U24*

| Switch No. | Signal | Default Setting | Description (0 = ON = closed, 1 = OFF = open) |
|---|---|---|---|
| 1 | DEV_AD19 | OFF | ON = DDR SDRAM address/control signals toggle on falling edge of DRAM clock.<br>OFF = DDR SDRAM address/control signals toggle on rising edge of DRAM clock. |
| 2 | DEV_AD21 | ON | ON = DDR SDRAM two pipe stages (up to 133MHz SDRAM clock)<br>OFF = DDR SDRAM three pipe stages (up to 183MHz SDRAM clock) |
| 3 | DEV_AD22 | ON | ON = DDR SDRAM read data is synchronized to the MV64360 core clock.<br>OFF = DDR SDRAM read data is synchronized to the MV64360 FBClkIn clock signal. |
| 4 | DEV_AD23 | ON | DDR SDRAM Read Control Logic Delay<br>ON = Disabled<br>OFF = Enabled |
| 5 | DEV_AD24 | ON | DDR SDRAM Read Data Delay<br>ON = Disabled<br>OFF = Enabled |
| 6 | PCIMODE_TARGET/ HOST_N | OFF | ON = board operates as a PCI Host<br>OFF = board operates as a PCI Adapter |
| 7 | FLASH_N/SRAM_SEL | ON | ON = 8-bit Flash resides at a higher address than the 8-bit SRAM<br>OFF = 8-bit SRAM resides at a higher address than the 8-bit Flash<br>**Note:** This switch can be useful for putting SRAM at the CPU reset vector during boot ROM code development. |
| 8 | DEV_AD14 | ON | This switch should always be in the ON position since only 8- or 32-bit wide devices are available for booting. |

# 7. Fuses, Batteries, Regulators, and Fans

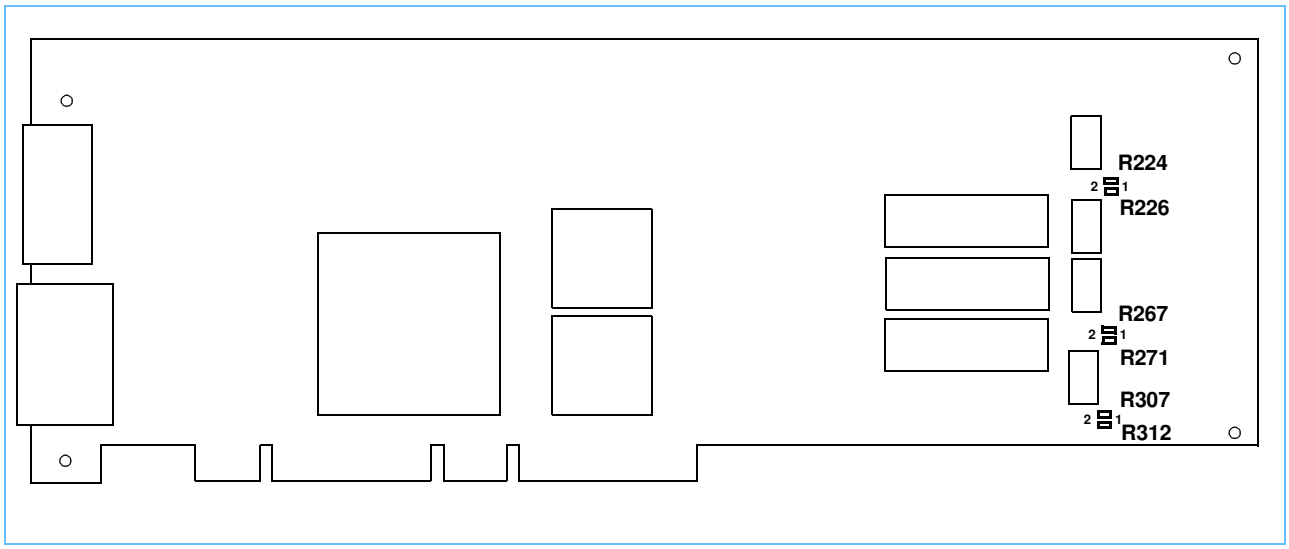
This evaluation board is a PCI card and contains no fuses, batteries, or user-adjustable regulators. All supply voltages needed for the various board components are developed by fixed, board-mounted regulators that use the voltage provided through the PCI edge connector or the ATX power connector. There are three on-board voltages that can be monitored for current drain or replaced with external voltage sources for variable voltage testing.

There is one fan installed on the board that cools the two processor chips.

## 7.1 On-Board Current Monitoring and Variable Voltage Testing

The PPC750FX logic, PLL, and 60X bus voltages can be monitored for current drain. In addition, external supplies can be connected in place of the fixed, on-board regulators to perform variable voltage testing. Removal of zero-ohm resistors, as shown in *Figure 7-1*, is required to implement current measurement or supply substitution.

*Figure 7-1. Resistor Location Diagram*




### 7.1.1 1.45 V Supplies

The PPC750FX uses +1.45 V for the logic and PLL voltages. There are two +1.45 V supplies on the board. They are identified as VCCA1 and VCCA2. A pair of zero-ohm resistors can be removed for either of both supplies to create a connection point or points for current measurement or external supply connection. *Figure 7-1* shows the location of the zero-ohm resistors. Current measurements can be made between terminals 1 and 2 of either resistor. External supplies should be connected to terminal 2.

*Table 7-1. Current Measurement of the1.4 V Supplies*

| Voltage | Remove resistors at: |
|---|---|
| VCCA1 | R267 and R271 |

*Table 7-1. Current Measurement of the1.4 V Supplies*

| Voltage | Remove resistors at: |
|---|---|
| VCCA2 | R307 and R312 |

### 7.1.2 2.5 V Supply

The PPC750FX uses 2.5 V for the 60x bus. There is one 2.5 V supply on the board. It is identified as VCCA3. A pair of zero-ohm resistors can be removed to create a connection point or points for current measurement or external supply connection. *Figure 7-1* shows the location of the zero-ohm resistors. Current measurements can be made between terminals 1 and 2 of either resistor. External supplies should be connected to terminal 2.

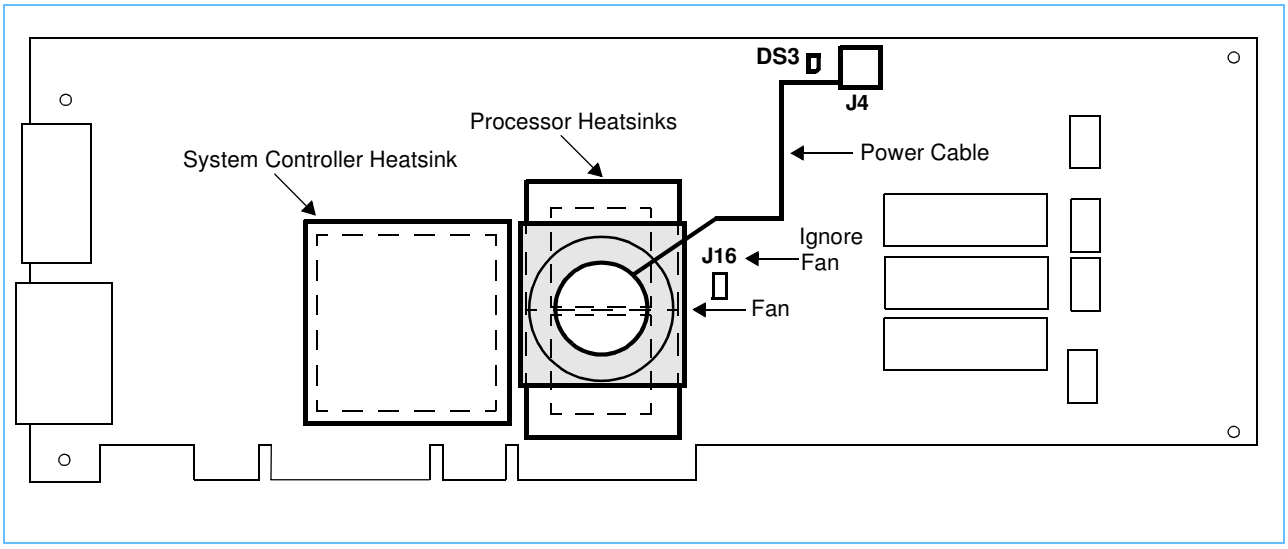
*Table 7-2. Current Measurement of the 2.5 V Supply*

| Voltage | Remove resistors at: |
|---|---|
| VCCA3 | R224 and R226 |

## 7.2 Fan

There is one fan provided on the board. This fan is mounted on the processor heatsinks and cools both of the processor chips. See *Figure 7-2*. The system controller chip has a heatsink attached, but no fan.

*Figure 7-2. Fan and Heatsink Location Diagram*




The processor fan is connected to J4. If the Ignore Fan jumper at J16 is not installed, programming in the CPLD activates LED DS3 when a failure is detected at connector J4, and shuts down the power to the processors and the sytem controller. Disconnecting the fan is detected as a failure.

# 8. Displays

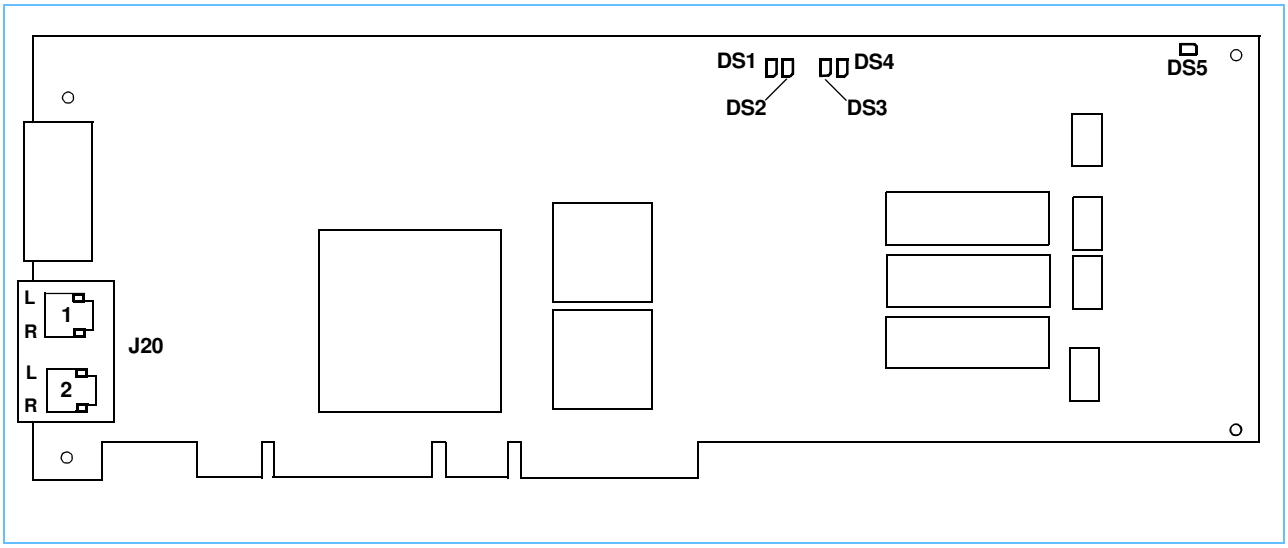
There are nine LED displays on the board. *Table 8-1* identifies the location, color and function of each display. *Figure 8-1* shows the location on the board.

**Note:** There are four Ethernet status LEDs integrated in the two-port Ethernet connector at J20. There are two LEDs per port which are physically located at the corners of each of the two sockets. Each of the two sockets is assigned to one of the Ethernet ports 1 or 2. The location designation L (left) and R (right) applies when the connector is oriented horizontally and viewed looking into the sockets.

*Table 8-1. Displays*

| Location | Name | Color | Description |
|---|---|---|---|
| DS1<br>DS2 | LED0<br>LED1 | Green | CPLD status<br>User programmable by setting bits in a CPLD register. |
| DS3 | RED_LED | Red | Board problem.<br>1. SYSRESET_N signal is active<br>2. Fan failure (Not running or disconnected)<br>3. Jumper J16 installed. |
| DS4 | LED2 | Amber | CPLD status<br>User programmable by setting bits in a CPLD register. |
| DS5 | ATX_OK_LED_N | Green | ATX power<br>Indicates power from external ATX supply connected at J34 is good. |
| J20 2 L | Ethernet Link, Port 1 | Green | Dual LEDs, part of the RJ45 Ethernet 1 port, indicating Ethernet status. |
| J20 2 R | Ethernet Activity, Port 1 | Yellow | |
| J20 1 L | Ethernet Link, Port 2 | Green | Dual LEDs, part of the RJ45 Ethernet 2 port, indicating Ethernet status. |
| J20 1 R | Ethernet Activity, Port 2 | Yellow | |

*Figure 8-1. Display Location Diagram*



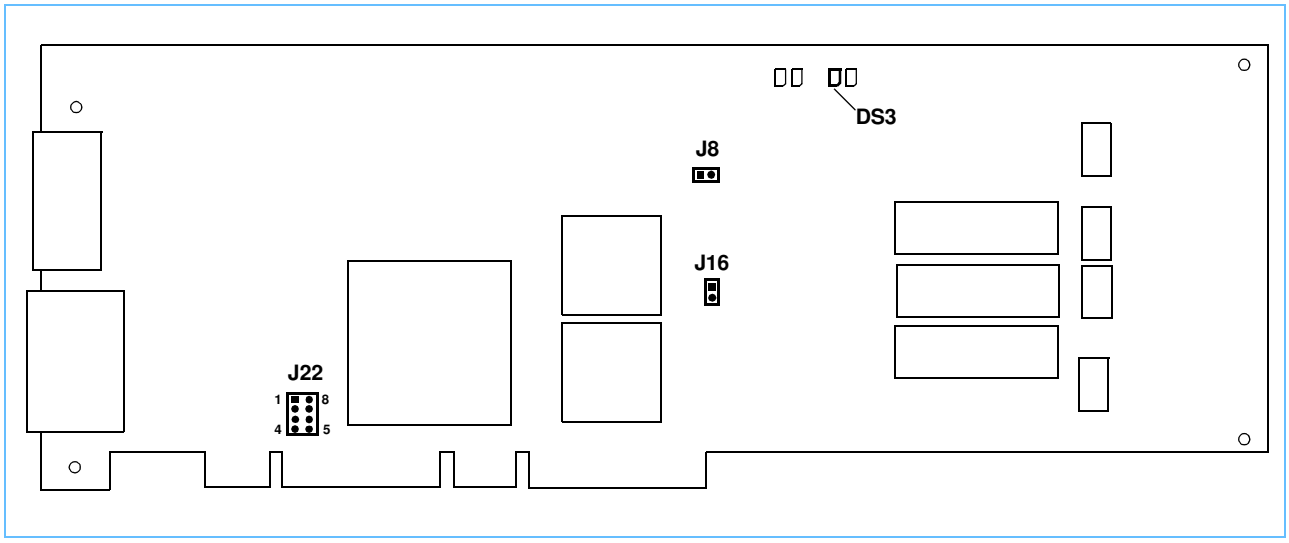
# 9. Jumpers

The location, type, and function for all jumpers on the board are described in the following sections.

*Table 9-1. Jumpers*

| Location | Function | Page |
|---|---|---|
| J8 | 32-bit Flash write protection | 50 |
| J16 | Ignore Fan | 50 |
| J22 | PCI interrupt selection | 51 |
| J27–J30 | Factory test only (not populated) | – |

*Figure 9-1. Jumper Location Diagram*



## 9.1 Write-Protect 32-Bit Flash

32-bit Flash memory can be rendered read-only (write-protected) by this jumper. This is a 1x2 Berg type header.

*Figure 9-2. Write-Protect 32-bit Flash Memory Jumper—J8*


*Table 9-2. Write-Protect 32-bit Flash Memory—J8*

| J8 | Description | Factory Setting |
|---|---|---|
| 1–2 | Write Protected (read-only) | |
| Open | Write Enabled | X |

## 9.2 Ignore Fan

There is one fan on this board that cools both of the PPC750FX processor modules. The connector for the fan power provides a feedback signal to the CPLD indicating a properly functioning fan. When the jumper at J16 is installed, the fan is not monitored. Removing the jumper at J16 allows LED DS3 to be activated if the fan fails or is not connected.

In normal operation, a fan failure causes the power to the processors and system controller to be shut down. The J16 jumper can be used to avoid this power shutdown if the user wishes to continue operation following a fan failure. It is the user's responsibility to ensure that adequate cooling is available to the processors and system controller if the fan is not operating and J16 is installed.

**Note:** If the jumper at J16 is installed, the Check Fan LED is turned on as an indication to the user that the fan is not being monitored.

These are 1x2 Berg type headers.

*Figure 9-3. Ignore Fan Jumper—J16*


*Table 9-3. Ignore Fans—J16*

| J16 | Description | Factory Setting |
|---|---|---|
| 1–2 | Ignore fan | |
| Open | Monitor fan | X |

## 9.3 PCI Interrupt Selection

Jumper J22 configures the adapter mode PCI interrupt output from the system controller to one or more of the four PCI interface interrupts. This a 2x4 Berg type header.

*Figure 9-4. PCI Interrupt Selection Jumper—J22*


*Table 9-4. PCI Interrupt Selection—J22*

| J22 | Description | Factory Setting |
|---|---|---|
| 1–8 | Adapter mode interrupt to PCI interrupt A | X |
| 2–7 | Adapter mode interrupt to PCI interrupt B | |
| 3–6 | Adapter mode interrupt to PCI interrupt C | |
| 4–5 | Adapter mode interrupt to PCI interrupt D | |

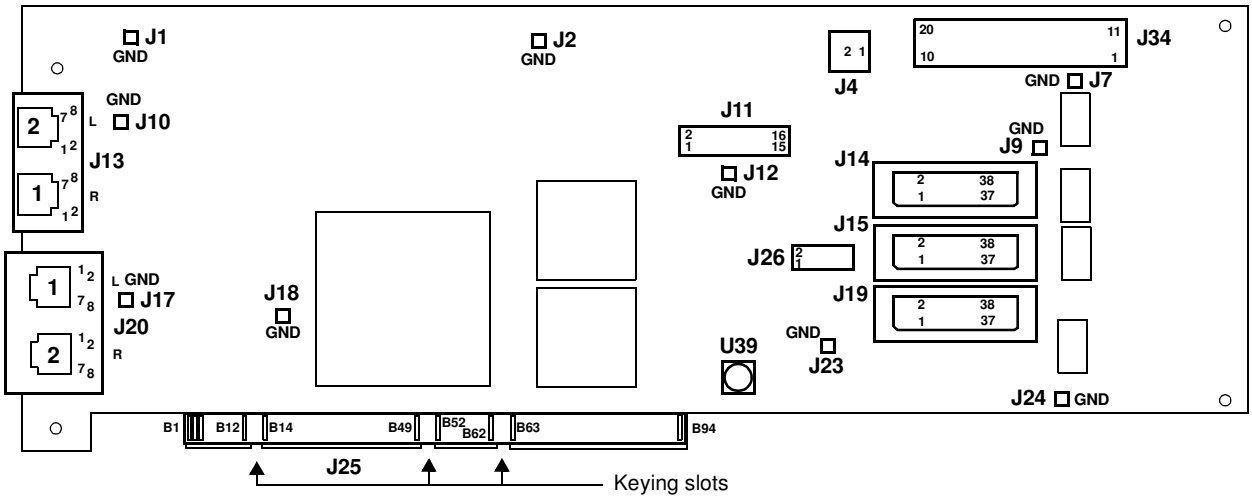
# 10. Connectors

The location, type, function, and pin assignment for all board connections are described in the following sections. Connectors are listed in *Table 10-1* and shown in *Figure 10-1*. Test connections are described in *Section 10.12* and shown in *Figure 10-13*.

*Table 10-1. Connectors*

| Location | Function | Page |
|---|---|---|
| J34 | ATX Power | 55 |
| J4 | CPU Fan Power | 56 |
| J11 | RISCWatch JTAG | 57 |
| J13 | RJ11, Serial Ports 1 (Right) and 2 (Left) | 63 |
| J14 | System Controller Device Address Bus | 64 |
| J15 | Memory Control | 66 |
| J19 | Spare Connector | – |
| J20 | RJ45, Ethernet Ports 1(Left) and 2 (Right) | 58 |
| J25 | PCI Connector | 59 |
| J26 | CPLD JTAG Connector | 62 |
| Jxx | Ground connectors: J1, J2, J7, J9, J10, J12, J17, J18, J23, J24 | 56 |
| J31–J33, J36–J45 | Factory test only (not populated) | – |
| U39 | SMA, External Clock Input | 68 |
| TPxx | Test Connections | 69 |

*Figure 10-1. Connector Location Diagram, Top Side*




**Note:** PCI contacts A1 through A94 mirror the B1-B94 contacts on the bottom side of the card.

## 10.1 Auxiliary Power

The board is equipped with a standard ATX power connector. This allows the board to be externally powered from a standard ATX power supply when it is not installed in a PCI slot.

**Note:** If the board is plugged into a PCI slot, the external ATX power supply will not activate under any conditions, and pressing the ATX power-on pushbutton (U53) will have no effect.

*Figure 10-2. ATX Power Supply Connector—J34*

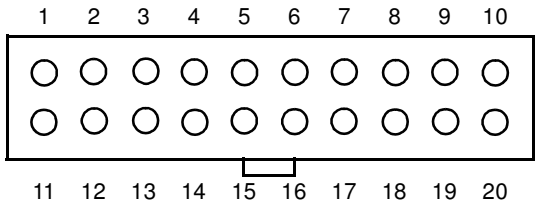



*Table 10-2. ATX Power Signals—J34*

| Pin | Name | Comment |
|---|---|---|
| 1 | +3.3V | Tolerance ± 4% |
| 2 | +3.3V | Tolerance ± 4% |
| 3 | GND | |
| 4 | +5V | Tolerance ± 5% |
| 5 | GND | |
| 6 | +5V | Tolerance ± 5% |
| 7 | GND | |
| 8 | Power OK | Active high indicator that +5V and +3.3V are above their undervoltage thresholds |
| 9 | +5V SB | Standby power, at least 10 mA, tolerance ± 5% |
| 10 | +12V | Not used |
| 11 | +3.3V | Tolerance ± 4%. Remote 3.3V sense. |
| 12 | -12V | Not used |
| 13 | GND | |
| 14 | PS-ON | Active low signal that turns on the ATX power supply. |
| 15 | GND | |
| 16 | GND | |
| 17 | GND | |
| 18 | -5V | Not used |
| 19 | +5V | Tolerance ± 5% |
| 20 | +5V | Tolerance ± 5% |

## 10.2 Ground

Test points for grounding logic analyzers and other test equipment are available on these connectors. These are 1x1 Berg type connectors.

*Figure 10-3. Ground Connectors—J1, J2, J7, J9, J10, J12, J17, J18, J23, J24*


*Table 10-3. Ground Connectors —J1, J2, J7, J9, J10, J12, J17, J18, J23, J24*

| Pin | Signal Name |
|---|---|
| 1 | GND |

## 10.3 Fan Power

There is one connector available at J4 for powering the fan that cools the two processor chips.

*Figure 10-4. Fan Power Connector—J4*

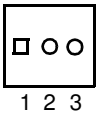



*Table 10-4. Fan Power Signals—J4,*

| Pin | Signal Name |
|---|---|
| 1 | GND |
| 2 | +5V |
| 3 | Fan feedback<br>**Note:** A low (near 0V) feedback signal from the fan is an indication that the fan is running. |

## 10.4 RISCWatch JTAG Debugger

The RISCWatch JTAG debugger connects to the board through a 2x8 header.

*Figure 10-5. RISCWatch JTAG Connector—J11*

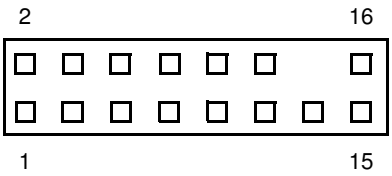



*Table 10-5. RISCWatch Signals—J11*

| Pin | Signal Name |
|---|---|
| 1 | TDO |
| 2 | unused |
| 3 | TDI |
| 4 | TRST_N |
| 5 | unused |
| 6 | PWRSENSE |
| 7 | TCK |
| 8 | unused |
| 9 | TMS |
| 10 | unused |
| 11 | SRESET_N |
| 12 | unused |
| 13 | HRESET_N |
| 14 | Key—no pin at this location. |
| 15 | CHECKSTOP_N |
| 16 | GND |

## 10.5 Ethernet

This board provides two Ethernet ports. The connections are through a single housing at J20 that contains two RJ45 connectors. Each connector contains integral magnetics and two LEDs. The ports are identified as 1 and 2. Both ports can be configured for Fast (10/100 Mbps) Ethernet interfaces, and can be used with Category 5 Unshielded Twisted-Pair (UTP) cable.

*Figure 10-6. Ethernet Connector—One of two RJ45 Sockets in J20*

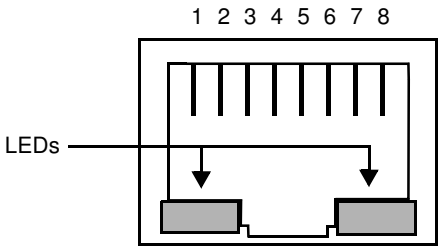


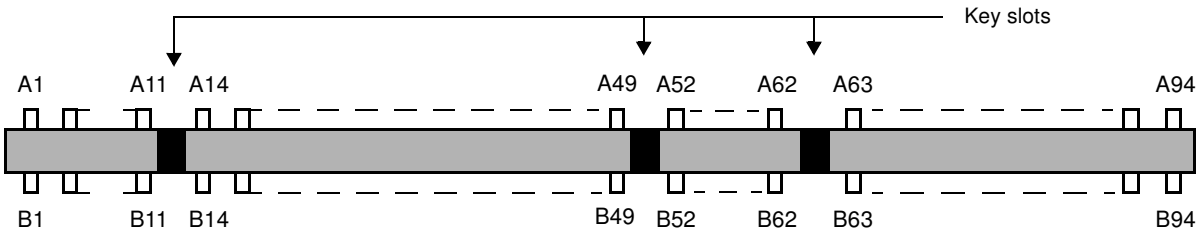

*Table 10-6. Ethernet UTP Signals—J20, both sockets*

| Pin | Signal Name | Description |
|---|---|---|
| 1 | RD+ | Receive data + |
| 2 | RD– | Receive data – |
| 3 | TD+ | Transmit data + |
| 4 | RCT | Receive center tap |
| 5 | TCT | Transmit center tap |
| 6 | TD– | Transmit data – |
| 7 | NC | No connection |
| 8 | GND | Ground |

## 10.6 PCI Connector

This evaluation board is a PCI card. It has a standard PCI connector that plugs into a standard +3.3V or +5V PCI socket on a PC system board. The signals on the PCI conector are the standard set of PCI signals.

*Figure 10-7. PCI Connector—J25*




**Note:** This view of the connector is from the top edge of the card.

*Table 10-7. PCI Connector Signals—J25*

| Pin | Signal | Pin | Signal |
|---|---|---|---|
| B1 | -12V | A1 | TRST |
| B2 | TCK | A2 | +12V |
| B3 | GND | A3 | TMS |
| B4 | TDO | A4 | TDI |
| B5 | +5V | A5 | +5V |
| B6 | +5V | A6 | INTA |
| B7 | INTB | A7 | INTC |
| B8 | INTD | A8 | +5V |
| B9 | PRSNT1 | A9 | Reserved |
| B10 | Reserved | A10 | +3.3V (I/O) |
| B11 | PRSNT2 | A11 | Reserved |
| – | Key slot | – | Key slot |
| – | | – | |
| B14 | Reserved | A14 | +3.3V (Aux) |
| B15 | GND | A15 | RST |
| B16 | CLK | A16 | +3.3V (I/O) |
| B17 | GND | A17 | GNT |
| B18 | REQ | A18 | GND |
| B19 | +3.3V (I/O) | A19 | Reserved |
| B20 | AD31 | A20 | AD30 |
| B21 | AD29 | A21 | +3.3V |

*Table 10-7. PCI Connector Signals—J25 (Continued)*

| Pin | Signal | Pin | Signal |
|---|---|---|---|
| B22 | GND | A22 | AD28 |
| B23 | AD27 | A23 | AD26 |
| B24 | AD25 | A24 | GND |
| B25 | +3.3V | A25 | AD24 |
| B26 | C/BE3 | A26 | IDSEL |
| B27 | AD23 | A27 | +3.3V |
| B28 | GND | A28 | AD22 |
| B29 | AD21 | A29 | AD20 |
| B30 | AD19 | A30 | GND |
| B31 | +3.3V | A31 | AD18 |
| B32 | AD17 | A32 | AD16 |
| B33 | C/BE2 | A33 | +3.3V |
| B34 | GND | A34 | FRAME |
| B35 | IRDY | A35 | GND |
| B36 | +3.3V | A36 | TRDY |
| B37 | DEVSEL | A37 | GND |
| B38 | GND | A38 | STOP |
| B39 | LOCK | A39 | +3.3V |
| B40 | PERR | A40 | Reserved |
| B41 | +3.3V | A41 | Reserved |
| B42 | SERR | A42 | GND |
| B43 | +3.3V | A43 | PAR |
| B44 | C/BE1 | A44 | AD15 |
| B45 | AD14 | A45 | +3.3V |
| B46 | GND | A46 | AD13 |
| B47 | AD12 | A47 | AD11 |
| B48 | AD10 | A48 | GND |
| B49 | M66EN/GND | A49 | AD9 |
| – | Key slot | – | Key slot |
| – | | – | |
| B52 | AD8 | A52 | C/BE0 |
| B53 | AD7 | A53 | +3.3V |
| B54 | +3.3V | A54 | AD6 |
| B55 | AD5 | A55 | AD4 |
| B56 | AD3 | A56 | GND |
| B57 | GND | A57 | AD2 |
| B58 | AD1 | A58 | AD0 |

*Table 10-7. PCI Connector Signals—J25 (Continued)*

| Pin | Signal | Pin | Signal |
|---|---|---|---|
| B59 | +3.3V (I/O) | A59 | +3.3V (I/O) |
| B60 | ACK64 | A60 | REQ64 |
| B61 | +5V | A61 | +5V |
| B62 | +5V | A62 | +5V |
| – | Key slot | – | Key slot |
| – | | – | |
| B63 | Reserved | A63 | GND |
| B64 | GND | A64 | C/BE7 |
| B65 | C/BE6 | A65 | C/BE5 |
| B66 | C/BE4 | A66 | +3.3V(I/O) |
| B67 | GND | A67 | PAR64 |
| B68 | AD63 | A68 | AD62 |
| B69 | AD61 | A69 | GND |
| B70 | +3.3V(I/O) | A70 | AD[60] |
| B71 | AD[59] | A71 | AD58 |
| B72 | AD57 | A72 | GND |
| B73 | GND | A73 | AD56 |
| B74 | AD55 | A74 | AD54 |
| B75 | AD53 | A75 | +3.3V(I/O) |
| B76 | GND | A76 | AD52 |
| B77 | AD51 | A77 | AD50 |
| B78 | AD49 | A78 | GND |
| B79 | +3.3V(I/O) | A79 | AD48 |
| B80 | AD47 | A80 | AD46 |
| B81 | AD45 | A81 | GND |
| B82 | GND | A82 | AD44 |
| B83 | AD43 | A83 | AD42 |
| B84 | AD41 | A84 | +3.3V(I/O) |
| B85 | GND | A85 | AD40 |
| B86 | AD39 | A86 | AD38 |
| B87 | AD37 | A87 | GND |
| B88 | +3.3V(I/O) | A88 | AD36 |
| B89 | AD35 | A89 | AD34 |
| B90 | AD33 | A90 | GND |
| B91 | GND | A91 | AD32 |

*Table 10-7. PCI Connector Signals—J25 (Continued)*

| Pin | Signal | Pin | Signal |
|---|---|---|---|
| B92 | Reserved | A92 | Reserved |
| B93 | Reserved | A93 | GND |
| B94 | GND | A94 | Reserved |

## 10.7 CPLD JTAG Connector

The CPLD may be programmed in place on the board via this JTAG connector and appropriate downloading software. This is a 2x5 Berg type connector.

*Figure 10-8. CPLD JTAG Connector—J26*

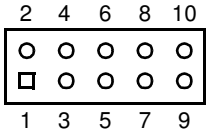



*Table 10-8. CPLD JTAG Connector—J26*

| Pin | Signal Name |
|---|---|
| 1 | ISP_TCK |
| 2 | GND |
| 3 | ISP_TDO |
| 4 | +3.3V |
| 5 | ISP_TMS |
| 6 | unused |
| 7 | unused |
| 8 | unused |
| 9 | ISP_TDI |
| 10 | GND |

## 10.8 Serial Ports

Serial Port 1 (J13 Right) and Serial Port 2 (J13 Left) are provided through standard RJ11/12 connectors, as shown in *Figure 10-9*. Both serial port interfaces are provided by the ST16C2552 attached to the system controller and support only four RS-232 signals.

*Table 10-9* describes the pin assignments for Serial Ports 1 and 2. Note that DTR appears on both pins 2 and 7.

*Figure 10-9. Serial Port Connector—J13, one of two RJ11/12 sockets*

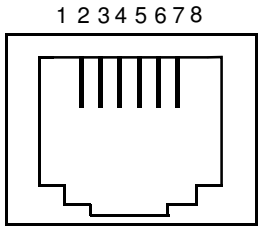



*Table 10-9. Serial Port Connector Signals—J13, both ports*

| Pin | Signal Name |
|---|---|
| 1 | Empty contact position |
| 2 | DTR |
| 3 | DSR |
| 4 | Rx |
| 5 | Frame Ground |
| 6 | Tx |
| 7 | DTR |
| 8 | Empty contact positiont |

## 10.9 System Controller Device Address Bus

Connection to HP logic analyzers via HP E5346A High Density Probe Adapters is provided by board mounted connectors, Mictor Part Number 2-767004-2. This connector provides user access to the system controller peripheral address bus for test and debug purposes.

*Figure 10-10. System Controller Device Address Connector—J14*

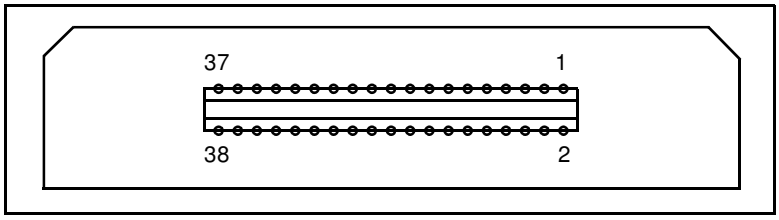



*Table 10-10. System Controller Device Address Signals—J14*

| Pin | Analyzer | Signal Name |
|---|---|---|
| 1 | | unused |
| 2 | | unused |
| 3 | | GND |
| 4 | | unused |
| 5 | addr_pod0 – CLK | ALE |
| 6 | addr_pod1 – CLK | unused |
| 7 | addr_pod0 – D15 | DEV_ADR(31) |
| 8 | addr_pod1 – D15 | DEV_ADR(15) |
| 9 | addr_pod0 – D14 | DEV_ADR(30) |
| 10 | addr_pod1 – D14 | DEV_ADR(14) |
| 11 | addr_pod0 – D13 | DEV_ADR(29) |
| 12 | addr_pod1 – D13 | DEV_ADR(13) |
| 13 | addr_pod0 – D12 | DEV_ADR(28) |
| 14 | addr_pod1 – D12 | DEV_ADR(12) |
| 15 | addr_pod0 – D11 | DEV_ADR(27) |
| 16 | addr_pod1 – D11 | DEV_ADR(11) |
| 17 | addr_pod0 – D10 | DEV_ADR(26) |
| 18 | addr_pod1 – D10 | DEV_ADR(10) |
| 19 | addr_pod0 – D9 | DEV_ADR(25) |
| 20 | addr_pod1 – D9 | DEV_ADR(9) |
| 21 | addr_pod0 – D8 | DEV_ADR(24) |
| 22 | addr_pod1 – D8 | DEV_ADR(8) |
| 23 | addr_pod0 – D7 | DEV_ADR(23) |

*Table 10-10. System Controller Device Address Signals—J14 (Continued)*

| Pin | Analyzer | Signal Name |
|---|---|---|
| 24 | addr_pod1 – D7 | DEV_ADR(7) |
| 25 | addr_pod0 – D6 | DEV_ADR(22) |
| 26 | addr_pod1 – D6 | DEV_ADR(6) |
| 27 | addr_pod0 – D5 | DEV_ADR(21) |
| 28 | addr_pod1 – D5 | DEV_ADR(5) |
| 29 | addr_pod0 – D4 | DEV_ADR(20) |
| 30 | addr_pod1 – D4 | DEV_ADR(4) |
| 31 | addr_pod0 – D3 | DEV_ADR(19) |
| 32 | addr_pod1 – D3 | DEV_ADR(3) |
| 33 | addr_pod0 – D2 | DEV_ADR(18) |
| 34 | addr_pod1 – D2 | DEV_ADR(2) |
| 35 | addr_pod0 – D1 | DEV_ADR(17) |
| 36 | addr_pod1 – D1 | DEV_ADR(1) |
| 37 | addr_pod0 – D0 | DEV_ADR(16) |
| 38 | addr_pod1 – D0 | DEV_ADR(0) |

## 10.10 Memory Control

Connection to HP logic analyzers via HP E5346A High Density Probe Adapters is provided by board mounted connectors, Mictor Part Number 2-767004-2. This connector carries the burst address bus and the chip select signals from the system controller.

*Figure 10-11. Memory Control Connector—J15*

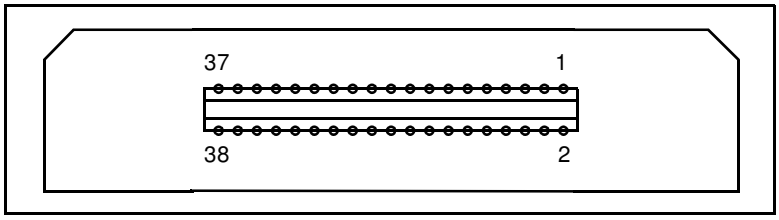



*Table 10-11. Memory Control Signals—J15*

| Pin | Analyzer | Signal Name |
|---|---|---|
| 1 | | unused |
| 2 | | unused |
| 3 | | GND |
| 4 | | unused |
| 5 | cntl_pod0 – CLK | unused |
| 6 | cntl_pod1 – CLK | unused |
| 7 | cntl_pod0 – D15 | BADR(2) |
| 8 | cntl_pod1 – D15 | TESTPIN_A |
| 9 | cntl_pod0 – D14 | BADR(1) |
| 10 | cntl_pod1 – D14 | SRAM_LO_CS_N |
| 11 | cntl_pod0 – D13 | BADR(0) |
| 12 | cntl_pod1 – D13 | unused |
| 13 | cntl_pod0 – D12 | SMALL_FLASH_LO_CS |
| 14 | cntl_pod1 – D12 | unused |
| 15 | cntl_pod0 – D11 | DEV_WE(3) |
| 16 | cntl_pod1 – D11 | unused |
| 17 | cntl_pod0 – D10 | DEV_WE(2) |
| 18 | cntl_pod1 – D10 | SYSRESET_N |
| 19 | cntl_pod0 – D9 | DEV_WE(1) |
| 20 | cntl_pod1 – D9 | CPU1_HRESET_2.5_N |
| 21 | cntl_pod0 – D8 | DEV_WE(0) |
| 22 | cntl_pod1 – D8 | CPU1_SRESET_2.5_N |
| 23 | cntl_pod0 – D7 | CS_TIMING_N |

*Table 10-11. Memory Control Signals—J15 (Continued)*

| Pin | Analyzer | Signal Name |
|---|---|---|
| 24 | cntl_pod1 – D7 | CPU0_HRESET_2.5_N |
| 25 | cntl_pod0 – D6 | NVRAM_CS_N |
| 26 | cntl_pod1 – D6 | CPU0_SRESET_2.5_N |
| 27 | cntl_pod0 – D5 | BIG_FLASH_CS_N |
| 28 | cntl_pod1 – D5 | unused |
| 29 | cntl_pod0 – D4 | SMALL_FLASH_HI_CS_N |
| 30 | cntl_pod1 – D4 | TESTPIN_D |
| 31 | cntl_pod0 – D3 | SRAM_HI_CS_N |
| 32 | cntl_pod1 – D3 | TESTPIN_B |
| 33 | cntl_pod0 – D2 | READ_N |
| 34 | cntl_pod1 – D2 | unused |
| 35 | cntl_pod0 – D1 | WRITE_N |
| 36 | cntl_pod1 – D1 | FPGA_CS_N |
| 37 | cntl_pod0 – D0 | UART_CS_N |
| 38 | cntl_pod1 – D0 | TESTPIN_C |

## 10.11 External Clock Input

An external 133 MHz board clock may be provided by an external oscillator connected to this board-mounted SMA connector. The oscillator output should have 3.3 V logic levels. The input impedance to this connector is approximately 50 Ω.

**Note:** Board rework is required to use this connector. See the schematic diagrams.

*Figure 10-12. External Clock Input Connector—U39*

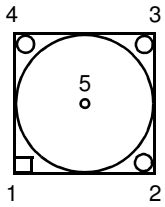



*Table 10-12. External Clock Input Signal—U39*

| Pin | Signal Name |
|---|---|
| 1 | Ground |
| 2 | Ground |
| 3 | Ground |
| 4 | Ground |
| 5 | CLK_EXT |

## 10.12 Test Connections

Access to selected points in the board circuits, not available through connectors or jumpers, is provided by small test connections. All of the test connections are small pads with a center hole. An electrical connection can be made to any of these test connections for test purposes. These test connections are documented in the schematic, in *Table 10-13*, and in *Figure 10-13*.

There are two kinds of test connections. The basic difference between the two is the size of the pad and the center hole. One type of test connection is called a *test point* and has a large pad and center hole. Test points are suitable for soldering wires to them for test and design purposes. The reference designators for test points are TPnn where nn are numbers. The other type of test connection is called an *ehole* or *test access via*. These test connections have a very small pad and center hole. The reference designators for eholes are Unn where nn are numbers.

To determine what signal the test connection connects to, refer to the schematic. In *Figure 10-13*, the test connections are labeled with their reference designators only. The actual pad and hole is not shown. Other major components on the board are shown for assistance in locating the test connections.

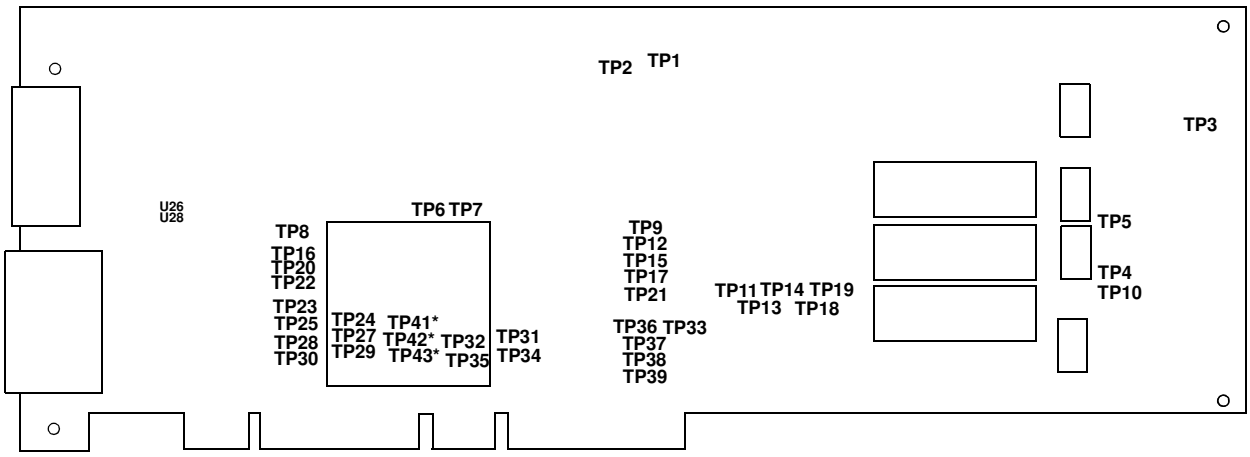
*Table 10-13. Test Connections*

| Location | Component | Signal |
|---|---|---|
| TP1 | U20 | LBOOT_CS_N |
| TP2 | U20 | LDEV_RW_N |
| TP3 | U18 +2.5V | VADJ |
| TP4 | U50 | P07 |
| TP5 | U50 | INT_N |
| TP6 | U3 System Controller | CS2_N |
| TP7 | U3 System Controller | CS3_N |
| TP8 | U3 System Controller | JTD0 |
| TP9 | U1 750FX | QREQ0 |
| TP10 | U50 | P17 |
| TP11 | U3 System Controller | MPP13 |
| TP12 | U1 750FX | WTI |
| TP13 | U3 System Controller | MPP16 |
| TP14 | U3 System Controller | MPP15 |
| TP15 | U1 750FX | CI |
| TP16 | U3 System Controller | DEVDP0 |
| TP17 | U1 750FX | RSRV0 |
| TP18 | U3 System Controller | MPP18 |
| TP19 | U3 System Controller | MPP17 |
| TP20 | U3 System Controller | DEVDP1 |
| TP21 | U1 750FX | CLK_OUT0 |
| TP22 | U3 System Controller | DEVDP3 |

*Table 10-13. Test Connections (Continued)*

| Location | Component | Signal |
|---|---|---|
| TP23 | U3 System Controller | MPP3 |
| TP24 | U3 System Controller | MPP2 |
| TP25 | U3 System Controller | MPP1 |
| TP27 | U3 System Controller | MPP6 |
| TP28 | U3 System Controller | MPP4 |
| TP29 | U3 System Controller | MPP5 |
| TP30 | U3 System Controller | MPP14 |
| TP31 | U3 System Controller | MPP27 |
| TP32 | U3 System Controller | MPP20 |
| TP33 | U2 750FX | QREQ1 |
| TP34 | U3 System Controller | MPP22 |
| TP35 | U3 System Controller | ENUM_N |
| TP36 | U2 750FX | WT |
| TP37 | U2 750FX | RSRV1 |
| TP38 | U2 750FX | CLK_OUT1 |
| TP39 | U2 750FX | T4 |
| TP41 | U3 System Controller | MPP0 |
| TP42 | U3 System Controller | MPP9 |
| TP43 | U3 System Controller | MPP10 |
| U26 | U27 | VREF_IN |
| U28 | U27 | SHDN_N |

*Figure 10-13. Test Connection Locations*




* TP41, TP42, and TP43 are located on the bottom of the board

# 11. CPLD Programming

This chapter contains logic of the CPLD (formerly FPGA) module and timing values for the signals associated with the CPLD.

*Table 11-1. Section Contents*



## 11.1 Programming—Registers and Control Functions

General software functions such as address decoding and board status registers are provided by the CPLD at U29. The following figures provide a combined graphical and textual representation of the functional programming within the CPLD.

### 11.1.1 I/O Pin List

*Table 11-2* lists all of the signals entering and exiting the CPLD module. The module pin number and I/O function are shown.

**Note:** Pins that are labeled as ~NAME~ are special purpose pins defined by the CPLD to program functions or set reference voltages.

*Table 11-2. CPLD I/O Pin List*

| Name | Pin | Function |
|---|---|---|
| cpu0_chkstop_n | 1 | Input |
| ale | 2 | Input |
| VCCIO1 | 3 | Power |
| ~TDI~ | 4 | Input |
| rw_trst | 5 | Input |
| sram_hi_cs_n | 6 | Output |
| small_flash_hi_cs_n | 7 | Output |
| badr[1] | 8 | Input |
| led0 | 9 | Output |
| led2 | 10 | Output |
| GND | 11 | Gnd |
| ~VREFA~ | 12 | Input |
| fpga_cs_n | 13 | Output |
| pci_reset_n | 14 | Bidir |

*Table 11-2. CPLD I/O Pin List (Continued)*

| Name | Pin | Function |
|---|---|---|
| ~TMS~ | 15 | Input |
| write_n | 16 | Output |
| uart_cs_n | 17 | Output |
| VCCIO1 | 18 | Power |
| nvram_cs_n | 19 | Output |
| led1 | 20 | Output |
| led_red_n | 21 | Output |
| ignore_fans_n | 22 | Input |
| mpp0_sreset_n | 23 | Input |
| cpu1_chkstop_n | 24 | Input |
| sysreset | 25 | Output |
| GND | 26 | Gnd |
| dev_adr[5] | 27 | Bidir |
| dev_adr[6] | 28 | Bidir |
| dev_adr[7] | 29 | Bidir |
| sysreset_n | 30 | Output |
| mpp1_hreset_n | 31 | Input |
| NOFAN_N | 32 | Output |
| sram_cs_n | 33 | Output |
| VCCIO1 | 34 | Power |
| small_flash_cs_n | 35 | Output |
| read_n | 36 | Output |
| big_flash_cs_n | 37 | Output |
| GND | 38 | Gnd |
| VCCINT | 39 | Power |
| badr[0] | 40 | Input |
| initact | 41 | Input |
| flash_n/sram_sel | 42 | Input |
| GND | 43 | Gnd |
| lcs_n[2] | 44 | Input |
| pwrgd | 45 | Input |
| switch_a | 46 | Input |
| testpin_c | 47 | Output |
| testpin_b | 48 | Output |
| testpin_a | 49 | Output |
| textpin_d | 50 | Output |
| VCCIO2 | 51 | Power |

*Table 11-2. CPLD I/O Pin List (Continued)*

| Name | Pin | Function |
|---|---|---|
| cpu0_hreset_n | 52 | Output |
| cpu_trst_n | 53 | Output |
| cpu1_sreset_n | 54 | Output |
| cpu1_hreset_n | 55 | Output |
| cpu0_sreset_n | 56 | Output |
| lcs_n[1] | 57 | Input |
| mpp_reset_out_n | 58 | Input |
| GND | 59 | Gnd |
| ~VREFB~ | 60 | Input |
| atx_ok_n | 61 | Input |
| ~TCK~ | 62 | Input |
| dev_we_n[0] | 63 | Input |
| lcs_n[0] | 64 | Input |
| bootsmall_n | 65 | Input |
| VCCIO2 | 66 | Power |
| badr[2] | 67 | Input |
| rw_sreset | 68 | Input |
| jtag_chkstop_n | 69 | Output |
| cpu_tben | 70 | Output |
| cpu1_smi_n | 71 | Output |
| cpu0_smi_n | 72 | Output |
| ~TDO~ | 73 | Output |
| GND | 74 | Gnd |
| lcs_n[3] | 75 | Input |
| mpp0_hreset_n | 76 | Input |
| ldev_addr[21] | 77 | Input |
| cstiming_n | 78 | Input |
| unused_pin | 79 | Output |
| cpu_mcp1 | 80 | Output |
| cpu_mcp0 | 81 | Output |
| VCCIO2 | 82 | Power |
| mpp1_sreset_n | 83 | Input |
| rw_hreset | 84 | Input |
| switch_b | 85 | Input |
| GND | 86 | Gnd |
| pld_sysclk | 87 | Input |
| target/host_n | 88 | Input |

*Table 11-2. CPLD I/O Pin List (Continued)*

| Name | Pin | Function |
|---|---|---|
| GND+ | 89 | |
| pld25mhz | 90 | Input |
| VCCINT | 91 | Power |
| dev_adr[0] | 92 | Bidir |
| dev_adr[1] | 93 | Bidir |
| dev_adr[2] | 94 | Bidir |
| GND | 95 | Gnd |
| dev_adr[3] | 96 | Bidir |
| dev_adr[4] | 97 | Bidir |
| ldev_addr[20] | 98 | Input |
| cpufan_ok_n | 99 | Input |
| ldev_addr[19] | 100 | Input |

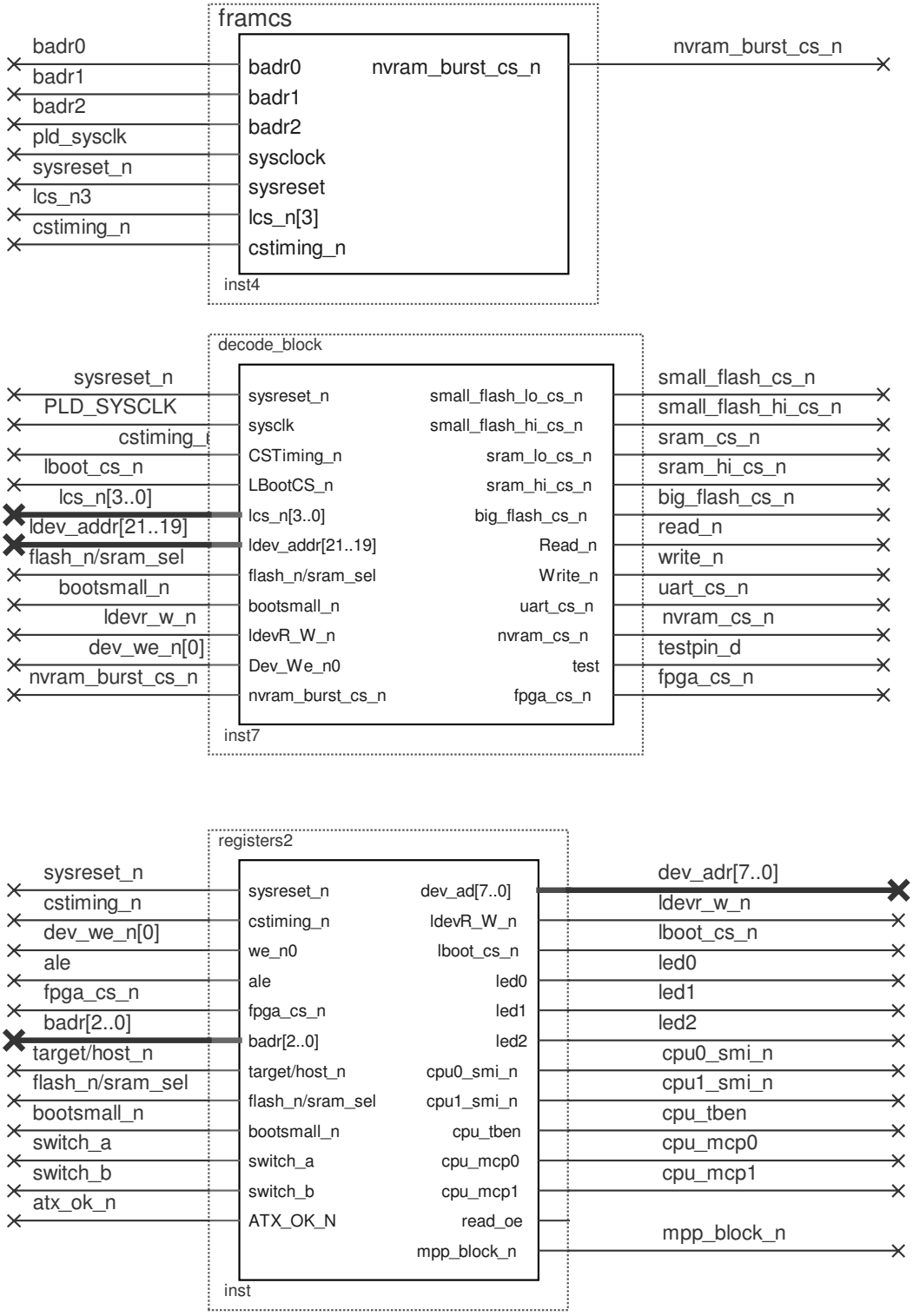
### 11.1.2 CPLD Logic

The folllowing sections provided a representation of the logic of the complete CPLD in either graphical or code listing format.

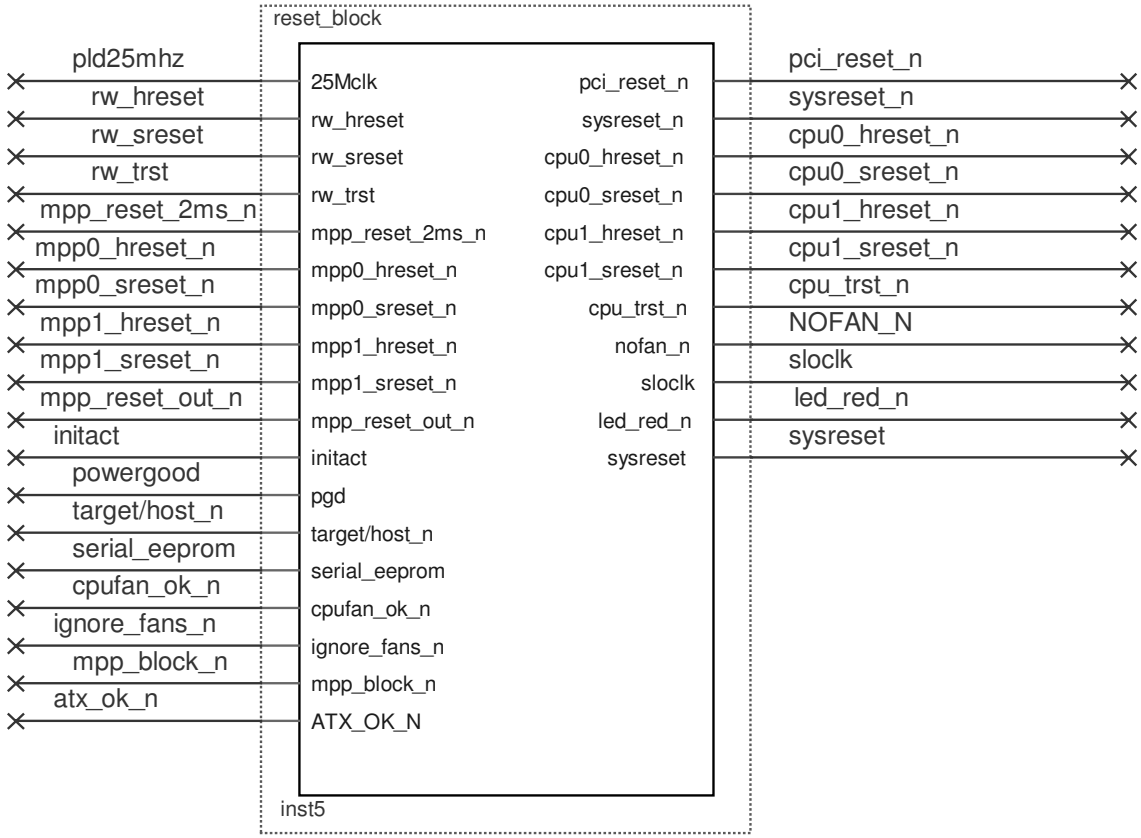
*Table 11-3. CPLD Logic Descriptions*

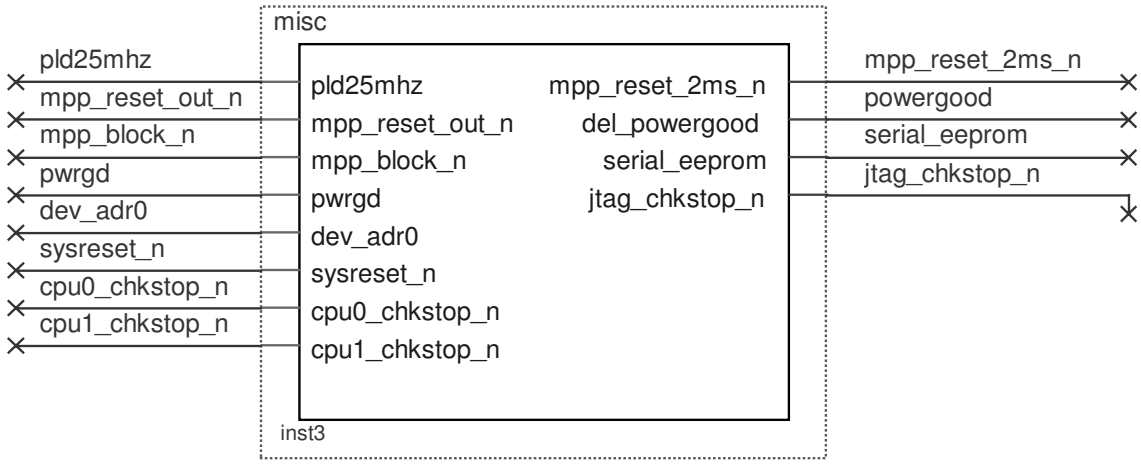
| Section | Description | Page |
|---|---|---|
| *Top Level Block Diagram 1* | First three of five CPLD logic sections | 77 |
| *Top Level Block Diagram 2* | Last two of five CPLD logic sections | 78 |
| *framcs Logic* | framcs detailed logic diagram | 79 |
| *decode_block Program* | decode_block code listing | 80 |
| *registers2 Program* | register2 code listing | 84 |
| *reset_block Program* | reset_block code listing | 89 |
| *misc Logic* | misc detailed logic diagram | 93 |

### *11.1.2.1 Top Level Block Diagram 1*



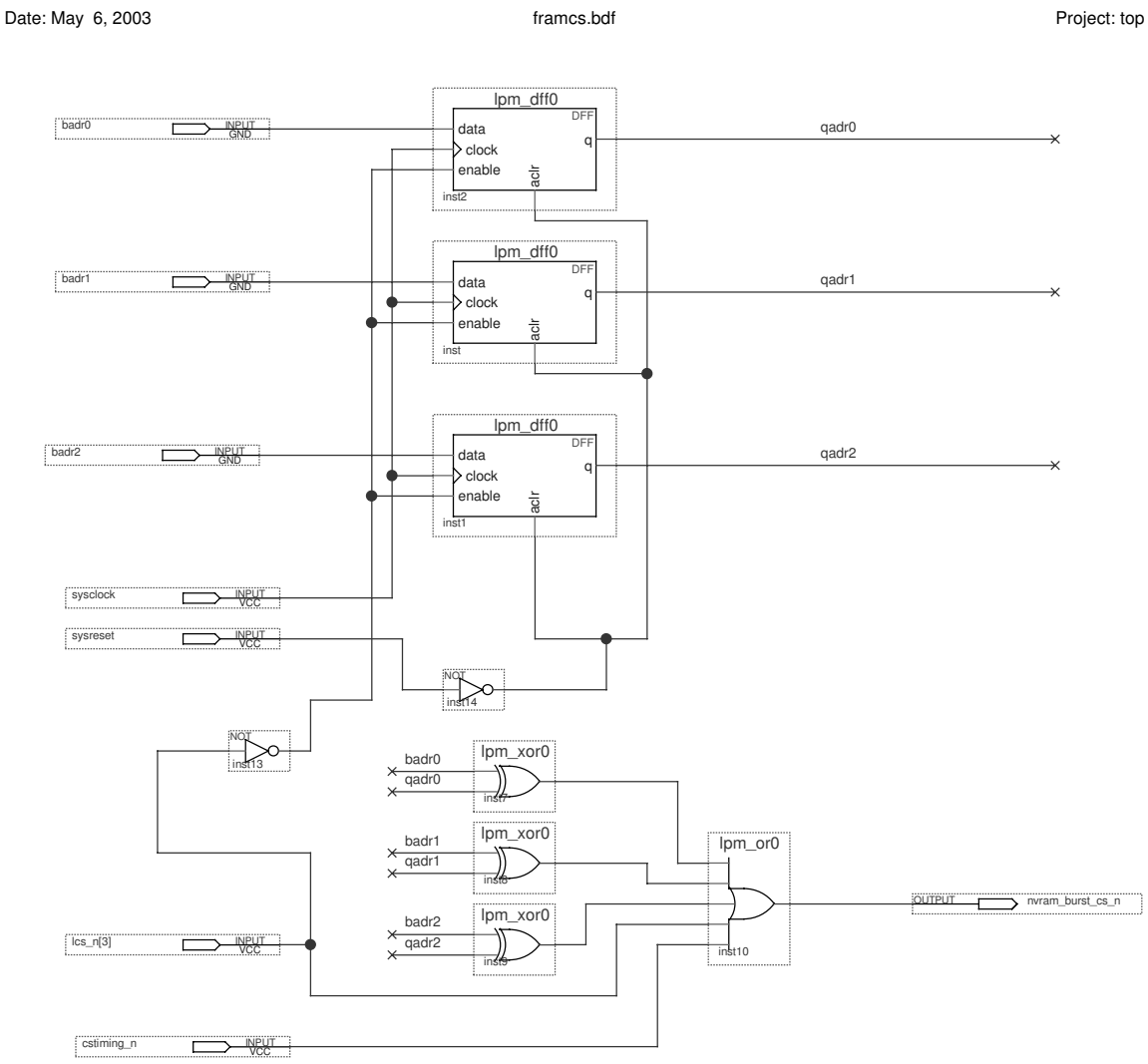
### *11.1.2.2 Top Level Block Diagram 2*



### *11.1.2.3 framcs Logic*

The following logic diagram defines the function of the logic in the ***framcs*** part of the CPLD:



### *11.1.2.4 decode_block Program*

The following code listing defines the function of the logic in the ***decode_block*** part of the CPLD:

```
INCLUDE "lpm_ff.inc"; -- 10:30 AM Mar 26, 2003
INCLUDE "lpm_counter.inc";

SUBDESIGN decode_block
(
        sysreset_n      : INPUT;
        sysclk          : INPUT;
        CSTiming_n      : INPUT;
        LBootCS_n       : INPUT; -- 16 Bit flash/sram or 32 bit FLASH (32MB) chip select
        lcs_n[3..0]     : INPUT; -- latched from upper address/data bus
        ldev_addr[21..19] : INPUT; -- latched device address.  Note that this is not the PowerPC conventional
                -- address ordering.  Device Addr0 is the LSB.
        flash_n/sram_sel  : INPUT;
        bootsmall_n     : INPUT;
        ldevR_W_n       : INPUT;
        Dev_We_n0       : INPUT;
        nvram_burst_cs_n: INPUT;

        small_flash_lo_cs_n: OUTPUT;  -- default lboot_cs_n, swappable to lcs_n[0]
        small_flash_hi_cs_n: OUTPUT;  -- default lboot_cs_n, swappable to lcs_n[0]
        sram_lo_cs_n    : OUTPUT;  -- default lboot_cs_n, swappable to lcs_n[0]
        sram_hi_cs_n    : OUTPUT;  -- default lboot_cs_n, swappable to lcs_n[0]
        big_flash_cs_n  : OUTPUT;  -- default lcs_n[0], swappable to lboot_cs_n
        Read_n          : OUTPUT;
        Write_n         : OUTPUT;
        uart_cs_n       : OUTPUT;  -- lcs_n[2]
        nvram_cs_n      : OUTPUT;  -- lcs_n[3]
        test            : OUTPUT;  -- test only
        fpga_cs_n       : OUTPUT;  -- lcs_n[1]
)

VARIABLE

        abovebootarea   : NODE; -- memory above boot area
        bootarea        : NODE; -- 750FX boots from this area
        lowerarea       : NODE;
        upperarea       : NODE;
        hiarea          : NODE;
        loarea          : NODE;
        toparea         : NODE;
        botarea         : NODE;

        SRAM/FLASH/BIG : NODE; -- boot from small flash
```

```
                FLASH/SRAM/BIG : NODE; -- boot from SRAM
                BIG/SRAM/FLASH : NODE; -- boot BIG, then SRAM, flash on CS0
                BIG/FLASH/SRAM : NODE; -- boot BIG, then flash, SRAM on CS0
                halfclk        : NODE;
                sram_cs_n      : NODE; -- node is low when either sram_cs is active low
                WriteNVRAM_n   : NODE; -- latched low write_n

-- |     Lboot          /   LCS0
-- flash_n/sram_sel  | bootsmall_n  | above boot, boot    /  upper, lower
-- ============= |==========|=========================================
--      0          |   0          | SRAM, small flash  /  big flash
--      0          |   1          |  - big flash -     /  SRAM, small flash
--      1          |   0          | small flash, SRAM  /  big flash
--      1          |   1          |  - big flash -     /  small flash, SRAM

                del_nvramcs : lpm_ff WITH (
                                LPM_WIDTH = 12,enable,aset,
                                LPM_FFTYPE = "DFF"
                                );

                del_uart : lpm_ff WITH (
                                LPM_WIDTH = 10,enable,aset,
                                LPM_FFTYPE = "DFF"
                                );

                halfclk_ : lpm_ff WITH (
                                LPM_WIDTH = 1,
                                LPM_FFTYPE = "DFF"
                                );

BEGIN
--  basic chip selects
                !fpga_cs_n      = !lcs_n[1] & !CSTiming_n; -- lcs1_n chip select

-- divide down 133Mhz clock to make 66MHz clock
                halfclk_.enable = !nvram_burst_cs_n;
                halfclk_.aset = nvram_burst_cs_n;
                halfclk_.clock = sysclk;
                halfclk_.data[0] = !halfclk_.q[0];
                halfclk = halfclk_.q[0 ];

-- delay nvram_burst_cs_n for nine 66MHz clocks
                del_nvramcs.enable =  !nvram_burst_cs_n;
                del_nvramcs.aset =  nvram_burst_cs_n;
                del_nvramcs.clock = halfclk;
                del_nvramcs.data[0] = gnd;
                del_nvramcs.data[1] = del_nvramcs.q[0];
                del_nvramcs.data[2] = del_nvramcs.q[1];
```

```
                del_nvramcs.data[3] = del_nvramcs.q[2];
                del_nvramcs.data[4] = del_nvramcs.q[3];
                del_nvramcs.data[5] = del_nvramcs.q[4];
                del_nvramcs.data[6] = del_nvramcs.q[5];
                del_nvramcs.data[7] = del_nvramcs.q[6];
                del_nvramcs.data[8] = del_nvramcs.q[7];
                del_nvramcs.data[9] = del_nvramcs.q[8];
                nvram_cs_n = del_nvramcs.q[4] # (!del_nvramcs.q[9] & !writeNVRAM_n) # nvram_burst_cs_n;

                !WriteNVRAM_n =!CSTiming_n & !LDevR_W_n;

    !uart_cs_n   = !lcs_n[2] & !CSTiming_n;

-- LBOOTCS_N,  Boot area and aboveboot area
                abovebootarea    = (!lbootcs_n &  ldev_addr[20] &  ldev_addr[19]);
                bootarea         = (!lbootcs_n &  ldev_addr[20] & !ldev_addr[19]);
                hiarea           = (!lbootcs_n & !ldev_addr[20] &  ldev_addr[19]);
                loarea           = (!lbootcs_n & !ldev_addr[20] & !ldev_addr[19]);
-- Chip select 0, upperarea and lowerarea
                toparea          = (!lcs_n[0]  &  ldev_addr[20] &  ldev_addr[19]);
                botarea          = (!lcs_n[0]  &  ldev_addr[20] & !ldev_addr[19]);
                upperarea        = (!lcs_n[0]  & !ldev_addr[20] &  ldev_addr[19]);
                lowerarea        = (!lcs_n[0]  & !ldev_addr[20] & !ldev_addr[19]);

                FLASH/SRAM/BIG = (!flash_n/sram_sel & !bootsmall_n); -- boot from small flash
                BIG/FLASH/SRAM = (!flash_n/sram_sel &  bootsmall_n); -- boot BIG, flash at top of CS0
                SRAM/FLASH/BIG = ( flash_n/sram_sel & !bootsmall_n); -- boot from SRAM
                BIG/SRAM/FLASH = ( flash_n/sram_sel &  bootsmall_n); -- boot BIG, SRAM at top of CS0

                test = small_flash_hi_cs_n;  -- for debug only

                !small_flash_hi_cs_n = (!CSTiming_n &
                                        ((FLASH/SRAM/BIG & abovebootarea)
                                                #  (BIG/FLASH/SRAM & toparea)
                                        #  (SRAM/FLASH/BIG & hiarea)
                                        #  (BIG/SRAM/FLASH & upperarea)));

                !small_flash_lo_cs_n = (!CSTiming_n &
                                        ((FLASH/SRAM/BIG & bootarea)
                        #  (BIG/FLASH/SRAM & botarea)
                                        #  (SRAM/FLASH/BIG & loarea)
                                        #  (BIG/SRAM/FLASH & lowerarea)));

                !sram_hi_cs_n    =    (!CSTiming_n &
                                       ((FLASH/SRAM/BIG & hiarea)
                                        #  (BIG/FLASH/SRAM & upperarea)
                                        #  (SRAM/FLASH/BIG & abovebootarea)
                                        #  (BIG/SRAM/FLASH & toparea)));
```

```
!sram_lo_cs_n        =  (!CSTiming_n &
                         ((FLASH/SRAM/BIG & loarea)
                        #(BIG/FLASH/SRAM & lowerarea)
                        #(SRAM/FLASH/BIG & bootarea)
                        #(BIG/SRAM/FLASH & botarea)));

!big_flash_cs_n   =  (!CSTiming_n &
                        ((!bootsmall_n & !lcs_n[0])
                            #  ( bootsmall_n & !lbootcs_n)));

sram_cs_n         =sram_hi_cs_n & sram_lo_cs_n;

!Read_n           =!CSTiming_n &  LDevR_W_n;

!Write_n =        ((!CSTiming_n  &  !LDevR_W_n)  &  !nvram_burst_cs_n)  #  (!Dev_We_n0  &
nvram_burst_cs_n);
END;
```

### *11.1.2.5 registers2 Program*

The following code listing defines the function of the logic in the ***registers2*** part of the CPLD:

```
INCLUDE "lpm_ff.inc";
INCLUDE "lpm_mux.inc";

SUBDESIGN registers2
(
        sysreset_n          : INPUT;
        cstiming_n          : INPUT;
        we_n0               : INPUT;  -- Dev_we_n[0] from controller
        ale                  : INPUT;  -- for internal latching of address/control
        dev_ad[7..0]        : BIDIR;  -- when input address/control from system controller it needs latched,
--   "                "     -- when output to system controller it is data
        fpga_cs_n           : INPUT;  -- uses lcs_n1
        badr[2..0]          : INPUT;  -- burst addresses
        target/host_n       : INPUT;  -- PCI switch
        flash_n/sram_sel    : INPUT;  -- Flash/SRAM switch
        bootsmall_n         : INPUT;  -- Boot switch
        switch_a            : INPUT;  -- spare switch A
        switch_b            : INPUT;  -- spare switch B
        ATX_OK_N            : INPUT;  -- low if ATX power is on.

        ldevR_W_n           : OUTPUT;  -- latched directly from muxed address/data bus
        lboot_cs_n          : OUTPUT;  -- latched directly from muxed address/data bus
        led0                : OUTPUT;
        led1                : OUTPUT;
        led2                : OUTPUT;

        cpu0_smi_n          : OUTPUT;  -- 2.5V logic to CPU0, resets to high
        cpu1_smi_n          : OUTPUT;  -- 2.5V logic to CPU1, resets to high
        cpu_tben            : OUTPUT;  -- 2.5V logic to both CPUs, resets to high
        cpu_mcp0            : OUTPUT;  -- 2.5V logic to both CPUs, resets to high
        cpu_mcp1            : OUTPUT;  -- 2.5V logic to both CPUs, resets to high

        read_oe             : OUTPUT;
        mpp_block_n         : OUTPUT;  -- 1 = pass mpp_Xreset pins
)

VARIABLE
        ldev_adr[8..3]      : NODE;  -- shifting addresses from [7..2] to match other designs.

        PLDversion_sel : node;-- register 0
        PLDversion[7..0] : NODE;

        Register1_sel : NODE;-- register 1
```

```
        Register1 : lpm_ff WITH (
                        LPM_WIDTH = 8,
                        LPM_FFTYPE = "DFF"
                        );

        Register2_sel : NODE;-- register 2
        Register2 : lpm_ff WITH (
                        LPM_WIDTH = 8,
                        LPM_FFTYPE = "DFF"
                        );

        Register3_sel : NODE;-- register 3
        Register3 : lpm_ff WITH (
                        LPM_WIDTH = 8,
                        LPM_FFTYPE = "DFF"
                        );

        BoardRev_sel : node;-- register 4
        BoardRev[7..0] : NODE;

        led0_node : node;
        led1_node : node;
        led2_node : node;

        cpu0_smi_n_node: node;  --
        cpu1_smi_n_node: node;  --
        cpu_tben_node    : node;  --
        cpu_mcp0_node  : node;  --
        cpu_mcp1_node  : node;  --

        data_sel[3..0] : NODE; -- which bus to read
        bux_muxer : lpm_mux WITH (
                        LPM_WIDTH = 8, -- size of each bus
                        LPM_SIZE = 5, -- number of buses
                        LPM_WIDTHS = 8 -- ???
                        );
        lpm_ff_component : lpm_ff WITH (
                        LPM_WIDTH = 8,
                        LPM_FFTYPE = "DFF"
                        );

BEGIN
--  Since we need to drive out the data on dev_ad[7..0] anyway,
--  no need to bring in the latched versions of them.
        lpm_ff_component.clock = !ale;
        lpm_ff_component.data[7..0] = dev_ad[7..0];

-- latch the address and control signals from AD bus
```

```
                ldev_adr[8..3] = lpm_ff_component.q[7..2];

    -- latched control signals
                ldevR_W_n = lpm_ff_component.q[1];  -- read or write cycle
                lboot_cs_n = lpm_ff_component.q[0] # cstiming_n;  -- on board flash bank 0

    -- all accesses to PLD registers are assumed to be 8-bit hw wise.

    --************************************************************************
    -- Register0, Read only
                PLDversion[7..0] = B"00011000"; -- "18"
    -- Register4, Read only
                BoardRev[7..0]   = B"00000010"; -- "02"

    -- board rev 02 = Argan 2.
    --************************************************************************

    -- Register1, Read with clear (must write prior to each read)
                Register1.enable = Register1_sel & !ldevR_W_n;
                Register1.aclr = !sysreset_n;
                Register1.clock = we_n0;

                Register1.data[5] = ATX_OK_N;--
                Register1.data[4] = switch_b;-- spare switch B
                Register1.data[3] = switch_a;-- spare switch A
                Register1.data[2] = target/host_n;  -- PCI switch
                Register1.data[1] = flash_n/sram_sel;  -- Flash/SRAM switch
                Register1.data[0] = bootsmall_n;  -- Boot switch

    -- Register2, Write/Read, use to drive 3 leds, SMI0, SMI1, TBEN, MCP0, MCP1
                Register2.enable = Register2_sel & !ldevR_W_n;
                Register2.aclr = !sysreset_n;  -- active sysreset_n inverted = 1 here, clears the output
                Register2.clock = we_n0;
                Register2.data[7..0] = dev_ad[7..0];

                led2_node          = !Register2.q[2];  -- sysreset turns on led (changed Ver 07)
                led1_node          = !Register2.q[1];  -- sysreset turns on led
                led0_node          = !Register2.q[0];  -- sysreset turns on led

                led2 = tri(gnd,led2_node); -- tri (signal,oe)
                led1 = tri(gnd,led1_node); -- tri (signal,oe)
                led0 = tri(gnd,led0_node); -- tri (signal,oe)

                cpu0_smi_n_node= !Register2.q[3];  --
                cpu1_smi_n_node= !Register2.q[4];  --
                cpu_tben_node    = !Register2.q[5];  --
                cpu_mcp0_node  = !Register2.q[6];  --
                cpu_mcp1_node  = !Register2.q[7];  --
```

```
                cpu0_smi_n = cpu0_smi_n_node;  --
                cpu1_smi_n = cpu1_smi_n_node;  --
                cpu_tben = cpu_tben_node;  --
                cpu_mcp0 = cpu_mcp0_node;  --
                cpu_mcp1 = cpu_mcp1_node;  --

-- Register3, Write/Read
                Register3.enable = Register3_sel & !ldevR_W_n;
                Register3.aclr = !sysreset_n;  -- !sysreset_n = 1 in reset, clears the output
                Register3.clock = we_n0;
                Register3.data[7..0] = dev_ad[7..0];

--              Register3.data[0] = dev_ad[0];-- mpp_block_n, 1 = pass mpp_Xreset pins
                mpp_block_n       = Register3.q[0];-- mpp_block_n, 1 = pass mpp_Xreset pins

-- Register 0
                if ((badr[2..0] == B"000")) then
                                PLDversion_sel = !fpga_cs_n;
                                data_sel[] = 0;
                else
                                PLDversion_sel = gnd;
                end if;

-- Register 1
                if ((badr[2..0] == B"001")) then
                                Register1_sel = !fpga_cs_n;
                                data_sel[] = 1;
                else
                                Register1_sel = gnd;
                end if;

-- Register 2
                if ((badr[2..0] == B"010")) then
                                Register2_sel = !fpga_cs_n;
                                data_sel[] = 2;
                else
                                Register2_sel = gnd;
                end if;

-- Register 3
                if ((badr[2..0] == B"011")) then
                                Register3_sel = !fpga_cs_n;
                                data_sel[] = 3;
                else
                                Register3_sel = gnd;
                end if;
```

```
-- Register 4
        if ((badr[2..0] == B"100")) then
                BoardRev_sel = !fpga_cs_n;
                data_sel[] = 4;
        else
                BoardRev_sel = gnd;
        end if;

-- read logic
        bux_muxer.data[0][7..0] = PLDversion[7..0];
        bux_muxer.data[1][7..0] = Register1.q[7..0];
        bux_muxer.data[2][7..0] = Register2.q[7..0];
        bux_muxer.data[3][7..0] = Register3.q[7..0];
        bux_muxer.data[4][7..0] = BoardRev[7..0];
        bux_muxer.sel[] = data_sel[];

-- output enable for read from FPGA only
        read_oe = !cstiming_n & ldevR_W_n &
                (Register1_sel # Register2_sel # Register3_sel
                # PLDversion_sel # BoardRev_sel);

        dev_ad[0] = TRI( bux_muxer.result[0], read_oe );
        dev_ad[1] = TRI( bux_muxer.result[1], read_oe );
        dev_ad[2] = TRI( bux_muxer.result[2], read_oe );
        dev_ad[3] = TRI( bux_muxer.result[3], read_oe );
        dev_ad[4] = TRI( bux_muxer.result[4], read_oe );
        dev_ad[5] = TRI( bux_muxer.result[5], read_oe );
        dev_ad[6] = TRI( bux_muxer.result[6], read_oe );
        dev_ad[7] = TRI( bux_muxer.result[7], read_oe );
END;
```

#### *11.1.2.6 reset_block Program*

The following code listing defines the function of the logic in the ***reset_block*** part of the CPLD:

```
INCLUDE "lpm_ff.inc";

SUBDESIGN reset_block
(
        25Mclk          : INPUT;
        rw_hreset       : INPUT;
        rw_sreset       : INPUT;
        rw_trst         : INPUT;
        mpp_reset_2ms_n: INPUT;
        mpp0_hreset_n   : INPUT;
        mpp0_sreset_n   : INPUT;
        mpp1_hreset_n   : INPUT;
        mpp1_sreset_n   : INPUT;
        mpp_reset_out_n : INPUT;
        initact         : INPUT;
        pgd             : INPUT;
        target/host_n   : INPUT;
        serial_eeprom   : INPUT;

        pci_reset_n     : BIDIR;

        sysreset_n      : OUTPUT;
        cpu0_hreset_n   : OUTPUT;
        cpu0_sreset_n   : OUTPUT;
        cpu1_hreset_n   : OUTPUT;
        cpu1_sreset_n   : OUTPUT;
        cpu_trst_n      : OUTPUT;

        cpufan_ok_n     : INPUT;
        ignore_fans_n   : INPUT;
        mpp_block_n     : INPUT;

        nofan_n         : OUTPUT;
        sloclk          : OUTPUT;
        ATX_OK_N        : INPUT;
        led_red_n       : OUTPUT;
        sysreset        : OUTPUT;
)

VARIABLE
        del_rw_hreset[1..0]: dff;
        del_rw_sreset[1..0]: dff;
        del_rw_trst[1..0]   : dff;
```

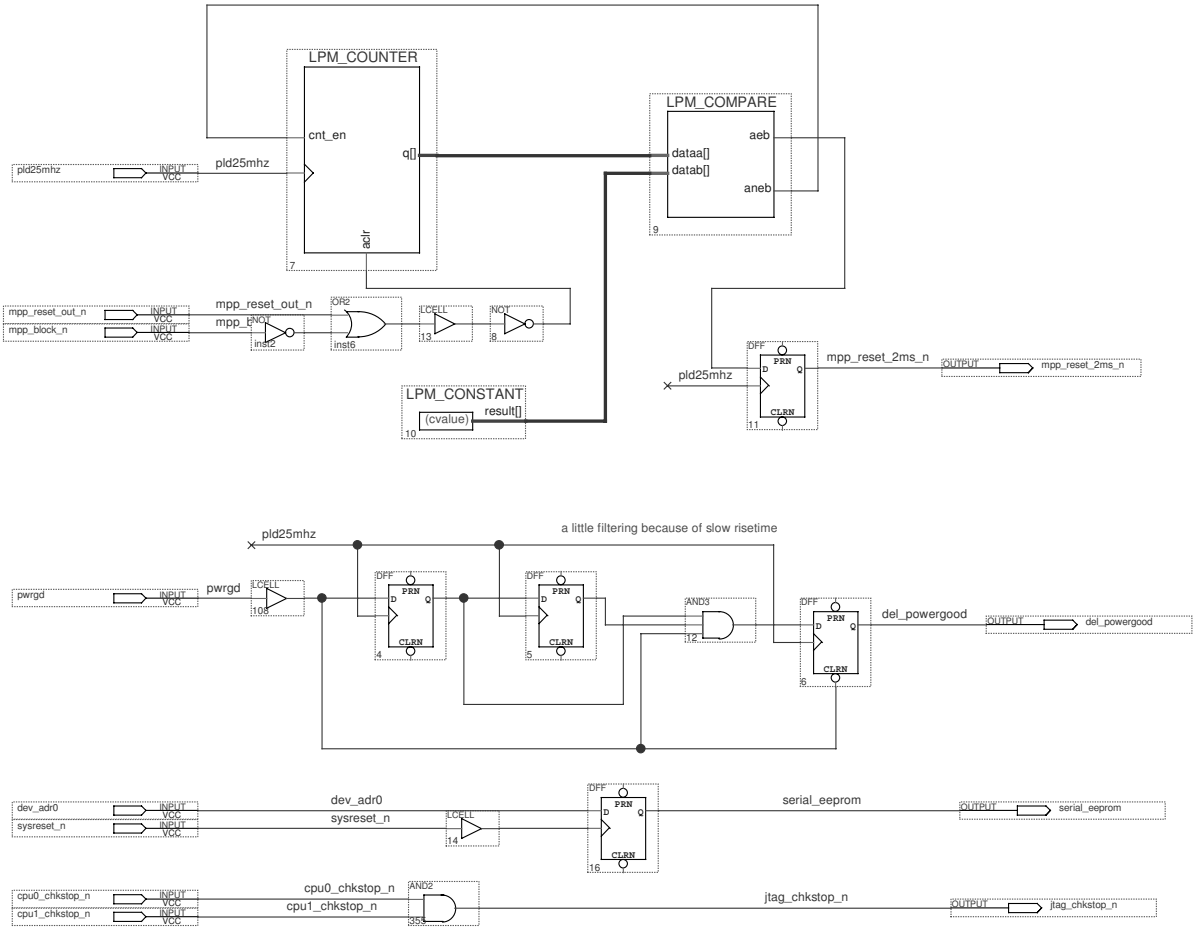
```
        cpu_trst_n_         : NODE;
        cpu0_hreset_n_      : NODE;
        cpu0_sreset_n_      : NODE;
        cpu1_hreset_n_      : NODE;
        cpu1_sreset_n_      : NODE;
        del_sysreset_n      : NODE;
        del_sysreset_n_     : lpm_ff WITH (
                            LPM_WIDTH = 8,
                            LPM_FFTYPE = "DFF"
                            );

        del_pgd             : NODE;
        del_pgd_ : lpm_ff WITH (
                            LPM_WIDTH = 8,
                            LPM_FFTYPE = "DFF"
                            );

        sloclk_             : lpm_ff WITH (
                            LPM_WIDTH = 1,
                            LPM_FFTYPE = "DFF"
                            );

        sloclk2             : NODE;
        sloclk2_ : lpm_ff WITH (
                            LPM_WIDTH = 1,
                            LPM_FFTYPE = "DFF"
                            );

        sloclk3[1..0]       : dff;
    sloclk3_    : node;

BEGIN

--      If target mode then pci_reset_n causes  reset_n.  If host mode, ignore pci_reset_n.
        sysreset_n = (pci_reset_n # !target/host_n # !ATX_OK_N) & mpp_reset_2ms_n & del_pgd;
        sysreset =          !sysreset_n;

        del_sysreset_n_.clock = 25Mclk;
        del_sysreset_n_.enable = sloclk3_;
        del_sysreset_n_.data[0] = sysreset_n;
        del_sysreset_n_.data[1] = del_sysreset_n_.q[0];
        del_sysreset_n_.data[2] = del_sysreset_n_.q[1];
        del_sysreset_n_.data[3] = del_sysreset_n_.q[2];
        del_sysreset_n_.data[4] = del_sysreset_n_.q[3];
        del_sysreset_n_.data[5] = del_sysreset_n_.q[4];
        del_sysreset_n_.data[6] = del_sysreset_n_.q[5];
        del_sysreset_n_.data[7] = del_sysreset_n_.q[6];
        del_sysreset_n = del_sysreset_n_.q[7]; -- delayed version of sysreset.
```

```
-- Cascade tff's - create a 2 bit counter
-- that runs at the full rate.  AND the output bits together
-- and whenever count = "11" a pulse is generated.
-- Use that 1/4 rate pulse as an enable for the dff's.

        sloclk3[].clk = 25Mclk;
        sloclk3[].d = sloclk3[].q + 1;

        sloclk3_ = sloclk3[1] & sloclk3[0];

-- delay pgd using 25Mhz clock
        del_pgd_.clock = 25Mclk;
        del_pgd_.enable = sloclk3_;
        del_pgd_.data[0] = pgd;
        del_pgd_.data[1] = del_pgd_.q[0];
        del_pgd_.data[2] = del_pgd_.q[1];
        del_pgd_.data[3] = del_pgd_.q[2];
        del_pgd_.data[4] = del_pgd_.q[3];
        del_pgd_.data[5] = del_pgd_.q[4];
        del_pgd_.data[6] = del_pgd_.q[5];
        del_pgd_.data[7] = del_pgd_.q[6];
        del_pgd = del_pgd_.q[0] & del_pgd_.q[1] & del_pgd_.q[2] & del_pgd_.q[3]
                & del_pgd_.q[4] & del_pgd_.q[5] & del_pgd_.q[6] & del_pgd_.q[7];

-- this circuitry below is a filter and designed to provide cleaner signals to CPU
-- *** hreset logicstart
        del_rw_hreset[1..0].clk = 25Mclk;
        del_rw_hreset[0].d = rw_hreset;
        del_rw_hreset[1].d = del_rw_hreset[0].q;
        cpu0_hreset_n_ = !(initact & serial_eeprom)
                & (del_rw_hreset[0].q & del_rw_hreset[1].q)
                & sysreset_n & del_sysreset_n
                & (mpp0_hreset_n # !mpp_block_n);
        cpu0_hreset_n  = cpu0_hreset_n_;

        cpu1_hreset_n_ = !(initact & serial_eeprom)
                & (del_rw_hreset[0].q & del_rw_hreset[1].q)
                & sysreset_n & del_sysreset_n
                & (mpp1_hreset_n # !mpp_block_n); -- Ver 10 test

        cpu1_hreset_n  = cpu1_hreset_n_;
-- *** hreset logicend

-- *** sreset logicstart
        del_rw_sreset[0].d = rw_sreset;
        del_rw_sreset[1..0].clk = 25Mclk;
```

```
                del_rw_sreset[1].d = del_rw_sreset[0].q;
                cpu0_sreset_n_   = (del_rw_sreset[0].q & del_rw_sreset[1].q) & (mpp0_sreset_n # !mpp_block_n);
--              above line: if mpp_block_n is 1 then allow mpp0_sreset# through.
                cpu0_sreset_n = cpu0_sreset_n_;
                cpu1_sreset_n_   = (del_rw_sreset[0].q & del_rw_sreset[1].q) & (mpp1_sreset_n # !mpp_block_n);
                cpu1_sreset_n = cpu1_sreset_n_;
-- *** sreset logicend

-- *** treset logic start
                del_rw_trst[1..0].clk = 25Mclk;
                del_rw_trst[0].d = rw_trst;
                del_rw_trst[1].d = del_rw_trst[0].q;

                cpu_trst_n_  = !(initact & serial_eeprom)
                                & (del_rw_trst[0].q & del_rw_trst[1].q)
                                & sysreset_n & del_sysreset_n;
-- *** treset logicend

-- this reset signal is output to PCI bus (this is also the PCI reset input)
                pci_reset_n = TRI(del_pgd, !target/host_n);

--  driven from 2.5V IO
                cpu_trst_n          = cpu_trst_n_;

-- If a fan fails (cpufan_ok_n = 1) turn on RED LED and shut down power supplies unless
-- "ignore_fans_n" =0 jumper is installed, then just turn on RED LED.

--              nofan_n = tri(gnd,fans_n);
                nofan_n = tri(gnd,(cpufan_ok_n & ignore_fans_n));  -- good in ver 12_01

--  If ignore_fans_n jumper is installed (ignore_fans_n = 0) or
--  during sysreset_n = 0 or
--  cpufan_ok_n = 1 (bad or no fan) then turn on RED LED.
                led_red_n = tri(gnd,(!ignore_fans_n # !sysreset_n # cpufan_ok_n));
END;
```

### *11.1.2.7 misc Logic*

The following logic diagram defines the function of the logic in the ***misc*** part of the CPLD:



## 11.2 Timing—Registers and Control Functions

The timing data in the following tables are based on simulation.

### 11.2.1 Maximum Clock Frequency

*Table 11-4* provides the actual frequency at which the indicated clock is running, and the highest frequency at which it can be allowed to run before the period becomes shorter than the worst case signal propagation time.

*Table 11-4. Maximum Clock Frequency*

| Clock | Actual Operating Frequency | Allowed Maximum Frequency (period) |
|---|---|---|
| pld25mhz | 25MHz | 166.67MHz (6ns) |
| dev_we_n[0] | None | 181.82MHz (5.5ns) |
| pld_sysclk | 133.33MHz | 185.19MHz (5.4ns) |

### 11.2.2 Clock-to-Output Time

*Table 11-5* provides the longest and shortest input-to-output delay for each output signal clocked through a register for the all clocks that gate the signal.

*Table 11-5. Clock-to-Output Time*

| Output Signal | Clock | Longest Delay (ns) | Shortest Delay (ns) |
|---|---|---|---|
| big_flash_cs_n | ale | 9.100 | 9.100 |
| cpu_mcp0 | dev_we_n[0] | 10.900 | 10.900 |
| cpu_mcp1 | dev_we_n[0] | 10.900 | 10.900 |
| cpu_tben | dev_we_n[0] | 10.900 | 10.900 |
| cpu_trst_n | pld25mhz | 26.400 | 18.500 |
| cpu0_hreset_n | dev_we_n[0] | 15.500 | 15.500 |
| cpu0_hreset_n | pld25mhz | 23.100 | 15.200 |
| cpu0_smi_n | dev_we_n[0] | 10.900 | 10.900 |
| cpu0_sreset_n | dev_we_n[0] | 14.700 | 14.700 |
| cpu0_sreset_n | pld25mhz | 14.500 | 14.400 |
| cpu1_hreset_n | dev_we_n[0] | 19.200 | 19.200 |
| cpu1_hreset_n | pld25mhz | 26.800 | 18.900 |
| cpu1_smi_n | dev_we_n[0] | 10.900 | 10.900 |
| cpu1_sreset_n | dev_we_n[0] | 14.700 | 14.700 |
| cpu1_sreset_n | pld25mhz | 14.500 | 14.400 |
| dev_adr[0] | ale | 12.500 | 12.500 |
| dev_adr[0] | dev_we_n[0] | 9.500 | 9.400 |
| dev_adr[1] | ale | 12.500 | 12.500 |
| dev_adr[1] | dev_we_n[0] | 9.400 | 9.400 |
| dev_adr[2] | ale | 12.500 | 12.500 |

*Table 11-5. Clock-to-Output Time (Continued)*

| Output Signal | Clock | Longest Delay (ns) | Shortest Delay (ns) |
|---|---|---|---|
| dev_adr[2] | dev_we_n[0] | 9.400 | 9.400 |
| dev_adr[3] | ale | 12.500 | 12.500 |
| dev_adr[3] | dev_we_n[0] | 9.400 | 9.400 |
| dev_adr[4] | ale | 12.500 | 12.500 |
| dev_adr[4] | dev_we_n[0] | 9.400 | 9.400 |
| dev_adr[5] | ale | 12.500 | 12.500 |
| dev_adr[5] | dev_we_n[0] | 9.400 | 9.400 |
| dev_adr[6] | ale | 12.500 | 12.500 |
| dev_adr[6] | dev_we_n[0] | 9.400 | 9.400 |
| dev_adr[7] | ale | 12.500 | 12.500 |
| dev_adr[7] | dev_we_n[0] | 9.400 | 9.400 |
| led_red_n | pld25mhz | 9.500 | 9.400 |
| led0 | dev_we_n[0] | 5.700 | 5.700 |
| led1 | dev_we_n[0] | 5.700 | 5.700 |
| led2 | dev_we_n[0] | 5.700 | 5.700 |
| nvram_cs_n | ale | 18.500 | 18.500 |
| nvram_cs_n | pld_sysclk | 23.100 | 17.700 |
| pci_reset_n | pld25mhz | 9.500 | 9.500 |
| read_n | ale | 9.400 | 9.400 |
| small_flash_cs_n | ale | 9.100 | 9.100 |
| small_flash_hi_cs_n | ale | 9.100 | 9.100 |
| sram_cs_n | ale | 9.100 | 9.100 |
| sram_hi_cs_n | ale | 9.100 | 9.100 |
| sysreset | pld25mhz | 14.500 | 14.400 |
| sysreset_n | pld25mhz | 14.500 | 14.400 |
| textpin_d | ale | 9.300 | 9.300 |
| uart_cs_n | pld_sysclk | 10.500 | 9.000 |
| write_n | ale | 9.800 | 9.800 |
| write_n | pld_sysclk | 11.300 | 9.800 |

### 11.2.3 Pin-to-Pin Signal Delay

*Table 11-6* provides the input pin-to-output pin delay time for all signals that are not clocked through a register.

*Table 11-6. Pin-to-Pin Signal Delay*

| Source | Destination | Longest Delay (ns) | Shortest Delay (ns) |
|---|---|---|---|
| atx_ok_n | cpu_trst_n | 14.200 | 14.200 |
| atx_ok_n | cpu0_hreset_n | 10.900 | 10.500 |
| atx_ok_n | cpu1_hreset_n | 14.200 | 14.200 |
| atx_ok_n | led_red_n | 5.300 | 5.300 |
| atx_ok_n | sysreset | 10.300 | 10.300 |
| atx_ok_n | sysreset_n | 10.300 | 10.300 |
| badr[0] | dev_adr[0] | 8.700 | 5.600 |
| badr[0] | dev_adr[1] | 8.700 | 5.600 |
| badr[0] | dev_adr[2] | 8.700 | 5.600 |
| badr[0] | dev_adr[3] | 8.700 | 5.600 |
| badr[0] | dev_adr[4] | 8.700 | 5.600 |
| badr[0] | dev_adr[5] | 8.700 | 5.600 |
| badr[0] | dev_adr[6] | 8.700 | 5.600 |
| badr[0] | dev_adr[7] | 8.700 | 5.600 |
| badr[0] | nvram_cs_n | 14.300 | 14.300 |
| badr[0] | uart_cs_n | 5.600 | 5.600 |
| badr[0] | write_n | 6.400 | 5.600 |
| badr[1] | dev_adr[0] | 8.700 | 5.600 |
| badr[1] | dev_adr[1] | 8.700 | 5.600 |
| badr[1] | dev_adr[2] | 8.700 | 5.600 |
| badr[1] | dev_adr[3] | 8.700 | 5.600 |
| badr[1] | dev_adr[4] | 8.700 | 5.600 |
| badr[1] | dev_adr[5] | 8.700 | 5.600 |
| badr[1] | dev_adr[6] | 8.700 | 5.600 |
| badr[1] | dev_adr[7] | 8.700 | 5.600 |
| badr[1] | nvram_cs_n | 14.300 | 14.300 |
| badr[1] | uart_cs_n | 5.600 | 5.600 |
| badr[1] | write_n | 6.400 | 5.600 |
| badr[2] | dev_adr[0] | 8.700 | 5.600 |
| badr[2] | dev_adr[1] | 8.700 | 5.600 |
| badr[2] | dev_adr[2] | 8.700 | 5.600 |
| badr[2] | dev_adr[3] | 8.700 | 5.600 |
| badr[2] | dev_adr[4] | 8.700 | 5.600 |

*Table 11-6. Pin-to-Pin Signal Delay (Continued)*

| Source | Destination | Longest Delay (ns) | Shortest Delay (ns) |
|---|---|---|---|
| badr[2] | dev_adr[5] | 8.700 | 5.600 |
| badr[2] | dev_adr[6] | 8.700 | 5.600 |
| badr[2] | dev_adr[7] | 8.700 | 5.600 |
| badr[2] | nvram_cs_n | 14.700 | 14.700 |
| badr[2] | uart_cs_n | 7.100 | 5.600 |
| badr[2] | write_n | 7.900 | 5.600 |
| bootsmall_n | big_flash_cs_n | 5.300 | 5.300 |
| bootsmall_n | small_flash_cs_n | 5.300 | 5.300 |
| bootsmall_n | small_flash_hi_cs_n | 5.300 | 5.300 |
| bootsmall_n | sram_cs_n | 5.300 | 5.300 |
| bootsmall_n | sram_hi_cs_n | 5.300 | 5.300 |
| bootsmall_n | textpin_d | 5.500 | 5.500 |
| cpu0_chkstop_n | jtag_chkstop_n | 5.600 | 5.600 |
| cpu1_chkstop_n | jtag_chkstop_n | 5.600 | 5.600 |
| cpufan_ok_n | led_red_n | 5.100 | 5.100 |
| cpufan_ok_n | NOFAN_N | 5.100 | 5.100 |
| cstiming_n | big_flash_cs_n | 5.700 | 5.700 |
| cstiming_n | dev_adr[0] | 8.800 | 8.800 |
| cstiming_n | dev_adr[1] | 8.800 | 8.800 |
| cstiming_n | dev_adr[2] | 8.800 | 8.800 |
| cstiming_n | dev_adr[3] | 8.800 | 8.800 |
| cstiming_n | dev_adr[4] | 8.800 | 8.800 |
| cstiming_n | dev_adr[5] | 8.800 | 8.800 |
| cstiming_n | dev_adr[6] | 8.800 | 8.800 |
| cstiming_n | dev_adr[7] | 8.800 | 8.800 |
| cstiming_n | fpga_cs_n | 5.700 | 5.700 |
| cstiming_n | nvram_cs_n | 15.900 | 15.900 |
| cstiming_n | read_n | 5.700 | 5.700 |
| cstiming_n | small_flash_cs_n | 5.700 | 5.700 |
| cstiming_n | small_flash_hi_cs_n | 5.700 | 5.700 |
| cstiming_n | sram_cs_n | 5.700 | 5.700 |
| cstiming_n | sram_hi_cs_n | 5.700 | 5.700 |
| cstiming_n | textpin_d | 5.900 | 5.900 |
| cstiming_n | uart_cs_n | 7.200 | 5.700 |
| cstiming_n | write_n | 8.000 | 5.700 |
| dev_we_n[0] | uart_cs_n | 6.900 | 5.400 |
| dev_we_n[0] | write_n | 7.700 | 5.800 |

*Table 11-6. Pin-to-Pin Signal Delay (Continued)*

| Source | Destination | Longest Delay (ns) | Shortest Delay (ns) |
|---|---|---|---|
| flash_n/sram_sel | small_flash_cs_n | 5.300 | 5.300 |
| flash_n/sram_sel | small_flash_hi_cs_n | 5.300 | 5.300 |
| flash_n/sram_sel | sram_cs_n | 5.300 | 5.300 |
| flash_n/sram_sel | sram_hi_cs_n | 5.300 | 5.300 |
| flash_n/sram_sel | textpin_d | 5.500 | 5.500 |
| ignore_fans_n | led_red_n | 5.100 | 5.100 |
| ignore_fans_n | NOFAN_N | 5.100 | 5.100 |
| initact | cpu_trst_n | 13.900 | 13.900 |
| initact | cpu0_hreset_n | 11.000 | 10.200 |
| initact | cpu1_hreset_n | 14.300 | 13.900 |
| lcs_n[0] | big_flash_cs_n | 5.200 | 5.200 |
| lcs_n[0] | small_flash_cs_n | 5.200 | 5.200 |
| lcs_n[0] | small_flash_hi_cs_n | 5.200 | 5.200 |
| lcs_n[0] | sram_cs_n | 5.200 | 5.200 |
| lcs_n[0] | sram_hi_cs_n | 5.200 | 5.200 |
| lcs_n[0] | textpin_d | 5.400 | 5.400 |
| lcs_n[1] | dev_adr[0] | 8.500 | 8.500 |
| lcs_n[1] | dev_adr[1] | 8.500 | 8.500 |
| lcs_n[1] | dev_adr[2] | 8.500 | 8.500 |
| lcs_n[1] | dev_adr[3] | 8.500 | 8.500 |
| lcs_n[1] | dev_adr[4] | 8.500 | 8.500 |
| lcs_n[1] | dev_adr[5] | 8.500 | 8.500 |
| lcs_n[1] | dev_adr[6] | 8.500 | 8.500 |
| lcs_n[1] | dev_adr[7] | 8.500 | 8.500 |
| lcs_n[1] | fpga_cs_n | 5.400 | 5.400 |
| lcs_n[2] | uart_cs_n | 5.000 | 5.000 |
| lcs_n[2] | write_n | 5.800 | 5.800 |
| lcs_n[3] | nvram_cs_n | 15.300 | 15.300 |
| lcs_n[3] | uart_cs_n | 6.600 | 6.600 |
| lcs_n[3] | write_n | 7.400 | 5.100 |
| ldev_addr[19] | small_flash_cs_n | 5.200 | 5.200 |
| ldev_addr[19] | small_flash_hi_cs_n | 5.200 | 5.200 |
| ldev_addr[19] | sram_cs_n | 5.200 | 5.200 |
| ldev_addr[19] | sram_hi_cs_n | 5.200 | 5.200 |
| ldev_addr[19] | textpin_d | 5.400 | 5.400 |
| ldev_addr[20] | small_flash_cs_n | 5.200 | 5.200 |
| ldev_addr[20] | small_flash_hi_cs_n | 5.200 | 5.200 |

*Table 11-6. Pin-to-Pin Signal Delay (Continued)*

| Source | Destination | Longest Delay (ns) | Shortest Delay (ns) |
|---|---|---|---|
| ldev_addr[20] | sram_cs_n | 5.200 | 5.200 |
| ldev_addr[20] | sram_hi_cs_n | 5.200 | 5.200 |
| ldev_addr[20] | textpin_d | 5.400 | 5.400 |
| mpp0_hreset_n | cpu0_hreset_n | 10.600 | 10.200 |
| mpp0_sreset_n | cpu0_sreset_n | 10.200 | 10.200 |
| mpp1_hreset_n | cpu1_hreset_n | 14.700 | 13.900 |
| mpp1_sreset_n | cpu1_sreset_n | 10.200 | 10.200 |
| pci_reset_n | cpu_trst_n | 14.500 | 14.100 |
| pci_reset_n | cpu0_hreset_n | 11.200 | 10.400 |
| pci_reset_n | cpu1_hreset_n | 14.900 | 14.500 |
| pci_reset_n | led_red_n | 5.200 | 5.200 |
| pci_reset_n | sysreset | 10.200 | 10.200 |
| pci_reset_n | sysreset_n | 10.200 | 10.200 |
| target/host_n | cpu_trst_n | 14.200 | 14.200 |
| target/host_n | cpu0_hreset_n | 10.900 | 10.500 |
| target/host_n | cpu1_hreset_n | 14.600 | 14.200 |
| target/host_n | led_red_n | 5.300 | 5.300 |
| target/host_n | pci_reset_n | 4.300 | 4.300 |
| target/host_n | sysreset | 10.300 | 10.300 |
| target/host_n | sysreset_n | 10.300 | 10.300 |

### 11.2.4 Setup and Hold Time

*Table 11-7* provides the minimum setup and hold time for the indicated input relative to the indicated clock.

*Table 11-7. Setup and Hold Time*

| Input | Clock | Setup Time (ns) | Hold Time (ns) |
|---|---|---|---|
| atx_ok_n | dev_we_n[0] | 1.300 | 0.500 |
| atx_ok_n | pld25mhz | 1.400 | 0.400 |
| badr[0] | dev_we_n[0] | 3.100 | 0.200 |
| badr[0] | pld_sysclk | 2.000 | <= 0 |
| badr[1] | dev_we_n[0] | 3.100 | 0.200 |
| badr[1] | pld_sysclk | 2.000 | <= 0 |
| badr[2] | dev_we_n[0] | 3.100 | 0.200 |
| badr[2] | pld_sysclk | 2.000 | <= 0 |
| bootsmall_n | dev_we_n[0] | 1.300 | 0.500 |
| cstiming_n | dev_we_n[0] | 3.200 | 0.100 |
| dev_adr[0] | ale | 1.400 | 0.400 |
| dev_adr[0] | dev_we_n[0] | 1.000 | 0.800 |
| dev_adr[0] | pld25mhz | -6.700 | 8.600 |
| dev_adr[1] | ale | 1.500 | 0.300 |
| dev_adr[1] | dev_we_n[0] | 1.100 | 0.700 |
| dev_adr[2] | dev_we_n[0] | 1.000 | 0.800 |
| dev_adr[3] | dev_we_n[0] | 1.000 | 0.800 |
| dev_adr[4] | dev_we_n[0] | 1.000 | 0.800 |
| dev_adr[5] | dev_we_n[0] | 1.000 | 0.800 |
| dev_adr[6] | dev_we_n[0] | 1.000 | 0.800 |
| dev_adr[7] | dev_we_n[0] | 1.000 | 0.800 |
| flash_n/sram_sel | dev_we_n[0] | 1.300 | 0.500 |
| lcs_n[1] | dev_we_n[0] | 2.900 | 0.400 |
| lcs_n[3] | pld_sysclk | 1.500 | 0.300 |
| pci_reset_n | pld25mhz | 1.700 | 0.100 |
| pwrgd | pld25mhz | 5.200 | <= 0 |
| rw_hreset | pld25mhz | 1.500 | 0.300 |
| rw_sreset | pld25mhz | 1.500 | 0.300 |
| rw_trst | pld25mhz | 1.500 | 0.300 |
| switch_a | dev_we_n[0] | 1.400 | 0.400 |
| switch_b | dev_we_n[0] | 1.400 | 0.400 |
| target/host_n | dev_we_n[0] | 1.300 | 0.500 |
| target/host_n | pld25mhz | 1.400 | 0.400 |

# 12. Bills of Materials

The following sections provide information about components and tools associated with the board. This includes components mounted on the board, those that are included in the board package but not mounted on the board, and other components that are useful in working with the board but are not provided in the package. In addition, a description of how to use the location grid on the board in combination with the location charts in the schematics to physically locate components on the board is provided.

*Table 12-5* identifies all parts that can possibly be assembled on the board. Those parts that are not assembled on the board, as it ships from the factory, are labeled DNP (do not populate) in the schematics. In *Table 12-5*, the component is not assembled if the DNP column contains TRUE. In some cases installation of DNP parts requires the removal of installed parts and constitutes a change in the design of the board. IBM does not support boards that have been modified by removing normally installed parts.

*Table 12-1. Section Contents*

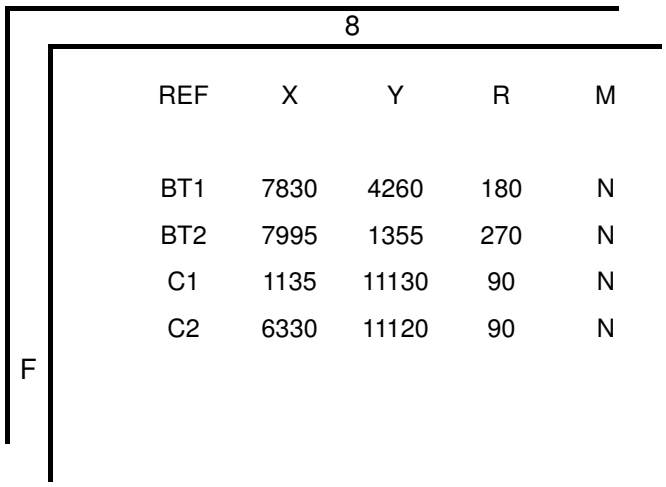


## 12.1 Component Location

The location of any component on the board can be identified by using the coordinate grid that appears along the outside edges of the board. An example of this grid, as it appears on the top side of the board is shown in *Figure 12-2*. The X-coordinate scale is along the short edge of the board and the Y-coordinate scale is along the long edge of the board.

The coordinates of each component are provided in a table that appears in the schematic diagrams. The schematic pages that contain this table are titled *Component Placement*. *Figure 12-1* shows an example of the table as it appears on the schematic page.

*Figure 12-1. Example of a Component Placement List in the Schematics*




The information provided in each column of the component placement table is described in *Table 12-2*:

*Table 12-2. Component Placement Data Description*

| Column Heading[1] | Description |
|---|---|
| REF | Reference designator as it appears in the BOM and the schematics |
| X | X-coordinate of the component location |
| Y | Y-coordinate of the component location |
| R | Rotation of the component in degrees |
| M | Shows which side of the board the component is on:<br>N = top side<br>Y = bottom side |
| **Note 1:** The column headings shown are representative only. The actual column headings will vary from schematic to schematic. However, the information provided is the same in all cases. | |

All of the coordinates are referenced to the 0,0 datum point which is at the center of the mounting hole in the bottom left corner of the board as viewed from the top side. The coordinate values in the table and the labels on the board are given in thousandths of inches. For example, an X-coordinate value of 7995 indicates that the component has an X-coordinate that is 7.995 inches from the 0,0 datum. Note that it is possible for components to have negative coordinate values.

The normal procedure to locate a component is to look up the coordinates in the Component Placement table in the schematic diagram, then use the X and Y scales on the edge of the board to get to the approximate location of the component on the board. The reference designator that is silk-screened on the board is then used to locate the component precisely. The location of component BT1, at coordinates 7830 and 4260 in the example table, is shown in *Figure 12-1*.

**Note:** The appearance of the scales as shown in the example figure is representative only. The actual appearance will vary from board to board due to constraints imposed by available board space and component placement. However, the basic format is the same in all cases with a reference mark appearing, whenever possible, every inch.

*Figure 12-2. Board Location Grid—Top View*

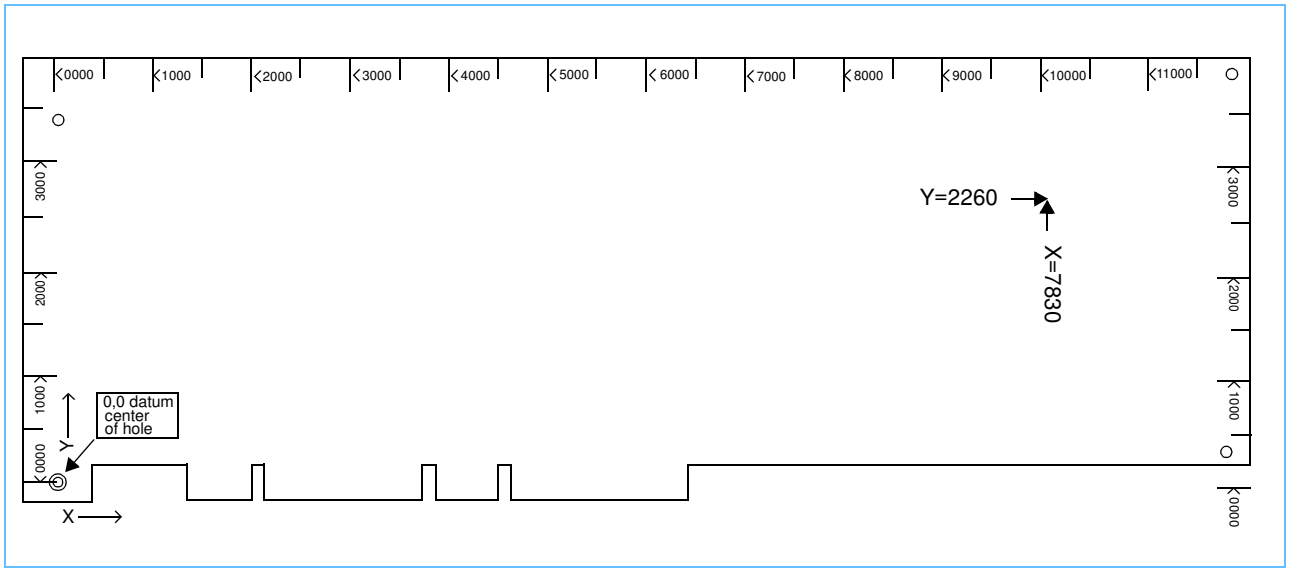
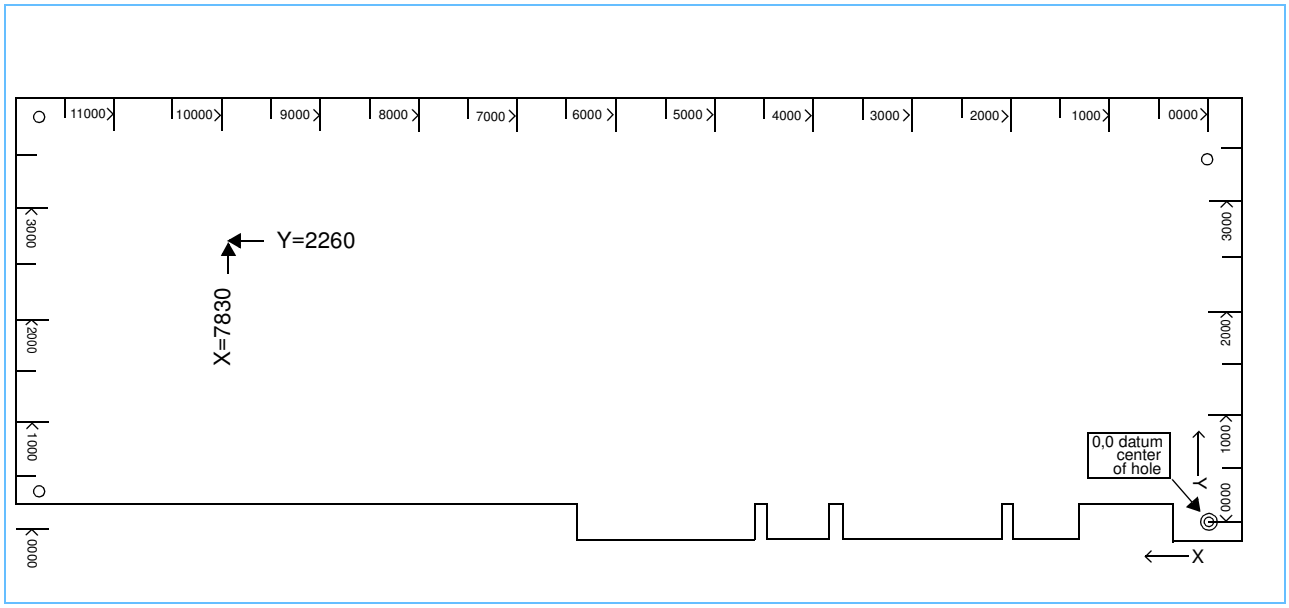



*Figure 12-3. Board Location Grid—Bottom View*



## 12.2 Debugging Tools

*Table 12-3* identifies components that are recommended as aids for hardware monitoring. None of these components are shipped in the board package.

*Table 12-3. Debugging Tools*

| Quantity | Description |
|---|---|
| 1 | 184 Pin DIMM Analysis Probe<br>Vendor 1: FuturePlus Inc.<br>Vendor 1 P/N: FS2330-1 |
| 1 | 64 Bit PCI-X Analysis Probe (Passive)<br>Vendor 1: FuturePlus Inc.<br>Vendor 1 P/N: FS2005 |
| 3 | High Density Termination Adapter<br>Vendor 1: HP<br>Vendor 1 P/N: E5346A |

## 12.3 Auxiliary Materials

*Table 12-4* identifies components that are included in the board package, but are not listed in the board BOM (see *Table 12-5*). Examples of this type of component are the jumpers (shunts) installed on the board to configure its functional characteristics.

*Table 12-4. Auxiliary Materials in Kit*

| Quantity | Part Number | Supplier | Description |
|---|---|---|---|
| 2 | AM29LV040B-70JC | AMD | Flash EEPROM, 512K x 8, 3V |
| 1 | 1-382811-6 | AMP/Tyco | Standard shunt, 2.54mm |
| 4 | 86730-001 | FCI-Berg | Miniature shunt, 2mm |
| 2 | 31D1-164000 | CableWholesale | DB9 to RJ12 adapter cable |
| 2 | H2663-07-ND | Digi-Key | RJ12 modular cable |
| 1 | 68P4504 | IBM | PCI I/O plate |
| 1 | 68P4505 | IBM | System controller module heat sink |
| 1 | 68P4513 | IBM | System controller module backplate |
| 1 | 68P4509 | IBM | System controller module backplate insulator |
| 2 | 68P4506 | IBM | Processor module heat sink |
| 1 | 68P4512 | IBM | Processor module backplate |
| 1 | 68P4510 | IBM | Processor module top insulator |
| 1 | 68P4511 | IBM | Processor module backplate insulator |
| 1 | EFB0405HHA | Delta Fans | Fan, 40x40x10, 5VDC |
| 12 | 91771A059 | McMaster-Carr | Mounting screw, 1/2 in., 0-80, Flathead |
| 2 | 91241A412 | McMaster-Carr | Mounting screw, 1/4 in., 4-40, Flathead |
| 4 | 90353A148 | McMaster-Carr | Mounting screw, M3-20, Panhead |
| 4 | 95610A130 | McMaster-Carr | Washer, M3, nylon |

## 12.4 Board Bill of Materials

*Table 12-5* provides a complete list of all components used to manufacture the board assembly. In addition, all components that appear in the schematics but that are not actually assembled on the board (DNP column contains TRUE) appear in this table.

*Table 12-5. Evaluation Board Bill of Materials*

| DNP | Value | Qty | Ref Des | JEDEC Type | Vendor | Vendor PN | ALT1 Vendor | ALT1 Vendor PN | ALT2 Vendor | ALT2 Vendor PN |
|---|---|---|---|---|---|---|---|---|---|---|
| | 33U | 24 | C1-C5, C43, C66, C67, C69-C71, C75, C78, C81, C82, C98-C101, C242, C283, C493, C496, C498 | SMEC2816R | SPRAGUE | 293D336X0020D2T | NEC | NRD336M20RBM | AVX | TAJD336M020R |
| | 0.1U_10V | 385 | C6-C14, C16-C25, C27-C42, C44-C63, C65, C77, C86, C87, C89, C93, C104-C118, C120-C122, C125, C126, C128-C132, C137-C139, C143-C148, C151-C155, C158, C169, C171, C173-C181, C188-C200, C203-C205, C207, C208, C213-C222, C225-C228, C232-C241, C243-C247, C249-C272, C274-C276, C279-C282, C284-C311, C313, C316-C330, C333-C337, C339, C341-C343, C345-C350, C358-C370, C372-C374, C376-C378, C380, C382-C395, C397-C410, C412-C438, C441-C457, C459, C460, C462-C481, C483, C484, C486-C490, C492, C494, C495, C497, C499-C509 | SMEC0402 | AVX | 0402ZC104MAT2A | - | - | - | - |
| | 0.1U | 10 | C102, C133-C136, C159, C160, C165, C166, C312 | SMEC0402 | AVX | 0402ZC104MAT2A | - | - | - | - |
| | 0.01U | 6 | C79, C97, C103, C119, C344, C371 | SMEC0402 | KEMET | C0402C103K3RAC | - | - | - | - |
| | 12P | 6 | C73, C76, C95, C96, C123, C124 | SMEC0402 | AVX | 04025A120KAT2A | - | - | - | - |
| | 22U | 6 | C127, C170, C248, C340, C458, C491 | SMEC1206R | PANASONIC | ECS-T0JY226R | - | - | - | - |
| | 10U | 25 | C15, C26, C64, C72, C84, C140-C142, C149, C150, C172, C223, C278, C331, C332, C338, C353, C354, C356, C357, C396, C439, C440, C461, C485 | SMEC0805R | AVX | T491R106M006AS | - | - | - | - |
| | 100P | 17 | C156, C157, C161-C164, C167, C168, C186, C187, C201, C202, C211, C212, C229, C230, C315 | SMEC0402 | PHYCOMP | 0402CG101J9B200 | PANASONIC | ECJ0EC1H101J | - | - |
| | 0.1U | 12 | C182, C184, C206, C210, C231, C277, C314, C351, C355, C375, C381, C482 | SMEC0603 | PANASONIC | ECJ-1VB1C104K | KEMET | C0603C104K4RACTU | YAGEO | 06032R104K7B20D |

*Table 12-5. Evaluation Board Bill of Materials (Continued)*

| DNP | Value | Qty | Ref Des | JEDEC Type | Vendor | Vendor PN | ALT1 Vendor | ALT1 Vendor PN | ALT2 Vendor | ALT2 Vendor PN |
|---|---|---|---|---|---|---|---|---|---|---|
| | 2.2U | 4 | C183, C209, C352, C379 | SMC0805R | TAIYO YUDEN | CEJMK212BJ225KG-T | TDK | C2012X5R0J225KT000N | MURATA | GRM21BR60J225KC01L |
| | 0.001U | 4 | C74, C85, C92, C185 | SMEC0603 | AVX | 06035C102JAT2A | KEMET | C0603C102J5RAC | MURATA | GRM39X7R102J050AD |
| | 330U | 2 | C224, C273 | SMEC2816R | KEMET | T495X337M006AS4823 | - | - | - | - |
| | 820U | 1 | C68 | CAP_F12 | SANYO | 2SV820M | SANYO | 6SVP820M | - | - |
| | DIODE3PIN | 2 | CR2, CR3 | SOT23 | ON SEMICONDUCTOR | BAS16LT1 | ROHM | BAS16 SOT-23 | - | - |
| | | 3 | DS1, DS2, DS5 | SMC_LED3 | PANASONIC | LN1351C | - | - | - | - |
| | | 1 | DS3 | SMC_LED3 | PANASONIC | LN1251C | - | - | - | - |
| | | 1 | DS4 | SMC_LED3 | PANASONIC | LN1451C | - | - | - | - |
| | | 1 | J11 | HDRM2X8_JTAGCLIP | AMP-TYCO | 147062-3 | - | - | - | - |
| | RJ11/12_DUAL_SHIELD | 1 | J13 | RJ11_1X2 | MOLEX | 44248-0093 | - | - | - | - |
| | | 3 | J14, J15, J19 | CONN_MICTOR38 | AMP | 2-767004-2 | - | - | - | - |
| | | 2 | J8, J16 | BERG2X1 | AMP | 104350-1 | - | - | - | - |
| | RJ45ENETXFRM_LED | 1 | J20 | RJ45_1X2ENET | PULSE | J8064D628A | - | - | - | - |
| | | 1 | J22 | HDR2X4_2MM_SM | FCI | 57202-G52-04 | - | - | - | - |
| | PCIGOLDTABUNIVERSAL94 | 1 | J25 | GOLDTAB_PCI_UNIV94 | - | - | - | - | - | - |
| | | 1 | J26 | HDR2X5A | AMP | 104352-5 | - | - | - | - |
| | | 1 | J34 | CONN2X10_VHDR_PEGS | MOLEX | 39-29-9202 | - | - | - | - |
| | HEADER1X3_SHROUD | 1 | J4 | CONN_CST100_3 | AMP | 644892-3 | - | - | - | - |
| | 1K@100Mhz | 16 | L1-L3, L5, L7-L10, L15-L20, L22, L25 | SMEC0603 | FAIR-RITE CORP | 2506031027Y0 | VISHAY | ILBB-0603-1000 | - | - |
| | 0U | 8 | L11-L14, L21, L23, L28, L30 | SMC0805R | MURATA | BLM21PG300SN1 / BLM21P300SPT | - | - | - | - |
| | 17@100Mhz | 4 | L24, L26, L31, L32 | SMEC0805R | VISHAY DALE | ILBB0805RK170V | - | - | - | - |

*Table 12-5. Evaluation Board Bill of Materials (Continued)*

| DNP | Value | Qty | Ref Des | JEDEC Type | Vendor | Vendor PN | ALT1 Vendor | ALT1 Vendor PN | ALT2 Vendor | ALT2 Vendor PN |
|---|---|---|---|---|---|---|---|---|---|---|
| | 70@100 Mhz | 2 | L27, L29 | SMC1206R | MURATA | BLM31AF700SN1 | - | - | - | - |
| | 3.3U | 1 | L4 | D03316 | COILCRAFT | D03316P-332HC | PULSE_ENG | P0751.332T | GOWANDA | SMP3316-331M \| |
| | 60@100 Mhz | 1 | L6 | SMC1210 | TDK | HF50ACB-322513-T | - | - | - | - |
| | 4.7K | 108 | R1, R2, R54, R115, R230, R231, R238, R239, R268, R290, R315-R317, R331, R333, R337-R341, R351, R364, R365, R367-R370, R374-R383, R385-R387, R390-R395, R397, R399, R403-R406, R408-R410, R412, R415, R420, R421, R434, R435, R443, R446, R448, R449, R451, R453-R456, R458-R462, R464-R470, R472-R474, R476-R479, R481-R484, R487-R492, R496, R497, R503-R506, R529-R531 | SMER0402 | PANASONIC | ERJ2GEYJ472V | PHYCOMP | 232270570472 | - | - |
| | 22 | 100 | R5-R14, R17-R52, R55-R62, R64-R81, R83, R85-R92, R235, R245, R249, R260, R276, R279, R280, R285, R286, R292, R302, R407, R414, R417, R422, R429, R436, R439, R441 | SMER0402 | PHYCOMP | 9C1A040222R0FLHF3 | - | - | - | - |
| | 120 | 7 | R82, R96-R100, R330 | SMER0402 | YAGEO AMERICA | 9C04021A1100FLHF3 | DIGIKEY | 311-120LCT-ND | - | - |
| | 3.30K | 37 | R93-R95, R101, R103-R105, R107, R109, R318-R324, R326-R329, R344, R346, R348, R350, R356, R366, R398, R402, R411, R413, R416, R418, R430, R433, R437, R440, R442 | SMER0402 | PHYCOMP | 9C04021A3301FLHF3 | - | - | - | - |
| | 10.0K | 7 | R102, R254, R493-R495, R522, R535 | SMER0402 | PANASONIC | ERJ2RKF1002X | PHYCOMP | 9C1A04021002FLHF3 | - | - |
| | 24.90 | 113 | R106, R108, R110-R113, R117, R118, R120, R121, R125-R223, R233, R240, R241, R355 | SMER0402 | PANASONIC | ERJ2RKF24R9V | PHILIPS | - | ROHM | MCR01EZPFX24R9 |
| | 100 | 10 | R114, R122-R124, R236, R244, R246, R248, R521, R523 | SMER0402 | PHYCOMP | 232270570101 | DIGIKEY | 311-100JTR-ND | YAGEO | 9C04021A1000JLHF3 |

*Table 12-5. Evaluation Board Bill of Materials (Continued)*

| DNP | Value | Qty | Ref Des | JEDEC Type | Vendor | Vendor PN | ALT1 Vendor | ALT1 Vendor PN | ALT2 Vendor | ALT2 Vendor PN |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1.0K | 22 | R116, R237, R308, R309, R332, R352, R424, R426, R427, R431, R438, R444, R445, R447, R450, R520, R524, R527, R528, R532, R534, R537 | SMER0402 | PANASONIC | ERJ2RKF1001X | PHYCOMP | 9C1A04021001FLHF3 | - | - |
| | 200 | 5 | R3, R4, R15, R16, R533 | SMER0805R | PANASONIC | ERJ6GEYJ201V | PHILIPS | 232273061201 | ROHM | MCR10EZHJ201 |
| | 0 | 6 | R224, R226, R267, R271, R307, R312 | SMER1206R | PANASONIC | ERJ8GEYOR00V | ROHM | MC18EZHJ000 | AVX | CJ32-000-T |
| | 100 | 3 | R243, R256, R257 | SMER0805R | PANASONIC | ERJ6ENF1000V | ROHM | MCR10EZHF1000 | PHILIPS | 232273461001 |
| | 1K | 1 | R247 | SMER0805R | ROHM | MCR10EZHJ102 | PANASONIC | ERJ6GEYJ102V | PHILIPS | 232273061102 |
| | 1.30K | 1 | R251 | SMER0402 | PANASONIC | ERJ-2RKF1301X | DIGIKEY | P1.30KLCT-ND | - | - |
| | 10K | 1 | R252 | SMER0805R | PANASONIC | ERJ6GEYJ103V | ROHM | MCR10EZHJ103 | PHILIPS | 232273061103 |
| | 200 | 4 | R253, R259, R310, R314 | SMER0402 | YAGEO | 9C04021A2000JLHF3 | PANASONIC | ERJ-2GEJ201X | DIGIKEY | P200JCT-ND |
| | 33.2 | 2 | R261, R262 | SMER0402 | PANASONIC | ERJ-2RKF33R2X | - | - | - | - |
| | 49.9 | 10 | R264, R265, R269, R270, R289, R296, R301, R305, R306, R463 | SMER0402 | PHYCOMP | 9C1A040249R9FLHF3 | PANASONIC | ERJ-2RKF49R9X | - | - |
| | 340K | 2 | R273, R313 | SMER0402 | VENKEL | CR0402-16W-3403FT | - | - | - | - |
| | 0 | 15 | R278, R281, R284, R287, R299, R359, R372, R373, R396, R400, R401, R419, R508, R509, R538 | SMER0402 | PANASONIC | ERJ2GE0R00 | - | - | - | - |
| | 10 | 1 | R325 | SMER0805R | PHILIPS | 232273061109 | PANASONIC | ERJ6GEYJ100V | ROHM | MCR10EZHJ100 |
| | 1.5K | 14 | R334-R336, R342, R343, R345, R347, R349, R357, R358, R360-R362, R371 | SMER0402 | PHYCOMP | 9C04021A1501FLHF3 | PANASONIC | ERJ-2RKF1501X | - | - |
| | 470 | 4 | R353, R354, R388, R471 | SMER0402 | PANASONIC | ERJ2GEYJ471V | YAGEO | 9C04021A4700JLHF3 | DIGIKEY | ERJ-2GEJ471X |
| | 100K | 1 | R363 | SMER0805R | PANASONIC | ERJ6GEYJ104V | ROHM | MCR10EZHJ564 | AVX | CR21-564J-T |
| | 12.1K | 1 | R384 | SMER0402 | DIGIKEY | 311-12.1KLTR-ND | YAGEO | 9C04021A1212FLHF3 | - | - |
| | 10K-5% | 5 | R423, R425, R428, R432, R536 | SMER0402 | PANASONIC | ERJ-2GEJ103X | PHYCOMP | 9C1A04021002JLHF3 | - | - |
| | 36 | 1 | R480 | SMER0402 | YAGEO AMERICA | 9C04021A36R0FLHF3 | - | - | - | - |
| | 10 | 2 | R485, R486 | SMER0402 | PANASONIC | ERJ2GEJ100X | PHYCOMP | 9C1A040210R0JLHF3 | - | - |
| | 100K | 1 | R53 | SMER0402 | PANASONIC | ERJ2RKF1003X | PHYCOMP | 9C1A04021003FLHF3 | - | - |

*Table 12-5. Evaluation Board Bill of Materials (Continued)*

| DNP | Value | Qty | Ref Des | JEDEC Type | Vendor | Vendor PN | ALT1 Vendor | ALT1 Vendor PN | ALT2 Vendor | ALT2 Vendor PN |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 38 | TP1-TP3, TP6-TP9, TP11-TP25, TP27-TP39, TP41-TP43 | TP014 | DO NOT ORDER | DO NOT ORDER | - | - | - | - |
| | | 1 | U1 | 750FXUPPER | IBM | PPC750FX-GB2533T | - | - | - | - |
| | 256MBIT_16KX16 | 8 | U10, U11, U13, U14, U44, U45, U47, U48 | TSOP2_66P65 | INFINEON | HYB25D256160BT-7 | - | - | - | - |
| | 256MBIT_32KX8 | 2 | U12, U46 | TSOP2_66P65 | INFINEON | HYB25D256800BT-7 | - | - | - | - |
| | | 2 | U15, U22 | PLCC_SOCKET_32_SMT | AMP | 822498-1 | - | - | - | - |
| | ST16C2552CJ | 1 | U16 | PLCC44 | EXAR | ST16C2552CJ | - | - | - | - |
| | | 4 | U17, U24, U30, U35 | DIP_SMCSWITCH8_P50 | ALCOSWITCH | GDH08S | - | - | - | - |
| | | 2 | U18, U23 | PWR_VT_FLORENCE_14 | AMBIT | SFE-007G025 | ACBEL | API2VR25-030 | - | - |
| | | 2 | U19, U20 | TSSOP48P5 | FAIRCHILD | 74VCX16373MTD | - | - | - | - |
| | | 1 | U2 | 750FXLOWER | IBM | PPC750FX-GB2533T | - | - | - | - |
| | | 2 | U9, U21 | TSSOP16P65 | MAXIM | MAX3232CUE | - | - | - | - |
| | CLOCK_BUF_1TO4 | 1 | U25 | SOE08_280R | ICS | ICS553MI | - | - | - | - |
| | | 2 | U26, U28 | EHOLE | DO NOT ORDER | DO NOT ORDER | - | - | - | - |
| | ML6554 | 1 | U27 | SOE16_280R_HS | MICRO LINEAR | ML6554CU | - | - | - | - |
| | EPM7256BTC100-5 | 1 | U29 | TQFP100P50 | ALTERA | EPM7256BTC100-5 | - | - | - | - |
| | | 1 | U3 | EPBGA_34X34_724SOCKET | MARVELL TECHNOLOGY GROUP | MV64360-A2-BAY-C133 | - | - | - | - |
| | | 1 | U31 | PWR_VT_FLORENCE_14 | AMBIT | SFE-007G018 | ACBEL | API2VR24-030 | - | - |
| | CLOCK_MULT | 1 | U32 | SOE16_280R | ICS | ICS670M-01 | - | - | - | - |
| | | 1 | U33 | SSOP28P635_282 | ICS | MK74CB218R | - | - | - | - |
| | | 2 | U34, U40 | PWR_VT_FLORENCE_14 | AMBIT | SFE-007G015 | - | - | - | - |
| | | 2 | U36, U55 | SOE08_280R | MICROCHIP | 24LC64-I/SN | - | - | - | - |

*Table 12-5. Evaluation Board Bill of Materials (Continued)*

| DNP | Value | Qty | Ref Des | JEDEC Type | Vendor | Vendor PN | ALT1 Vendor | ALT1 Vendor PN | ALT2 Vendor | ALT2 Vendor PN |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | U37 | QFPE100RP65 | BROADCOM | BCM5222KQM | - | - | - | - |
| | C9531AT | 1 | U38 | TSSOP28_P65 | IMI | C9531AT | IMI | IMI9531CT | | |
| | | 1 | U39 | SMA | AMP | 221789-1 | - | - | - | - |
| | | 1 | U4 | SOT223A | DALLAS SEMICONDUCTOR | DS1233AZ-10 | - | - | - | - |
| | | 2 | U6, U41 | TSOP56P5 | INTEL | E28F128J3A-150 | - | - | - | - |
| | | 1 | U42 | SOE28_450R | RAMTRON | FM18L08-70-S | - | - | - | - |
| | | 2 | U7, U43 | TSOP2_32 | HITACHI | HM62V8512BLTT-7 | SAMSUNG | KM68V4000CLT-7L | MITSUBISHI | M5M5V408BTP-70L |
| | 74LVC1G17 | 1 | U49 | SOT235 | TEXAS INSTRUMENTS | SN74LVC1G17DBVT | TEXAS INSTRUMENTS | SN74LVC1G17DBVR | - | - |
| | | 2 | U5, U53 | SPDTF | APEM | A2216 | - | - | - | - |
| | | 1 | U51 | SOE14_280R | FAIRCHILD | MM74HC74AMX | MOTOROLA | MC74HC74AD | - | - |
| | | 1 | U52 | SOT235 | TI | SN74LVC1G06DBVR | TI | SN74LVC1G06DBVT | - | - |
| | 74HC1G14 | 1 | U54 | SOT235 | PHILIPS SEMICONDUCTORS | 74HC1G14GV | FAIRCHILD | NC7S14M5X | ALLIED ELECTRONICS | 263-0257 |
| | 74LVC1G14 | 1 | U56 | SOT235 | PHILIPS SEMICONDUCTORS | 74LVC1G14GV | TI | SN74LVC1G14DBVR | DIGIKEY | 296-11607-1-ND |
| | 74LVC1G125 | 1 | U57 | SOT235 | PHILIPS SEMICONDUCTORS | 74LVC1G125GV | TI | SN74LVC1G125DBVR | DIGIKEY | 296-11603-1-ND |
| | 120 | 1 | U8 | TSSOP48P5 | ICS | ICS93V857xG-025 | - | - | - | - |
| | | 1 | Y1 | SG710 | EPSON | SG-710ECK-25.000MB | - | - | - | - |
| | 3.6864MHZ | 1 | Y2 | SG636 | EPSON | SG-636PCE3.6864C | - | - | - | - |
| | MA-306-33.3333M-C | 1 | Y3 | MA306 | EPSON | MA-306-33.3333M-C | - | - | - | - |
| TRUE | 100P | 3 | C91, C94, C411 | SMEC0402 | PHYCOMP | 0402CG101J9B200 | PANASONIC | ECJ0EC1H101J | - | - |
| TRUE | 12P | 4 | C80, C83, C88, C90 | SMEC0402 | AVX | 04025A120KAT2A | - | - | - | - |
| TRUE | | 17 | J1, J2, J7, J9, J10, J12, J17, J18, J23, J24, J36, J40-J45 | BERG1X1_LG | KEYSTONE | 5001 | - | - | - | - |
| TRUE | | 4 | J27-J30 | BERG2X1 | AMP | 104350-1 | - | - | - | - |

*Table 12-5. Evaluation Board Bill of Materials (Continued)*

| DNP | Value | Qty | Ref Des | JEDEC Type | Vendor | Vendor PN | ALT1 Vendor | ALT1 Vendor PN | ALT2 Vendor | ALT2 Vendor PN |
|---|---|---|---|---|---|---|---|---|---|---|
| TRUE | | 3 | J31-J33 | HDR1X3_MTA 100 | AMP | 640452-3 | - | - | - | - |
| TRUE | HEADER _4_VERT | 3 | J37-J39 | HDR_4X1_100 | MILL-MAX | 800-10-004-10-001 | DIGIKEY | ED7204-ND | - | - |
| TRUE | 0 | 27 | R63, R84, R119, R225, R228, R229, R242, R250, R263, R266, R274, R275, R277, R282, R288, R291, R293-R295, R297, R298, R304, R507, R510-R513 | SMER0402 | PANASONIC | ERJ2GE0R00 | - | - | - | - |
| TRUE | 1.0K | 4 | R227, R389, R501, R502 | SMER0402 | PANASONIC | ERJ2RKF1001X | PHYCOMP | 9C1A04021001F LHF3 | - | - |
| TRUE | 909K | 4 | R232, R234, R255, R258 | SMER0402 | PANASONIC | ERJ2RKF9093X | - | - | - | - |
| TRUE | 1.21M | 2 | R272, R311 | SMER0402 | VENKEL | CR0402-16W-1214FT | - | - | - | - |
| TRUE | 4.7K | 15 | R283, R452, R457, R475, R498-R500, R514-R519, R525, R526 | SMER0402 | PANASONIC | ERJ2GEYJ472V | PHYCOMP | 232270570472 | - | - |
| TRUE | 22 | 2 | R300, R303 | SMER0402 | PHYCOMP | 9C1A040222R0FL HF3 | - | - | - | - |
| TRUE | | 3 | TP4, TP5, TP10 | TP020 | DO NOT ORDER | DO NOT ORDER | - | - | - | - |
| TRUE | | 1 | U50 | TSSOP24_P65 | PHILIPS | PCF8575S | - | - | - | - |
| TRUE | 133.33M HZ | 1 | Y4 | OSC_5X7MM | EPSON | EG-2001CA133.0000M -PCHL3 | EG-2001CA133.0000M-PCHLB | - | - | - |
| TOTAL | | 1228 | | | | | | | | |

# Index

## N



## O



## P



## R



## S



## V

# Revision Log

| Revision Date | Contents of Modification |
|---|---|
| 06/10/03 | Initial creation of the book. |