ICP2432
Hardware Description and
Theory of Operation

DC 900-1501A

SIMPACT

Contents

List of Figures

List of Tables

Preface

Purpose of Document

This manual describes Simpact's ICP2432 front-end communications processor, its architecture, and how it works in an PCIbus system.

Intended Audience

This document should be read by maintenance technicians, computer system integrators, and software developers who need detailed information about the operating theory and features of the ICP2432 hardware.

Required Equipment

The ICP2432 board mounts in any full PCIbus slot that supports bus mastering.

Organization of Document

If you are not familiar with Simpact's front-end communications processors, you should read Chapter 1 and Chapter 2. If you are familiar with front-end communications processors, you may wish to skip these overviews and go directly to the detailed technical descriptions in Chapter 3.

Chapter 1 contains a functional overview of the ICP2432.

Chapter 2 describes the basic operation of the ICP2432 in a system.

Chapter 3 contains a detailed theory of operation with hardware descriptions.

References

Central Processor Unit Reference Manual, CPU32+RM/AD, Motorola, Inc.

ICP2432-to-PCI Host Protocol Specification, Simpact, Inc., DC-900-1509

MC68349 Integrated Processor with DMA User's Manual, MC68349UM/AD, Motorola Inc.

S5933 PCI Controller Data Book, AMCC

Z16C30/Z16C32 IUSC User's Manual, Zilog, Inc.

Document Conventions

The following conventions apply throughout this document:

- A signal name that appears with an overline, for example, $\overline{\text{ZAS}}$, indicates that the signal is asserted low true.

- "Communications bus" is equivalent to "Z-BUS."

- Bits are numbered from right to left, beginning with zero. Bit zero is the low-order bit.

- Hexadecimal values are shown with the notation (hex). The underscore (_) is for readability purposes only.

Revision History

The revision history of the *ICP2432 Hardware Description and Theory of Operation*, Simpact document DC900-1501, is recorded below:

Document Revision	Release Date	Description
DC 900-1501A	February 1998	Original release

Customer Support

If you are having trouble with any Simpact product, call us at 1-800-275-3889 Monday through Friday between 8 a.m. and 5 p.m. Pacific time.

You can also fax your questions to us at (619) 560-2838 or (619) 560-2837 any time. Please include a cover sheet addressed to "Customer Service."

We are always interested in suggestions for improving our products. You can use the report form in the back of this manual to send us your recommendations.

ICP2432 Overview

The ICP2432 intelligent communications processor is an ancillary computer dedicated to the processing of communications-related data.

The ICP2432 hardware is a single-board computer with a central processing unit (CPU), random access memory (RAM), programmable read-only memory (PROM), and input/output (I/O) circuitry. The software consists of an onboard operating system, diagnostic tests, host interface drivers, and application routines.

1.1 Purpose of the ICP2432

The main purpose of the ICP2432 is to improve the overall computing efficiency of the host computer. To do this, low-level communications tasks traditionally performed by the host central processor are migrated to the ICP.

An ICP increases overall system bandwidth by distributing the I/O processing away from the host CPU. In the traditional minicomputer or microcomputer architecture, the host services all I/O requests. This load on the CPU has grown steadily as computer peripherals have become increasingly more powerful. Modern operating systems allow intelligent front-end processors to perform these relatively simple tasks. The result is an overall increase in system throughput.

Table 1–1 sums up the attributes of the ICP.

Table 1–1: ICP Hardware Features and Benefits

Features	Benefits
68349 integrated processor	4-gigabyte linear address space Allows use of familiar, powerful program development tools (assemblers, C compiler, editors) Integrated DMA support for PCI transfers
S5933 PCI interface	32-bit master/slave interface PCI revision 2.1 compliant
Large RAM memory	Space for application tasks and real-time executive Applications are downloaded from the host rather than from ROM to make it easier to update software
Individually programmable ports	One ICP services lines with different characteristics such as parity, data rate, synchronous *vs.* asynchronous, CRC checking Full modem support Flexibility in changing environments
DMA-controlled data transfer	Full duplex support for high-speed applications
Multiple electrical interfaces	Supports EIA-232 (V.28), EIA-422 (V.11 and X.27), EIA-423 (V.10 and X.26), EIA-449, EIA-485, EIA-530, EIA-562, MIL-STD-188C, MIL-STD-188-114A Unbalanced, MIL-STD-188-114A Balanced Type II, and V.35
Single-module form factor	Easy mounting in standard backplane

1.2 ICP2432 Applications

The ICP2432 is a general-purpose computer that can do many different tasks. At system startup, the ICP2432 gains its run-time personality from downloaded application software. The application software may be customer-specific or part of an ICP2432 turnkey product.

Communications software systems that use the ICP2432 can fit the International Standards Organization (ISO) open system interconnection (OSI) seven-layer reference model for communications protocols. Figure 1–1 shows this model. In most cases the ICP is programmed to perform up to the first three levels of this model.

Once the software download has been completed, the ICP2432 hardware and software combination performs its assigned task until a board-level reset initiates another startup sequence.

	Layer Number	Function
APPLICATION	7	Services applications
PRESENTATION	6	Code conversion, data format
SESSION	5	Coordinates interaction between applications
TRANSPORT	4	End-to-end data integrity
NETWORK	3	Routes information
DATA LINK	2	Exchanges data with physical link
PHYSICAL	1	Transmits/receives bit stream to medium

3042

Figure 1–1: ISO OSI Reference Model

Chapter

2

System Operations

This chapter describes the ICP2432 hardware and its operation in host systems.

2.1 ICP2432 Architecture Overview

The ICP2432 computer provides high-performance, front-end processing in communications environments as shown in Figure 2–1. The processor bus consists of a CPU, memory, and PCI interface and support devices. This bus is used to execute the application program and provide computing resources as required by high-level protocols.

The communications bus (the subsidiary bus) consists of the communications devices with integral DMA controllers. This bus is used for the movement of data between the serial ports and memory buffers. These two independent buses are bridged by circuitry that translates appropriate signals to control transactions between them during inter-bus operations.
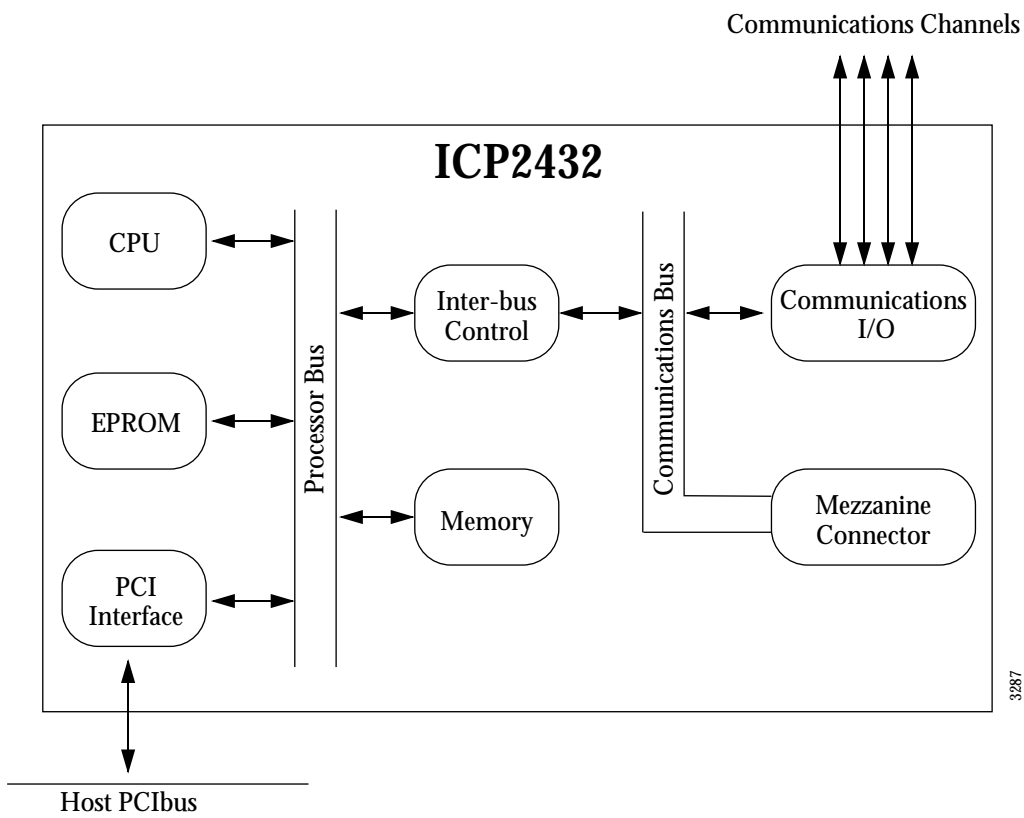
Figure 2–1: ICP2432 Architecture Overview

2.2 General Operation

Operation of the ICP2432 occurs in two phases: startup and routine operation. When power is initially applied to the ICP2432, its CPU begins execution at a memory address in the PROM device. This location contains instructions to set up the basic working environment for the CPU.

2.2.1 Startup

Startup begins with the execution of diagnostic firmware that tests the major sections of the ICP2432 hardware. Upon successful completion of these tests, the PCI interface is initialized and the host downloads the ICP's operating system and application program software, through the PCIbus interface circuits, into the ICP memory. After these programs have been downloaded, the CPU begins to execute the downloaded software.

2.2.2 Routine Operation

During routine operation, the ICP2432 does all the communications-related work. The software program downloaded during startup is used to operate the general-purpose ICP2432 communications hardware in the mode required by the host system. This program is typically an implementation of some special purpose communications protocol such as X.25, SDLC, or ADCCP.

The ICP2432 remains in the routine phase of operation until a board reset is issued by the host software, the host-bus hardware, or the reset button.

2.2.3 Reset

A hardware reset causes the ICP2432 to stop execution of program instructions. No attempt is made to save the current state of the CPU operation. The hardware reset causes the CPU to go back to the startup phase.

2.3 Operation in the Application Environment

The ICP2432 offloads the host, thus increasing overall system performance. Traditionally, the host's CPU executes the tasks related to the operation of serial communications. These tasks can be extensive, and many require processing in real time. The ICP2432 hardware, combined with application software, is capable of executing many of these protocol-dependent, low-level tasks. The ICP2432 receives the application software download from the host system during the system startup phase.

2.3.1 Communications Traffic

When the software download is complete, communications traffic is accepted and processed for transfer to or from the host. In most applications, the data format used by the host processor is very different from the format transmitted on the serial communications links. Many additional structures must be added to the raw data processed by the host system in order to detect errors, retransmit frames, and maintain data integrity at the data link level. The software programs downloaded into the ICP2432 perform these tasks.

2.3.2 Ports

The ICP2432 is equipped with two, four, or eight high-performance serial communications ports. The ICP2432, when combined with the appropriate distribution cables and/or adapters, can support such electrical interface specifications as EIA-232 (V.28), EIA-422 (V.11 and X.27), EIA-423 (V.10 and X.26), EIA-449, EIA-485, EIA-530, EIA-562, MIL-STD-188C, MIL-STD-188-114A Unbalanced, MIL-STD-188-114A Balanced Type II, and V.35.

Hardware Survey

This chapter describes the ICP2432 architecture, memory map, buses, and devices. Refer to the block diagram of the board, Figure 3–1, throughout the chapter.

3.1 Architecture

The ICP2432 is a single-board computer designed to optimize communications processing for systems with PCIbus support. The base configuration has two high-speed serial ports. Four- and eight-port mezzanine configurations are optional. Devices are not installed in the on-board port locations when a mezzanine board is used. See Figure 3–1.

3.1.1 Dual-bus Architecture

The ICP2432 uses a dual-bus architecture to overcome the limitations that high-speed data lines impose on CPU processing power. The processing engine resides on a processor bus while the serial communications devices with their integral DMA controllers reside on an isolated Z-BUS. Each of these buses is optimized for its primary function. The processor bus has a Motorola MC68349 32-bit processor, longword-wide memory, and a non-multiplexed bus architecture that is optimized for program execution. The communications bus is implemented as a Zilog 16-bit multiplexed bus (Z-BUS) that is optimized for DMA-controlled serial data movement. The 68349 CPU has access to the memory and all devices on both buses, whereas the DMA devices on the Z-BUS have access only to the memory system.
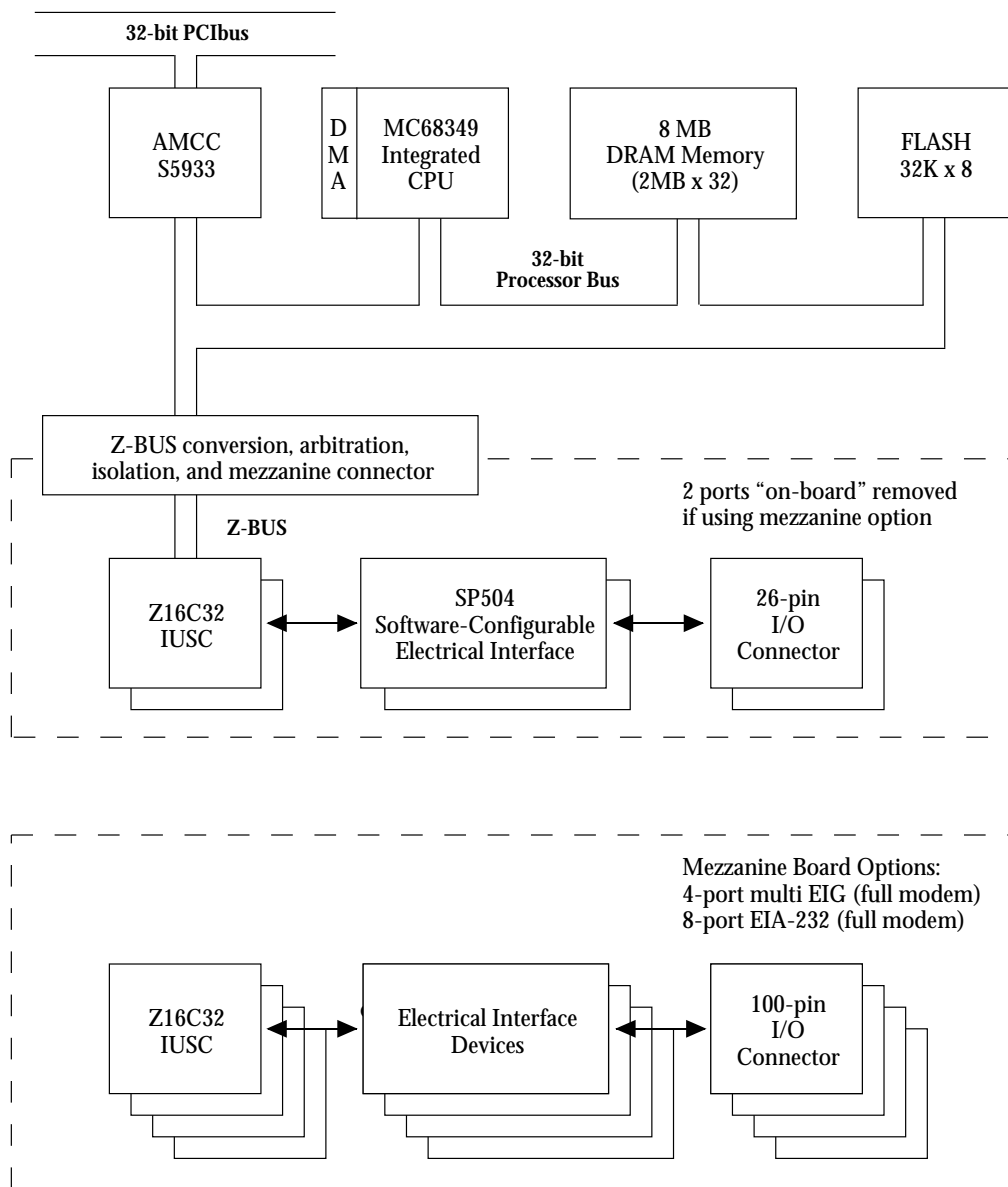
Figure 3–1: ICP2432 Block Diagram

3.1.2 PCI Interface

In most cases, the host machine is the ultimate source or destination of the data being transmitted and received through the serial communications devices. The board requires a mechanism for issuing commands, determining status, and controlling data exchange with the host system in which it resides.

The ICP2432 provides both a 32-bit PCIbus master and bus slave "end point" device (not a bridge). It is a 33 Mhz, +5v interface compliant with Rev 2.1 of the *PCI Local Bus Specification*. See Section 3.5 on page 45 for details.

3.2 Memory Map

The ICP2432 memory map, shown in Figure 3–2, is divided into four main sections: non-volatile memory space, peripheral space, volatile memory space, and Z-BUS space. These main sections are expanded further as illustrated in Figure 3–3 and Figure 3–4 to show detailed addressing information.

3.2.1 Non-volatile Memory

The four-megabyte space beginning at address 000_0000 hex is the non-volatile memory space. See Figure 3–3 on page 27. The ICP2432 FLASH occupies the lowest 64 kilobytes. The additional space is reserved for future expansion.

3.2.2 Peripheral Space

The four-megabyte space beginning at address 040_0000 hex is the peripheral device space. See Figure 3–3 on page 27. To simplify the address decoding logic on the board, each peripheral device is assigned a 64 kilobyte window. The ICP2432 has one assigned device: the PCI interface device (see Section 3.5 on page 45). The additional space is reserved for future expansion.

16 Mbytes	1FF_FFFF Z-BUS Space 100_0000
8 Mbytes	0FF_FFFF Volatile Memory Space 080_0000
4 Mbytes	07F_FFFF Peripheral Space 040_0000
4 Mbytes	03F_FFFF Non-volatile Memory Space 000_0000
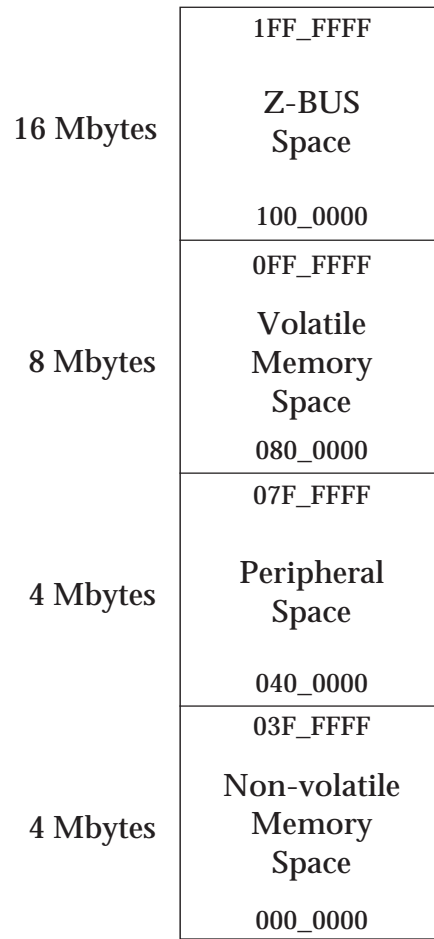
3047

Figure 3–2: ICP2432 Memory Map
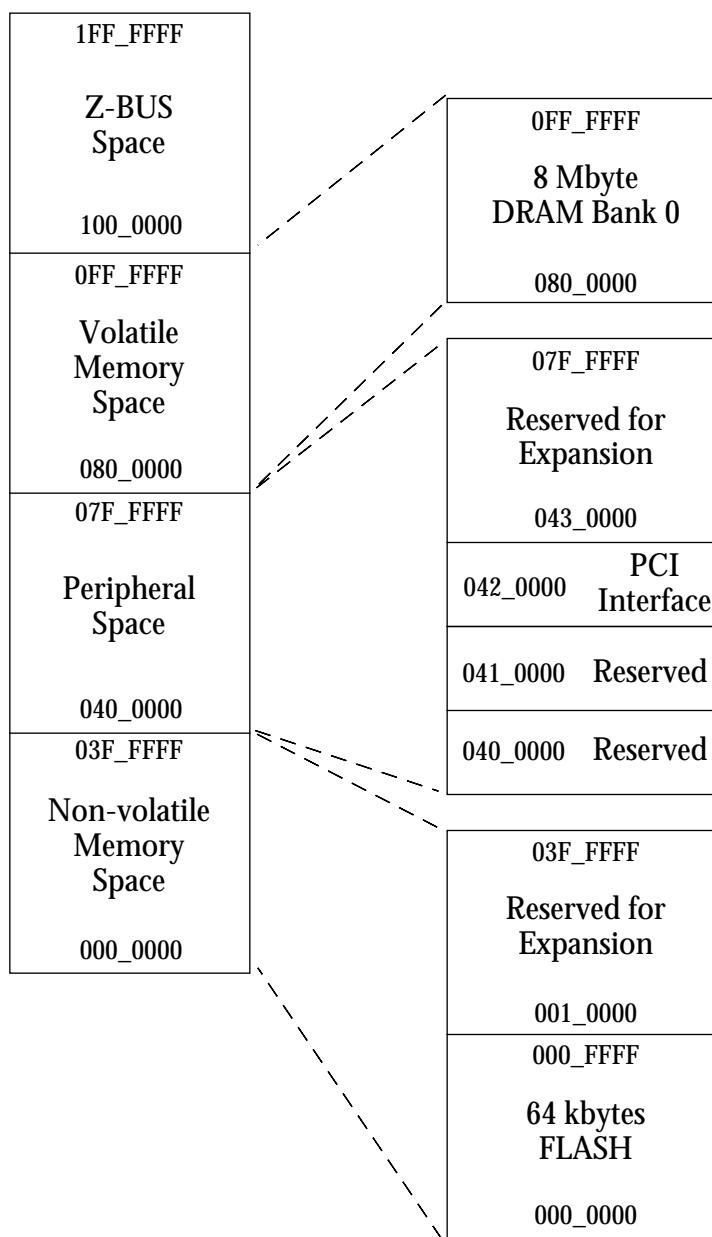
```
                                    ┌──────────────────┐
                                    │ 0FF_FFFF         │
┌──────────────────┐                │                  │
│ 1FF_FFFF         │                │  8 Mbyte         │
│                  │                │  DRAM Bank 0     │
│   Z-BUS          │                │                  │
│   Space          │                │ 080_0000         │
│                  │                └──────────────────┘
│ 100_0000         │
├──────────────────┤                ┌──────────────────┐
│ 0FF_FFFF         │                │ 07F_FFFF         │
│                  │                │                  │
│   Volatile       │                │  Reserved for    │
│   Memory         │                │  Expansion       │
│   Space          │                │                  │
│                  │                │ 043_0000         │
│ 080_0000         │                ├──────────────────┤
├──────────────────┤                │            PCI   │
│ 07F_FFFF         │                │ 042_0000   Interface │
│                  │                ├──────────────────┤
│   Peripheral     │                │ 041_0000   Reserved │
│   Space          │                ├──────────────────┤
│                  │                │ 040_0000   Reserved │
│ 040_0000         │                └──────────────────┘
├──────────────────┤
│ 03F_FFFF         │                ┌──────────────────┐
│                  │                │ 03F_FFFF         │
│   Non-volatile   │                │                  │
│   Memory         │                │  Reserved for    │
│   Space          │                │  Expansion       │
│                  │                │                  │
│ 000_0000         │                │ 001_0000         │
└──────────────────┘                ├──────────────────┤
                                    │ 000_FFFF         │
                                    │                  │
                                    │  64 kbytes       │
                                    │  FLASH           │
                                    │                  │
                                    │ 000_0000         │
                                    └──────────────────┘
                                                    3291
```

Figure 3–3: Detailed Processor Bus Memory Map

3.2.3 Volatile Memory

The eight-megabyte space beginning at address 080_0000 hex is the volatile memory space. See Figure 3–3 on page 27. ICP-standard DRAM occupies the entire eight megabytes of this space.

3.2.4 Z-BUS Space

The 16-megabyte space beginning at address 100_0000 hex is the Z-BUS space. See Figure 3–4 on page 29. This is further divided into specific address sections: a section reserved for memory expansion and the peripheral device section.

3.2.4.1 Reserved Space

The eight-megabyte space beginning at 100_0000 hex is reserved for memory expansion on future mezzanine board products.

3.2.4.2 Peripheral Devices

The eight-megabyte space beginning at 180_0000 hex is used for Z-BUS peripheral devices. To simplify address decoding logic, each device is assigned a four-kilobyte window. These devices are the integrated universal serial controllers (IUSCs) with integral DMA controllers (Section 3.4.1 on page 40), the EIA mode control registers for the Sipex electrical interface devices (Section 3.4.5 on page 44), the input register for the Test Mode inputs (Section 3.4.4 on page 43), and the board ID register (Section 3.4.6 on page 44). The additional space is reserved for future mezzanine board products.
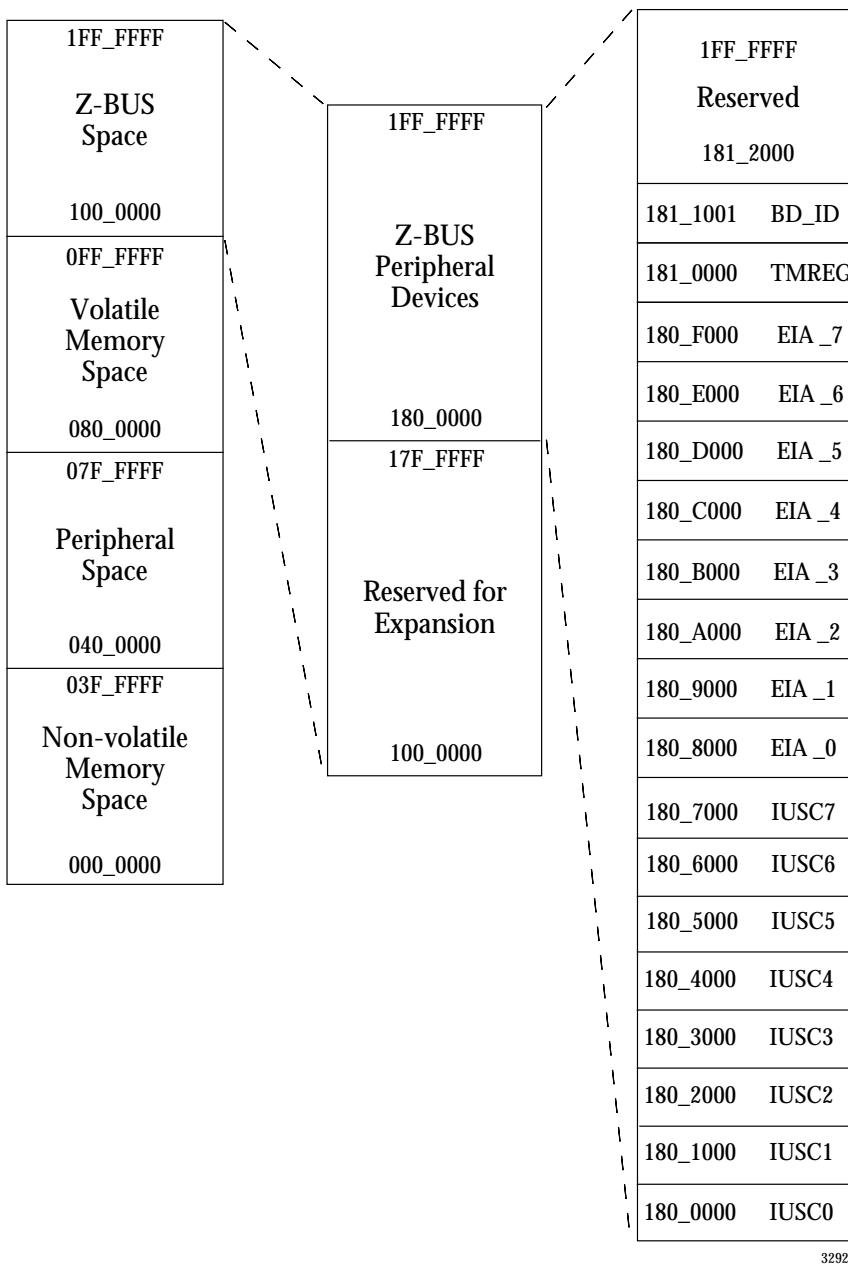
Figure 3–4: Detailed Z-BUS Memory Map

3.3 Processor Bus

The ICP2432 processor bus is composed of the CPU, DRAM, FLASH, PCI interface device, and support circuitry required for timing and control of the bus. Descriptions of the major devices, support circuitry, bus signals, and operation of the processor bus are contained in this section. For clarity, the processor description precedes the descriptions of the bus operation and other devices that connect to this bus.

3.3.1 68349 Integrated Processor

The 68349 processor, shown in Figure 3–5, is a highly integrated device that provides extremely flexible configurations. For complete information on the 68349, see Motorola's *MC68349 Integrated Processor with DMA User's Manual.* The 68349 processor includes the following major functional blocks:

- CPU030 processor

- DMA controller

- Serial controller

- System integration module (SIM)

3.3.1.1 CPU030 Processor

The processing power for the 68349 is provided by the CPU030 central processing unit. The CPU030 delivers full 68030 performance through the use of its three modular elements:

- CPU32+ 32-bit processor

- Configurable instruction cache (CIC)
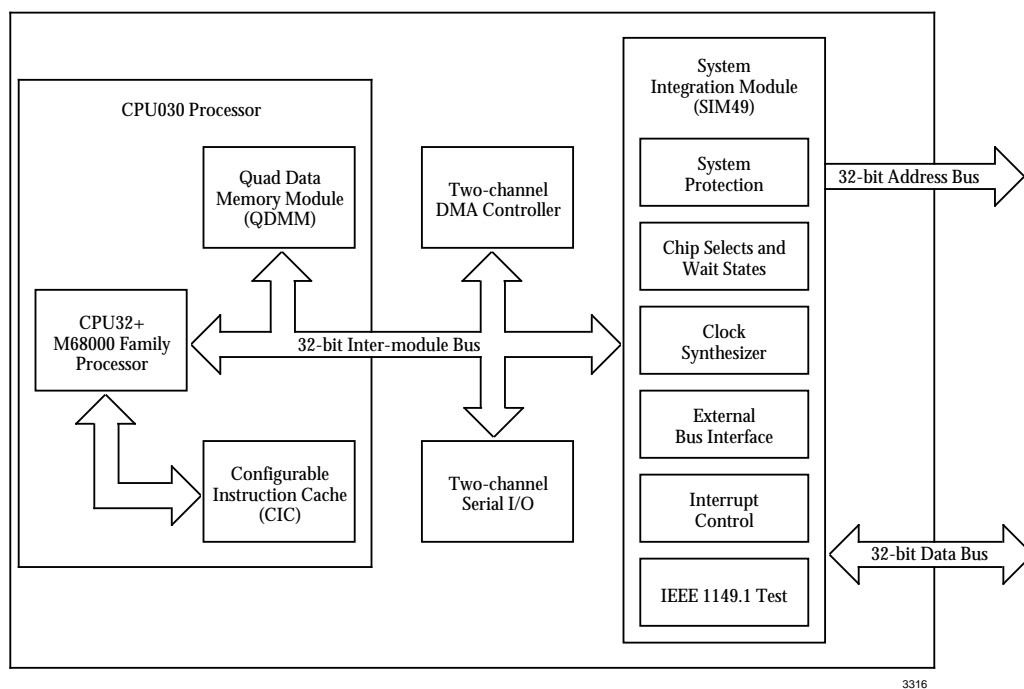
- Quad data memory module (QDMM)

Figure 3–5: 68349 Integrated Processor Block Diagram

CPU32+ Processor

The CPU32+ is a 32-bit execution unit with 32-bit data paths (internal and external). It has a two-clock basic instruction execution rate for 32-bit operation. The CPU32+ is completely software compatible with the 68000.

Configurable Instruction Cache (CIC)

The CIC is a highly configurable memory resource designed to optimize the supply of instructions to the CPU32+. It is closely coupled to the CPU32+, but isolated from the inter-module bus, allowing the processor to continue instruction execution while the IUSCs DMA controllers use the external bus.

The CIC can be configured as either a 1 kilobyte instruction cache or as 2 kilobytes of SRAM. The CIC can be used for high-priority interrupt servicing, O/S routines, or as a general instruction cache.

Quad Data Memory Module (QDMM)

The QDMM provides four independent blocks of 1 kilobyte SRAM. The QDMM can be used as scratchpad memory, stack caches for independent tasks, or parameter storage. As general SRAM, this space can contain instructions for fast access to additional inter-rupt handlers, O/S code, or other frequently accessed code.

3.3.1.2 On-chip Peripherals

The on-chip peripherals described below are independent modules. An inter-module bus (IMB) is used for all on-chip communications. The IMB is similar to a traditional external bus with address, data, clock, interrupt, arbitration, and handshake signals. The IMB ensures that communication between the modules is fully synchronized and that arbitration and interrupts can be handled in parallel with data transfers.

Direct Memory Access Module

The 68349 contains a high-speed 32-bit DMA controller. The DMA module consists of two independently programmable channels. Each channel has separate request, acknowledge, and done signals. Each channel can operate in flyby or dual address mode and supports byte, word, and longword transfers. It supports both burst and cycle steal external request modes. The DMA controller can be configured to release the bus back to the CPU when a high-priority interrupt occurs. The CPU32+ and the DMA controller arbitrate for the bus in parallel with executing bus cycles, typically eliminating all bus arbitration overhead and allowing DMA and CPU bus cycles to occur back-to-back without intervening idle clocks.

The ICP2432 uses the DMA controller to move data between the memory and the FIFOs in the PCIbus interface controller. PCI writes (data transfers from the ICP to the host's main memory) use DMA channel 1 configured for external request, single address, burst writes to the S5933 FIFO. PCI reads (data transfers from the host's main memory to the ICP) use DMA channel 2 configured for external request, single address, cycle-steal reads from the S5933 FIFO.

Serial Module

The 68349 contains two highly configurable full-duplex USARTs.

The ICP2432 uses serial channel A, configured as a 9600 baud asynchronous port, as a console to provide command and status input/output to software development and manufacturing test personnel.

The channel A RTS and CTS pins are used by diagnostic code to determine mode of operation. Manufacturing loopback connectors tie RTSA to CTSA to signify that full manufacturing diagnostics are to be executed.

The channel A RxRDY and TxRDY output pins are used to drive the LED indicators on the I/O panel. See Section 3.3.5 on page 38 for console mechanical and pinout information.

- RxRDYA drives the green LED

- TxRDYA drives the red LED

Serial channel B is not used.

System Integration Module (SIM)

The SIM49 provides the external bus interface for both the CPU32+ and the DMA modules. It also provides programmable circuits to perform address decoding, chip selects, wait-state insertion, interrupt handling, clock generation, bus arbitration, watchdog timing, discrete I/O, and power-on reset timing.

The internal registers for all the modules are contained in a single 4-kilobyte block that is relocatable along 4-kilobyte boundaries. The ICP2432-specific register implementation is integral to the hardware design and should not be modified by user application programs.

3.3.1.3 68349 Power-up Defaults

At power-up or reset, the 68349 registers and the external pins they control are reset to default conditions. The functionality of these pins is configurable and, since external devices and logic are connected to them, unique to each hardware design. A method is required to enable the 68349's first fetches from ROM to operate properly before these SIM control registers are initialized.

The 68349 samples several hardware inputs on the rising edge of RESET for the design-dependent power-up configuration.

The ICP2432 implementation pulls:

- D29 low: selects port A pins as inputs

- D31 high and D30 low: selects byte-wide global chip select

- MODCLK low: selects external clock

3.3.1.4 System Configuration and Protection

After initial power-up, the system configuration is determined by initializing the MCR and AVR registers. These registers control the port pin functions, the debug support, the interrupt arbitration of on-chip modules, and the external interrupts that require auto-vector support. The ICP2432 is configured as follows:

Module Base = EE00_FF00 hex
Clock Mode = External with PLL (external oscillator @ 16 Mhz)

Port	A-7 = out	Driven high (not used)
	A-6 = out	IUSC interrupt acknowledges (IACK_USC)
	A-5 = out	Driven high (not used)
	A-4 = in	Mezzanine-installed indicator (MEZZ_ON)
	A-3 = out	Enable bus master PCI writes (PCI_W_ENB)
	A-2 = out	Enable bus master PCI reads (PCI_R_ENB)
	A-1 = out	Enables IUSC BRs (BR_ENBL)
	A-0 = out	Driven high (not used)
Port	B-7 = in	IRQ 7 for NMI interrupt
	B-6 = in	IRQ 6 for IUSC interrupt
	B-5 = in	IRQ 5 for PCI interrupt
	B-4 = out	CS3, Z-BUS, external DSACK, A23 high = devices
	B-3 = out	Driven high (not used)
	B-2 = out	CS2 for PCI interface, internal 0-wait, LW DSACK
	B-1 = out	CS1 for DRAM memory, external DSACK
	B-0 = out	CS0 for ROM, internal 2-wait, byte-wide DSACK (global select at power-up)

Autovector: AV7 (NMI) and AV5 (PCI) use autovector
Periodic Interrupt Timer: OS/Impact tick timer
Software Watchdog Timer: Reserved for IUSC hang
BERR Monitor: 64 clocks (4 microseconds)

The ICP2432 IARBs are:

- F = Module configuration register (MCR for SIM)

- E = DMA controller

- B = Serial controller

3.3.2 Chip Select Registers

The four chip select registers are initialized as shown in Table 3–1.

Table 3–1: Device Parameters

Device/Memory Space	Width	DTACK
Processor Memory	longword	external, 0 wait state
FLASH	byte	internal, 2 wait states
PCI Interface Device	longword	internal, 1 wait state
Z-BUS Space	word	external, 2 wait states

3.3.3 Volatile Memory (DRAM)

The DRAM memory is the ICP2432's central resource and is used for both executable code and protocol data buffers. The memory can be accessed by the 68349's CPU32+ and DMA controller and by the integrated DMA controller within each of the IUSCs. The 68349 provides all of the arbitration for this central memory resource. The 68349 arbitration takes place concurrently with bus transactions to ensure high bandwidth availability. The arbitration priority, from highest to lowest, is external bus masters (IUSCs), internal DMA controller, and CPU32+.

The memory system is implemented as a 32-bit, zero-wait state DRAM memory system that supports byte, word, and longword access.

The refresh mechanism for ICP2432 uses embedded refresh cycles for each CPU32+ access of the FLASH or IUSCs and during each IUSC memory transaction. CPLD logic

includes both a 16 microsecond timer and a refresh counter. The timer forces a refresh every 16 microseconds if no embedded refreshes have occurred. The counter increments for each embedded refresh cycle completed and decrements at each 16 microsecond timer interval. Embedded refreshes from multiple accesses of FLASH, IUSC, or IUSC transactions delay timer refreshes, thereby maximizing memory bandwidth.

3.3.4 Non-volatile Memory (FLASH)

The ICP2432 requires firmware to execute manufacturing diagnostics, power-up self tests, and the bootload code that coordinates the download of OS/Impact and protocol images from the host.

The ICP2432 provides a 32-pin PLCC socket for a non-volatile device. The socket supports any 512 kilobit-to-4 megabit (64 kilobyte-to-512kilobyte) EPROM/FLASH/OTP device whose footprint meets the JEDEC standard. Since the firmware code is seldom changed, in-circuit write cycles for FLASH or EEPROM devices are not supported. A 512 kilobit (64 kilobyte) device is standard at this time.

3.3.4.1 Diagnostics

Self-test diagnostics are executed on initial power-up. These tests validate the control circuitry, memory, processor and Z-BUS peripheral devices, interrupt operation, and bus error logic of the ICP2432 hardware. Test status and error reporting may be monitored by connecting a terminal to the console port on the optional debug module.

The successful completion of the diagnostics allows the downloader to execute on request from the host.

Additional diagnostics are run if:

- the DTR modem signal of the ICP2432 console port drives its DCD input signal, or

- the DCD signal is driven by a console device's DTR and it is "true" and the console device responds with a "p" (for production) in response to an ENQ.

The production diagnostics include IUSC DMA tests and IUSC/Sipex digital I/O loop-back tests. These tests are not run in real environments because the data and control signals do not respond predictably without self-port loopback connectors installed.

If the console device responds with a "d" (for debug), the PROM code calls the memory/device display routine so that a programmer can examine various memory locations and device registers that may not have been changed by the reset.

Note that most program debugging should be done with a symbolic debugger such as SingleStep from Software Development Systems, Inc., which can be downloaded with OS/Impact and the application software.

3.3.4.2 Downloader

The CPU uses the boot loader code and the host driver to download the OS/Impact executive and applications from the host to the board. The ICP2432 does this by moving the executable images to the onboard memory. The CPU begins execution of the downloaded code at the completion of the download sequence.

3.3.5 Debug Port

The ICP2432 implements all debug support circuits on a plug-in module. The debug module is intended for use by software developers, production personnel, and toolkit customers.

The debug panel includes the following:

- Reset push-button

- NMI push-button

- EIA-232 line driver/receiver for 68349's serial channel A

- Cable from board to module and cable from module to terminal

The ICP baseboard has a 10-position box-header located at the back edge of the board. This header is easily accessible and allows debug module attachment without removing the ICP from the PCI slot. The connector for the cable between the ICP and the module is keyed, and requires a keyed cable. Failure to properly connect the cable between the ICP and the module will damage the circuitry on the module. The terminal connector is keyed, but operates correctly with either cable orientation.

The debug module connector pinouts are shown in Table 3–2 and Table 3–3.

Table 3–2: Debug Module to ICP

Pin Number	Signal
1	+5v
2	GND
3	TD
4	RD
5	RTS
6	CTS
7	RESET_BUTTON
8	NMI_BUTTON
9	GND
10	+5v

Table 3–3: Debug Module to Terminal

Pin Number	Signal
1, 10	TD
2, 9	RD
3, 8	DTR
4, 7	DCD
5, 6	GND

3.4 Communications Bus (Z-BUS)

The ICP2432 communications bus exists as a space in the CPU memory map and is composed of integrated universal serial controllers (IUSCs), electrical interface control registers, a modem Test Mode input register, a board ID register, and the support circuitry required for timing, data transfer, bus arbitration, and interrupt operations. This section gives a full description of the major devices, support circuitry, bus signals, and operation of the communications bus. The communications bus is implemented as a Zilog multiplexed bus and is referred to as the Z-BUS throughout this section.

3.4.1 Communication Devices with Integral DMA

The ICP2432 uses two, four, or eight Z16C32 integrated universal serial controllers (IUSCs) with integral DMA. Each IUSC is a single-channel, multi-protocol data-communications device that functions as a serial-to-parallel and parallel-to-serial data converter. It may be software configured to satisfy a wide variety of serial communications applications. Each IUSC may be programmed to handle asynchronous formats, synchronous byte-oriented protocols such as IBM bisynchronous, and synchronous bit-oriented protocols such as HDLC and IBM SDLC. The IUSC can generate and check CRC codes in any synchronous mode and can be programmed to check data integrity in various modes. Each IUSC includes modem control signals, data clock generation, vector-includes-status interrupts, and an integral DMA interface.

The transmitters and receivers of each IUSC have a 32-byte FIFO that can be loaded or emptied by byte or by word. The ICP2432 also has a nominal baud clock source of 3.6864 megahertz, plus a socket for an optional oscillator for specific, non-standard bit clocks.

The integral DMA controllers in the IUSCs can operate in any of four modes: Single Buffer, Pipelined, Array, and Linked List. Command and status information can be extracted or put into the lists of the last two modes. This feature allows automated reception of back-to-back frames of SDLC data without loss of data or frame rejection

even at a 10 megabit per second rate. As with the serial controller sections of the IUSCs, the DMA controller sections can generate interrupts with Vector Includes Status which enhances interrupt response. Up to 64 kilobytes may be sent or received in one block out of or into multiple buffers.

3.4.1.1 ICP2432 IUSC Configuration

Like the 68349, the IUSC is highly configurable and requires hardware indicators to determine basic operation before its configuration registers are initialized. The IUSC determines its operational mode by sampling several pins during the first write to it. The IUSC ignores its address pins and always initializes the bus configuration register on this first write.

The ICP2432's initial write to each IUSC BCR is as follows:

```
0007h with the ZAS and S/D pins low
```

This results in the following operational characteristics:

- ZAS low: multiplexed, direct address bus implementation

- S/D low: WAIT/RDY pin is acknowledge for slave and IACK cycles

- IACK mode: 00, capture INTACK at start of cycle

- BRQTP: 0, open drain bus request

- 16Bit: 1, device is word-wide bus

- IRQTP: 1, open drain interrupt request

- SRight: 1, capture slave register address on AD6-0

The IUSC's DMA clock is 8 Mhz, the baud rate generator clock is 3.6864 Mhz. Optionally, a 16 Mhz clock or a unique oscillator socket can be selected as sources of serial data clocking.

3.4.2 Circuits

Table 3–4 lists the complete set of circuits provided by the ICP2432 platform for each port. The ICP2432 is a DTE device. Some interfaces may only support a subset of these circuits. The EIA and MIL standards use unique names to identify some equivalent circuits. To simplify Simpact's design and production documentation, the ICP2432 uses a set of generic names for the interface circuits it supports.

Table 3–4: Interface Circuits

Transmit Circuits

TD	Transmit data
TRXCO	Transmit clock out (ICP provides transmit clock)
RTS	Request to send
DTR	Data terminal ready
LPBKL	Local loopback
LPBKR	Remote loopback

Receive Circuits

RD	Receive data
RTXC	Receive clock
TRXCI	Transmit clock in (modem provides transmit clock)
CTS	Clear to send
DCD	Data carrier detect
SYNC	Ring indicator
DSR	Data set ready
TM	Test mode

3: Hardware Survey

3.4.3 Clock Steering

Clock steering is required for both the TXC and RXC clock pins of the IUSC. Software initialization routines must control the steering logic in conjunction with the clock control register selections made in the device.

The SP504 supports direct DTE/DCE transmit clock source selection through the use of its TTEN and STCEN inputs. Each IUSC's TxREQ output pin is connected to these SP504 pins. A low on the IUSC's TxREQ pin selects the DTE device as the source of the transmit clock (internal). A high on the IUSC's TxREQ pin selects the DCE device as the source of the transmit clock (external).

RXC steering is based upon the IUSC RxREQ output. A high on the RxREQ pin steers the clock from the I/O connector to the RXC input pin on the IUSC. A low on the RxREQ pin steers the output from the optional oscillator to the RXC input pin on the IUSC.

3.4.4 Test Mode Input Register

All modem control signals except Test Mode are handled directly by each port's IUSC. The test mode input status for all ports is accessible by the 68349 via the test mode register located at 181_0000 hex on the Z-BUS. See Figure 3–6.

```
Address = 181_0000     Bit # = Port #
Byte wide              0 = TM on
Read only              1 = TM off
```

7	6	5	4	3	2	1	0

3289

Figure 3–6: Test Mode Register

3.4.5 Sipex Mode Select Register

The two- and four-port versions of the ICP2432 use configurable drivers/receivers which allow the software to select the electrical interface to be used while communicating with a modem or other device. The electrical interfaces are:

- EIA-232 (V.28)

- EIA-422 (V.11 and X.27)

- EIA-423 (V.10 and X.26)

- EIA-449

- EIA-485

- EIA-530

- EIA-530A

- MIL-STD-188C (interoperable)

- MIL-STD-188-114A Balanced Type II

- V.35

- V.36

Some of the electrical interfaces allow optional termination resistors in some applications. To support this, the ICP2432 has sockets for single-inline pack (SIP) resistors that can be inserted by the customer.

3.4.6 Board ID Register

The ICP supports the use of mezzanine boards. Therefore, mezzanine assemblies have unique board IDs that diagnostic code, OS/Impact, and protocol software use to determine the hardware configuration.

The 68349's port A, bit 3 is defined as an input and is connected to a pull-up resistor and to the mezzanine connector. Diagnostic code checks this bit at reset time to deter-

mine if a mezzanine board is present. If a mezzanine board is present, the CPU determines which one by reading its board ID register. There are currently two mezzanine products:

- Four-port mezzanine ID = 04

- Eight-port mezzanine ID = 08

3.5 PCI Interface

The ICP2432 front-end processor may be used with any computer that has PCIbus support.

The interface comprises multiple mailbox registers and dual FIFOs. The S5933 has an integrated DMA controller on the PCIbus side and supports the use of the 68349 DMA channels on the add-in side. As a bus master (initiator), it supports memory read and memory write commands. As a slave (target), it supports I/O reads/writes and configuration reads/writes. Figure 3–7 shows the PCI interface block diagram.

The mailboxes are used to pass messages between the ICP and the host to coordinate data exchange while the FIFOs are used for actual data transfer. Writing to the outbound mailbox from one side can generate an interrupt on the other side.

Detailed descriptions of the ICP implementation are found in Simpact's *ICP2432-to-PCI Host Protocol Specification*, DC 900-1509.

3.5.1 PCI Configuration Registers

The PCI specification defines a configuration space that all devices must support. The S5933 provides both hard-wired and "soft" registers that are modified by the ICP2432.

The S5933 includes an interface to a serial EEROM device. At reset time, the EEROM is used to modify the 5933's "soft" configuration registers with ICP2432-specific parameters.
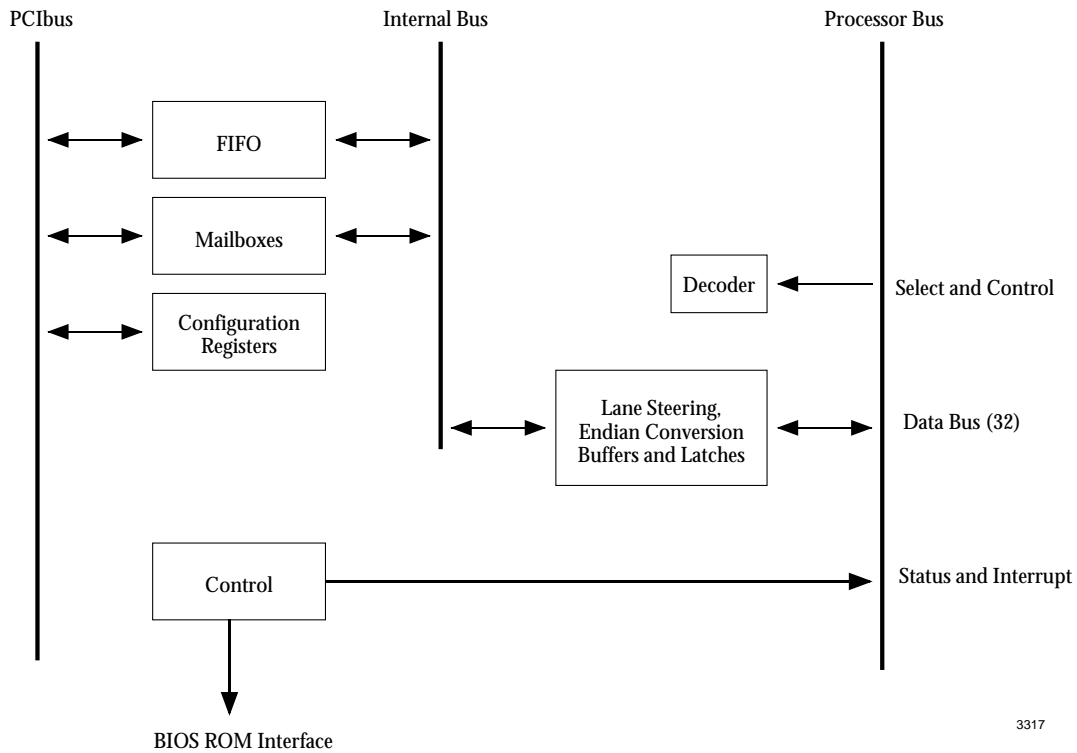
PCIbus   Internal Bus   Processor Bus

FIFO

Mailboxes

Configuration
Registers

Decoder   Select and Control

Lane Steering,
Endian Conversion   Data Bus (32)
Buffers and Latches

Control   Status and Interrupt

BIOS ROM Interface   3317

Figure 3–7: PCI Interface Block Diagram

The ICP2432 EEPROM modifies the 56933 registers as shown in Figure 3–8.

PCI Configuration Space Header			
31 24	23 16	15 8	7 00
DEVICE ID = 0002, 0004, or 0008		VENDOR ID = 12A1	
STATUS		COMMAND	
CLASS CODE = FF0000			REV ID
BIST = 00	HEADER TYPE = 00	LATENCY TIMER = F8	CACHE LINE SIZE
BASE ADDRESS REGISTER #0 = I/O (64 bytes)			
BASE ADDRESS REGISTER #1 = 0000_0000, disabled			
BASE ADDRESS REGISTER #2 = 0000_0000, disabled			
BASE ADDRESS REGISTER #3 = 0000_0000, disabled			
BASE ADDRESS REGISTER #4 = 0000_0000, disabled			
BASE ADDRESS REGISTER #5 = 0000_0000, disabled			
RESERVED = 0's			
RESERVED = 0's			
EXPANSION ROM BASE ADDRESS = 0000_0000, disabled			
RESERVED = 0's			
RESERVED = 0's			
MAX_LAT = 0F	MIN_GNT = FF	INTERRUPT PIN = 1, INTA#	INTERRUPT LINE

3429

Figure 3–8: PCI Configuration Space Header

3.6 Details of Operation

This section describes the interrupt, board timing, Z-BUS arbitration, reset, bus error, and NMI functions of the ICP2432.

3.6.1 Interrupts

The ICP2432 implements the 68349 interrupt mechanism as described in Section 3.6.1.1, the external interrupt level assignments as described in Section 3.6.1.2, and the interrupt arbitration as described in Section 3.6.3.2.

3.6.1.1 Interrupt Operation

The ICP2432 uses three of the seven 68349 processor interrupt levels. Each interrupt source accesses its own interrupt request pin (INTRQn). These input pins are sampled by the processor. If any of them are asserted, and if the highest priority pin is greater than the current interrupt mask level, an interrupt request is made pending. The processor services the pending interrupt at the next instruction boundary. The 68349 supports both device-supplied vector (DSV) and autovector (AV) peripherals. The IUSCs are DSV, and the NMI and AMCC5933 PCIbus interface are AV. When the processor is ready to initiate interrupt servicing, it begins an interrupt acknowledge cycle. This cycle is similar to a normal read cycle. An $\overline{\text{IACK}}$ cycle at the pending interrupt level enables the highest priority device at that level to present a vector byte. The processor uses this byte as an index to locate the peripheral's interrupt service entry point.

3.6.1.2 Interrupt Level Assignments

The interrupt levels can be summarized as follows:

> **Level 7**   A non-maskable interrupt that causes the 68349 to stop execution of the current program and break to the debugging tool. NMIs may be issued by pushing the NMI push-button on the optional debug module

| **Level 6** | Used by the integrated universal serial controllers (IUSCs) |
Level 5	Used by the PCIbus interface AMCC5933
Level 4	Not used
Level 3	Not used
Level 2	Not used
Level 1	Not used
Level 0	Normal operation, no interrupts

3.6.2 Timing

This section describes the clock generation logic of the ICP2432 and the transaction timings that result from its implementation.

3.6.2.1 Clock Generation

The ICP2432 uses synchronous state machine design techniques throughout the control logic to ensure accurate circuit performance and to increase circuit reliability. All clock signals are derived from a divider circuit that is clocked by the 32 MHz system oscillator as shown in Figure 3–9.

The 32 MHz state clock is used by the processor bus memory controller, the Z-BUS conversion, and arbitration state machines. The 16 MHz clock is used by the 68349 CPU and as an optional input to the IUSC. The 8 MHz clock is used by the IUSC DMA controllers and the Z-BUS interrupt state machine.
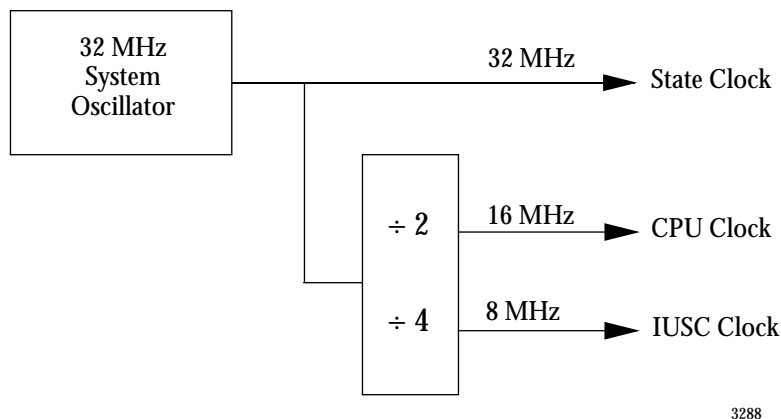
3288

Figure 3–9: ICP2432 System Timing

3.6.2.2 Transaction Timing

The advanced architecture of the 68349 makes exact instruction timing calculations difficult. The three-stage prefetch pipeline, execution overlap capabilities, and effects of operand misalignment complicate these calculations. Timing is also affected by memory system refreshes and Z-BUS arbitration delays. Refresh delays and PCIbus memory usage also affect IUSC DMA controller memory cycle times. Table 3–5 shows typical transaction times for each Master/Slave cycle.

Table 3–5: Transaction Timing

	MEM	**FLASH**	**AMCC**	**Z-BUS**
63349 cycle at 16 MHz	187 ns [a]	312 ns	187 ns	375 ns
16C32 cycle at 8 MHz	375 ns	N/A	N/A	N/A
	Typical instruction time MOVL (Am)+, (An)+ Worst case = 9 clocks at 62.5 ns = .562 µs Cache case = 7 clocks at 62.5 ns = .438 µs			

[a] For cycles that are not delayed by refresh.

3.6.3 Z-BUS Operation

The low-level hardware operation of the 68349 and IUSC devices is controlled by the 68349 bus arbiter and a bus conversion state machine. The four- and eight-port ICPs also require interrupt arbitration and DMA arbitration state machines as described below. Refer to .

3.6.3.1 Bus Conversion State Machine

The bus conversion state machine controls any transactions that span the processor and the communications buses. When the 68349 accesses a device on the Z-BUS, the state machine translates the Motorola control signals to the appropriate Z-BUS control signals and manages the multiplexing of the Motorola address and data buses onto the multiplexed Z-BUS. When an IUSC is bus master and accesses the DRAM memory, the state machine translates the Z-BUS control signals to the appropriate Motorola control signals and manages the de-multiplexing of the Z-BUS address/data bus onto the separate Motorola address and data buses. The bus conversion state machine also monitors the 68349 interrupt and bus arbitration signals and manages the inter-bus buffers during interrupt and bus arbitration cycles.

3.6.3.2 Interrupt and DMA Arbitration

The IUSC supports a daisy-chain mechanism for both its interrupt and bus request operations. The following paragraphs discuss the interrupt request and acknowledge operation, but are also valid for the bus request and acknowledge operation. The descriptions of the interrupt daisy-chain and the state machine details are duplicated for the DMA. See .

The Zilog interrupt daisy-chain implementation connects multiple IUSC devices to a common interrupt request line. When an interrupt acknowledge cycle is started, an interrupt enable (IEI and IEO) propagates serially through the devices. A device that does not have an interrupt pending passes the enable to the next device in the chain. The device at the highest physical priority on the chain, that has an interrupt pending,
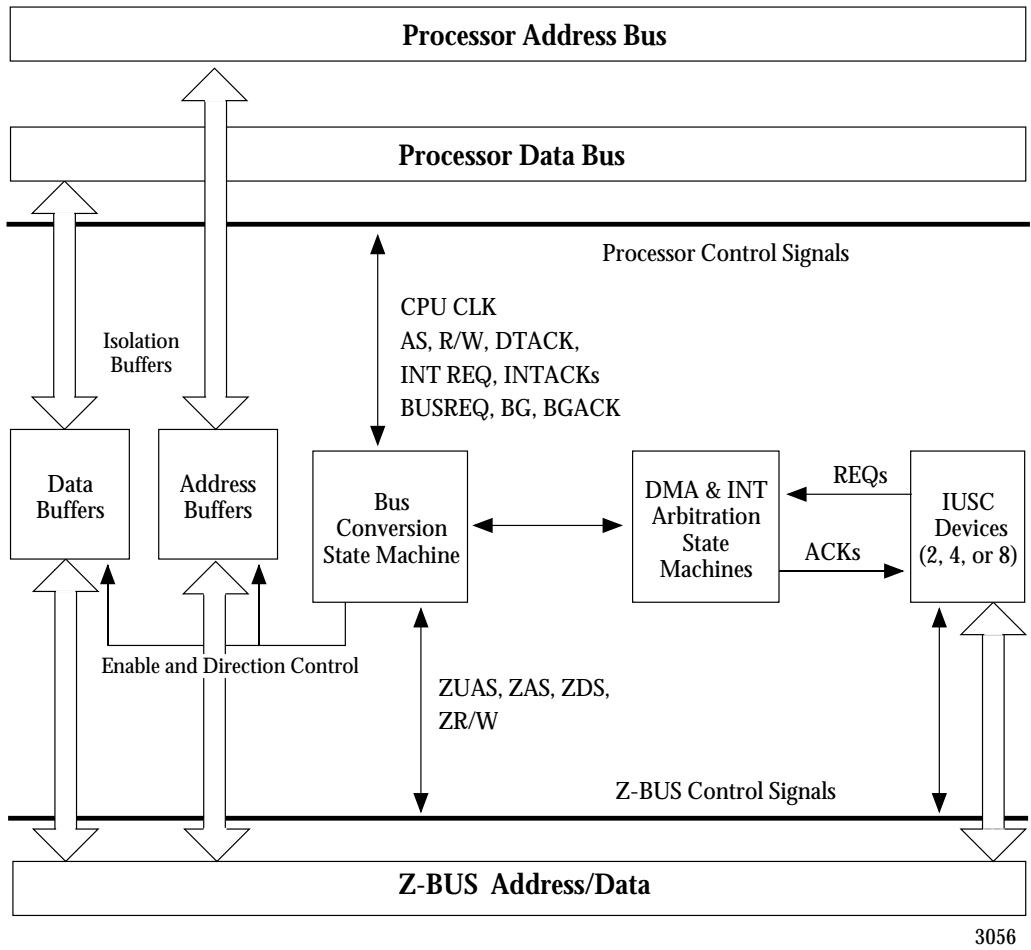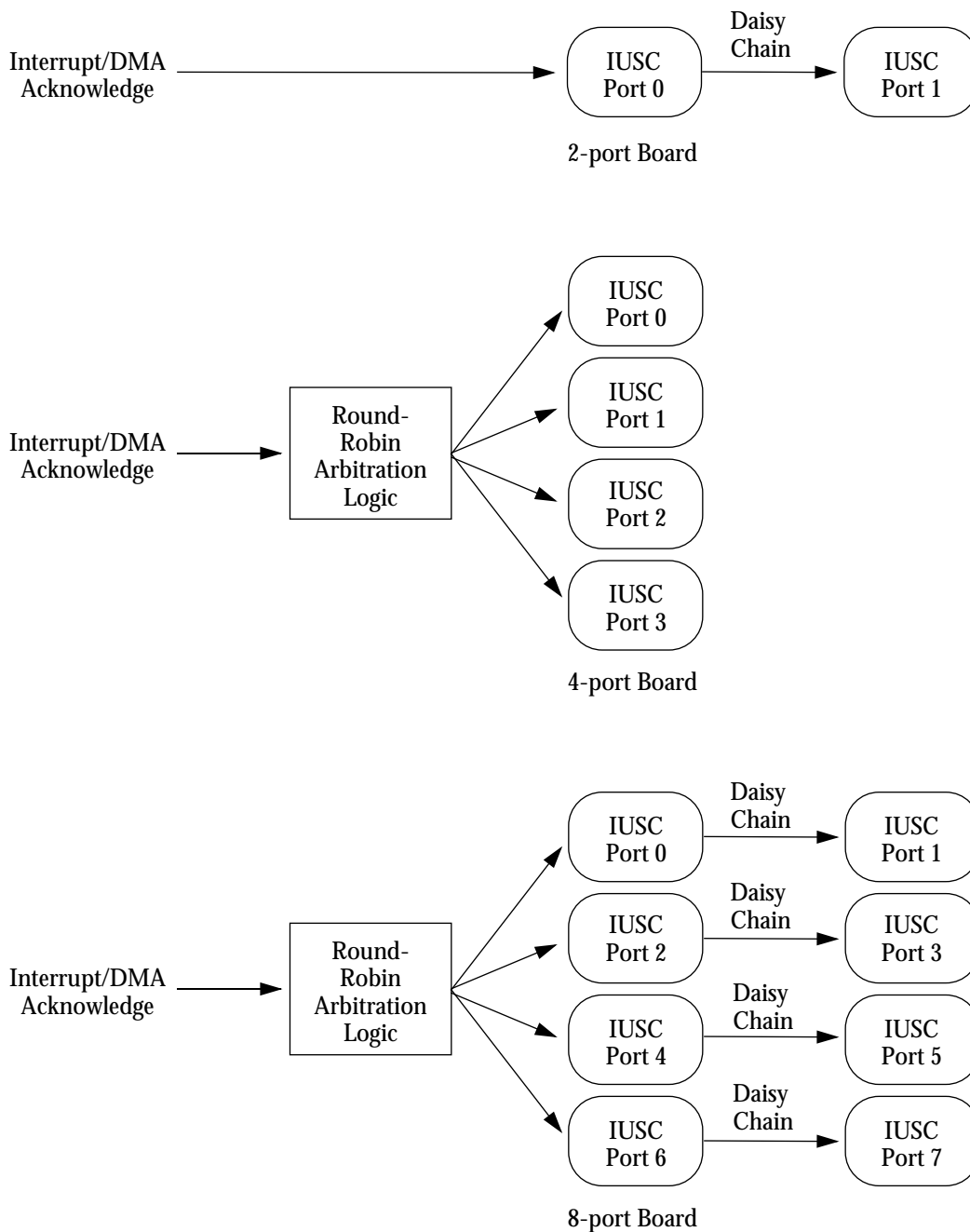
Figure 3–10: Z-BUS Arbitration Block Diagram

Figure 3–11: Interrupt and DMA Priority Arbitration

blocks the enable signal to the next device in the chain and presents its vector to the processor for servicing.

The two baseboard ports on the two-port ICP2432 are on a single daisy-chain with port 0 at a higher priority than port 1. The interrupt and bus requests are tied directly to the bus control state machine and do not use a complex arbitration state machine.

On the four-port ICP2432, each IUSC has its interrupt and bus request control signals connected to complex arbitration state machines. These machines make the request to the 68349 and, according to the current priority, pass the acknowledge to the appropriate device. Each of these state machines implements a modified round-robin algorithm. This scheme uses a "last user = lowest priority" method that eliminates the delays associated with true round-robin, or time-slice, schemes. When only one IUSC has a request, the state machine acknowledges that IUSC without waiting for a time-slice window to occur. If more than one IUSC has a request, the state machine issues a grant according to the current IUSC priority list.

The eight-port ICP2432, interrupt and DMA arbitration comprises a combination of the baseboard and four-port schemes. The arbitration state machines are identical to those described for the four-port board, but each of the four levels has two daisy-chained IUSCs as described for the two baseboard ports.

3.6.4 Reset

In normal operation, the reset controller keeps the $\overline{\text{RESET}}$ signal in a high-impedance state. This allows the 68349 to drive the line during the execution of a reset instruction. The reset controller PLD generates an 100 microsecond minimum reset pulse for the following conditions:

1. At power-up

2. When the PCIbus signal RST# is asserted

3. When bit 24 of the bus master control register is set by the host

4. When the reset switch on the optional debug module is pushed

All of the devices, state machines, and configuration registers on the board are initialized to a known state when the $\overline{\text{RESET}}$ signal is driven low.

3.6.5 Non-maskable Interrupt (NMI)

The interrupt request controller issues a non-maskable interrupt (NMI) request when the NMI button on the optional debug module is pressed. The 68349 executes the code pointed to by the AutoVector 7 vector.

3.7 I/O Port Details

Figure 3–12 shows the signal mapping for the EIA-449 interface. Figure 3–13 shows the signal mapping for the EIA-530 interface (which is also used for EIA-530A and EIA-232).

The eight-port version of the ICP2432 always provides an EIA-232 electrical interface on all ports. Figure 3–14 shows the signal mapping for the EIA-232 interface.

3.7.1 Two-port Baseboard

Each of the two ports on the baseboard use 26-pin micro-D to Dsub adapter cables which map to standard EIA (DTE, male pin) connectors. The cable options are:

- AS-100-0185-01 = EIA-530 (25-pin) connector

- AS-100-0186-01 = EIA-449 (37-pin) connector

- AS-100-0185-01 + AS-000-283-01 = V.35 (34-pin) connector

3.7.2 Four-port Mezzanine Board

The four-port mezzanine board provides four full-function, software-programmable EIA interface ports, as described for the two baseboard ports except that the four-port mezzanine board uses a high-density 100-pin connector (similar to those used in SCSI
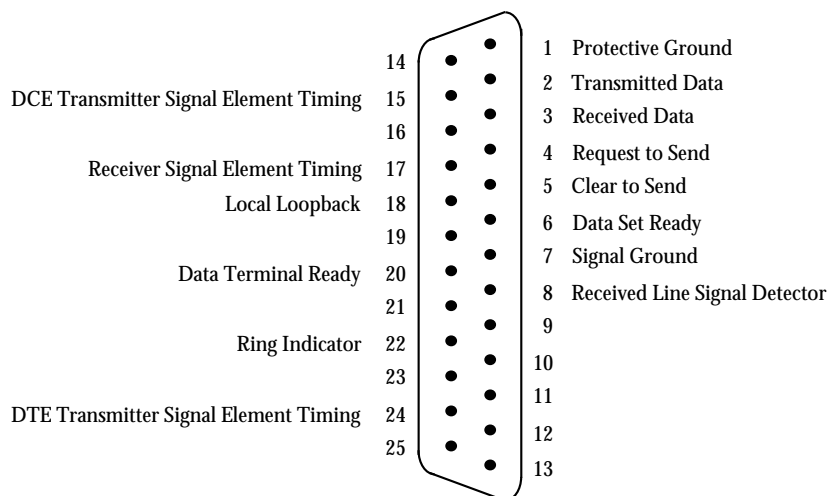
		1	Shield
20		2	
21		3	
Send Data (B)	22	4	Send Data (A)
Send Timing (B)	23	5	Send Timing (A)
Receive Data (B)	24	6	Receive Data (A)
Request to Send (B)	25	7	Request to Send (A)
Receive Timing (B)	26	8	Receive Timing (A)
Clear to Send (B)	27	9	Clear to Send (A)
	28	10	Local Loopback
Data Mode (B)	29	11	Data Mode (A)
Terminal Ready (B)	30	12	Terminal Ready (A)
Receiver Ready (B)	31	13	Receiver Ready (A)
	32	14	Remote Loopback
	33	15	Incoming Call
	34	16	
Terminal Timing (B)	35	17	Terminal Timing (A)
	36	18	Test Mode
Send Common	37	19	Signal Ground

3295

Figure 3–12: Signal Mapping for the 37-pin EIA-449 Interface

Transmitted Data (B) 14
Transmitter Signal Element Timing DCE (A) 15
Received Data (B) 16
Receiver Signal Element Timing DCE (A) 17
Local Loopback 18
Request to Send 19
DTE Ready (A) 20
Remote Loopback 21
DCE Ready (B) 22
DTE Ready (B) 23
Transmitter Signal Element Timing DTE (A) 24
Test Mode 25

1 Shield (A)
2 Transmitted Data (A)
3 Received Data (A)
4 Request to Send (A)
5 Clear to Send (A)
6 DCE Ready (A)
7 Signal Ground
8 Received Line Signal Detector (A)
9 Received Signal Element Timing DCE (B)
10 Received Line Signal Detector (B)
11 Transmitter Signal Element Timing DTE (B)
12 Transmitter Signal Element Timing DCE (B)
13 Clear to Send (B)

3294

Figure 3–13: Signal Mapping for the 25-pin EIA-530 Interface

14
DCE Transmitter Signal Element Timing 15
16
Receiver Signal Element Timing 17
Local Loopback 18
19
Data Terminal Ready 20
21
Ring Indicator 22
23
DTE Transmitter Signal Element Timing 24
25

1 Protective Ground
2 Transmitted Data
3 Received Data
4 Request to Send
5 Clear to Send
6 Data Set Ready
7 Signal Ground
8 Received Line Signal Detector
9
10
11
12
13

3293

Figure 3–14: Signal Mapping for the 25-pin EIA-232 Interface

interfaces) for the I/O ports. The following "quadrapus" cable options are available which map the 100-pin connector to four standard EIA (DTE, male pin) connectors:

- AS-100-0194-01 = four EIA-530 (25-pin) connectors

- AS-100-0195-01 = four EIA-449 (37-pin) connectors

- AS-100-0194-01 + four AS-000-283-01 = four V.35 (34-pin) connectors

3.7.3 Eight-port Mezzanine Board

The eight-port mezzanine board provides eight full-function EIA-232 ports. The ports are EIA-232 only; they are not software configurable to other electrical interfaces.

The eight-port mezzanine board uses a high-density 100-pin connector (similar to those used in SCSI interfaces) for the I/O ports. The following "octopus" cable maps the 100-pin connector to eight standard EIA (DTE, male pin) connectors.

- AS-100-0193-01 = eight EIA-23 (25-pin) connectors

Index

SIMPACT

Customer Report Form

We are constantly improving our products. If you have suggestions or problems you would like to report regarding the hardware, software or documentation, please complete this form and mail it to Simpact at 9210 Sky Park Court, San Diego, CA 92123, or fax it to (619) 560-2838.

If you are reporting errors in the documentation, please enter the section and page number.

Your Name: _____

Company: _____

Address: _____

_____

_____

Phone Number: _____

Product: _____

Problem or
Suggestion: _____

_____

_____

_____

_____

_____