intel

iPSC™ System
Product Summary

ORDER NUMBER 280101-002

The iPSC™ is the first family of expandable concurrent computers. It is designed to be a flexible base of hardware and software upon which to build concurrent programming tools and application programs.

The iPSC system consists of one, two, or four computational units. Each unit contains up to 32 high-performance microcomputers, each with its own numeric processing unit and local memory. The microcomputers are interconnected in a hypercube topology. A resident node operating system, in conjunction with communication coprocessors, provides the user application with process-to-process message delivery capabilities.

The iPSC also includes Intel's MULTIBUS®-based System 286/310 microcomputer which is connected to each node by an Ethernet communication channel. The System 310 serves as Cube Manager, providing the user interface to the Cube as well as hosting the programming tools and system diagnostics.

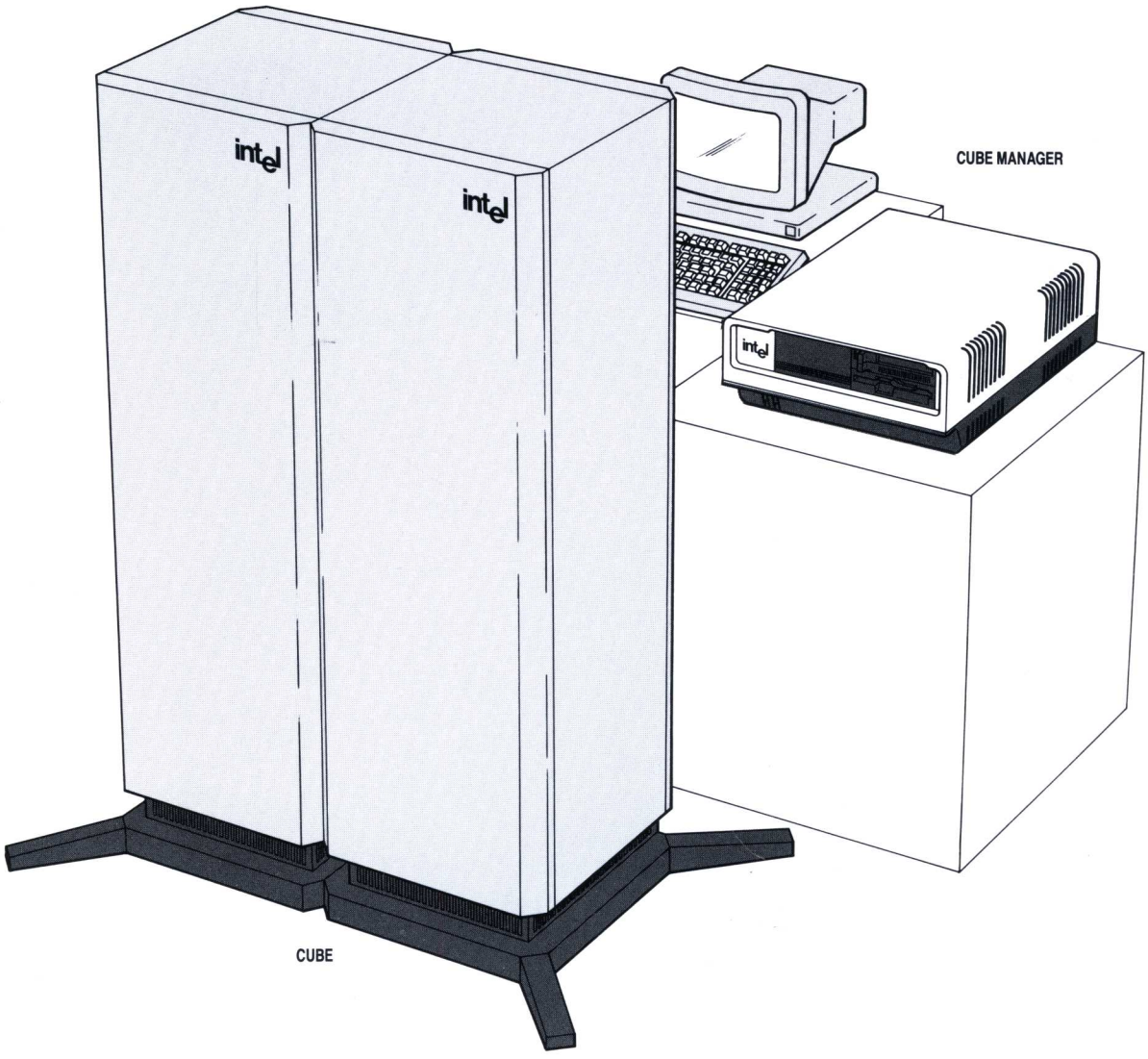

Figure 1-iPSC

FEATURES

Cube

Concurrent Architecture	Multiple, high-performance microcomputers work simultaneously, but independently on parts of a larger problem.
Versatile Hypercube Interconnect Topology	Efficient communication topology which is easily adaptable to a wide range of applications.
Reliable High-Speed Node-to-Node Communication	Bandwidth of 10 Mbit/second point-to-point internode serial communication channels with reliable message delivery between nearest neighbors.
Protected Software Environment	Intel 80286 central processing unit provides hardware memory protection between user programs and operating system.
High-Performance Numeric Computation	Intel 80287 numeric processing unit on each node. Supports 32-, 64-, and 80-bit floating point arithmetic operations (IEEE 754).
Large High-Speed Local Memory	512 KBytes of RAM per node, expandable to 4.5 MBytes.
Efficient Node Operating System	Multiple processes per node with cube-wide process-to-process message delivery; dynamic loading, execution, and termination of processes.
iLBX II Node Expansion Capability	Provides the facilities to extend memory or processing capacity with the addition of enhancement boards in adjacent node slots.
Field Expandable Computational Units	Base 32-node Cube is field upgradable to 64 or 128-node systems; expandable to 4.5 MBytes per node.
Low Mean-Time-To-Repair	A spare node board in each 32-node computational unit reduces mean-time-to-repair.
Office/Lab Environmental Design	Compact packaging, low acoustical noise, and an air-cooled design allow convenient placement.

Cube Manager

Supermicro System	Intel's powerful System 286/310 microcomputer with 80286/80287 and 2 MBytes of RAM.
Integrated Mass Storage	140 MByte Winchester disk, 360 KByte floppy disk, and 45 MByte cartridge tape drive integrated into one compact unit.
UNIX-Based Programming Environment	XENIX 3.0 operating system plus FORTRAN, C, and assembler languages.
High-Speed Manager-to-Cube Communication Channel	10 Mbit/second Ethernet channel bandwidth provides rapid direct access to every node with reliable message delivery.
Powerful Diagnostic Tools	Comprehensive confidence and diagnostic tests ensure system integrity and rapid fault isolation.
Flexible MULTIBUS Expansion	3 additional slots are available for system expansion from a choice of over 1500 MULTIBUS boards: 1 slot for high-speed memory expansion, 2 slots for other functions.
TCP/IP Network Access	Optional Ethernet, TCP/IP interface enables access to remote network resources.
Expandable Memory	Optional addition of DRAM memory board for a total of 5 MBytes of main memory.
Expandable Terminal Ports	Optional 8-port serial expansion board allows Cube Manager to support a total of 9 terminals.

SYSTEM OVERVIEW

The iPSC provides users with an expandable concurrent processing system for computationally intensive scientific applications. It consists of multiple high-performance microcomputers that each work concurrently on parts of a larger problem. This architecture enables users to exploit the inherent concurrency present in many scientific problems, and obtain high computational performance at the relatively low costs made possible by VLSI technology.

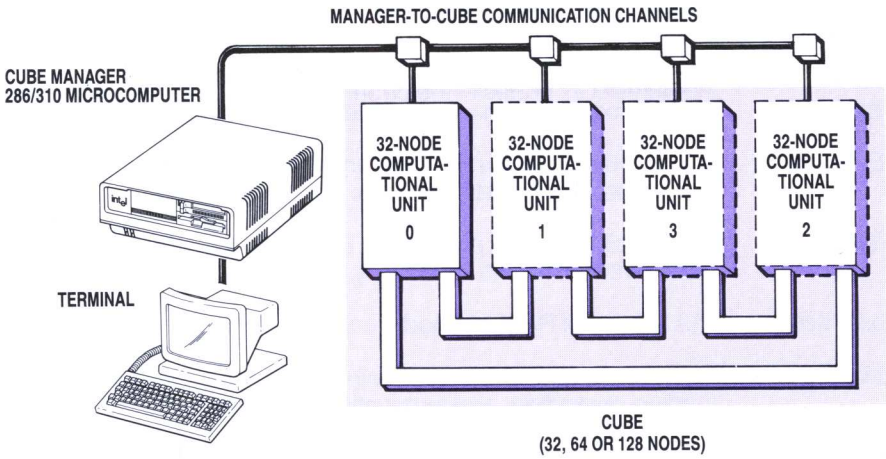

Figure 2–iPSC System Architecture

The iPSC consists of two major functional elements: the ***Cube*** and the ***Cube Manager.***

Cube

Cube hardware consists of an ensemble of one, two, or four computational units consisting of 32, 64, or 128 microcomputers or nodes. Nodes contain high-performance Intel 80286 CPU's coupled with 80287 numeric processing units and 512 KBytes of local memory. Nodes are interconnected by high-speed communication channels in a hypercube topology.

Cube software consists of a monitor and kernel residing on each node board. The monitor is in PROM and the kernel is loaded into node RAM after successful initialization. The kernel provides user application processes with a cube-wide process-to-process message passing service.

Cube Manager

Cube Manager hardware consists of an Intel System 286/310 microcomputer which is linked to each node over a global Ethernet communication channel.

Users interact with the System 310 via an ANSI-compatible alphanumeric terminal. MULTIBUS-based boards for the System 310 provide an optional Ethernet, TCP/IP LAN capability which enables networking to remote host environments.

Cube Manager software consists of a UNIX-based programming and development environment with FORTRAN, C, Assembler, cube control utilities and communications, and complete system diagnostics.

CUBE DESCRIPTION

Cube Hardware

The term Cube refers to the ensemble of microcomputers (nodes) connected in a concurrent hypercube architecture. The Cube may consist of one, two, or four 32-node computational units. The major hardware elements are described below:

Topology

The interconnection scheme, or topology, for the iPSC™ is a ***"binary n-cube"*** or ***"hypercube."*** This is an *n*-dimensional cube where *n* represents the number of directly-connected nodes that establish the cube's dimension. The "dimension" is equal to the power of two, corresponding to the number of nodes in the cube. A 32-node system is a 5-dimensional cube with each node connected to its five nearest neighbors (i.e., d5). A 64-node system is a 6-dimensional cube and a 128-node system is a 7-dimensional cube (i.e., d6 and d7 respectively). Examples of this topology are shown in Figure 3. For simplicity, this diagram also shows a 4-dimensional (16-node) and 5-dimensional (32-node) hypercube.

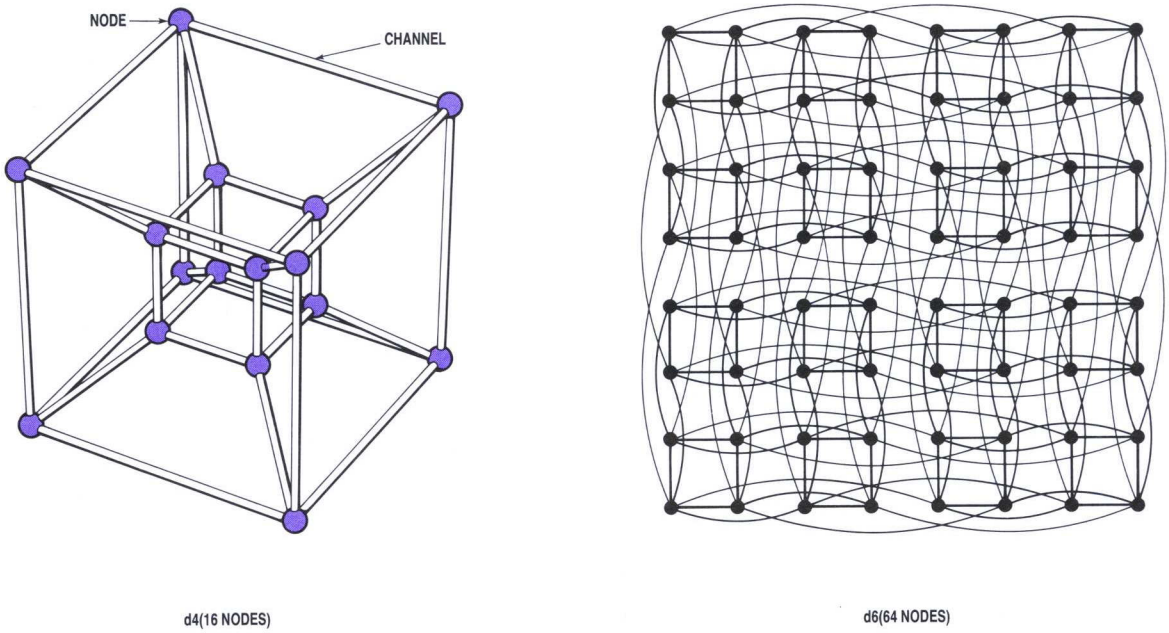

Figure 3 – Hypercube Topology

The various iPSC models are designated according to the dimension of the associated hypercube.

The available models are:

iPSC-d5	5-dimensional cube	(2^5 = 32 nodes)
iPSC-d6	6-dimensional cube	(2^6 = 64 nodes)
iPSC-d7	7-dimensional cube	(2^7 = 128 nodes)

Memory is local to each node, and interprocess communication is accomplished by message passing, eliminating the bus contention problems associated with shared memory systems.

The hypercube provides a nearly ideal balance between performance and communication overhead. Any node in the cube can be reached in a small number of message transmissions. Internode delays increase proportionally to the dimension of the cube. For example, in the 32-node system the maximum internode delay is 5.0 node-to-node transmission periods with an average delay of only 2.5, assuming uniformly distributed message distances. Typical application delays are much shorter.

Because of the versatility of the architecture, other topologies...such as rings, trees, etc.... can be mapped onto the system because the hypercube is a superset of these other topologies. Specifically, these topologies can be realized by appropriate programming. This multi-dimensional structure also makes the hypercube well suited to both homogeneous and heterogeneous computational problems.

The hypercube interconnection is physically implemented via the backplane within each computational unit. Cables running between units implement the interconnect for d6 and larger dimension cubes (see Figure 2).

Nodes

Each node in the Cube is an independent, single-board computer. The node contains a high-performance Intel 80286 central processing unit and its companion 80287 numeric processing unit which support 32-, 64-, and 80-bit floating-point formats. The node also contains 512 KBytes of dynamic RAM, with byte parity. An initialization and self-test monitor is contained in 64 KBytes of PROM (see Figure 4).

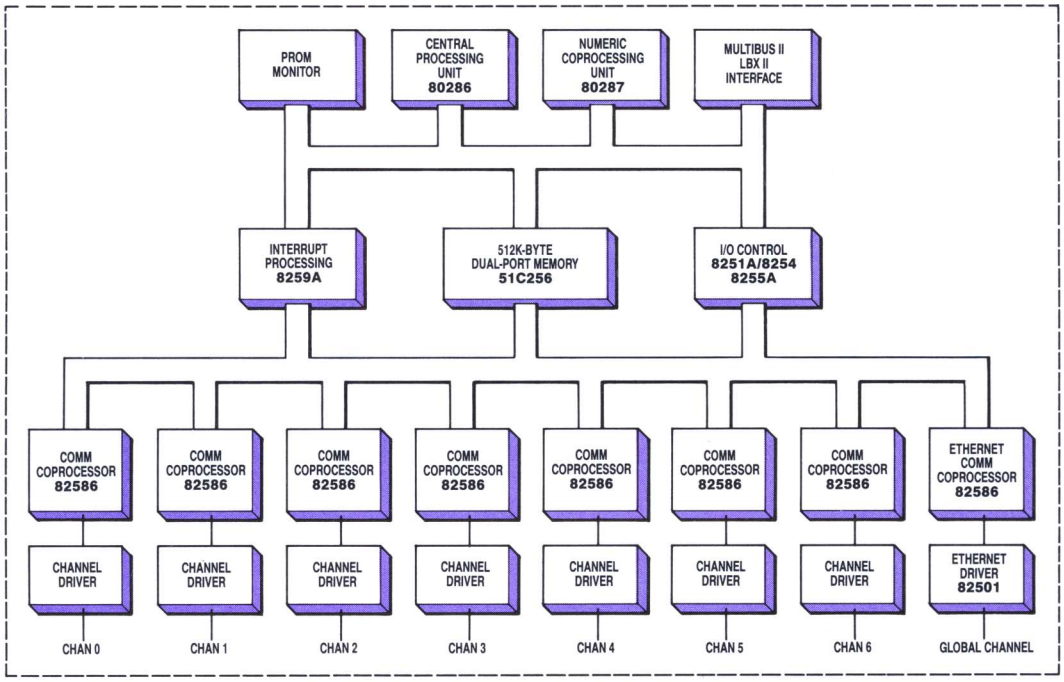

Figure 4 – Computational Node

Each node also contains eight bidirectional communication channels managed by dedicated 82586 communication coprocessors. Seven of these channels physically link the nodes together and serve as dedicated point-to-point communication channels. The eighth channel is a global Ethernet channel that provides direct access to and from the Cube Manager for program loading, data input/output, and diagnostics.

Each node board has red and green LED indicators which are used during initialization and reset for diagnostic purposes. The LED indicators may also be controlled by the user's node program during execution for monitoring and debugging.

Node Expansion Port

For enhancement purposes, the local processor memory bus on each node is accessible via a standard MULTIBUS II iLBX™ high-speed bus interface. The iLBX-II interface reaches the backplane via one of the two 96-pin DIN connectors. On the backplane, the iLBX-II bus is routed from each even-numbered board slot to the adjacent odd-numbered board slot. The odd-numbered slots may be populated by boards that extend the memory or processing capacity of the nodes. The node memory option expands each node's memory through use of the iLBX-II bus.

Manager-to-Cube Communication Channel

The Manager-to-Cube communication channels are shown in Figure 5. There are two channels: global and diagnostics.

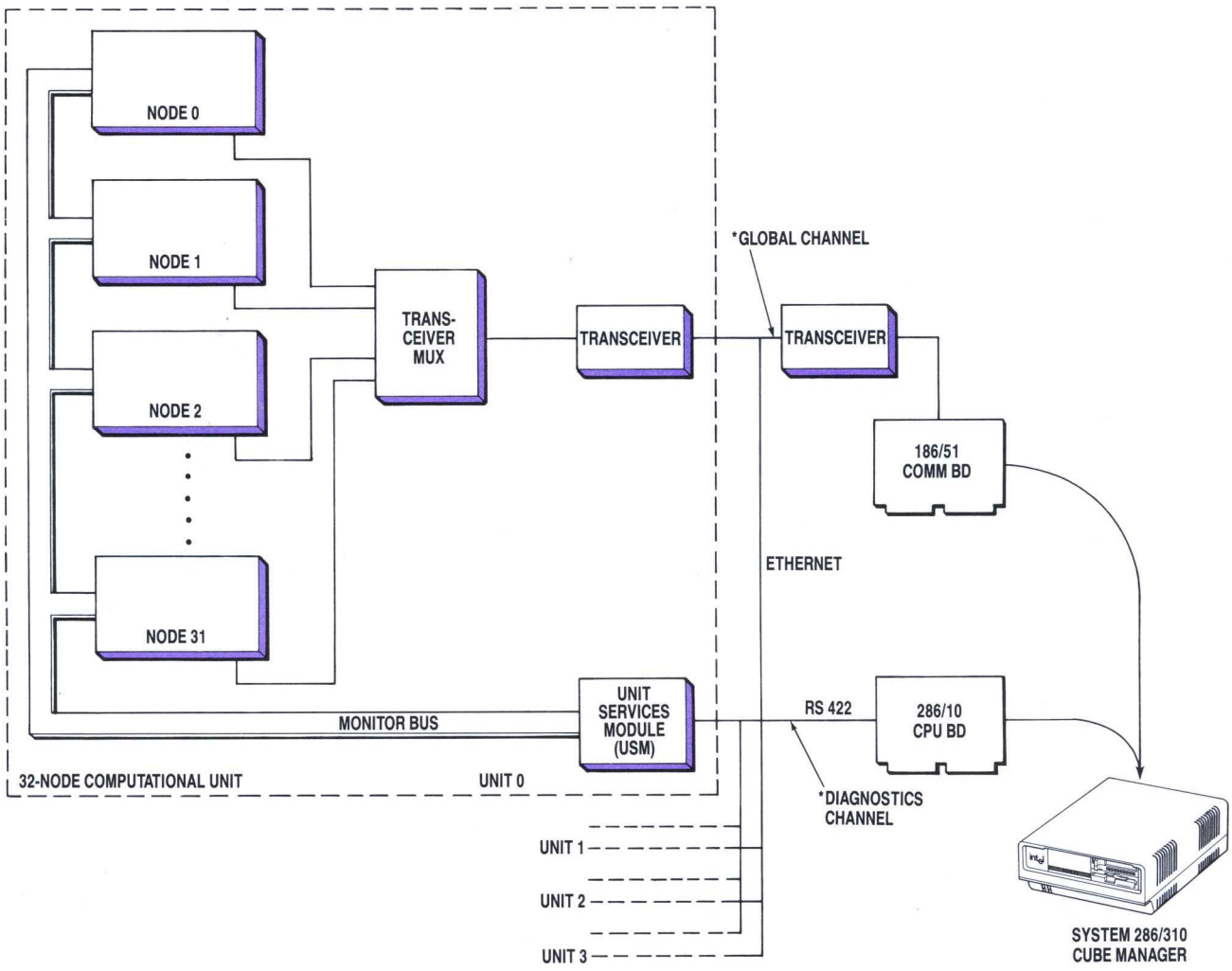

Figure 5–Manager-to-Cube Communication Channels

Channels

The global channel enables Cube Manager processes to communicate with the node processes. This channel physically links the Cube Manager to each node from the iSBC 186/51 communications board to the global communications coprocessor chip set on each node board. This channel complies with the standard Ethernet specification (IEEE 802.3).

The diagnostics channel is a separate path for communicating with the nodes. This channel links the Cube Manager to the unit services module (USM) through an RS 422 port. The USM manages the multidrop monitor bus supplying all nodes in a single unit with communication, interrupt, and reset lines. In the event of a failure, this alternate path is used to determine if the fault is within a node or within the global communication channel to the nodes. It is used to reset the nodes, initiate on-board diagnostics, and monitor the results, thus providing a more fundamental level of communication than is possible over the global channel.

Cube Enclosure

Each 32-node computational unit is a free-standing enclosure which is 49″ high and 16″ square with a 26.75″ × 26.75″ footprint. Each enclosure contains 32-node boards, one spare node board, and the USM board in one 34-slot card cage. Communications transceivers, multiplexer, power supplies, air cooling system, and associated cables are also housed in the enclosure (see Figure 6).

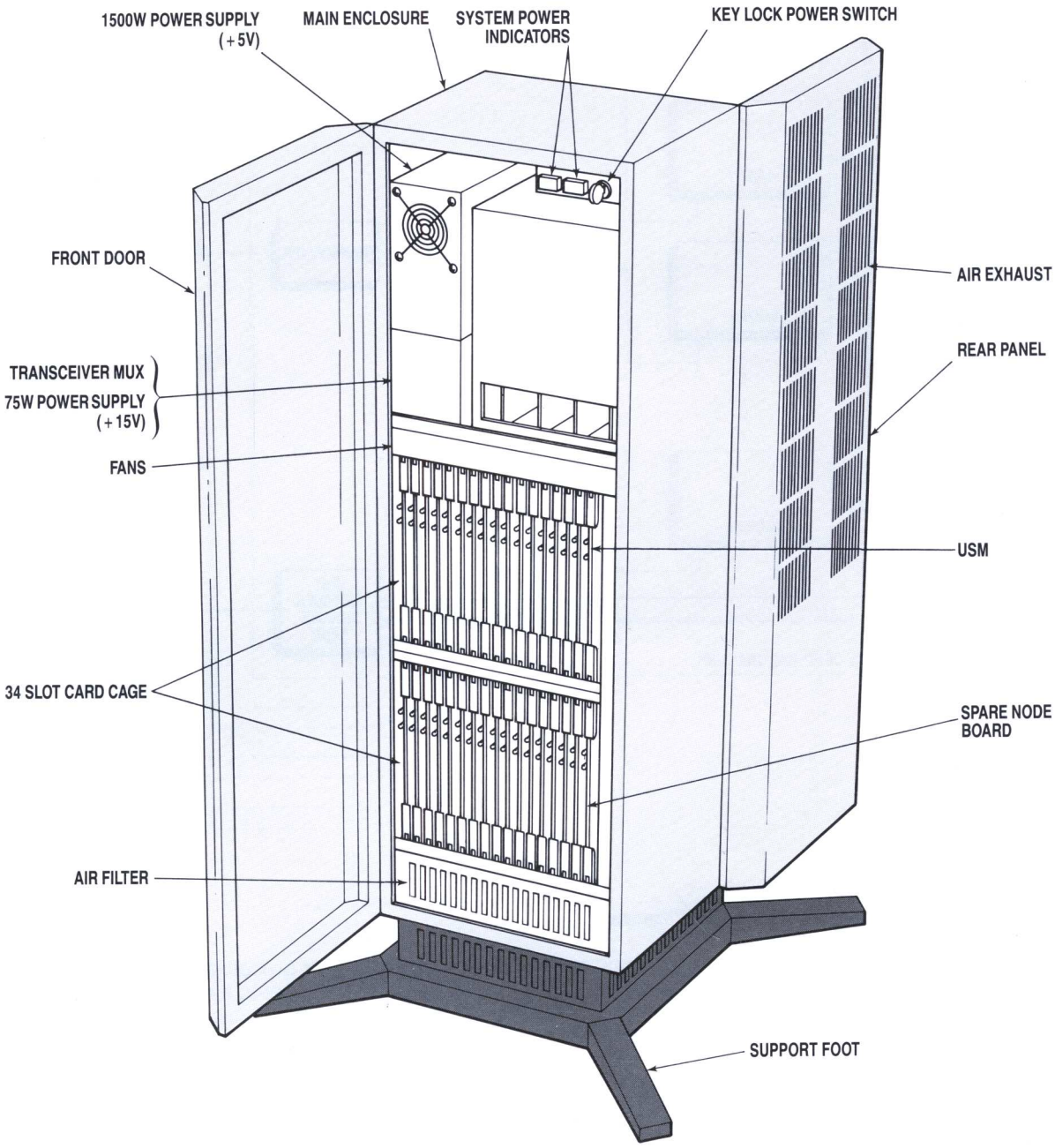

Figure 6–Cube Enclosure

Power Distribution System

A removable key, in the switch on the front panel, turns on the cooling fans and +15-volt, and +5-volt power. Front-face lamps indicate AC on and DC on in that sequence. A power module at the base of the unit houses a filter and circuit breaker which can be externally reset from the rear. An automatic thermal shutdown circuit in the +5-volt power supply protects the system from catastrophic heat failure (see Figure 7).

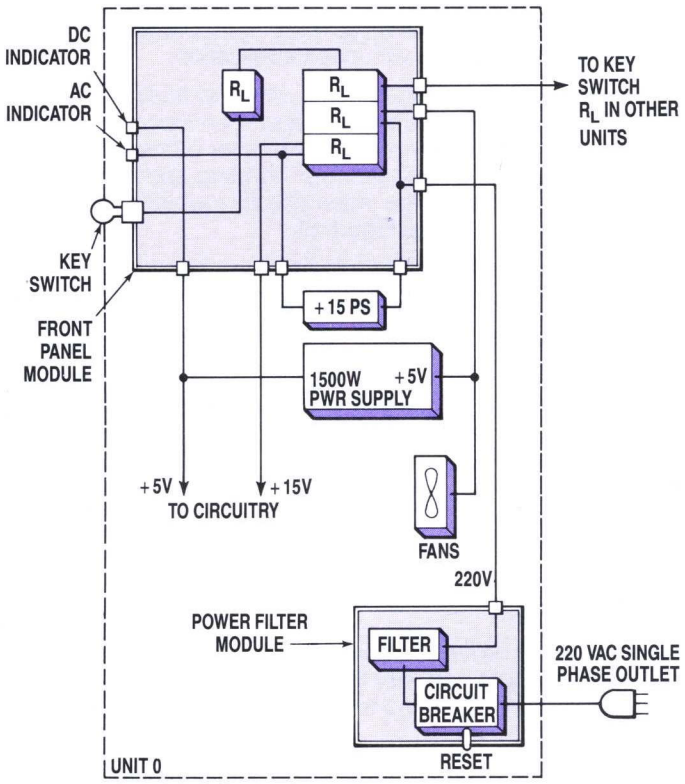

Figure 7–Power Distribution System

In a multi-unit system, Unit 0 enables the power-up sequence for each of the other units.

Cube Software

Cube software consists of a monitor and a kernel operating system residing on each node.

Node Monitor

The node monitor is responsible for initializing each node at power-up or system reset, verifying node operability, and loading the node operating system. It is stored in PROM on each node board.

Initialization

Initializes the system by resetting and enabling the node memory, communication controllers, I/O controller, interrupt controller, and CPU. Node identity is also set by reading the slot ID from the backplane.

Confidence Testing

Verifies the node board by running the node confidence test (*"NCT"*) on the RAM, peripheral devices, and the communications controllers.

Static Loading

Loads the node operating system and, optionally, a statically bound application program.

Node Operating System

The node operating system provides the application programmer with the necessary set of software services for dynamically loading programs, managing multiple processes, and delivering variable length messages between processes, all within a protected, large-model, multiple address space environment.

Dynamic Loading

Allows multiple application programs to be loaded into node memory, placed in execution, and subsequently stopped or killed, all under user program control. Eliminates the need to reload the node operating system with each application.

Interprocess Communication

The operating system provides users with a flexible set of capabilities that help optimize communication in a concurrent processing environment. The communication interface is consistent whether communicating with other processes in the same node, to remote nodes, or to processes in the Cube Manager. Sending and receiving can be synchronous or asynchronous. Messages are automatically routed from node to node, if necessary, to reach the destination process (see Figure 8).

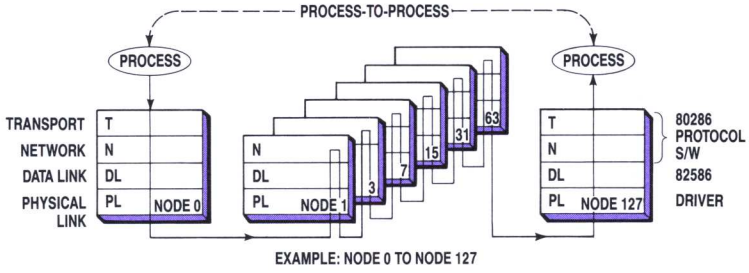

Figure 8–Automatic Process-to-Process Message Routing

Reliable message delivery service is provided between nearest node neighbors (see Figure 9). Data message length can vary from 0 to 16 KBytes. Messages larger than 1 KByte are automatically fragmented and reassembled at the destination node, transparently to the user process.

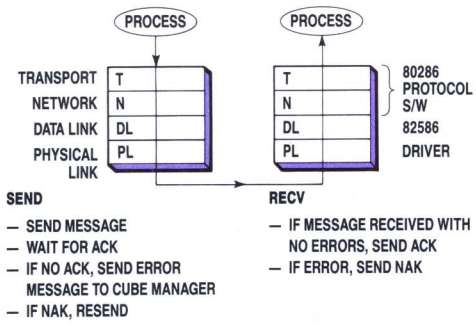

Figure 9–Nearest Neighbor Communication Protocol

	To send or receive messages, the user, via FORTRAN or C interface libraries (refer to Table 1), simply opens a channel. To send a message, the user invokes the SEND library routine with the appropriate message parameters. To receive a message, the user calls RECV with the appropriate parameters.
Process Management	Controls process scheduling by allowing each process to run for a specified interval in a round-robin fashion. The number of processes supported is limited only by available memory.
Physical Memory Management	Provides memory space for each process on the node as well as message buffering. Memory management may be controlled by the user, if desired.
Protected Address Spaces	Local descriptor tables (one for each process) provide a "shield" between the operating system and user space and between multiple user process spaces. This is hardware enforced so that the user cannot corrupt the operating system software.

Table 1. Node Operating System Services

Library Name	Function Performed
COPEN	Creates a channel for node process communication. A descriptor block for message parameters is set up in the operating system and a channel identifier (CI) is assigned and returned to the user process.
CCLOSE	Destroys the communication channel created by the **COPEN** call.
SEND	Initiates transmission of a message to another process. The return to the calling process acknowledges the **SEND** request. **STATUS** must be used to determine if the user message descriptor is available for reuse.
SENDW	Initiates transmission of a message to another process. The return to the calling process indicates that the user message descriptor is available for reuse.
RECV	Initiates the receipt of a message. The return to the calling process acknowledges receipt of the request by the operating system.
RECVW	Initiates the receipt of a message. The calling process is blocked until the message has been received.
STATUS	Informs the calling process as to the availability of the user message descriptor for another call.
PROBE	Provides the calling process with the ability to determine if messages of a specified type have been received on a channel prior to invoking a **RECV** call. It returns the length of the received message to the calling process if a message has been received.
FLICK	Relinquishes the CPU to other processes prior to completion of the scheduled time interval.
MYNODE	Returns the node number of the process that initiated the call.
CUBEDIM	Returns the dimension of the cube to the calling process. (For example, "6" for a 64-node cube.)
CLOCK	Returns a snapshot value of elapsed time, in seconds, that started at initialization of the node. Maximum period is approximately 49 days.
GREENLED	Allows process to turn on or off the node's green LED.
REDLED	Allows process to turn on or off the node's red LED.
MYPID	Returns the process ID of the calling process.

Optional Cube Software

Source Code

The sources for iPSC node communication libraries, dynamic loader, node operating system, and XENIX communication driver may be purchased. A PL/M compiler is provided with the source code to facilitate additions and changes. The product is distributed on 5.25″ floppy diskette, cartridge tape and on half-inch 9-track tape.

CUBE MANAGER DESCRIPTION

Cube Manager Hardware

The Cube Manager hardware consists of a microcomputer system, related options, and an alphanumeric terminal.

Intel System 310 Microcomputer

The System 310 is a MUTLIBUS-based microcomputer built with Intel's 80286 CPU and 80287 numeric processing unit (see Figure 10). It contains a 5¼" 140 MByte Winchester disk, a 360 KByte floppy disk, and 2 MByte ECC RAM memory. The 310 also has an integrated Ethernet interface for communicating with the Cube.

Internally, the 310 has a 7-slot card cage powered by a 360-watt power supply with the following MULTIBUS boards integrated into four slots:

iSBC 286/12 CPU board
iSBC 010CEX iLBX memory board
iSBC 214 disk controller board
iSBC 186/51 communications board

The communication board is connected to an external Ethernet transceiver which is cabled to the cube. An RS 422 link runs from the CPU board to the cube to provide the diagnostic channel.

Expansion Options

Printers and other peripherals can easily be added to the Cube Manager via a Centronics compatible printer port and three spare MULTIBUS expansion slots. One slot is reserved for iLBX boards and the other two are available for standard MULTIBUS boards (see Figure 11).

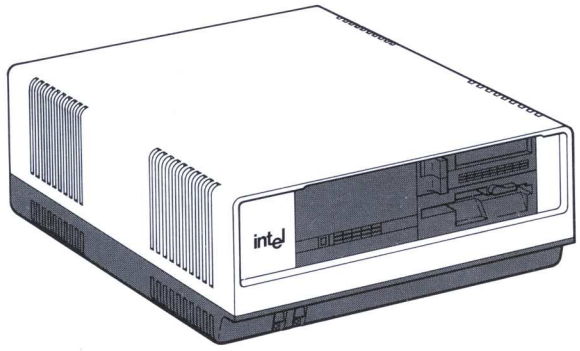

Figure 10 - System 286/310 Microcomputer

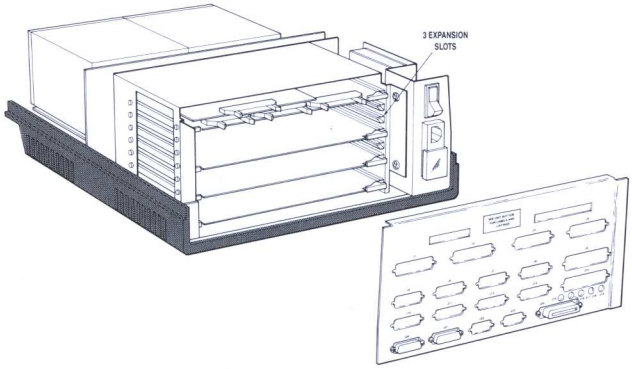

Figure 11 - System 310 Expansion Capability

Expansion options include:

Ethernet, TCP/IP network option–Enables access to remote host environment resources such as mass storage, workstations, or printers. It consists of an Ethernet front-end processor and associated software. Provides protocol package conforming to latest DOD ARPANET TCP/IP specifications. Compatible with UC Berkeley Version 4.2.

4 MByte memory expansion option–Expands memory capacity to 5 MBytes. It consists of a 2048 KByte memory board with dual port capability (MULTIBUS and iLBX interface). Includes double-bit error detection and single-bit error correction logic. An error status register provides error logging to the host CPU board.

Serial port option–Provides eight additional serial ports. It is an intelligent communications controller which functions as a single board controller or as an intelligent slave for multi-terminal communications expansion. The on-board iAPX 188 CPU provides communications control and buffer management for up to eight programmable synchronous/asynchronous channels. The board includes 64 KBytes of dual-ported parity RAM buffer space to handle messages at data rates up to 19.2 K Baud.

Over 1500 other board options–available from Intel and other MULTIBUS vendors.

Note: For more details, refer to Reference Literature section.

Terminal

The terminal is an alphanumeric terminal which meets ANSI X3.64 specifications (VT100 software compatible). It features high resolution characters (upper/lower case) on a 14″ non-glare green phosphor tilt display with detachable keyboard. The 101 keys include a numeric pad and function keys in addition to a standard typewriter keypad (see Figure 12).

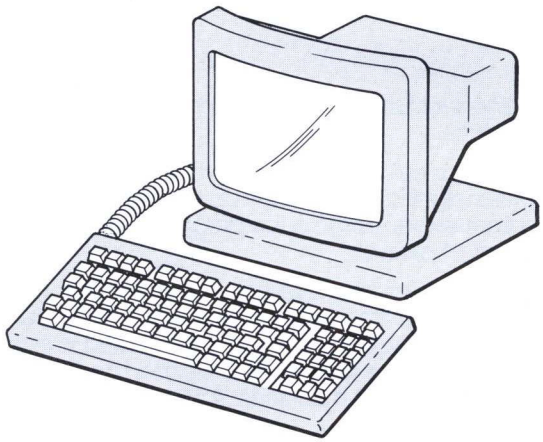

Figure 12–Cube Manager Terminal

Optional Development Station

iPSC Development Station

Providing an environment identical to the Cube Manager, the iPSC Development Station offers additional access to program development services. The station provides simulators, compilers, and libraries for creating executable load modules. A TCP/IP network interface is also provided for easily copying code modules to or from remote Cube Managers.

Cube Manager/Development Station Software

The Cube Manager software is divided into three basic categories: programming and development software, cube control software, and diagnostics.

Programming & Development Software

Operating System

XENIX 3.0 a fully-licensed derivative of UNIX System III. It includes enhancements from University of California at Berkeley as well as Microsoft and Intel. It is compatible with UNIX Version 7.

XENIX 3.0 includes:

UNIX System III–kernel, utilities, libraries, tools.

Berkeley Enhancements–curses, termcaps, C shell, VI editor, various utilities.

Microsoft Enhancements–large model C compiler, UNIX version 7 compatibility, visual shell, MS DOS file transfer features, shared data, file locking, keyed access to files.

Intel Enhancements–performance tuning to the system 286/310, device drivers for: 4- and 8-channel intelligent I/O controllers, Ethernet controller, Winchester, floppy, and tape controllers, RAM disk, console, and line printer.

FORTRAN Compiler

Meets ANSI FORTRAN 77 subset language specifications and supports real math IEEE floating-point standard, transcendentals, complex numbers, interrupt procedures, and run-time exception handling.

This is a large-model FORTRAN compiler and handles very large arrays.

Macro Assembler

Native 286 Assembler ("ASM 286") plus macro extension.

Tools

Linker
Loader

Simulator

The iPSC Simulator is included with the purchase of each iPSC System, but also is available as a separate product.

The iPSC Simulator is a software program which simulates the actions of the Intel iPSC system. Use of the simulator significantly speeds up the program developent cycle and provides facilities for locating program errors. Large cube programs will run much more slowly on the Simulator than the actual Cube, however.

The Simulator package provides documentation, simulator program, and a set of libraries. Documentation consists of installation instructions, Users Manual, Internal Design Specification, and iPSC System Overview. The package is distributed in its source code form on both half-inch 9-track tape and 5.25″ floppy and is easily installed on Cube Managers, Development Stations, and Unix BSD 4.2 or Xenix 3.0 machines. A Simulator Source Code License is required.

Note: For more details, refer to Reference Literature section.

Cube Control Software

Node Facilities

Support multiple application processes and provides reliable message delivery services to any node in the cube via the shortest route. Consistent with the Cube Manager interface for sending and receiving messages. (Refer to Table 2 for a list of FORTRAN and C services.)

Cube Manager

During initialization, each node is queried for node confidence test ("NCT") status. Control is deferred to user diagnostic procedures if an error is indicated.

After node initialization, several utilities exist (refer to Table 3) to load, start, kill, or investigate processes on the nodes. These utilities are available as a collection of XENIX programs which may be invoked from the terminal using standard XENIX shells, or executed by XENIX programs.

Diagnostics

The iPSC diagnostics consist of a set of software confidence and diagnostic tests.

The confidence tests are used to verify overall system integrity prior to normal usage. The diagnostic tests are targeted to individual boards or system modules and are used to isolate faults to this level.

System Confidence Testing

At power up, or upon reset, PROM-based confidence tests in the Cube Manager and Cube verify individual module integrity and basic system functionality. Immediately following, during the boot-up sequence, XENIX verifies the system configuration. If an error occurs during initialization confidence testing, the user can invoke system diagnostic tests as appropriate.

System Diagnostic Testing

Under user control, individual floppy-based diagnostic tests can be invoked to facilitate isolation of faults to the board/system module level. The diagnostics strategy is to verify the functionality of the Cube Manager first, Cube Manager-to-cube communications next, and then targets the Cube. Test clusters target such module elements as the central processing unit, numeric processing unit, RAM, I/O peripheral devices, Winchester, and floppy disk.

Table 2. Cube Manager Communication Services

Library Name	Function Performed
COPEN	Creates a channel for Cube Manager process communication. A descriptor for message parameters is set up and a channel identifier (C1) is assigned and returned to the user process.
CCLOSE	Destroys the communication channel created by the COPEN call.
SEND MSG	Initiates transmission of a message to another process. The return to the calling process indicates that the user descriptor is available for reuse.
RECVMSG	Initiates the receipt of a message from a designated process (node and process ID specified). The calling process is blocked until the message has been received.
CUBEDIM	Returns the dimension of the cube to the calling process.

Table 3. Cube Manager Commands

Command Name	Function Performed
LOAD	Loads a file into a node or group of nodes. Checks for sufficient memory, valid process ID, valid node ID, and valid file record.
LOADSTART	Starts execution of a file in a node or group of nodes. Checks for sufficient memory, valid process ID, valid node ID, and valid file record.
LKILL	Kills the process specified by process ID. Checks for sufficient memory, valid process ID, valid node ID, and valid file record.
LWAIT	Waits for a process (or processes) to complete on a node (or group of nodes). Checks for sufficient memory, valid process ID, valid node ID, valid file record, and active process for which to wait.
CUBELOG	Instructs the system to perform various manipulations on the system logfile; such as, emptying the contents or changing the logfile name.

SYSTEM SPECIFICATIONS

Cube	/d5	/d6	/d7
Number of Nodes	32	64	128
Number of Spare Nodes	1	2	4
Total RAM Memory	16 MBytes (expandable)	32 MBytes (expandable)	64 MBytes
Total Node-to-Node Communication Channels	80	192	448
Cube Footprint	26.75″ x 26.75″	42.7″ x 26.75″	74.5″ x 26.75″

Node

Central Processor (CPU)	Intel 80286 16 MBytes physical addressing Memory management and protection
Numeric Processor (NPU)	Intel 80287 32-, 64-, 80-bit floating point (IEEE 754) 32-, 64-bit integer 18-digit BCD operands
Memory	512 KBytes dual-port RAM with byte parity, expandable to 4.5 MBytes of RAM 64 KBytes PROM monitor (expandable)
Communication Channels	Eight total. Seven to nearest neighbor nodes via Intel's 82586 communication coprocessors and 1 Ethernet channel to Cube Manager via 82586/82501
Communication Bandwidth	H/W: 20 Mbit/sec max.
Control I/O	Reset, interrupt
iLBX II Port	Bus bandwidth: 8 MByte/sec max. 16 MByte addressing 8- and 16-bit data transfers over 32-bit path Even numbered slots are masters; odd are slaves
Indicators	Red LED Green LED
Size	2 x 4 Eurocard (9.2″ by 11″) with two 96-pin male DIN connectors

Cube Manager

Central Processor (CPU)	Intel 80286
Numeric Processor (NPU)	Intel 80287 32-, 64-, 80-bit floating point (IEEE 754) 32-, 64-bit integer 18-digit BCD operands
Memory	2 MByte iLBX memory with ECC expandable to 5 MBytes
Mass Storage	140 MByte 5¼″ Winchester disk 360 KByte 5¼″ floppy disk (DS,DD) 45 MByte cartridge tape drive
MULTIBUS Expansion Slots (IEEE 796)	3 available slots at 0.6″ spacing 1 reserved for iLBX memory, 2 for standard MULTIBUS 93 watts available power
Printer Port	Centronics compatible
User Processes Supported	50 (XENIX configuration parameter)
Global Channel	Ethernet (IEEE 802.3)

Terminal — WYSE 75

Compatibility	VT100, ANSI X 3.64
Keyboard	Detachable, 101 keys Typewriter, numeric, and function key pads 2-position tilt 16 programmable function keys (32 combinations)
Display	14″ swivel and tilt display Non-glare green phosphor 24 rows by 80 or 132 columns
Character Set	7x13 matrix in 10x13 cell 128 ASCII character set Upper/lower case with line drawing graphics Cursor block or underline selectable, with or without blinking

Physical

	32-Node Unit	Cube Manager	Terminal	Keyboard
Height	49″	6.5″	12″	2.25″
Width	16″	17″	12.3″	17.25″
Depth	16″	20″	13″	7.6″
Weight	200 lbs	40 lbs	12 lbs	2 lbs
Footprint	26.75″ × 26.75″	17″ × 20″	12″ × 12.3″	17.25″ × 7.6″

Electrical & Environmental

	32-Node Unit	Cube Manager	Terminal/Keyboard
ELECTRICAL			
AC Voltage	230 VAC ± 15%	115/230 VAC ± 10%	115/230 VAC ± 10%
AC Current	13.8 amps	5.8/2.9 amps	2/1 amps
Frequency	50/60 Hz ± 5%	50/60 Hz ± 5%	50/60 Hz ± 5%
Power	2704 watts 9226 btu/hr	367 watts 1320 btu/hr	45 watts 154 btu/hr
SAFETY/RFI/EMI	UL 478	UL 114	UL
(designed to meet)	CSA C22.2 No. 154	CSA 22.2	
	VDE 0806	FCC Docket 20780	
	VDE 0871		
	IEC 380	IEC 435	
	FCC 47 CFRJ Class A	VDE 0871	
ENVIRONMENTAL			
Operating Temp.	10-35°C	10-35°C	0-50°C
Humidity	85%, max. non-condensing	20-80%	10-90%
Altitude	0-10,000 ft	0-8000 ft	0-15,000 ft
Acoustical	50 dBA, max		

iPSC DOCUMENTATION

The iPSC Documentation Set is composed of the following reference manuals. Each one is also available separately under the order numbers shown.

iPSC Documents

Title	Order Number	Content
System Overview Manual	175278-001	Introduces the user to the iPSC system and explains the general concepts...stressing those associated with concurrent processing.
User's Guide	175279-001	Contains all of the material normally required to use the system. Includes both hardware and software information. Explains how to start up the system and develop and execute programs. Contains general housekeeping procedures as well as customer preventive and corrective maintenance. Also includes various applications program examples.
Dynamic Loader Manual	310103-001	Describes the facilities of the dynamic loader for programmers. Details the procedure for loading, starting, waiting on, and killing processes on the nodes
Simulator Manual	310104-001	Describes the facilities of the iPSC Simulator for programmers.
iPSC Diagnostics User's Guide	175306-001	Describes the diagnostic facilities available to the service engineer
iPSC Site Preparation Guide	280112-002	Provides information concerning space, power, cooling, and cabling for the iPSC system.

Reference Literature

Additional information is available from the following Intel documents:

Title	Order Number	Content
Systems Handbook	210941-003	MULTIBUS board and system product information.
Introduction to the System 310 Microcomputer	173202-002	System 310 and XENIX technical product overview/reference document.
FORTRAN 286 User's Guide	122196-000	Technical product information.

iPSC USER TRAINING

Training for one customer staff member is included with the purchase of each iPSC System. This 5-day course titled "Programming Concurrent Computers" provides students with the fundamental knowledge, strategies, skills, and tools needed to design and implement large-scale concurrent software for the iPSC.

The course is held approximately once a month at Intel Scientific Computers in Beaverton, Oregon. Study consists of lectures, guest lectures, examples and case studies, hands-on programming experience, lab assistance, and class discussions.

Training for other customer staff members is available for an additional fee. Please phone Intel Scientific Computers at (503) 629-7629 to arrange training dates and fees.

iPSC WARRANTY/SERVICE

Included in the purchase price of any base iPSC configuration are the following:

Pre-Installation Site Planning

An Intel service consultant will provide assistance regarding site and environmental requirements, including power and air conditioning needs. Intel will arrange an installation schedule that allows adequate time for site preparation and that insures a successful and timely installation. (Refer to Site Preparation Guide)

Installation

Intel field engineers will arrive on site on the designated installation date. The equipment will be assembled and system integrity verified. Basic system orientation training will also be provided.

Hardware Warranty

Intel warrants iPSC related products to conform to Intel-published specifications and to be free from defects in material and workmanship for a period of one year from date of shipment. During the warranty period, service is provided on site for the first 90 days and factory Return Receipt Authorization (RRA) is provided for the remaining nine months of the year.

Software Support

Intel provides software support for a period of one year from date of shipment. During the warranty period, service includes software updates, technical reports, and software problem reporting service. Software support also includes a HOTLINE telephone number for priority assistance. (Refer to Software Support information below for more details)

Software Support

Intel provides the following software support during the warranty period and under annual software support contract after warranty. Further details are described in iSC's Terms & Conditions of Services.

Technical Reports

Periodic publication containing solutions to known problems, advance notice of software updates, programming tips, iPSC training schedule, and listings of latest available manuals/documentation. Manual updates are automatically distributed as additions, changes, and clarifications are published.

Software Problem Reporting (SPR) Service

Intel will respond to written questions and/or problems (submitted on a standard SPR form) on system software or documentation. Intel will verify receipt of the SPR within 48 hours and will respond within 3 weeks in writing. Intel does not guarantee a resolution will always be available.

Technical Information Phone Service (TIPS)

Intel will provide telephone assistance to software related problems during normal working hours (8:00 a.m. to 5:00 p.m., Pacific Time, Monday through Friday).

Software Updates

Intel will provide, at no additional charge, software updates or new releases to existing software products. Updates include such items as problem fixes and performance improvements. Documentation is included as well as installation assistance. Individual software support items are available on an annual basis. Contact your iSC sales representative.

To be eligible for software support, customer's system software and hardware must be at the currently-supported revision level.

On-Site Hardware Maintenance Service

On-site service is available for hardware under an annual service agreement or on a per-call basis. Refer to iSC Service Agreement for complete details.

Hardware Maintenance Agreement

Intel offers a hardware maintenance agreement that allows budgeting of all hardware service costs, guarantees priority response, and insures that the hardware remains current.

Support Includes:

Unlimited service calls during contract hours
Unlimited problem determination and resolution on a priority basis
All replacement parts necessary are provided on an exchange basis
Installation of all necessary field change orders
Scheduled preventive maintenance

Per-Call Service

Field service is also available on a time and materials basis. Normal response is on a best-effort basis. Travel time, repair time, expenses, and parts are billed on an as-used basis:

Hourly rates:

8:00 a.m. to 5:00 p.m., Monday through Friday—$100/hr

Minimum charges:

2 hrs within a service zone
3 hrs outside a service zone
Parts charges are at prevailing rates

ORDERING INFORMATION

Base Configurations*

Part Number	Description
iPSC/d5	Complete 32-Node iPSC system
iPSC/d6	Complete 64-Node iPSC system
iPSC/d7	Complete 128-Node iPSC system

Cube Options*

Part Number	Description
iPSC/d4M	Complete 16-Node iPSC system with 4.5 MBytes of RAM per node
iPSC/d5M	Complete 32-Node iPSC system with 4.5 MBytes of RAM per node
iPSC/d6M	Complete 64-Node iPSC system with 4.5 MBytes of RAM per node
iPSC/d5E	Field Upgrade for existing systems. Upgrades d5 to d6 Hypercube
iPSC/d6E	Field Upgrade for existing systems. Upgrades d6 to d7 Hypercube

Cube Manager Options

Part Number	Description
iPSC/DEV	iPSC Software Development Station
iPSC/SX	Serial Expansion Kit. Adds 8 serial RS232 lines for terminal support
iPSC/MX	2 MBytes additional RAM memory for Cube Manager or Development Station
iPSC/NET	Ethernet, TCP/IP network interface
iPSC/ETC	Ethernet Transceiver & Cable hardware for use with iPSC/NET option

Optional Software

Part Number	Description
iPSC/SRC	iPSC Software Source Code. Requires iPSC Source License Agreement
iPSC/SIM	iPSC Simulator for execution on VAX/BSD4.2 systems or Intel Cube Managers/Development Stations. Simulator source code included for those who wish to re-host on alternate systems. Requires simulator license agreement.

*Each base system includes software and languages, a 1-, 2-, or 4-computational unit hypercube, a Cube Manager, alphanumeric terminal, transceivers, and necessary cables.

INTEL SCIENTIFIC COMPUTERS
15201 N.W. Greenbrier Parkway
Beaverton, Oregon 97006
(503) 629-7629

Printed in USA/SC6007/5K/LC/VJC
SCIENTIFIC COMPUTERS