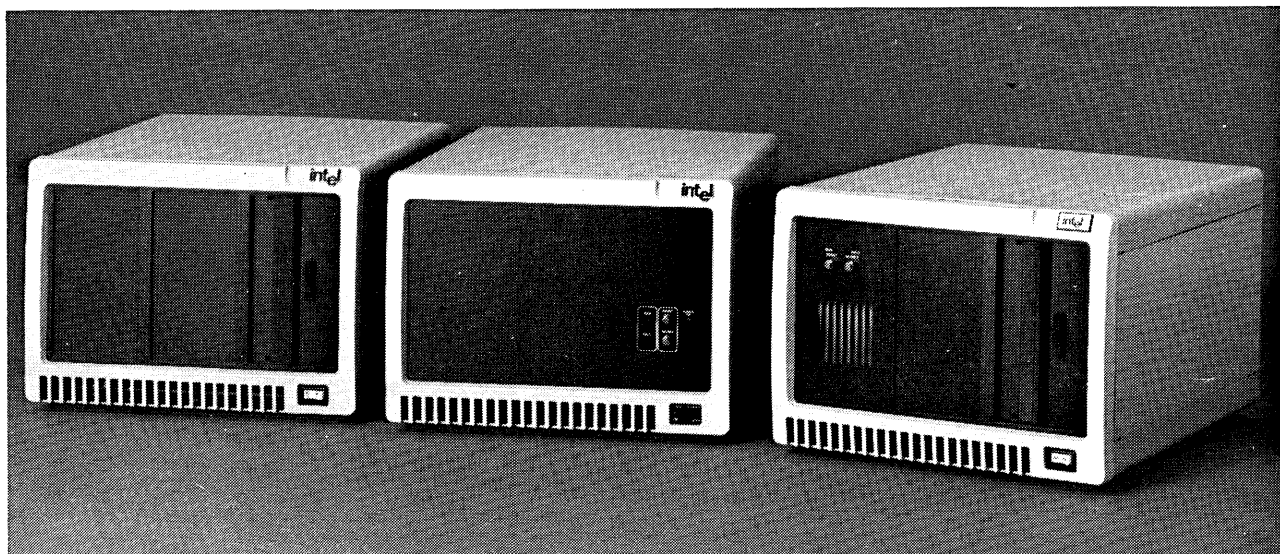




INTRODUCTION TO THE SYSTEM 86/380 AND SYSTEM 86/330A MICROCOMPUTER SYSTEMS



**INTRODUCTION TO THE SYSTEM 86/380
AND SYSTEM 86/330A
MICROCOMPUTER SYSTEMS**

Order Number: 172758-001

REV.	REVISION HISTORY	PRINT DATE
-001	Original Issue.	3-83

Additional copies of this manual or other Intel literature may be obtained from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

The information in this document is subject to change without notice.

Intel Corporation makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Intel Corporation assumes no responsibility for any errors that may appear in this document. Intel Corporation makes no commitment to update nor to keep current the information contained in this document.

Intel Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in an Intel product. No other circuit patent licenses are implied.

Intel software products are copyrighted by and shall remain the property of Intel Corporation. Use, duplication or disclosure is subject to restrictions stated in Intel's software license, or as defined in ASPR 7-104.9(a) (9).

No part of this document may be copied or reproduced in any form or by any means without prior written consent of Intel Corporation.

The following are trademarks of Intel Corporation and its affiliates and may be used only to identify Intel products:

AEDIT	iLBX	iOSP	MULTICHANNEL
BITBUS	i _m	iPDS	MULTIMODULE
BXP	iMMX	iRMX	Plug-A-Bubble
COMMputer	Insite	iSBC	PROMPT
CREDIT	Intel	iSBX	Promware
i	Intel	iSDM	Ripplemode
iICE	intelBOS	iSXM	RMX/80
iATC	Intelevison	Library Manager	RUPI
ICE	intelligent Identifier	MCS	System 2000
iCS	intelligent Programming	Megachassis	UPI
iDBP	Intellec	MICROMAINFRAME	
iDIS	Intellink	MULTIBUS	

Printed in USA/om-023/8K/0383/mr

* XENIX is a trademark of MICROSOFT, INC.

PREFACE

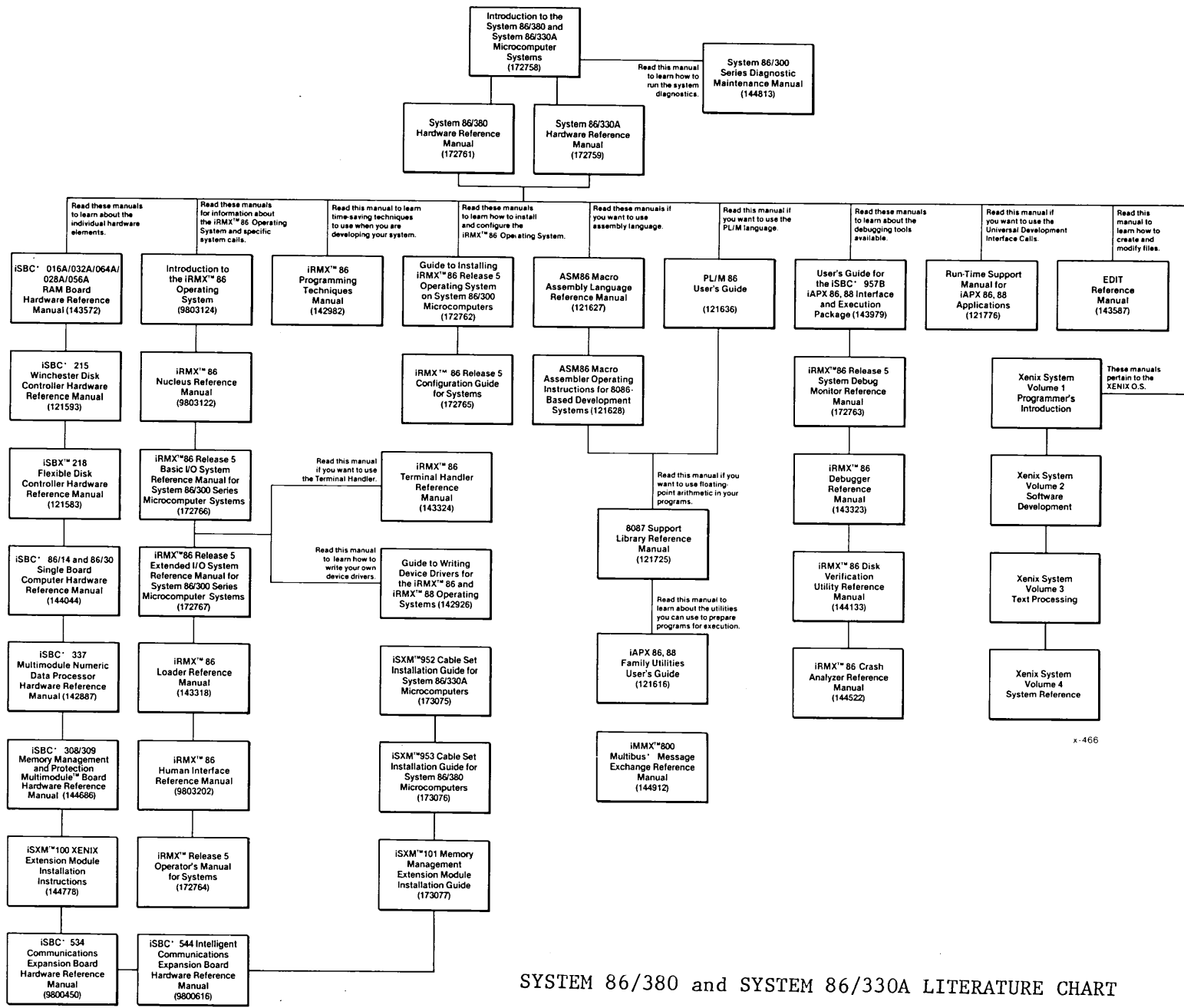
This manual provides an overview of the System 86/380 and System 86/330A Microcomputer Systems, two high-performance systems that combine Intel board products, state-of-the-art peripherals, the iRMX 86 Operating System, or the Xenix* Operating System, Intel language products, debugging and diagnostic tools into a one or two-chassis package.

Chapter 1 provides a brief introduction to the two systems. Chapter 2 covers the hardware for each system. Chapter 3 deals specifically with the iRMX 86 Operating System. Chapter 4 gives an overview of the Xenix Operating System. Chapter 5 briefly discusses the monitor and system diagnostics. Chapter 6 goes over the literature which is related to these products.

RELATED PUBLICATIONS

This manual includes a chart of the System 86/380 and System 86/330A manual set. This chart should help you decide which manuals you need to read and in which order you should read them. Chapter 6 contains brief descriptions of each of the manuals in the chart.

* Xenix is a trademark of Microsoft, Inc.



x-466

SYSTEM 86/380 and SYSTEM 86/330A LITERATURE CHART

CONTENTS

	PAGE
CHAPTER 1	
INTRODUCTION	
System 86/380.....	1-1
System 86/330A.....	1-2
System Advantages.....	1-3
Standard Intel Hardware.....	1-3
iRMX 86: A Configurable, Multi-Tasking Operating System.....	1-3
Xenix: A General-Purpose Multi-User Operating System.....	1-4
Dual Purpose Software Environment.....	1-4
Service And Support.....	1-5
About The Rest Of This Manual.....	1-5
CHAPTER 2	
HARDWARE OVERVIEW	
System 86/380 Hardware Description.....	2-1
System 86/380 Processor Chassis.....	2-1
System 86/380 Control Panel.....	2-1
System 86/380 Boards And Cardcage.....	2-2
System 86/380 Processor Chassis Input/Output Connectors.....	2-3
System 86/380 Peripheral Chassis.....	2-4
System 86/330A Hardware Description.....	2-5
System 86/330A Chassis.....	2-5
System 86/330A Control Panel.....	2-5
System 86/330A Boards And Cardcage.....	2-6
System 86/330A System Input/Output Connectors.....	2-6
System Boards Provided.....	2-7
Processor Board.....	2-7
Memory Board.....	2-8
Disk And Tape Controller Boards.....	2-8
Communications Controller Board.....	2-8
Memory Management And Protection Board.....	2-8
Winchester Disk Drive.....	2-9
Flexible Disk Drive.....	2-9
Optional Cartridge Tape Drive.....	2-9
CHAPTER 3	
iRMX 86 SOFTWARE OVERVIEW	
Processor Management.....	3-2
Priority-Based Scheduling.....	3-2
Interrupt-Driven Processing.....	3-2
Object-Oriented Architecture.....	3-3
Error Processing.....	3-5
Device-Independent I/O Management.....	3-5
Hierarchical Naming Of Files.....	3-6
Simultaneous Multiple Terminal Support.....	3-6
Configurability.....	3-7
Preconfigured Operating System.....	3-11

CONTENTS (continued)

	PAGE
Text Editor.....	3-12
Language Translators And Utilities.....	3-13
Standard Language Products.....	3-13
Optional Language Products.....	3-14
iRMX 86 System Debug Monitor.....	3-14
CHAPTER 4	
XENIX SYSTEM SOFTWARE	
General Overview.....	4-2
Xenix 86 In The Microcomputer Environment.....	4-4
Continuing Xenix 86 Support.....	4-5
A Quick Look At Xenix 86.....	4-5
The Xenix 86 Kernal.....	4-6
The Xenix 86 Shell.....	4-6
Xenix 86 Utilities.....	4-7
C Compiler.....	4-7
Editors.....	4-8
Text Processing.....	4-8
Development Tools.....	4-9
System Communications.....	4-9
Summary.....	4-9
CHAPTER 5	
SYSTEM MONITOR AND DIAGNOSTICS	
Monitor.....	5-1
System Confidence Test (SCT).....	5-1
System Diagnostic Test (SDT).....	5-2
System Analysis Test (SAT).....	5-3
CHAPTER 6	
RELATED LITERATURE	
System 86/380 And System 86/330A Manuals.....	6-1
Hardware Manuals.....	6-2
iRMX 86 Manuals.....	6-4
Debug Monitor Manual.....	6-7
Language Translators And Utilities Manuals.....	6-7
Xenix Operating System Manuals And References.....	6-8
Inter-Board Communication Manual.....	6-9
APPENDIX A - SYSTEM 86/380 SPECIFICATIONS.....	
	A-1
APPENDIX B - SYSTEM 86/330A SPECIFICATIONS.....	
	B-1

CONTENTS (continued)

PAGE

TABLES

A-1.	System 86/380 Specifications.....	A-1
A-2.	Winchester Drive Specifications.....	A-4
A-3.	Flexible Disk Drive Specifications.....	A-5
A-4.	Power Supply Specifications.....	A-5
B-1.	System 86/330A Specifications.....	B-1
B-2.	Winchester Drive Specifications.....	B-4
B-3.	Flexible Disk Drive Specifications.....	B-5
B-4.	Power Supply Specifications.....	B-6

FIGURES

1-1.	System 86/380 Microcomputer System.....	1-1
1-2.	System 86/330A Microcomputer System.....	1-2
2-1.	System 86/380 Processor Chassis Control Panel.....	2-2
2-2.	System 86/380 Processor Chassis Rear Panel	2-3
2-3.	System 86/380 Peripheral Chassis Rear Panel.....	2-4
2-4.	System 86/330A Control Panel.....	2-5
2-5.	System 86/330A Rear Panel Connectors.....	2-7
3-1.	An iRMX 86 Job.....	3-4
3-2.	Hierarchical File Naming.....	3-6
3-3.	iRMX 86 Layers.....	3-7
4-1.	Intel's Xenix Operating System	4-3
4-2.	Xenix Operating System Structure.....	4-6

CHAPTER 1. INTRODUCTION

The System 86/380 and the System 86/330A are comprehensive hardware and software system packages (see Figure 1-1) designed to solve sophisticated 16-bit OEM computational problems. Each system can be used with either of two industry-standard operating systems: Intel's iRMX 86 Operating System or Microsoft's Xenix* 86 Operating System. Both operating systems are available through Intel.

When connected to standard RS-232-compatible terminals, the systems can run off-the-shelf, commercially-available programs, or they can be used to develop software for your own system. When configured with your software products, the systems become cost-effective execution systems.

SYSTEM 86/380

The System 86/380 is the larger, more flexible of the two systems because it consists of a processor chassis with a 14-slot Multibus cardcage, and a peripheral chassis with two disk drives (see Figure 1-1). Resident in the cardcage are a processor board (8086-based) with 128K bytes of on-board random access memory (RAM), a disk controller board, and a 256K-byte random access memory (RAM) memory board. Eleven slots remain for expansion.

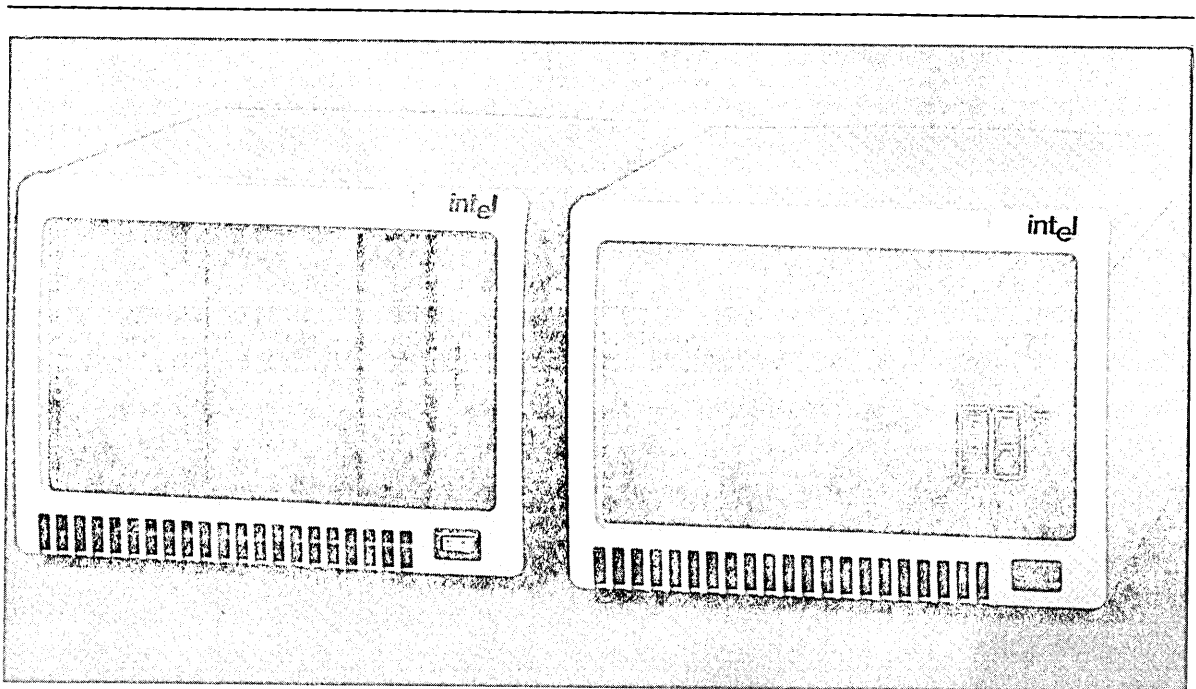


Figure 1-1. System 86/380

* Xenix is a trademark of Microsoft, Inc.

INTRODUCTION

A second chassis, the peripheral chassis, houses the 35M-byte, 8-inch Winchester drive, and an 8-inch disk drive which can handle 1M-byte, double-sided, double-density flexible disks. A second Winchester disk drive or a second flexible disk drive can be installed into a spare cavity in the peripheral chassis. For backup purposes, an optional 1/4-inch cartridge tape drive can also be installed in the spare cavity.

The two chassis are linked together with two 4.5-foot-long round cables. Each chassis is equipped with its own power supply and AC power cord.

SYSTEM 86/330A

Consisting of the same complement of boards and peripherals as the System 86/380, but residing in a single chassis, the System 86/330A (Figure 1-2) has a six-slot Multibus cardcage. Two of these slots are available for expansion. The system provides space for two peripherals. Typically the peripherals are a Winchester disk drive and a flexible disk drive.

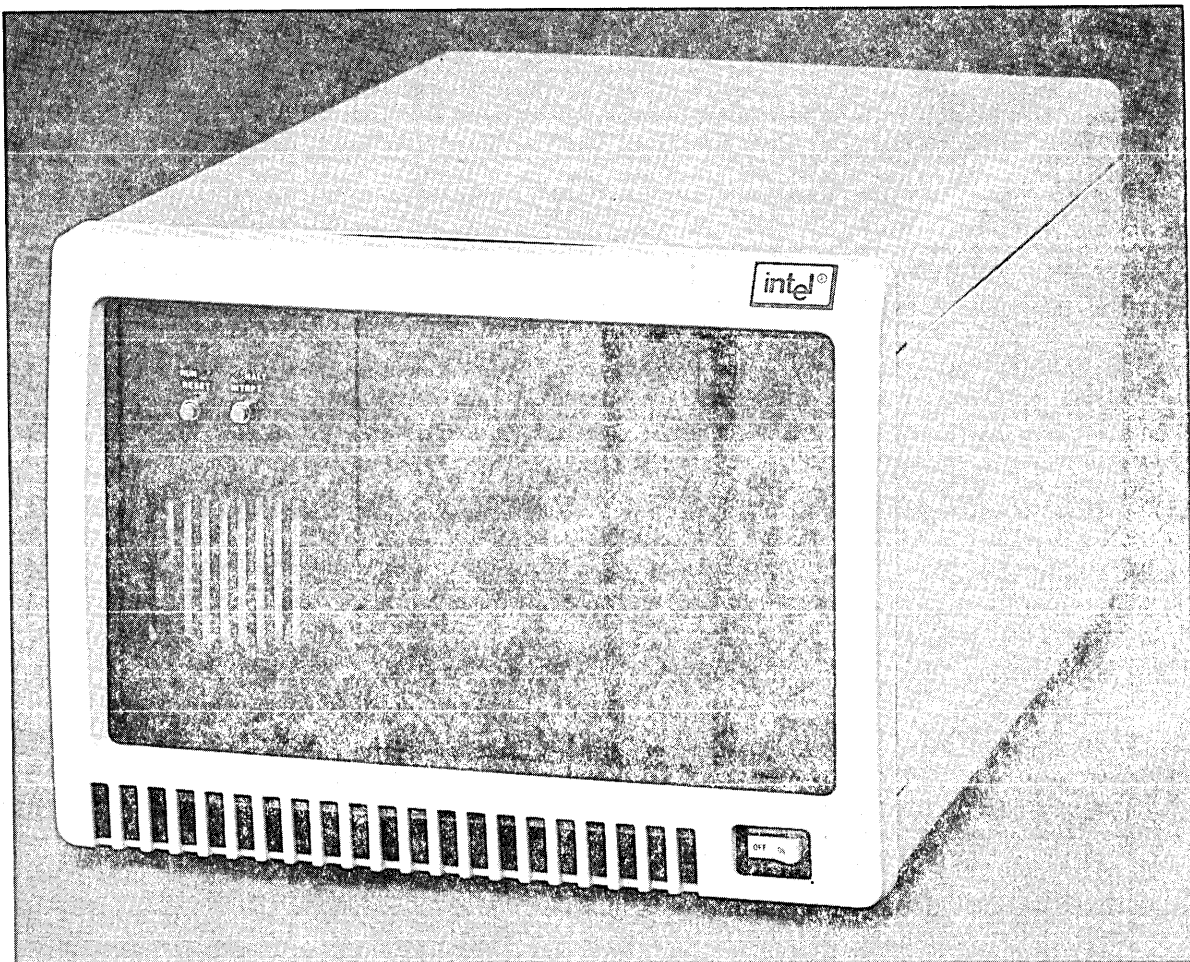


Figure 1-2. System 86/330A

INTRODUCTION

SYSTEM ADVANTAGES

The System 86/380 and System 86/330A each offer the following advantages:

- Off-the-shelf Multibus expansion boards from over 100 vendors.
- Choice of two operating systems.
- Broad selection of languages for each operating system.
- Support for multiprocessing and coprocessors.
- Dual-purpose software environment (program development and program execution).
- Available support packages.
- Available product service and technical support.
- Built-in diagnostics firmware and additional diagnostic software supplied on diskette.

The major advantages of the System 86/380 and System 86/330A are described in the remainder of this chapter.

STANDARD INTEL HARDWARE

The System 86/380 and System 86/330A each consist of a processor board, a memory board, and a disk controller board. All of these products are available individually from Intel. This provides two important benefits. First, because the application software you develop to run on your System 86/380 or System 86/330A runs on a system made up of standard parts, it will also run on Intel single board computer products that you purchase separately. Thus, you can build your initial products using the entire System 86/380 or System 86/330A and later build more-specialized products using individual single board computer products.

Second, because the systems are made up of modular parts, you can easily upgrade your system when new, more-advanced Multibus boards become available. You can also customize your system by adding existing Multibus boards which are available from Intel and many other vendors.

iRMX™ 86: A CONFIGURABLE, MULTITASKING OPERATING SYSTEM

The iRMX 86 Operating System, available with the System 86/380 and System 86/330A, complements the configurable nature of the system hardware. It is a multitasking operating system, making it very effective for real-time applications.

INTRODUCTION

It is also configurable, allowing you to change your operating environment to support new or additional hardware. With this configurable operating system, you can decrease your memory requirements by excluding parts of the operating system for which you have no use. You can also provide new or additional software features by extending the operating system.

XENIX 86: A GENERAL-PURPOSE MULTI-USER OPERATING SYSTEM

The Xenix 86 Operating System is a general-purpose time-sharing operating system available for the System 86/380 and System 86/330A. It is a multi-user system that provides a powerful command language; the ability to execute sequential, asynchronous or background processes; language translators and utilities; the flexibility to allow new device drivers to be easily integrated into the system; and flexible document preparation facilities.

SUPPORT FOR MULTIPROCESSING

The configurable nature of the System 86/380 and System 86/330A hardware and software allows the system to support a multiprocessing environment. Because the hardware is configurable, you can add additional Intel processor boards to the system.

Because the iRMX 86 software is configurable, you can add Intel's Multibus Message Exchange (iMMX 800) software to allow a processor board to communicate with other processor boards via the Multibus interface. This enables you to increase the general-purpose processing power of the System 86/380 or System 86/330A by adding boards such as the iSBC 86/05 board and the iSBC 88/25 board. For special-purpose processing, you can also add intelligent, high-performance, microprocessor-based boards such as the iSBC 550 Ethernet* Communications Controller, the iSBC 534 Serial Communications Controller, and the iSBC 88/40 Analog Measurement and Control Computer.

DUAL-PURPOSE SOFTWARE ENVIRONMENT WITH iRMX™ 86 AND XENIX 86

The System 86/380 and System 86/330A are versatile and powerful vehicles for iRMX 86 program execution. However, unlike previous microcomputer systems, they can also be used for iRMX 86 or Xenix 86 program development. The systems, when shipped with the iRMX 86 Operating System, contain ASM 86, PL/M-86 and other standard Intel language products which you can use to develop iRMX 86 application software and which you can include in your product.

* Ethernet is trademark of Xerox Corporation

INTRODUCTION

When shipped with the Xenix Operating System, the systems contain an assembler, a high-level language (C), and language utilities that you can use to develop Xenix application software for inclusion in your product.

SERVICE AND SUPPORT

Intel offers several maintenance-support options that you can tailor to your needs. These options include a board exchange program and on-site service (within areas serviced by Intel). Technical support is also available through qualified Field Application Engineers in Intel field offices. Contact your local Intel sales representative or distributor for more information.

ABOUT THE REST OF THIS MANUAL

The remaining chapters of this manual provide an overview of the software and hardware available with the System 86/380 and System 86/330A Microcomputer Systems:

- Chapter 2 provides an overview of the system hardware. It identifies each element of the system and points to other manuals where you can find more-detailed information.
- Chapter 3 provides an overview of the iRMX 86 system software. It introduces you to the iRMX 86 Operating System, and discusses related software components. This chapter also refers you to other manuals for more-specific information.
- Chapter 4 provides an overview of the Xenix system software. It introduces you to the Xenix Operating System and discusses each of the other software components. This chapter also refers you to other manuals for more-specific information.
- Chapter 5 gives a brief introduction to the system monitor program and system diagnostic tests.
- Chapter 6 contains an overview of the manuals available with each system. It lists the manuals and provides a one-paragraph description of each manual.
- Appendix A provides a detailed list of System 86/380 specifications, including power specifications, board features, disk drive characteristics, and memory addressing.
- Appendix B provides a detailed list of System 86/330A specifications, including power specifications, board features, disk drive characteristics, and memory addressing.

CHAPTER 2. HARDWARE OVERVIEW

The System 86/380 and the System 86/330A consist of similar hardware modules. This chapter describes the hardware associated with each system. The System 86/380 chassis are first described; a similar description of the System 86/330A chassis follows; descriptions of system boards and disk drives complete the system hardware overview. Appendix A and Appendix B provide detailed system specifications.

SYSTEM 86/380 HARDWARE DESCRIPTION

The System 86/380 hardware consists of a processor chassis which contains a 14-slot Multibus cardcage, an 8086-based processor board with 128K bytes of on-board random access memory (RAM), a numeric coprocessor board, a Winchester disk controller board, a flexible disk controller board, and a 256K-byte random access memory (RAM) board. A second chassis, the peripheral chassis, houses the 35M-byte, 8-inch Winchester drive, and an 8-inch disk drive for 1M-byte double-sided, double-density flexible disks. The two chassis are linked together with two 4.5 foot long round cables. Each chassis is equipped with its own power supply and AC power cord. The following sections discuss each of these hardware elements. Refer to Appendix A for System 86/380 specifications.

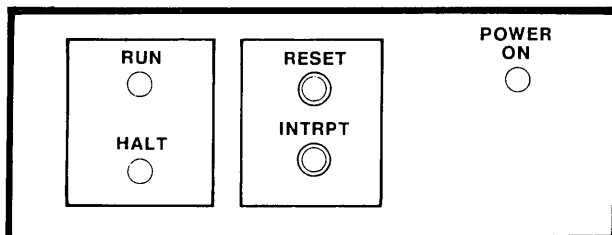
SYSTEM 86/380 PROCESSOR CHASSIS

The processor chassis consists of a 14-slot Multibus-compatible cardcage and backplane which utilizes a parallel priority resolution scheme, a control panel with switches and indicators, and a switching power supply. Three of the backplane slots are designed with extra width to accommodate a host iSBC board and Multimodule expansion board set without slot penalty.

System 86/380 Control Panel

The processor chassis control panel consists of two pushbutton switches and three indicator lamps (see Figure 2-1). One switch is a system RESET switch; the other switch is an INTERRUPT switch, which is wired to a level 1 interrupt (INT1/) on the processor board.

Two indicator lamps are used to show RUN and HALT conditions. RUN (green lamp) means that the processor board is executing an instruction. HALT (red lamp) means that the processor halt instruction (HLT) has been executed; both lamps off means the processor is idle (or that the system is powered off). A third lamp (yellow) illuminates when power is applied to the system. The power switch is a rocker type, located on the front of the chassis.



1013

Figure 2-1. System 86/380 Processor Chassis Control Panel

System 86/380 Boards and Cardcage

The System 86/380 is supplied with an 8086-based processor board with a specialized math Multimodule expansion board, a disk controller board set, and a random access memory expansion board. These boards occupy the first three slots of the 14 available slots. Multibus priority is resolved by a built-in parallel priority resolver network. As shipped, the disk controller board has the highest priority and the processor board has a lower priority.

If you order the Xenix 86 Operating System, your system must be field-equipped with an optional communications controller board, providing four additional connections for terminals; an optional memory management Multimodule board; and the necessary internal cables. The System 86/380 will support a maximum of four communications controller boards. This hardware is available as several optional iSXM extension modules. Contact your local Intel sales office for ordering information.

You can also order optional iSXM communications packages for the iRMX 86 Operating System. If you order one of the optional iSXM Communications packages for systems using the iRMX 86 product, your system will be field-equipped with an optional communications controller board, and cables, providing connections for four additional terminals.

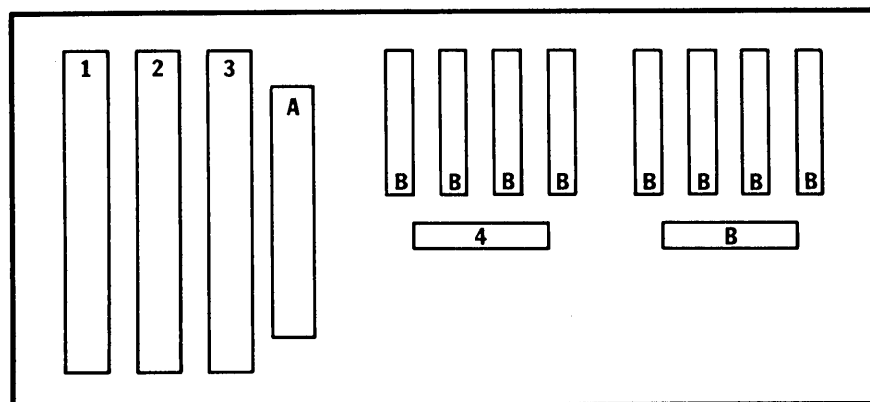
HARDWARE OVERVIEW

System 86/380 Processor Chassis Input/Output Connectors

The System 86/380 Processor Chassis rear panel provides the following ready-to-use I/O connectors (Figure 2-2):

- A 25-pin serial I/O connector for a video display terminal or other RS232-compatible I/O device.
- A 36-pin parallel I/O communications port that is supplied with signal levels which are compatible with the industry-standard Centronics printer interface.
- Two 50-pin connectors that are used to link the processor chassis to the peripheral chassis. A third connector is present when the tape option is included.

The connector plate on the back of the processor chassis is pre-punched for two styles of I/O connectors (see Figure 2-2). This includes space for 10 subminiature "D" type connectors, three 50 pin I/O type connectors and one 36-pin connector. Refer to the SYSTEM 86/380 HARDWARE REFERENCE MANUAL for additional information about the processor chassis.



- 1 = Flexible Disk Drive Link to Peripheral Chassis
- 2 = Empty 50-pin or Tape Connector
- 3 = Winchester Disk Drive Link to Peripheral Chassis
- 4 = Serial Port Connector for Keyboard/Terminal
- A = 36-pin Connector
- B = Empty 25-pin I/O Connector Cut-out

x-470

Figure 2-2. System 86/380 Processor Chassis Rear Panel

HARDWARE OVERVIEW

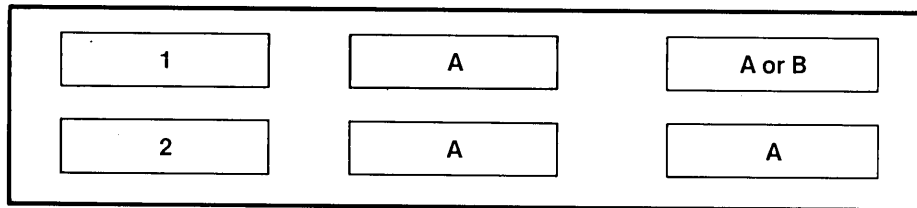
SYSTEM 86/380 PERIPHERAL CHASSIS

This chassis is shipped with one 8-inch Winchester disk drive and one 8-inch flexible disk drive. An unoccupied slot is available in the chassis for another Winchester drive, another flexible disk drive, or a cartridge tape drive. Internal connectors are available for easy installation. A switching power supply provides all power for standard drives and for the auxiliary drive, if installed.

The rear panel of the chassis supports I/O connectors whose functions are as follows (Figure 2-3):

- 50-pin connector for round cable to processor chassis for Winchester disk drive.
- 50-pin connector for round cable to processor chassis for flexible disk drive.
- 50-pin connector for round cable to processor chassis for optional tape drive.

Refer to the SYSTEM 86/380 HARDWARE REFERENCE MANUAL for additional information about the peripheral chassis.



1 = Flexible Disk Drive Link to Processor Chassis
2 = Winchester Disk Drive Link to Processor Chassis
A = Empty 50-pin Connector Cut-out
B = Tape Drive Connector

x-467

Figure 2-3. System 86/380 Peripheral Chassis Rear Panel

SYSTEM 86/330A HARDWARE DESCRIPTION

The System 86/330A hardware consists of a chassis which contains a 6-slot Multibus cardcage, an 8086-based processor board with 128K bytes of on-board random access memory (RAM), a numeric coprocessor board, a Winchester disk controller board, a flexible disk controller board, and a 256K-byte random access memory (RAM) memory board. The chassis also houses the 35M-byte, 8-inch Winchester drive, and an 8-inch disk drive for 1M-byte, double-sided, double-density flexible disks. Appendix B provides detailed System 86/330A specifications.

SYSTEM 86/330A CHASSIS

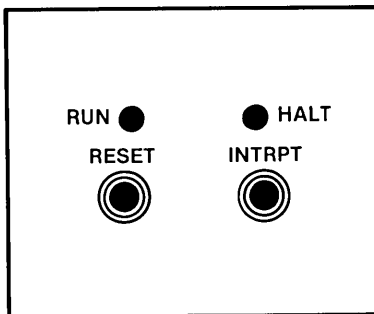
The System 86/330A chassis consists of the cardcage and backplane described above, a control panel with switches and indicators, a highly efficient switching power supply, and fans for board and power supply cooling.

System 86/330A Control Panel

The chassis control panel consists of two pushbutton switches and two indicator lamps (see Figure 2-4). One switch is a system RESET switch; the other switch is an INTERRUPT switch. This switch is wired to a level 1 interrupt (INT1/) on the processor board.

Two indicator lamps are used to show RUN and HALT conditions. RUN (green lamp) means that the processor board is executing an instruction. HALT (red lamp) means that the processor halt instruction (HLT) has been executed; both lamps off means the the processor is idle (or that the system is powered off).

The system power ON/OFF switch is located on the front of the chassis. When powered on, the switch is green.



x-469

Figure 2-4. System 86/330A Control Panel

HARDWARE OVERVIEW

System 86/330A Boards and Cardcage

The System 86/330A is supplied with an 8086-based processor board having a specialized math Multimodule expansion board, a disk controller board set, and a random-access memory-expansion board. Two slots remain for expansion. Multibus priority is resolved by a built-in parallel priority resolver network. As shipped, the disk controller board has the highest priority and the processor board has the lowest priority.

If you order the Xenix 86 Operating System, your system must be field-equipped with an optional communications controller board, providing four additional connections for terminals; an optional memory management Multimodule board; and the necessary internal cables. This hardware is available as several optional iSXM extension modules. Contact your local Intel sales office for ordering information.

You can also order optional iSXM communications packages for the iRMX 86 Operating System. If you order one of the optional iSXM Communications packages for systems using the iRMX 86 product, your system will be field-equipped with an optional communications controller board, and cables, providing connections for four additional terminals. With this board installed, one slot remains for expansion in the cardcage.

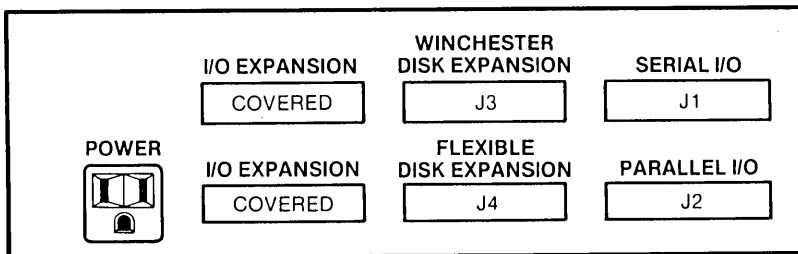
This cardcage/backplane supports the Intel Multibus configuration for 8 or 16-bit data transfers and 16, 20, or 24-bit addresses.

System 86/330A Input/Output Connectors

The System 86/330A chassis rear panel provides the following, ready-to-use I/O connectors (Figure 2-5):

- A 25-pin serial I/O connector for video display terminal, or other RS232 compatible I/O device.
- A 50-pin parallel I/O communications port that is supplied with signal levels which are compatible with the industry-standard Centronics printer interface.
- A 50-pin connector for an additional Winchester disk drive.
- A 50-pin connector for an additional flexible disk drive.

If you order an optional iSXM Communications package, an optional rear panel with additional serial I/O connectors is provided.



x-46B

Figure 2-5. System 86/330A Rear Panel Connectors

SYSTEM BOARDS PROVIDED

The Intel Multibus boards which are provided with both systems are briefly described in the following paragraphs.

PROCESSOR BOARD

The processor board consists of an Intel 8086-based single board computer upon which is mounted a numeric data processor.

The processor board is a Multibus master with a 16-bit microprocessor (8086), 32K bytes of read-only memory (ROM), 128K bytes of dual port random access memory (RAM), a Programmable Interval Timer (8253) that controls baud rates, a Programmable Interrupt Controller (8259A), a programmable USART (8251A) for RS232 serial I/O communications, and a Programmable Peripheral Interface (8255A).

The numeric data processor adds extensive high-speed numeric processing capabilities to the processor board. This Multimodule board includes an Intel 8087 numeric processor extension (NPX) device. The interface between the 8087 and the 8086 adds 60 numeric instructions and six data types.

The processor board will accept single-wide and/or double-wide Intel iSBX Multimodule Expansion boards. These Multimodule boards allow you to add features to your basic processor at low cost. One example is an iSBX 352 Serial Communications board which allows you to add another serial interface to the processor board.

HARDWARE OVERVIEW

MEMORY BOARD

The memory board provides the system with 256K bytes of additional Multibus random access memory (RAM). The system is equipped with a total of 384K bytes of RAM. Additional memory is available.

DISK AND TAPE CONTROLLER BOARDS

The disk controller boards consist of a Winchester disk controller board to which is attached a flexible disk controller Multimodule board. If your System 86/380 has the optional cartridge tape drive, a tape interface Multimodule board is also installed on the Winchester disk controller board.

The Winchester disk controller is an 8089-based I/O processor which controls the Winchester drive in the Multibus environment. It provides full sector buffering, on-board error checking and correction, automatic error recovery and retry, and transparent data-error corrections. It can detect errors of up to 32 bits in length and can correct errors of up to 11 bits in length. The board also features automatic defective-track handling.

The flexible disk controller board is a double-wide Multimodule board which is attached to the Winchester controller board. It controls all command, data, and status lines between the flexible disk drive and the processor board via the Winchester disk board.

The optional tape interface Multimodule board handles the interface between the Winchester disk controller board and the tape drive. The tape interface board is designed specifically for high-speed data transfers to and from the cartridge tape drive.

COMMUNICATIONS CONTROLLER BOARD

If your system is running with a multi-user version operating system, it will require one or more communications controller boards to allow you to attach additional terminals to the system. These boards operate as Multibus slave boards, providing additional programmable serial I/O channels for use with additional terminals.

MEMORY MANAGEMENT AND PROTECTION BOARD

If you are using the Xenix 86 Operating System, your system will require the iSBC 309 Memory Management and Protection Multimodule board. This board is designed specifically for use with the Xenix 86 product and the Intel processor board. It resides directly on the processor board. The board provides a socket for the iAPX 8087 Numeric Data Processor. The board is included with the optional iSXM 100 and iSXM 101 packages.

HARDWARE OVERVIEW

WINCHESTER DISK DRIVE

Both systems use the same type of Winchester disk drive. The Winchester disk drive is a 35M-byte, non-removable-media, 8-inch disk drive. The head/disk assembly is completely enclosed and does not require any scheduled maintenance.

The Winchester drive is mounted in the center peripheral bay with a filler panel mounted in front of the drive. A feature of the System 86/380 only, a second Winchester disk drive can be added to the empty bay in the peripheral chassis, provided it is a compatible unit.

FLEXIBLE DISK DRIVE

Both systems use the same type of flexible disk drive. The flexible disk drive is designed for use with 1M-byte double-sided, double-density disks. However, it will also operate with certain Intel single-sided, single-density, and single-sided, double-density disks.

The drive is mounted in the right-most peripheral bay such that the door access button is positioned to the right of the diskette door when viewed from the front. A feature of the System 86/380 only, a second flexible disk drive can be added to the empty bay in the peripheral chassis, provided it is a compatible unit.

OPTIONAL CARTRIDGE TAPE DRIVE

If you are using the iRMX 86 Operating System (Release 5 or newer) or the Xenix 86 Operating System in your System 86/380, an optional 1/4-inch cartridge tape drive can be added to your system. The tape drive allows you to quickly and easily copy (backup) files for archiving purposes. The files can then be loaded back (restore) into the system at a later date.

On the System 86/380, the tape drive fits into the empty peripheral bay in the peripheral chassis.

The tape drive is a "serpentine" type with a maximum formatted capacity of 18 megabytes per cartridge (600 ft. cartridge).

CHAPTER 3. iRMX™ 86 SOFTWARE OVERVIEW

The following iRMX 86 software is available with the System 86/380 and System 86/330A Microcomputer Systems:

- A ready-to-run version of the iRMX 86 Operating System (called the configured version of the Operating System), which is already configured for use in a System 86/380 or System 86/330A environment.
- A configurable version of the iRMX 86 Operating System, which you can extend to provide any additional support you need.
- A powerful, line-oriented text editor, for creating and modifying text files.
- Language translators and utilities, which allow you to assemble or compile your programs, link them together, place them in libraries, and assign absolute addresses to them.

The following features are available with all operating systems:

- A monitor which allows you to perform basic debugging and memory examination functions without the presence of an operating system.
- A debugger which works in conjunction with the monitor to provide a powerful debugging tool.
- System diagnostic programs, which check the System 86/380 and System 86/330A hardware and help you find hardware problems.

This chapter discusses each of the software elements in general and refers you to other Intel manuals where you can obtain more information.

The iRMX 86 Operating System is the heart of the System 86/380 and System 86/330A software. It is a real-time, multiprogramming operating system that manages the resources of the System 86/380 and System 86/330A Microcomputer Systems and allows them to respond to their environment. The next sections describe some of the features of the iRMX 86 Operating System.

PROCESSOR MANAGEMENT

The iRMX 86 Operating System contains the facilities to support the concurrent execution of multiple programs (called tasks). Although tasks can communicate with each other by invoking Operating System calls, each task runs independently of other tasks in the system. The Operating System resolves contentions between tasks by scheduling tasks for execution based on priority and readiness.

PRIORITY-BASED SCHEDULING

The iRMX 86 Operating System provides a priority-based, event-driven scheduling mechanism that supports up to 256 different priority levels. The Operating System uses task priority to determine which task gains control of the processor; it always gives control of the processor to the highest-priority task that is ready to run. That task continues to run until a higher-priority interrupt occurs, the task requests resources that are not currently available, or the task makes a higher-priority task ready.

INTERRUPT-DRIVEN PROCESSING

The iRMX 86 Operating System is not an active monitor that runs constantly and wastes processor time. Instead, it is passive, using no processor resources until an interrupt occurs or a task makes a call to the Operating System. The interrupts represent requests for service from peripheral devices, such as the Winchester disk or the system terminal. Operating System calls represent requests for service from tasks. When the Operating System receives an interrupt or a system call, it stops the task that is currently running, responds to the request by performing some service, and returns control to the interrupted task.

In an iRMX 86-based application system, interrupt lines have corresponding iRMX 86 priorities. When an interrupt occurs, the Operating System immediately calls a procedure to handle the interrupt and, if necessary, schedules a task of equal priority to perform further processing. By calling the procedure, which is limited in the amount of processing it can do, the Operating System maximizes response time. By scheduling an interrupt task whose priority corresponds to the priority of the interrupt, the Operating System allows you to perform extensive processing while at the same time guaranteeing that high-priority interrupts will be serviced before low-priority interrupts are serviced.

OBJECT-ORIENTED ARCHITECTURE

The iRMX 86 Operating System provides data structures, called objects, which serve as the building blocks of your application system. It also provides a number of system calls that operate on these objects and manage the resources of the system. The seven object types are:

Tasks	Tasks do the work of an iRMX 86 application system. Each task in the Operating System has the characteristics of a unique processor. Each has its own code, priority, stack, data area, registers, and status. The Operating System schedules tasks to run based on their priority.
Segments	Segments are blocks of memory that the Operating System dynamically allocates to tasks when the tasks request memory. Segment size varies according to task needs.
Mailboxes	Mailboxes provide a mechanism for intertask communication. They are structures to which some tasks send objects and other tasks wait to receive objects. Mailboxes are generally used to pass segments from one task to another, but they can be used to transfer any kind of object.
Semaphores	Semaphores provide a mechanism for task synchronization and mutual exclusion. A semaphore is an integer counter that tasks can increment or decrement. This provides a means for one task to signal another that processing is complete or that resources are available.
Regions	Regions provide mutual exclusion by guarding a specific collection of shared data. A region permits only one task at a time to access the data guarded by the region.
Jobs	Jobs are the environments in which tasks exist. Each job contains a group of tasks, the objects used by the tasks, an object directory, and an area of memory. Jobs provide a multiprogramming environment, with limited interaction between tasks in different jobs.
Composite objects	Composite objects are objects that a user can create to provide functions not already available. The Operating System treats each new just as it treats pre-defined objects, permitting the user to manipulate and obtain information about composite objects.

Figure 3-1 provides a pictorial description of a single job, showing how the job contains other iRMX 86 objects as well as a directory and an area of memory.

The Operating System provides a set of system calls that allows tasks to create objects, delete objects, manipulate objects, and request services from the Operating System. The iRMX 86 reference manuals describe these system calls in detail.

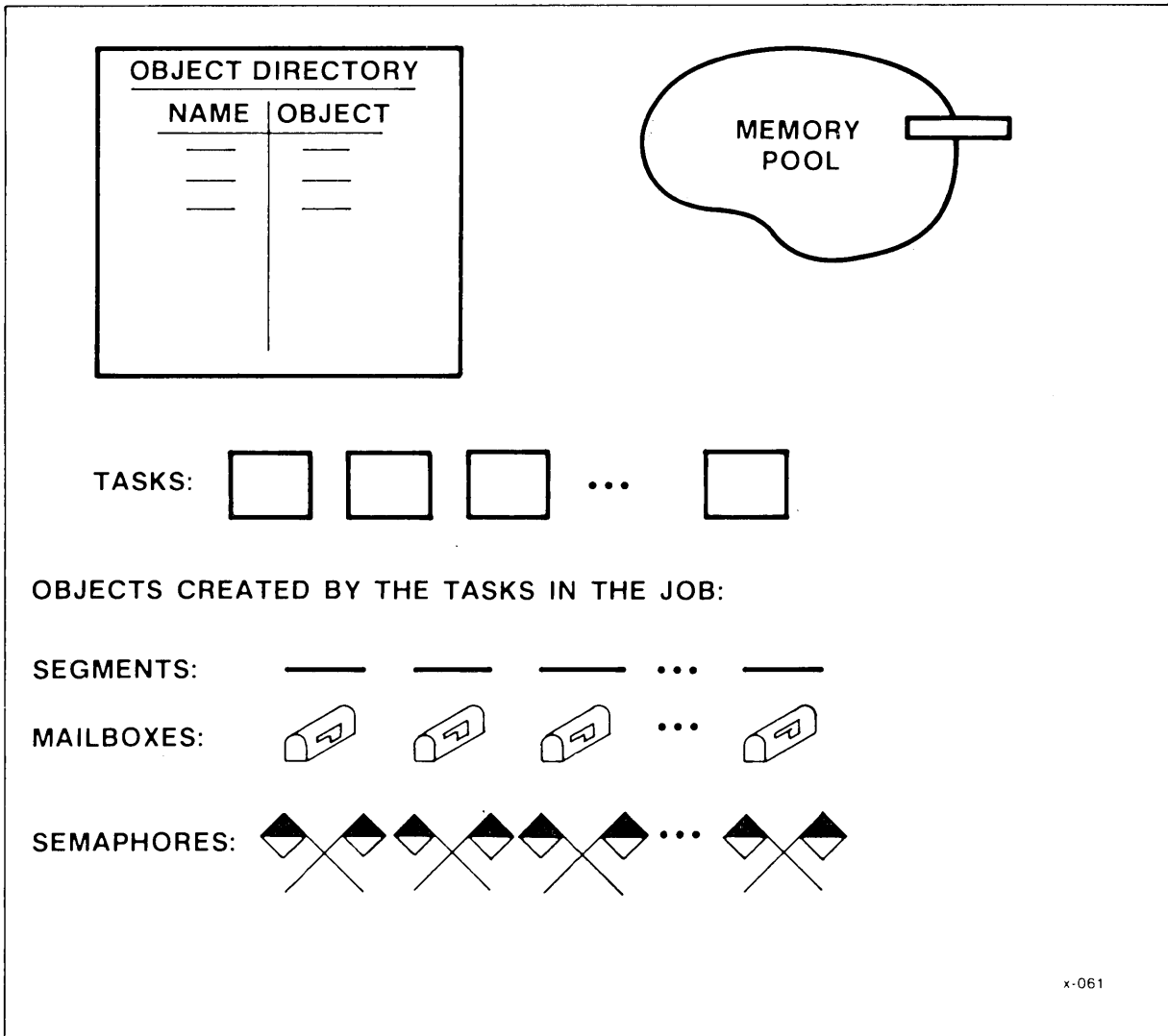


Figure 3-1. An iRMX™ 86 Job

ERROR PROCESSING

When a task issues an iRMX 86 system call, the results may not always be what the task expects. For example, a task might request memory that is not available, or it might use an invalid parameter. The Operating System provides a mechanism for transferring control, in the event of an error condition, to an error-handling routine. The Operating System permits hierarchical error handling and selective error processing.

Hierarchical error handling means that different levels of errors can be handled by different error-handling routines. For example, a system-wide error handler can process errors common to a number of tasks. Job-level error handlers can process application-specific errors. In addition, an individual task can have its own error handler, if necessary.

Selective error processing means that each task can select the type of errors to be processed by the error handlers. Errors are divided into two categories: programming errors (such as specifying invalid parameters) and environmental conditions (such as having insufficient memory to perform an operation).

DEVICE-INDEPENDENT I/O MANAGEMENT

The iRMX 86 Operating System provides a single set of system calls for managing I/O operations. These system calls allow an application program to be device independent, meaning that a system call which performs an I/O operation on one device can also perform the same operation on a different device. The iRMX 86 I/O System achieves this device independence by dividing its I/O processing software into two parts: file drivers and device drivers.

File drivers are parts of the I/O System that communicate with files. There is a separate file driver for each type of file that the I/O System supports. The file types include:

- | | |
|----------------|---|
| Named files | Named files allow applications to refer to files by name. Each file name is cataloged in a directory, to allow the file to be accessed with a minimum of overhead. Directories are special named files that store directory and access information about other named and directory files. |
| Physical files | Physical files provide a mechanism for applications to treat an entire I/O device as a single file. This type of file is typically used to communicate with simple devices such as printers and terminals. |
| Stream files | Stream files are mechanisms for communication between tasks. They allow tasks to exchange data as if the data were written and then read from a first-in/first-out file. |

iRMX™ 86 SOFTWARE OVERVIEW

The file drivers are completely device independent, allowing you to support many different kinds of devices without having to modify this part of the Operating System.

Device drivers are the device-dependent portions of the I/O System. There is a separate device driver for each type of I/O device in your system. The iRMX 86 Operating System supplies device drivers for many standard devices. It also gives you the capability to add your own device drivers to the I/O System in order to support your own special devices.

When your application program wishes to communicate with an I/O device, it first must invoke a system call that specifies a file driver and a device driver to be used with the device. This establishes the linkage between the file driver and the device driver, allowing your application program to communicate with a specific device. If, at a later time, you wish to switch to a different kind of device, you need only change one system call (the call that establishes the linkage between file and device driver). The remainder of your application program stays the same.

Refer to the iRMX 86 BASIC I/O SYSTEM REFERENCE MANUAL, the iRMX 86 EXTENDED I/O SYSTEM REFERENCE MANUAL, and the GUIDE TO WRITING DEVICE DRIVERS FOR THE iRMX 86 AND iRMX 88 I/O SYSTEMS for more information about file drivers, device drivers, and the system calls used to communicate with I/O devices.

HIERARCHICAL NAMING OF FILES

The iRMX 86 Operating System allows your application system to organize its named files into a tree-like structure like the one shown in Figure 3-2. There are two kinds of files in a hierarchical structure: data files (represented by triangles in the figure) and directory files (represented by rectangles in the figure). This hierarchy allows data to be grouped logically and to be accessed with a minimum of overhead.

SIMULTANEOUS MULTIPLE TERMINAL SUPPORT

The iRMX 86 Operating System can communicate with multiple terminals simultaneously, providing your system has the appropriate hardware (communications controller boards and necessary connectors). The Operating System provides high-level support in which each operator can run commands, development programs (such as editors or compilers), or application programs. It also provides low-level support to allow your software to communicate directly with several terminals.

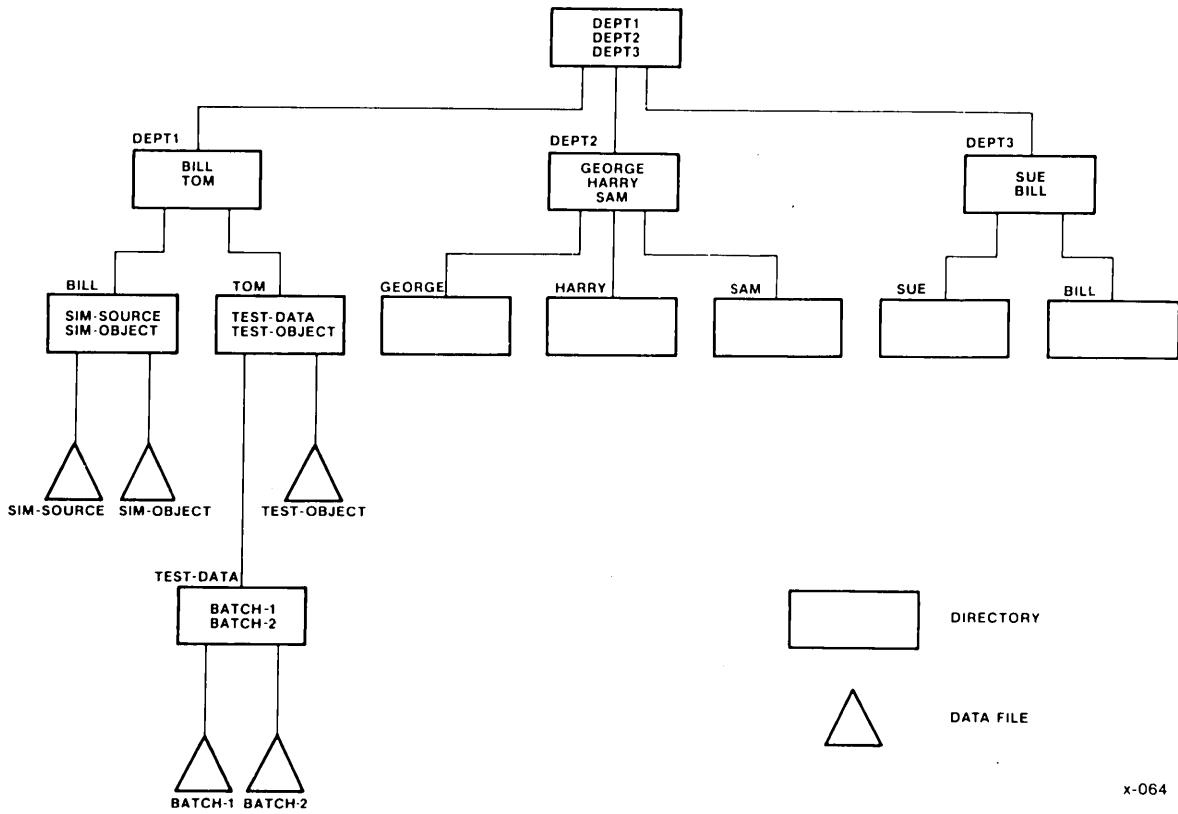
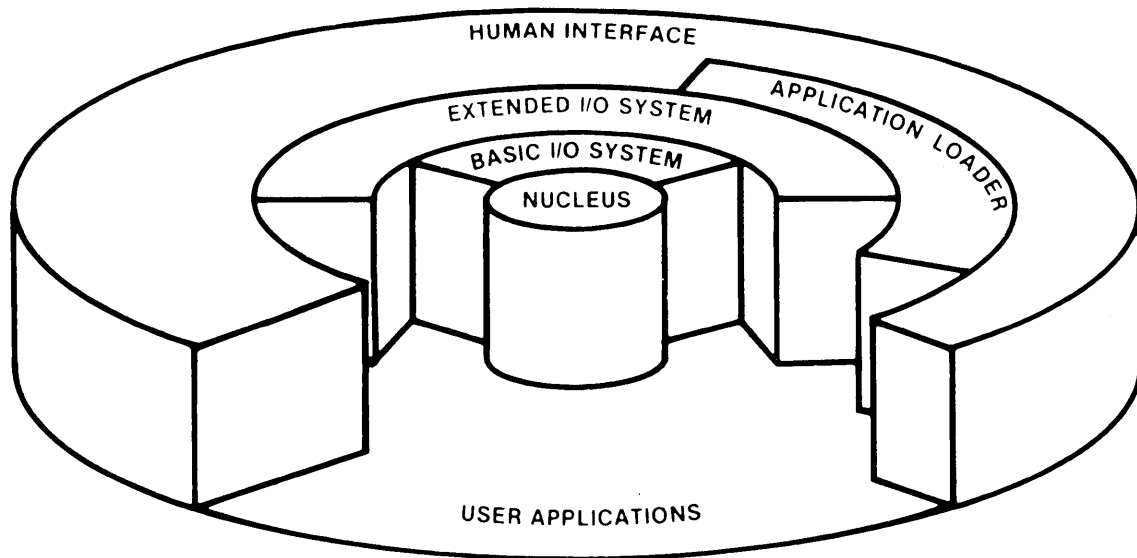


Figure 3-2. Hierarchical File Naming

CONFIGURABILITY

The iRMX 86 Operating System is the foundation on which a great variety of application systems can be built. As such, it is structured in a manner that allows users to select the particular features of the Operating System that they want and to discard any unused portions. This selection process is called configuration.

To aid in the configuration process, the Operating System is divided into a number of major pieces, called layers. When you configure the Operating System, you can select the layers that you want for your application system and discard the rest. Figure 3-3 illustrates the layers that form the preconfigured version of the iRMX 86 Operating System. Other layers can be configured into customized systems.



x-062

Figure 3-3. iRMX™ 86 Layers

The layers available with the configurable version of the iRMX 86 Operating System include:

Nucleus	The Nucleus is the core of the iRMX 86 Operating System and is required by every application system. It provides facilities that perform processor management and scheduling, interrupt management, memory management, object control, and error management. Refer to the iRMX 86 NUCLEUS REFERENCE MANUAL for detailed information about the Nucleus.
---------	--

iRMX™ 86 SOFTWARE OVERVIEW

Terminal Handler	The Terminal Handler provides a real-time interface between a terminal and the application system. It is useful for users who require the ability to communicate with the system but who do not need the full services of an I/O System. Refer to the iRMX 86 TERMINAL HANDLER REFERENCE MANUAL for more information.
Debugger	The Debugger provides the facilities for interactively debugging tasks. It allows you to debug several tasks while the remainder of the system continues to run. Refer to the iRMX 86 DEBUGGER REFERENCE MANUAL for more information.
Basic I/O System	The Basic I/O System provides an extensive facility for device-independent I/O. It supplies all file drivers and several device drivers. It implements an asynchronous interface to I/O operations, allowing tasks explicitly to overlap I/O functions with other operations. Refer to the iRMX 86 BASIC I/O SYSTEM REFERENCE MANUAL for more information.
Extended I/O System	The Extended I/O System provides a higher-level interface to files than the Basic I/O System provides. The Extended I/O System provides a simple, synchronous interface to I/O operations, one which automatically performs read-ahead and write-behind buffering. This synchronous interface also allows tasks to use logical names to refer to files. Refer to the iRMX 86 EXTENDED I/O SYSTEM REFERENCE MANUAL for more information.
Application Loader	The Application Loader provides a simple mechanism for loading application code and data files from I/O devices into system memory. It can load absolute code into fixed locations or relocatable code into dynamically-allocated memory locations. Moreover, it can load files containing overlays. Refer to the iRMX 86 LOADER REFERENCE MANUAL for more information.
Bootstrap Loader	The Bootstrap Loader provides a means of loading an entire application system into system memory from an I/O device. It can be configured to load from a specific device or to use the first device that becomes ready once the system has been started. It can also be configured to load a file specified by the operator at the terminal. A version of the Bootstrap Loader resides in PROM on the processor board, allowing you to load your application system immediately.

iRMX™ 86 SOFTWARE OVERVIEW

Human Interface The Human Interface is the interactive interface between the user and the application system. It gives the user the ability to invoke an application program from the terminal by specifying the name of the file that contains the program. The Human Interface supports multiple terminals (with appropriate hardware).

Supplied with the Human Interface are several programs called commands. The Human Interface commands perform many operations, including:

- Creating, copying, renaming, and deleting files.
- Loading and starting application programs.
- Formatting and verifying device volumes.
- Backing up and restoring files on devices.
- Reading commands from a file, instead of from the terminal.
- Communicating with the iSBC 957B package to debug programs and to copy files to and from an Intel development system.

The Human Interface also provides several system calls that an application program can invoke to access Human Interface services. Refer to the iRMX 86 OPERATOR'S MANUAL and the iRMX 86 HUMAN INTERFACE REFERENCE MANUAL for more information.

Universal Development Interface The Universal Development Interface (UDI) provides a standard, flexible protocol to allow language translators, language run-time packages, and other software development tools to run on the iRMX 86 Operating System. It consists of several system calls that you can use in your programs to allow them to run on any Operating System that supports UDI.

In addition to being able to select the layers of the Operating System that you want to include in your application system, you can also select desired parts of the individual iRMX 86 layers.

You can include or exclude features, include or exclude system calls, and specify hardware port addresses so that your customized Operating System can run on any iAPX 86- or iAPX 88-based system.

A utility called the Interactive Configuration Utility (ICU) is supplied to guide you through the configuration process. The ICU displays a series of "menus" which describe the features available. By responding to the menus, you can select or change your configuration. The ICU also allows you to save the results of a previous configuration, so that you can make a small change quickly without stepping through all the questions. Refer to the iRMX 86 CONFIGURATION GUIDE for more information about the configuration process.

Because the Operating System is configurable, it also allows you to add support for loosely-coupled multiprocessing. The Multibus Message Exchange (iMMX 800) software package provides the software needed to allow communications between processor boards via the Multibus interface. If you add the iMMX 800 package, your iRMX 86 tasks can communicate with tasks running on other processor boards. The other boards can be operating under control of the iRMX 80, 86, or 88 Operating Systems. Refer to the iMMX 800 MULTIBUS MESSAGE EXCHANGE REFERENCE MANUAL for more information about iMMX 800 software.

The iRMX 86 Operating System available with the System 86/380 and System 86/330A Microcomputer Systems includes both a configurable version and a preconfigured version. With the preconfigured version, you can start running your System 86/380 or System 86/330A Microcomputer System immediately upon power up. You can use this "standard" system to tailor the configurable version of the Operating System to your needs and generate your own customized application system. Then, you can bootstrap load your customized system and continue from there.

Because the iRMX 86 Operating System is configurable, you can also customize your system to allow communication between the System 86/380 and System 86/330A Microcomputer System and devices on an Ethernet network.

PRECONFIGURED OPERATING SYSTEM

One of the diskettes available with the System 86/380 or System 86/330A Microcomputer System, the System Installation Diskette, contains as one of its files a version of the iRMX 86 Operating System that you can run immediately. This version, called the preconfigured version of the Operating System, is configured to run on the standard System 86/380 and System 86/330A hardware. As illustrated earlier in Figure 3-3, it contains the following fully-configured layers:

- Nucleus
- Basic I/O System, which includes device drivers for the following devices:
 - Terminal
 - Flexible and Winchester disk drives (iSBC 215 and iSBX 218 controllers)
 - Cartridge tape drive interface board (iSBX 217 board)
 - Line printer with parallel interface
- Extended I/O System
- Application Loader
- Human Interface
- Universal Development Interface (UDI)

In addition to these layers, the preconfigured version also contains a job which initializes the iRMX 86 System Debug Monitor and then deletes itself. The System Debug Monitor is an extension of the iSBC 957B monitor which allows you to debug your application system by examining iRMX 86 data structures. These monitors are discussed later in this chapter.

Finally, the preconfigured version includes the iRMX 86 Bootstrap Loader, which resides in PROM on the System 86/380 and System 86/330A Microcomputer System, and is used to load the Operating System from diskette and to begin system operation.

TEXT EDITOR

The System 86/380 and System 86/330A Microcomputer System contains a sophisticated, line-oriented editing facility called EDIT. You invoke EDIT by entering a command at the terminal. You can then use EDIT commands to create and modify files of text. EDIT provides a number of simple, one-character commands to:

- Add text (A)
- Insert text (I)
- Move text (M)
- Substitute text (S)
- Print text at the terminal (P)
- Read text from a file (R)
- Write text to a file (W)

These easy-to-use commands allow you, if you are unfamiliar with EDIT, to begin creating and modifying files of text immediately, without having to spend considerable time learning how to use the editor.

Later, when you become familiar with the basic editing facilities, you can take advantage of EDIT's more powerful features to perform extensive global editing functions with compact, one-line commands. You can also use the EDIT macro processor, which allows you to process a complex string of commands by entering a single command and which allows you to define blocks of text which can be included anywhere in the text file.

For more information about EDIT, refer to the EDIT REFERENCE MANUAL, which contains a tutorial to lead you step-by-step through the basic EDIT commands. It also contains a comprehensive description of each EDIT command. The next chapter of this manual contains an example which shows you how to use EDIT when writing a program.

LANGUAGE TRANSLATORS AND UTILITIES

Included with the iRMX 86 Operating System as standard products are several language translators and utilities. Other language products are also available as options. The following sections describe these products.

STANDARD LANGUAGE PRODUCTS

When you choose the iRMX 86 Operating System, the System 86/380 and System 86/330A Microcomputer System contains several language translators and utilities which you can invoke by entering commands at the terminal. These translators and utilities allow you to compile or assemble programs, link programs together, assign absolute addresses to programs, create libraries of programs, and convert absolute object modules to hexadecimal format. The translators and utilities supplied with the iRMX 86 Operating System consist of:

- ASM86 The 8086/8087/8088 macro assembler.
- PLM86 The PL/M-86 compiler.
- LINK86 The 8086 Linker, which combines individually-compiled object modules into a single, relocatable object module.
- CREF86 A program that scans 8086 object modules to provide a cross-reference of external and public symbols between multiple modules.
- LOC86 The 8086 Locator, which assigns absolute addresses to relocatable object modules.
- LIB86 The 8086 Librarian, which creates and maintains object module libraries.
- OH86 A program which converts absolute object modules to hexadecimal format.

With these products you can create executable programs which you can invoke from the terminal. You can also create application tasks and configure the iRMX 86 Operating System to include these tasks. Finally, if you are an OEM (original equipment manufacturer) you can include these languages with your end product to give your customers the ability to program the System 86/380 and System 86/330A Microcomputer System. See the Intel Master Software License Agreement for your rights concerning passing software to your customers.

All the Intel language products supplied with the System 86/380 and System 86/330A language products are compatible with those available on Intel Microcomputer Development Systems. Thus you can develop part of your application on a development system such as a Series III and part of it on System 86/380 and System 86/330A Microcomputer Systems.

iRMX™ 86 SOFTWARE OVERVIEW

For more information about the language translators and utilities, you should refer to the individual manuals for the specific products. These manuals include:

ASM86 MACRO ASSEMBLY LANGUAGE REFERENCE MANUAL

ASM86 MACRO ASSEMBLER OPERATING INSTRUCTIONS FOR 8086-BASED DEVELOPMENT SYSTEMS

PL/M-86 USER'S GUIDE

iAPX 86, 88 FAMILY UTILITIES USER'S GUIDE

OPTIONAL LANGUAGE PRODUCTS

In addition to the Intel standard language products, you can also obtain the following Intel language products for use on the System 86/380 and System 86/330A Microcomputer System while running the iRMX 86 Operating System:

Pascal-86 A Pascal compiler that provides a strict implementation of the proposed ISO standard. It also provides extensions of the language to allow you to write programs specifically for microcomputers.

FORTRAN-86 A FORTRAN compiler that provides total compatibility with existing Intel FORTRAN-86 code, plus many new language features listed in the ANSI FORTRAN-77 language standard.

Contact your local Intel sales representative or distributor for more information about optional Intel language products.

Also, because the iRMX 86 Operating System supports UDI (Universal Development Interface), a wide variety of optional language products, both compilers and interpreters, are available from independent software vendors for use on the iRMX 86 Operating System. Contact your local Intel sales representative or distributor for more information about the optional language products that are available through independent software vendors.

iRMX™ 86 SYSTEM DEBUG MONITOR

The iRMX 86 System Debug Monitor (SDB) is an extension of the iSBC 957B monitor. It adds commands to the monitor to allow you to examine iRMX 86 objects directly. If you include the SDB in your application system, you can use the SDB commands whenever you access the iSBC 957B monitor, after bootstrap loading the Operating System. Refer to the iRMX 86 SYSTEM DEBUG MONITOR REFERENCE MANUAL for more information.

The preconfigured version of the Operating System includes the SDB. The configurable version of the Operating System contains libraries for the SDB so that you can include the SDB in your customized Operating System.

CHAPTER 4. XENIX 86 SYSTEM SOFTWARE

The Xenix 86 Operating System is a general purpose, multi-user, interactive, time-sharing system specifically designed to make the user and programmer environment efficient and productive. The Xenix 86 product has the following features:

- A hierarchical file system that permits organization of files and directories in a logical and flexible way.
- A powerful, easy-to-use command language that can be tailored to meet your specific needs.
- The ability to execute sequential, asynchronous, and background processes.
- A set of related context editors.
- A high-level programming language, C, conducive to structured programming.
- Flexible document preparation and text processing facilities.
- Sophisticated "desk-calculator" packages.

In addition, the Xenix 86 Operating System has many features which are considered "user-friendly:"

- Simple and consistent naming conventions. That is, names can be absolute or relative to any directory in the file system hierarchy.
- Mountable and dismountable file systems and volumes.
- File linking across directories.
- Automatic file space allocation and de-allocation that is invisible to the user.
- A complete set of flexible directory- and file-protection modes that allows all combinations of read, write, and execute access for the owner. Protection modes can be set dynamically.
- Facilities for creating, accessing, moving, and processing files, directories, or sets of these in a simple, uniform, and natural way.
- Easy to implement file- protection and file-locking schemes.

XENIX 86 SYSTEM SOFTWARE

- Automatic disk-recovery facilities to restore data in the event of a power failure.
- Device-independent I/O operation: each physical device, from interactive terminals to main memory, is treated like a file, allowing uniform file and device I/O operation.

GENERAL OVERVIEW

The Xenix 86 Operating System is derived from the well known UNIX* Operating System, which is available for many computer systems. Xenix 86 is a fully licensed derivative of UNIX, functionally equivalent to UNIX version 7, and includes features found in UNIX System III and Berkeley UNIX version 4.1. Figure 4-1 shows a chart of how Xenix 86 software has evolved, and shows some of the features available to the Xenix 86 user. Xenix 86 system installations are typically engaged in applications such as commercial applications, software development, computer science education, and document preparation.

In addition to the operating systems proper, the following software is included with the Xenix 86 product:

- A "C" compiler
- A text editor
- An assembler
- A linking loader
- A symbolic debugger
- A text formatter
- Phototypesetting and equation-setting programs

Xenix 86 is a multi-processing (multi-tasking) and multi-user (time-sharing) operating system. In the Xenix 86 system, a task is a process. Xenix 86 allows multiple users to perform multiple tasks and also allows a single user to perform more than one task at once. Most Xenix 86 modules are designed to facilitate interactive use. The output of one process can easily be redirected to the input of a concurrent process.

In the Xenix 86 operating system, numerous users may link to one file (with appropriate permissions) and place the links anywhere within their respective directory systems. The file system therefore becomes a network to Xenix 86. This is practical because linked users do not erase files on Xenix 86; they simply detach the file from their own file directories.

* UNIX is a trademark of Bell Laboratories

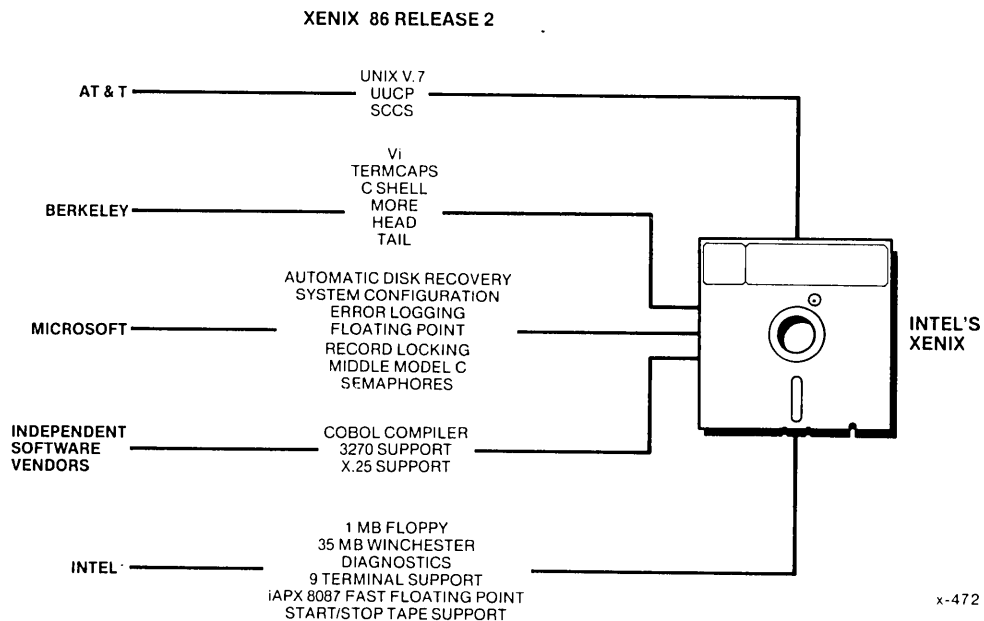


Figure 4-1. Intel's Xenix 86 Operating System

When no links remain to a particular file, Xenix 86 itself erases the file. To Xenix 86, any user file is only a string of bytes; structuring is left entirely to the programs that access the file. Filespace allocation is dynamic.

One of the most creative aspects of Xenix 86 is its I/O generality. Files, physical devices, processes (at the time of their creation), and even communications channels to other systems are all addressed with the same syntax. This makes them appear essentially interchangeable to executing programs. Combined with the flexible I/O redirection facilities, this allows programs to be written without any preconceived notion of their input sources and output destinations.

Thus, a single program may be alternately used in interactive terminal work, as a file-to-file processor, and as an intermediary between two other processes.

XENIX 86 SYSTEM SOFTWARE

The standard programming language with Xenix 86 is C. The C language is successful in both high and low-level work because it combines features of both high and low-level languages. It has a large set of powerful operators, extensive data types, facilities for handling pointers and addressing arrays, structured control flow, and a substantial I/O library. Moreover, it is equally at home with machine concepts such as bytes and physical addresses, and it was designed with very few assumptions about the environment in which it would run. C is a compact language, and its efficiency often rivals the efficiency of an assembler.

XENIX 86 IN THE MICROCOMPUTER ENVIRONMENT

One of the most appealing characteristics of Xenix 86 is that it is very easy to learn and use, and yet is very powerful and sophisticated. Since UNIX has been successfully used since 1970, many sources of information exist for UNIX and Xenix 86. Multitudes of text books, reference materials, user's groups, software tools and utilities, and information exchange channels are available to the UNIX and Xenix 86 user. Much of this information is in the public domain or in the case of software, may be purchased without licensing fees.

Many of the lesser-known Xenix 86 features make it a highly productive system, compared to other operating systems. For example, with its powerful command language and scripts (shell procedures), Xenix 86 allows many tasks to be accomplished without writing a C program. In addition, another feature, called a filter, passes the text of a file through various generic filters to accomplish spelling verification, sorting, and printing. Another example of system efficiency and productivity comes from the use of "pipes" to connect processes together. Pipes are uni-directional data channels which are accessed through the standard file system primitives.

Both Xenix 86 and the C language are highly portable. This means that many programs, I/O routines, and C modules which are written on one vendor's 16-bit microprocessor system (such as Zilog's Z8000) may easily be used on another vendor's 16-bit microprocessor system (such as Intel's 8086), with minimal conversion effort. The C language itself was developed with portability in mind.

The Xenix 86 system is based on separate modules called "tools." This is particularly useful to microcomputer users and OEMs because the most useful tools can be upgraded and enhanced, while those that are not used can be deleted or archived.

Another advantage of the Xenix 86 operating system is the abundance of experienced UNIX and Xenix 86 programmers in the microcomputer business. Because Xenix 86 retains the original concept of UNIX and because Xenix 86 is written mostly in the C language, a large base of programmers is available to begin your application work. Rarely can this statement be made about a microcomputer operating system.

CONTINUING XENIX 86 SUPPORT

As stated in Microsoft Corporation's Xenix 86 Product Abstract, "Microsoft has incorporated the strengths of the UNIX design and the same field-tested code that has been developed over the last ten years into Xenix 86. The goal of the Xenix 86 project is to produce an operating system environment that is fully upward-compatible with Unix so that Xenix 86 users can take advantage of the growing body of Unix-compatible software. This operating environment will be fully supported by Microsoft. In addition, the Xenix 86 Operating System will be constantly improved to better meet the needs of the commercial marketplace." (From The Xenix 86 Operating System Abstract, January 1982; Microsoft Corporation.)

A QUICK LOOK AT XENIX 86

This section presents a brief look at the operating system from a technical standpoint. If you are familiar with other microcomputer operating systems, this section may help you compare those systems to the Xenix 86 Operating System. Additional Xenix 86 reading references are listed in Chapter 6.

Conceptually, Xenix 86 can be thought of as being composed of three major elements: the kernal, the shell, and one or more utilities. Figure 4-2 shows a diagram of the Xenix 86 structure and some typical utility programs. The following sections briefly cover each of these topics.

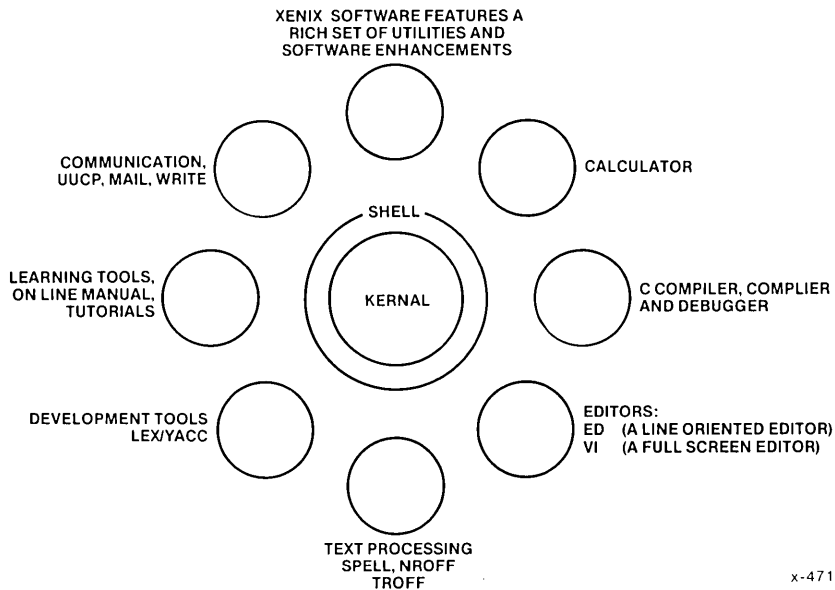


Figure 4-2. Xenix 86 Operating System Structure

XENIX 86 SYSTEM SOFTWARE

THE XENIX 86 KERNAL

The kernal is the central system controller, or nucleus of the system. It performs, transparently to the user, such tasks as memory management, file protection, process synchronization, implementation of the file system, process scheduling, and memory protection. The kernal is written almost entirely in the C language and is considered to be highly "programmer-friendly," meaning that it is easy to learn and understand. Of course, most application programmers and users never need to directly access to the kernal.

Of particular interest to programmers is the Xenix 86 file-protection system. File protection means that another user or process can be prevented from accessing a file. All Xenix 86 files can be protected to ensure data security. Several layers of protection are available to the programmer or system administrator. For example, a file can be allowed access by a particular user only, by a team of specific users, or by another system. Protection modes include write capability, read capability, and execute capability.

Related to the file protection scheme, but slightly different, is memory locking. Memory locking allows a process to be "locked" in memory. Xenix 86 in particular has improved on UNIX's memory locking feature to reduce fragmentation of main memory when a process is locked against swapping (multi-user access).

THE XENIX 86 SHELL

The shell can be thought of as the interface between the kernal and the utility program. The shell is sometimes described as a recursive command line interpreter. Its main functions in the Xenix 86 Operating System are to parse commands and arguments, substitute variables, search for commands, and subsequently execute commands. In a sense, the shell is the "human interface" to the kernal.

One of the most useful features of the shell is its ability to feed the output of one program directly into the input of another program. This procedure can be repeated through a chain of several programs, if necessary. Many of the programs in this chain can be "filters" which process text or perform formatting functions on the way to a final output process.

Command lines may contain multiple commands. With simple keystroke conventions, these commands can be specified to execute one at a time, with each command finishing before starting the next one, or they can be specified to execute immediately. In the immediate mode, commands can execute in the background. Entire files of commands can be written and executed by specifying the name of the file. These are called "scripts" in the Xenix 86 system.

XENIX 86 SYSTEM SOFTWARE

In summary, the shell language provides a very powerful and convenient environment in which to execute commands. Commands themselves are simply executable files within the file system, and the shell offers several sophisticated ways of executing these files. As previously stated, the shell supports command libraries, sequential multiple execution, background processing, redirection of terminal I/O, chaining of commands through pipes, and batch command files supporting an advanced structured control language. To the programmer this means more productivity and greater flexibility in customizing user programs.

XENIX 86 UTILITIES

In general, the utilities are programs which aid in the development of your software and help in the preparation of documents to support your software. The Xenix 86 system has an extensive and rich set of utilities that come with the product. In addition, many optional utilities are available from other vendors for special-purpose applications.

The Xenix 86 product includes the following major utilities: C compiler and debugger, two editors (ED and SED), a text processor (NROFF), several learning tools (on-line manual, tutorials), development tools (LEX/YACC), and system communications (UUCP, mail write). Many others are included with Xenix 86. This section briefly describes some of these utilities.

C Compiler

The fundamental function of the C compiler is to produce executable programs by processing C source files. All programs must be compiled before they can run. The compilation process involves several phases: preprocessing, optimization, generation of assembly code, assembly, linking, and finally, loading.

In the preprocessing phase, the C source program is examined for macro definitions and include file directives. The optimization phase is optional during initial program development, but can be used in the final compilation pass. Its purpose is to increase execution speed and to decrease the size of the program. Since the optimization phase is quite time-consuming, it is generally omitted in the debugging phase of program development.

The assembly code is generated and subsequently stored in one file before assembly, and saved; then after assembly, a second file is created for use during linking and loading. In the linking and loading phase the file created during assembly is loaded into memory along with any required modules. The modules are then linked together to create a final executable (ready-to-run) module. The loader parameters can be specified separately under a special loader program (ld).

All phases of compilation can be done separately with individual commands or automatically, from start to finish. During this compilation, each phase creates a temporary file that is used by the next program in the sequence. The final output is the load image that is loaded into memory when the final executable file is run.

Editors

Xenix 86 comes with three editors: ED, SED and VI. ED is the standard, interactive line editor used to create and modify text. The text can be a program, a document, or data for a program. The simplicity of ED makes it quick to learn and easy to use. It has two basic modes, a command mode and a text entry mode. ED is not a screen editor; that is, you can edit only one line at a time.

SED, on the other hand, is not an interactive editor. It is intended for more global editing functions. When using SED, you typically specify intended changes before starting the program. Then SED processes the changes in a background mode.

VI (visual editor) is a full screen editor, meaning that you can edit anything you see on the screen by simply moving the cursor to the appropriate location and then making the desired changes. VI has at its disposal all of ED's commands plus many other unique commands. For example, VI has advanced pattern-matching commands, undo and redo commands, and the ability to define your own commands with different sets of parameters.

Useful VI options include autoindenting, ignorecase, and number. Autoindenting allows positioning lines forward or backward to a previously selected alignment. Ignorecase allows pattern matching to be performed without regard to case. Uppercase and lowercase are equally recognized. Number displays line numbers for input and output lines.

Text Processing

In the Xenix 86 system, text processing means preparing a document for final output. Text processing is very similar to word processing. That is, many types of documents can be quickly created, edited, and printed. These document types include letters, technical specifications, reference manuals, schedules, articles, and internal correspondence papers.

The main text processors with Xenix 86 are called NROFF and TROFF. NROFF drives ASCII terminals, while TROFF drives a specific phototypesetter. A third program, called MS, is a user-friendly standardized manuscript preparation package for use with NROFF and TROFF. Together, they are capable of producing a staggering variety of document formats, with an equally impressive number of attributes. For example, you can produce a technical paper with different font styles throughout the document, such as headers and footers, automatically numbered sections, dual- or single-column pages, landscape or portrait style, with footnotes, and with many other common publications attributes.

Development Tools

Xenix 86 comes with a wealth of development tools for software programmers. Two such tools are LEX and YACC. The LEX tool is actually a program generator designed for lexical processing of character input streams. The YACC tool provides a general means for structuring input to a program. The LEX tool is concerned with the low-level statements in an input stream, while YACC deals with the high-level structure of the input stream. These two input utilities can greatly increase the ease and efficiency by which your programs are prepared for compilation and debugging.

The Source Code Control System (SCCS) is another useful development feature that comes with Xenix 86. SCCS provides a facility for managing software development projects. Under SCCS every version of a program file is saved and differences between each version can be listed. Mechanisms limit update access and mark versions as they are restored. SCCS can reconstruct code from previous versions of a file, if desired. The SCCS program has its own set of commands which make it easy to implement and maintain.

System Communications

The Xenix 86 system has several types of communications utilities at its disposal. This discussion covers only three: MAIL, WRITE, and UUCP. The first utility, MAIL, allows a Xenix 86 user to send a message to one or more users on the system. It also provides the facility to read incoming mail, and subsequently to dispose of messages.

The second utility, WRITE, enables one user on a system to establish direct communication with another user. WRITE allows direct user-to-user dialog to ensue, provided the running program on one of the participants terminals allows WRITE (NROFF does not allow WRITE messages).

The UUCP utility facilitates communication from one Xenix 86 system to another. The communication can be through standard modem connection or a hard-wired direct connection. The UUCP utility can be used for file transfers or remote command execution. The UUCP utility can also perform spooled file transfers between any two machines via a packet-switching device driver.

SUMMARY

The Xenix 86 Operating System is an enhanced, fully-licensed microprocessor implementation of Bell Laboratories' UNIX. Xenix 86 is designed to make your programmers productive. A flexible human interface coupled with the powerful C programming language, sophisticated file structure and system accounting tools make Xenix 86 an excellent base for your terminal-oriented application code. There is also a wealth of independent vendor software ranging from data base management packages to editors and screen menu programs that you can use in your program development and pass through to your customers. Xenix 86 can be thought of as a "software bus" with hundreds of available programs ready to be "plugged into" your system.

CHAPTER 5. SYSTEM MONITOR AND DIAGNOSTICS

The System 86/380 and System 86/330A are equipped with the latest version of the iSBC 957B monitor program. In addition, three diagnostic packages accompany the System 86/380 and System 86/330A Microcomputer Systems. These packages include:

- System Confidence Test (SCT)
- System Diagnostic Test (SDT)
- System Analysis Test (SAT)

These diagnostic packages exercise the System 86/380 and System 86/330A hardware and check for errors. The following sections discuss the monitor and the individual diagnostic packages.

MONITOR

In addition to an operating system, the System 86/380 and System 86/330A Microcomputer Systems contain the iSBC 957B monitor, which resides in a PROM set on the processor board. This monitor allows you to perform basic debugging and memory examination functions without the presence of an operating system. With the monitor, you can perform the following operations:

- Set breakpoints in programs or single-step through code.
- Examine and modify registers and memory.
- Perform I/O via 8086 input and output ports.
- Move and compare blocks of memory.

Also, if you change the jumpers on the processor board and use a parallel-load cable to connect the parallel port of the board to an Intel Microcomputer Development System, you can use the load features of the iSBC 957B package to transfer files between the development system and your System 86/380 and System 86/330A Microcomputer System. See the SYSTEM 86/380 HARDWARE REFERENCE MANUAL or the SYSTEM 86/330A HARDWARE REFERENCE MANUAL for more information on how to change the jumpers on the processor board. Refer to the USER'S GUIDE FOR THE iSBC 957B iAPX 86, 88 INTERFACE AND EXECUTION PACKAGE for information about monitor commands and for information on how to set up your hardware to support the load features.

SYSTEM MONITOR AND DIAGNOSTICS

You can access the monitor in one of two ways:

- By pressing the INTRPT button on the front panel. This method works if you include the System Debug Monitor (discussed in the next section) in your application system or if you connect the INTRPT button to the nonmaskable interrupt of processor board. In both cases, pressing the INTRPT button interrupts the application system and gives control to the monitor, which waits for your entry.
- By entering the iRMX 86 DEBUG command. This loads a program into memory and gives control to the monitor, permitting you to examine the program in detail. The iRMX 86 OPERATOR'S MANUAL describes the DEBUG command in more detail.

SYSTEM CONFIDENCE TEST (SCT)

The SCT provides a basic check of the critical hardware elements of the System 86/380 and System 86/330A. It resides in PROM on the processor board (along with the monitor and the Bootstrap Loader) and runs when you apply power to the system or press the RESET button.

When the SCT starts running, it first performs a memory test and checks to see if it can communicate with the terminal. If your terminal's baud rate is 9600, after approximately five seconds, it displays a series of asterisks at the terminal.

After entering an upper-case "U" at the terminal, another display shows the progress of the remainder of the SCT. The SCT updates this display as it completes each routine in the test. For more detailed information about the SCT, refer to the SYSTEM 86/300 SERIES DIAGNOSTIC MAINTENANCE MANUAL.

Upon successful completion, the SCT passes control to the Bootstrap Loader which loads the Operating System from disk and starts it running.

SYSTEM DIAGNOSTIC TEST (SDT)

The System Diagnostic Test (SDT) provides a means of isolating a hardware problem to a specific sub-assembly. It contains a number of subtests that you can specify to isolate problems to a particular board in the system.

To invoke the SDT, you must interrupt into the monitor and call the Bootstrap Loader. Refer to the SYSTEM 86/300 SERIES DIAGNOSTIC MAINTENANCE MANUAL for detailed information about the SDT.

SYSTEM MONITOR AND DIAGNOSTICS

SYSTEM ANALYSIS TEST (SAT)

The System Analysis Test (SAT) provides a means to exercise the system hardware by running the system software for extended periods of time. This permits isolation of subtle problems to a sub-assembly if elusive, intermittently-reported errors occur. The iRMX 86 Operating System must use the RSAT version of the SAT and the Xenix 86 Operating System must use the XSAT version.

You invoke the SAT by entering a command at the terminal while running under operating system control. The SAT then compiles a test program, links the program with necessary libraries and starts the program running. The test program checks out the various parts of the system and returns error messages, if any, to the system console. Refer to the SYSTEM 86/300 SERIES DIAGNOSTIC MAINTENANCE MANUAL for detailed information about the SAT.

CHAPTER 6. RELATED LITERATURE

This chapter briefly describes most of the related literature available for the System 86/380 or the System 86/330A. This literature includes system manuals, software manuals, a diagnostics manual, a monitor manual, and hardware manuals for the chassis and boards. Most of these manuals can be purchased from:

Literature Department
Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

SYSTEM 86/380 AND SYSTEM 86/330A MANUALS

The following manuals deal specifically with the System 86/380 and the System 86/330A.

- SYSTEM 86/380 HARDWARE REFERENCE MANUAL

This manual describes how to take the System 86/380 Microcomputer System as it is shipped from the factory, unpack it, and set up the hardware. It also gives procedures to allow you to access and service the individual hardware elements.

- SYSTEM 86/330A HARDWARE REFERENCE MANUAL

This manual describes how to take the System 86/330A Microcomputer System as it is shipped from the factory, unpack it, and set up the hardware. It also contains procedures to allow you to access and service the individual hardware elements.

- SYSTEM 86/300 SERIES DIAGNOSTIC MAINTENANCE MANUAL

This manual describes the diagnostic programs available with the 86/300 Series Microcomputer Systems. It discusses how to invoke the diagnostics and how to interpret the error messages that can result from running the diagnostics.

RELATED LITERATURE

HARDWARE MANUALS

The following manuals document the individual hardware elements of the systems.

- iSBC® 016A/032A/064A/028A/056A RAM BOARD HARDWARE REFERENCE MANUAL

This manual provides information concerning the specifications, jumper configurations, and principles of operation of the iSBC 056A RAM memory board.
- iSBC® 215 WINCHESTER DISK CONTROLLER HARDWARE REFERENCE MANUAL

This manual provides information concerning the specifications, interfacing considerations, programming considerations, and principles of operation of the iSBC 215 Winchester Disk Controller board.
- iSBX™ 218 FLEXIBLE DISK CONTROLLER HARDWARE REFERENCE MANUAL

This manual provides information concerning the specifications, jumper configurations, programming considerations, and principles of operation of the iSBX 218 Flexible Disk Controller board.
- iSBC® 86/30 SINGLE BOARD COMPUTER HARDWARE REFERENCE MANUAL

This manual provides information concerning the specifications, jumper configurations, interfacing considerations, programming considerations, and principles of operation of the iSBC 86/30 Single Board Computer.
- iSBC® 337 NUMERIC DATA PROCESSOR MULTIMODULE HARDWARE REFERENCE MANUAL

This manual provides information concerning the specifications, jumper configurations, programming considerations, and principles of operation of the iSBC 337 Numeric Data Processor.
- iSBC® 308/309 MEMORY MANAGEMENT AND PROTECTION MULTIMODULE BOARD HARDWARE REFERENCE MANUAL

This manual describes the iSBC 309 board used on the iSBC 86/30 processor board, for the Xenix Operating System. The manual provides jumpering information, programming information, detailed specifications, and a replacement parts list.

RELATED LITERATURE

- iSXM™ 100 XENIX EXTENSION MODULE INSTALLATION INSTRUCTIONS

The optional iSXM 100 package contains all the necessary equipment to set-up your system with the Xenix Operating System. This manual provides instructions for installing the hardware elements such as extra boards and cables; and instructions for installing and configuring the operating system. The manual also explains how to add more random access memory (RAM) to your system.

- iSXM™ 101 MEMORY MANAGEMENT AND PROTECTION EXTENSION MODULE INSTALLATION GUIDE

This manual explains how to install the iSBC 309 Multimodule board and 8087 Numeric Data Processor into either the System 86/330A or the System 86/380. It also provides Xenix 86 Installation Instructions.

- iSXM™ 952 INSTALLATION GUIDE

This manual provides instructions for installing the iSXM 902 Cable Set. This cable set applies only to the System 86/330A. The manual also describes how to install the iSXM 534 and iSXM 544 Communications Extension Modules.

- iSXM™ 953 INSTALLATION GUIDE

This manual provides instructions for installing the iSXM 903 Cable Set. This cable set applies only to the System 86/380. The manual also describes how to install the iSXM 534 and iSXM 544 Communications Extension Modules.

- iSBC® 534 COMMUNICATIONS EXPANSION BOARD HARDWARE REFERENCE MANUAL

This manual provides information concerning the specifications, jumper configurations, interfacing considerations, programming considerations, and principles of operation of the iSBC 544 Expansion Board.

- USER'S GUIDE FOR THE iSBC® 957B iAPX 86, 88 INTERFACE AND EXECUTION PACKAGE

This manual provides general information, interfacing instructions, and programming information for the iSBC 957B loader and monitor. It provides detailed descriptions of the loader and monitor commands. It also contains configuration information, which may be of little importance to you since the monitor is already configured and available in PROM on your system.

RELATED LITERATURE

iRMX™ 86 MANUALS

The following manuals document the iRMX 86 Operating System.

- GUIDE TO INSTALLING THE iRMX™ 86 OPERATING SYSTEM ON SYSTEM 86/300 SERIES MICROCOMPUTER SYSTEMS

This manual describes how to install the iRMX 86 Operating System onto your system, and how to set up the proper directory structure. It also discusses the features of the preconfigured version of the Operating System. This manual lists the contents of the release diskettes.

- iRMX™ 86 OPERATOR'S MANUAL FOR SYSTEM 86/300 SERIES MICROCOMPUTER SYSTEMS

This manual describes the commands that an operator can enter from a terminal. It documents the line editing and control characters, Human Interface commands, and the Patching Utility.

- INTRODUCTION TO THE iRMX™ 86 OPERATING SYSTEM

This manual is designed to introduce engineers and managers to the iRMX 86 Operating System. It describes how the iRMX 86 Operating System can help you develop your application system in less time and at less expense.

- iRMX™ 86 NUCLEUS REFERENCE MANUAL

This manual documents the Nucleus, the central portion of the iRMX 86 Operating System required by all application systems. It provides overview information, discusses the functions of the Nucleus in detail, and contains detailed descriptions of the system calls available to application programmers.

- iRMX™ 86 TERMINAL HANDLER REFERENCE MANUAL

This manual documents the Terminal Handler, a layer of the iRMX 86 Operating System that supports the use of a terminal as an I/O device for an application system. It describes the effects of entering certain special keyboard characters as well as how to use the Terminal Handler for supplying information to a task or receiving task output at a terminal. The preconfigured version of the iRMX 86 Operating System does not include the Terminal Handler. However, the special characters described in the manual are the same as those available with the System 86/380 Microcomputer System.

RELATED LITERATURE

- iRMX™ 86 DEBUGGER REFERENCE MANUAL

This manual documents the Debugger, a layer of the iRMX 86 Operating System that allows you to interactively examine your application system in order to find and correct errors. It contains introductory and overview material as well as detailed descriptions of all Debugger commands. The preconfigured version of the iRMX 86 Operating System does not include this Debugger. However, this manual is useful if you configure the Debugger into your application system.

- iRMX™ 86 BASIC I/O SYSTEM REFERENCE MANUAL FOR SYSTEM 86/300 SERIES MICROCOMPUTER SYSTEMS

This manual describes the Basic I/O System, a layer of the iRMX 86 Operating System that provides flexible I/O features that are useful in a broad range of applications. It contains some introductory and overview material as well as detailed descriptions of the system calls available to application programmers.

- iRMX™ 86 EXTENDED I/O SYSTEM REFERENCE MANUAL FOR SYSTEM 86/300 SERIES MICROCOMPUTER SYSTEMS

This manual describes the Extended I/O System, a layer of the iRMX 86 Operating System that provides easy-to-use, more-automatic I/O features. It contains some introductory and overview material as well as detailed descriptions of the system calls available to application programmers.

- iRMX™ 86 LOADER REFERENCE MANUAL

This manual describes the two loaders available with the iRMX 86 operating system: the Bootstrap Loader and the Application Loader. It contains some introductory and overview material as well as detailed descriptions of the system calls available with the Application Loader.

- iRMX™ 86 HUMAN INTERFACE REFERENCE MANUAL

This manual documents the Human Interface, a layer of the iRMX 86 Operating System that provides an interactive interface between the user and the application system. It provides introductory and overview information, discusses the process of creating your own Human Interface commands, and contains detailed descriptions of the system calls available to the application programmer.

RELATED LITERATURE

- iRMX™ 86 DISK VERIFICATION UTILITY REFERENCE MANUAL

This manual documents the Disk Verification Utility, a software tool that runs as a Human Interface command, verifying and modifying iRMX 86 named and physical volumes. The manual describes how to invoke the utility, provides a detailed description of all utility commands, and discusses the format of iRMX 86 named volumes. You should read this manual if you suspect that one of your volumes is damaged or contains inconsistent information.

- iRMX™ 86 CRASH ANALYZER REFERENCE MANUAL

This manual documents the Crash Analyzer, a utility that you can use to produce post-mortem memory dumps. It prints a formatted display that describes iRMX 86 objects (memory segments, tasks, etc.) along with the state of each object at the time the system failed.

- iRMX™ 86 PROGRAMMING TECHNIQUES MANUAL

This manual provides a number of programming techniques that can reduce the amount of time you spend designing and implementing your iRMX 86-based application system. It includes discussions on PL/M-86 size controls, interface procedures, timer routines, assembly language programming, job communication, configuration, deadlock, terminal I/O, and stack sizes.

- GUIDE TO WRITING DEVICE DRIVERS FOR THE iRMX™ 86 AND iRMX™ 88 I/O SYSTEMS

This manual provides instructions that allow you to write your own iRMX 86 device drivers. You will need to read this manual if you want to communicate with devices for which the iRMX 86 Operating System does not already supply device drivers.

- iRMX™ 86 CONFIGURATION GUIDE FOR SYSTEM 86/300 SERIES MICROCOMPUTER SYSTEMS

This manual provides the information that allows you to select the iRMX 86 layers that are appropriate for your application system, tailor them to meet your individual needs, and combine them with your own application software to form a functional application system. It describes the Interactive Configuration Utility (ICU), a utility that leads you through the configuration process by displaying a series of "menus" that outline the choices you can make. You do not need to read this manual if you plan to use only the preconfigured version of the Operating System.

RELATED LITERATURE

DEBUG MONITOR MANUAL

- iRMX™ 86 SYSTEM DEBUG MONITOR REFERENCE MANUAL

This manual documents the extension of the iSBC 957B monitor that allows you to examine iRMX 86 objects. This manual contains introductory and overview material as well as detailed descriptions of all debug monitor commands. It also describes configuration, installation, and invocation.

LANGUAGE TRANSLATORS AND UTILITIES MANUALS

The following manuals document the language products that accompany the iRMX 86 Operating System on System 86/380 and System 86/330A Microcomputer Systems.

- EDIT REFERENCE MANUAL

This manual documents EDIT, an iRMX 86-based text editor. It contains introductory and tutorial material as well as detailed descriptions of all EDIT commands.

- ASM86 MACRO ASSEMBLY LANGUAGE REFERENCE MANUAL

This manual documents the 8086/8087/8088 assembly language, ASM86. It describes the assembly language instructions as well as the data types, registers, codemacros, and the macro processing language.

- ASM86 MACRO ASSEMBLER OPERATING INSTRUCTIONS FOR 8086-BASED DEVELOPMENT SYSTEMS

This manual describes how to invoke the assembler. It provides complete descriptions of all the assembler controls. It also describes how to link assembly language programs with PL/M-86 programs.

- PL/M-86 USER'S GUIDE

This manual describes the PL/M-86 language and the use of the PL/M-86 compiler. It provides complete descriptions of all PL/M-86 language statements, discusses compiler invocation, and documents each of the compiler controls.

RELATED LITERATURE

- iAPX 86,88 FAMILY UTILITIES USER'S GUIDE

This manual contains detailed descriptions of the iAPX 86, 88 program development utilities available with the System 86/380 Microcomputer System. It describes LINK86, LOC86, LIB86, CREF86, and OH86.

- RUN-TIME SUPPORT MANUAL FOR iAPX 86, 88 APPLICATIONS

This manual describes the run-time aids that Intel offers for the iAPX 86, 88 family of processors. It discusses the run-time support available for iRMX 86 applications and for applications based on other operating systems. It also provides detailed information about logical record systems and the Universal Development Interface (UDI).

- 8087 SUPPORT LIBRARY REFERENCE MANUAL

This manual describes the package of software tools that enables the 8087 to provide a range of floating-point capabilities to ASM 86 and PL/M-86 programs.

XENIX OPERATING SYSTEM MANUALS AND REFERENCES

The following Microsoft Corporation manuals are currently shipped with the optional Xenix Operating System:

- XENIX SYSTEM - VOLUME 1, PROGRAMMER'S INTRODUCTION

This manual is the programmer's primary source for a broad introduction to Xenix. The manual covers most of the system software, such as the file system, commands, the editor, the shell, and various advanced functions, including the advanced shell, the advanced editor, and SED, a context editor.

- XENIX SYSTEM - VOLUME 2, SOFTWARE DEVELOPMENT

The Software Development manual describes, in several hundred pages, the Xenix tools available for software development. Knowledge of the Xenix system and the C programming language is assumed. Topics covered in this manual include details about the various Xenix compilers (CC, YACC), a program checker (LINT), a maintenance program (MAKE), a debugger (ADB), an assembler (AS), various macros, and an extensive discussion of the C interface to the operating system.

RELATED LITERATURE

- XENIX SYSTEM - VOLUME 3, TEXT PROCESSING

Volume 3 of the Xenix System documentation is a manual which concentrates on the text editors ED and SED. In addition to a broad discussion of the editor commands, this manual describes how to use the editor utilities such as, GREP, AWK, DIFF, and SPELL.

An entire chapter is devoted to editor command explanations. A portion of the manual is printed using the formatting packages, nroff and troff, which provides a direct example of the formatting and output facilities available with the Xenix system.

- XENIX SYSTEM - VOLUME 4, SYSTEM REFERENCE

This manual is written to acquaint users or a system administrator with system level concepts, functions and commands. Topics include user directories, passwords, accounts, system security, file structure integrity, and periodic backup. The manual also provides information for writing device drivers, information on inter-system communication (telecommunications or direct), and suggestions for system administration. An appendix gives a brief outline of the demonstration game programs available on Xenix.

The following publications are recommended for additional Unix/Xenix reading, and are available through most commercial book suppliers. These manuals are not supplied with the System 86/380 or the System 86/330A.

- A USER GUIDE TO THE UNIX SYSTEM by Thomas and Yates. ISBN: 0-931988-71-3; Osborne/McGraw-Hill.
- THE C PROGRAMMING LANGUAGE by Kernighan and Ritchie. ISBN: 0-13-110163-3; Prentice Hall.
- SOFTWARE TOOLS by Kernighan and Plaughter. ISBN: 0-201-03669-X; Addison-Wesley.
- USING THE UNIX SYSTEM by R.L. Gauthier. Prentice Hall.
- A UNIX PRIMER by Lomuto and Lomuto. Prentice Hall.

INTER-BOARD COMMUNICATION SOFTWARE MANUALS

- IMM^x™ 800 MULTIBUS[®] MESSAGE EXCHANGE REFERENCE MANUAL

This manual explains how to set up and use the IMM 800 product. It describes the required protocol for message passing and shows how to implement communication via a simple set of system calls. Numerous examples are provided and an appendix in the manual explains how to use the product with an Ethernet controller board.

APPENDIX A. SYSTEM 86/380 SPECIFICATIONS

Table A-1 lists the System 86/380 specifications. Table A-2 lists the Winchester disk drive specifications. Table A-3 lists the floppy disk drive specifications. Table A-4 lists the power supply specifications for the power supply in each chassis.

Table A-1. System 86/380 Specifications

PROCESSOR BOARD	iSBC 86/30 Single Board Computer. Includes iAPX 86/10 processor (8086-2) operating at 5MHz, iAPX 86/20 Numeric Data Processor (NDP 8087), two iSBX connectors, 24 parallel programmable I/O lines, one serial port, two programmable timers, 9 interrupt levels, 32K bytes of ROM (monitor) and 128K bytes of RAM.
WORD SIZE	
Instruction:	8, 16, or 32 bits
Data:	8/16 bits
INSTRUCTION CYCLE TIME :	400 nanoseconds for fastest executable instructions (assuming instruction is in the queue). 1.0 microsecond for fastest executable instructions (assuming instruction is not in the queue).
MEMORY CAPACITY	
RAM:	The base system is shipped with 384K bytes which can be expanded with other Intel RAM boards to 1 megabyte. The iSBC 86/30 board's on-board address range is 00000 to 1FFFFH. The iSBC 056A board's on-board address range is 20000H through 5FFFFH.
ROM:	32K bytes, on-board the iSBC 86/30 board; address range F8000H through FFFFFH.
INTERRUPTS:	Eight-level, maskable, nested-priority interrupt network and 1 nonmaskable interrupt.

SYSTEM 86/380 SPECIFICATIONS

Table A-1. System 86/380 Specifications (continued)

INTERFACE	EIA Standard RS232C signals provided and supported.
Serial:	9600 baud (asynchronous) or 150 to 19.2k baud (synchronous); programmable baud rates and serial formats.
Parallel:	A parallel I/O port configured to provide a non-standard Centronics printer interface.
MEMORY BOARD	iSBC 056A RAM Expansion Board. Provides 256K bytes of dynamic random access memory expansion on a standard Multibus board. An on-board RAM controller refreshes the active RAMs every 14 microseconds. Each refresh cycle utilizes memory for 480 nanoseconds (maximum). Worst case access time is 570 ns, and 530 ns typical; the board can optionally utilize a parity checking register.
DISK CONTROLLER BOARD	iSBC 215 Winchester Disk Controller Board. Uses 8089 I/O processor for both DMA and user-programmable I/O modes. Controls up to four Winchester type drives or up to 100MB of capacity. Includes on-board diagnostic firmware, Error Checking Code (ECC) provisions, full sector buffering, and 1MB addressing.
DISKETTE CONTROLLER BOARD	iSBX 218 Flexible Disk Drive Controller Multimodule Board. This board attaches directly to the iSBC 215 board, and works in conjunction with it. The iSBX 218 board includes a phase lock loop data separator to ensure maximum data integrity, and is capable of read and write operations on single and multiple sectors. Transfer rate for double density diskettes is 62.5KB per second. Head load and unload times are programmable from 0 to 240ms.
AC REQUIREMENTS (Domestic Systems)	
Processor Chassis:	12.5A @ 90 to 138VAC, 60Hz <u>+5%</u> , single-phase. Maximum total power consumption = 1250 Watts.
Peripheral Chassis:	6.5A @ 90 to 138VAC, 60Hz <u>+5%</u> , single-phase. Maximum total power consumption = 500 Watts.

SYSTEM 86/380 SPECIFICATIONS

Table A-1. System 86/380 Specifications (continued)

AC REQUIREMENTS (European Systems)

Processor Chassis: 6.25A @ 180 to 250VAC, 50Hz +5%, single-phase.
Maximum total power consumption = 1250 Watts.

Peripheral Chassis: 3.25A @ 180 to 250VAC, 50Hz +5%, single-phase.
Maximum total power consumption = 500 Watts.

AC REQUIREMENTS (Japanese Systems)

Processor Chassis: 12.5A @ 90 to 138VAC, 50Hz +5%, single-phase.
Maximum total power consumption = 1250 Watts.

Peripheral Chassis: 6.5A @ 90 to 138VAC, 50Hz +5%, single-phase.
Maximum total power consumption = 500 Watts.

ENVIRONMENTAL REQUIREMENTS

OPERATING

Temperature: 15°C to 35°C
26°C (maximum wet bulb)

Relative Humidity: 20% to 80% non-condensing over the operating
temperature range.

Note: the environmental combination of
humidity and temperature together cannot
exceed 26°C wet bulb.

Altitude: Sea level to 10000 feet.

Vibration & shock: 0.0014" PTP 5 to 25Hz
0.007" PTP 25 to 55Hz
0.3g 0 to peak 55 to 300Hz
1.0g shock for 11ms duration (1/2 sine wave)

NON-OPERATING

Temperature: -25°C to 60°C

Relative Humidity: 20% to 80% non-condensing.

Altitude: Sea level to 12000 feet.

Vibration & shock: 0.008" PTP 5 to 25Hz
0.004" PTP 25 to 55Hz
2.0g 0 to peak 55 to 300Hz
15.0g shock for 11ms duration (1/2 sine wave)

SYSTEM 86/380 SPECIFICATIONS

Table A-1. System 86/380 Specifications (continued)

PHYSICAL CHARACTERISTICS	
PROCESSOR CHASSIS	
Width:	16.8 in. (42.6 cm)
Height:	12.2 in. (31.1 cm)
Depth:	21.0 in. (53.3 cm)
Weight:	55 pounds (25 kg)
PERIPHERAL CHASSIS	
Width:	16.8 in. (42.6 cm)
Height:	12.2 in. (31.1 cm)
Depth:	21.0 in. (53.3 cm)
Weight:	70 pounds (32 kg)

Table A-2. Winchester Drive Specifications

Recording Density:	PRIAM 3450 Drive
Track Density:	6670 bpi
Cylinders:	480 tpi
R/W heads:	526
Bytes/sector:	5
Sectors/track:	1024
Bytes/track:	12
Rotational speed:	13.4K
Avg. latency:	3600 RPM
Transfer rate:	8.34 ms
Head settling time:	6.44M bits/sec.
	8 ms
ACCESS TIMES:	
Track-to-track:	10 ms
Average:	45 ms
Maximum:	90 ms
Motor/on time:	30 sec
Soft error rate:	1×10^{10}
Hard error rate:	1×10^{13}
Seek error rate:	1×10^6

SYSTEM 86/380 SPECIFICATIONS

Table A-3. Flexible Disk Drive Specifications

Data Storage Capacity:	800K bytes (unformatted) per side, or 500K bytes (IBM format - 26 sectors) per side.
Recording Density:	6816 bpi
Track Density:	48 tpi
Number of tracks:	77 per side
Number of heads:	2
Recording method:	MFM
Transfer rate:	500K bits/sec
Rotational speed:	360 RPM
Rotational latency:	83 ms (average)
Access time:	
Track-to-track:	3 ms
Average:	91 ms
Head settling time:	15 ms
Head loading time:	35 ms
Motor Start time:	2 sec.
ERROR RATES:	
Soft:	1 in 10 ⁹ bits read
Hard:	1 in 10 ¹² bits read
Seek:	1 in 10 ⁶ seeks

Table A-4. Power Supply Specifications

Component	Current Requirement					
	+5V	+12V	-5V	-12V	+24V	-24V
PROCESSOR CHASSIS						
iSBC 86/30 board	6.10A	0.31A	-	-	-	-
iSBC 215 board	3.25A	-	0.15A	-	-	-
iSBX 218 board	0.90A	-	-	-	-	-
iSBC 056A board	4.60A	-	-	-	-	-
Total	14.85A	0.31A	0.15A	-	-	-
Available for options	55A	5.7A	2.8A	5.0A	-	-
Regulation:	< 0.4%					
Ripple +5V:	50mV p-p					
+12V:	120mV p-p					

SYSTEM 86/380 SPECIFICATIONS

Table A-4. Power Supply Specifications (continued)

Component	Current Requirement					
	+5V	+12V	-5V	-12V	+24V	-24V
PERIPHERAL CHASSIS						
Floppy drive	1.0A	-	0.05A	-	0.6A	-
Winchester drive	2.5A	-	1.5A	0.4A	3.5A	-
Total	3.5A	-	1.55A	0.4A	4.1A	-
Available for options	26.5A	2.9A	2.45A	2.6A	3.7A	1.5A
Regulation:	+4%					
Ripple +5V:	50mV p-p					
+12V:	50mV p-p					
+24V:	240mV p-p					
-24V:	360mV p-p					

APPENDIX B. SYSTEM 86/330A SPECIFICATIONS

Table B-1 lists the System 86/330A specifications. Table B-2 lists the Winchester drive specifications. Table B-3 lists the flexible disk drive specifications. Table B-4 lists the power supply specifications.

Table B-1. System 86/330A Specifications

PROCESSOR BOARD	iSBC 86/30 Single Board Computer. Includes iAPX 86/10 processor (8086-2) operating at 5MHz, iAPX 86/20 Numeric Data Processor (NDP 8087), two iSBX connectors, 24 parallel programmable I/O lines, one serial port, two programmable timers, 9 interrupt levels, 32K bytes of ROM (monitor) and 128K bytes of RAM.
WORD SIZE	
Instruction:	8, 16, or 32 bits
Data:	8/16 bits
INSTRUCTION CYCLE TIME :	400 nanoseconds for fastest executable instructions (assuming instruction is in the queue). 1.0 microsecond for fastest executable instructions (assuming instruction is not in the queue).
MEMORY CAPACITY	
RAM:	The base system is shipped with 384K bytes which can be expanded with other Intel RAM boards to 1 megabyte. The iSBC 86/30 board's on-board address range is 00000 to 1FFFFH. The iSBC 056A board's on-board address range is 20000H through 5FFFFH.
ROM:	32K bytes, on-board the iSBC 86/30 board; address range F8000H through FFFFFH.
INTERRUPTS:	Eight-level, maskable, nested-priority interrupt network and 1 nonmaskable interrupt.

SYSTEM 86/330A SPECIFICATIONS

Table B-1. System 86/330A Specifications (continued)

INTERFACE	EIA Standard RS232C signals provided and supported.
Serial:	9600 baud (asynchronous) or 150 to 19.2k baud (synchronous); programmable baud rates and serial formats.
Parallel:	A parallel I/O port configured to provide a non-standard Centronics printer interface.
WORD SIZE	
Instruction:	8, 16, or 32 bits
Data	8/16 bits
INSTRUCTION CYCLE TIME:	
8 MHz:	750 nanoseconds. 250 nanoseconds (assumes instruction is in the queue).
5 MHz	1.2 microseconds. 400 nanoseconds (assuming instruction is in the queue).
MEMORY CAPACITY:	
RAM	The base system is shipped with 384K bytes which can be expanded with other Intel RAM memory boards up to 1 Mbyte. On board address range 00000H to 1FFFFH. Off board address range 20000H through 5FFFFH.
ROM	32K bytes. Address range F8000H through FFFFFH.
INTERRUPTS:	Eight-level, maskable, nested-priority interrupt network and 1 nonmaskable interrupt.
INTERFACE	EIA Standard RS232C signals provided and supported.
Serial:	A terminal at 9600 Baud (asynchronous); programmable baud rates and serial formats.
Parallel:	A parallel I/O port configured to interface a printer with a Centronics signal interface.

SYSTEM 86/330A SPECIFICATIONS

Table B-1. System 86/330A Specifications (continued)

<p>MEMORY BOARD</p>	<p>iSBC 056A RAM Expansion Board. Provides 256K bytes of dynamic random access memory expansion on a standard Multibus board. An on-board RAM controller refreshes the active RAMs every 14 microseconds. Each refresh cycle utilizes memory for 480 nanoseconds (maximum). Worst case access time is 570 ns, and 530 ns typical; the board can optionally utilize a parity checking register.</p>
<p>DISK CONTROLLER BOARD</p>	<p>iSBC 215 Winchester Disk Controller Board. Uses 8089 I/O processor for both DMA and user-programmable I/O modes. Controls up to four Winchester type drives or up to 100MB of capacity. Includes on-board diagnostic firmware, Error Checking Code (ECC) provisions, full sector buffering, and 1MB addressing.</p>
<p>DISKETTE CONTROLLER BOARD</p>	<p>iSBX 218 Flexible Disk Drive Controller Multimodule Board. This board attaches directly to the iSBC 215 board, and works in conjunction with it. The iSBX 218 board includes a phase lock loop data separator to ensure maximum data integrity, and is capable of read and write operations on single and multiple sectors. Transfer rate for double density diskettes is 62.5KB per second. Head load and unload times are programmable from 0 to 240ms.</p>
<p>AC REQUIREMENTS:</p>	<p><u>Domestic</u>: 6.5A @ 90 to 126 VAC, 60 Hz +5%, single-phase; <u>European</u>: 3.25A @ 180 to 250 VAC, 50 Hz +5%, single-phase; <u>Japanese</u>: 6.5A @ 90 to 126 VAC, 50 Hz +5%, single-phase. Maximum total power consumption 715W.</p>

SYSTEM 86/330A SPECIFICATIONS

Table B-1. System 86/330A Specifications (continued)

ENVIRONMENTAL REQUIREMENTS:	
Operating:	
Temperature:	15° C to 35° C 26° C Maximum wet bulb
Relative Humidity:	20% to 80% non-condensing over the operating temperature range.*
Altitude:	Sea level to 10,000 feet.
Vibration & shock:	0.0014" peak-to-peak 5 to 25 Hz 0.007" peak-to-peak 25 to 55 Hz 0.3g 0 to peak 55 to 300 Hz 1.0g shock for 10 to 11 ms duration
Non-Operating:	
Temperature:	-25° C to 60° C
Relative Humidity:	20% to 80% non-condensing.
Altitude:	Sea level to 12000 feet.
Vibration & shock:	0.008" peak-to-peak 5 to 25 Hz 0.004" peak-to-peak 25 to 55 Hz 2.0g 0 to peak 55 to 300 Hz 15.0g shock for 10 to 11 ms duration
Shipping:	15.0g shock for 10 to 11 ms duration
PHYSICAL CHARACTERISTICS:	
Width:	16.75 in. (42.55 cm)
Height:	12.25 in. (31.12 cm)
Depth:	21.00 in. (53.34 cm)
Weight:	75 pounds (34.02 kg)
*Note: The environmental combination of humidity and temperature together cannot exceed 26° C wet bulb.	

Table B-2. Winchester Drive Specifications

Recording Density:	6670 bpi
Track Density:	480 tpi
Cylinders:	526
R/W heads:	5
Bytes/sector:	1024
Sectors/track:	12
Bytes/track:	13.4K
Rotational speed:	3600 RPM
Avg. latency:	8.34 ms

SYSTEM 86/330A SPECIFICATIONS

Table B-2. Winchester Drive Specifications (continued)

Transfer rate:	6.44M bits/sec.
Head settling time:	8 ms
ACCESS TIMES:	
Track-to-track:	10 ms
Avg.:	45 ms
Max.:	90 ms
Motor/on time:	30 sec
Soft error rate:	1×10^{10} bits read
Hard error rate:	1×10^{13} bits read
Seek error rate:	1×10^6 seeks

Table B-3. Flexible Disk Drive Specifications

Capacity per side:	800K bytes (unformatted) 500K bytes (IBM format - 26 sectors)
Recording Density:	6816 bpi
Track Density:	48 tpi
Number of tracks:	77/side
Number of heads:	2
Recording method:	FM or MFM
Transfer rate:	500K bits/sec
Rotational speed:	360 RPM
Rotational latency:	83ms (AVG)
Access time:	
Track-to-track:	3 ms
AVG:	91 ms
Head settling:	15 ms
Head load:	35 ms
Motor start:	2 sec.
ERROR RATES:	
Soft:	1 in 10^9 bits read
Hard:	1 in 10^{12} bits read
Seek:	1 in 10^6 seeks

All diskettes are formatted such that cylinder 0, head 0 (side 0, track 0) is recorded with 128-byte sectors using the single-density recording method (FM). The rest of the diskette can be formatted using either the single-density (FM) or the double-density (MFM) recording method with 256-, 512-, or 1024-byte sector size. The single-density recording method supports 128-byte sectors, but the double-density recording method does not. All double-sided, double-density diskettes are formatted such that cylinder 0, head 1 (side 1, track 0) is recorded with 256-byte sectors.

SYSTEM 86/330A SPECIFICATIONS

Table B-4. Power Supply Specifications

Component	+5V	+12V	-5V	-12V	+24V	-24V
Processor	6.10A	0.31A				
Win.(215)	3.25A		0.15A			
Flex.(218)	0.9A					
RAM (056)	4.6A					
Flex. drive	1.0A		0.05A		0.6A	
Win. drive	2.5A		1.5A	0.4A	3.5A	
Total	18.33A	0.31A	1.7A	0.4A	4.1A	
Available for options	12.2A	1.5A	0.3A	1.5A	3.7A	1.6A
Regulation: $\pm 4\%$ Ripple $\underline{+5V}$: 50mV peak-to-peak $\underline{+12V}$: 50mV peak-to-peak $\underline{+24V}$: 100mV peak-to-peak						

INDEX

Boards 2-7

C Compiler 4-7
Cardcage 2-2, 2-6
Chassis 2-1, 2-4, 2-5
Communications 4-9
Communications Controller Board 2-8
Configurability 3-7
Connectors 2-3, 2-6
Control Panel 2-1, 2-5

Development Tools 4-9
Device-Independent I/O Management 3-5
Disk Controller Boards 2-8, A-2, B-2

Editors 3-12, 4-8
Error Processing 3-5

Flexible Disk Drive 2-9

Hardware Description 2-1 thru 2-9, A-1, B-1

iRMX 86 3-1 thru 3-14

Kernal 4-6

Language 3-13, 4-7

Manuals 6-1 thru 6-9
Memory Board 2-8, A-2, B-2
Memory Management And Protection Board 2-8
Monitor 5-1

Peripheral Chassis 2-4
Preconfigured Operating System 3-11
Processor Board 2-7, A-1, B-1
Processor Chassis 2-1
Processor Management 3-2

Service 1-5
Shell 4-6
System Advantages 1-3
System Analysis Test (SAT) 5-3
System Confidence Test (SCT) 5-2
System Diagnostic Test (SDT) 5-2

INDEX (continued)

Tape Controller Board 2-8

Tape Drive 2-9

Terminal Support 3-9

Text Processing 4-8

Translators 3-13

UDI 3-10

Utilities 3-13, 4-7

Winchester Disk Drive 2-9

Xenix 86 1-4, 4-1 thru 4-9



REQUEST FOR READER'S COMMENTS

Intel Corporation attempts to provide documents that meet the needs of all Intel product users. This form lets you participate directly in the documentation process.

Please restrict your comments to the usability, accuracy, readability, organization, and completeness of this document.

1. Please specify by page any errors you found in this manual.

2. Does the document cover the information you expected or required? Please make suggestions for improvement.

3. Is this the right type of document for your needs? Is it at the right level? What other types of documents are needed?

4. Did you have any difficulty understanding descriptions or wording? Where?

5. Please rate this document on a scale of 1 to 10 with 10 being the best rating. _____

NAME _____ DATE _____

TITLE _____

COMPANY NAME/DEPARTMENT _____

ADDRESS _____

CITY _____ STATE _____ ZIP CODE _____

Please check here if you require a written reply.

WE'D LIKE YOUR COMMENTS . . .

This document is one of a series describing Intel products. Your comments on the back of this form will help us produce better manuals. Each reply will be carefully reviewed by the responsible person. All comments and suggestions become the property of Intel Corporation.



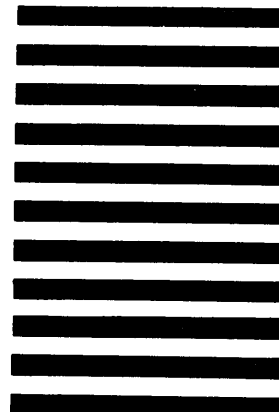
NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO.79 BEAVERTON, OR

POSTAGE WILL BE PAID BY ADDRESSEE

Intel Corporation
5200 N.E. Elam Young Pkwy.
Hillsboro, Oregon 97123

OMO Technical Publications





INTEL CORPORATION, 3065 Bowers Avenue, Santa Clara, California 95051 (408) 987-8080

Printed in U.S.A.