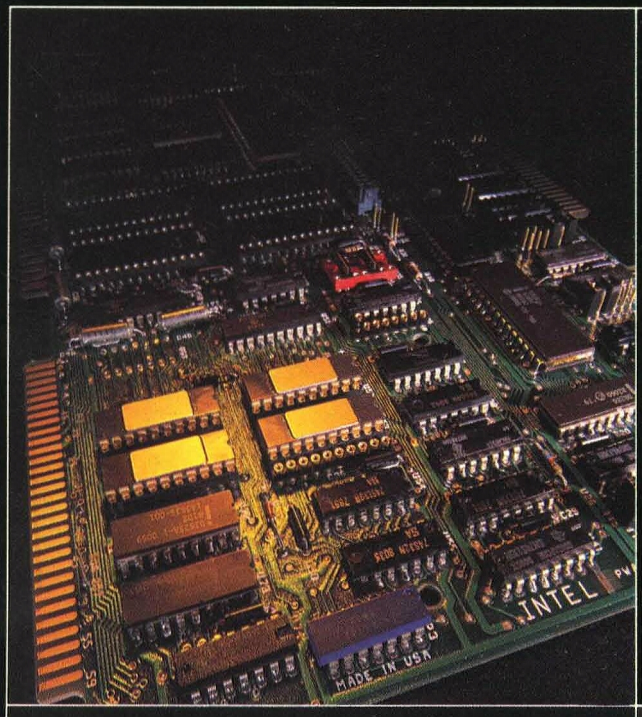
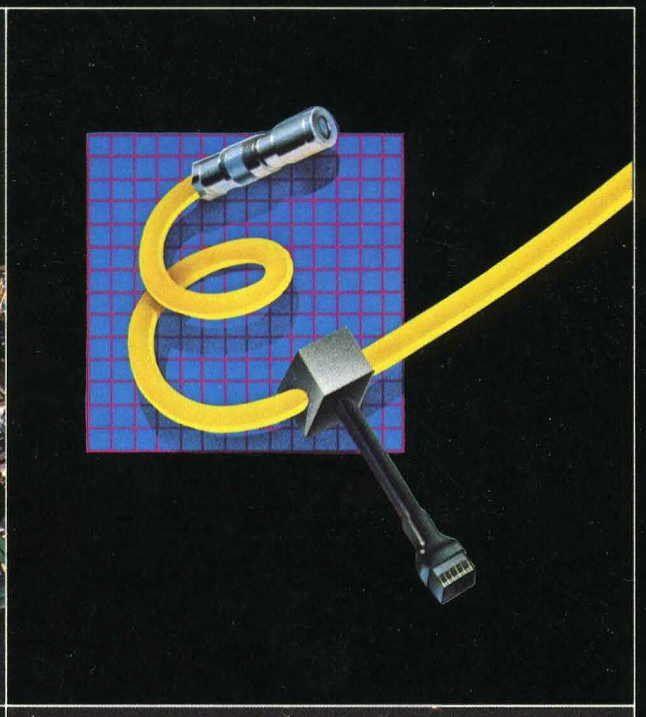
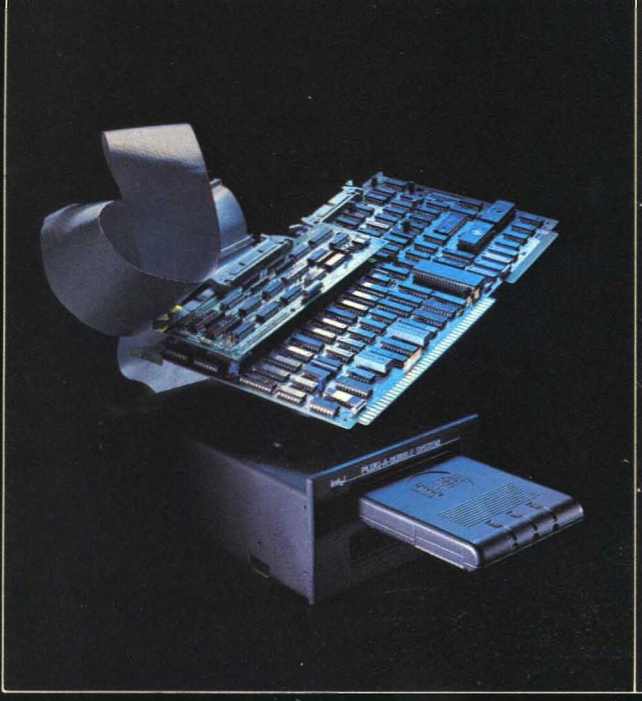
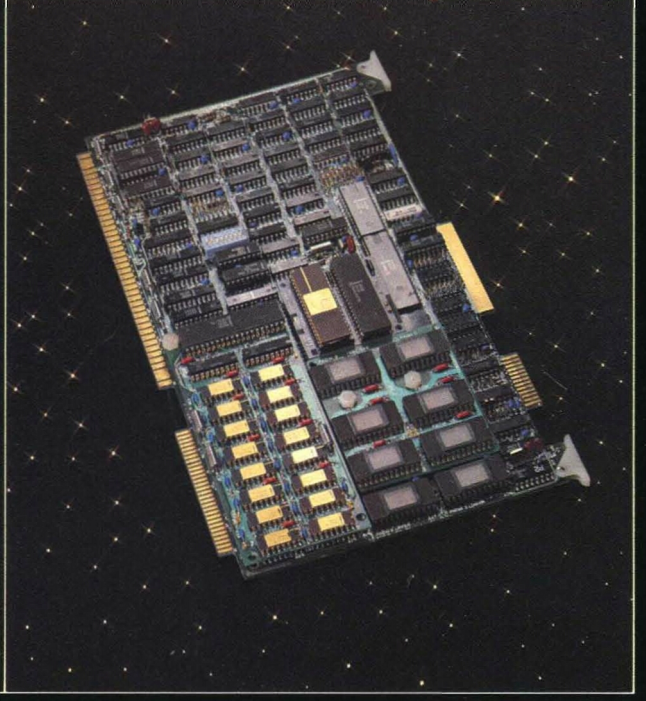
intel

Systems Data Catalog









Order Number: 210299-001

intel

SYSTEMS DATA CATALOG

JANUARY 1982

Intel Corporation makes no warranty for the use of its products and assumes no responsibility for any errors which may appear in this document nor does it make a commitment to update the information contained herein.

The following are trademarks of Intel Corporation and may only be used to identify Intel Products:

BXP, CREDIT, i, ICE, iCS, i_m, iMMX, Insite, Intel, int_el, Intelevision, Intellec, iOSP, iRMX, iSBC, iSBX, Library Manager, MCS, Megachassis, Micromainframe, Micromap, Multimodule, Plug-A-Bubble, PROMPT, RMX/80, System 2000 and UPI.

MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

*MULTIBUS is a patented Intel bus.

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

Intel Corporation
Literature Department SV3-3
3065 Bowers Avenue
Santa Clara, CA 95051

© INTEL CORPORATION, 1981

Table of Contents

CHAPTER 1
Integrated Microcomputer Systems

SYSTEM 86/330 Microcomputer System 1-1

CHAPTER 2
Single Board Computers

iSBC 80/04 Single Board Computer 2-1
iSBC 80/05 (or pSBC 80/05) Single Board Computer 2-7
iSBC 80/10B (or pSBC 80/10B) Single Board Computer 2-12
iSBC 80/20-4 (or pSBC 80/20-4) Single Board Computer 2-19
iSBC 80/24 (or pSBC 80/24) Single Board Computer 2-26
iSBC 80/30 (or pSBC 80/30) Single Board Computer 2-35
iSBC 86/05 Single Board Computer 2-43
iSBC 86/12A (or pSBC 86/12A) Single Board Computer 2-51
iSBC 86/14 and iSBC 86/30 Single Board Computers 2-59
iSBC 88/25 Single Board Computer 2-67
iSBC 88/40 Measurement and Control Computer 2-75
iSBC 310A High Speed Mathematics Unit 2-83

CHAPTER 3
iSBX™ MULTIMODULE™ Boards

iSBX 331 Fixed/Floating Point Math MULTIMODULE Board 3-1
iSBX 332 Floating Point Math MULTIMODULE Board 3-7
iSBX 350 Parallel I/O MULTIMODULE Board 3-12
iSBX 351 Serial I/O MULTIMODULE Board 3-16

CHAPTER 4
iCS™ Industrial Control Series

iCS 80 Industrial Chassis 4-1
iCS 910/920/930 Signal Conditioning/Termination Panels 4-6
iSBC 941 Industrial Digital Processor 4-14

CHAPTER 5
Run-Time Systems Software

iRMX 80 Real-Time Multi-Tasking Executive 5-1
iMMX 800 MULTIBUS Message Exchange Software 5-6
iRMX 86 Operating System 5-10
iRMX 86 Languages 5-16
iRMX 88 Real-Time Multi-Tasking Executive 5-21
iSBC 957B iAPX 86, 88 Interface and Execution Package 5-27
iOSP 86 iAPX 86/30 and iAPX 88/30 Support Package 5-32

CHAPTER 6
Peripheral Controllers

iSBC 204 Single Density Flexible Diskette Controller 6-1
iSBC 208 Flexible Disk Controller 6-4
iSBC 215A/iSBC 215B/215C Winchester Disk Controller 6-8
iSBX 218 Flexible Disk Controller 6-13
iSBC 220 SMD Disk Controller 6-17
iSBX 270 Video Display Controller 6-21

CHAPTER 7
Memory Expansion Boards

iSBC 012B RAM Memory Board 7-1
iSBC 016A/032A/064A/028A/056A RAM Memory Boards 7-3
iSBC 064 RAM Memory Board 7-6
iSBC 090 Memory System 7-8

iSBC 094 4K-Byte CMOS RAM Memory Battery Backup Board 7-12
iSBC 108A/116A Combination Memory and I/O Expansion Boards 7-15
iSBC 300 (or pSBC 300) 32K-Byte RAM Expansion Module, iSBC 340 (or pSBC 340) 16K-Byte EPROM Expansion Module 7-20
iSBC 301 4K-Byte RAM MULTIMODULE Board 7-24
iSBC 302 8K-Byte MULTIMODULE RAM 7-27
iSBC 303 MULTIMODULE Parity 7-29
iSBC 341 28-Pin MULTIMODULE EPROM 7-33
iSBC 416 16K EPROM Expansion Board 7-35
iSBC 464 64K-Byte EPROM Expansion Board 7-37

CHAPTER 8
Digital I/O Expansion and Signal Conditioning Boards

iSBC 337 MULTIMODULE Numeric Data Processor 8-1
iSBX 488 GPIB MULTIMODULE Board 8-9
iSBC 517 Combination I/O Expansion Board 8-13
iSBC 519 (or pSBC 519) Programmable I/O Expansion Board 8-17
iSBC 556 Optically Isolated I/O Board 8-21
iSBC 569 Intelligent Digital Controller 8-23

CHAPTER 9
Communication Controllers

iSBC 534 Four Channel Communications Expansion Board 9-1
iSBC 544 Intelligent Communications Controller 9-5
iSBC 550 Ethernet* Communications Controller 9-12
iSBX 352 Bit Serial Communications MULTIMODULE Board 9-16

CHAPTER 10
Analog I/O Expansion and Signal Conditioning Boards

iSBX 311 Analog Input MULTIMODULE Board 10-1
iSBX 328 Analog Output MULTIMODULE Expansion Board 10-5

CHAPTER 11
System Packaging and Power Supplies and Service

iSBC 604/614 Modular Cardcage/Backplane 11-1
iSBC 660 System Chassis 11-3
iSBC 680/iSBC 681 Multistore User System Package 11-6
iSBC 635 Power Supply 11-10
iSBC 640 Power Supply 11-13
iSBC 665/iSBC 645 System Chassis and Power Supply 11-15
iMBX 100/110/120 MULTIBUS Exchange Hardware Subscription Service 11-19

CHAPTER 12
Microcomputer Development Systems

Model 225 Intellec Series II/85 Microcomputer Development System 12-1
Model 286 Intellec Series III Microcomputer Development System 12-6
Model 675 Development System for Ethernet* 12-13

CHAPTER 13
Microcomputer Development Systems Options

MDS 201 Expansion Chassis Intellec Series II Microcomputer Development System 13-1
MDS 384 Mainframe Link for Distributed Development 13-3
Model 503 Double Density Upgrade Kit for Intellec Microcomputer Development System 13-6
Model 556 iAPX 86 Resident Processor Board Package 13-8
Model 677 DS/E Upgrade (Ethernet)* 13-11
ISIS II Software Tool Box 13-15
Credit CRT Based Text Editor Microcomputer Development System 13-17

CHAPTER 14
Flexible Disk Systems

MDS 720 Intellec Single/Double Density Flexible Disk System 14-1

CHAPTER 15
MCS-80/85™ Development Systems and Options

FORTRAN 80 8080/8085 ANS FORTRAN 77 Intellec Resident Compiler 15-1
Basic-80 Extended ANS 1978 Basic Intellec Resident Interpreter 15-5
PASCAL 80 Software Package 15-8
8080/8085 Fundamental Support Package (FSP) 15-13
iCIS COBOL Software Package 15-17
PL/M 80 High Level Programming Language Intellec Resident Compiler 15-21
ICE-80 8080 In-Circuit Emulator 15-24
ICE-85B In-Circuit Emulator 15-30

CHAPTER 16
iAPX 86/88 Support Options

iAPX 86, 88 Software Development Package for Series II 16-1
iAPX 286 Evaluation Package 16-11
PL/M 86/88 Software Package 16-13
PASCAL 86/88 Software Package 16-18
FORTRAN 86/88 Software Package 16-21
8087 Software Support Package 16-25
8087 Support Library 16-28
8089 IOP Software Support Package 16-32
ICE-86A iAPX 86 In-Circuit Emulator 16-35
ICE-88 8088 In-Circuit Emulator 16-43

CHAPTER 17
Prototype Microcomputer Kits

SDK-86 System Design Kit 17-1
SDK-C86 17-3
SDK-85 System Design Kit 17-9
SDK-51 System Design Kit 17-15
SDK-2920 System Design Kit 17-20

CHAPTER 18
MCS-48™ Development Systems

Intellec PROMPT 48 18-1
HSE-49 High-Speed Emulator 18-7
ICE-49, MCS-48 In-Circuit Emulator 18-13
EM-1 8021 Emulation Board 18-18
EM-2 8022 Emulation Board 18-21
ICE-22 8022 In-Circuit Emulator 18-24
MCS-48 Diskette-Based Software Support Package 18-30

CHAPTER 19
MCS-51™ Development Systems

8051 Software Development Package 19-1
EM-51 8051 Emulation Board 19-4
ICE-51 8051 In-Circuit Emulator 19-7

CHAPTER 20
UPI-41A™ Development Systems

ICE-41A, UPI-41A In-Circuit Emulator 20-1

CHAPTER 21
2920 Signal Processor Development Systems

2920 Signal Processing Applications Compliler 21-1
2920 Software Support Package 21-6

CHAPTER 22
Memory Systems

Series 90 General Purpose Memory System 22-1
Series 90 iQX Intelligent Memory System 22-4
CM 5044E UNIBUS* Add-in Memory 22-8
in-1671 PDP* 11/70 Add-on Memory System 22-10
in-5034 PDP* 11/04, 11/34 Add-in Memory Board 22-15
in-5160 Nova* 3 Add-In Memory 22-17
in-5770 Video Refresh Memory System 22-22
MU-5750 VAX* 11/750, PDP* 11/70 Add-in Memory Card 22-27
MU-5780 VAX*-11/780 Add-In Memory Card 22-29
iSBC 254 Bubble Memory Board 22-32
iPAB Plug-A-Bubble Memory System 22-36
iSBX 251 Magnetic Bubble Multimodule Board 22-42

CHAPTER 23

Insite™ User's Program Library 23-1

CHAPTER 24
Software Distribution Operation

Digital Research Inc. CP/M 2.2 Operating System 24-1
Digital Research Inc. CP/M-86 Operating System 24-4
Microsoft MACRO-80 Utility Software Package 24-7
Microsoft BASIC 80 Interpreter Software Package 24-9
Microsoft COBOL 80 Software Package 24-11
Microsoft FORTRAN 80 Software Package 24-15
Microsoft BASIC 86 Interpreter Software Package 24-17
Jovial 86 Cross-Compliler 24-19

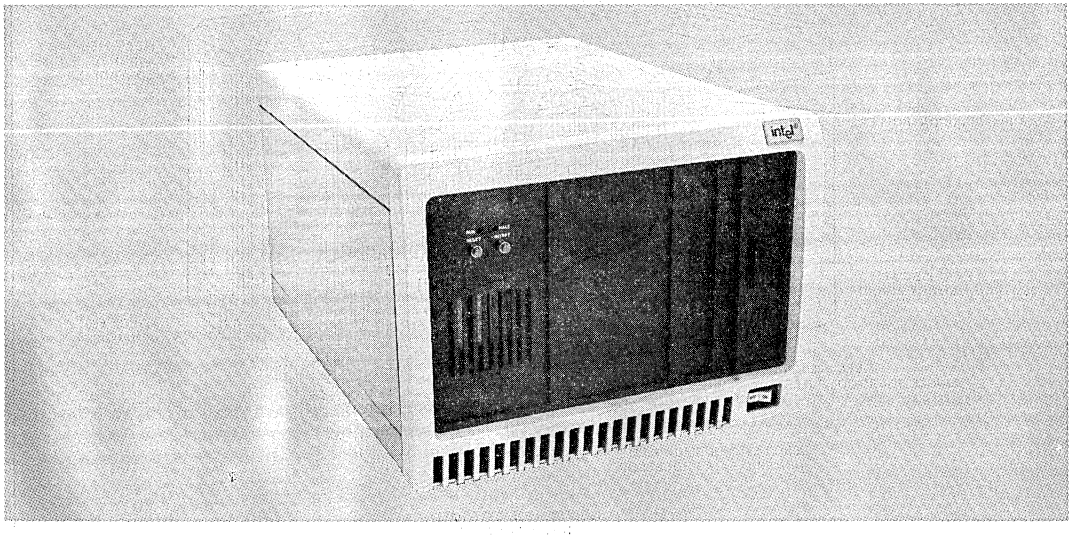
*Ethernet is a registered trademark of Xerox Corp.
*VAX, PDP and UNIBUS are registered trademarks of Digital Equipment Corp.
NOVA is a registered trademark of Data General Corp.

1 Integrated Microcomputer Systems

SYSTEM 86/330
MICROCOMPUTER SYSTEM

- **Compact desk-top or rack-mount integrated microsystem**
- **High performance 16-bit iAPX 86/20 processor set (iSBC™ 86/12A + iSBC™ 337 boards)**
- **Full function iRMX™ 86 real-time, multitasking operating system**
- **Intel® resident languages include PL/M-86 and ASSEMBLER-86. Intel® PASCAL-86, FORTRAN-86, plus independent software vendor languages (BASIC, COBOL, C), also available**
- **MULTIBUS® system bus (IEEE-796) multiprocessor architecture**
- **35MB Winchester and 1MB DD/DS 8″ floppy for program, data storage and back-up**
- **320KB of high speed RAM memory to execute multiple jobs and tasks**
- **Extensive self-test routines for reliable operation and simple fault isolation**
- **Modular, standard products allow flexible decomposition of the system to board level products for custom configurations**

The Intel SYSTEM 86/330 Microcomputer System is a comprehensive integrated, 16-bit hardware and software package designed to give the OEM the fastest path to high performance VLSI. The system, which is based on standard Intel MULTIBUS system bus board level products, also incorporates the VLSI operating system, iRMX 86, state-of-the-art high capacity mass storage and high density RAM memory boards. In addition to the 8086 general purpose microprocessor, the system provides 3 to 5 times the numeric performance of low-end minicomputers through the use of the 8087 numeric data processor. The system's capabilities can be greatly expanded through the use of additional general purpose processors and intelligent I/O boards.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1981

October, 1981
Order Number: 210275-001

SYSTEM TECHNICAL DESCRIPTION

Hardware

PROCESSOR SECTION

The single board computer used in the SYSTEM 86/330 is the MULTIBUS-based iSBC 86/12A board. The central processor of the iSBC 86/12A board is the powerful 5 MHz 16-bit 8086. The architecture includes four 16-bit byte addressable registers, two 16-bit index registers, all accessed by a total of 24 operand addressing modes for complex data handling and flexible memory addressing.

The base SYSTEM 86/330 is supplied with 320KB of high speed RAM memory. There is 64KB of dual ported (iSBC 300 memory expansion board supplied) RAM accessible to both the local and MULTIBUS system bus on the iSBC 86/12A processor board. The system can address up to 1 megabyte of main memory. The processor board includes an RS232C serial channel for the connection of a user supplied terminal. Cabling to the rear panel of the system is provided. Asynchronous baud rates of 110 to 19.2K are supported by the system. A parallel I/O interface is also available on the processor board. The port has been configured to be interfaced to a Centronics compatible line printer.

NUMERIC PROCESSING EXTENSION

Also included with the base SYSTEM 86/330 is the iSBC 337 Numeric Data Processor. This MULTIMODULE board contains the Intel 8087 numeric co-processor. The co-processor interface between the 8087 and the 8086 host CPU provides a simple means of extending the instruction set of the system with over 60 additional numeric instructions supporting six additional data types. The data formats used in the SYSTEM 86/330 conform to the proposed IEEE floating point standard insuring highly accurate results. Dramatic numeric performance improvements are provided by the 8087 co-processor. For example a 50X improvement in the Whetstone benchmark over the standard 8086 processor is afforded by the 8086 numeric co-processor.

The 8087 arithmetic, logarithmic, transcendental and trigonometric instructions are fully supported by ASM-86, as well as other Intel higher level languages such as PL/M-86, PASCAL-86 and FORTRAN-86.

MEMORY EXPANSION

In addition to the 64KB of dual ported RAM on the iSBC 86/12A processor board, the SYSTEM 86/330 also includes a 256KB memory expansion board.

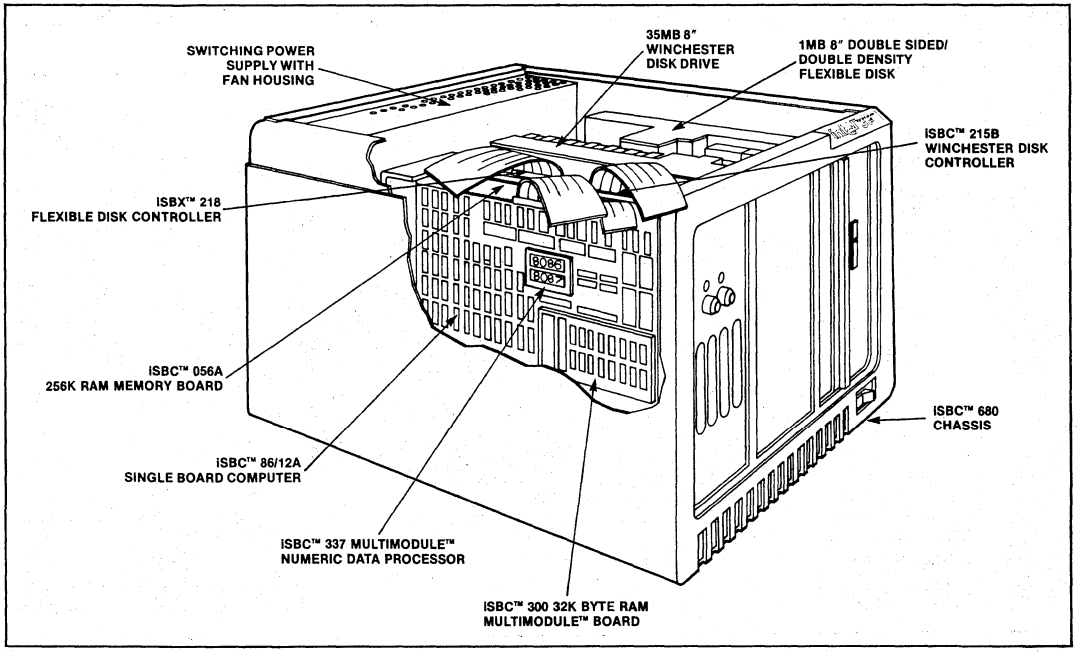


Figure 1. SYSTEM 86/330 Microcomputer System

This brings the system RAM memory supplied with the base SYSTEM 86/330 to 320KB. The 256KB memory board uses the latest technology 64K bit dynamic RAMs. Through the use of optional memory expansion boards, the total memory capacity of the SYSTEM 86/330 may be expanded to 1 megabyte.

MASS STORAGE

Intelligent Controller — The SYSTEM 86/330 uses the intelligent, 8089-based iSBC 215B Winchester controller. This high performance interface contains firmware which is executed directly on the 8089 processor. This intelligent I/O processor coupled with an on-board RAM buffer off loads a significant portion of disk I/O overhead from the host 8086 processor. In addition to the Winchester control firmware, there is also 8089 firmware resident on the iSBC 215B board to control the iSBX 218 floppy disk controller. The floppy controller plugs directly onto the iSBC 215B board via one of the two available iSBX connectors.

Mass storage expansion beyond the single Winchester and floppy diskette drives configured with the system is easily accomplished by the OEM through the use of daisy chain connector on the back panel of the SYSTEM 86/330 chassis. In addition to the Winchester and Floppy interfaces provided with the system, Intel also can provide a board level interface and iRMX 86 software for SMD compatible disk drives.

Winchester Disk Drive — The SYSTEM 86/330 contains a high performance 35MB (32MB formatted) 8″ Winchester technology hard disk drive for program and data storage. The drive has an average access time of 43 ms and a transfer rate of 6.44 Mbits/sec.

Floppy Diskette Drive — A double density/double sided, 8″, 1MB, floppy disk drive is included in the base system. This high density floppy drive, which has an average access time and a transfer rate of 250Kbits/sec, can be used for both data storage and system backup.

SYSTEM CHASSIS

The chassis used in the SYSTEM 86/330 is the compact iSBC 680 Multistore User System Package. This 21.0″ × 16.75″ × 12.25″ package houses up to six MULTIBUS cards (two available for expansion in the SYSTEM 86/330). The 8″ Winchester disk drive and 8″ floppy disk drive are housed in the iSBC 680 package with simple slide-in, slide-out service access. In addition to providing the necessary voltages for the MULTIBUS cards, the efficient switching power supply also provides the 24 volts needed for the DC powered Winchester drive. The SYSTEM 86/330 has been designed to meet UL, CSA, FCC and VDE safety and EMI/RFI requirements. Supplied with the SYSTEM 86/330 are connectors for the RS232 channel, parallel line printer, and additional floppy disk and Winchester disk drives. Cut-outs are also available for OEM supplied connectors.

System Decomposition for Low Cost

The entire SYSTEM 86/330, both hardware and software, is built using standard, modular off-the-shelf Intel products that are available separately. The system is designed so that the OEM can easily decompose the SYSTEM 86/330 into the individual Intel products needed to tailor a custom system. Through the use of industry standard modules and interfaces, the OEM can perform his own system integration without changing software code, thus lowering his end product cost.

System Software

The SYSTEM 86/330 can be used for iRMX 86 software development, as well as being the target system for the OEM application.

VLSI OPERATING SYSTEM (iRMX™ 86)

The powerful iRMX 86 Operating System is an easy-to-use, comprehensive multiprogramming software system designed not only for the SYSTEM 86/330, but for iAPX 86/88-based iSBC board and component level designs.

Services provided by the RAM-based iRMX 86 Operating System include facilities for executing programs concurrently, sharing resources and information, servicing asynchronous events, and interactively controlling system resources and utilities. In addition, the iRMX 86 Operating System provides all major real-time facilities, including priority-based system resource allocation, means for concurrently monitoring and controlling multiple external events, real-time clock control, interrupt management, and task dispatching. The iRMX 86 Operating System contains the following modules: an object-oriented Nucleus; Device Independent Basic and Extended I/O Systems; Terminal Handler; Bootstrap and Application Loaders; Human Interface with complete command line interpreter; and an interactive, object-oriented Debugger.

Because the modules and services provided by the operating system are user selectable, application specific operating systems can be created by iRMX 86 users. The iRMX 86 Operating System therefore eliminates the need for custom operating system design, thereby reducing development time, cost, effort, and risk.

iRMX™ UTILITIES PACKAGE (iRMX™ 860)

The iRMX Utilities Package consists of the following software:

iRMX™ 86 EDIT — The iRMX 86 EDIT program provides users with a powerful, sophisticated, line-oriented editing facility. EDIT delivers a range of capability suitable for novice users as well as advanced capabilities for sophisticated users. Its key features include a macro processor capable of creating and executing complex strings of commands, which ease the editing chore, as well as defining blocks of text which may be included anywhere in the text file. EDIT offers variable command sourcing, symbolic line numbering and reference by symbol. The facilities of EDIT allow users to create, maintain and manipulate extensive libraries of source code with minimal effort. For more information on iRMX 86 EDIT, see the EDIT Software Package data sheet (143883).

iRMX™ 86 LINK/LOCATE — The iRMX 86 LINK/LOCATE program connects object modules which

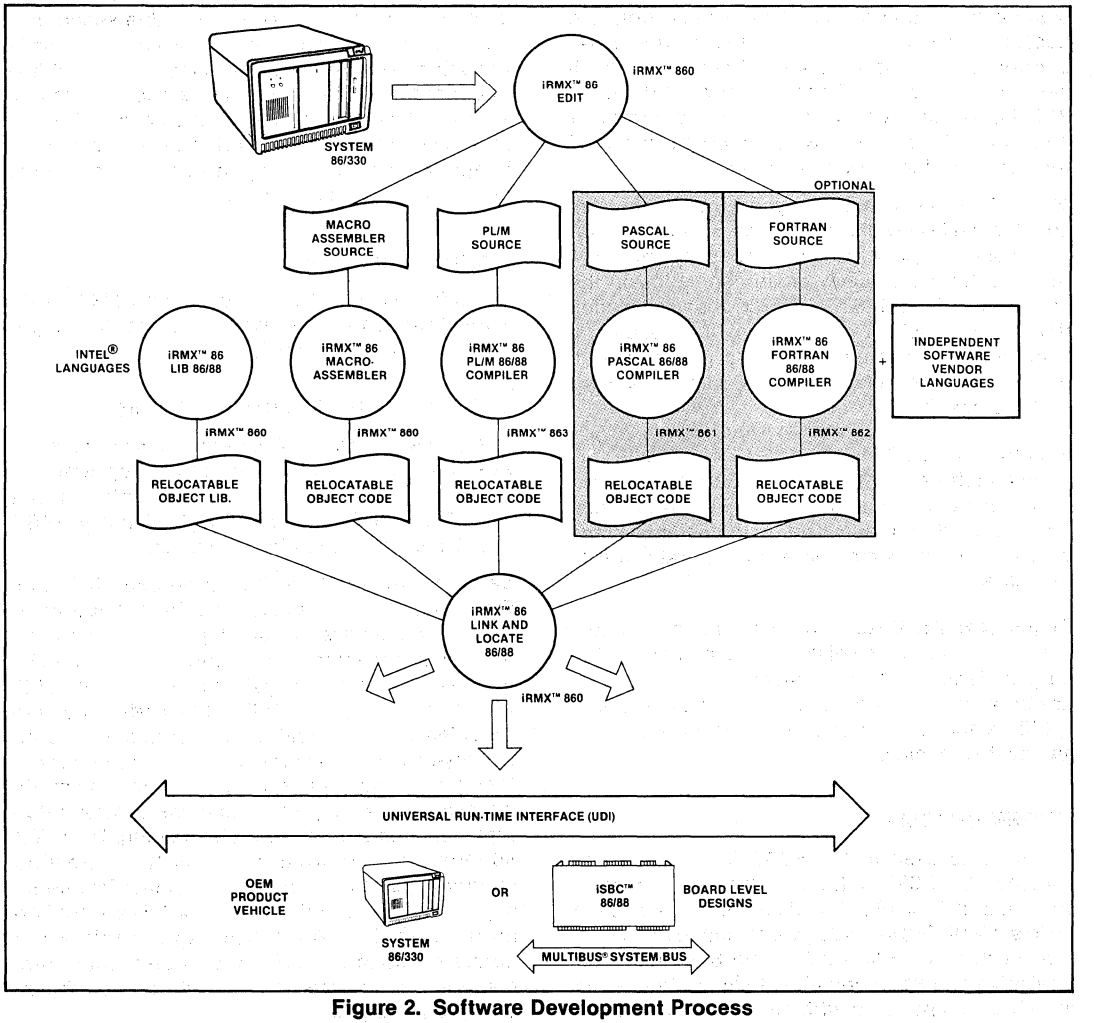


Figure 2. Software Development Process

have been individually compiled into a single, relocatable object module. The input object code may have been produced by any Object Module Format-compatible compiler. Output object modules may be recombined into larger object modules, allowing work from a large programming staff to be easily integrated into an application system.

iRMX™ 86 LIB — the iRMX 86 LIB "Library Manager" allows creation and maintenance of object module libraries. These libraries allow easy collection of related object code to reduce the overhead of maintaining many separate modules. Users may create new libraries, add and delete object modules, as well as list the contents of the library and their public symbols.

PL/M-86 (iRMX™ 863)

The PL/M-86 compiler provides users with a powerful, microcomputer-oriented system programming language. The PL/M-80 Language was introduced in 1976 by Intel. It was the first microcomputer-oriented, block structured, high-level language available. Since 1976, thousands of users, shipping over millions of microcomputer-based systems have generated their system software with PL/M-80 and PL/M-86.

PL/M-86 is a compatible superset of PL/M-80 which offers easy portability of software across the full range of microcomputers supplied by Intel. For more information about PL/M-86, see the PL/M 86/88 Software Package data sheet (402175).

FULL REALMATH SUPPORT

The iRMX 86 Languages support the REALMATH floating point standard. This allows users of all iRMX 86 languages to access the iAPX 86/20 Numeric Data Processors using the iSBC 337 MULTIMODULE board. These numeric processors offer over 100 times greater performance than comparable software-implemented algorithms, and reduce the system memory requirements by at least 16KB. The REALMATH standard (proposed IEEE standard) provides universal consistency in results of numeric computations. The iRMX 86 Languages provide efficient object code generation and access to the highest performance floating point package available on microcomputers.

COMPREHENSIVE SELF-TEST AND SYSTEM DIAGNOSTICS

In order to insure correct system operation and rapid location of both hardware and software problems, the SYSTEM 86/330 is supplied with three levels of diagnostic routines:

SCT (System Confidence Test) — Automatically executed at power on and system reset, the SCT performs a GO-NO GO condition for the processors, memory and devices in the system.

SDT (System Diagnostic Test) — In the event that the SCT finds a system problem, the SDT may be executed by the user for a detailed analysis of the hardware status. This menu driven monitor program can perform tests on all hardware boards in the system and mass storage peripherals.

SAT (System Analysis Test) — This diagnostic tool tests the integration of the hardware and software at the system level. This helps to isolate intermittent failures by running a load stress test on both hardware and software.

SYSTEM DEBUGGING TOOLS

The iRMX 86 Operating System provides a comprehensive tool for interactive software debugging. The Debugger has two capabilities that greatly simplify the process of debugging a multitasking system. First, the Debugger allows users to debug several tasks while the balance of the application system continues to run in real-time. Second, the Debugger allows programmers to interactively view and modify system constructs as well as the system RAM and CPU registers. The debugger is structured to enable system designers to track system-wide problems easily. It can also remain in the final application as a continuous maintenance tool.

OEM SOFTWARE LICENSING

The Intel software products listed above require the signing of an Intel Master Software License Agreement. All products include 1 year of update service. Software is shipped on floppy media in two object forms: 1) a ready-to-run, fully configured system, and 2) a configurable version of all software products.

OPTIONAL INTEL® SOFTWARE SUPPORT

In addition to ASM-86 and PL/M-86 included in the base SYSTEM 86/330, the following Intel languages are available for use on the system:

PASCAL-86 (iRMX™ 861) — The PASCAL-86 compiler provides a strict implementation of the proposed ISO language standard. All source programs are validated by the compiler to ensure its conformance to the standard. Many extensions to

the language are available which allow PASCAL programs to be written specifically for microcomputers. Separate module compilation and iAPX 86/20, 88/20 Numeric Data Processor support are a few of its many features. The ISO standard "source evaluator" can be switched off to accept these extensions. For more information on iRMX 86 PASCAL features, see the PASCAL 86/88 Software Package data sheet (121680).

FORTRAN-86 (iRMX™ 862) — The iRMX 86 FORTRAN compiler provides users total compatibility with existing FORTAN 86 language-generated code, plus many new language features provided by the FORTRAN 77 language standard. These new features offer FORTRAN programmers many new capabilities, including "IF-THEN-ELSE", random access I/O and character variables. For a more detailed explanation of iRMX 86 FORTRAN, see the FORTRAN 86/88 Software Package data sheet (400630).

iMMX 800 MULTIBUS® Message Exchange — The advanced design of the MULTIBUS system bus coupled with the iMMX 800 software package allows the SYSTEM 86/330 to easily support additional 8 and/or 16-bit Intel single board computers. This powerful option enables OEMs to increase the SYSTEM 86/330's general purpose processing power with boards such as the iSBC 86/05 (8 MHz 8086) and iSBC 88/25 (5 MHz 8088). Intelligent, high performance microprocessor based boards such as the Ethernet * communications controller (iSBC 550 board), serial communications controller (iSBC 544 board) and analog measurement and control computer (iSBC 88/40 board) can greatly expand the capabilities and processing power of the SYSTEM 86/330.

INDEPENDENT SOFTWARE VENDOR SUPPORT — Through the use of the standard UDI (Universal Development Interface) a wide variety of language products, both compilers and interpreters, are now available for use on iRMX 86 from independent software vendors. COBOL, BASIC, CBASIC, and C are a few examples of the available languages. Contact your Intel sales representative for more information.

System Options

HARDWARE

Any Intel MULTIBUS board may be installed by the OEM into the two available expansion slots in the system chassis.

SOFTWARE

Optional language products are available from both Intel and independent software vendors. Operating system support for additional 8085, 8088 and 8086-based processor boards is available through the use of iRMX 80, iRMX 88 and iRMX 86 software. Multiprocessor software support, iMMX 800, is also available for the SYSTEM 86/330.

* Ethernet is a trademark of Xerox Corp.

SPECIFICATIONS

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 microseconds for fastest executable instructions (assuming instruction is not in the queue).

Memory Capacity

RAM — 320K bytes supplied with the base system. Memory may be expanded to 1 megabyte.

Interface

EIA Standard RS232C signals provided and supported.

Serial — Configurable from 110 to 19.2K baud (asynchronous)

Parallel — A parallel I/O port configured to interface with an industry standard Centronics interface printer

AC Requirements

6.5A @ 88 to 126 VAC, 60 Hz, single-phase (US only); 3.25A @ 176 to 252 VAC, 50 Hz, single-phase (Europe only). Maximum total power consumption 350W.

Product Safety Standards

The system is listed under UL standard 114 Safety of Electronic Data Processing Units and Systems. It is also certified by the Canadian Standards Association, standard C22.2 154-1975 Safety of Data Processing Equipment. The system complies with the International Electronics Commission standard IEC 435 Safety of Data Processing Equip-

ment. The system also conforms to the applicable RFI/EMI requirements of VDE 0871/6.78, VDE 0875/6.77 and FCC rule 47 CFR part 15 subpart J Emmission Limits for Computing Devices.

Environmental Requirements

OPERATING

Temperature — 15°C to 35°C

Relative Humidity — 20% to 80% non-condensing over the operating temperature range*

Altitude — Sea level to 6000 feet

Vibration — 1.0g @ 10 to 55 Hz

***NOTE:** The environmental combination of humidity and temperature together cannot exceed 26°C wet bulb.

NON-OPERATING

Temperature — – 25°C to 60°C

Relative Humidity — 20% to 80% non-condensing

Altitude — Sea level to 12,000 feet

Vibration — 1.0g for 5 ms shock

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)

Extensive System Documentation

The SYSTEM 86/330 is shipped with six documentation binders containing over 28 in-depth manuals on all aspects of hardware and software operation. A system installation and maintenance manual, in addition to a system overview manual, are also included.

ORDERING INFORMATION

Part Number	Description
SYS 86/330 PKG	Desk-top, 120V 60 Hz
SYS 86/330E PKG	Desk-top, 220V 50 Hz
SYS 86/331 PKG	Rack-mount, 120V 60 Hz
SYS 86/331E PKG	Rack-mount, 220V 50 Hz
SYS 86/330 DOC	Complete hardware and software documentation set (included with system; use this order code for additional sets)

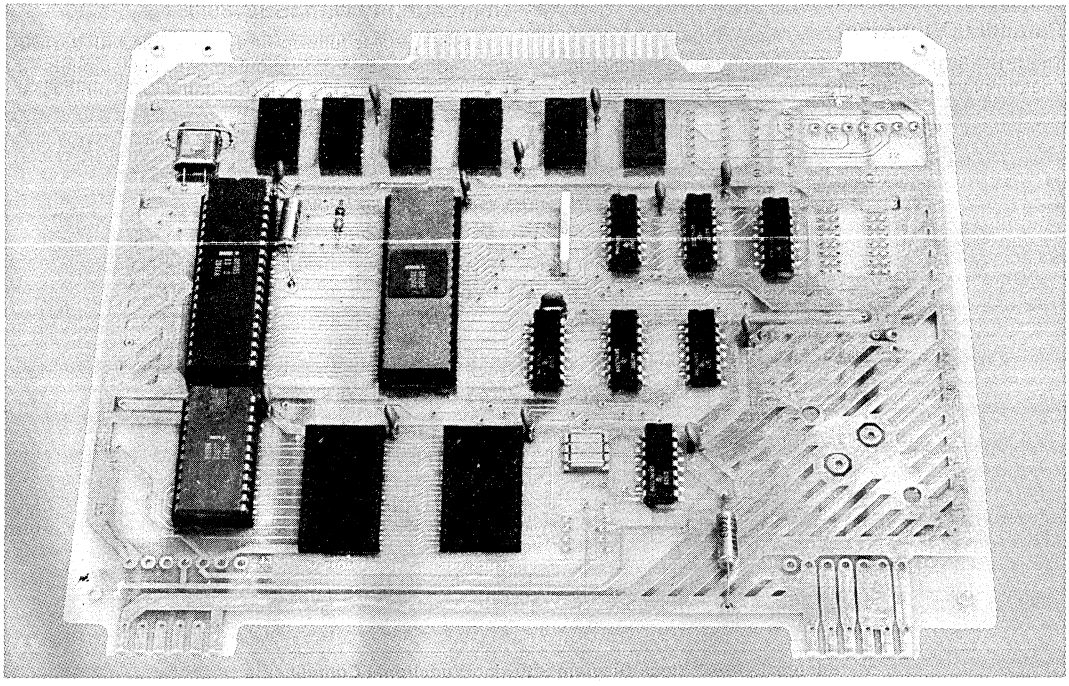
Single Board Computers

2

iSBC 80/04
SINGLE BOARD COMPUTER

- 8085A CPU used as central processor
- 256 bytes of static read/write memory
- Sockets for 4K bytes of erasable reprogrammable read only memory
- 22 programmable parallel I/O lines with sockets for interchangeable line drivers and terminators
- Optimized for stand-alone applications with provisions for on-board +5V regulator, heat sink, and mounting holes for attachment to user's equipment
- Programmable 14-bit binary timer
- TTL serial I/O interface with hole patterns for RS232C line drivers and receivers
- Four-level vectored interrupt
- Upward compatibility with iSBC 80/05
- Single +5V power supply

The iSBC 80/04 Single Board Computer is a member of Intel's complete line of OEM computer systems which take full advantage of Intel's LSI technology to provide economical, self-contained computer-based solutions for OEM applications. The iSBC 80/04 is a complete computer system on a single 6.75 x 7.85-inch printed circuit card. The CPU, system clock, read/write memory, nonvolatile read only memory, I/O ports and drivers, serial interface, priority interrupt logic, and programmable timer all reside on the board.



FUNCTIONAL DESCRIPTION

Intel's powerful 8-bit n-channel 8085A CPU, fabricated on a single LSI chip, is the central processor for the iSBC 80/04. The 8085A CPU is directly software compatible with the popular intel 8080A CPU. The 8085A contains six 8-bit general purpose registers and an accumulator. The six general purpose registers may be addressed individually or in pairs, providing both single and double precision operators. Minimum on-board instruction execution time is 2.03 microseconds. A block diagram of iSBC 80/04 functional components is shown in Figure 1.

Memory Addressing

The 8085A CPU has a 16-bit program counter which allows addressing of up to 65,536 bytes of memory. An external stack, located within any portion of iSBC 80/04 read/write memory, may be used as a last-in/first-out storage area for the contents of the program counter, flags, accumulator, and all of the six general purpose registers. A 16-bit stack pointer controls the addressing of this external stack. This stack provides subroutine nesting bounded only by memory size.

Memory Capacity

The iSBC 80/04 contains 256 bytes of read/write memory using the Intel 8155 RAM/IO/Timer. Two sockets for up to 4K bytes of nonvolatile read only memory are provided on the board. Read only memory may be added in 2K-byte increments using Intel 2716 erasable and electrically reprogrammable ROMs (EPROMs). Optionally, if only 2K bytes are required, read only memory may be added in 1K-byte increments using Intel 2708 EPROMs.

Parallel I/O Interface

The iSBC 80/04 contains 22 programmable parallel I/O lines implemented using the I/O ports of the Intel 8155 RAM/IO/Timer. The system software is used to configure the I/O lines in any combination of unidirectional input or output ports as indicated in Table 1. The I/O interface may, therefore, be customized to meet specific peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 22 programmable I/O lines and signal ground lines are brought out to a 50-pin edge connector that mates with flat, woven, or round cable.

Stand-Alone Applications

The iSBC 80/04 is designed to be a cost-effective solution for applications requiring a self-contained computer on a single board without the need for external memory or I/O options. In order to help minimize power supply cost in small systems, the iSBC 80/04 includes provision for an on-board + 5V regulator allowing unregulated voltage to be connected directly on the board. Regulated DC voltages are applied to the board through two 12-pin edge connectors which mate with flat, woven, or round cables. The iSBC 80/04 also includes pins that will accept MOLEX-type connectors for connection of regulated DC voltages. Mounting holes are provided in the corners of the iSBC 80/04 board which permit direct attachment to the user's equipment, thereby eliminating the need for cardcage and backplane.

Compatibility with iSBC 80/05

The iSBC 80/04 is fully upward compatible with the iSBC 80/05 Single Board Computer. Pin assignments for parallel I/O, serial I/O, and regulated DC voltages are

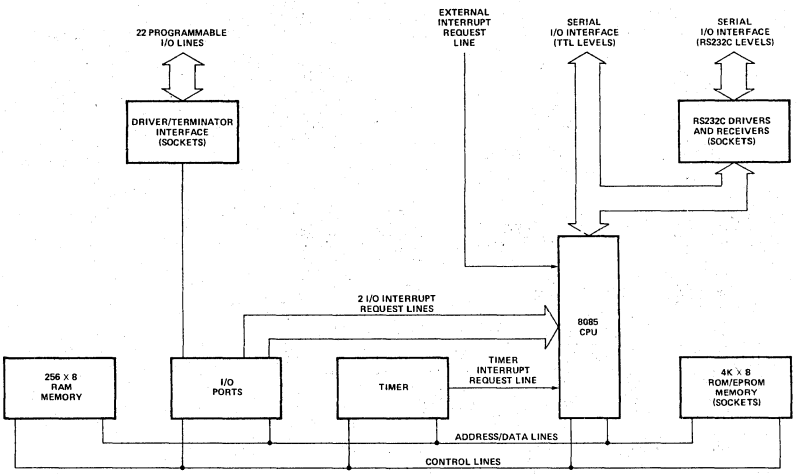


Figure 1. iSBC Block Diagram Showing Functional Components

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation				Control
		Unidirectional				
		Input		Output		
		Unlatched	Latched & Strobed	Latched	Latched & Strobed	
1	8	X	X	X	X	
2	8	X	X	X	X	
3	3	X		X		X[1]
4	3	X		X		X[2]

Notes

1. Port 3 must be used as a control port when port 1 is used as a latched and strobed input or a latched and strobed output port.
2. Port 4 must be used as a control port when port 2 is used as a latched and strobed input or a latched and strobed output port.

indentical to those of the iSBC 80/05. Additionally, software developed for the iSBC 80/04 will execute directly in the iSBC 80/05. In addition to the iSBC 80/04 features, the iSBC 80/05 contains a total of 512 bytes of read/write memory, allows for expansion of memory and I/O capacity, and provides full MULTIBUS arbitration control for multimaster applications.

Programmable Timer

The iSBC 80/04 provides a fully programmable binary 14-bit interval timer utilizing the Intel 8155 RAM/IO/Timer. The systems designer simply configures the time via software to meet system requirements. Whenever a given timer delay is needed, software commands to the programmable timer select the desired functions. Four functions are available as shown in Table 2. The contents of the timer counter may be read at any time during system operation.

Table 2. Programmable Timer Functions

Function	Operation
Programmable pulse	Timer out goes low during the second half of count. Therefore, the count loaded in the count length register should be twice the pulse width desired.
Square wave rate generator	Timer out remains high until one-half the count has been completed, and goes low for the other half of the count. The count length is automatically reloaded when terminal count is reached.
Rate generator	Divide by N counter. A repetitive timer out low pulse is generated and new timeout initiated every time terminal count is reached.
Programmable strobe	A single low pulse is generated upon reaching terminal count. This function is extremely useful for generation of real-time clocks.

Serial I/O Interface

The iSBC 80/04 prvides serial I/O capability through the serial input data (SID) and serial output data (SOD) functions of the Intel 8085A CPU. These functions are controlled exclusively by software through execution of the 8085A RIM and SIM instructions. The baud rate for the serial I/O interface is determined by the system time available for execution of serial I/O support software. Hence, the maximum baud rate supported by the iSBC 80/04 is solely dependent on the overall system real-time software requirements. Serial I/O signals are TTL compatible, and hole patterns are provided on the board for optional installation of RS232C line drivers and receivers.

Interrupt Capability

The iSBC 80/04 takes advantage of the powerful interrupt processing capability of the 8085A CPU. Interrupt requests are routed to four interrupt inputs of the 8085A CPU (i.e., TRAP, RST 7.5, RST 6.5, and RST 5.5 in order of priority, TRAP highest), and each input generates a unique memory address (i.e., TRAP: 26_{16}, RST 7.5: $3C_{16}$, RST 6.5: 34_{16}, RST 5.5: $2C_{16}$). A single 8085A jump instruction at each of these addresses then provides linkage to locate each interrupt service routine independently anywhere in memory. All interrupt inputs with the exception of one (TRAP) may be masked via software. The trap interrupt should be used for conditions such as power-down sequences which require attention by the 8085A CPU.

Interrupt Generation — The iSBC 80/04 accepts interrupts from four sources. An interrupt is automatically generated by the programmable interval timer/event counter upon completion of the selected function. Two interrupts are automatically generated by the I/O ports section of the 8155 when ports 1 or 2 of the 8155 are programmed to operate in the "latched and strobed" mode (see Table 1). The fourth interrupt source is available to the user and should be used to inform the 8085A CPU of catastrophic errors such as power failure. This user-defined source is connected to the trap input of the 8085A CPU.

Systems Development Capability

The development cycle of the iSBC 80/04-based products may be significantly reduced using an Intellec microcomputer development system. The resident macroassembler, text editor, and system monitor greatly simplify the design, development, and debug of iSBC 80/04 system software. An optional diskette operating system provides a relocating macroassembler, a relocating loader and linkage editor, and a library manager. A unique in-circuit emulator (ICE-85) option provides the capability of developing and debugging software directly on the iSBC 80/04.

Programming Capability

PL/M-80 — Intel's high level programming language, PL/M, is also available as a resident Intellec microcomputer development system option. PL/M provides the capability to program in a natural, algorithmic language and eliminates the need to manage register usage or allocate memory. PL/M programs can be written in a much shorter time than assembly language programs for a given application.

FORTRAN-80 — For applications requiring computational and formatted I/O capabilities, the high level FORTRAN-80 programming language is also available as a resident option of the Intellec system. The FORTRAN compiler produces relocatable object code that may be easily linked with PL/M or assembly language program modules. This gives the user a wide flexibility in developing software.

SPECIFICATIONS

Word Size

Instruction — 8, 16, or 24 bits
Data — 8 bits

Cycle Time

Basic Instruction Cycle — 2.03 μs, ± 0.1%

Note

Basic instruction cycle is defined as the fastest instruction (i.e., four clock cycles).

Memory Addressing

ROM/EPROM — $0\text{-}0FFF_H$
RAM — $3F00_H$

Memory Capacity

ROM/EPROM — 4K bytes (sockets only)
RAM — 256 bytes

I/O Addressing

On-Board Programming I/O — see Table 1

Port Control	8155 Port 1	8155 Port 2	8155 3 & 4	8155 Ports	8155 Timer Low-Order Byte	8155 Timer High-Order Byte
Address	00	01	02	03	04	05

I/O Capacity

Parallel — 22 programmable lines (see Table 1)

Serial Communications Characteristics

SID and SOD functions of the 8085 CPU are used for serial I/O. Controlled by software through RIM and SIM instructions of the 8085A CPU. Baud rate determined by system time available for serial I/O handling. On-board timer may be used to greatly ease serial I/O timing requirements.

Interrupts

Four-level interrupt routed to 8085 CPU interrupt inputs. Each interrupt automatically vectors the processor to a unique memory location.

Condition	Interrupt Input	Memory Address	Priority	Type
User-defined	TRAP	24_{16}	Highest	Non-maskable
Timer	RST 7.5	$3C_{16}$	↑	Maskable
I/O Port 2	RST 6.5	34_{16}	↓	Maskable
I/O Port 1	RST 5.5	$2C_{16}$	Lowest	Maskable

Timer

Input Frequency Reference — 122.88 kHz ± 0.1% (8.14 μs period nominal)

Output Frequencies/Timing Intervals

Function	Timer/Counter	
	Min	Max
Programmable pulse	8.14 μs	66.67 ms
Square wave rate generator	7.50 Hz	61.44 kHz
Rate generator	7.50 Hz	61.44 kHz
Programmable strobe	8.14 μs	133.33 ms

Interfaces

Parallel I/O — All signals TTL compatible
Interrupt Request — All TTL compatible (active-low)
Serial I/O — TTL; hole patterns available for user installation of RS232C line drivers and receivers

System Clock (8085 CPU)

1.966 MHz ± 0.1%

Connectors

Interface		Pins (no.)	Center (in.)	Mating Connectors[1]
Voltages	+5V, +12V, −5V[2]	7 single-sided	0.156	Molex 09-66-1071 Connector Molex 09-50-7071 Connector
				AMP 87194-6 Connector AMP 3-87025-4 Connector
	+5V, −12V[3]	7 single-sided	0.156	Molex 09-66-1071 Connector Molex 09-50-7071 Connector
				AMP 87194-6 Connector AMP 3-87025-4 Connector
	Unregulated +5V	2 single-sided	0.156	Molex 09-66-1021 Connector Molex 09-50-7021 Connector
				AMP 89194-1 Connector AMP 2-87025-5 Connector
Parallel I/O		50 double-sided	0.1	3M 3415-000 (flat cable)
Serial I/O		7 single-ended	0.156	Molex 09-66-1071 Connector Molex 09-50-7071 Connector
				AMP 87194-6 Connector AMP 3-87025-4 Connector

Notes

1. Connectors and pins from a given vendor may only be used with connectors and pins from the same vendor.
2. A single 86-contact edge-on connector may be used to connect the two groups of regulated voltages (i.e., +5V, +12V, −5V, and +5V, −12V).
3. Required only when RS232C line drivers and receivers are used.

Line Drivers and Terminators

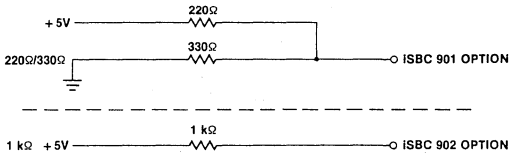
I/O Drivers — The following line drivers are all compatible with the I/O driver sockets on the iSBC 80/04:

Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note

I = inverting; NI = non-inverting; OC = open collector.

I/O Terminators — Intel provides 220Ω/330Ω divider and 1 kΩ pull-up resistive terminator packs for termination of I/O lines programmed as inputs. These options are as follows:



RS232C Drivers and Receivers

The following RS232C drivers and receivers are compatible with the RS232C socket on the iSBC 80/04:

RS232C Driver — National DS1488 or TI SN75188
RS232C Receiver — National DS1490 or TI SN75189

Sockets

Sockets may be installed in the hole patterns provided for the RS232C drivers and receivers. The following sockets are compatible with the iSBC 80/04: TI C93-14-02 and SCANBE US-2-14-160-N-B.

Compatible Voltage Regulator

National LM 323 — 3A, 5V Positive Regulator
Fairchild μA7805 KM — 1A, 5V Positive Regulator

Compatible Heat Sink

IERC — LA Series or
AAVID Engineering, Inc. — Series 5051

Physical Characteristics

Width — 7.85 in. (19.94 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 6.0 oz (169.9 gm)

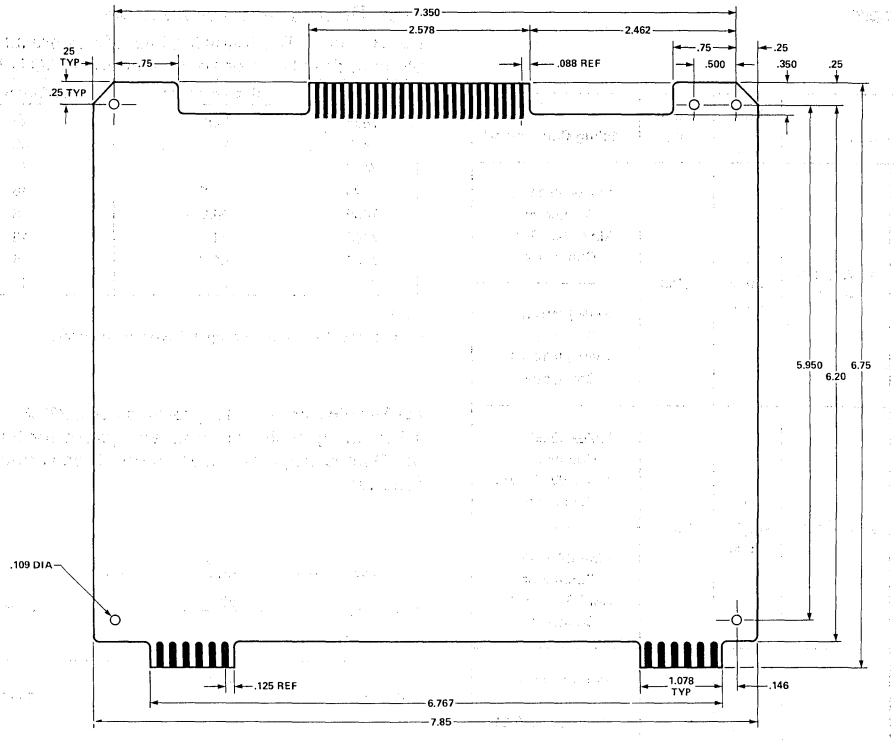


Figure 2. iSBC 80/04 Dimensions

Electrical Characteristics

DC Power Requirements

Voltage (±5%)	Without PROM[1] (max)	With 2716 EPROM[2] (max)	With 2708 EPROM[3] (max)
V_{CC} = +5V	I_{CC} = 600 mA	1.45A	1.25A
V_{DD} = +12V [4]	I_{DD} = 0	7 mA[5]	137 mA
V_{BB} = −5V [4]	I_{BB} = 0	0	90 mA
V_{AA} = −12V [5]	I_{AA} = 0	23 mA[5]	23 mA[5]

Notes

1. Does not include power required for optional EPROM/ROM, I/O drivers, and I/O terminators.
2. With two Intel 2716 EPROMs and 220Ω/330Ω terminators installed for 22 input ports; all terminator inputs low.
3. With two Intel 2708 EPROMs and 220Ω/330Ω terminators installed for 22 input ports; all terminator inputs low.
4. Required for 2708 EPROMs.
5. Required only when RS232C capability required.

Environmental Characteristics

Operating Temperature — 0°C to +55°C

Reference Manual

9800482-02 - iSBC 80/04 Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

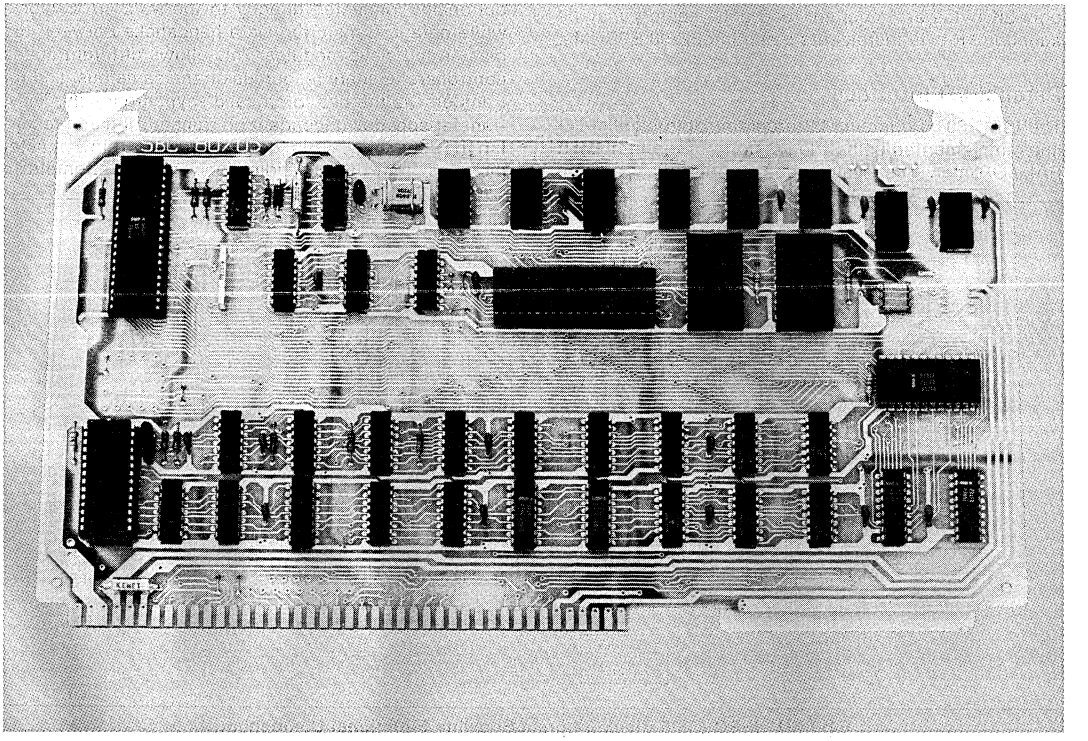
ORDERING INFORMATION

Part Number	Description
SBC 80/04	Single Board Computer

iSBC 80/05 (or pSBC 80/05*) SINGLE BOARD COMPUTER

- 8085A CPU used as central processor
- 512 bytes of static read/write memory
- Sockets for 4K bytes of erasable reprogrammable or masked read only memory
- 22 programmable parallel I/O lines with sockets for interchangeable line drivers and terminators
- Full MULTIBUS control logic allowing up to 16 masters to share system bus
- Programmable 14-bit binary timer
- TTL serial I/O interface with sockets for RS232C line drivers and receivers
- Four-level vectored interrupt
- Fully compatible with optional iSBC expansion boards and peripherals
- Single +5V power supply

The iSBC 80/05 Single Board Computer is a member of Intel's complete line of OEM computer systems which take full advantage of Intel's LSI technology to provide economical, self-contained computer-based solutions for OEM applications. The iSBC 80/05 is a complete computer system on a single 6.75 × 12.00-inch printed circuit card. The CPU, system clock, read/write memory, nonvolatile read only memory, I/O ports and drivers, serial interface, priority interrupt logic, programmable timer, MULTIBUS control logic, and bus expansion buffers all reside on the board.



*Same product, manufactured by Intel Puerto Rico, Inc.

FUNCTIONAL DESCRIPTION

Intel's powerful 8-bit n-channel 8085 CPU, fabricated on a single LSI chip, is the central processor for the iSBC 80/05. The 8085A CPU is directly software compatible with the popular Intel 8080A CPU. The 8085A contains six 8-bit general purpose registers and an accumulator. The six general purpose registers may be addressed individually or in pairs, providing both single and double precision operators. Minimum on-board instruction execution time is 2.03 microseconds. A block diagram of iSBC 80/05 functional components is shown in Figure 1.

Memory Addressing

The 8085A CPU has a 16-bit program counter which allows direct addressing of up to 65,536 bytes of memory. An external stack, located within any portion of read/write memory, may be used as a last-in/first-out storage area for the contents of the program counter, flags, accumulator, and all of the six general purpose registers. A 16-bit stack pointer controls the addressing of this external stack. This stack provides subroutine nesting bounded only by memory size.

Memory Capacity

The iSBC 80/05 contains 512 bytes of read/write memory using Intel's low power static RAMs. Two sockets for up to 4K bytes of nonvolatile read only memory are provided on the board. Read only memory may be added in 2K-byte increments using Intel 2716 erasable and electrically reprogrammable ROMs (EPROMs). Optionally, if only 2K bytes are required, read only memory may be added in 1K-byte increments using Intel 2708 EPROMs.

Parallel I/O Interface

The iSBC 80/05 contains 22 programmable parallel I/O lines implemented using the I/O ports of the Intel 8155 RAM/IO/Timer. The system software is used to configure the I/O lines in any combination of unidirectional input or output ports as indicated in Table 1. The I/O interface may, therefore, be customized to meet specific peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 22 programmable I/O lines and signal ground lines are brought out to a 40-pin edge connector that mates with flat, woven, or round cable.

Multimaster Capability

The iSBC 8085A is a full computer on a single board with resources capable of supporting a great variety of OEM system requirements. For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e., several CPUs and/or controllers logically share systems tasks with communication over the system bus), the iSBC 80/05 provides full MULTIBUS arbitration control logic. This control logic allows up to three bus masters (i.e., any combination of iSBC 80/05, iSBC 80/20-4, DMA controller, diskette controller, etc.) to share the system bus in serial (daisy-chain) priority fashion, and up to 16 masters may share the MULTIBUS with the addition of an external priority network. The MULTIBUS arbitration logic operates synchronously with a MULTIBUS clock (provided by the iSBC 80/05 or optionally connected directly to the MULTIBUS clock) while data is transferred via a handshake between the master and slave modules. This allows different speed controllers to share resources on the same bus, and for transfers via the bus to proceed asynchronously. Thus, transfer speed is dependent on transmitting and receiving devices only. This design prevents slow master modules from being handicapped in their attempts to

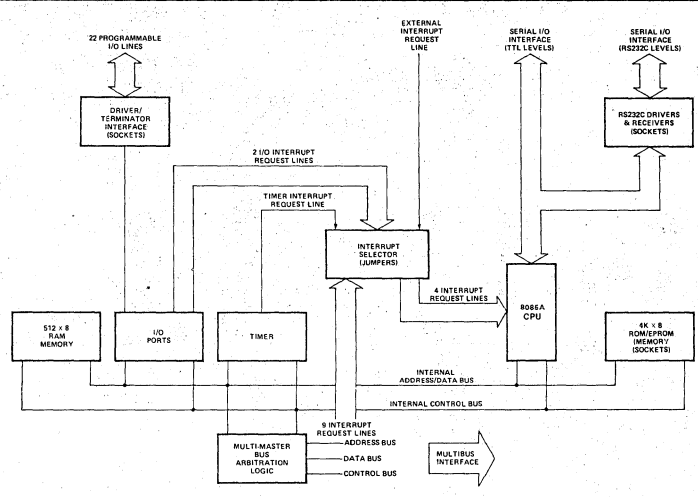


Figure 1. iSBC 80/05 Block Diagram Showing Functional Components

Table 1. Input/Output Modes of Operation

Port	Lines (qty)	Mode of Operation				Control
		Unidirectional				
		Input		Output		
		Unlatched	Latched & Strobed	Latched	Latched & Strobed	
1	8	X	X	X	X	
2	8	X	X	X	X	
3	3	X		X		X[1]
4	3	X		X		X[2]

Notes

1. Port 3 must be used as a control port when port 1 is used as a latched and strobed input or a latched and strobed output port.
2. Port 4 must be used as a control port when port 2 is used as a latched and strobed input or a latched and strobed output port.

gain control of the bus, but does not restrict the speed at which faster modules can transfer data via the same bus. The most obvious applications for the master-slave capabilities of the bus are multiprocessor configurations, high speed direct-memory-access (DMA) operations and high speed peripheral control, but are by no means limited to these three.

Programmable Timer

The iSBC 80/05 provides a fully programmable binary 14-bit interval timer utilizing the Intel 8155 RAM/IO/Timer. The system designer simply configures the timer via software to meet system requirements. Whenever a given time delay is needed, software commands to the programmable timer select the desired function. Four functions are available as shown in Table 2. The contents of the timer counter may be read at any time during system operation.

Serial I/O Interface

The iSBC 80/05 provides serial I/O capability through the serial input data (SID) and serial output data (SOD) functions of the Intel 8085A CPU. These functions are controlled exculsively by software through execution of the 8085A RIM and SIM instructions. The baud rate for the serial I/O interface is determined by the system time available for execution of serial I/O support software. Hence, the maximum baud rate supported by the iSBC 80/05 is solely dependent on the overall system real-time software requirements. Serial I/O signals are TTL compatible and sockets are provided on the board for optional connection of RS232C line drivers and receivers.

Interrupt Capability

The iSBC 80/05 takes advantage of the powerful interrupt processing capability of the 8085A CPU. Interrupt requests are routed to the four interrupt inputs of the 8085A CPU (i.e., TRAP, RST 7.5, RST 6.5, and RST 5.5 in order of priority, TRAP highest), and each input generates a unique memory address (i.e., TRAP: 24_{16}, RST 7.5: $3C_{16}$, RST 6.5: 34_{16}, RST 5.5: $2C_{16}$). A single 8085A jump instruction at each of these addresses then provides linkage to locate each interrupt service routine independently anywhere in memory. All interrupt inputs with the exception of one (TRAP) may be masked via software. The trap interrupt should be used for conditions such as power-down sequences which require immediate attention by the 8085A CPU.

Table 2. Programmable Timer Functions

Function	Operation
Programmable pulse	Timer out goes low during the second half of count. Therefore, the count loaded in the count length register should be twice the pulse width desired.
Square wave rate generator	Timer out will remain high until one-half the count has been completed, and go low for the other half of the count. The count length is automatically reloaded when terminal count is reached.
Rate generator	Divide by N counter. A repetitive timer out low pulse is generated and new timeout initiated every time terminal count is reached.
Programmable strobe	A single low pulse is generated upon reaching terminal count. This function is extremely useful for generation of real-time clocks.

Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS compatible expansion boards. High speed integer and floating-point arithmetic capabilities may be added by using the iSBC 310A High Speed Mathematics Unit. Memory may be expanded to 65,536 bytes by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be increased by adding digital I/O and analog I/O expansion boards. Mass storage capability may be achieved by adding

single or double density diskette controllers as subsystems. Modular expandable backplanes and cardcages are available to support multiboard systems.

A unique in-circuit emulator (ICE-85) option provides the capability of developing and debugging software directly on the iSBC 80/05.

Systems Development Capability

The development cycle of iSBC 80/05-based products may be significantly reduced using an Intellec microcomputer development system. The resident macroassembler, text editor, and system monitor greatly simplify the design, development, and debug of iSBC 80/05 system software. An optional diskette operating system provides a relocating macroassembler, a relocating loader and linkage editior, and a library manager.

Programming Capability

PL/M-80 — Intel's high level programming language, PL/M, is also available as a resident Intellec microcomputer development system option. PL/M provides the capability to program in a natural, algorithmic language and eliminates the need to manage register usage or allocate memory. PL/M programs can be written in a much shorter time than assembly language programs for a given application.

SPECIFICATIONS

Word Size

Instruction — 8, 16, or 24 bits
Data — 8 bits

Cycle Time

Basic Instruction Cycle — 2.03 μs, ± 0.1%

Note

Basic instruction cycle is defined as the fastest instruction (i.e., four clock cycles).

Memory Addressing

ROM/EPROM — $0\text{-}0FFF_H$
RAM — $3E00_H$

Memory Capacity

On-Board ROM/EPROM — 4K bytes (with Intel 2716) or 2K bytes (with Intel 2708)

On-Board RAM — 512 bytes

Off-Board Expansion — Up to 65,536 bytes in user specified combination of RAM, ROM, and PROM

I/O Addressing

On-Board Programmable I/O — see Table 1

Port Control	8155 Port 1	8155 Port 2	8155 Ports 3 & 4	8155 Port	8155 Timer Low-Order Byte	8155 Timer High-Order Byte
Address	00	01	02	03	04	05

I/O Capacity

Parallel — 22 programmable lines (see Table 1)

Note

The iSBC 80/05 may be expanded to 1102 programmable input/output lines by using optional iSBC 80 I/O boards.

Serial Communications Characteristics

SID and SOD functions of the 8085A CPU are used for serial I/O. They are controlled by software through RIM and SIM instructions of the 8085A CPU. Baud rate is determined by system time available for serial I/O handling. On-board timer may be used to greatly ease serial I/O timing requirements.

Interrupts

Four-level interrupt routed to 8085A CPU interrupt inputs. Each interrupt automatically vectors the processor to a unique memory location.

Interrupt Input	Memory Address	Priority	Type
TRAP	24_{16}	Highest	Non-maskable
RST 7.5	$3C_{16}$	↑	Maskable
RST 6.5	34_{16}	↓	Maskable
RST 5.5	$2C_{16}$	Lowest	Maskable

Timer

Input Frequency Reference — 122.88 kHz ± 0.1% (8.14 μs period nominal)

Output Frequencies/Timing Intervals

Function	Timer/Counter	
	Min	Max
Programmable pulse	8.14 μs	66.67 ms
Square wave rate generator	7.50 Hz	61.44 kHz
Rate generator	7.50 Hz	61.44 kHz
Programmable strobe	8.14 μs	133.33 ms

Interfaces

Bus — All signals TTL compatible
Parallel I/O — All signals TTL compatible
Interrupt Request — All TTL compatible (active-low)
Serial I/O — TTL; sockets available for RS232C line drivers and receivers

System Clock (8085A CPU)

1.966 MHz ± 0.1%

Interface	Lines (qty)	Centers (in.)	Mating Connector
Bus	86 double-sided	0.156	Viking 2KH43/9AMK12
Parallel I/O	50 double-sided	0.100	3M 3415-000
Serial I/O[1]	7 single-sided	0.156	Molex 09-66-1071 Connector Molex 09-50-7071 Connector
			AMP 87194-6 Connector AMP 3-87025-4 Connector

Note

1. Connectors and pins from one vendor may only be used with connectors and pins from the same vendor.

Line Drivers and Terminators

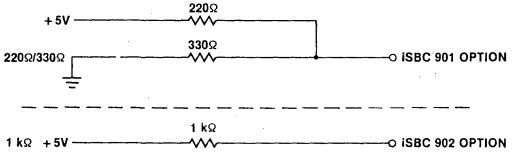
I/O Drivers — The following line drivers are all compatible with the I/O driver sockets on the iSBC 80/05:

Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note

I = inverting; NI = non-inverting; OC = open collector.

I/O Terminators — Intel provides 220Ω/330Ω divider and 1 kΩ pull-up resistive terminator packs for termination of I/O lines programmed as inputs. These options are as follows:



Bus Drivers

Driver	Characteristic	Sink Current (mA)
Data	Tri-state	50
Address	Tri-state	50
Commands	Tri-state	32

RS232C Drivers and Receivers

The following RS232C drivers and receivers are compatible with the RS232C socket on the iSBC 80/05:

RS232C Driver — National DS1488 or TI SN75188
RS232C Receiver — National DS1490 or TI SN75189

Physical Characteristics

Width — 12.00 in. (30.49 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 12.0 oz (339.8 gm)

Electrical Characteristics

DC Power Requirements

Voltage (±5%)	Without PROM[1] (max)	With 2716 EPROM[2] (max)	With 8708 EPROM[3] (max)
V_{CC} = +5V	I_{CC} = 1.80 mA	2.65A	2.45A
V_{DD} = +12V[4]	I_{DD} = 0	7 mA[5]	137 mA
V_{BB} = −5V[4]	I_{BB} = 0	0	90 mA
V_{AA} = −12V[5]	I_{AA} = 0	23 mA[5]	23 mA[5]

Notes

1. Does not include power required for optional EPROM/ROM, I/O drivers, and I/O terminators.
2. With two Intel 2716 EPROMs and 220Ω/330Ω terminators installed for 22 input ports; all terminator inputs low.
3. With two Intel 2708 EPROMs and 220Ω/330Ω terminators installed for 22 input ports; all terminator inputs low.
4. Required for 2708 EPROMs.
5. Required only when RS232C capability required.

Environmental Characteristics

Operating Temperature — 0°C to +55°C.

Reference Manual

9800483D — iSBC 80/05 Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

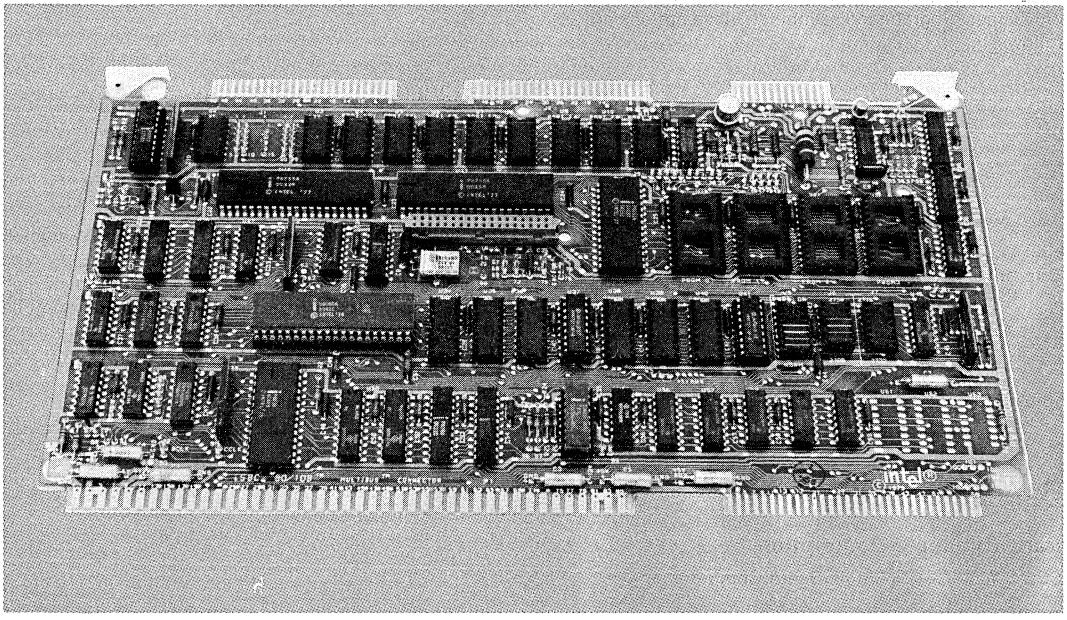
ORDERING INFORMATION

Part Number	Description
SBC 80/05	Single Board Computer

iSBC™ 80/10B (or pSBC 80/10B*) SINGLE BOARD COMPUTER

- **Upward compatible with iSBC™ 80/10A Single Board Computer**
- **8080A CPU used as central processing unit**
- **One iSBX™ bus connector for iSBX™ MULTIMODULE™ board expansion**
- **1K byte of read/write memory with sockets for expansion up to 4K bytes**
- **Sockets for up to 16K bytes of read only memory**
- **48 programmable parallel I/O lines with sockets for interchangeable line drivers and terminators**
- **Programmable synchronous/asynchronous communications interface with selectable RS232C or teletypewriter compatibility**
- **Single level interrupt with 11 interrupt sources**
- **Auxiliary power bus and power-fail interrupt control logic for RAM battery backup**
- **1.04 millisecond interval timer**
- **Limited master MULTIBUS® interface**

The Intel® iSBC 80/10B board is a member of Intel's complete line of OEM microcomputer systems which take full advantage of Intel's LSI technology to provide economical, self-contained computer-based solutions for OEM applications. The iSBC 80/10B board is a complete computer system on a single 6.75 x 12.00-inch printed circuit card. The CPU, system clock, iSBX bus interface, read/write memory, read only memory sockets, I/O ports and drivers, serial communications interface, bus control logic, and drivers all reside on the board.



*Same product, manufactured by Intel Puerto Rico, Inc.

FUNCTIONAL DESCRIPTION

Intel's powerful 8-bit n-channel MOS 8080A CPU, fabricated on a single LSI chip, is the central processor for the iSBC 80/10B board. The 8080A contains six 8-bit general purpose registers and an accumulator. The six general purpose registers may be addressed individually or in pairs, providing both single and double precision operators. A block diagram of iSBC 80/10B board functional components is shown in Figure 1.

iSBX Bus MULTIMODULE Board Expansion

The new iSBX bus interface brings an entirely new dimension to system design offering incremental on-board expansion with small iSBX boards. One iSBX bus connector interface is provided to accomplish plug-in expansion with any iSBX MULTIMODULE board. iSBX boards are available to provide expansion equivalent to the I/O available on the iSBC 80/10B board or the user may configure entirely new functionality such as math directly on-board. The iSBX 350 programmable I/O MULTIMODULE board provides 24 I/O lines using an 8255A programmable peripheral interface. Therefore, the iSBX 350 module together with the iSBC 80/10B board may offer 72 lines of programmable I/O. Alternately, a serial port may be added using the iSBX 351 serial I/O multimodule board or math may be configured on-board with the iSBX 332 floating point math MULTIMODULE board.

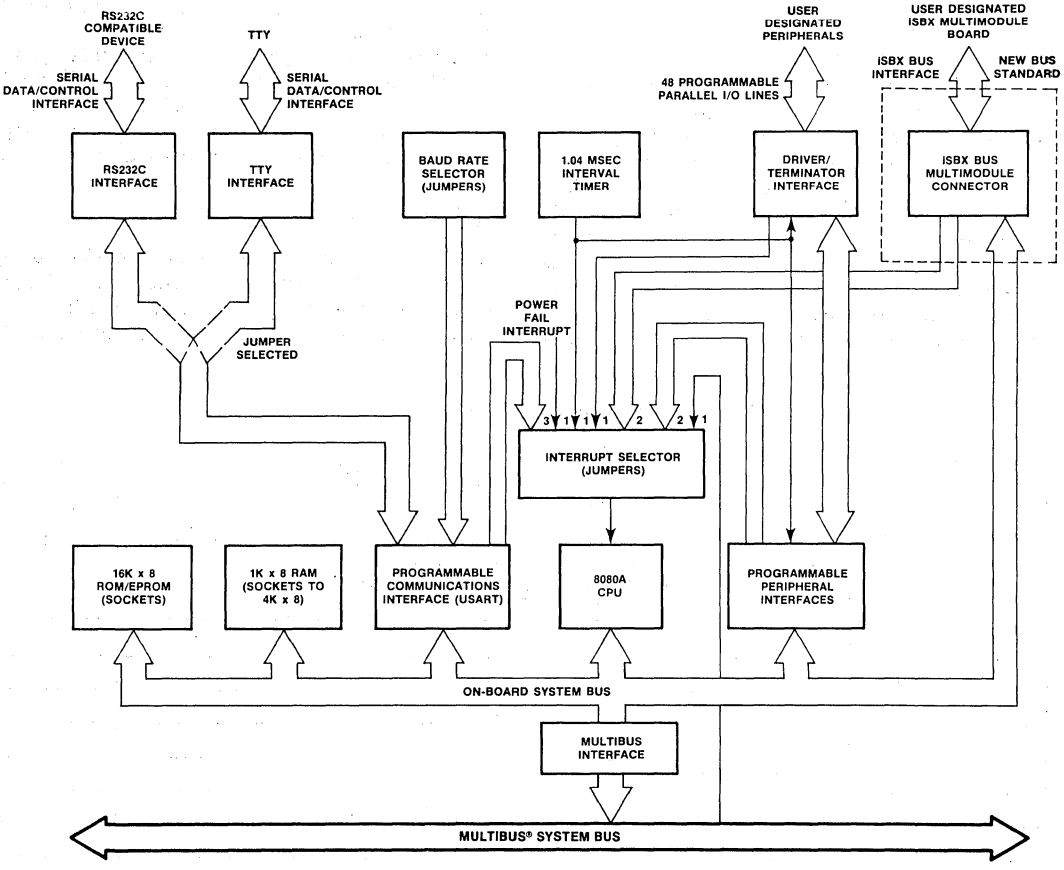


Figure 1. iSBC™ 80/10B Single Board Computer Block Diagram

The iSBX board is a logical extension of the on-board programmable I/O and is accessed by the iSBC 80/10B single board computer as common I/O port locations. The iSBX board is coupled directly to the 8080A CPU and therefore becomes an integral element of the iSBC 80/10B single board computer providing optimum performance.

Memory Addressing

The 8080A has a 16-bit program counter which allows direct addressing of up to 64K bytes of memory. An external stack, located within any portion of read/write memory, may be used as a last-in/first-out storage area for the contents of the program counter, flags, accumulator, and all of the six general purpose registers. A 16-bit stack pointer controls the addressing of this external stack. This stack provides subroutine nesting bounded only by memory size.

Memory Capacity

The iSBC 80/10B board contains 1K bytes of read/write static memory. In addition, sockets for up to 4K bytes of RAM memory are provided on board. Read/write memory may be added in 1K byte increments using two 1K×4 Intel 2114A-5 static RAMs. All on-board RAM read and write operations are performed at maximum processor speed. Sockets for up to 16K bytes of nonvolatile read-only-memory are provided on the board. Read-only-memory may be added in 1K byte increments up to 4K bytes (using Intel 2708 or 2758); in 2K byte increments up to 8K bytes (using Intel 2716); or in 4K byte increments up to 16K bytes (using Intel 2732). All on-board ROM or EPROM read operations are performed at maximum processor speed.

Parallel I/O Interface

The iSBC 80/10B board contains 48 programmable parallel I/O lines implemented using two Intel 8255A programmable peripheral interfaces. The system software is used to configure the I/O lines in any combination of unidirectional input/output, and bidirectional ports indicated in Table 1. Therefore, the I/O interface may be customized to meet specific peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 48 programmable I/O lines and signal ground lines are brought out to two 50-pin edge connectors that mate with flat cable or round cable.

Serial I/O Interface

A programmable communications interface using the Intel® 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the board. A jumper selectable baud rate

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	8	X		X			X[1]
4	8	X		X			
5	8	X		X			
6	4	X		X			
	4	X		X			

Notes

Port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.

generator provides the USART with all common communications frequencies. The USART can be programmed by the system software to select the desired synchronous or asynchronous serial data transmission technique (including IBM Bi-Sync). The mode of operation (i.e., synchronous or asynchronous), data format, control character format and parity are all under program control. The 8251A provides full duplex, double-buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The inclusion of jumper selectable TTY or RS232C compatible interfaces on the board, in conjunction with the USART, provides a direct interface to teletypes, CRTs, RS232C compatible cassettes, and asynchronous and synchronous modems. The RS232C or TTY command lines, serial data lines, and signal ground lines are brought out to a 26-pin edge connector that mates with RS232C compatible flat or round cable.

Interrupt Capability

Interrupt requests may originate from 11 sources. Two jumper selectable interrupt requests can be automatically generated by the programmable peripheral interface when a byte of information is ready to be transferred to the CPU (i.e., input buffer is full) or a byte of information has been transferred to a peripheral device (i.e., output buffer is empty). Three jumper selectable interrupt requests can be automatically generated by the USART when a character is ready to be transferred to the CPU (i.e., receive channel buffer is full), a character is ready to be transmitted (i.e., the USART is ready to accept a character from the CPU), or when the transmitter is empty (i.e., the USART has no character to transmit). These five interrupt request lines are all maskable under program control. Two interrupt request lines may be interfaced directly to user designated peripheral devices; one via the MULTIBUS system bus and the other via the I/O edge connector. One jumper selectable interrupt request may be interfaced to the power-fail interrupt control logic. One jumper selectable interrupt request may originated from the interval timer. Two general purpose interrupt requests are jumper selectable from the iSBX interface. These two signals permit a user installed MULTIMODULE board to interrupt the 8080A CPU. The eleven interrupt request lines share a single CPU interrupt level. When an interrupt request is recognized, a restart instruction (RESTART 7) is generated. The processor responds by suspending program execution and executing a user defined interrupt service routine originating at location 38_{16}.

Power-Fail Control

A power-fail interrupt may be detected through the AC-low signal generated by the power supply. This signal may be configured to interrupt the 8080A CPU to initiate an orderly power down instruction sequence.

Interval Timer

A 1.04 millisecond timer is available for interval interrupts or as a clock output to the parallel I/O connector. The timer output is jumper selectable to the programmable parallel interface, the parallel I/O connector (J1), or directly to the 8080A CPU.

MULTIBUS® System Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS™ system compatible expansion boards. Memory may be expanded to 65,536 bytes by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be increased by adding digital I/O and analog I/O expansion boards. In addition, the iSBC 80/10B board performs as a limited bus master in that it must occupy the lowest priority when used with other MULTIBUS masters. The bus master may take control of the MULTIBUS system bus by halting the iSBC 80/10B board program execution. Mass storage capability may be achieved by adding single density diskette, double density diskette, or hard disk controllers. Modular expandable backplanes and cardcages are available to support multiboard systems.

Real-Time Software

The iRMX 80 executive, which contains all major real-time facilities including priority-based system resource allocation, intertask communication and control, interrupt driven control for standard I/O devices, and interrupt handling, occupies 2K bytes of memory which can be stored on-board in EPROM. Optional linkable and relocatable modules for console control (CRT or TTY), disk file system, and analog subsystems are provided with the iRMX 80 package. User configurability is aided on the Intellec microcomputer development system by the Interactive Configuration Utility program provided with the iRMX 80 package.

System Development Capability

The development cycle of iSBC 80/10B-based products may be significantly reduced using Intel's system development tools available today. The

Intellec Series II family of compatible microcomputer development systems provides a range of capability from a low cost disk-based edit debug workstation to a high performance, fully compatible hard-disk-based software development system. Also, a unique in-circuit emulator (ICE-80) option provides the capability of developing and debugging software directly on the iSBC 80/10B board.

Programming Capability

PL/M-80 — Intel's high level programming language, PL/M, is also available as a resident Intellec microcomputer development system option. PL/M provides the capability to program in a natural, algorithmic language and eliminates the need to manage register usage or allocate memory. PL/M programs can be written in a much shorter time than assembly language programs for a given application.

FORTRAN-80 — For applications requiring computational and formatted I/O capabilities, the ANSI 77 standard high level FORTRAN-80 programming language is available as a resident option of the Intellec system. The FORTRAN compiler produces relocatable object code that may be easily linked with PL/M or assembly language program modules. In addition, the iSBC 801 FORTRAN-80 run-time package is a complete, ready-to-use set of linkable object modules which are fully compatible with iRMX 80 systems. The modules, when combined with the FORTRAN-80 coded application, provide the appropriate interfaces to the disk file and terminal I/O of iRMX 80, and to the iSBC 310A Math Unit for applications requiring high speed math.

BASIC-80 — A high level language interpreter is available with extended disk capabilities which operates under the iRMX 80 Real-Time Multitasking Executive and translates BASIC-80 source programs into an internally executable form. This language interpreter, provided as a set of linkable object modules, is ideally suited to the OEM who requires a pass through programming language. The BASIC-80 programs may be created, stored, and interpreted on the iSBC 80 based systems using the iSBC 802 BASIC-80 Configurable iRMX 80 Disk-Based Interpreter. The iSBC 802 Interpreter has a complete ready-to-use set of linkable object modules which are fully compatible with Intel's iRMX 80 Real-Time Multitasking Executive Software. The modules provide interfaces to disk file and terminal I/O, software floating point, or interface to other routines provided by the user.

SPECIFICATIONS

Word Size

Instruction — 8, 16, or 24 bits
Data — 8 bits

Cycle Time

Basic Instruction Cycle — 1.95 μsec

Note
Basic instruction cycle is defined as the fastest instruction (i.e., four clock cycles).

Memory Addressing

On-Board ROM/EPROM

0-0FFF using 2708, 2758
0-1FFF using 2716
0-3FFF using 2732

On-Board RAM

3C00-3FFF with no RAM expansion
3000-3FFF with 2114A-5 expansion

Note
All RAM configurations are automatically moved up to a base address of 4XXX when configuring EPROM for 2732.

Memory Capacity

On-Board ROM/EPROM

16K bytes (sockets only)

On-Board RAM

1K byte with user expansion in 1K increments to 4K bytes using Intel 2114A-5 RAMs

Off-Board Expansion

Up to 64K bytes using user specified combinations of RAM, ROM, and EPROM.

I/O Addressing

On-Board Programmable I/O

Device	I/O Address
8255A No. 1	
Port A	E4
Port B	E5
Port C	E6
Control	E7
8255A No. 2	
Port A	E8
Port B	E9
Port C	EA
Control	EB
8251A	
Data	EC
Control	ED
iSBX Multimodule	
MCS0	F0-F7
MCS1	F8-FF

I/O Capacity

Parallel — 48 programmable lines
Serial — 1 transmit, 1 receive
MULTIMODULE — 1 iSBX Bus MULTIMODULE Board

Serial Baud Rates

Frequency (kHz) (Jumper Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous (Program Selectable)	
		÷ 16	÷ 64
307.2	—	19200	4800
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
6.98	6980	—	110
4.8	4800	300	75

Serial Communications Characteristics

Synchronous — 5-8 bit characters; internal or external character synchronization; automatic sync insertion

Asynchronous — 5-8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detectors

Interrupts

Single-level with on-board logic that automatically vectors the processor to location 38H using a restart instruction (RESTART7). Interrupt requests may originate from user specified I/O (2); the programmable peripheral interface (2); the iSBX MULTIMODULE board (2); the programmable communications interface (3); the power fail interrupt (1); or the interval timer (1).

Interfaces

MULTIBUS — All signals TTL compatible

iSBX Bus — All signals TTL compatible

Parallel I/O — All signals TTL compatible

Serial I/O — RS232C or a 20 mil current loop TTY interface (jumper selectable)

Interrupt Requests — All TTL compatible (active-low)

Clocks

System Clock — 2.048 MHz ± 0.1%

Interval Timer — 1.042 msec ± 0.1% (959.5 HZ)

Connectors

Interface	Double-Sided Pins (qty)	Centers (In.)	Mating Connectors
MULTIBUS System	86	0.156	Viking 2KH43/9AMK12 Wire-wrap
iSBX Bus	36	0.1	iSBX 960-5
Parallel I/O (2)	50	0.1	3M 3415-000 Flat
Serial I/O	26	0.1	AMP 87194-6 Flat

Physical Characteristics

Width — 12.00 in. (30.48 cm)

Height — 6.75 in. (17.15 cm)

Depth — 0.05 in. (1.27 cm)

Weight — 14 oz. (484.4 gm)

Electrical Characteristics

DC Power Requirements

Voltage	Without EPROM[1]	With 2708 EPROM[2]	With 2758, 2716, or 2732 EPROM[3]	Power Down Requirements (RAM and Support Circuit)
V_{CC} = +5V ±5%	I^4_{CC} = 2.0A	3.1 A	3.46 A	84 mA + 140 mA/K (2114A-5)
V_{DD} = +12V ±5%	I_{DD} = 150 mA	400 mA	150 mA	Not Required
V_{BB} = −5V ±5%	I_{BB} = 2 mA	200 mA	2 mA	Not Required
V_{AA} = −12V ±5%	I_{AA} = 175 mA	175 mA	175 mA	Not Required

NOTES:

1. Does not include power required for optional ROM/EPROM, I/O drivers, or I/O terminators.
2. With four Intel 2708 EPROMS and 220Ω/330Ω for terminators, installed for 48 input lines. All terminator inputs low.
3. Same as #2 except with four 2758s, 2716s, or 2732s installed.
4. I_{CC} shown without RAM supply current. For 2114A-5 add 140 mA per K byte to a maximum of 560 mA.

Line Drivers and Terminators

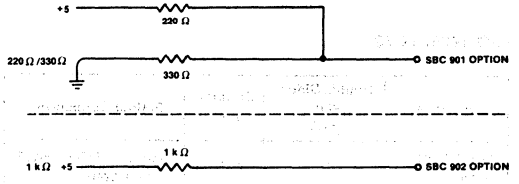
I/O Drivers — The following line drivers and terminators are all compatible with the I/O driver sockets on the iSBC 80/10B Board:

Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note
I - inverting, NI - non-inverting, OC - open collector.

Port 1 has 25 nA totem pole drivers and 1 kΩ terminators.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pull up.



MULTIBUS Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-State	25
Address	Tri-State	25
Commands	Tri-State	25

Environmental Characteristics

Operating Temperature — 0°C to 55°C

Equipment Supplied

iSBC 80/10B Single Board Computer
iSBC 80/10B Schematics

Reference Manual

9803119-01 — iSBC 80/10B Single Board Computer Hardware Reference Manual (NOT SUPPLIED).

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

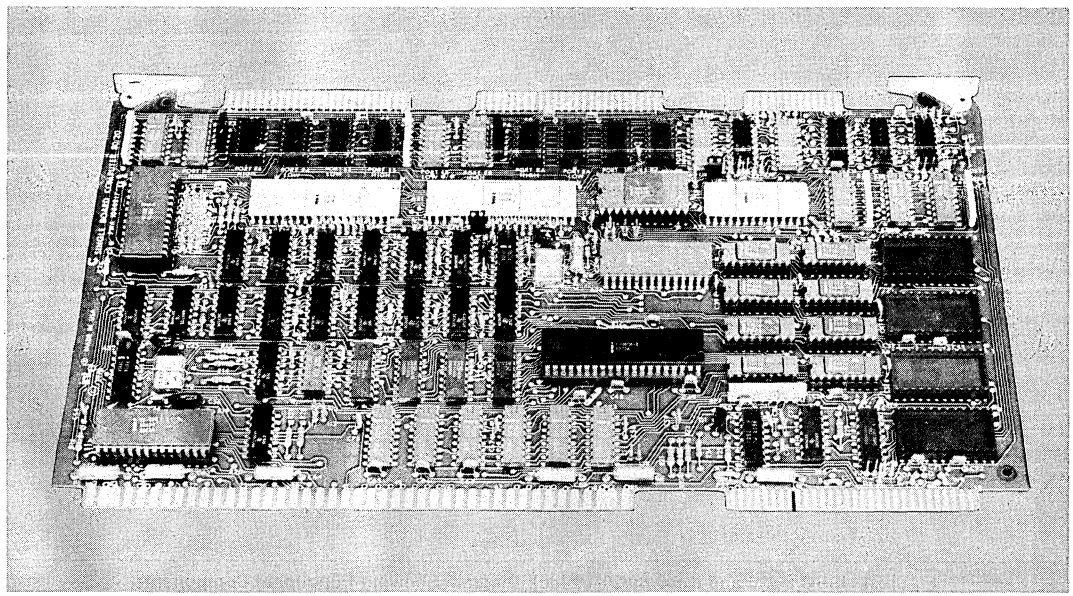
ORDERING INFORMATION

Part Number	Description
SBC 80/10B	Single Board Computer

iSBC 80/20-4 (or pSBC 80/20-4*) SINGLE BOARD COMPUTER

- **8080A CPU used as central processor**
- **4K bytes of static read/write memory**
- **Sockets for up to 8K bytes of erasable reprogrammable or masked read only memory**
- **48 programmable parallel I/O lines with sockets for interchangeable line drivers and terminators**
- **Programmable synchronous/asynchronous RS232C compatible serial interface with fully software selectable baud rate generation**
- **Full MULTIBUS control logic allowing up to 16 masters to share system bus**
- **Two programmable 16-bit BCD and binary timers**
- **Eight-level programmable interrupt control**
- **Compatible with optional memory and I/O expansion boards**
- **Auxiliary power bus, memory protect, and power-fail interrupt control logic provided for battery backup RAM requirements**

The iSBC 80/20-4 Single Board Computer is a member of Intel's complete line of OEM computer systems which take full advantage of Intel's LSI technology to provide economical, self-contained computer-based solutions for OEM applications. Each iSBC 80/20-4 is a complete computer system on a single 6.75 × 12.00-inch printed circuit card. The CPU, system clock, read/write memory, nonvolatile read only memory, I/O ports and drivers, serial communications interface, priority interrupt logic, two programmable timers, MULTIBUS control logic, and bus expansion drivers all reside on each board.



*Same product, manufactured by Intel Puerto Rico, Inc.

FUNCTIONAL DESCRIPTION

Intel's powerful 8-bit n-channel MOS 8080A CPU, fabricated on a single LSI chip, is the central processor for the iSBC 80/20-4. The 8080A contains six 8-bit general purpose registers and an accumulator. The six general purpose registers may be addressed individually or in pairs, providing both single and double precision operators. Minimum instruction execution time is 1.86 microseconds. A block diagram of iSBC 80/20-4 functional components is shown in Figure 1.

Memory Addressing

The 8080A has a 16-bit program counter which allows direct addressing of up to 65,536 bytes of memory. An external stack, located within any portion of read/write memory, may be used as a last-in/first-out storage area for the contents of the program counter, flags, accumulator, and all of the six general purpose registers. A 16-bit stack pointer controls the addressing of this external stack. This stack provides subroutine nesting bounded only by memory size.

Memory Capacity

The iSBC 80/20-4 contains 4K bytes of static read/write memory using Intel low power static RAMs. All on-board RAM read and write operations are performed at maximum processor speed. Power for on-board RAM memory is provided on an auxiliary power bus, and memory protect logic is included for battery backup RAM requirements. Sockets for up to 8K bytes of nonvolatile read only memory are provided on the board. Read only memory may be added in 1K-byte increments using Intel 2708 erasable and electrically reprogrammable ROMs (EPROMs), or read only memory may be added in 2K-byte increments using Intel 2716 EPROMs. All on-board ROM read operations are performed at maximum processor speed.

Parallel I/O Interface

The iSBC 80/20-4 contains 48 programmable parallel I/O lines implemented using two Intel 8255 programmable peripheral interfaces. The system software is used to configure the I/O lines in any combination of the unidirectional input/output, and bidirectional ports indicated in Table 1. Therefore, the I/O interface may be customized to meet specified peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 48 programmable I/O lines and signal ground lines are brought out to two 50-pin edge connectors that mate with flat, woven, or round cable.

Serial I/O Interface

A programmable communications interface using Intel's 8251 Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the iSBC

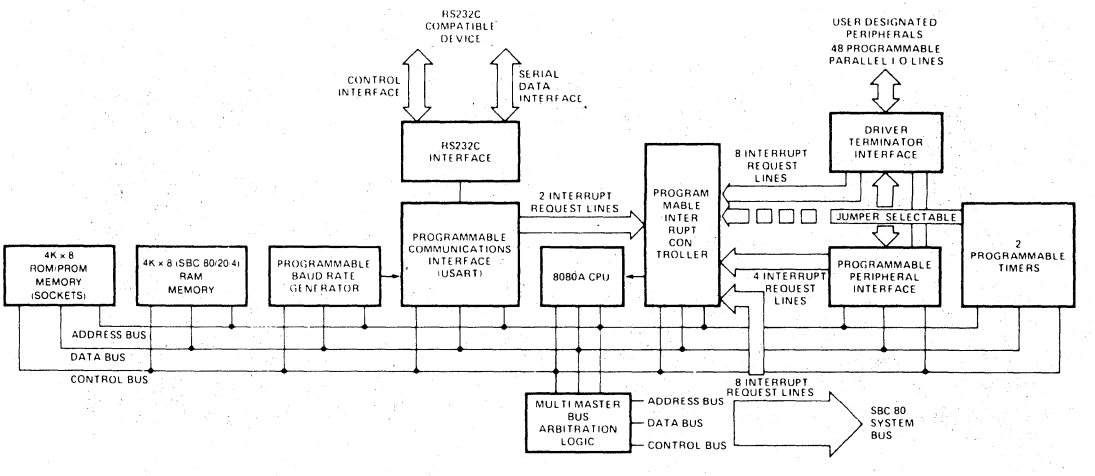


Figure 1. iSBC 80/20 and iSBC 80/20-4 Block Diagram Showing Functional Components

80/20-4 board. A software selectable baud rate generator provides the USART with all common communications frequencies. The USART can be programmed by the system software to select the desired asynchronous or synchronous serial data transmission technique (including IBM Bi-Sync). The mode of operation (i.e., synchronous or asynchronous), data format, control character parity, and baud rate are all under program control. The 8251 provides full duplex, double-buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The RS232C compatible interface on each board, in conjunction with the USART, provides a direct interface to RS232C compatible terminals, cassettes, and asynchronous and synchronous modems. The RS232C command lines, serial data lines, and signal ground line are brought out to a 26-pin edge connector that mates with RS232C compatible flat or round cable.

Multimaster Capability

The iSBC 80/20-4 is a full computer on a single board with resources capable of supporting the majority of OEM system requirements. For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e., several CPUs and/or controllers logically share system tasks with communication over the system bus), the iSBC 80/20-4 provides full MULTIBUS arbitration control logic. This control logic allows up to three iSBC 80/20-4 or high speed controllers to share the system bus in serial (daisy chain) priority fashion, and up to 16 masters may share the system bus with the addition of an external priority network. Once bus control is attained, a bus bandwidth of up to 5M bytes/sec may be achieved.

The bus controller provides its own clock which is derived independently from the processor clock. This allows different speed controllers to share resources on the same bus, and transfers via the bus proceed asynchronously. Thus, transfer speed is dependent on transmitting and receiving devices only. This design prevents slow master modules from being handicapped in their attempts to gain control of the bus, but does not restrict the speed at which faster modules can transfer data via the same bus. Once a bus request is granted, single or multiple read/write transfers can proceed at a maximum rate of 5 million data words per second. The most obvious applications for the master-slave capabilities of the bus are multiprocessor configurations, high speed direct-memory-access (DMA) operations and high speed peripheral control, but are by no means limited to these three.

Programmable Timers

The iSBC 80/20-4 board provides three fully programmable and independent BCD and binary 16-bit interval timers/event counters utilizing an Intel 8253 Programmable Interval Timer. Two of these timers/counters are available to the systems designer to generate accurate time intervals under software control. Routing of these counters is jumper selectable. Each may be independently routed to the programmable interrupt controller, the I/O line drivers and terminators, or outputs from the 8255 programmable peripheral interfaces. The third interval timer in the 8253 provides the programmable baud

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]
4	8	X	X	X	X	X	
5	8	X	X	X	X		
6	4	X		X			X[2]
	4	X		X			X[2]

Notes

1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.

2. Part of port 6 must be used as a control port when either port 4 or port 5 are used as a latched and strobed input or a latched and strobed output port or port 4 is used as a bidirectional port.

rate generator for the iSBC 80/20-4 RS232C USART serial port. In utilizing the iSBC 80/20-4, the systems designer simply configures, via software, each timer independently to meet system requirements. Whenever a given time delay or count is needed, software commands to the programmable timers/event counters select the desired function. Seven functions are available, as shown in Table 2. The contents of each counter may be read at any time during system operation with simple read operations for event counting applications, and special commands are included so that the contents of each counter can be used "on the fly".

Table 2. Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated. This function is extremely useful for generation of real-time clocks.
Programmable one-shot	Output goes low upon receipt of an external trigger edge or software command and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low-going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge on counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counting "window" has been enabled or an interrupt may be generated after N events occur in the system.

Interrupt Capability

Operation and Priority Assignments — An Intel 8259 Programmable Interrupt Controller (PIC) provides vectoring for eight interrupt levels. As shown in Table 3, a selection of four priority processing modes is available to the systems designer so that the manner in which requests are processed may be configured to match system requirements. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from the programmable parallel and serial I/O interfaces, the programmable timers, the system bus, or directly from peripheral equipment. The PIC then determines which of the incoming requests is of the highest priority, determines whether this request is of higher priority than the level currently being serviced, and if appropriate, issues an interrupt to the CPU. Any combination of interrupt levels may be masked through storage via software, of a single byte to the interrupt register of the PIC.

Table 3. Programmable Interrupt Modes

Mode	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until the next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.
Polled	System software examines priority-encoded system interrupt status via interrupt status register.

Interrupt Addressing — The PIC generates a unique memory address for each interrupt level. These addresses are equally spaced at intervals of 4 or 8 (software selectable) bytes. This 32- or 64-byte block may be located to begin at any 32- or 64-byte boundary in the 65,536-byte memory space. A single 8080 jump instruction at each of these addressed then provides linkage to locate each interrupt service routine independently anywhere in memory.

Interrupt Request Generation — Interrupt requests may originate from 26 sources. Four jumper selectable interrupt requets can be automatically generated by the programmable peripheral interface when a byte of information is ready to be transferred to the CPU (i.e., input buffer is full) or a byte of information has been transferred to a peripheral device (i.e., output buffer is empty). Two jumper selectable interrupt requests can be automatically generated by the USART when a character is ready to be transfer to the CPU (i.e., receive channel buffer is full), or a character is ready to be transmitted (i.e., transmit channel data buffer is empty). A jumper selectable request can be generated by each of the programmable timers. Nine additional interrupt request lines are available to the user for direct interface to user designated peripheral devices via the system bus, and eight interrupt request lines may be jumper routed directly from peripherals via the parallel I/O driver/terminator section.

Power-Fail Control — Control logic is also included for generation of a power-fail interrupt which works in conjunction with the AC-low signal from iSBC 635 Power Supply or equivalent.

Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS compatible expansion boards. High speed integer and floating-point arithmetic capabilities may be added by using the iSBC 310A High Speed Mathematics Unit. Memory may be expanded to 65,536 bytes by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be increased by adding digital I/O and analog I/O expansion boards. Mass storage capability may be achieved by adding single or double density diskette controllers as subsystems. Modular expandable backplanes and cardcages are available to support multiboard systems.

Real-Time Software

The iSBC 80/20-4 is totally compatible with Intel's iRMX 80 Real-Time Multi-Tasking Executive. iSBC 80/20-4 based user programs (tasks) can take advantage of the iRMX 80 executive to do all necessary scheduling, intertask communication, and memory space allocation. iRMX 80 also provides standard I/O support software such as disk file handling, Intel analog board handling, and terminal handling.

System Development Capability

The development cycle of iSBC 80/20-4 based products may be significantly reduced using an Intellec microcomputer development system. The resident macroassembler, text editor, and system monitor greatly simplify the design, development, and debug of iSBC 80/20-4 system software. An optional diskette operating system provides a relocating macroassembler, a relocating loader and linkage editor, and a library manager. A unique in-circuit emulator (ICE-80) option provides the capability of developing and debugging software directly on the iSBC 80/20-4 single board computer.

Programming Capability

PL/M-80 — Intel's high level programming language, PL/M, is also available as a resident Intellec microcomputer development system option. PL/M provides the capability to program in a natural, algorithmic language and eliminates the need to manage register usage or allocate memory. PL/M programs can be written in a much shorter time than assembly langauge programs for a given application.

FORTRAN-80 — For applications requiring computational and formatted I/O capabilities, the high level FORTRAN-80 programming language is also available as a resident option of the Intellec system. The FORTRAN compiler produces relocatable object code that may be easily linked with PL/M or assembly language program modules. This gives the user a wide flexibility in developing software.

BASIC-80 — A high level language interpreter with extended disk capabilities which operates under the RMX/80 Real-Time Multi-Tasking Executive and translates BASIC-80 source programs into an internally executable form. This language interpreter, provided as a set of linkable object modules, is ideally suited to the OEM who requires a pass through programming language. The BASIC-80 programs may be created, stored and interpreted on the iSBC 80 based system. The BASIC-80 language has a rich complement of statements, functions, and commands to program applications requiring a full range of 1) string manipulation and disk I/O for data processing, 2) single and double precision floating point and array handling for numeric analysis, or 3) port I/O with mask operations controlled through bit-wise Boolean logical operators.

SPECIFICATIONS

Word Size

Instruction — 8, 16, or 24 bits
Data — 8 bits

Cycle Time

Basic Instruction Cycle — 1.86 μs

Note
Basic instruction cycle is defined as the fastest instruction (i.e., four clock cycles).

Memory Addressing

On-Board ROM/EPROM — 0-0FFF (2708) or 0-1FFF (2716)

On-Board RAM — 4K bytes ending on a 16K boundary (e.g., $3FFF_H$, $7FFF_H$, $BFFF_H$, . . . $FFFF_H$)

Memory Capacity

On-Board ROM/EPROM — 8K bytes (sockets only)

On-Board RAM — 4K bytes

Off-Board Expansion — Up to 65,536 bytes in user specified RAM, ROM, and EPROM

Note
ROM/EPROM may be added in 1K or 2K-byte increments.

I/O Addressing

On-Board Programmable I/O (see Table 1)

Port	8255 No. 1			8255 No. 2			8255 No. 1 Control	8255 No. 2 Control	USART Data	USART Control
	1	2	3	4	5	6				
Address	E4	E5	E6	E8	E9	EA	E7	EB	EC	ED

I/O Capacity

Parallel — 48 programmable lines (see Table 1)

Note
Expansion to 504 input and 504 output lines can be accomplished using optional I/O boards.

Serial Communications Characteristics

Synchronous — 5-8 bit characters; internal or external character synchronization; automatic sync insertion

Asynchronous — 5-8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detection

Baud Rates

Frequency (kHz) (Software Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous	
		÷ 16	÷ 64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
2.4	2400	150	—
1.76	1760	110	—

Note
Frequency selected by I/O write of appropriate 16-bit frequency factor to baud rate register.

Register Address (hex notation, I/O address space)

DE Baud rate register

Note
Baud rate factor (16 bits) is loaded as two sequential output operations to same address (DE_H).

Interrupts

Register Addresses (hex notation, I/O address space)

DA Interrupt request register
DA In-service register
DB Mask register
DA Command register
DB Block address register
DA Status (polling register)

Note
Several registers have the same physical address; sequence of access and one data bit of control word determine which register will respond.

Timers

Register Addresses (hex notation, I/O address space)

DF Control register
DC Timer 1
DD Timer 2

Note
Timer counts loaded as two sequential output operations to same address, as given.

Input Frequencies

Reference	Event Rate
1.0752 MHz ± 10% (0.930 μs period, nominal)	1.1 MHz max

Note
Maximum rate for external events in event counter function.

Output Frequencies/Timing Intervals

Function	Single Timer/Counter		Dual Timer/Counter (Two Timers Cascaded)	
	Min	Max	Min	Max
Real-time interrupt	1.86 μs	60.948 ms	3.72 μs	1.109 hr
Programmable one-shot	1.86 μs	60.948 ms	3.72 μs	1.109 hr
Rate generator	16.407 Hz	537.61 kHz	0.00025 Hz	268.81 kHz
Square-wave rate generator	16.407 Hz	537.61 kHz	0.00025 Hz	268.31 kHz
Software triggered strobe	1.86 μs	60.948 ms	3.72 μs	1.109 hr
Hardware triggered strobe	1.86 μs	60.948 ms	3.72 μs	1.109 hr

Interfaces

Bus — All signals TTL compatible
Parallel I/O — All signals TTL compatible
Interrupt Requests — All TTL compatible
Timer — All signals TTL compatible
Serial I/O — RS232C compatible, data set configuration

System Clock (8080A CPU)

2.1504 MHz ± 0.1%

Auxiliary Power

An auxiliary power bus is provided to allow separate power to RAM for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Memory Protect

An active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences.

Connectors

Interface	Double-Sided Pins (qty)	Centers (in.)	Mating Connectors*
MULTIBUS System Bus	86	0.156	ELFAB BS1562043PBB Viking 2KH43/9AMK12 Soldered PCB Mount EDAC 337086540201 ELFAB BW1562D43PBB EDAC 337086540202 ELFAB BW1562A43PBB Wire Wrap
Auxiliary Bus	60	0.100	EDAC 345060524802 ELFAB BS1020A30PBB EDAC 345060540201 ELFAB BW1020D30PBB Wire Wrap
Parallel I/O (2)	50	0.100	3M 3415-001 Flat Crimp GTE Sylvania 6AD01251A1DD Soldered
Serial I/O	26	0.100	AMP 15837151 EDAC 345026520202 PCB Soldered 3M 3462-0001 AMP 88373-5 Flat Crimp

***Note:** Connectors compatible with those listed may also be used.

Line Drivers and Terminators

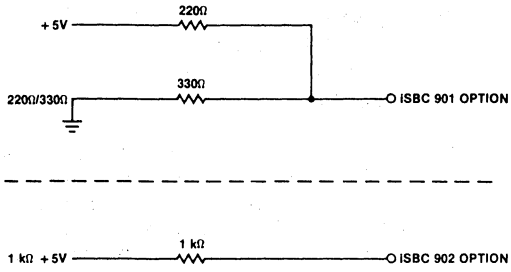
I/O Drivers — The following line drivers are all compatible with the I/O driver sockets on the iSBC 80/20-4.

Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note
I = inverting; NI = non-inverting; OC = open collector.

Ports 1 and 4 have 20 mA totem-pole bidirectional drivers and 1 kΩ terminators.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pull-up



Bus Drivers

Driver	Characteristic	Sink Current (mA)
Data	Tri-state	50
Address	Tri-state	50
Commands	Tri-state	32

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.26 cm)
Weight — 14 oz (397.6 gm)

Electrical Characteristics

DC Power Requirements

Voltage (±5%)	Without PROM[1] (max)	With 4K PROM[2] (max)	With ISBC 530[3] (max)	RAM Only[4] (max)	With 8K PROM[5] (max)
V_{CC} = +5V	I_{CC} = 4.0A	4.9A	4.9A	1.1A	5.2A
V_{DD} = +12V	I_{DD} = 90 mA	350 mA	450 mA	—	90 mA
V_{BB} = −5V	I_{BB} = 2 mA	180 mA	180 mA	—	2 mA
V_{AA} = −12V	I_{AA} = 20 mA	20 mA	120 mA	—	20 mA

Notes

1. Does not include power required for optional PROM, I/O drivers, and I/O terminators.
2. With four 2708 EPROMs and 220Ω/330Ω input terminators installed for 32 I/O lines, all terminator inputs low.
3. With four 2708 EPROMs, 220Ω/330Ω input terminators installed for 32 I/O lines, all terminator inputs low, and iSBC 530 Teletypewriter Adapter drawing power from serial port connector.
4. RAM chips powered via auxiliary power bus.
5. With four 8716 EPROMs and eight 220Ω/330Ω input terminators installed, all terminator inputs low.

Environmental Characteristics

Operating Temperature — 0°C to 55°CC

Reference Manual

9800317D — iSBC 80/20-5 Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

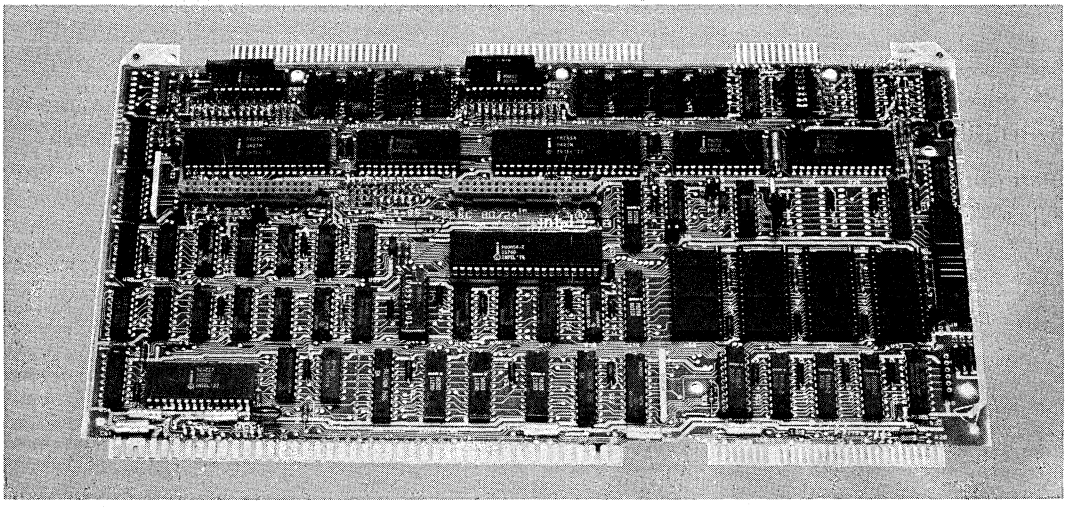
ORDERING INFORMATION

Part Number	Description
SBC 80/20-4	Single Board Computer with 4K bytes RAM

iSBC™ 80/24 (or pSBC 80/24*) SINGLE BOARD COMPUTER

- **Upward compatible with iSBC 80/20-4 Single Board Computer**
- **8085A-2 CPU operating at 4.8 or 2.4 MHz**
- **Two iSBX™ bus connectors for iSBX MULTIMODULE™ board expansion**
- **4K bytes of static read/write memory expandable on-board to 8K bytes using the iSBC 301 MULTIMODULE Board**
- **Sockets for up to 32K bytes of read only memory**
- **48 programmable parallel I/O lines with sockets for interchangeable line drivers and terminators**
- **Programmable synchronous/asynchronous RS232C compatible serial interface with software selectable baud rates**
- **Full MULTIBUS® control logic for multimaster configurations and system expansion**
- **Two programmable 16-bit BCD or binary timers/event counters**
- **12 levels of programmable interrupt control**
- **Auxiliary power bus, memory protect, and power-fail interrupt control logic provided for battery backup RAM requirements**

The Intel® iSBC 80/24 Single Board Computer is a member of Intel's complete line of OEM microcomputer systems which take full advantage of Intel's LSI technology to provide economical, self-contained computer-based solutions for OEM applications. The iSBC 80/24 board is a complete computer system on a single 6.75 × 12.00-inch printed circuit card. The CPU, system clock, iSBX bus interface, read/write memory, read only memory sockets, I/O ports and drivers, serial communications interface, priority interrupt logic, and programmable timers all reside on the board. Full MULTIBUS interface logic is included to offer compatibility with the Intel OEM Microcomputer Systems family of Single Board Computers, expansion memory options, digital and analog I/O expansion boards, and peripheral and communications controllers.



*Same product, manufactured by Intel Puerto Rico, Inc.

The following are trademarks of Intel Corporation and may be used only to describe Intel products: Index, Intel, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, iRMX, UPI, μScope, Promware, MCS, ICE, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1980, 1981

FUNCTIONAL DESCRIPTION

Central Processing Unit

Intel's powerful 8-bit N-channel 8085A-2 CPU fabricated on a single LSI chip, is the central processor for the iSBC 80/24 board operating at either 4.8 or 2.4 MHz (jumper selectable). The 8085A-2 CPU is directly software compatible with the Intel 8080A CPU. The 8085A-2 contains six 8-bit general purpose registers and an accumulator. The six general purpose registers may be addressed individually or in pairs, providing single and double precision operators. Minimum instruction execution time is 826 nanoseconds. A block diagram of the iSBC 80/24 functional components is shown in Figure 1.

MULTIMODULE Board Expansion

The new iSBX bus interface brings an entirely new dimension to system design offering incremental on-board expansion at minimal cost. Two iSBX bus MULTIMODULE connectors are provided for plug-in expansion of any iSBX MULTIMODULE board. The iSBX MULTIMODULE concept provides the ability to adapt quickly to new technology, the economy of buying only what is needed, and the ready availability of a spectrum of functions for greater application potential. iSBX boards are available to provide expansion equivalent to the I/O available on the iSBC 80/24 board or the user may configure entirely new functionality, such as math, directly on board. The iSBX 350 Parallel I/O MULTIMODULE board provides 24 I/O lines using an 8255A Programmable Peripheral Interface. Therefore, two iSBX 350 modules together with the iSBC 80/24 board may offer 96 lines of programmable I/O. Alternately, a serial port may be added using the iSBX 351 Serial I/O MULTIMODULE board and math may be configured on-board with the iSBX 332 Floating Point Math or iSBX 331 Fixed/Floating Point Math MULTIMODULE board. Future iSBX products are also planned. The iSBX MULTIMODULE board is a logical extension of the on-board programmable I/O and is accessed by the iSBC 80/24 single board computer as common I/O port locations. The iSBX board is coupled directly to the 8085A-2 CPU and therefore becomes an integral element of the iSBC 80/24 single board computer providing optimum performance. In addition, RAM memory capacity may be expanded to 8K bytes using the iSBC 301 4K Byte RAM MULTIMODULE board. All MULTIMODULE boards ranging from the iSBC 301 module to the iSBX modules offer incremental expansion, optimum performance, and minimal cost.

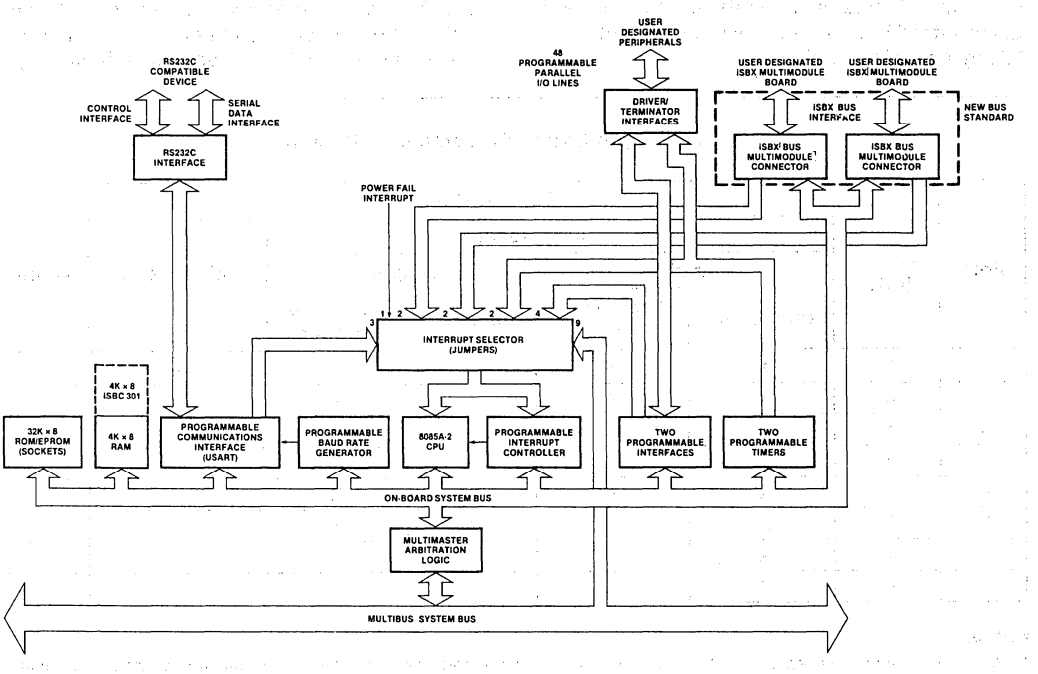


Figure 1. iSBC 80/24 Single Board Computer Block Diagram

AFN-01485A

Memory Addressing

The 8085A-2 has a 16-bit program counter which allows direct addressing of up to 64K bytes of memory. An external stack, located within any portion of read/write memory, may be used as a last-in/first-out storage area for the contents of the program counter, flags, accumulator, and all of the six general purpose registers. A 16-bit stack pointer controls the addressing of this external stack. This stack provides subroutine nesting bounded only by memory size.

Memory Capacity

The iSBC 80/24 board contains 4K bytes of static read/write memory using Intel 8185-2 RAMs. In addition, the on-board RAM capacity may be expanded to 8K bytes with the iSBC 301 4K byte RAM MULTIMODULE board. All RAM read and write operations are performed at maximum processor speed. Power for the on-board RAM may be provided on an auxiliary power bus, and memory protect logic is included for RAM battery backup requirements.

Four sockets are provided for up to 32K bytes of nonvolatile read only memory on the iSBC 80/24 board. EPROM may be added in 1K byte increments up to 4K bytes (using Intel 2708 or 2758); in 2K byte increments up to 8K bytes (using Intel 2716); in 4K byte increments up to 16K bytes (using Intel 2732); or in 8K byte increments up to 32K bytes (using Intel 2764).

Parallel I/O Interface

The iSBC 80/24 board contains 48 programmable parallel I/O lines implemented using two Intel 8255A Programmable Peripheral Interfaces. The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports as indicated in Table 1. Therefore, the I/O interface may be customized to meet specific peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 48 programmable I/O lines and signal ground lines are brought out to two 50-pin edge connectors that mate with flat, woven, or round cables.

Serial I/O Interface

A programmable communications interface using the Intel 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the iSBC 80/24 board. A software selectable baud rate generator provides the USART with all common communication frequencies. The USART can be programmed by the system software to select the desired asynchronous or synchronous serial data transmission technique (including IBM

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					Control
		Unidirectional				Bidirectional	
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]
4	8	X	X	X	X	X	
5	8	X	X	X	X		
6	4	X		X			X[2]
	4	X		X			X[2]

NOTES:

1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.
2. Part of port 6 must be used as a control port when either port 4 or port 5 are used as a latched and strobed input or a latched and strobed output port or port 4 is used as a bidirectional port.

Bi-Sync). The mode of operation (i.e. synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. The 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The RS232C compatible interface, in conjunction with the USART, provides a direct interface to RS232C compatible terminals, cassettes, and asynchronous and synchronous modems. The RS232C command lines, serial data lines, and signal ground line are brought out to a 26-pin edge connector that mates with RS232C compatible flat or round cable.

Multimaster Capability

The iSBC 80/24 board is a full computer on a single board with resources capable of supporting a large variety of OEM system requirements. For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e. several CPUs and/or controllers logically sharing system tasks through communication over the system bus), the iSBC 80/24 board provides full MULTIBUS arbitration control logic. This control logic allows up to three iSBC 80/24 boards or other bus masters to share the system bus in serial (daisy chain) priority fashion, and up to 16 masters to share the MULTIBUS system bus with the addition of an external priority network. The MULTIBUS arbitration logic operates synchronously with a MULTIBUS clock (provided by the iSBC 80/24 board or optionally connected directly to the MULTIBUS clock) while data is transferred via a handshake between the master and slave modules. This allows different speed controllers to share resources on the same bus since transfers via the bus proceed asynchronously. Thus, transfer speed is dependent on transmitting and receiving devices only. This design provides slow master modules from being handicapped in their attempts to gain control of the bus, but does not restrict the speed at which faster modules can transfer data via the same bus. The most obvious applications for the master-slave capabilities of the bus are multiprocessor configurations, high speed direct memory access (DMA) operations, and high speed peripheral control, but are by no means limited to these three.

Programmable Timers

The iSBC 80/24 board provides three independent, fully programmable 16-bit interval timers/event counters utilizing the Intel 8253 Programmable Interval Timer. Each counter is capable of operating in either BCD or binary modes. Two of these timers/counters are available to the systems designer to generate accurate time intervals under software control. Routing for the outputs and gate/trigger inputs of two of these counters is jumper selectable. The outputs may be independently routed to the 8259A Programmable Interrupt Controller, to the I/O line drivers associated with the 8255A Programmable Peripheral Interface, or may be routed as inputs to the 8255A chip. The gate/trigger inputs may be routed to I/O terminators associated with the 8255A or as output connections from the 8255A. The third interval timer in the 8253 provides the programmable baud rate generator for the RS232C USART serial port. In utilizing the iSBC 80/24 board, the systems designer simply configures, via software, each timer independently to meet system require-

Table 2. Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated. This function is extremely useful for generation of real-time clocks.
Programmable one-shot	Output goes low upon receipt of an external trigger edge or software command and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low-going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge on counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counting "window" has been enabled or an interrupt may be generated after N events occur in the system.

ments. Whenever a given time delay or count is needed, software commands to the programmable timers/event counters select the desired function. Seven functions are available, as shown in Table 2. The contents of each counter may be read at any time during system operation with simple read operations for event counting applications, and special commands are included so that the contents of each counter can be read "on the fly".

Interrupt Capability

The iSBC 80/24 board provides vectoring for 12 interrupt levels. Four of these levels are handled directly by the interrupt processing capability of the 8085A-2 CPU and represent the four highest priority interrupts of the iSBC 80/24 board. Requests are routed to the 8085A-2 interrupt inputs—TRAP, RST 7.5, RST 6.5, and RST 5.5 (in decreasing order of priority), each of which generates a call instruction to a unique address (TRAP: 24H; RST 7.5: 3CH; RST 6.5: 34H; and RST 5.5: 2CH). An 8085A-2 JMP instruction at each of these addresses then provides linkage to interrupt service routines located independently anywhere in memory. All interrupt inputs with the exception of the trap interrupt may be masked via software. The trap interrupt should be used for conditions such as power-down sequences which require immediate attention by the 8085A-2 CPU. The Intel 8259A Programmable Interrupt Controller (PIC) provides vectoring for the next eight interrupt levels. As shown in Table 3, a selection of four priority processing modes is available to the systems designer for use in designing request processing configurations to match system requirements. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from the programmable parallel and serial I/O interfaces, the programmable timers, the system bus, iSBX bus, or directly from peripheral equipment. The PIC then determines which of the incoming requests is of the highest priority, determines whether this request is of higher priority than the level currently being serviced, and, if appropriate, issues an interrupt to the CPU. Any combination of interrupt levels may be masked, via software, by storing a single byte in the interrupt mask register of the PIC. The PIC generates a unique memory address for each interrupt level. These addresses are equally spaced at intervals of 4 or 8 (software selectable) bytes. This 32 or 64-byte block may be located to begin at any 32 or 64-byte boundary in the 65,536-byte memory space. A single 8085A-2 JMP instruction at each of these addresses then provides linkage to locate each interrupt service routine independently anywhere in memory.

Table 3. Programmable Interrupt Modes

Mode	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Autorotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.
Polled	System software examines priority-encoded system interrupt status via interrupt status register.

Interrupt Request Generation

Interrupt requests may originate from 23 sources. Two jumper selectable interrupt requests can be generated by each iSBX MULTIMODULE board. Two jumper selectable interrupt requests can be automatically generated by each programmable peripheral interface when a byte of information is ready to be transferred to the CPU (i.e., input buffer is full) or a byte of information has been transferred to a peripheral device (i.e., output buffer is empty). Three jumper selectable interrupt requests can be automatically generated by the USART when a character is ready to be transferred to the CPU (i.e., receiver channel buffer is full), a character is ready to be transmitted (i.e., the USART is ready to accept a character from the CPU), or when the transmitter is empty (i.e., the USART has no character to transmit). A jumper selectable request can be generated by each of the programmable timers. Nine interrupt request lines are available to the user for direct interface to user designated peripheral devices via the MULTIBUS system bus. A power-fail signal can also be selected as an interrupt source.

Power-Fail Control

A power-fail interrupt may be detected through the AC-low signal generated by the power supply. This signal may be configured to interrupt the 8085A-2 CPU to initiate an orderly power down instruction sequence.

MULTIBUS System Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS system compatible expansion boards. Memory may be expanded to 65,536 bytes by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capa-

city may be increased by adding digital I/O and analog I/O expansion boards. Mass storage capability may be achieved by adding single or double density diskette or hard disk controllers as subsystems. Expanded communication needs can be handled by communication controllers. Modular expandable backplanes and card cages are available to support multiboard systems.

Real-Time Software

The iRMX™ 80 executive, which contains all major real-time facilities including priority-based system resource allocation, intertask communication and control, interrupt driven control for standard I/O devices, and interrupt handling, occupies 2K bytes of memory which can be stored on-board in EPROM. Optional linkable and relocatable modules for console control (CRT or TTY), disk file system, and analog subsystems are provided with the iRMX 80 package. These facilities eliminate the need for users to design and implement application specific executives, greatly simplifying application design and reducing development time and risk.

System Development Capability

The development cycle of iSBC 80/24-based products may be significantly reduced using Intel's system development tools available today. The Intellec Series II family of compatible microcomputer development systems provides a range of capability from a low cost disk-based edit debug workstation to a high performance, fully compatible hard-disk-based software development system. Also, a unique in-circuit emulator (ICE-85A) option provides the capability of developing and debugging software directly on the iSBC 80/24 board.

Programming Capability

PL/M-80—Intel's high level system programming language, PL/M, is also available as a resident Intellec microcomputer development system option. PL/M provides the capability to program in a natural, algorithmic language and eliminates the need to manage register usage or allocate memory. PL/M programs can be written in a much shorter time than assembly language programs.

FORTRAN-80—For applications requiring computational and formatted I/O capabilities, the ANSI 77 standard high level FORTRAN-80 programming language is available as a resident option of the Intellec system. The FORTRAN compiler produces relocatable object code that may be easily linked with PL/M or assembly language program modules. In addition, the iSBC 801 FORTRAN-80 Run-Time Package is a complete, ready-to-use set of linkable object modules which are fully compatible with iRMX 80 systems. The modules, when combined with the FORTRAN-80 coded application, provide the appropriate interfaces to the disk file and terminal I/O of iRMX 80, and to the iSBC 310A Math Unit for applications requiring high speed math.

BASIC-80—A high level language interpreter is available with extended disk capabilities which operates under the iRMX 80 Real-Time Multitasking Executive and translates BASIC-80 source programs into an internally executable form. This language interpreter, provided as a set of linkable object modules, is ideally suited to the OEM who requires a pass through programming language. The BASIC-80 programs may be created, stored, and interpreted on the iSBC 80-based systems using the iSBC 802 BASIC-80 Configurable iRMX 80 Disk-Based Interpreter. The iSBC 802 Interpreter has a complete ready-to-use set of linkable object modules which are fully compatible with Intel's iRMX 80 Real-Time Multitasking Executive Software. The modules provide interfaces to disk file and terminal I/O, software floating point, or interface to other routines provided by the user.

SPECIFICATIONS

Word Size

Instruction—8, 16, or 24 bits
Data—8 bits

Cycle Time

Basic Instruction Cycle

826 nsec (4.84 MHz operating frequency)
1.65 μsec (2.42 MHz operating frequency)

NOTE:
Basic instruction cycle is defined as the fastest instruction (i.e., four clock cycles).

Memory Addressing

On-Board EPROM

0-0FFF using 2708, 2758 (1 wait state)
0-1FFF using 2716 (1 wait state)
0-3FFF using 2732 (1 wait state)
using 2732A (no wait states)
0-7FFF using 2764A (no wait states)

On-Board RAM

3000-3FFF with no RAM expansion
2000-3FFF with optional RAM (iSBC 301 board)

NOTE:
Default configuration—may be reconfigured to top end of any 16K boundary.

Memory Capacity

On-Board EPROM

32K bytes (sockets only)

May be added in 1K (using Intel 2708 or 2758), 2K (using Intel 2716), 4K (using Intel 2732), or 8K (using Intel 2764) byte increments.

On-Board RAM

4K bytes (8K bytes using iSBC 301 4K byte RAM MULTIMODULE Board)

Off-Board Expansion

Up to 64K bytes using user specified combinations of RAM, ROM, and EPROM.

Up to 128K bytes using bank select control via I/O port and 2 jumper options.

May be disabled using PROM ENABLE via I/O port and jumper option, resulting in off-board RAM overlay capability.

I/O Addressing

On-Board Programmable I/O

Device	I/O Address
8255A No. 1	
Port A	E4
Port B	E5
Port C	E6
Control	E7
8255A No. 2	
Port A	E8
Port B	E9
Port C	EA
Control	EB
8251A	
Data	EC, EE
Control	ED, EF
iSBX MULTIMODULE J5	
MCS0	C0-C7
MCS1	C8-CF
iSBX MULTIMODULE J6	
MCS0	F0-F7
MCS1	F8-FF

I/O Capacity

Parallel—48 programmable lines

Serial—1 transmit, 1 receive, 1 SID, 1 SOD

iSBX MULTIMODULE — 2 iSBX MULTIMODULE Boards

Serial Communications Characteristics

Synchronous—5-8 bit characters; internal or external character synchronization; automatic sync insertion

Asynchronous—5-8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detectors

Baud Rates

Output Frequency in kHz	Baud Rate (Hz)		
	Synchronous	Asynchronous	
		÷16	÷64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
2.4	2400	150	—
1.76	1760	110	—

NOTE:
Frequency selected by I/O write of appropriate 16-bit frequency factor to baud rate register.

Register Address (hex notation, I/O address space)

DE Baud rate register

NOTE:
Baud rate factor (16 bits) is loaded as two sequential output operations to same address (DE_H).

Interrupts

Addresses for 8259A Registers (hex notation, I/O address space)

DA or D8 Interrupt request register
DA or D8 In-service register
DB or D9 Mask register
DA or D8 Command register
DB or D9 Block address register
DA or D8 Status (polling register)

NOTE:
Several registers have the same physical address; sequence of access and one data bit of control word determine which register will respond.

Interrupt levels routed to 8085A-2 CPU automatically vector the processor to unique memory locations:

Interrupt Input	Memory Address	Priority	Type
TRAP	24	Highest	Non-maskable
RST 7.5	3C	↓	Maskable
RST 6.5	34	↓	Maskable
RST 5.5	2C	Lowest	Maskable

Timers

Register Addresses (hex notation, I/O address space)

DF Control register
DC Timer 0
DD Timer 1
DE Timer 2

NOTE:
Timer counts loaded as two sequential output operations to same address as given.

Input Frequencies

Reference: 1.0752 MHz ±0.1% (0.930 μsec period, nominal)

Event Rate: 1.1 MHz max.

Output Frequencies/Timing Intervals

Function	Single Timer/Counter		Dual Timer/Counter (Two Timers Cascaded)	
	Min.	Max.	Min.	Max.
Real-Time Interrupt	1.86 μsec	60.948 msec	3.72 μsec	1.109 hrs
Programmable One-Shot	1.86 μsec	60.948 msec	3.72 μsec	1.109 hrs
Rate Generator	16.407 Hz	537.61 kHz	0.00025 Hz	268.81 kHz
Square-Wave Rate Generator	16.407 Hz	537.61 kHz	0.00025 Hz	268.81 kHz
Software Triggered Strobe	1.86 μsec	60.948 msec	3.72 μsec	1.109 hrs
Hardware Triggered Strobe	1.86 μsec	60.948 msec	3.72 μsec	1.109 hrs

NOTE:
Input frequency to timers is 1.0752 MHz (default configuration).

Interfaces

MULTIBUS—All signals TTL compatible

iSBX Bus—All signals TTL compatible

Parallel I/O—All signals TTL compatible

Serial I/O—RS232C compatible, configurable as a data set or data terminal

Timer—All signals TTL compatible

Interrupt Requests—All TTL compatible

System Clock (8085A-2 CPU)

4.84 or 2.42 MHz ±0.1% (jumper selectable)

Auxiliary Power

An auxiliary power bus is provided to allow separate power to RAM for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Connectors

Interface	Double-Sided Pins (qty)	Centers (In.)	Mating Connectors*
MULTIBUS System Bus	86	0.156	ELFAB BS1562043PBB Viking 2KH43/9AMK12 Soldered PCB Mount EDAC 337086540201 ELFAB BW1562D43PBB EDAC 337086540202 ELFAB BW1562A43PBB Wire Wrap
Auxiliary Bus	60	0.100	EDAC 345060524802 ELFAB BS1020A30PBB EDAC 345060540201 ELFAB BW1020D30PBB Wire Wrap
iSBX Bus (2)	36	0.100	iSBX 960-5
Parallel I/O (2)	50	0.100	3M 3415-001 Flat Crimp GTE Sylvania 6AD01251A1DD Soldered
Serial I/O	26	0.100	AMP 15837151 EDAC 345026520202 PCB Soldered 3M 3462-0001 AMP 88373-5 Flat Crimp

***Note:** Connectors compatible with those listed may also be used.

Memory Protect

An active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences.

Line Drivers and Terminators

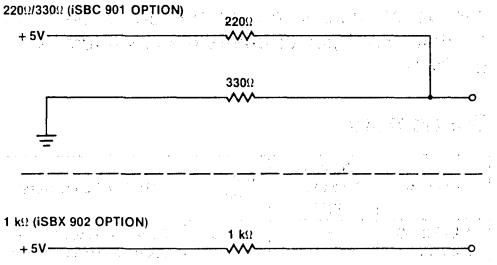
I/O Drivers—The following line drivers and terminators are all compatible with the I/O driver sockets on the iSBC 80/24 Board:

Driver	Characteristic	Sink Current (mA)
7438	I, OC	48
7437	I	48
7432	NI	16
7426	I, OC	16
7409	NI, OC	16
7408	NI	16
7403	I, OC	16
7400	I	16

NOTE:
I = inverting; NI = non-inverting; OC = open collector.

Ports E4 and E8 have 32 mA totem-pole drivers and 1K terminators.

I/O Terminators—220Ω/330Ω divider or 1 kΩ pullup



Bus Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-State	32
Address	Tri-State	32
Commands	Tri-State	32

Physical Characteristics

Width—12.00 in. (30.48 cm)
Height—6.75 in. (17.15 cm)
Depth—0.50 in. (1.27 cm)
Weight—12.64 oz. (354 gm)

Electrical Characteristics

DC Power Requirements

Configuration	Current Requirements			
	V_{CC} = +5V ±5% (max)	V_{DD} = +12V ±5% (max)	V_{BB} = −5V ±5% (max)	V_{AA} = −12V ±5% (max)
Without EPROM[1]	3.34A	40 mA	—	20 mA
RAM Only[2]	0.14A	—	—	—
With iSBC 530[3]	3.34A	140 mA	—	120 mA
With 4K EPROM[4] (using 2708)	3.74A	300 mA	180 mA	20 mA
With 4K EPROM[4] (using 2758)	4.43A	40 mA	—	20 mA
With 8K EPROM[4] (using 2716)	4.43A	40 mA	—	20 mA
With 16K EPROM[4] (using 2732)	4.71A	40 mA	—	20 mA
With 32K EPROM[4] (using 2764)	4.71A	40 mA	—	20 mA

NOTES:
1. Does not include power for optional EPROM, I/O drivers, and I/O terminators.
2. RAM chips powered via auxiliary power bus.
3. Does not include power for optional EPROM, I/O drivers, and I/O terminators. Power for iSBC 530 Adapter is supplied via serial port connector.
4. Includes power required for four EPROM chips, and I/O terminators installed for 16 I/O lines; all terminator inputs low.

Environmental Characteristics

Operating Temperature—0°C to 55°C

Reference Manual

142648-001—iSBC 80/24 Single Board Computer Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

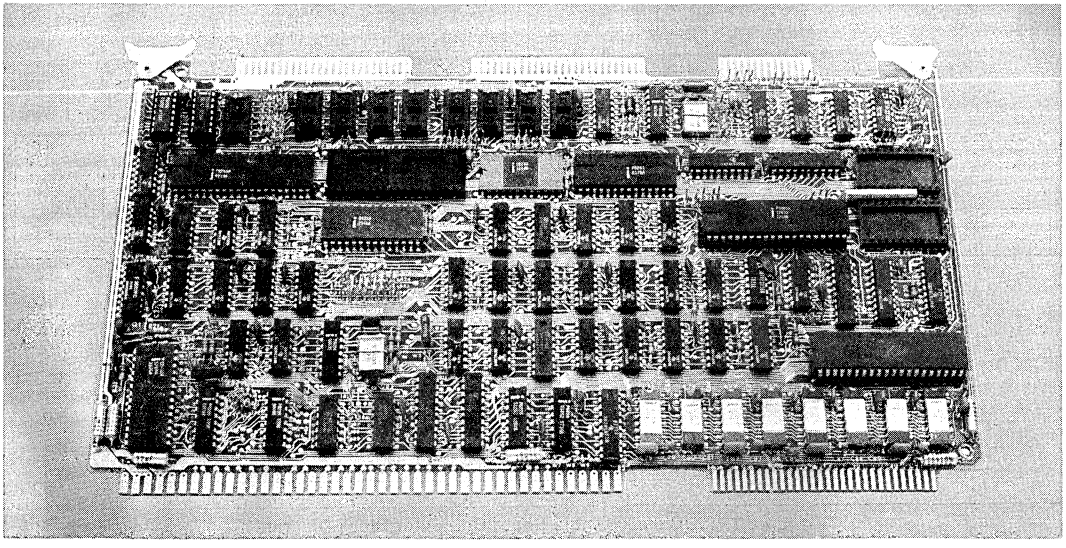
ORDERING INFORMATION

Part Number	Description
SBC 80/24	Single Board Computer

iSBC™ 80/30 (or pSBC 80/30*) SINGLE BOARD COMPUTER

- 8085A CPU used as central processing unit
- 16K bytes of dual port dynamic read/write memory with on-board refresh
- Sockets for up to 8K bytes of read only memory
- Sockets for 8041A/8741A Universal Peripheral Interface and interchangeable line drivers and line terminators
- 24 programmable parallel I/O lines with sockets for interchangeable line drivers and terminators
- Programmable synchronous/asynchronous RS232C compatible serial interface with fully software selectable baud rate generation
- Full MULTIBUS® control logic allowing up to 16 masters to share the system
- 12 levels of programmable interrupt control
- Two programmable 16-bit BCD or binary counters
- Auxiliary power bus, memory protect, and power-fail interrupt control logic for RAM battery backup
- Compatible with optional iSBC 80 CPU, memory, and I/O expansion boards

The iSBC 80/30 Single Board Computer is a member of Intel's complete line of OEM computer systems which take full advantage of Intel's LSI technology to provide economical self-contained computer-based solutions for OEM applications. The iSBC 80/30 is a complete computer system on a single 6.75 × 12.00-inch printed circuit card. The CPU, system clock, read/write memory, nonvolatile read only memory, universal peripheral interface capability, I/O ports and drivers, serial communications interface, priority interrupt logic, programmable timers, MULTIBUS control logic, and bus expansion drivers all reside on the board.



*Same product, manufactured by Intel Puerto Rico, Inc.

FUNCTIONAL DESCRIPTION

Central Processing Unit

Intel's powerful 8-bit n-channel 8085A CPU, fabricated on a single LSI chip, is the central processor for the iSBC 80/30. The 8085A CPU is directly software compatible with the Intel 8080A CPU. The 8085A contains six 8-bit general purpose registers and an accumulator. The six general purpose registers may be addressed individually or in pairs, providing both single and double precision operators. The minimum instruction execution time is 1.45 microseconds. The 8085A CPU has a 16-bit program counter. An external stack, located within any portion of iSBC 80/30 read/write memory, may be used as a last-in/first-out storage area for the contents of the program counter, flags, accumulator, and all of the six general purpose registers. A 16-bit stack pointer controls the addressing of this external stack. This stack provides subroutine nesting bounded only by memory size.

Bus Structure

The iSBC 80/30 has an internal bus for all on-board memory and I/O operations and a system bus (i.e., the MULTIBUS) for all external memory and I/O operations. Hence, local (on-board) operations do not tie up the system bus, and allow true parallel processing when several bus masters (i.e., DMA devices, other single board computers) are used in a multimaster scheme. A block diagram of the iSBC 80/30 functional components is shown in Figure 1.

RAM Capacity

The iSBC 80/30 contains 16K bytes of dynamic read/write memory using Intel 2117 RAMs. All RAM read and write operations are performed at maximum processor speed. Power for the on-board RAM may be provided on an auxiliary power bus, and memory protect logic is included for RAM battery backup requirements. The iSBC 80/30 contains a dual port controller, which provides dual port capability for the on-board RAM memory. RAM accesses may occur from either the iSBC 80/30 or from any other bus master interfaced via the

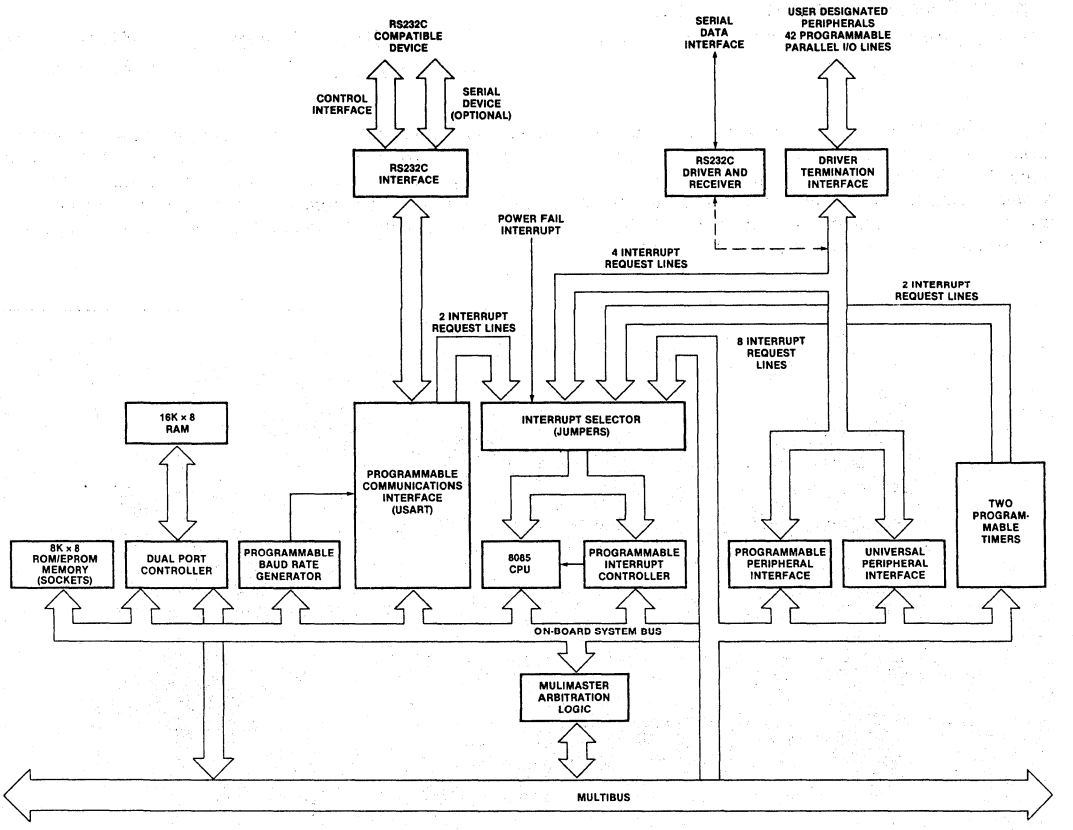


Figure 1. iSBC 80/30 Single Board Computer Block Diagram

MULTIBUS. Since on-board RAM accesses do not require the MULTIBUS, the bus is available for any other concurrent operations (e.g., DMA data transfers) requiring the use of the MULTIBUS. Dynamic RAM refresh is accomplished automatically by the iSBC 80/30 for accesses originating from either the CPU or via the MULTIBUS. Memory space assignment can be selected independently for on-board and MULTIBUS RAM accesses. The on-board RAM, as seen by the 8085A CPU, may be placed anywhere within the 0- to 64K-address space. The iSBC 80/30 provides extended addressing jumpers to allow the on-board RAM to reside within a one megabyte address space when accessed via the MULTIBUS. In addition, jumper options are provided which allow the user to reserve 8K- and 16K-byte segments of on-board RAM for use by the 8085A CPU only. This reserved RAM space is not accessible via the MULTIBUS and does not occupy any system address space.

EPROM/ROM Capacity

Sockets for up to 8K bytes of nonvolatile read only memory are provided on the iSBC 80/30 board. Read only memory may be added in 1K-byte increments up to a maximum of 2K bytes using Intel 2708 or 2758 erasable and electrically reprogrammable ROMs (EPROMs); in 2K-byte increments up to a maximum of 4K bytes using Intel 2716 EPROMs; or in 4K-byte increments up to 8K bytes maximum using Intel 2732 EPROMs. All on-board EPROM/ROM operations are performed at maximum processor speed.

Parallel I/O Interface

The iSBC 80/30 contains 24 programmable parallel I/O lines implemented using the Intel 8255A Programmable Peripheral Interface. The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports indicated in Table 1. Therefore, the I/O interface may be customized to meet specific peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 24 programmable I/O lines and signal ground lines are brought out to a 50-pin edge connector that mates with flat, woven, or round cable.

Universal Peripheral Interface (UPI)

The iSBC 80/30 provides sockets for a user supplied Intel 8041A/8741A Universal Peripheral Interface (UPI) chip and the associated line drivers and terminators for the UPI's I/O ports. The 8041A/8741A is a single chip microcomputer containing a CPU, 1K bytes of ROM (8041A) or EPROM (8741A), 64 bytes of RAM, 18 programmable I/O lines, and an 8-bit timer. Special interface registers included in the chip allow the 8041A to function as a slave processor to the iSBC 80/30's 8085A CPU. The UPI allows the user to specify algorithms for controlling user peripherals directly in the chip, thereby relieving the 8085A for other system functions. The iSBC 80/30 provides an RS232C driver and an RS232C receiver for optional connection to the 8041A/8741A in applications where the UPI is programmed to handle simple serial interfaces. For additional information, including 8041A/8741A instructions, refer to the UPI-41A User's Manual and application note AP-41.

Serial I/O

A programmable communications interface using the Intel 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the iSBC 80/30. A software selectable baud rate generator provides the USART with all common communication frequencies. The USART can be programmed by the system software to select the desired asynchronous or synchronous serial data transmission technique (including IBM By-Sync). The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. The 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]

Note

1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.

USART. The RS232C compatible interface on each board, in conjunction with the USART, provides a direct interface to RS232C compatible terminals, cassettes, and asynchronous and synchronous modems. The RS232C command lines, serial data lines, and signal ground line are brought out to a 26-pin edge connector that mates with RS232C compatible flat or round cable.

Multimaster Capability

The iSBC 80/30 is a full computer on a single board with resources capable of supporting a great variety of OEM system requirements. For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e., several CPUs and/or controllers logically sharing system tasks through communication over the system bus), the iSBC 80/30 provides full MULTIBUS arbitration control logic. This control logic allows up to three iSBC 80/30's or other bus masters to share the system bus in serial (daisy chain) priority fashion, and up to 16 masters to share the MULTIBUS with the addition of an external priority network. The MULTIBUS arbitration logic operates synchronously with a MULTIBUS clock (provided by the iSBC 80/30 or optionally connected directly to the MULTIBUS clock) while data is transferred via a handshake between the master and slave modules. This allows different speed controllers to share resources on the same bus, and transfers via the bus proceed asynchronously. Thus, transfer speed is dependent on transmitting and receiving devices only. This design prevents slow master modules from being handicapped in their attempts to gain control of the bus, but does not restrict the speed at which faster modules can transfer data via the same bus. The most obvious applications for the master-slave capabilities of the bus are multiprocessor configurations, high speed direct memory access (DMA) operations, and high speed peripheral control, but are by no means limited to these three.

Programmable Timers

The iSBC 80/30 provides three independent, fully programmable 16-bit interval timers/event counters utilizing the Intel 8253 Programmable Interval Timer. Each counter is capable of operating in either BCD or binary modes. Two of these timers/counters are available to the systems designer to generate accurate time intervals under software control. Routing for the outputs and gate/trigger inputs of two of these counters is jumper selectable. The outputs may be independently routed to the 8259A Programmable Interrupt Controller, to the I/O line drivers associated with the 8255A Programmable Peripheral Interface, and to the 8041A/8741A Universal Programmable Interface, or may be routed as inputs to the 8255A and 8041A/8741A chips. The gate/trigger inputs may be routed to I/O terminators associated with the 8255A or as output connections from the 8255A. The third interval timer in the 8253 provides the programmable baud rate generator for the iSBC 80/30 RS232C USART serial port. In utilizing the iSBC 80/30, the systems designer simply configures, via software, each timer independently to meet system requirements. Whenever a given time delay or count is needed, software commands to the programmable timers/event counters select the desired function. Seven functions are available, as shown in Table 2. The contents of each counter may be read at any time during system operation with simple read operations for event counting applications, and special commands are included so that the contents of each counter can be read "on the fly".

Table 2. Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated. This function is extremely useful for generation of real-time clocks.
Programmable one-shot	Output goes low upon receipt of an external trigger edge or software command and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low-going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge on counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counting "window" has been enabled or an interrupt may be generated after N events occur in the system.

Interrupt Capability

The iSBC 80/30 provides vectoring for 12 interrupt levels. Four of these levels are handled directly by the interrupt processing capability of the 8085A CPU and represent the four highest priority interrupts of the iSBC 80/30. Requests are routed to the 8085A interrupt inputs, TRAP, RST 7.5, RST 6.5, and RST 5.5 (in decreasing order of priority) and each input generates a unique memory address (TRAP: 24H; RST 7.5: 3CH; RST 6.5: 34H; and RST 5.5: 2CH). An 8085A jump instruction at each of these addresses then provides linkage to interrupt ser-

vice routines located independently anywhere in memory. All interrupt inputs with the exception of the trap interrupt may be masked via software. The trap interrupt should be used for conditions such as power-down sequences which require immediate attention by the 8085A CPU. The Intel 8259A Programmable Interrupt Controller (PIC) provides vectoring for the next eight interrupt levels. As shown in Table 3, a selection of four priority processing modes is available to the systems designer for use in designing request processing configurations to match system requirements. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from the programmable parallel and serial I/O interfaces, the programmable timers, the system bus, or directly from peripheral equipment. The PIC then determines which of the incoming requests is of the highest priority, determines whether this request is of higher priority than the level currently being serviced, and, if appropriate, issues an interrupt to the CPU. Any combination of interrupt levels may be masked, via software, by storing a single byte in the interrupt mask register of the PIC. The PIC generates a unique memory address for each interrupt level. These addresses are equally spaced at intervals of 4 or 8 (software selectable) bytes. This 32- or 64-byte block may be located to begin at any 32- or 64-byte boundary in the 65,536-byte memory space. A single 8085A jump instruction at each of these addresses then provides linkage to locate each interrupt service routine independently anywhere in memory.

Table 3. Programmable Interrupt Modes

Mode	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.
Polled	System software examines priority-encoded system interrupt status via interrupt status register.

Interrupt Request Generation — Interrupt requests may originate from 18 sources. Two jumper selectable interrupt requests can be automatically generated by the programmable peripheral interface when a byte of information is ready to be transferred to the CPU (i.e., input buffer is full) or a byte of information has been transferred to a peripheral device (i.e., output buffer is empty). Two jumper selectable interrupt requests can be automatically generated by the USART when a character is ready to be transferred to the CPU (i.e., receive channel buffer is full), or a character is ready to be transmitted (i.e., transmit channel data buffer is empty). A jumper selectable request can be generated by each of the programmable timers and by the universal peripheral interface, eight additional interrupt request lines are available to the user for direct interface to user designated peripheral devices via the system bus, and two interrupt request lines may be jumper routed directly from peripherals via the parallel I/O driver/terminator section.

Power-Fail Control

Control logic is also included to accept a power-fail interrupt in conjunction with the AC-low signal from the iSBC 635 Power Supply or equivalent.

Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added by using Intel MULTIBUS compatible expansion boards. High speed integer and floating point arithmetic capabilities may be added by using the iSBC 310A High Speed Mathematics Unit. Memory may be expanded to 65,536 bytes by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be increased by adding digital I/O and analog I/O expansion boards. Mass storage capability may be achieved by adding single or double density diskette controllers as subsystems. Modular expandable backplanes and cardcages are available to support multi-board systems.

Real-Time Software

Intel's iRMX 80 Real-Time Multi-Tasking Executive software, specifically designed for Intel iSBC 80 single board computers, provides the capability to monitor and control multiple asynchronous external events. The iRMX 80 executive, which synchronizes and controls the execution of multiple tasks, is provided as a linkable and relocatable module requiring only 2K bytes of memory space. Optional linkable and relocatable modules for teletypewriter and CRT control, diskette file system, high speed math unit, and analog subsystems are also available.

System Development Capability

The development cycle of iSBC 80/30-based products may be significantly reduced using the Intellec series microcomputer development systems. The resident macroassembler, text editor, and system monitor greatly simplify the design, development, and debug of iSBC 80/30 system software. An optional diskette operating system provides a relocating macroassembler, relocating loader and linkage editor, and a library manager. A unique in-circuit emulator (ICE-85) option provides the capability of developing and debugging software directly on the iSBC 80/30.

Programming Capability

PL/M-80 — Intel's high level programming language, PL/M, is also available as a resident Intellec microcomputer development system option. PL/M provides the capability to program in a natural, algorithmic language and eliminates the need to manage register usage or allocate memory. PL/M programs can be written in a much shorter time than assembly language programs for a given application.

FORTRAN-80 — For applications requiring computational and formatted I/O capabilities, the high level FORTRAN-80 programming language is also available as a resident option of the intellec system. FORTRAN-80 meets and exceeds the ANS FORTRAN 77 subset language specification. The FORTRAN-80 compiler produces relocatable object code that may be easily linked with other FORTRAN-80, PL/M, or assembly language program modules. This gives the user wide flexibility in developing software by using the best software tool for a particular functional module within the user's application.

BASIC-80 — A high level language interpreter with extended disk capabilities which operates under the iRMX 80 Real-Time Multi-tasking Executive and translates BASIC-80 source programs into an internally executable form. This language interpreter, provided as a set of linkable object modules, is ideally suited to the OEM who requires a pass thru programming language. The BASIC-80 programs may be created, stored and interpreted on the iSBC 80-based system. The BASIC-80 language has a rich complement of statements, functions, and commands to program applications requiring a full range of 1) string manipulation and disk I/O for data processing, 2) single and double precision floating point and array handling for numeric analysis, or 3) port I/O with mask operations controlled through bit-wise Boolean logical operators.

SPECIFICATIONS

Word Size

Instruction — 8, 16, or 24 bits
Data — 8 bits

Cycle Time

Basic Instruction Cycle — 1.45 μs

Note
Basic instruction cycle is defined as the fastest instruction (i.e., four clock cycles).

Memory Addressing

On-Board ROM/EPROM — 0-07FF (using 2708 or 2758 EPROMs); 0-0FFF (using 2716 EPROMs); 0-1FFF (using 2716 EPROMs; 0-1FFF (using 2732 EPROMs).

On-Board RAM — 16K bytes of dual port RAM starting on a 16K boundary. One or two 8K-byte segments may be reserved for CPU use only.

Memory Capacity

On-Board Read Only Memory — 8K bytes (sockets only)

On-Board RAM — 16K bytes

Off-Board Expansion — Up to 65,536 bytes in user specified combinations of RAM, ROM, and EPROM

Note
Read only memory may be added in 1K, 2K, or 4K-byte increments.

I/O Addressing

On-Board Programmable I/O (see Table 1)

Port	8255A				8041A/8741A		USART	
	1	2	3	Control	Data	Control	Data	Control
Address	E8	E9	EA	EB	E4 or E6	E5 or E7	EC	ED

I/O Capacity

Parallel — 42 programmable lines using one 8255A (24 I/O lines) and an optional 8041A/8741A (18 I/O lines)
Serial — 2 programmable lines using one 8251A and an optional 8041A/8741A programmed for serial operation

Note:
For additional information on the 8041A/8741A refer to the UPI-41 User's Manual (Publication 9800504).

Serial Communications Characteristics

Synchronous — 5—8 bit characters; internal or external character synchronization; automatic sync insertion.

Asynchronous — 5—8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detection.

Baud Rates

Frequency (kHz) (Software Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous	
		÷ 16	÷ 64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
2.4	2400	150	—
1.76	1760	110	—

Note
Frequency selected by I/O write of appropriate 16-bit frequency factor to baud rate register (8253 Timer 2).

Interrupts

Addresses for 8259A Registers (Hex notation, I/O address space)

DA Interrupt request register
DA In-service register
DB Mask register
DA Command register
DB Block address register
DA Status (polling register)

Note
Several registers have the same physical address; sequence of access and one data bit of control word determine which register will respond.

Interrupt Levels routed to 8085A CPU automatically vector the processor to unique memory locations:

Interrupt Input	Memory Address	Priority	Type
TRAP	24	Highest	Non-maskable
RST 7.5	3C	↑	Maskable
RST 6.5	34	↓	Maskable
RST 5.5	2C	Lowest	Maskable

Timers

Register Addresses (Hex notation, I/O address space)

DF Control register

DC Timer 0

DD Timer 1

DE Timer 2

Note
Timer counts loaded as two sequential output operations to same address, as given.

Input Frequencies

Reference: 2.46 MHz ±0.1% (0.041 μs period, nominal); 1.23 MHz ± 0.1%(0.81 μs period, nominal); or 153.60 kHz ±0.1%(6.51 μs period nominal).

Note
Above frequencies are user selectable

Event Rate: 2.46 MHz max

Note
Maximum rate for external events in event counter function.

Output Frequencies/Timing Intervals

Function	Single Timer/Counter		Dual Timer/Counter (Two Timers Cascaded)	
	Min	Max	Min	Max
Real-time interrupt	1.63 μs	427.1 ms	3.26 μs	466.50 min
Programmable one-shot	1.63 μs	427.1 ms	3.26 μs	466.50 min
Rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Square-wave rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Software triggered strobe	1.63 μs	427.1 ms	3.26 μs	466.50 min
Hardware triggered strobe	1.63 μs	427.1 ms	3.26 μs	466.50 min

Interfaces

MULTIBUS — All signals TTL compatible
Parallel I/O — All signals TTL compatible
Interrupt Requests — All TTL compatible
Timer — All signals TTL compatible
Serial I/O — RS232C compatible, data set configuration

System Clock (8085A CPU)

2.76 MHz ±0.1%

Auxiliary Power

An auxiliary power bus is provided to allow separate power to RAM for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Connectors

Interface	Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	Viking 2KH43/9AMK12
Parallel I/O	50	0.1	3M 3415-000
Serial I/O	26	0.1	3M 3462-000

Memory Protect

An active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences.

Line Drivers and Terminators

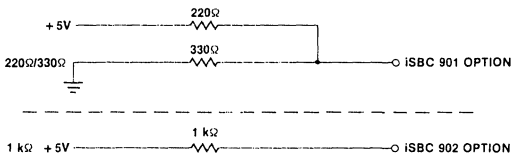
I/O Drivers— The following line drivers are all compatible with the I/O driver sockets on the iSBC 80/30.

Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note
I = inverting; NI = non-inverting; OC = open collector.

Port 1 of the 8255A has 20 mA totem-pole bidirectional drivers and 1 kΩ terminators.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pullup



Bus Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-state	50
Address	Tri-state	50
Commands	Tri-state	32

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 18 oz. (509.6 gm)

Electrical Characteristics

DC Power Requirements

Configuration	Current Requirements			
	V_{CC} = +5V ±5%(max)	V_{DD} = +12V ±5%(max)	V_{BB} = −5V ±5%(max)	V_{AA} = −12V ±5%(max)
Without EPROM[1]	I_{CC} = 3.5A	I_{DD} = 220 mA	I_{BB} = —	I_{AA} = 50 mA
With 8041/8741[2]	3.6A	220 mA	—	50 mA
RAM only[3]	350 mA	20 mA	2.5 mA	—
With iSBC 530[4]	3.5A	320 mA	—	150 mA
With 2K EPROM[5] (using 8708)	4.4A	350 mA	95 mA	40 mA
With 2K EPROM[5] (using 2758)	4.6A	220 mA	—	50 mA
With 4K EPROM[5] (using 2716)	4.6A	220 mA	—	50 mA
With 8K EPROM[5] (using 2332)	4.6A	220 mA	—	50 mA

Notes

1. Does not include power required for optional EPROM/ROM, 8041A/8741A I/O drivers, and I/O terminators.
2. Does not include power required for optional EPROM/ROM, I/O drivers and I/O terminators.
3. RAM chips powered via auxiliary power bus
4. Does not include power required for optional EPROM/ROM, 8041A/8741A I/O drivers, and I/O terminators. Power for iSBC 530 is supplied through the serial port connector.
5. Includes power required for two EPROM/ROM chips, 8041A/8741A and 220Ω/330Ω input terminators installed for 34 I/O lines; all terminator inputs low.

Environmental Characteristics

Operating Temperature — 0°C to 55°C

Reference Manual

9800611B — iSBC 80/30 Single Board Computer Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

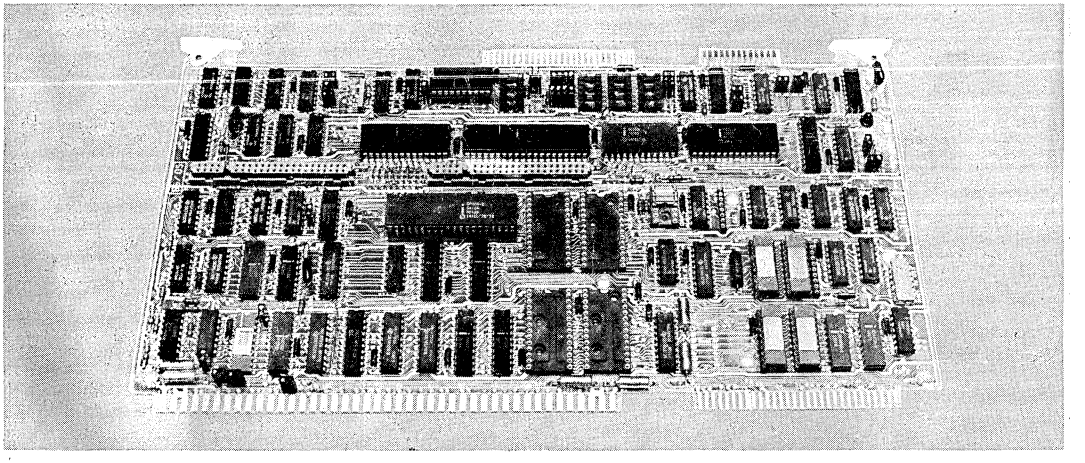
ORDERING INFORMATION

Part Number	Description
SBC 80/30	Single Board Computer with 16K bytes RAM

iSBC 86/05™
SINGLE BOARD COMPUTER

- iAPX 86/10 (8086-2) Microprocessor with 5 or 8 MHz CPU clock
- Fully software compatible with iSBC 86/12A Single Board Computer
- Optional iAPX 86/20 Numeric Data Processor with iSBC 337 MULTIMODULE Processor
- 8K bytes of static RAM; expandable on-board to 16K bytes
- Sockets for up to 64K bytes of JEDEC 24/28-pin standard memory devices; expandable on-board to 128K bytes
- Two iSBX™ bus connectors
- 24 programmable parallel I/O lines
- Programmable synchronous/asynchronous RS232C compatible serial interface with software selectable baud rates
- Two programmable 16-bit BCD or binary timers/event counters
- 9 Levels of vectored interrupt control, expandable to 65 levels
- MULTIBUS interface for multimaster configurations and system expansion
- Supported by a complete family of single board computers, memory, digital and analog I/O, peripheral controllers, packaging and software

The iSBC 86/05 Single Board Computer is a member of Intel's complete line of OEM microcomputer systems which take full advantage of Intel's technology to provide economical, self-contained, computer-based solutions for OEM applications. The iSBC 86/05 board is a complete computer system on a single 6.75 x 12.00-in. printed circuit card. The CPU, system clock, read/write memory, nonvolatile read only memory, I/O ports and drivers, serial communications interface, priority interrupt logic and programmable timers, all reside on the board. The large control storage capacity makes the iSBC 86/05 board ideally suited for control-oriented applications such as process control, instrumentation, industrial automation, and many others.



FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processor for the iSBC 86/05 board is Intel's iAPX 86/10 (8086-2) CPU. A clock rate of 8 MHz is supported with a jumper selectable option of 5 MHz. The CPU architecture includes four 16-bit byte addressable data registers, two 16-bit memory base pointer registers and two 16-bit index registers, all accessed by a total of 24 operand addressing modes for comprehensive memory addressing and for support of the data structures required for today's structured, high level languages as well as assembly language.

Instruction Set

The 8086 instruction repertoire includes variable length instruction format (including double operand instructions), 8-bit and 16-bit signed and unsigned arithmetic operators for binary, BCD and unpacked ASCII data, and iterative word and byte string manipulation functions.

For enhanced numerics processing capability, the iSBC 337 MULTIMODULE Numeric Data Processor extends the iAPX 86/10 architecture and data set. Over 60 numeric instructions offer arithmetic, trigonometric, transcendental, logarithmic and exponential instructions. Supported data types include 16, 32, and 64-bit integer, and 32 and 64-bit floating point, 18-digit packed BCD and 80-bit temporary.

Architectural Features

A 6-byte instruction queue provides pre-fetching of sequential instructions and can reduce the 750 nsec minimum instruction cycle to 250 nsec for queued instructions. The stack-oriented architecture readily supports modular programming by facilitating fast, simple, inter-module communication, and other programming constructs needed for asynchronous real-time systems. The memory expansion capabilities offer a 1 megabyte addressing range. The dynamic relocation scheme allows ease in segmentation of pure procedure and data for efficient memory utilization. Four segment registers (code, stack, data, extra) contain program loaded offset values which are used to map 16-bit addresses to 20-bit addresses. Each register maps 64K bytes at a time and activation of a specific register is controlled explicitly by program control and is also selected implicitly by specific functions and instructions. All Intel languages support the extended memory capability, relieving the programmer of managing the megabyte memory space, yet allowing explicit control when necessary.

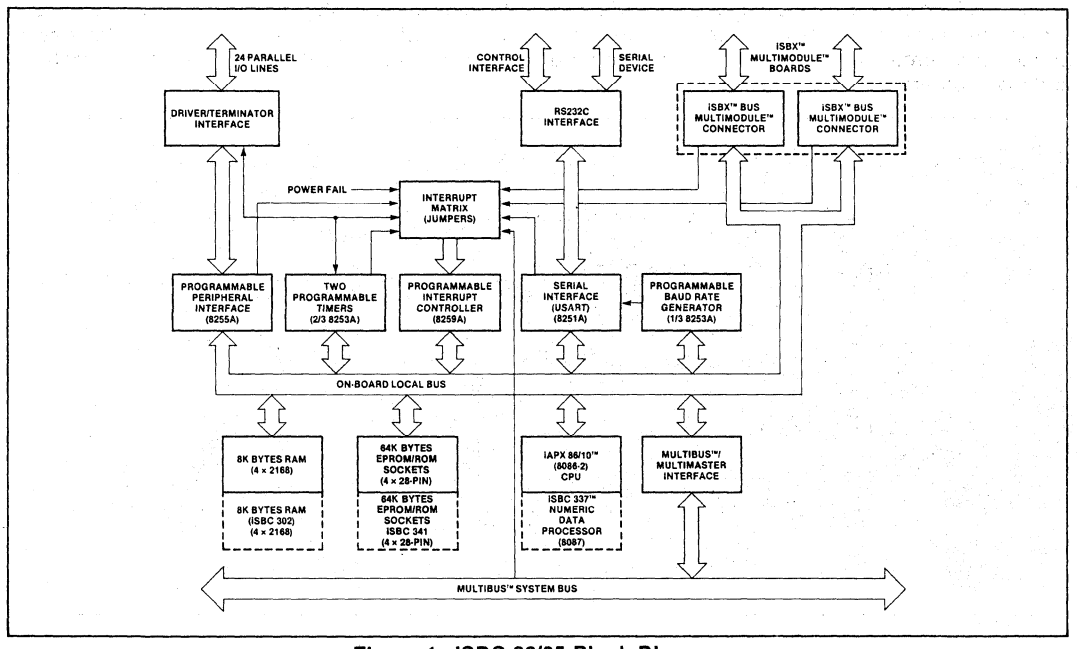


Figure 1. iSBC 86/05 Block Diagram

Memory Configuration

The iSBC 86/05 microcomputer contains 8K bytes of high-speed static RAM on-board. In addition, the on-board RAM may be expanded to 16K bytes with the iSBC 302 MULTIMODULE RAM option which mounts on the iSBC 86/05 board. All on-board RAM is accessed by the 8086-2 CPU with no wait states, yielding a memory cycle time of 500 nsec.

In addition to the on-board RAM, the iSBC 86/05 board has four 28-pin sockets, configured to accept JEDEC 24/28-pin standard memory devices. Up to 64K bytes of EPROM are supported in 16K-byte increments with Intel 27128 EPROMs. The iSBC 86/05 board is also compatible with the 2716, 2732, and 2764 EPROMs offering expansion to 8, 16 and 32K bytes, respectively.

With the addition of the iSBC 341 MULTIMODULE EPROM option, the on-board capacity for these devices is doubled, providing up to 128K bytes of EPROM capacity on-board.

Parallel I/O Interface

The iSBC 86/05 Single Board Computer contains 24 programmable parallel I/O lines implemented using the Intel 8255A Programmable Peripheral Interface. The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports indicated in Table 1. In order to take advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators, allowing the selection of the appropriate combination of optional line drivers and terminators with the required drive/termination characteristics. The 24 programmable I/O lines and signal ground lines are brought out to a 50-pin edge connector.

Serial I/O

A programmable communications interface using the Intel 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the iSBC 86/05 board. A software selectable baud rate generator provides the USART with all common communication frequencies. The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. The 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The RS232C compatible interface on each board, in conjunction with the USART, provides a direct interface to RS232C compatible terminals, cassettes, and asynchronous and synchronous modems. The RS232C command lines, serial data lines and signal ground line are brought out to a 26-pin edge connector.

Programmable Timers

The iSBC 86/05 board provides three independent, fully programmable 16-bit interval timers/event counters utilizing the Intel 8253 Programmable

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					Control
		Unidirectional				Bidirectional	
		Input		Output			
		Latched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]

NOTE:
1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.

Interval Timer. Each counter is capable of operating in either BCD or binary modes. Two of these timers/counters are available to the systems designer to generate accurate time intervals under software control. Routing for the outputs and gate/trigger inputs of two of these counters is jumper selectable. The outputs may be independently routed to the 8259A Programmable Interrupt Controller and to the I/O terminators associated with the 8255A to allow external devices or an 8255A port to gate the timer or to count external events. The third interval timer in the 8253 provides the programmable baud rate generator for the iSBC 86/05 board RS232C USART serial port. The system software configures each timer independently to select the desired function. Seven functions are available as shown in Table 2. The contents of each counter may be read at any time during system operation.

Table 2. Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated. This function is extremely useful for generation of real-time clocks.
Programmable one-shot	Output goes low upon receipt of an external trigger edge or software command and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counter "window" has been enabled or an interrupt may be generated after N events occur in the system.

iSBX MULTIMODULE On-Board Expansion

Two 8/16-bit iSBX MULTIMODULE connectors are provided on the iSBC 86/05 microcomputer. Through these connectors, additional on-board I/O functions may be added. iSBX MULTIMODULES optimally support functions provided by VLSI peripheral components such as additional parallel and serial I/O, analog I/O, small mass storage device controllers (e.g., cassettes and floppy disks), and other custom interfaces to meet specific needs. By mounting directly on the single board computer, less interface logic, less power, simpler packaging, higher performance, and lower cost result when compared to other alternatives such as MULTIBUS form factor compatible boards. The iSBX connectors on the iSBC 86/05 provide all signals necessary to interface to the local on-board bus, including 16 data lines for maximum data transfer rates. iSBX MULTIMODULE boards designed with 8-bit data paths and using the 8-bit iSBX connector are also supported on the iSBC 86/05 microcomputer. A broad range of iSBX MULTIMODULE options are available in this family from Intel. Custom iSBX modules may also be designed for use on the iSBC 86/05 board. An iSBX bus interface specification and iSBX connectors are available from Intel.

MULTIBUS™ SYSTEM BUS AND MULTIMASTER CAPABILITIES

Overview

The MULTIBUS system bus is Intel's industry standard microcomputer bus structure. Both 8 and 16-bit single board computers are supported on the MULTIBUS structure with 24 address and 16 data lines. In its simplest application, the MULTIBUS system bus allows expansion of functions already contained on a single board computer (e.g., memory and digital I/O). However, the MULTIBUS structure also allows very powerful distributed processing configurations with multiple processors and intelligent slave I/O, and peripheral boards capable of solving the most demanding microcomputer applications. The MULTIBUS system bus is supported with a broad array of board level products, LSI interface components, detailed published specifications and application notes.

Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS

compatible expansion boards. Memory may be expanded by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be added with digital I/O and analog I/O expansion boards. Mass storage capability may be achieved by adding single or double density diskette controllers, or hard disk controllers. Modular expandable backplanes and cardcages are available to support multiboard systems.

Multimaster Capabilities

For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e., several CPUs and/or controllers logically sharing system tasks through communication of the system bus), the iSBC 86/05 board provides full MULTIBUS arbitration control logic. This control logic allows up to three iSBC 86/05 boards or other bus masters, including iSBC 80 family MULTIBUS compatible 8-bit single board computers to share the system bus using a serial (daisy chain) priority scheme and allows up to 16 masters to share the MULTIBUS system bus with an external parallel priority decoder. In addition to the multiprocessing configurations made possible with multimaster capability, it also provides a very efficient mechanism for all forms of DMA (Direct Memory Access) transfers.

Interrupt Capability

The iSBC 86/05 board provides 9 vectored interrupt levels. The highest level is the NMI (Non-Maskable Interrupt) line which is directly tied to the 8086 CPU. This interrupt is typically used for signaling catastrophic events (e.g., power failure). The Intel 8259A Programmable Interrupt Controller (PIC) provides control and vectoring for the next eight interrupt levels. As shown in Table 3, a selection of four priority processing modes is available for use in designing request processing configurations to match system requirements for efficient interrupt servicing with minimal latencies. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from all on-board I/O resources and from the MULTIBUS system bus. The PIC then resolves requests according to the selected mode and, if appropriate, issues an interrupt to the CPU. Any combination of interrupt levels may be masked via software, by storing a single byte in the interrupt mask register of the PIC. In systems requiring additional interrupt levels, slave 8259A PICs may be interfaced via the MULTIBUS system bus, to generate additional vector addresses, yielding a total of 65 unique interrupt levels.

Table 3. Programmable Interrupt Modes

Mode	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.
Polled	System software examines priority-encoded system interrupt status via interrupt status register.

Interrupt Request Generation

Interrupt requests to be serviced by the iSBC 86/05 board may originate from 24 sources. Table 4 includes a list of devices and functions supported by interrupts. All interrupt signals are brought to the interrupt jumper matrix where any combination of interrupt sources may be strapped to the desired interrupt request level on the 8259A PIC or the NMI input to the CPU directly.

Power-Fail Control and Auxiliary Power

Control logic is also included to accept a power-fail interrupt in conjunction with the AC-low signal from the iSBC 635 and iSBC 640 Power Supply or equivalent, to initiate an orderly shut down of the system in the event of a power failure. Additionally, an active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences. An auxiliary power bus is also provided to allow separate power to RAM for systems requiring battery back-up of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

System Development Capabilities

The development cycle of iSBC 86/05 products can be significantly reduced and simplified by

using the Intellec Series Microcomputer Development Systems. The Assembler, Locating Linker, Library Manager, Text Editor and System Monitor are all supported by the ISIS-II disk-based operating system. To facilitate conversion of 8080A/8085A assembly language programs to run on the iSBC 86/05 board, CONV-86 is available under the ISIS-II operating system.

IN-CIRCUIT EMULATOR

The ICE-86 In-Circuit Emulator provides the necessary link between the software development environment provided by the Intellec system and the "target" iSBC 86/05 execution system. In addition to providing the mechanism for loading executable code and data into the iSBC 86/05 board, the ICE-86 In-Circuit Emulator provides a sophisticated command set to assist in debugging software and final integration of the user hardware and software.

PL/M-86

Intel's system's implementation language, PL/M-86, is also available as an Intellec Microcomputer Development System option. PL/M-86 provides the capability to program in algorithmic language and eliminates the need to manage register usage or allocate memory while still allowing explicit control of the system's resources when needed.

Run-Time Support

Intel also offers two run-time support packages; iRMX 88 Realtime Multitasking Executive and the iRMX 86 Operating System. iRMX 88 is a simple, highly configurable and efficient foundation for small, high performance applications. Its multitasking structure establishes a solid foundation for modular system design and provides task scheduling and management, intertask communication and synchronization, and interrupt servicing for a variety of peripheral devices. Other configurable options include terminal handlers, disk file system, debuggers and other utilities. iRMX 86 is a high functional operating system with a very rich set of features and options based on an object-oriented architecture. In addition to being modular and configurable, functions beyond the nucleus include a sophisticated file management and I/O system, and powerful human interface. Both packages are easily customized and extended by the user to match unique requirements.

Table 4. Interrupt Request Sources

Device	Function	Number of Interrupts
MULTIBUS interface	Requests from MULTIBUS resident peripherals or other CPU boards	8; may be expanded to 64 with slave 8259A PICs on MULTIBUS boards
8255A Programmable Peripheral Interface	Signals input buffer full or output buffer empty; also BUS INTR OUT general purpose interrupt from driver/terminator sockets	3
8251A USART	Transmit buffer empty and receive buffer full	2
8253 Timers	Timer 0, 1 outputs; function determined by timer mode	2
iSBX connectors	Function determined by iSBX MULTIMODULE board	4 (2 per iSBX connector)
Bus fail safe timer	Indicates addressed MULTIBUS resident device has not responded to command within 6 msec	1
Power fail interrupt	Indicates AC power is not within tolerance	1
Power line clock	Source of 120 Hz signal from power supply	1
External interrupt	General purpose interrupt from auxiliary (P2) connector on backplane	1
iSBC 337 MULTIMODULE Numeric Data Processor	Indicates error or exception condition	1

SPECIFICATIONS

Word Size

INSTRUCTION — 8, 16, 24, or 32 bits
DATA — 8, 16 bits

System Clock

5.00 MHz or 8.00 MHz ±0.1% (jumper selectable)

Cycle Time

BASIC INSTRUCTION CYCLE

At 8 MHz — 750 nsec
— 250 nsec (assumes instruction in the queue)
At 5 MHz— 1.2 μsec
— 400 nsec (assumes instruction in the queue)

NOTES:
Basic instruction cycle is defined as the fastest instruction time (i.e., two clock cycles).

Memory Cycle Time

RAM — 500 nsec (no wait states)
EPROM — Jumper selectable from 500 nsec to 875 nsec

Memory Capacity/Addressing

ON-BOARD EPROM

Device	Total Capacity	Address Range
2716	8K bytes	FE000-FFFFF$_H$
2732	16K bytes	FC000-FFFFF$_H$
2764	32K bytes	F8000-FFFFF$_H$
27128	64K bytes	F0000-FFFFF$_H$

WITH iSBC 341 MULTIMODULE EPROM

Device	Total Capacity	Address Range
2716	16K bytes	FC000-FFFFF$_H$
2732	32K bytes	F8000-FFFFF$_H$
2764	64K bytes	F0000-FFFFF$_H$
27128	128K bytes	E0000-FFFFF$_H$

NOTES:
iSBC 86/05 EPROM sockets support JEDEC 24/28-pin standard EPROMs and RAMs; iSBC 341 sockets also support E^2PROMs.

ON-BOARD RAM

8K bytes — 0-1FFF$_H$

WITH iSBC 302 MULTIMODULE RAM

16K bytes — 0-3FFF$_H$

I/O Capacity

PARALLEL — 24 programmable lines using one 8255A.
SERIAL — 1 programmable line using one 8251A
iSBX MULTIMODULE — 2 iSBX MULTIMODULE boards

Serial Communications Characteristics

SYNCHRONOUS — 5-8 bit characters; internal or external character synchronization; automatic sync insertion

ASYNCHRONOUS — 5-8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detection

BAUD RATES

Frequency (kHz) (Software Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous	
		÷16	÷64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
2.4	2400	150	—
1.76	1760	110	—

NOTES:
Frequency selected by I/O write of appropriate 16-bit frequency factor to baud rate register (8253 Timer 2).

Timers

INPUT FREQUENCIES

Reference: 2.46 MHz ±0.1% (0.041 μsec period, nominal); or 153.60 kHz ±0.1% (6.51 μsec period, nominal)

NOTES:
Above frequencies are user selectable.

Event Rate: 2.46 MHz max

OUTPUT FREQUENCIES/TIMING INTERVALS

Function	Single Timer/Counter		Dual Timer/Counter (Two Timers Cascaded)	
	Min	Max	Min	Max
Real-time Interrupt	1.63 μs	427.1 ms	3.26s	466.50 min
Programmable one-shot	1.63 μs	427.1 ms	3.26s	466.50 min
Rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Square-wave rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Software triggered strobe	1.63 μs	427.1 ms	3.26s	466.50 min
Hardware triggered strobe	1.63 μs	427.1 ms	3.26s	466.50 min
Event counter	—	2.46 MHz	—	—

Interfaces

MULTIBUS — All signals TTL compatible
iSBX BUS — All signals TTL compatible
PARALLEL I/O — All signals TTL compatible
SERIAL I/O — RS232C compatible, configurable as a data set or data terminal
TIMER — All signals TTL compatible
INTERRUPT REQUESTS — All TTL compatible

Connectors

Interface	Double-Sided Pins (qty)	Centers (in.)	Mating Connectors
MULTIBUS™ System	86	0.156	Viking 3KH43/9AMK12 Wire Wrap
iSBX™ Bus			
8-Bit Data	36	0.1	iSBX 960-5
16-Bit Data	44	0.1	iSBX 961-5
Parallel I/O (2)	50	0.1	3M 3415-000 Flat or TI H312125 Pins
Serial I/O	26	0.1	3M 3462-0001 Flat or AMP 88106-1 Flat

Line Drivers and Terminators

I/O DRIVERS — The following line drivers are all compatible with the I/O driver sockets on the iSBC 86/05 board

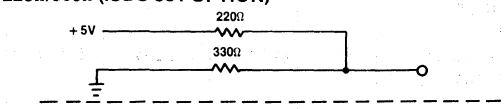
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

NOTES:
I = inverting; NI = non-inverting; OC = open collector.

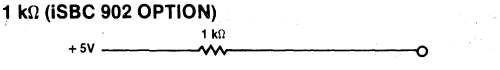
Port 1 of the 8255A has 20 mA totem-pole bidirectional drivers and 1 kΩ terminators

I/O TERMINATORS — 220Ω/330Ω divider or 1 kΩ pullup

220Ω/330Ω (iSBC 901 OPTION)



1 kΩ (iSBC 902 OPTION)



MULTIBUS Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-State	50
Address	Tri-State	50
Commands	Tri-State	32
Bus Control	Open Collector	20

Physical Characteristics

WIDTH — 12.00 in. (30.48 cm)
HEIGHT — 6.75 in. (17.15 cm)
DEPTH — 0.70 in. (1.78 cm)
WEIGHT — 14 oz (388 gm)

Electrical Characteristics

DC POWER REQUIREMENTS

Configuration	Current Requirements (All Voltages ±5%)		
	+5V	+12V	−12V
Without EPROM[1]	4.7A	25 mA	23 mA
RAM only[2]	120 mA		
With 8K EPROM[3] (using 2716)	5.0A	25 mA	23 mA
With 16K EPROM[3] (using 2732)	4.9A	25 mA	23 mA
With 32K EPROM[3] (using 2764)	4.9A	25 mA	23 mA

NOTES:
1. Does not include power for optional ROM/EPROM, I/O drivers, and I/O terminators.
2. RAM chips powered via auxiliary power bus in power-down mode.
3. Includes power required for 4 ROM/EPROM chips, and I/O terminators installed for 16 I/O lines; all terminator inputs low.

Environmental Characteristics

OPERATING TEMPERATURE — 0°C to 55°C
RELATIVE HUMIDITY — to 90% (without condensation)

Reference Manual

143153-001 — iSBC 86/05 Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

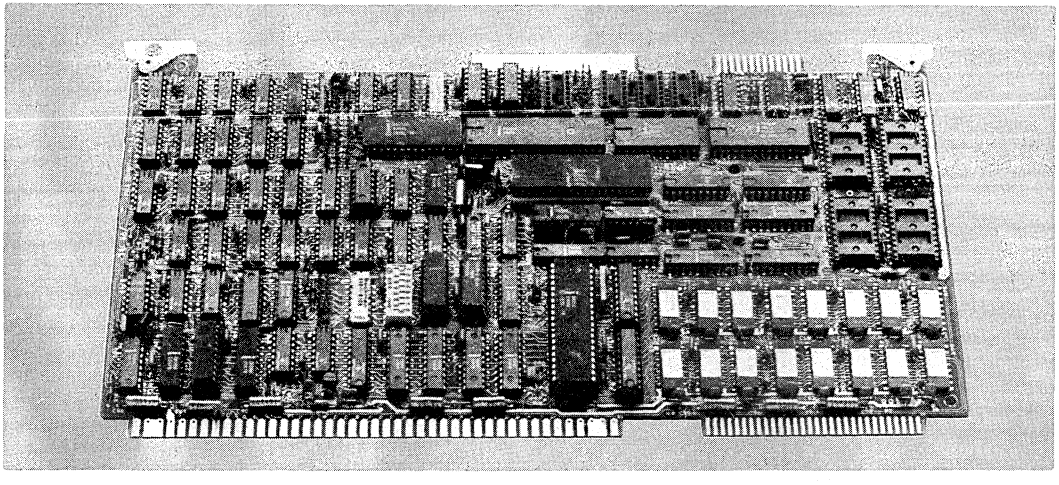
ORDERING INFORMATION

Part Number	Description
SBC 86/05	16-bit Single Board Computer with 8K bytes RAM

iSBC™ 86/12A or (pSBC 86/12A*) SINGLE BOARD COMPUTER

- 8086 16-bit HMOS microprocessor central processor unit
- 32K bytes of dual-port read/write memory expandable on-board to 64K bytes with on-board refresh
- Sockets for up to 16K bytes of read only memory expandable on-board to 32K bytes
- System memory expandable to 1 megabyte
- 24 programmable parallel I/O lines with sockets for interchangeable line drivers and terminators
- Programmable synchronous/asynchronous RS232C compatible serial interface with software selectable baud rates
- Two programmable 16-bit BCD or binary timers/event counters
- 9 levels of vectored interrupt control, expandable to 65 levels
- Auxiliary power bus and power fail interrupt control logic for read/write memory battery backup
- MULTIBUS® interface for multimaster configurations and system expansion
- Compatible with iSBC 337 MULTIMODULE™ Numeric Data Processor
- Compatible with iSBC 80 family single board computers, memory, digital and analog I/O, and peripheral controller boards

The iSBC 86/12A Single Board Computer is a member of Intel's complete line of OEM microcomputer systems which take full advantage of Intel's LSI technology to provide economical self-contained computer based solutions for OEM applications. The iSBC 86/12A board is a complete computer system on a single 6.75 x 12.00-inch printed circuit card. The CPU, system clock, read/write memory, nonvolatile read only memory, I/O ports and drivers, serial communications interface, priority interrupt logic and programmable timers, all reside on the board. Full MULTIBUS interface logic is included to offer compatibility with the Intel OEM Microcomputer Systems family of Single Board Computers, expansion memory options, digital and analog I/O expansion boards and peripheral controllers.



*Same product, manufactured by Intel Puerto Rico, Inc.

The following are trademarks of Intel Corporation and may be used only to describe Intel products: Index, Intel, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, iRMX, UPI, μScope, Promware, MCS, ICE, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1980, 1981

September 1981
9803115A.01

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processor for the iSBC 86/12A board is Intel's 8086, a powerful 16-bit HMOS device. The 225 sq. mil chip contains 29,000 transistors and has a clock rate of 5MHz. The architecture includes four (4) 16-bit byte addressable data registers, two (2) 16-bit memory base pointer registers and two (2) 16-bit index registers, all accessed by a total of 24 operand addressing modes for complex data handling and very flexible memory addressing.

Instruction Set — The 8086 instruction repertoire includes variable length instruction format (including double operand instructions), 8-bit and 16-bit signed and unsigned arithmetic operators for binary, BCD and unpacked ASCII data, and iterative word and byte string manipulation functions. In addition, the iSBC 337 MULTIMODULE Numeric Data Processor may be installed to add over 60 numeric instructions and hardware support for multiple precision integer and floating point data types.

Architectural Features — A 6-byte instruction queue provides pre-fetching of sequential instructions and can reduce the 1.2μsec minimum instruction cycle to 400 nsec for queued instructions. The stack oriented architecture facilitates nested subroutines and co-routines, reentrant code and powerful interrupt handling. The memory expansion capabilities offer a 1 megabyte addressing range. The dynamic relocation scheme allows ease in segmentation of pure procedure and data for efficient memory utilization. Four segment registers (code, stack, data, extra) contain program loaded offset values which are used to map 16-bit addresses to 20-bit addresses. Each register maps 64K-bytes at a time and activation of a specific register is controlled explicitly by program control and is also selected implicitly by specific functions and instructions.

Bus Structure

The iSBC 86/12A microcomputer has three buses: an internal bus for communicating with on-board memory and I/O options, the MULTIBUS system bus for referencing additional memory and I/O options, and the dual-port bus which allows access to RAM from the on-board CPU and the MULTIBUS system bus. Local (on-board) accesses do not require MULTIBUS communication, making the system bus available for use by other MULTIBUS masters (i.e. DMA devices and other single board computers transferring to additional system memory). This feature allows true parallel processing in a multiprocessor environment. In addition, the MULTIBUS interface can be used for system expansion through the use of other 8- and 16-bit iSBC computers, memory and I/O expansion boards.

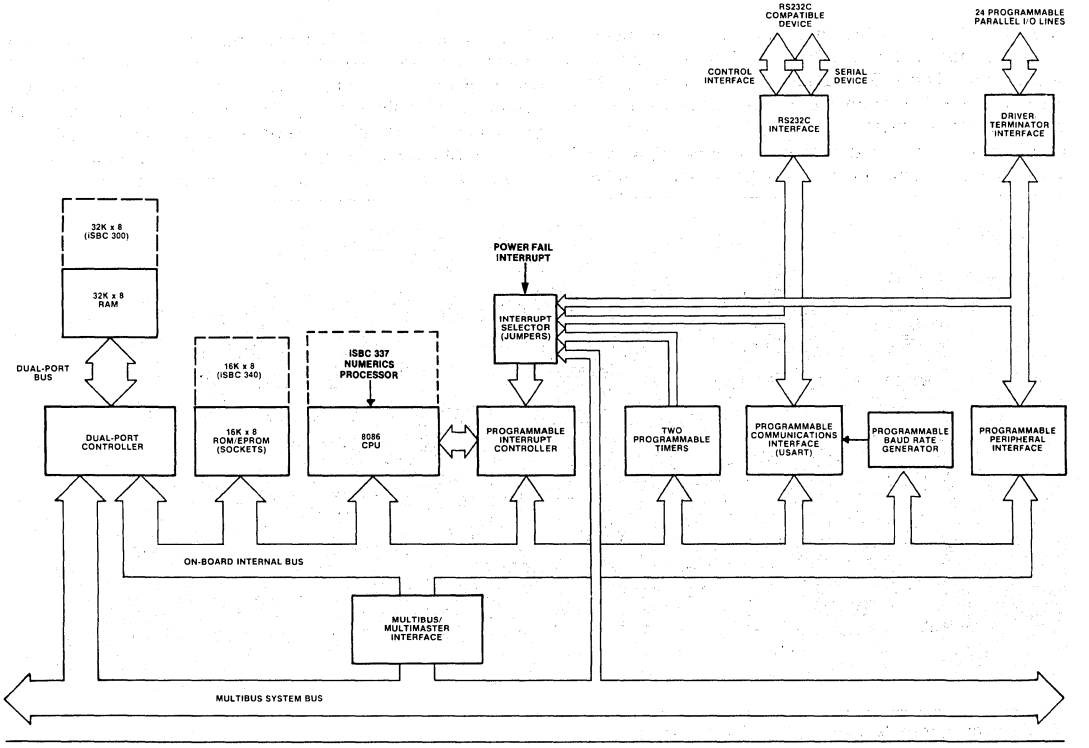


Figure 1. iSBC 86/12A Single Board Computer Block Diagram

RAM Capabilities

The iSBC 86/12A microcomputer contains 32K bytes of dynamic read/write memory using 16K-bit 2117 RAMs. In addition, the on-board RAM complement may be expanded to 64K bytes with the iSBC 300 32K-byte MULTIMODULE RAM option. Power for the on-board RAM and refresh circuitry may be optionally provided on an auxiliary power bus, and memory protect logic is included for RAM battery backup requirements. The iSBC 86/12A board contains a dual-port controller which allows access to the on-board RAM (32K bytes or 64K bytes when the iSBC 300 module is included with the iSBC 86/12A board) from the iSBC 86/12A CPU and from any other MULTIBUS master via the system bus. The dual-port controller allows 8- and 16-bit accesses from the MULTIBUS system bus, and the on-board CPU transfers data to RAM over a 16-bit data path. Priorities have been established such that memory refresh is guaranteed by the on-board refresh logic and that the on-board CPU has priority over MULTIBUS system bus requests for access to RAM. The dual-port controller includes independent addressing logic for RAM access from the on-board CPU and from the MULTIBUS system bus. The on-board CPU will always access RAM starting at location 00000_H. Address jumpers allow on-board RAM to be located starting on any 8K-byte boundary within a 1 megabyte address range for accesses from the MULTIBUS system bus. In conjunction with this feature, the iSBC 86/12A microcomputer has the ability to protect on-board memory from MULTIBUS access to any contiguous 8K-byte segments (or 16K-byte segments with iSBC 300 module). These features allow the multiprocessor systems to establish local memory for each processor and shared system (MULTIBUS) memory configurations where the total system memory size (including local on-board memory) can exceed 1 megabyte without addressing conflicts.

EPROM Capabilities

Four sockets are provided for up to 16K bytes of nonvolatile read only memory on the iSBC 86/12A board. EPROM may be added in 2K-byte increments up to a maximum of 4K bytes by using Intel® 2758 electrically programmable ROMs (EPROMs); in 4K-byte increments up to 8K bytes by using Intel 2716 EPROMs; or in 8K-byte increments up to 16K bytes using Inel 2732 EPROMs. On-board EPROM is accessed via 16-bit data paths. On-board EPROM capacity may be expanded to 32K bytes with the addition of the iSBC 340 16K-byte MULTIMODUE EPROM option. It provides an additional four sockets for Intel 2732 EPROMs. With user modification of the iSBC 86/12A's on-board memory and MULITBUS address decode, Intel 2758 and 2716 EPROMs may be optionally supported. System memory size is easily expanded by the addition of MULTIBUS system bus compatible memory boards available in the iSBC product family.

Parallel I/O Interface

The iSBC 86/12A single board computer contains 24 programmable parallel I/O lines implemented using the Intel 8255A Programmable Peripheral Interface. The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports indicated in Table 1. Therefore, the I/O interface may be customized to meet specific peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 24 programmable I/O lines and signal ground lines are brought out to a 50-pin edge connector that mates with flat, woven, or round cable.

Serial I/O

A programmable communications interface using the Intel 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the iSBC 86/12A board. A software selectable baud rate generator provides the USART with all common communication

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Latched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]

Note
1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.

frequencies. The USART can be programmed by the system software to select the desired asynchronous or synchronous serial data transmission technique (including IBM Bi-Sync). The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. The 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The RS232C compatible interface on each board, in conjunction with the USART, provides a direct interface to RS232C compatible terminals, cassettes, and asynchronous and synchronous modems. The RS232C command lines, serial data lines, and signal ground line are brought out to a 26 pin edge connector that mates with RS232C compatible flat or round cable. The iSBC 530 Teletypewriter Adapter provides an optically isolated interface for those systems requiring a 20 mA current loop. The iSBC 530 unit may be used to interface the iSBC 86/12A board to teletypewriters or other 20 mA current loop equipment.

Programmable Timers

The iSBC 86/12A board provides three independent, fully programmable 16-bit interval timers/event counters utilizing the Intel 8253 Programmable Interval Timer. Each counter is capable of operating in either BCD or binary modes. Two of these timers/counters are available to the systems designer to generate accurate time intervals under software control. Routing for the outputs and gate/trigger inputs of two of these counters is jumper selectable. The outputs may be independently routed to the 8259A Programmable Interrupt Controller and to the I/O line drivers associated with the 8255A Programmable Peripheral Interface, or may be routed as inputs to the 8255A chip. The gate/trigger inputs may be routed to I/O terminators associated with the 8255A or as output connections from the 8255A. The third interval timer in the 8253 provides the programmable baud rate generator for the iSBC 86/12A board RS232C USART serial port. In utilizing the iSBC 86/12A board the systems designer simply configures, via software, each timer independently to meet system requirements. Whenever a given time delay or count is needed, software commands to the programmable timers/event counters select the desired function. Seven functions are available, as shown in Table 2. The contents of each counter may be read at any time during system operation with simple read operations for event counting applications, and special commands are included so that the contents can be read "on the fly".

MULTIBUS System Bus and Multimaster Capabilities

The MULTIBUS system bus features asynchronous data transfers for the accommodation of devices with various transfer rates while maintaining maximum throughput. Twenty address lines and sixteen separate data lines eliminate the need for address/data multiplexing/demultiplexing logic used in other systems, and allow for data transfer rates up to 5 megawords/sec. A failsafe timer is included in the iSBC 86/12A board which can be used to generate an interrupt if an addressed device does not respond within 6 msec.

Table 2. Programmable Ti er Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated. This function is extremely useful for generation of real-time clocks.
Programmable one-shot	Output goes low upon receipt of an external trigger edge or software command and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counting "window" has been enabled or an interrupt may be generated after N events occur in the system.

Multimaster Capabilities — The iSBC 86/12A board is a full computer on a single board with resources capable of supporting a great variety of OEM system requirements. For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e., several CPUs and/or controllers logically sharing system tasks through communication over the system bus), the iSBC 86/12A board provides full MULTIBUS arbitration control logic. This control logic allows up to three iSBC 86/12A boards or other bus masters, including iSBC 80 family MULTIBUS compatible 8-bit single board computers, to share the system bus in serial (daisy chain) priority fashion and up to 16 masters to share the MULTIBUS system bus with the addition of an external priority network. The MULTIBUS arbitration logic operates synchronously with a MULTIBUS clock (provided by the iSBC 86/12A board or optionally provided directly from the MULTIBUS) while data is transferred via a handshake between the master and slave modules. This allows different speed controllers to share resources on the same bus, and transfers via the bus proceed asynchronously. Thus, transfer speed is dependent on transmitting and

receiving devices only. This design prevents slow master modules from being handicapped in their attempts to gain control of the bus, but does not restrict the speed at which faster modules can transfer data via the same bus. The most obvious applications for the master-slave capabilities of the bus are multiprocessor configurations, high speed peripheral control, but are by no means limited to these three.

Interrupt Capablity

The iSBC 86/12A board provides 9 vectored interrupt levels. The highest level is the NMI (Non-maskable Interrupt) line which is directly tied to the 8086 CPU. This interrupt cannot be inhibited by software and is typically used for signalling catastrophic events (i.e., power failure). On servicing this interrupt, program control will be implicitly transferred through location 00008_H. The Intel 8259A Programmable Interrupt Controller (PIC) provides vectoring for the next eight interrupt levels. As shown in Table 3, a selection of four priority processing modes is available to the systems designer for use in designing request processing configurations to match system requirements. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from the programmable parallel and serial I/O interfaces, the programmable timers, the system bus, or directly from peripheral equipment. The PIC then determines which of the incoming requests is of the highest priority, determines whether this request is of higher priority than the level currently being serviced, and, if appropriate, issues an interrupt to the CPU. Any combination of interrupt levels may be masked, via software, by storing a single byte in the interrupt mask register of the PIC. The PIC generates a unique memory address for each interrupt level. These addresses are equally spaced at 4 byte intervals. This 32-byte block may begin at any 32-byte boundary in the lowest 1K-bytes of memory,* and contains unique instruction pointers and code segment offset values (for expanded memory operation) for each interrupt level. After acknowledging an interrupt and obtaining a device identifier byte from the 8259A PIC, the CPU will store its status flags on the stack and execute an indirect CALL instruction through the vector location (derived from the device identifier) to the interrupt service routine. In systems requiring additional interrupt levels, slave 8259A PIC's may be interfaced via the MULTIBUS system bus, to generate additional vector addresses, yielding a total of 65 unique interrupt levels.

Interrupt Request Generation — Interrupt requests may originate from 18 sources. Two jumper selectable interrupt requests can be automatically generated by the programmable peripheral interface when a byte of information is ready to be transferred to the CPU (i.e., input buffer is full) or a byte of information has been transferred to a peripheral device (i.e., output buffer is empty). Two jumper selectable interrupt requests can be automatically generated by the USART when a character is ready to be transferred to the CPU (i.e., receive channel buffer is full, or a character is ready to be transmitted (i.e., transmit channel data buffer is empty). A jumper selectable request can be generated by each of the programmable timers. An additional interrupt request line may be jumpered directly from the parallel I/O driver terminator section. Eight prioritized interrupt request lines allow the iSBC 86/12A board to recognize and service interrupts originating from peripheral boards interfaced via the MULTIBUS system bus. The MULTIBUS fail safe timer of the iSBC 337 processor and the exception and error output signal also can be selected as interrupt sources.

Table 3. Programmable Interrupt Modes

Mode	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.
Polled	System software examines priority-encoded system interrupt status via interrupt status register.

*Note: The first 32 vector locations are reserved by Intel for dedicated vectors. Users who wish to maintain compatibility with present and future Intel products should not use these locations for user-defined vector addresses.

Power-Fail Control

Control logic is also included to accept a power-fail interrupt in conjunction with the AC-low signal from the iSBC 635 and iSBC 640 Power Supply or equivalent.

Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS compatible expansion boards. Memory may be expanded by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be increased by adding digital I/O and analog I/O expansion boards. Mass storage capability may be achieved by adding single or double density diskette controllers, or hard disk controllers. Modular expandable backplanes and cardcages are available to support multiboard systems.

Note: Certain system restrictions may be incurred by the inclusion of some of the iSBC 80 family options in an iSBC 86/12A system. Consult the Intel OEM Microcomputer System Configuration Guide for specific data.

System Development Capabilities

The development cycle of iSBC 86/12A products can be significantly reduced by using the Intellec® series microcomputer development system. The Assembler, High Level Languages, Locating Linker, Library Manager, Text Editor and System Monitor are all supported by the ISIS-II disk-based operating system.

In-Circuit Emulator — ICE™-86 in-circuit emulator provides the necessary link between the software development environment provided by the Intellec system and the "target" iSBC 86/12A execution system. In addition to providing the mechanism for loading executable code and data into the iSBC 86/12A board, ICE-86 in-circuit emulator provides a sophisticated command set to assist in debugging software and final integration of the user hardware and software. ICE-86 in-circuit emulator maximizes the use of available development resources by allowing Intellec resident resources (e.g., memory and peripherals) to be accessed by software running on the target iSBC 86/12A system. In addition, software can be executed without an iSBC 86/12A execution vehicle, in 2K bytes of RAM resident in the ICE-86 system itself. Symbolic references to instruction and data locations can be made through ICE-86 in-circuit emulator to allow the user to reference memory locations with assigned names.

PL/M-86 — Intel's high level programming language, PL/M-86, is also available as an Intellec Microcomputer Development System option. PL/M-86 provides the capability to program in a natural, algorithmic language and eliminates the need to manage register usage or allocate memory. PL/M-86 programs can be written in a much shorter time than assembly language programs for a given application. PL/M-86 includes byte and word, integer, pointer and floating point (32-bit) data types and also includes conditional compilation and macro features.

SPECIFICATIONS

Word Size

Instruction — 8, 16, 24, or 32 bits
Data — 8, 16 bits

Cycle Time

Basic Instruction Cycle — 1.2 μsec
— 400 nsec (assumes instruction in the queue)

Note:
Basic instruction cycle is defined as the fastest instruction time (i.e., two clock cycles)

Memory Capacity

On-Board Read Only Memory — 16K bytes (sockets only); expandable to 32K bytes with iSBC 340 16K-byte MULTIMODULE EPROM option.

On-Board RAM — 32K bytes; expandable to 64K bytes with iSBC 300 32K-byte MULTIMODULE RAM option.

Off-Board Expandion — Up to 1 megabyte in user specified combinations of RAM and EPROM.

Note:
Read only memory may be added in 2K, 4K, or 8K-byte increments.

Memory Addressing

On-Board EPROM — FF000-FFFFF$_H$ (using 2758 EPROMs); FE000-FFFFF$_H$ (using 2716 EPROMs); FC000-FFFFF$_H$ (using 2732 EPROMS); F8000-FFFFF$_H$ (with iSBC 340 EPROM option and four additional 2732 EPROMs).

On-Board RAM — 32K bytes of dual port RAM. Optionally expandable to 64K bytes with iSBC 300 RAM option.

CPU Access — 32K bytes: 00000-07FFF$_H$; 64K bytes: 00000-0FFFF$_H$.

MULTIBUS Access — Jumper selectable for any 8K-byte boundary, but not crossing a 128K-byte boundary. Access for 8K, 16K, 24K or 32K (16K, 32K, 48K, 64K with iSBC 300 option) bytes may be selected for on-board CPU use only.

I/O Capacity

Parallel — 24 programmable lines using one 8255A.
Serial — 1 programmable line using one 8251A.

I/O Addressing

On-Board Programmable I/O

Port	8255A				USART	
	1	2	3	Control	Data	Control
Address	C8	CA	CC	CE	D8 or DC	DA or DE

Serial Communications Characteristics

Synchronous — 5—8 bit characters; internal or external character synchronization; automatic sync insertion.

Asynchronous — 5—8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detection.

Baud Rates

Frequency (kHz) (Software Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous	
		÷ 16	÷ 64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
2.4	2400	150	—
1.76	1760	110	—

Note:
Frequency selected by I/O write of appropriate 16-bit frequency factor to baud rate register (8253 Timer 2).

Interrupts

Addresses for 8259A Registers (Hex notation I/O address space)

C0 or C4 Write: Initialization Command Word 1 (ICW1) and Operation Control Words 2 and 3 (OCW2 and OCW3)
Read: Status and Poll Registers

C2 or C6 Write: ICW2, ICW3, ICW4, OCW1 (Mask Register)
Read: OCW1 (Mask Register)

Note:
Several registers have the same physical address; sequence of access and one data bit of control word determine which register will respond.

Interrupt Levels — 8086 CPU includes a non-maskable Interrupt (NMI) and a maskable interrupt (INTR). NMI interrupt is provided for catastrophic events such as power failure. NMI vector address is 00008. INTR interrupt is driven by on-board 8259A PIC, which provides 8-bit identifier of interrupting device to CPU. CPU multiplies identifier by four to derive vector address. Jumpers select interrupts from 18 sources without necessity of external hardware. PIC May be programmed to accommodate edge-sensitive or level-sensitive inputs.

Timers

Register Addresses (Hex notation, I/O address space)

D0 Timer 0
D2 Timer 1
D4 Timer 2
D6 Control register

Note:
Timer counts are loaded as two sequential output operations to same address as given.

Input Frequencies

Reference: 2.46 MHz ±0.1% (0.041 μs period, nominal); 1.23 MHz ±0.1% (0.81 μs period, nominal); or 153.60 kHz ±0.1% (6.51 μs period nominal).

Note:
Above frequencies are user selectable.

Event Rate: 2.46 MHz max

Output Frequencies/Timing Intervals

Function	Single Timer/Counter		Dual Timer/Counter (Two Timers Cascaded)	
	Min	Max	Min	Max
Real-time interrupt	1.63 μs	427.1 ms	3.26 s	466.50 min
Programmable one-shot	1.63 μs	427.1 ms	3.26 s	466.50 min
Rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Square-wave rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Software triggered strobe	1.63 μs	427.1 ms	3.26 s	466.50 min
Hardware triggered strobe	1.63 μs	427.1 ms	3.26 s	466.50 min
Event counter	—	2.46 MHz	—	—

Interfaces

MULTIBUS — All signals TTL compatible
Parallel I/O — All signals TTL compatible
Interrupt Requests — All TTL compatible
Timer — All signals TTL compatible
Serial I/O — RS232C compatible, data set configuration

System Clock (8086 CPU)

5.00 MHz ± 0.1%

Auxiliary Power

An auxiliary power bus is provided to allow separate power to RAM for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Connectors

Interface	Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	VIKING 3KH43/9AMK12
Parallel I/O	50	0.1	3M 3415-000
Serial I/O	26	0.1	3M 3462-000

Memory Protect

An active low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power down sequences.

Line Drivers and Terminators

I/O Drivers — The following line drivers are all compatible with the I/O driver sockets on the iSBC 86/12A board

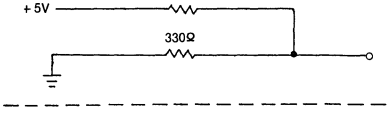
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note:
I = inverting; NI = non-inverting; OC = open collector.

Port 1 of the 8255A has 20 mA totem-pole bidirectional drivers and 1 kΩ terminators.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pullup

220Ω/330Ω (ISBC 901 OPTION)



1 K Ω (ISBC 902 OPTION)

Bus Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-state	50
Address	Tri-state	50
Commands	Tri-state	32

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.70 in. (1.78 cm)
Weight — 19 oz. (539 gm)

Electrical Characteristics

DC Power Requirements

Configuration	Current Requirements			
	V_{CC} = +5V ±5% (max)	V_{DD} = +12V ±5% (max)	V_{BB} = −5V ±5% (max)	V_{AA} = −12V ±5% (max)
Without EPROM[1]	5.2A	350 mA	—	40 mA
RAM Only[3]	390 mA	40 mA	1.0 mA	—
With iSBC 530[4]	5.2A	450 mA	—	140 mA
With 4K EPROM[5] (using 2758)	5.5A	350 mA	—	40 mA
With 8K EPROM[5] (using 2716)	5.5A	350 mA	—	40 mA
With 16K EPROM[5] (using 2732)	5.4A	350 mA	—	40 mA

Notes:

1. Does not include power for optional EPROM, I/O drivers, and I/O terminators.
2. Does not include power required for optional EPROM, I/O drivers, and I/O terminators.
3. RAM chips powered via auxiliary power bus.
4. Does not include power for optional EPROM, I/O drivers, and I/O terminators. Power for iSBC 530 is supplied via serial port connector.
5. Includes power required for four EPROM chips, and I/O terminators installed for 16 I/O lines; all terminator inputs low.

Environmental Characteristics

Operating Temperature — 0°C to 55°C
Relative Humidity — to 90% (without condensation)

Reference Manual

9803074-01 — iSBC 86/12A Single Board Computer Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

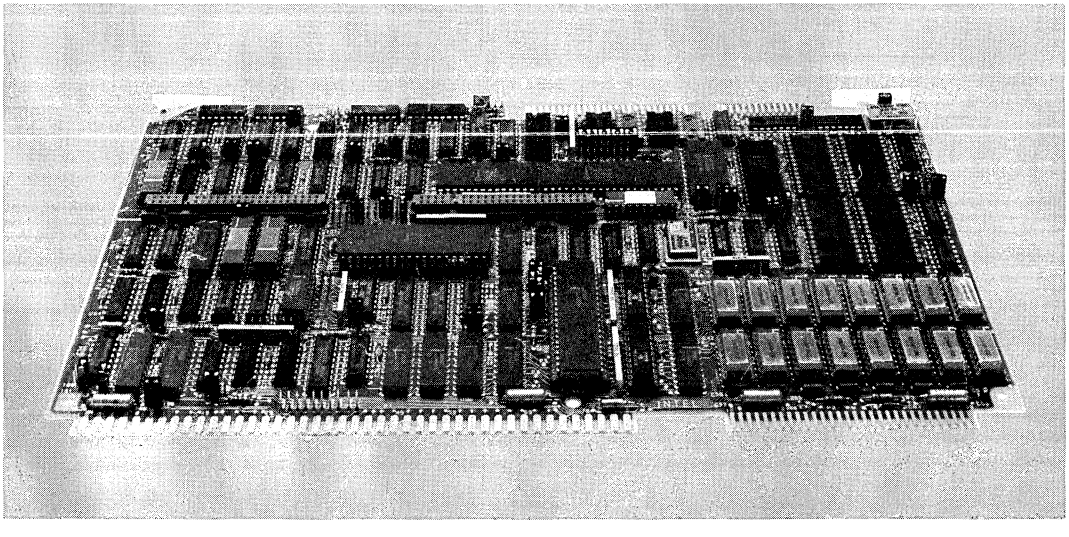
ORDERING INFORMATION

Part Number	Description
SBC 86/12A	Single Board Computer with 32K bytes RAM

iSBC™ 86/14 and iSBC™ 86/30 SINGLE BOARD COMPUTERS

- iAPX 86/10 (8086-2) Microprocessor with 5 or 8 MHz CPU clock
- Fully software compatible with iSBC™ 86/12A Single Board Computer
- Optional iAPX 86/20 Numeric Data Processor with iSBC™ 337 MULTIMODULE™ processor
- 32K/128K bytes of dual-port read/write memory expandable on-board to 256K bytes with on-board refresh
- Sockets for up to 64K bytes of JEDEC 24/28-pin standard memory devices
- Two iSBX™ bus connectors
- 24 programmable parallel I/O lines
- Programmable synchronous/asynchronous RS232C compatible serial interface with software selectable baud rates
- Two programmable 16-bit BCD or binary timers/event counters
- 9 Levels of vectored interrupt control, expandable to 65 levels
- MULTIBUS® interface for multimaster configurations and system expansion
- Supported by a complete family of single board computers, memory, digital and analog I/O, peripheral controllers, packaging and software

The iSBC 86/14 and iSBC 86/30 Single Board Computers are members of Intel's complete line of OEM microcomputer systems which take full advantage of Intel's technology to provide economical, self-contained, computer-based solutions for OEM applications. Each board is a complete computer system on a single 6.75 × 12.00-in. printed circuit card distinguished by RAM memory content with 32K bytes and 128K bytes provided on the iSBC 86/14 and iSBC 86/30 board, respectively. The CPU, system clock, read/write memory, nonvolatile read only memory, I/O ports and drivers, serial communications interface, priority interrupt logic and programmable timers, all reside on the boards.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: CREDIT, Index, Intel, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1981

October 1981
Order Number: 210219-001

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processor for the iSBC 86/XX(1) boards is Intel's iAPX 86/10 (8086-2) CPU. A clock rate of 8 MHz is supported with a jumper selectable option of 5 MHz. The CPU architecture includes four 16-bit byte addressable data registers, two 16-bit memory base pointer registers and two 16-bit index registers, all accessed by a total of 24 operand addressing modes for comprehensive memory addressing and for support of the data structures required for today's structured, high level languages as well as assembly language.

(1) iSBC 86/XX designates both the iSBC 86/14 and iSBC 86/30 CPU boards.

Instruction Set

The 8086 instruction repertoire includes variable length instruction format (including double operand instructions), 8-bit and 16-bit signed and unsigned arithmetic operators for binary, BCD and unpacked ASCII data, and iterative word and byte string manipulation functions.

For enhanced numerics processing capability, the iSBC 337 MULTIMODULE Numeric Data Processor extends the iAPX 86/10 architecture and data set. Over 60 numeric instructions offer arithmetic, trigonometric, transcendental, logarithmic and exponential instructions. Supported data types include 16, 32, and 64-bit integer, and 32 and 64-bit floating point, 18-digit packed BCD and 80-bit temporary.

Architectural Features

A 6-byte instruction queue provides pre-fetching of sequential instructions and can reduce the 750 nsec minimum instruction cycle to 250 nsec for queued instructions. The stack-oriented architecture readily supports modular programming by facilitating fast, simple, inter-module communication, and other programming constructs needed for asynchronous real-time systems. The memory expansion capabilities offer a 1 megabyte addressing range. The dynamic relocation scheme allows ease in segmentation of pure procedure and data for efficient memory utilization. Four segment registers (code, stack, data, extra) contain program loaded offset values which are used to map 16-bit addresses to 20-bit addresses. Each register maps 64K bytes at a time and activation of a specific register is controlled explicitly by program control and is also selected implicitly by specific functions and instructions.

RAM Capabilities

The iSBC 86/14 and iSBC 86/30 microcomputers contain 32K bytes and 128K bytes of dual-port dynamic RAM, respectively. In addition, on-board

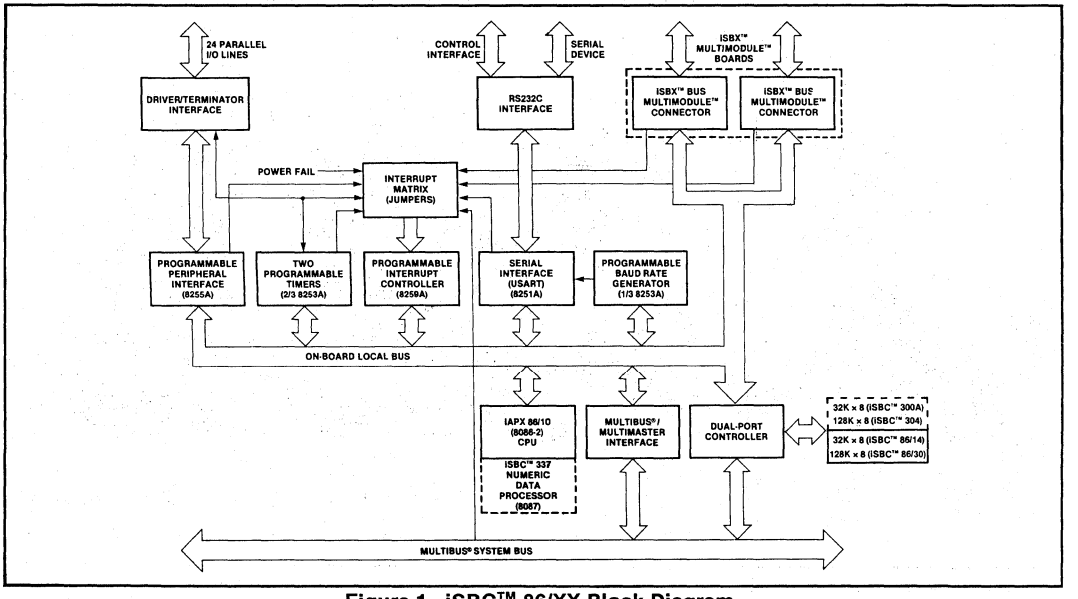


Figure 1. iSBC™ 86/XX Block Diagram

RAM may be doubled on each microcomputer by optionally adding RAM MULTIMODULE boards. The on-board RAM may be expanded to 256K bytes with the iSBC 304 MULTIMODULE Board mounted onto the iSBC 86/30 board. Likewise, the iSBC 86/14 microcomputer may be expanded to 64K bytes with the iSBC 300A MULTIMODULE option. The dual-port controller allows access to the on-board RAM (including RAM MULTIMODULE options) from the iSBC 86/XX boards and from any other MULTIBUS master via the system bus. Segments of on-board RAM may be configured as a private resource, protected from MULTIBUS system access. The amount of memory allocated as a private resource may be configured in increments of 25% of the total on-board memory ranging from 0% to 100% (optional RAM MULTIMODULE boards double the increment size). These features allow the multiprocessor systems to establish local memory for each processor and shared system memory configurations where the total system memory size (including local on-board memory) can exceed one megabyte without addressing conflicts.

EPROM Capabilities

Four 28-pin sockets are provided for the use of Intel 2716s, 2732s, 2764s, 27128s, and their respective ROMs. When using 27128s, the on-board EPROM capacity is 64K bytes. Other JEDEC standard pinout devices are also supported, including byte-wide static RAMs.

Parallel I/O Interface

The iSBC 86/XX Single Board Computers contain 24 programmable parallel I/O lines implemented using the Intel 8255A Programmable Peripheral Interface. The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports indicated in Table 1. In order to take advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators, allowing the selection of the appropriate combination of optional line drivers and terminators with the required drive/termination characteristics. The 24 programmable I/O lines and signal ground lines are brought out to a 50-pin edge connector.

Serial I/O

A programmable communications interface using the Intel 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the iSBC 86/XX boards. A software selectable baud rate generator provides the USART with all common communication frequencies. The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. The 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The RS232C command lines, serial data lines and signal ground line are brought out to a 26-pin edge connector.

Programmable Timers

The iSBC 86/XX boards provide three independent, fully programmable 16-bit interval timers/event counters utilizing the Intel 8253 Programmable Interval Timer. Each counter is capable of operating in either BCD or binary modes. Two of these timers/counters are available to the systems designer to generate accurate time intervals under software control. Routing for the outputs and gate/trigger inputs of two of these counters is jumper selectable. The outputs may be indepen-

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Latched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]

NOTE:

1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.

dently routed to the 8259A Programmable Interrupt Controller and to the I/O terminators associated with the 8255A to allow external devices or an 8255A port to gate the timer or to count external events. The third interval timer in the 8253 provides the programmable baud rate generator for the iSBC 86/XX boards' RS232C USART serial port. The system software configures each timer independently to select the desired function. Seven functions are available as shown in Table 2. The contents of each counter may be read at any time during system operation.

Table 2. Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated. This function is extremely useful for generation of real-time clocks.
Programmable one-shot	Output goes low upon receipt of an external trigger edge or software command and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counter "window" has been enabled or an interrupt may be generated after N events occur in the system.

iSBX™ MULTIMODULE™ On-Board Expansion

Two 8/16-bit iSBX MULTIMODULE connectors are provided on the iSBC 86/XX microcomputers. Through these connectors, additional on-board I/O functions may be added. iSBX MULTIMODULE boards optimally support functions provided by VLSI peripheral components such as additional parallel and serial I/O, analog I/O, small mass storage device controllers (e.g., cassettes and floppy disks), and other custom interfaces to meet specific needs. By mounting directly on the single board computer, less interface logic, less power, simpler packaging, higher performance, and lower cost result when compared to other alternatives such as MULTIBUS form factor compatible boards. The iSBX connectors on the iSBC 86/XX boards provide all signals necessary to interface to the local on-board bus, including 16 data lines for maximum data transfer rates. iSBX MULTIMODULE boards designed with 8-bit data paths and using the 8-bit iSBX connector are also supported on the iSBC 86/XX microcomputers. A broad range of iSBX MULTIMODULE options are available in this family from Intel. Custom iSBX modules may also be designed for use on the iSBC 86/XX boards. An iSBX bus interface specification and iSBX connectors are available from Intel.

MULTIBUS® SYSTEM BUS AND MULTIMASTER CAPABILITIES

Overview

The MULTIBUS system bus is Intel's industry standard microcomputer bus structure. Both 8 and 16-bit single board computers are supported on the MULTIBUS structure with 24 address and 16 data lines. In its simplest application, the MULTIBUS system bus allows expansion of functions already contained on a single board computer (e.g., memory and digital I/O). However, the MULTIBUS structure also allows very powerful distributed processing configurations with multiple processors and intelligent slave I/O, and peripheral boards capable of solving the most demanding microcomputer applications. The MULTIBUS system bus is supported with a broad array of board level products, LSI interface components, detailed published specifications and application notes.

Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS compatible expansion boards. Memory may be expanded by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be added with digital I/O and analog I/O expansion boards. Mass storage capability may be achieved by adding single or double density diskette controllers, or

hard disk controllers. Modular expandable backplanes and cardcages are available to support multiboard systems.

Multimaster Capabilities

For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e., several CPUs and/or controllers logically sharing system tasks through communication of the system bus), the iSBC 86/XX boards provide full MULTIBUS arbitration control logic. This control logic allows up to three iSBC 86/XX boards or other bus masters, including iSBC 80 family MULTIBUS compatible 8-bit single board computers to share the system bus using a serial (daisy chain) priority scheme and allows up to 16 masters to share the MULTIBUS system bus with an external parallel priority decoder. In addition to the multiprocessing configurations made possible with multimaster capability, it also provides a very efficient mechanism for all forms of DMA (Direct Memory Access) transfers.

Interrupt Capability

The iSBC 86/XX boards provide 9 vectored interrupt levels. The highest level is the NMI (Non-Maskable Interrupt) line which is directly tied to the 8086 CPU. This interrupt is typically used for signaling catastrophic events (e.g., power failure). The Intel 8259A Programmable Interrupt Controller (PIC) provides control and vectoring for the next eight interrupt levels. As shown in Table 3, a selection of four priority processing modes is available for use in designing request processing configurations to match system requirements for efficient interrupt servicing with minimal latencies. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from all on-board I/O resources and from the MULTIBUS system bus. The PIC then resolves requests according to the selected mode and, if appropriate, issues an interrupt to the CPU. Any combination of interrupt levels may be masked via software, by storing a single byte in the interrupt mask register of the PIC. In systems requiring additional interrupt levels, slave 8259A PICs may be interfaced via the MULTIBUS system bus, to generate additional vector addresses, yielding a total of 65 unique interrupt levels.

Table 3. Programmable Interrupt Modes

Mode	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.
Polled	System software examines priority-encoded system interrupt status via interrupt status register.

Interrupt Request Generation

Interrupt requests to be serviced by the iSBC 86/XX boards may originate from 28 sources. Table 4 includes a list of devices and functions supported by interrupts. All interrupt signals are brought to the interrupt jumper matrix where any combination of interrupt sources may be strapped to the desired interrupt request level on the 8259A PIC or the NMI input to the CPU directly.

Power-Fail Control and Auxiliary Power

Control logic is also included to accept a power-fail interrupt in conjunction with the AC-low signal from the iSBC 635 and iSBC 640 Power Supply or equivalent, to initiate an orderly shut down of the system in the event of a power failure. Additionally, an active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences. An auxiliary power bus is also provided to allow separate power to RAM for systems requiring battery back-up of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

System Development Capabilities

The development cycle of iSBC 86/XX products can be significantly reduced and simplified by using either the System 86/330 or the Intellec Series Microcomputer Development Systems. The Assembler, Locating Linker, Library Manager, Text Editor and System Monitor are all supported by the ISIS-II disk-based operating system. To facilitate conversion of 8080A/8085A assembly language programs to run on the iSBC 86/XX boards, CONV-86 is available under the ISIS-II operating system.

IN-CIRCUIT EMULATOR

The Intellec ICE-86 In-Circuit Emulator provides the necessary link between the software development environment provided by the Intellec system and the "target" iSBC 86/XX execution system. In addition to providing the mechanism for loading executable code and data into the iSBC 86/XX boards, the ICE-86 In-Circuit Emulator provides a sophisticated command set to assist in debugging software and final integration of the user hardware and software.

PL/M-86

Intel's system's implementation language, PL/M-86, is standard in the System 86/330 and is also available as an Intellec Microcomputer Development System option. PL/M-86 provides the capability to program in algorithmic language and eliminates the need to manage register usage or allocate memory while still allowing explicit control of the system's resources when needed. FORTRAN 86 and PASCAL 86 are also available on Intellec or 86/330 systems.

Run-Time Support

Intel also offers two run-time support packages; iRMX 88 Realtime Multitasking Executive and the iRMX 86 Operating System. The iRMX 88 executive is a simple, highly configurable and efficient foundation for small, high performance applications. Its multitasking structure establishes a solid foundation for modular system design and provides task scheduling and management, intertask communication and synchronization, and interrupt servicing for a variety of peripheral devices. Other configurable options include terminal handlers, disk file system, debuggers and other utilities. The iRMX 86 Operating System is a high functional operating system with a very rich set of features and options based on an object-oriented architecture. In addition to being modular and configurable, functions beyond the nucleus include a sophisticated file management and I/O system, and powerful human interface. Both packages are easily customized and extended by the user to match unique requirements.

Table 4. Interrupt Request Sources

Device	Function	Number of Interrupts
MULTIBUS® interface	Requests from MULTIBUS® resident peripherals or other CPU boards	8; may be expanded to 64 with slave 8259A PICs on MULTIBUS® boards
8255A Programmable Peripheral Interface	Signals input buffer full or output buffer empty; also BUS INTR OUT general purpose interrupt from driver/terminator sockets	3
8251A USART	Transmit buffer empty and receive buffer full	2
8253 Timers	Timer 0, 1 outputs; function determined by timer mode	2
iSBX™ connectors	Function determined by iSBX™ MULTIMODULE™ board	4 (2 per iSBX™ connector)
Bus fail safe timer	Indicates addressed MULTIBUS® resident device has not responded to command within 6 msec	1
Power fail interrupt	Indicates AC power is not within tolerance	1
Power line clock	Source of 120 Hz signal from power supply	1
External interrupt	General purpose interrupt from auxiliary (P2) connector on backplane	1
iSBC™ 337 MULTIMODULE™ Numeric Data Processor	Indicates error or exception condition	1
Parity error	Indicates on-board RAM parity error from iSBC™ 303 parity MULTIMODULE™ board (iSBC™ 86/14 option)	1
Edge-level conversion	Converts edge triggered interrupt request to level interrupt	1
OR-gate matrix	Outputs of OR-gates on-board for multiple interrupts	2

SPECIFICATIONS

Word Size

INSTRUCTION — 8, 16, 24, or 32 bits
DATA — 8, 16 bits

System Clock

5.00 MHz or 8.00 MHz ±0.1% (jumper selectable)

Cycle Time

BASIC INSTRUCTION CYCLE

8 MHz — 750 ns
— 250 ns (assumes instruction in the queue)
5 MHz — 1.2 μsec
— 400 ns (assumes instruction in the queue)

NOTE: Basic instruction cycle is defined as the fastest instruction time (i.e., two clock cycles).

Memory Cycle Time

RAM — 750 ns
EPROM — Jumper selectable from 500 ns to 875 ns

Memory Capacity/Addressing

ON-BOARD EPROM

Device	Total Capacity	Address Range
2716	8K bytes	$FE000-FFFFF_H$
2732	16K bytes	$FC000-FFFFF_H$
2764	32K bytes	$F8000-FFFFF_H$
27128	64K bytes	$F0000-FFFFF_H$

NOTE: iSBC 86/XX EPROM sockets support JEDEC 24/28-pin standard EPROMs and RAMs.

ON-BOARD RAM

Board	Total Capacity	Address Range
iSBC 86/14	32K bytes	$0-07FFF_H$
iSBC 86/30	128K bytes	$0-1FFFF_H$

WITH MULTIMODULE™ RAM

Board	Total Capacity	Address Range
iSBC 300A (with iSBC 86/14)	64K bytes	$0-0FFFF_H$
iSBC 304 (with iSBC 86/30)	256K bytes	$0-3FFFF_H$

I/O Capacity

PARALLEL — 24 programmable lines using one 8255A.
SERIAL — 1 programmable line using one 8251A
iSBX™ MULTIMODULE™ — 2 iSBX boards

Serial Communications Characteristics

SYNCHRONOUS — 5-8 bit characters; internal or external character synchronization; automatic sync insertion

ASYNCHRONOUS — 5-8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detection

BAUD RATES

Frequency (kHz) (Software Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous	
		÷16	÷64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
2.4	2400	150	—
1.76	1760	110	—

NOTE: Frequency selected by I/O write of appropriate 16-bit frequency factor to baud rate register (8253 Timer 2).

Timers

INPUT FREQUENCIES

Reference: 2.46 MHz ±0.1% (0.041 μsec period, nominal); or 153.60 kHz ±0.1% (6.51 μsec period, nominal)

NOTE: Above frequencies are user selectable.

Event Rate: 2.46 MHz max

OUTPUT FREQUENCIES/TIMING INTERVALS

Function	Single Timer/Counter		Dual Timer/Counter (Cascaded)	
	Min	Max	Min	Max
Real-time Interrupt	1.63 μs	427.1 ms	3.26s	466.50 min
Programmable one-shot	1.63 μs	427.1 ms	3.26s	466.50 min
Rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Square-wave rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Software triggered strobe	1.63 μs	427.1 ms	3.26s	466.50 min
Hardware triggered strobe	1.63 μs	427.1 ms	3.26s	466.50 min
Event counter	—	2.46 MHz	—	—

Interfaces

MULTIBUS® — All signals TTL compatible
iSBX™ BUS — All signals TTL compatible
PARALLEL I/O — All signals TTL compatible
SERIAL I/O — RS232C compatible, configurable as a data set or data terminal

TIMER — All signals TTL compatible
INTERRUPT REQUESTS — All TTL compatible

Connectors

Interface	Double-Sided Pins	Centers (in.)	Mating Connectors
MULTIBUS® System	86	0.156	Viking 3KH43/9AMK12 Wire Wrap
iSBX™ Bus 8-Bit Data	36	0.1	iSBX 960-5
Parallel I/O (2)	50	0.1	3M 3415-000 Flat or TI H312125 Pins
Serial I/O	26	0.1	3M 3462-0001 Flat or AMP 88106-1 Flat

Line Drivers and Terminators

I/O DRIVERS — The following line drivers are all compatible with the I/O driver sockets on the iSBC 86/05 board

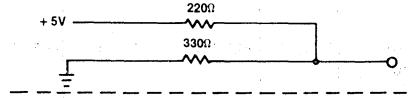
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

NOTE: I = inverting; NI = non-inverting; OC = open collector.

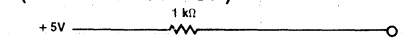
Port 1 of the 8255A has 20 mA totem-pole bidirectional drivers and 1 kΩ terminators

I/O TERMINATORS — 220Ω/330Ω divider or 1 kΩ pullup

220Ω/330Ω (iSBC™ 901 OPTION)



1 kΩ (iSBC™ 902 OPTION)



MULTIBUS® Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-State	32
Address	Tri-State	32
Commands	Tri-State	32
Bus Control	Open Collector	20

Physical Characteristics

WIDTH — 12.00 in. (30.48 cm)
HEIGHT — 6.75 in. (17.15 cm)
DEPTH — 0.70 in. (1.78 cm)
WEIGHT — 14 oz (388 gm)

Environmental Characteristics

OPERATING TEMPERATURE — 0°C to 55°C
RELATIVE HUMIDITY — to 90% (without condensation)

Electrical Characteristics

DC POWER REQUIREMENTS

Configuration	Current Requirements (All Voltages ±5%)		
	+5V	+12V	−12V
Without EPROM[1]	5.1A	25 mA	23 mA
RAM only[2]	600 mA	—	—
With 8K EPROM[3] (using 2716)	5.4A	25 mA	23 mA
With 16K EPROM[3] (using 2732)	5.5A	25 mA	23 mA
With 32K EPROM[3] (using 2764)	5.6A	25 mA	23 mA

NOTES:
1. Does not include power for optional ROM/EPROM, I/O drivers, and I/O terminators.
2. RAM chips powered via auxiliary power bus in power-down mode.
3. Includes power required for 4 ROM/EPROM chips, and I/O terminators installed for 16 I/O lines; all terminator inputs low.

Environmental Characteristics

OPERATING TEMPERATURE — 0°C to 55°C
RELATIVE HUMIDITY — to 90% (without condensation)

Reference Manual

144044-001 — iSBC 86/14 and iSBC 86/30 Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

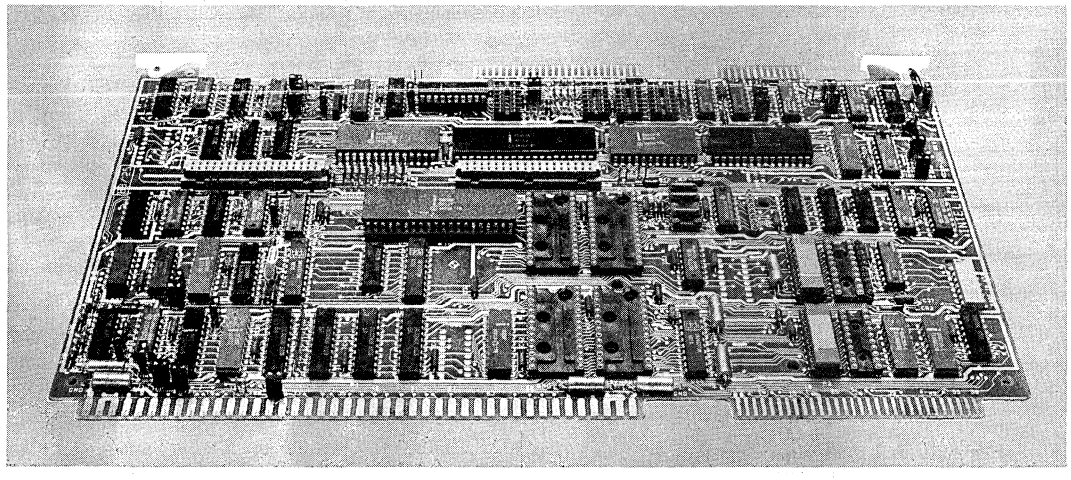
ORDERING INFORMATION

Part Number	Description
SBC 86/14	Single Board Computer
SBC 86/30	Single Board Computer

iSBC™ 88/25
SINGLE BOARD COMPUTER

- 8-bit 8088 Microprocessor operating at 5 MHz
- One megabyte addressing range
- Two iSBX™ bus connectors
- Optional Numeric Data Processor with iSBC 337 MULTIMODULE™ Processor
- 4K bytes of static RAM; expandable on-board to 16K bytes
- Sockets for up to 64K bytes of JEDEC 24/28-pin standard memory devices; expandable on-board to 128K bytes
- Programmable synchronous/asynchronous RS232C compatible serial interface with software selectable baud rates
- 24 programmable parallel I/O lines
- Two programmable 16-bit BCD or binary timers/event counters
- 9 Levels of vectored interrupt control, expandable to 65 levels
- MULTIBUS® interface for multimaster configurations and system expansion
- Supported by a complete family of single board computers, memory, digital and analog I/O, peripheral controllers, packaging and software

The iSBC 88/25 Single Board Computer is a member of Intel's complete line of OEM microcomputer systems which take full advantage of Intel's technology to provide economical, self-contained, computer-based solutions for OEM applications. The iSBC 88/25 board is a complete computer system on a single 6.75 × 12.00-in. printed circuit card. The CPU, system clock, read/write memory, nonvolatile read only memory, I/O ports and drivers, serial communications interface, priority interrupt logic and programmable timers, all reside on the board. The large control storage capacity makes the iSBC 88/25 board ideally suited for control-oriented applications such as process control, instrumentation, industrial automation, and many others.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: CREDIT, Index, Intel, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1981

FUNCTIONAL DESCRIPTION

Central Processing Unit

The central processor for the iSBC 88/25 board is Intel's 8088 CPU operating at 5 MHz. The CPU architecture includes four 16-bit byte addressable data registers, two 16-bit memory base pointer registers and two 16-bit index registers, all accessed by a total of 24 operand addressing modes for comprehensive memory addressing and for support of the data structures required for today's structured, high level languages, as well as assembly language.

Instruction Set

The 8088 instruction repertoire includes variable length instruction format (including double operand instructions), 8-bit and 16-bit signed and unsigned arithmetic operators for binary, BCD and unpacked ASCII data, and iterative word and byte string manipulation functions.

For enhanced numerics processing capability, the iSBC 337 MULTIMODULE Numeric Data Processor extends the architecture and data set. Over 60 numeric instructions offer arithmetic, trigonometric, transcendental, logarithmic and exponential instructions. Supported data types include 16, 32, and 64-bit integer, and 32 and 64-bit floating point, 18-digit packed BCD and 80-bit temporary.

Architectural Features

A 4-byte instruction queue provides pre-fetching of sequential instructions and can reduce the 750 nsec minimum instruction cycle to 250 nsec for queued instructions. The stack-oriented architecture readily supports modular programming by facilitating fast, simple, inter-module communication, and other programming constructs needed for asynchronous real-time systems. The memory expansion capabilities offer a 1 megabyte addressing range. The dynamic relocation scheme allows ease in segmentation of pure procedure and data for efficient memory utilization. Four segment registers (code, stack, data, extra) contain program loaded offset values which are used to map 16-bit addresses to 20-bit addresses. Each register maps 64K bytes at a time and activation of a specific register is controlled explicitly by program control and is also selected implicitly by specific functions and instructions. All Intel® languages support the extended memory capability, relieving the programmer of managing the megabyte memory space, yet allowing explicit control when necessary.

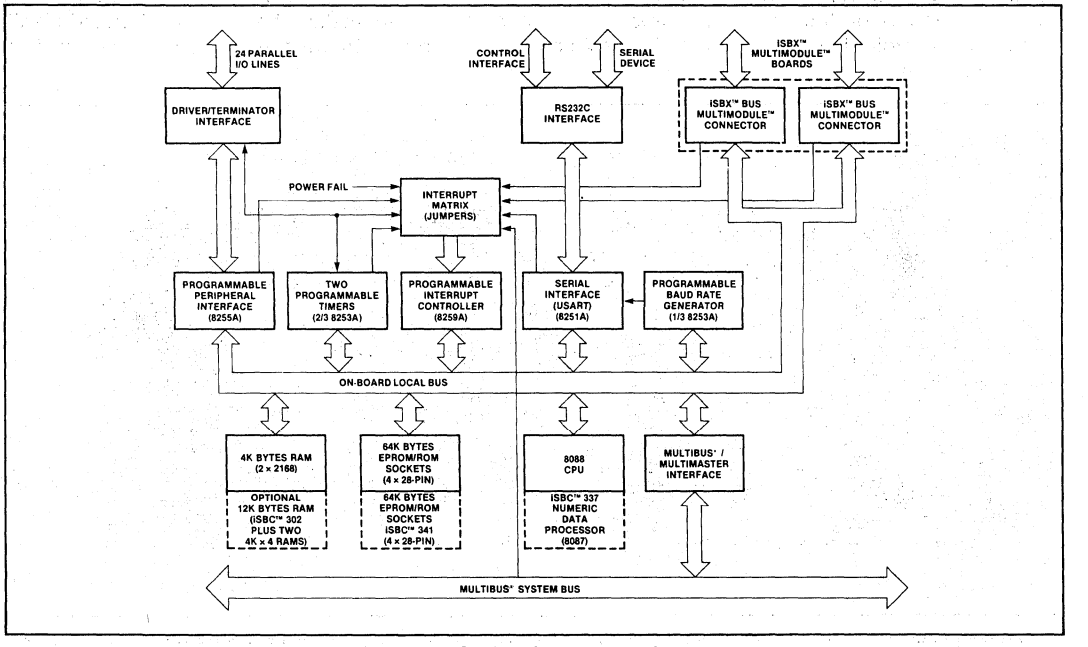


Figure 1. iSBC 88/25 Block Diagram

Memory Configuration

The iSBC 88/25 microcomputer contains 4K bytes of high-speed static RAM on-board. In addition, the on-board RAM may be expanded to 12K bytes via the iSBC 302 8K byte RAM module which mounts on the iSBC 88/25 board and then to 16K bytes by adding two 4K×4 RAM devices in sockets on the iSBC 302 module. All on-board RAM is accessed by the 8088 CPU with no wait states, yielding a memory cycle time of 800 nsec.

In addition to the on-board RAM, the iSBC 88/25 board has four 28-pin sockets, configured to accept JEDEC 24/28-pin standard memory devices. Up to 64K bytes of EPROM are supported in 16K-byte increments with Intel 27128 EPROMs. The iSBC 88/25 board is also compatible with the 2716, 2732, and 2764 EPROMs allowing a capacity of 8K, 16K, and 32K bytes, respectively. Other JEDEC standard pinout devices are also supported, including byte-wide static and integrated RAMs.

With the addition of the iSBC 341 MULTIMODULE EPROM option, the on-board capacity for these devices is doubled, providing up to 128K bytes of EPROM capacity on-board.

Parallel I/O Interface

The iSBC 88/25 Single Board Computer contains 24 programmable parallel I/O lines implemented using the Intel 8255A Programmable Peripheral Interface. The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports indicated in Table 1. In order to take advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators, allowing the selection of the appropriate combination of optional line drivers and terminators with the required drive/termination characteristics. The 24 programmable I/O lines and signal ground lines are brought out to a 50-pin edge connector.

Serial I/O

A programmable communications interface using the Intel 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the iSBC 88/25 board. A software selectable baud rate generator provides the USART with all common communication frequencies. The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. The 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The RS232C compatible interface on each board, in conjunction with the USART, provides a direct interface to RS232C compatible terminals, cassettes, and asynchronous and synchronous modems. The RS232C command lines, serial data lines and signal ground line are brought out to a 26-pin edge connector.

Programmable Timers

The iSBC 88/25 board provides three independent, fully programmable 16-bit interval timers/event counters utilizing the Intel 8253 Programmable

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					Control
		Unidirectional				Bidirectional	
		Input		Output			
		Latched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]

NOTE:

1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.

Interval Timer. Each counter is capable of operating in either BCD or binary modes. Two of these timers/counters are available to the systems designer to generate accurate time intervals under software control. Routing for the outputs and gate/trigger inputs of two of these counters is jumper selectable. The outputs may be independently routed to the 8259A Programmable Interrupt Controller and to the I/O terminators associated with the 8255A to allow external devices or an 8255A port to gate the timer or to count external events. The third interval timer in the 8253 provides the programmable baud rate generator for the iSBC 88/25 board RS232C USART serial port. The system software configures each timer independently to select the desired function. Seven functions are available as shown in Table 2. The contents of each counter may be read at any time during system operation.

Table 2. Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated. This function is extremely useful for generation of real-time clocks.
Programmable one-shot	Output goes low upon receipt of an external trigger edge or software command and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counter "window" has been enabled or an interrupt may be generated after N events occur in the system.

iSBX MULTIMODULE On-Board Expansion

Two 8-bit iSBX MULTIMODULE connectors are provided on the iSBC 88/25 microcomputer. Through these connectors, additional on-board I/O functions may be added. iSBX MULTIMODULES optimally support functions provided by VLSI peripheral components such as additional parallel and serial I/O, analog I/O, small mass storage device controllers (e.g., cassettes and floppy disks), and other custom interfaces to meet specific needs. By mounting directly on the single board computer, less interface logic, less power, simpler packaging, higher performance, and lower cost result when compared to other alternatives such as MULTIBUS form factor compatible boards. The iSBX connectors on the iSBC 88/25 provide all signals necessary to interface to the local on-board bus. A broad range of iSBX MULTIMODULE options are available in this family from Intel. Custom iSBX modules may also be designed for use on the iSBC 88/25 board. An iSBX bus interface specification and iSBX connectors are available from Intel.

MULTIBUS SYSTEM BUS AND MULTIMASTER CAPABILITIES

Overview

The MULTIBUS system bus is Intel's industry standard microcomputer bus structure. Both 8 and 16-bit single board computers are supported on the MULTIBUS structure with 24 address and 16 data lines. In its simplest application, the MULTIBUS system bus allows expansion of functions already contained on a single board computer (e.g., memory and digital I/O). However, the MULTIBUS structure also allows very powerful distributed processing configurations with multiple processors and intelligent slave I/O, and peripheral boards capable of solving the most demanding microcomputer applications. The MULTIBUS system bus is supported with a broad array of board level products, LSI interface components, detailed published specifications and application notes.

Expansion Capabilities

Memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS compatible expansion boards. Memory may be expanded by adding user specified combinations

of RAM boards, EPROM boards, or combination boards. Input/output capacity may be added with digital I/O and analog I/O expansion boards. Mass storage capability may be achieved by adding single or double density diskette controllers, or hard disk controllers. Modular expandable backplanes and cardcages are available to support multiboard systems.

Multimaster Capabilities

For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e., several CPUs and/or controllers logically sharing system tasks through communication of the system bus), the iSBC 88/25 board provides full MULTIBUS arbitration control logic. This control logic allows up to three iSBC 88/25 boards or other bus masters, including iSBC 80 and iSBC 86 family MULTIBUS compatible single board computers to share the system bus using a serial (daisy chain) priority scheme and allows up to 16 masters to share the MULTIBUS system bus with an external parallel priority decoder. In addition to the multiprocessing configurations made possible with multimaster capability, it also provides a very efficient mechanism for all forms of DMA (Direct Memory Access) transfers.

Interrupt Capability

The iSBC 88/25 board provides 9 vectored interrupt levels. The highest level is the NMI (Non-Maskable Interrupt) line which is directly tied to the 8088 CPU. This interrupt is typically used for signaling catastrophic events (e.g., power failure). The Intel 8259A Programmable Interrupt Controller (PIC) provides control and vectoring for the next eight interrupt levels. As shown in Table 3, a selection of four priority processing modes is available for use in designing request processing configurations to match system requirements for efficient interrupt servicing with minimal latencies. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from all on-board I/O resources and from the MULTIBUS system bus. The PIC then resolves requests according to the selected mode and, if appropriate, issues an interrupt to the CPU. Any combination of interrupt levels may be masked via software, by storing a single byte in the interrupt mask register of the PIC. In systems requiring additional interrupt levels, slave 8259A PICs may be interfaced via the MULTIBUS system bus, to generate additional vector addresses, yielding a total of 65 unique interrupt levels.

Table 3. Programmable Interrupt Modes

Mode	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.
Polled	System software examines priority-encoded system interrupt status via interrupt status register.

Interrupt Request Generation

Interrupt requests to be serviced by the iSBC 88/25 board may originate from 24 sources. Table 4 includes a list of devices and functions supported by interrupts. All interrupt signals are brought to the interrupt jumper matrix where any combination of interrupt sources may be strapped to the desired interrupt request level on the 8259A PIC or the NMI input to the CPU directly.

Power-Fail Control and Auxiliary Power

Control logic is also included to accept a power-fail interrupt in conjunction with the AC-low signal from the iSBC 635 and iSBC 640 Power Supply or equivalent, to initiate an orderly shut down of the system in the event of a power failure. Additionally, an active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences. An auxiliary power bus is also provided to allow separate power to RAM for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

System Development Capabilities

The development cycle of iSBC 88/25 products can be significantly reduced and simplified by

using the Intellec Series Microcomputer Development Systems. The Assembler, Locating Linker, Library Manager, Text Editor and System Monitor are all supported by the ISIS-II disk-based operating system. To facilitate conversion of 8080A/8085A assembly language programs to run on the iSBC 88/25 board, CONV-86 is available under the ISIS-II operating system.

IN-CIRCUIT EMULATOR

The ICE-88 In-Circuit Emulator provides the necessary link between the software development environment provided by the Intellec system and the "target" iSBC 88/25 execution system. In addition to providing the mechanism for loading executable code and data into the iSBC 88/25 board, the ICE-88 In-Circuit Emulator provides a sophisticated command set to assist in debugging software and final integration of the user hardware and software.

PL/M-86

Intel's system's implementation language, PL/M-86, is also available as an Intellec Microcomputer Development System option. PL/M-86 provides the capability to program in algorithmic language and eliminates the need to manage register usage or allocate memory while still allowing explicit control of the system's resources when needed.

Run-Time Support

Intel also offers two run-time support packages; iRMX 88 Realtime Multitasking Executive and the iRMX 86 Operating System. iRMX 88 is a simple, highly configurable and efficient foundation for small, high performance applications. Its multitasking structure establishes a solid foundation for modular system design and provides task scheduling and management, intertask communication and synchronization, and interrupt servicing for a variety of peripheral devices. Other configurable options include terminal handlers, disk file system, debuggers and other utilities. iRMX 86 is a high functional operating system with a very rich set of features and options based on an object-oriented architecture. In addition to being modular and configurable, functions beyond the nucleus include a sophisticated file management and I/O system, and powerful human interface. Both packages are easily customized and extended by the user to match unique requirements.

Table 4. Interrupt Request Sources

Device	Function	Number of Interrupts
MULTIBUS interface	Requests from MULTIBUS resident peripherals or other CPU boards	8; may be expanded to 64 with slave 8259A PIC's on MULTIBUS boards
8255A Programmable Peripheral Interface	Signals input buffer full or output buffer empty; also BUS INTR OUT general purpose interrupt from driver/terminator sockets	3
8251A USART	Transmit buffer empty and receive buffer full	2
8253 Timers	Timer 0, 1 outputs; function determined by timer mode	2
iSBX connectors	Function determined by iSBX MULTIMODULE board	4 (2 per iSBX connector)
Bus fail safe timer	Indicates addressed MULTIBUS resident device has not responded to command within 6 msec	1
Power fail interrupt	Indicates AC power is not within tolerance	1
Power line clock	Source of 120 Hz signal from power supply	1
External interrupt	General purpose interrupt from parallel port J1 connector	1
iSBC 337 MULTIMODULE Numeric Data Processor	Indicates error or exception condition	1

SPECIFICATIONS

Word Size

INSTRUCTION — 8, 16, 24, or 32 bits
DATA — 8 bits

System Clock

5.00 MHz or 4.17 MHz ± 0.1% (jumper selectable)

NOTE: 4.17 MHz required with the optional iSBC 337 module.

Cycle Time

BASIC INSTRUCTION CYCLE

At 5 MHz — 1.2 μsec
— 400 nsec (assumes instruction in the queue)

NOTES: Basic instruction cycle is defined as the fastest instruction time (i.e., two clock cycles).

Memory Cycle Time

RAM — 800 nsec (no wait states)
EPROM — Jumper selectable from 800 nsec to 1400 nsec

Memory Capacity/Addressing

ON-BOARD EPROM

Device	Total Capacity	Address Range
2716	8K bytes	$FE000-FFFFF_H$
2732	16K bytes	$FC000-FFFFF_H$
2764	32K bytes	$F8000-FFFFF_H$
27128	64K bytes	$F0000-FFFFF_H$

WITH iSBC 341 MULTIMODULE EPROM

Device	Total Capacity	Address Range
2716	16K bytes	$FC000-FFFFF_H$
2732	32K bytes	$F8000-FFFFF_H$
2764	64K bytes	$F0000-FFFFF_H$
27128	128K bytes	$E0000-FFFFF_H$

NOTES: iSBC 88/25 EPROM sockets support JEDEC 24/28-pin standard EPROMs and RAMs (2 sockets); iSBC 341 sockets also support E^2PROMs.

ON-BOARD RAM
4K bytes — $0-0FFF_H$

WITH iSBC 302 MULTIMODULE RAM
12K bytes — $0-2FFF_H$

WITH iSBC 302 MULTIMODULE BOARD AND TWO 4K x 4 RAM CHIPS
16K bytes — $0-3FFF_H$

I/O Capacity

PARALLEL — 24 programmable lines using one 8255A
SERIAL — 1 programmable line using one 8251A
iSBX MULTIMODULE — 2 iSBX MULTIMODULE boards

Serial Communications Characteristics

SYNCHRONOUS — 5-8 bit characters; internal or external character synchronization; automatic sync insertion

ASYNCHRONOUS — 5-8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detection

BAUD RATES

Frequency (kHz) (Software Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous	
		÷ 16	÷ 64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
2.4	2400	150	—
1.76	1760	110	—

NOTES:
Frequency selected by I/O write of appropriate 16-bit frequency factor to baud rate register (8253 Timer 2).

Timers

INPUT FREQUENCIES

Reference: 2.458 MHz ± 0.1% (406.9 nsec period, nominal); or 1.229 MHz ± 0.1% (813.8 nsec period, nominal); or 153.6 kHz ± 0.1% (6.510 μsec period, nominal)

NOTES:
Above frequencies are user selectable.

Event Rate: 2.46 MHz max

OUTPUT FREQUENCIES/TIMING INTERVALS

Function	Single Timer/Counter		Dual Timer/Counter (Two Timers Cascaded)	
	Min	Max	Min	Max
Real-time Interrupt	1.63 μs	427.1 ms	3.26s	466.50 min
Programmable one-shot	1.63 μs	427.1 ms	3.26s	466.50 min
Rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Square-wave rate generator	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Software triggered strobe	1.63 μs	427.1 ms	3.26s	466.50 min
Hardware triggered strobe	1.63 μs	427.1 ms	3.26s	466.50 min
Event counter	—	2.46 MHz	—	—

Interfaces

MULTIBUS — All signals TTL compatible
iSBX BUS — All signals TTL compatible
PARALLEL I/O — All signals TTL compatible
SERIAL I/O — RS232C compatible, configurable as a data set or data terminal
TIMER — All signals TTL compatible
INTERRUPT REQUESTS — All TTL compatible

Connectors

Interface	Double-Sided Pins (qty)	Centers (in.)	Mating Connectors
MULTIBUS System	86	0.156	Viking 3KH43/9AMK12 Wire Wrap
iSBX Bus 8-Bit Data	36	0.1	iSBX 960-5
Parallel I/O (2)	50	0.1	3M 3415-000 Flat or TI H312125 Pins
Serial I/O	26	0.1	3M 3462-0001 Flat or AMP 88106-1 Flat

Line Drivers and Terminators

I/O DRIVERS — The following line drivers are all compatible with the I/O driver sockets on the iSBC 88/25 board

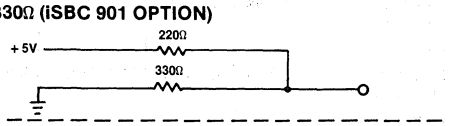
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

NOTES:
I = inverting; NI = non-inverting; OC = open collector.

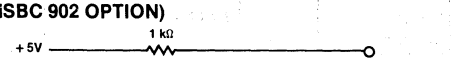
Port 1 of the 8255A has 32 mA totem-pole bidirectional drivers and 10 kΩ terminators

I/O TERMINATORS — 220Ω/330Ω divider or 1 kΩ pullup

220Ω/330Ω (iSBC 901 OPTION)



1 kΩ (iSBC 902 OPTION)



MULTIBUS Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-State	32
Address	Tri-State	24
Commands	Tri-State	32
Bus Control	Open Collector	20

Physical Characteristics

WIDTH — 12.00 in. (30.48 cm)
HEIGHT — 6.75 in. (17.15 cm)
DEPTH — 0.70 in. (1.78 cm)
WEIGHT — 14 oz (388 gm)

Electrical Characteristics

DC POWER REQUIREMENTS

Configuration	Current Requirements (All Voltages ±5%)		
	+5V	+12V	−12V
Without EPROM[1]	3.8A	25 mA	23 mA
RAM only[2]	104 mA		
With 8K EPROM[3] (using 2716)	4.3A	25 mA	23 mA
With 16K EPROM[3] (using 2732)	4.4A	25 mA	23 mA
With 32K EPROM[3] (using 2764)	4.4A	25 mA	23 mA

NOTES:
1. Does not include power for optional ROM/EPROM, I/O drivers, and I/O terminators.
2. RAM chips powered via auxiliary power bus in power-down mode. Does not include power for optional RAM.
3. Includes power required for 4 ROM/EPROM chips, and I/O terminators installed for 16 I/O lines; all terminator inputs low.

Environmental Characteristics

OPERATING TEMPERATURE — 0°C to 55°C
RELATIVE HUMIDITY — to 90% (without condensation)

Reference Manual

143825-001 — iSBC 88/25 Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

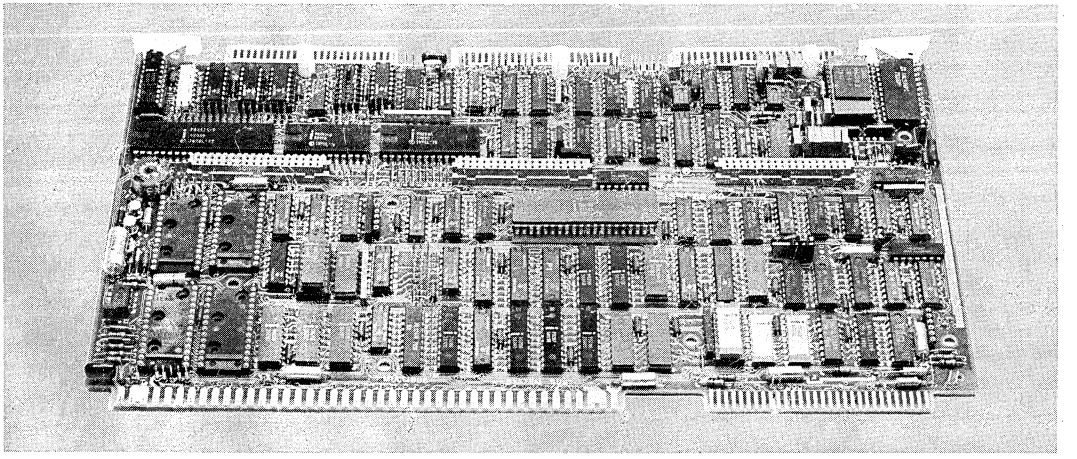
Part Number	Description
SBC 88/25	8-bit Single Board Computer with 4K bytes RAM

intel

iSBC 88/40
MEASUREMENT AND CONTROL COMPUTER

- **High performance 5 MHz iAPX 88/10 8-bit HMOS processor**
- **12-bit, 20 kHz analog-to-digital converter with programmable gain control**
- **16 differential/32 single-ended analog input channels**
- **Three iSBX MULTIMODULE connectors for analog, digital, and other I/O expansion**
- **4K bytes static RAM, expandable via iSBC 301 MULTIMODULE RAM to 8K bytes (1K byte dual-ported)**
- **Four ERPOM/E²PROM sockets for up to 32K bytes, expandable to 64K bytes with iSBC 341 expansion MULTIMODULE**
- **On-board 21-volt power supply for E²PROM modification under program control**
- **MULTIBUS Intelligent Slave or Multimaster**

The Intel iSBC 88/40 Measurement and Control Computer is a member of Intel's large family of Single Board Computers that takes full advantage of Intel's VLSI technology to provide an economical self-contained computer based solution for applications in the areas of process control and data acquisition. The on-board iAPX 88/10 processor with its powerful instruction set allows users of the iSBC 88/40 board to update process loops as much as 5–10 times faster than previously possible with other 8-bit microprocessors. For example, the high performance iSBC 88/40 can concurrently process and update 16 control loops in less than 200 milliseconds using a traditional PID (Proportional-Integral-Derivative) control algorithm. The iSBC 88/40 board consists of a 16 differential/32 single ended channel analog multiplexer with input protected circuits, A/D converter, programmable central processing unit, dual port and private RAM, read only memory sockets, interrupt logic, 24 channels of parallel I/O, three programmable timers and MULTIBUS control logic on a single 6.75 by 12.00-inch printed circuit card. The iSBC 88/40 board is capable of functioning by itself in a stand-alone system or as a multimaster or intelligent slave in a large MULTIBUS system.



FUNCTIONAL DESCRIPTION

Three Modes of Operation

The iSBC 88/40 Measurement and Control Computer (MACC) is capable of operating in one of three modes: stand-alone controller, bus multimaster or intelligent slave. A block diagram of the iSBC 88/40 Measurement and Control Computer is shown in Figure 1.

Stand-Alone Controller

The iSBC 88/40 Measurement and Control Computer may function as a stand-alone single board controller with CPU, memory and I/O elements on a single board. The on-board 4K bytes of RAM and up to 32K bytes of read only memory, as well as the analog-to-digital converter and programmable parallel I/O lines allow significant control and monitoring capabilities from a single board.

Bus Multimaster

In this mode of operation the iSBC 88/40 board may interface and control a wide variety of iSBC memory and I/O boards or even with additional iSBC 88/40 boards or other single board computer masters or intelligent slaves.

Intelligent Slave

The iSBC 88/40 board can perform as an intelligent slave to any Intel 8 or 16-bit MULTIBUS master CPU by not only offloading the master of the analog data collection, but it can also do a significant amount of pre-processing and decision making on its own. The distribution of processing tasks to intelligent slaves frees the system master to do other system functions. The dual port RAM with flag bytes for signalling allows the iSBC 88/40 board to process and store data without MULTIBUS memory or bus contention.

Central Processing Unit

The central processor unit for the iSBC 88/40 board is a powerful 8-bit HMOS iAPX 88/10 microprocessor. The 22.5 sq. mil. chip contains approximately 29,000 transistors and has a clock rate of 8 MHz. The architecture includes four (4) addressable data registers and two (2) 16-bit memory base pointer registers and two (2) 16-bit index registers, all accessed by a total of 24 operand addressing modes for complex data handling and very flexible memory addressing.

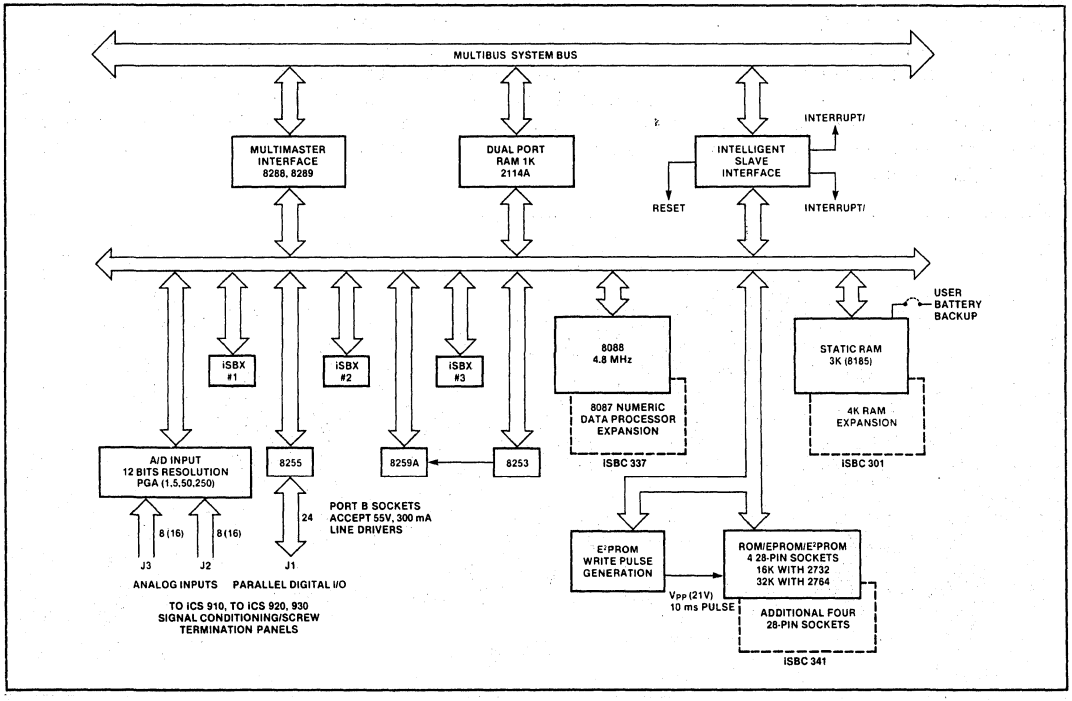


Figure 1. iSBC 88/40 Measurement and Control Computer Block Diagram

INSTRUCTION SET — The iAPX 88/10 instruction repertoire includes variable length instruction format (including double operand instructions), 8-bit and 16-bit signed and unsigned arithmetic operators for binary, BCD and unpacked ASCII data, and iterative word and byte string manipulation functions. The instruction set of the iAPX 88/10 is a superset of the 8080A/8085A family and with available software tools, programs written for the 8080A/8085A can be easily converted and run on the iAPX 88/10 processor. Programs can also be run that are implemented on the iAPX 86/10 with little or no modification.

ARCHITECTURAL FEATURES — A 4-byte instruction queue provides pre-fetching of sequential instructions and can reduce the 750 nsec minimum instruction cycle to 250 nsec for queued instructions. The stack oriented architecture facilitates nested subroutines and co-routines, reentrant code and powerful interrupt handling. The memory expansion capabilities offer a 1 megabyte addressing range. The dynamic relocation scheme allows ease in segmentation of pure procedure and data for efficient memory utilization. Four segment registers (code, stack, data, extra) contain program loaded offset values which are used to map 16-bit addresses to 20-bit addresses. Each register maps 64K bytes at a time and activation of a specific register is controlled explicitly by program control and is also selected implicitly by specific functions and instructions.

Bus Structure

The iSBC 88/40 single board computer has three buses: 1) an internal bus for communicating with on-board memory, analog-to-digital converter, iSBX MULTIMODULES and I/O options; 2) the MULTIBUS system bus for referencing additional memory and I/O options, and 3) the dual-port bus which allows access to RAM from the on-board CPU and the MULTIBUS system bus. Local (on-board) accesses do not require MULTIBUS communication, making the system bus available for use by other MULTIBUS masters (i.e. DMA devices and other single board computers transferring to additional system memory). This feature allows true parallel processing in a multiprocessor environment. In addition, the MULTIBUS interface can be used for system expansion through the use of other 8- and 16-bit iSBC computers, memory and I/O expansion boards.

RAM Capabilities

DUAL-PORT RAM — The dual-port RAM of the iSBC 88/40 board consists of 1K bytes of static RAM, implemented with Intel 2114A chips. The on-board base addess of this RAM is 00C00 (3K) normally; it is relocated to 01C00 (7K) when the iSBC 301 MULTIMODULE RAM is added to the protected RAM. The MULTIBUS port base address of the dual-port RAM can be jumpered to any 1K byte boundary in the 1M byte address space. The dual-port RAM can be accessed in a byte-wide fashion from the MULTIBUS system bus. When accessed from the MULTIBUS system bus, the dual-port RAM decode logic will generate INH1/ (Inhibit RAM) to allow dual-port RAM to overlay other system RAM. The dual-port control logic is designed to favor an on-board RAM access. If the dual-port is not currently performing a memory cycle for the MULTIBUS sytem port, only one wait state will be required. The on-board port may require more than one wait state if the dual-port RAM was busy when the on-board cycle was requested. The LOCK prefix facility of the iAPX 88/10 assembly language will disallow system bus accesses to the dual-port RAM. In addition, the on-board port to the dual-port RAM can be locked by other compatible MULTIBUS masters, which allows true symmetric semaphore operation. When the board is functioning in the master mode, the LOCK prefix will additionally disable other masters from obtaining the system bus.

PRIVATE RAM — In addition to the 1K byte dual-port RAM, there is a 3K byte section of private static RAM not accessible from the system bus. This RAM has a base address of 00000, and consists of three Intel 8185 RAM chips which are interfaced to the multiplexed address/data bus of the iAPX 88/10 microprocessor. Expansion of this private RAM from 3K to 7K bytes can be accomplished by the addition of an iSBC 301 MULTIMODULE RAM (4K bytes). When the 301 is added, protected RAM extends from 0 to 7K, and the base address of the dual-port RAM is relocated from 3K (00C00) to 7K (01C00). All protected RAM accesses require one wait state. The private RAM resides on the local on-board bus, which eliminates contention problems between on-board accesses to private RAM and system bus accesses to dual-port RAM. The private RAM can be battery backed (up to 16K bytes).

Additional RAM can be added by utilizing JEDEC-compatible static or pseudo-static RAMs in the available EPROM sockets.

Parallel I/O Interface

The iSBC 88/40 single board computer contains 24 programmable parallel I/O lines implemented using the Intel 8255A Programmable Peripheral In-

terface. The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports indicated in Table 1. There the I/O interface may be customized to meet specific peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Port 2 can also accept TTL compatible peripheral drives, such as 75461/462, 75471/472, etc. These are open collector, high voltage drivers (up to 55 volts) which can sink 300 mA. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 24 programmable I/O lines and signal ground lines are brought out to a 50-pin edge connector that mates with flat, woven, or round cable. This edge connector is also compatible with the Intel iCS 920 Digital I/O and iCS 930 AC Signal Conditioning/Termination Panels, for field wiring, optical isolation and high power (up to 3 amp) power drive.

EPROM Capabilities

Four (4) 28-pin sockets are provided for the use of Intel 2716s, 2732s, 2764s, future JEDEC-compatible 128K and 256K bit EPROMs and their respective ROMs. When using 2764s the on-board EPROM capacity is 32K bytes. Read only memory expansion is available through the use of the iSBC 341 EPROM/ROM memory expansion MULTIMODULE. When the iSBC 341 is used an additional four (4) EPROM sockets are made available, for a total iSBC 88/40 board capacity of 64K bytes EPROM with Intel 2764s.

E^2PROM Capabilities

The four 28-pin sockets can also accommodate Intel 2816 E^2PROMs, for dynamic storage of control loop setpoints, conversion parameters, or other data (or programs) that change periodically but must be kept in nonvolatile storage. To give the user dynamic control of this nonvolatile memory, the iSBC 88/40 board also contains an on-board DC to DC converter which under program control will furnish the voltage necessary for modifying the contents of Intel 2816/2815 E^2PROMs.

Timing Logic

The iSBC 88/40 board provides an 8253-5 Programmable Interval Timer, which contains three independent, programmable 16-bit timers/event counters. All three of these counters are available to generate time intervals or event counts under software control. The outputs of the three counters may be independently routed to the interrupt matrix. The inputs and outputs of timers 0 and 1 can be connected to parallel I/O lines on the J1 connector, where they replace 8255A port C lines. The third counter is also used for timing E^2PROM write operations.

Interrupt Capability

The iSBC 88/40 board provides 9 vectored interrupt levels. The highest level is the NMI (Nonmaskable Interrupt) line which is directly tied to the iAPX 88/10 CPU. This interrupt cannot be inhibited by software and is typically used for signalling catastrophic events (i.e., power failure). On servicing this interrupt, program control will be implicitly transferred through location 00008_H. The Intel 8259A Programmable Interrupt Controller (PIC) provides vectoring for the next eight interrupt levels. As shown in Table 2, a selection

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Latched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]

NOTE:

1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.

of four priority processing modes is available to the designer to match system requirements. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from the programmable parallel and/or iSBX interfaces, the programmable timers, the system bus, or directly from peripheral equipment. The PIC then determines which of the incoming requests is of the highest priority, determines whether this request is of higher priority than the level currently being serviced, and, if appropriate, issues an interrupt to the CPU. Any combination of interrupt levels may be masked, via software, by storing a single byte in the interrupt mask register of the PIC. The PIC generates a unique memory address for each interrupt level. These addresses are equally spaced at 4-byte intervals. This 32-byte block may begin at any 32-byte boundary in the lowest 1K bytes of memory*, and contains unique instruction pointers and code segment offset values (for expanded memory operation) for each interrupt level. After acknowledging an interrupt and obtaining a device identifier byte from the 8259A PIC, the CPU will store its status flags on the stack and execute an indirect CALL instruction through the vector location (derived from the device identifier) to the interrupt service routine.

***NOTE:** The first 32 vector locations are reserved by Intel for dedicated vectors. Users who wish to maintain compatibility with present and future Intel products should not use these locations for user-defined vector addresses.

Table 2. Programmable Interrupt Modes

Mode	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.
Polled	System software examines priority-encoded system interrupt status via interrupt status register.

INTERRUPT REQUEST GENERATION — Interrupt requests may originate from 26 sources. Two jumper selectable interrupt requests can be automatically generated by the programmable peripheral interface when a byte of information is ready to be transferred to the CPU (i.e., input buffer is full) or a byte of information has been transferred to a peripheral device (i.e., output buffer is empty). A jumper selectable request can be generated by each of the programmable timers. An additional interrupt request line may be jumpered directly from the parallel I/O driver terminator section. Eight prioritized interrupt request lines allow the iSBC 88/40 board to recognize and service interrupts originating from peripheral boards interfaced via the MULTIBUS system bus. The fail safe timer can be selected as an interrupt source. Also, interrupts are provided from the iSBX connectors (6), end-of-conversion, PFIN and from the power line clock.

Power-Fail Control

Control logic is also included to accept a power-fail interrupt in conjunction with the AC-low signal from the iSBC 635, iSBC 640, and iCS 645 Power Supply or equivalent.

iSBX MULTIMODULE Expansion Capabilities

Three iSBX MULTIMODULE connectors are provided on the iSBC 88/40 board. Up to three (3) single wide MULTIMODULE or one (1) double wide and one (1) single wide iSBX MULTIMODULE can be added to the iSBC 88/40 board. A wide variety of peripheral controllers, analog and digital expansion options are available. For more information on specific iSBX MULTIMODULES consult the Intel OEM Microcomputer System Configuration Guide.

Processing Expansion Capabilities

The addition of a iSBC 337 Multimodule Numeric Data Processor offers high performance integer and floating point math functions to users of the iSBC 88/40 board. The iSBC 337 incorporates the Intel 8087 and because of the MULTIMODULE implementation, it allows on-board expansion directly on the iSBC 88/40 board, eliminating the need for additional boards for floating point requirements.

MULTIBUS Expansion

Memory and I/O capacity may be expanded further and additional functions added using Intel MULTIBUS compatible expansion boards. Memory may be expanded by adding user specified combinations of RAM boards, EPROM boards, or memory combination boards. Input/output capacity may be increased by adding digital I/O and analog I/O MULTIBUS expansion boards. Mass storage capability may be acheived by adding single or double density diskette controllers, or hard disk con-

trollers either through the use of expansion boards and iSBX MULTIMODULES. Modular expandable backplanes and cardcages are available to support multiboard systems.

NOTE: Certain system restrictions may be incurred by the inclusion of some of the iSBC 80 family options in an iSBC 88/40 system. Consult the Intel OEM Microcomputer System Configuration Guide for specific data.

Analog Input Section

The analog section of the iSBC 88/40 board receives all control signals through the local bus to initiate channel selection, gain selection, sample and hold operation, and analog-to-digital conversion. See Figure 2.

INPUT CAPACITY — 32 separate analog signals may be randomly or sequentially sampled in single-ended mode with the 32 input multiplexers and a common ground. For noiser environments, differential input mode can be configured to achieve 16 separate differential signal inputs, or 32 pseudo differential inputs.

RESOLUTION — The analog section provides 12-bit resolution with a successive approximation analog-to-digital converter. For bipolar operation (−5 to +5 or −10 to +10 volts) it provides 11 bits plus sign.

SPEED — The A-to-D converter conversion speed is 50 μs (20 kHz samples per second). Combined with the programming interface, maximum throughput via the local bus and into memory will be 55 microseconds per sample, or 18 kHz samples per second, for a single channel, a random channel, or a sequential channel scan at a gain of 1, 5 ms at a gain of 5, 250 ms at a gain of 50, and 20 ms at a gain of 250. A-to-D conversion is initiated via a programmed command from the iAPX 88/10 central processor. Interrupt on end-of-conversion is a standard feature to ease programming and timing constraints.

ACCURACY — High quality components are used to achieve 12 bits resolution and accuracy of 0.035% full scale range ± ½ LSB. Offset is adjustable under program control to obtain a nominal ± 0.024% FSR ± ½ LSB accuracy at any fixed temperature between 0°C and 60°C (gain = 1). See specifications for other gain accuracies.

GAIN — To allow sampling of millivolt level signals such as strain gauges and thermocouples, gain is made configurable via user program commands up to 250× (20 millivolts full scale input range). User can select gain ranges of 1 (5V), 5 (1V), 50 (100 mV), 250 (20 mV) to match his application.

OPERATIONAL DESCRIPTION — The iSBC 88/40 single board computer addresses the analog-to-digital converter by executing IN or OUT instructions to the port address. Analog-to-digital conversions can be programmed in either of two modes: 1) start conversion and poll for end-of-conversion (EOC), or 2) start conversion and wait for interrupt at end of conversion. When the conversion is complete as signaled by one of the above techniques, INput instructions read two bytes (low and high bytes) containing the 12-bit data word as shown on the following page.

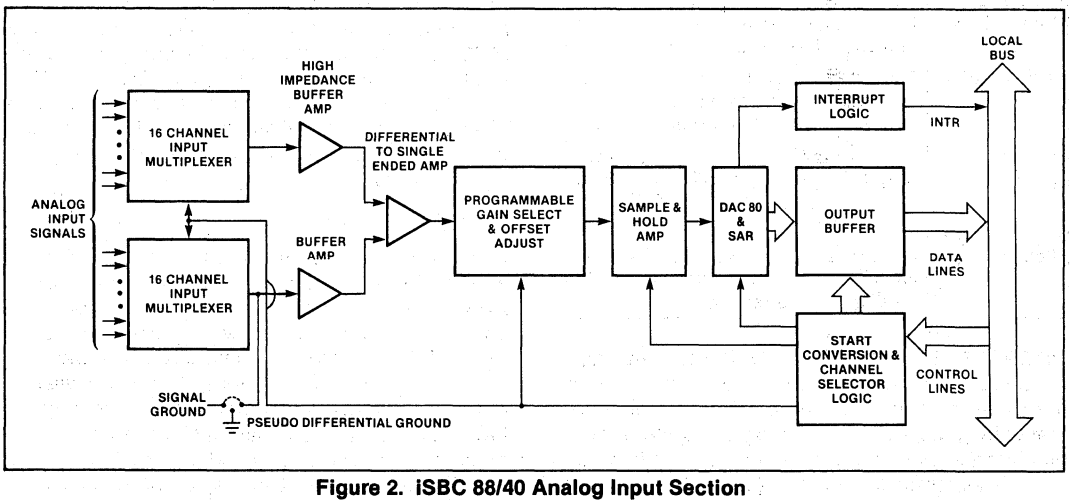


Figure 2. iSBC 88/40 Analog Input Section

Output Command — Select input channel and start conversion.

	GAIN		CONNECTOR		CHANNEL SELECT			
BIT POSITION	7	6	5	4	3	2	1	0
INPUT CHANNEL	G1	G2		J	C3	C2	C1	C0

Input Data — Read converted data (low byte) or Read converted data (high byte).

	DATA							
BIT POSITION	7	6	5	4	3	2	1	0
LOW/STATUS BYTE	D3	D2	D1	D0				EOC

	DATA							
HIGH BYTE	D11	D10	D9	D8	D7	D6	D5	D4

Offset Correction — At higher gains (×50, ×250) the voltage offset tempco in the A/D circuitry can sometimes cause unacceptable inaccuracies. To correct for this offset, one channel can be dedicated to be used as a reference standard. This channel can be read from the program to determine the amount of offset. The reading from this channel will then be subtracted from all other channel readings, in effect eliminating the offset tempco.

System Software Development

The development cycle of the iSBC 88/40 board may be significantly reduced using an Intel Intellec Microcomputer Development System with the optional iAPX 88/iAPX 86 Software Development package.

The iAPX 88/iAPX 86 Software Development package includes Intel's high-level programming language, PL/M 86. PL/M 86 provides the capability to program in a natural, algorithmic language and eliminates the need to manage register usage or allocate memory. PL/M 86 programs can be written in a much shorter time than assembly language programs for a given application.

SPECIFICATIONS

Word Size

Instruction — 8, 16, or 32 bits

Data — 8 bits

Instruction Cycle Time

417 nanoseconds for fastest executable instruction (assumes instruction is in the queue). 1.04 microseconds for fastest executable instruction (assumes instruction is not in the queue).

Memory Capacity

On-board ROM/EPROM/E²PROM

Up to 32K bytes; user installed in 2K, 4K or 8K byte increments or up to 64K if iSBC 341 MULTIMODULE EPROM option installed. Up to 8K bytes of E²PROM using Intel 2816s may be user-installed in increments of 2, 4 or 8K bytes.

On-board RAM

4K bytes or 8K bytes if the iSBC 301 MULTIMODULE RAM is installed. Integrity maintained during power failure with user-furnished batteries. 1K bytes are dual-ported.

Off-board Expansion

Up to 1 megabyte of user-specified combination of RAM, ROM, and EPROM.

Memory Addressing

On-board ROM/EPROM

FE000-FFFFF (using 2716 EPROMs)
FC000-FFFFF (using 2732 EPROMs)
F8000-FFFFF (using 2764 EPROMs)

On-board ROM/EPROM (With iSBC 341 MULTIMODULE EPROM option installed)

FC000-FFFFF (using 2716 EPROMs)
F8000-FFFFF (using 2732 EPROMs)
F0000-FFFFF (using 2764 EPROMs)

On-board RAM (CPU Access)

00000-00FFF

00000-01FFF (if iSBC 301 MULTIMODULE RAM option installed)

On-board RAM

Jumpers allow 1K bytes of RAM to act as slave RAM for access by another bus master. Addressing may be set within any 1K boundary in the 1-megabyte system address space.

Slave RAM Access

Average; 350 nanoseconds

Interval Timer

Output Frequencies —

Function	Single Timer		Dual Timers (Two Timers Cascaded)
	Min.	Max.	
Real-Time Interrupt Interval	0.977 μs	64 ms	69.9 minutes maximum
Rate Generator (Frequency)	15.625 Hz	1024 kHz	0.00024 Hz minimum

iAPX 88/10 CPU Clock

4.8 MHz ± 0.1%

I/O Addressing

All communications to parallel I/O ports, iSBX bus, A/D port, timers, and interrupt controller are via read and write commands from the on-board iAPX 88/10 CPU.

Interface Compatability

Parallel I/O — 24 programmable lines (8 lines per port); one port includes a bidirectional bus driver. IC sockets are included for user installation of line drivers and/or I/O terminators and/or peripheral drivers as required for interface ports.

iSBX Bus Connectors — Three iSBX bus connectors are provided. These connectors accept 8-bit iSBX MULTIMODULE boards. One set of the three iSBX MULTIMODULE connectors will accept a double wide iSBX MULTIMODULE board.

Interrupts

iAPX 88/10 CPU includes a non-maskable interrupt (NMI). NMI interrupt is provided for catastrophic events such as power failure. The on-board 8259A PIC provides 8-bit identifier of interrupting device to CPU. CPU multiplies identifier by four to derive vector address. Jumpers select interrupts from 26 sources without necessity of external hardware. PIC may be programmed to accommodate edge-sensitive or level-sensitive inputs.

Analog Input

16 differential (bipolar operation) or 32 single-ended (unipolar operation).

Full Scale Voltage Range — −5 to +5 volts (bipolar), 0 to +5 volts (unipolar).

Gain — Program selectable for gain of 1, 5, 50, or 250.

Resolution — 12 bits (11 bits plus sign for ±5, ±10 volts).

Accuracy — Including noise and dynamic errors

Gain	25°C
1	±0.05% FSR*
5	±0.075% FSR*
50	±0.085% FSR*
250	±0.12% FSR*

***NOTE:** FSR = Full Scale Range ± ½ LSB. Figures are in percent of full scale reading. At any fixed temperature between 0°C and 60°C, the accuracy is adjustable to ±0.05% of full scale.

Gain TC (at gain = 1) — 30 PPM (typical), 56 PPM (max) per degree centigrade, 40 PPM at other gains.

Offset TC — (in % of FSR/°C)	Gain	Offset TC (typical)
	1	0.0018%
	5	0.0036%
	50	0.024%
	250	0.12%

Sample and Hold-sample Time — 15 μs
Aperature-hold Aperature Time — 120 ns
Input Overvoltage Protection — 30 volts
Input Impedance — 20 megohms (min.)
Conversion Speed — 50 μs (max.) at gain = 1
Common Mode Rejection Ratio — 60 dB (min.)

Physical Characteristics

Width — 30.48 cm (12.00 in.)
Length — 17.15 cm (6.75 in.)
Height — 1.78 cm (0.7 in.)
2.82 cm (1.13 in.) with iSBC Memory Expansion, MULTIMODULES, iSBX Numeric Data Processor or iSBX MULTIMODULES.

Electrical Requirements

Power Requirements (Maximum) —

Configuration	+5V		+5V Aux		+12V		−12V	
	Typ	Max	Typ	Max	Typ	Max	Typ	Max
iSBC 88/40[1,2]	4	5.5	100	150	80	120	30	40

NOTES:
1. The current requirement includes one worst case (active-standby) EPROM current.
2. If +5V Aux is supplied by the iSBC 88/40 board, the total +5V current is the sum of the +5V and the +5V Aux.

Environmental Requirements

Operating Temperature — 0° to 55°C (32°C to 131°F) with 200 lfm air flow

Relative Humidity — to 90% without condensation

Equipment Supplied

The following are supplied with the iSBC 88/40 board:

a. Schematic diagram
b. Assembly drawing

Reference Manuals

124978-001 — iSBC 88/40 Measurement and Control Computer Hardware Reference Manual (NOT SUPPLIED).

Manuals may be ordered from an Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

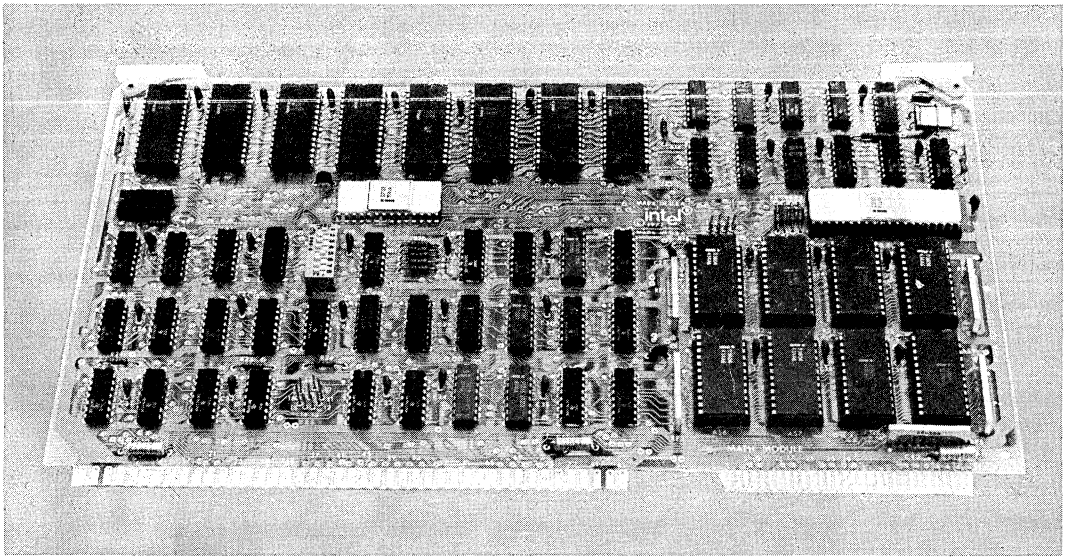
ORDERING INFORMATION

Part Number	Description
SBC 88/40	Measurement and Control Computer

iSBC 310A
HIGH SPEED MATHEMATICS UNIT

- **Provides iSBC 80 high speed mathematical functions**
- **Performs functions independently and concurrently with iSBC 80 Single Board Computer functions**
- **Fixed point integer arithmetic**
  - **— 16- and 32-bit format**
  - **— Multiply and divide**
  - **— Extended divide**
- **Floating point arithmetic**
  - **— Intel standard 32-bit format**
  - **— Add, subtract, multiply, divide**
  - **— Square and square root**
- **Compare and test operation**
  - **— Relative to zero**
  - **— Relative to floating point constant**
- **Float-to-fix and fix-to-float conversions**
- **Multimaster access — multiple masters may access iSBC 310A via system bus**
- **Single +5V power requirement**

The iSBC 310A High Speed Mathematics Unit is a member of Intel's complete family of OEM computers and expansion modules. The iSBC 310A acts as an intelligent slave processor to one or more iSBC computer masters as it performs its high speed arithmetic functions. It plugs into a standard iSBC 604/614 cardcage to interface directly into any iSBC 80 single board computer. Designed to increase the computational throughput of all computers in the iSBC 80 family, the iSBC 310A utilizes Intel's high speed Series 3000 Bipolar Microprocessor. The iSBC 310A performs arithmetic functions an order of magnitude faster than is possible with software routines. Standard operations include floating point add, subtract, multiply, divide, square, and square root; fixed point integer multiply, divide, and extended divide; and conversions between fixed and floating point representations, as well as test, compare, and argument exchange.



FUNCTIONAL DESCRIPTION

iSBC 80 single board computers communicate with the iSBC 310A using I/O and memory read/write commands. To pass arguments from the iSBC 80 to the iSBC 310A, a memory write command is used for each byte to be loaded into the iSBC 310A's working registers. An operation command is then given to the iSBC 310A by using an output instruction to pass the appropriate opcode. The mathematics unit will then perform the function independently from the single board computer; therefore, any iSBC 80 can continue to operate while the iSBC 310A is performing its arithmetic operations. Upon completion of its designated operation, the high speed mathematics unit notifies the iSBC 80 via an interrupt or by setting a status bit. The resultant data can then be read by the iSBC 80 via a memory read command to the proper memory address.

Arithmetic Functions

The iSBC 310A provides a full complement of arithmetic functions which operate on 16- and 32-bit unsigned fixed point integers, 32-bit signed fixed point integers, and 32-bit single precision floating point numbers. These functions are detailed in Table 1. The results of comparison operations are described in the operation results section.

Status Byte

The iSBC 310A may be operated in either an interrupt driven or polled mode. Three status indications are available:

Busy — The iSBC 310A is currently processing an arithmetic command, and cannot respond to further requests.

Complete — The iSBC 310A has completed an operation without an error. This line may be connected to an interrupt level via an on-board jumper.

Error — The iSBC 310A has completed an operation which results in an error condition. This line may be connected to an interrupt level via an on-board jumper.

Result Byte

After completion of an operation, a result byte may be read. This byte indicates the error conditions where applicable (see Specifications), and the results of a compare or test operation.

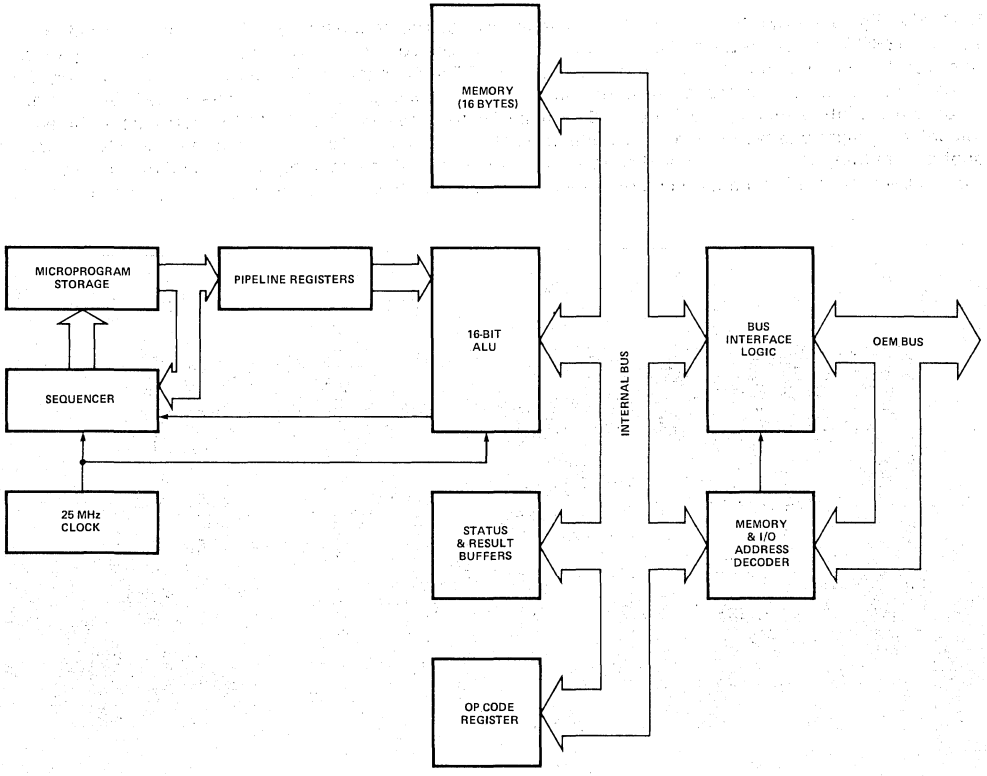


Figure 1. iSBC 310A Block Diagram Showing Functional Components

Table 1. ISBC 310A Arithmetic Functions

Operation	Op Code	Max Time[1] (μs)	Notes[2]
Fixed point multiply (MUL)	0	19	$M_1 = m_1 * m_1$
Fixed point divide (DIV)	1	28	$m_1 = m_1/M_2$, m_2 = remainder
Extended fixed point divide (EDIV)	E	94	$M_1 = M_1/m_2$, M_2 = remainder
Floating point multiply (FMUL)	2	91	$X_1 = X_1 * X_2$
Floating point divide (FDIV)	3	102	$X_1 = X_1/X_2$
Floating point add (FADD)	4	56	$X_1 = X_1 + X_2$
Floating point subtract (FSUB)	5	56	$X_1 = X_1 - X_2$
Square (FSQR)	6	91	$X_1 = X_1^2$
Square root (FSQRT)	7	199	$X_1 = \sqrt{X_1}$
Fixed-to-float-conversion (FLOAT)	8	89	$X_1 = N_1$
Float-to-fixed-conversion (FIX)	9	81	$N_1 = X_1$
Compare (FCOMP)	A	5	Compare X_1 and X_2
Test (FTST)	B	5	Compare X_1 and 0.0
Exchange (EXCH)	F	4	Exchange arguments (fixed or floating)

Notes
1. Does not include register setup time
2. m — 16-bit unsigned fixed point integers; M — 32-bit unsigned fixed point integer; N — 32-bit two's complement signed fixed integer; X — 32-bit single precision floating point number

SPECIFICATIONS

Arithmetic Functions

See Table 1

Formats

Single Precision Floating Point (32 Bits)

Memory Location		
Base Address (M)	$F_7 - F_0$	
M+1	$F_{15} - F_8$	
M+2	E_0	$F_{22} - F_{16}$
M+3	S	$E_7 - E_1$

where: S = sign bit
0 = positive
1 = negative
$E_2 - E_0$ = biased exponent (8 bits) (bias = $7F_H$)
$F_{22} - F_0$ = fraction (23 bits)

Note

F is always normalized (i.e., a "1" is assumed in the highest bit position), yielding an effective 24-bit fraction.

Fixed Point Integer (16-Bit)

Memory Location	
Base Address (M)	$F_7 - F_0$
M+1	$F_{15} - F_8$

where:
$F_{15} - F_0$ = 16-bit integer

Extended Precision Integer

Memory Location		
Base Address (M)	$F_7 - F_0$	
M+1	$F_{15} - F_8$	
M+2	$F_{23} - F_{16}$	
M+3	S	$F_{30} - F_{24}$

where: S = sign bit
$F_{30} - F_0$ = two's complement integer

Result Byte

Contains the following information:

7	6	5	4	3	2	1	0
=	>	<	R	R		ERR	

where: R is reserved for future use
= is equal (for FCOMP and FTST)
> is greater than (for FCOMP and FTST)
< is less than (for FCOMP and FTST)

and: ERR is a 3-bit error code that specifies one of the following error conditions:

000 No error
001 Divide by zero
010 Square root of negative number
011 Overflow
100 Underflow
101 First argument valid
110 Second argument valid
111 Reserved

Status Byte

Contains the following information:

7	6	5	4	3	2	1	0
R	R	R	R	R	E	C	B

where: R is reserved for future use
B is busy
C is operation complete without error
E is operation complete with error

Addressing

I/O Addressing — Used to pass operation codes, memory address boundaries, and result and status bytes between host processor and iSBC 310A.

Port Address	Output	Input
Base (P)	OP CODE	R
P+1	MEM LOW	Result byte
P+2	MEM HIGH	R
P+3	R	R
P+4	R	R
P+5	R	R
P+6	R	R
P+7	R	Status byte

where: P = I/O base address of X0 or X8 (where X = any hex digit)
R = reserved for iSBC 310A usage
OP CODE = mathematic commands (see Table 1)
MEM LOW, MEM HIGH = programmable base address (see Memory Addressing)

Memory Addressing — Sixteen memory locations are used; the first eight are used for argument/result storage; the second eight are reserved for future use. Memory addresses are assigned from the host processor via an I/O output instruction (see I/O Addressing). MEM LOW (the lower address byte) must be X0 (where X is any hex digit). MEM HIGH (the upper address byte) may be any value.

Interrupts

Interrupts are generated on operation complete and operation error. Either one or both interrupts may be connected to any of the 8 interrupt levels on the iSBC 80 bus via jumper selection.

Bus Interface

All signals are TTL compatible.

Bus Connector

Bus Connector — 86-pin, double-sided PC edge connector with 0.156-in. contact centers.

Mating Connector — Viking 3KH43/9AMK12

Physical Characteristics

Width — 12.00 in (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 12 oz (340.5 gm)

Electrical Characteristics

DC Power Requirements

V_{CC}	I_{CC}
5V ±5%	6.7A max; 4.9A typ

Environmental Characteristics

Operating Temperature — 0°C to 55°C

Equipment Supplied

High speed mathematics units
Standard preprogrammed ROMs (installed)
Schematics
Assembly drawing

Reference Manual

9800410A — iSBC 310A Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
SBC 310A	High Speed Mathematics Unit

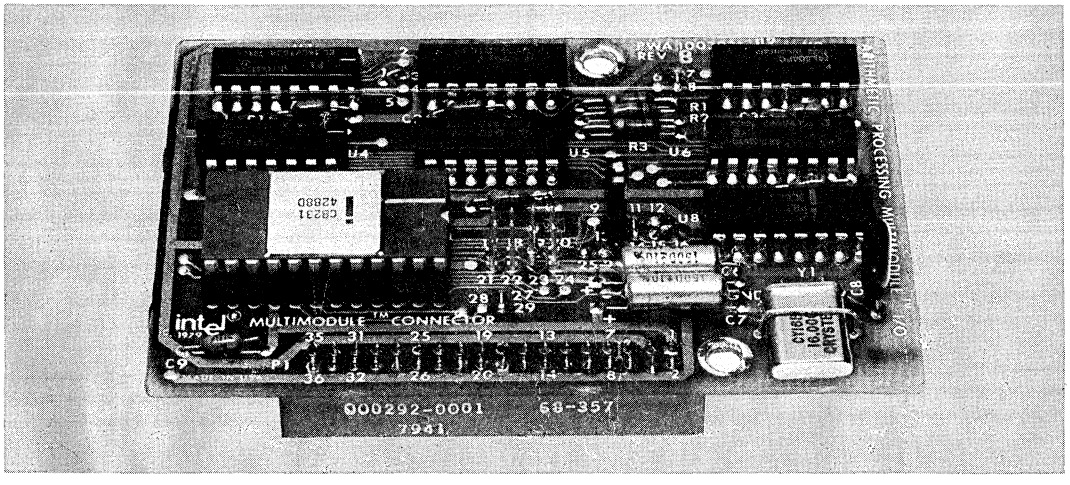
iSBX™ Multimodule™ Boards

3

iSBX 331
FIXED/FLOATING POINT MATH MULTIMODULE BOARD

- **iSBX bus compatible high speed fixed/floating point math expansion**
- **4 MHz operation**
- **Fixed point single and double precision (16/32-bit)**
- **Floating point double precision (32-bit)**
- **Binary data formats**
- **Add, subtract, multiply and divide**
- **Trignometric and inverse trigonometric functions**
- **Square root, log, and exponential functions**
- **Float-to-fixed and fixed-to-float conversions**
- **End of operation interrupt**
- **Software reset control**
- **Low power requirements**
- **iSBX bus on-board expansion eliminates MULTIBUS system bus latency and increases system throughput**

The Intel® iSBX 331 Fixed/Floating Point Math MULTIMODULE Board is a member of Intel's new line of iSBX bus compatible MULTIMODULE products. The iSBX MULTIMODULE board plugs directly into any iSBX bus compatible host board offering low cost incremental on-board expansion. As a result, any iSBX bus compatible host board may be expanded to perform high speed math computations, affording up to a 40× improvement in speed compared to software math. The iSBX 331 module performs single/double (16/32-bit) precision fixed point plus double (32-bit) precision floating point arithmetic operations. In addition, the module performs transcendental, data manipulation, and fixed to float/float to fixed point conversion operations. The command operations run entirely independent of the host board permitting efficient concurrent processing. The iSBX board is closely coupled to the host board through the iSBX bus, and as such, offers maximum on-board performance and frees MULTIBUS system traffic for other system resources. Incremental power dissipation is minimal requiring only 2.73 watts.



FUNCTIONAL DESCRIPTION

The iSBX 331 module uses the Intel 8231 Arithmetic Processing Unit (APU) to accomplish high speed (4 MHz) math operation. The system software may communicate with the iSBX 331 module across the iSBX bus using I/O read/write commands. All transfers, including operand, result, status, and command information, take place over an 8-bit bidirectional data bus. Operands are pushed onto an internal stack and commands are issued to perform operations on the data. Results are then available from the stack. A status byte may be read to monitor execution completion and the nature of the result (zero, sign, or errors). In addition, control logic is included on the iSBX 331 module to facilitate single instruction software reset control.

Command Functions

The iSBX 331 module commands fall into three categories: double precision floating point, single precision fixed point, and double precision fixed point (see Table 1). There are four arithmetic operations that can be performed in either fixed or floating point numbers: add, subtract, multiply, and divide. These operations require two operands. The 8231 assumes these operands are located in the internal stack as Top of Stack (TOS) and Next on Stack (NOS). The result will always be returned to TOS. There are four types of transcendental operations that can be performed in floating point numbers: trigonometric functions, logarithms, exponentials, and square roots. The results of these operations will be returned to TOS. There are four types of data manipulation operations that can be performed in either fixed or floating point numbers: sign change of TOS, exchange of TOS and NOS and copying or popping operands onto or off of TOS. Fixed to floating point conversion can be performed on floating point instructions and floating point to fixed point conversion can be performed on fixed point instructions.

The execution times of the commands are shown in Table 2.

Interrupt Requests

There is one interrupt line from the APU that may generate an interrupt request to the host: END (MINTRI). The END interrupt line is active upon command completion. The END signal is cleared by a reset or status register read.

Installation

The iSBX 331 module plugs directly into the female iSBX connector on the host board. The module is then secured at one additional point with nylon hardware to insure the mechanical security of the assembly (see Figures 1 and 2).

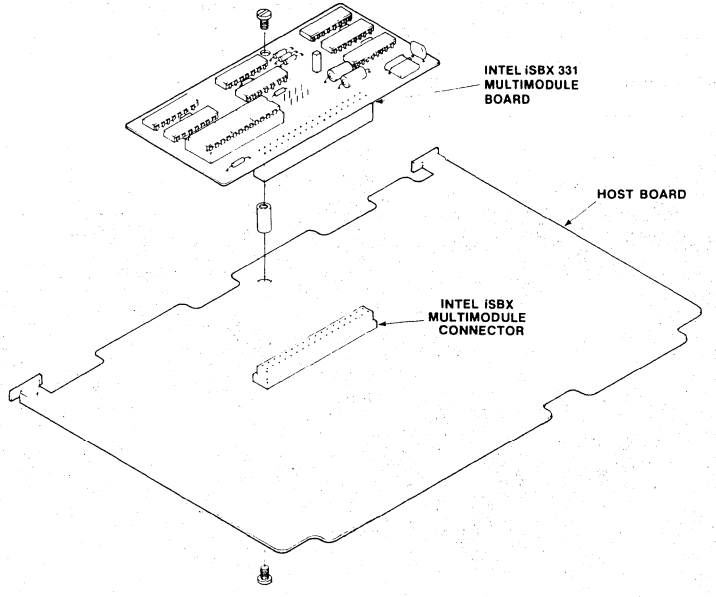


Figure 1. Installation of iSBX 331 Module on a Host Board

Table 1. Command Summary

Double Precision Floating Point Instructions (32-Bit)

Instruction	Description	Hex Code	Stack Contents After Execution[1] A B C D	Status Flags Affected[3]
ACOS	Inverse Cosine of A	0 6	R U U U	S, Z, E
ASIN	Inverse Sine of A	0 5	R U U U	S, Z, E
ATAN	Inverse Tangent of A	0 7	R B U U	S, Z
CHSF	Sign Change of A	1 5	R B C D	S, Z
COS	Cosine of A (radians)	0 3	R B U U	S, Z
EXP	e^A Function	0 A	R B U U	S, Z, E
FADD	Add A and B	1 0	R C D U	S, Z, E
FDIV	Divide B by A	1 3	R C D U	S, Z, E
FLTD	32-Bit Fixed to Floating Point Conversion	1 C	R B C U	S, Z
FLTS	16-Bit Fixed to Floating Point Conversion	1 D	R B C U	S, Z
FMUL	Multiply A and B	1 2	R C D U	S, Z, E
FSUB	Subtract A from B	1 1	R C D U	S, Z, E
LOG	Common Logarithm (base 10) of A	0 8	R B U U	S, Z, E
LN	Natural Logarithm of A	0 9	R B U U	S, Z, E
POPF	Stack Pop	1 8	B C D A	S, Z
PTOF	Stack Push	1 7	A A B C	S, Z
PUPI	Push π onto Stack	1 A	R A B C	S, Z
PWR	B^A Power Function	0 B	R C U U	S, Z, E
SIN	Sine of A (radians)	0 2	R B U U	S, Z
SQRT	Square Root of A	0 1	R B C U	S, Z, E
TAN	Tangent of A (radians)	0 4	R B U U	S, Z, E
XCHF	Exchange A and B	1 9	B A C D	S, Z

Double Precision Fixed Point Instructions (32-Bit)

Instruction	Description	Hex Code	Stack Contents After Execution[1] A B C D	Status Flags Affected[3]
CHSD	Sign Change of A	3 4	R B C D	S, Z, O
DADD	Add A and B	2 C	R C D A	S, Z, C, E
DDIV	Divide B by A	2 F	R C D U	S, Z, E
DMUL	Multiply A and B (R = lower 32 bits)	2 E	R C D U	S, Z, O
DMUU	Multiply A and B (R = upper 32 bits)	3 6	R C D U	S, Z, O
DSUB	Subtract A from B	2 D	R C D A	S, Z, C, O
FIXD	Floating to Fixed Point Conversion	1 E	R B C U	S, Z, O
POPD	Stack Pop	3 8	B C D A	S, Z
PTOD	Stack Push	3 7	A A B C	S, Z
XCHD	Exchange A and B	3 9	B A C D	S, Z

Table 1. Command Summary (continued)

Single Precision Fixed Point Instructions (16-Bit)

Instruction	Description	Hex Code	Stack Contents After Execution[2] A_U A_L B_U B_L C_U C_L D_U D_L	Status Flags Affected[3]
CHSS	Change Sign of A_U	7 4	R A_L B_U B_L C_U C_L D_U D_L	S, Z, O
FIXS	Floating to Fixed Point Conversion	1 F	R B_U B_L C_U C_L U U U	S, Z, O
POPS	Stack Pop	7 8	A_L B_U B_L C_U C_L D_U D_L A_U	S, Z
PTOS	Stack Push	7 7	A_U A_U A_L B_U B_L C_U C_L D_U	S, Z
SADD	Add A_U and A_L	6 C	R B_U B_L C_U C_L D_U D_L A_U	S, Z, C, E
SDIV	Divide A_L by A_U	6 F	R B_U B_L C_U C_L D_U D_L U	S, Z, E
SMUL	Multiply A_L by A_U (R = lower 16 bits)	6 E	R B_U B_L C_U C_L D_U D_L U	S, Z, E
SMUU	Multiply A_L by A_U (R = upper 16 bits)	7 6	R B_U B_L C_U C_L D_U D_L U	S, Z, E
SSUB	Subtract A_U from A_L	6 D	R B_U B_L C_U C_L D_U D_L A_U	S, Z, C, E
XCHS	Exchange A_U and A_L	7 9	A_L A_U B_U B_L C_U C_L D_U D_L	S, Z
NOP	No Operation	0 0	A_U A_L B_U B_L C_U C_L D_U D_L	

NOTES:

1. The stack initially is composed of four 32-bit numbers (A, B, C, D). A is equivalent to Top Of Stack (TOS) and B is Next On Stack (NOS). Upon completion of a command the stack is composed of: the result (R); undefined (U); or the initial contents (A, B, C, or D).
2. The stack initially is composed of eight 16-bit numbers (A_U, A_L, B_U, B_L, C_U, C_L, D_U, D_L). A_U is the TOS and A_L is NOS. Upon completion of a command the stack is composed of: the result (R); undefined (U); or the initial contents (A_U, A_L, B_U, B_L,...).
3. Nomenclature: Sign (S); Zero (Z); Overflow (O); Carry (C); Error Code Field (E).

Table 2. Command Execution Times

Command Mnemonic	μSeconds	Command Mnemonic	μSeconds
SADD	4.25	ASIN	1917
SSUB	7.5	ACOS	1933.5
SMUL	21-23.5	ATAN	1501.5
SMUU	20-24.5	LOG	1118.5-1783
SDIV	21-23.5	LN	1074.5-1739
DADD	5.25	EXP	948.5-1219.5
DSUB	9.5	PWR	2072.5-3008
DMUL	48.5-52.5	NOP	1
DMUU	45.5-54.5	CHSS	5.75
DDIV	52	CHSD	6.75
FIXS	23-54	CHSF	4.5
FIXD	25-86.5	PTOS	4
FLTS	24.5-46.5	PTOD	5
FLTD	24.5-94.5	PTOF	5
FADD	13.5-92	POPS	2.5
FSUB	17.5-92.5	POPD	3
FMUL	36.5-42	POPF	3
FDIV	38.5-46	XCHS	4.5
SQRT	200	XCHD	6.5
SIN	1116	XCHF	6.5
COS	1029.5	PUPI	4
TAN	1438.5		

NOTE: Assumes 4 MHz operation.

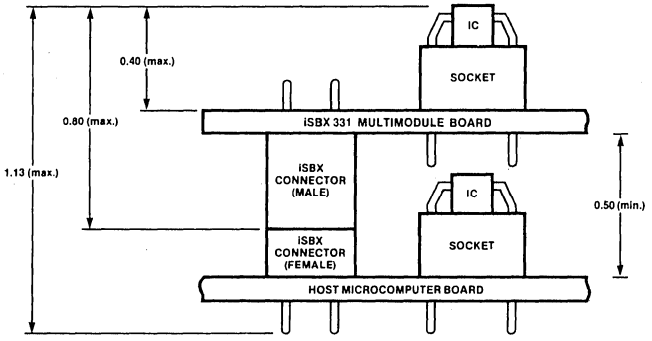


Figure 2. iSBX 331 MULTIMODULE Board Mounting Clearances (Inches)

SPECIFICATIONS

Word Size

Data—8 bits.

On-Board Clock Rate

4.0 MHz ±0.1%.

I/O Addressing

Function	Type of Operation	iSBX Connector Port Address
Data Transfer	Read or Write	X0, X2, X4, or X6
Command Transfer	Write	X1, X3, X5, or X7
Status Transfer	Read	X1, X3, X5, or X7
Reset	Write	X8 through XF

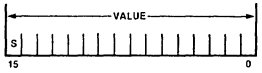
NOTE:
The port addresses are determined on the host iSBC microcomputer. Refer to the Hardware Reference Manual for your host iSBC microcomputer to determine the first digit (X) of the connector port addresses.

Arithmetic Functions

See Table 1.

Data Formats

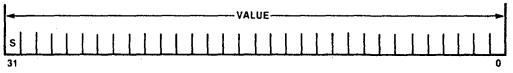
Single Precision Fixed Point (16 bits)



Bit 15: S = Sign of the operand. Positive values are represented by a sign bit of zero (S = 0). Negative values are represented by the two's complement of the corresponding positive value with a sign bit equal to 1 (S = 1).

Bits 0-14: Values in the range from −32, 768 to +32, 767.

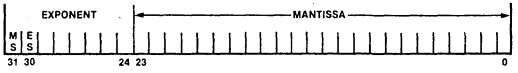
Double Precision Fixed Point (32 bits)



Bit 31: S = Sign of operand. Positive values are represented by a sign of zero (S = 0). Negative values are represented by the two's complement of the corresponding positive value with a sign bit equal to 1 (S = 1).

Bits 0-30: Values in the range from −2, 147, 483, 648 to +2, 147, 483, 647.

Double Precision Floating Point (32 bits)



Bit 31: MS = Sign of the mantissa. 1 represents negative and 0 represents positive.

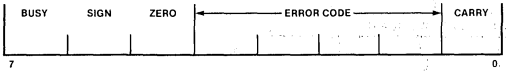
Bits 24-30: ES = the exponent expressed as a two's complement 7-bit value having a range of −64 to +63.

Bits 0-23: The mantissa is expressed as a 24-bit (fractional) value. The 8231 APU requires that floating point data be represented by a fractional mantissa value between 0.5 and 1 multiplied by 2 raised to an appropriate power (exponent). This is expressed as follows:

$$\text{Value} = \text{mantissa} \times 2^{\text{exponent}}$$

Device Status

Device status is provided by means of an internal status register whose format is shown below:



BUSY: Indicates that 8231 is currently executing a command (1 = Busy)

SIGN: Indicates that the value on the top of stack is negative (1 = Negative)

ZERO: Indicates that the value on the top of stack is zero (1 = Value is zero)

ERROR CODE: This field contains an indication of the validity of the result of the last operation. The error codes are:

- 0000 — No error
- 1000 — Divide by zero
- 0100 — Square root or log of negative number
- 1100 — Argument of inverse sine, cosine, or e^x too large
- XX10— Underflow
- XX01— Overflow

CARRY: Previous operation resulted in carry or borrow from most significant bit. (1 = Carry/Borrow, 0 = No Carry/No Borrow.)

If the BUSY bit in the status register is a one, the other status bits are not defined; if zero, indicating not busy, the operation is complete and the other status bits are defined as given above.

Access Time

Read—1900 ns (max.)
Write—1900 ns (max.)

NOTE:
Actual transfer speed is dependent upon the cycle time of the host microcomputer. The listed times assume no operation in progress. If an operation is executing when an access is attempted, the command execution time must be added to the above times for all accesses except status read.

Interrupts

One interrupt request may originate from the APU indicating command completion (END).

Interface

iSBX Bus—All signals TTL compatible

Physical Characteristics

Width—6.35 cm (2.50 in.)
Length—9.40 cm (3.70 in.)
Height*—2.04 cm (0.80 in.) iSBX 331 Board
—2.86 cm (1.13 in.) iSBX 331 Board + Host Board
Weight—51 gm (1.79 oz)

*See Figure 2.

Electrical Characteristics

DC Power Requirements

V_{CC} = +5V ±5% I_{CC} = 365 mA max.
V_{DD} = +12V ±5% I_{DD} = 75 mA max.

Environmental

Operating Temperature—0°C to 55°C

Free moving air across the base board and iSBX board.

Reference Manual

142668-01—iSBX 331 Floating Point Math MULTIMODULE Board (NOT SUPPLIED)

Reference manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

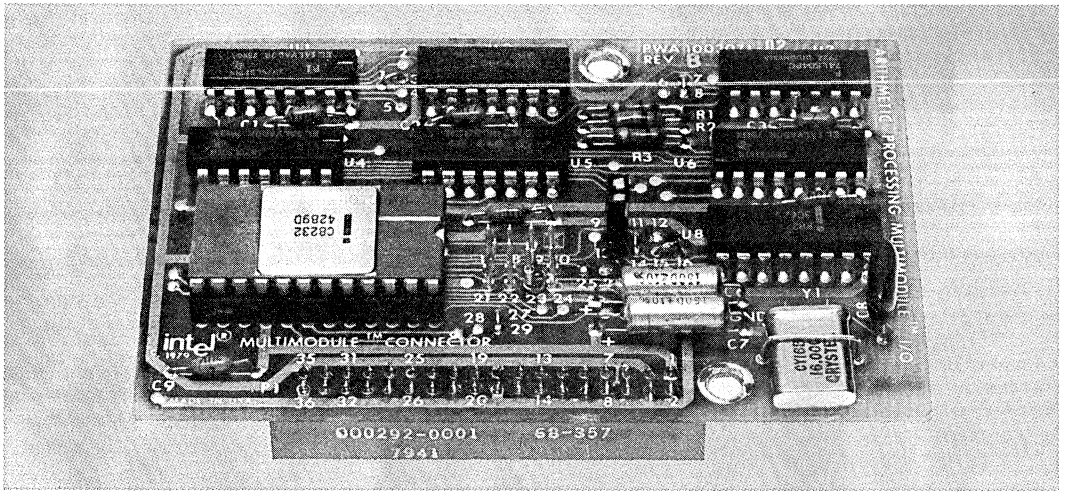
ORDERING INFORMATION

Part Number	Description
SBX 331	Fixed/Floating Point Math MULTIMODULE Board

iSBX 332
FLOATING POINT MATH MULTIMODULE BOARD

- **iSBX bus compatible high speed floating point math expansion**
- **4 MHz operation**
- **Compatible with proposed IEEE format and existing Intel floating point standard**
- **Single (32-bit)/double (64-bit) precision arithmetic and data manipulation commands**
- **Performs functions independently and concurrently with the MULTIBUS host board**
- **Add, subtract, multiply and divide functions**
- **End-of-operation and error interrupts**
- **Software reset control**
- **Accessed as I/O port locations**
- **Low power requirements**
- **iSBX bus on-board expansion eliminates MULTIBUS system bus latency and increases system throughput**

The Intel® iSBX 332 Floating Point Math MULTIMODULE Board is a member of Intel's new line of iSBX bus compatible MULTIMODULE products. The iSBX MULTIMODULE board plugs directly into any iSBX bus compatible host board offering incremental on-board expansion. The iSBX 332 module performs single (32-bit) and double (64-bit) precision floating point add, subtract, multiply, and divide functions compatible with the proposed IEEE floating point standard. The command operations run entirely independent of the host board permitting efficient concurrent processing. The iSBX board is closely coupled to the host board through the iSBX bus, and as such, offers maximum on-board performance and frees MULTIBUS system traffic for other system resources. In addition, incremental power dissipation is minimal requiring only 2.73 watts



AFN-01383A

FUNCTIONAL DESCRIPTION

The iSBX 332 module uses the Intel® 8232 Floating Point Processor (FPP) to accomplish high speed math operation. The system software may communicate with the iSBX 332 module across the iSBX bus using I/O read/write commands. All transfers, including operand, result, status, and command information, take place over an 8-bit bidirectional data bus. Operands are pushed onto an internal stack and commands are issued to perform operations on the data stack. Results are then available to be retrieved from the stack. A status byte may be read to monitor execution completion and the nature of the result (zero, sign, or errors). In addition, control logic is included on the iSBX 332 module to facilitate single instruction software reset control.

Command Functions

The iSBX 332 module commands fall into three categories: single precision arithmetic, double precision arithmetic and data manipulation (see Table 1). There are four arithmetic operations that can be performed with single precision (32-bit) or double precision (64-bit) floating point numbers: add, subtract, multiply and divide. These operations require two operands. The 8232 assumes that these operands are located in the internal stack as Top of Stack (TOS) and Next on Stack (NOS). The result will always be returned to the previous NOS which becomes the new TOS. Results from an operation are of the same precision and format as the operands. The results will be rounded to preserve the accuracy. In addition to the arithmetic operations, the 8232 implements eight data manipulating operations. These include changing the sign of a double or single precision operand located in TOS, exchanging single precision operands located at TOS and NOS, as well as copying and popping single or double precision operands. See also the sections on status register and operand formats.

The execution times of the commands are all data dependent. Table 2 shows one example of each command execution time.

Interrupt Requests

There are two interrupt lines from the FPP that may generate an interrupt request to the host: END (MINTR1) and ERINT (MINTR0). The END interrupt line is active upon command completion and the ERINT line is active when the current command execution results in an error condition. The error conditions are: attempt to divide by zero, exponent overflow and exponent underflow. Both the END and ERINT signals are cleared by a reset or status register read.

Installation

The iSBX 332 module plugs directly into the female iSBX connector on the host board. The module is then secured at one additional point with nylon hardware to insure the mechanical security of the assembly (see Figures 1 and 2).

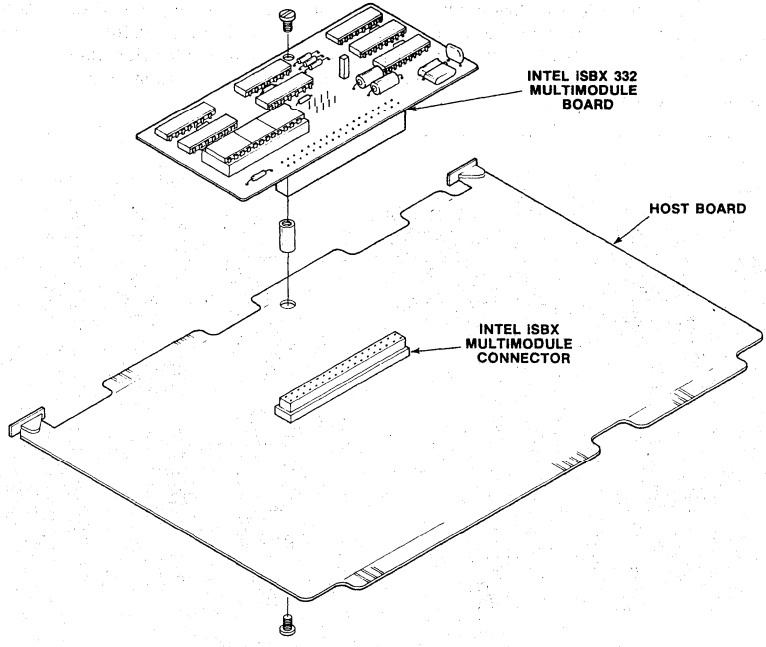


Figure 1. Installation of iSBX 332 Module on a Host Board

Table 1. Command Summary

Command Bits 7 6 5 4 3 2 1 0	Mnemonic	Description
X 0 0 0 0 0 0 1	SADD	Add TOS to NOS single precision and result to NOS. Pop stack.
X 0 0 0 0 0 1 0	SSUB	Subtract TOS from NOS single precision and result to NOS. Pop stack.
X 0 0 0 0 0 1 1	SMUL	Multiply NOS by TOS single precision and result to NOS. Pop stack.
X 0 0 0 0 1 0 0	SDIV	Divide NOS by TOS single precision and result to NOS. Pop stack.
X 0 0 0 0 1 0 1	CHSS	Change sign of TOS single precision operand.
X 0 0 0 0 1 1 0	PTOS	Push single precision operand on TOS to NOS.
X 0 0 0 0 1 1 1	POPS	Pop single precision operand from TOS. NOS becomes TOS.
X 0 0 0 1 0 0 0	XCHS	Exchange TOS with NOS single precision.
X 0 1 0 1 1 0 1	CHSD	Change sign of TOS double precision operand.
X 0 1 0 1 1 1 0	PTOD	Push double precision operand on TOS to NOS.
X 0 1 0 1 1 1 1	POPD	Pop double precision operand from TOS. NOS becomes TOS.
X 0 0 0 0 0 0 0	CLR	CLR status.
X 0 1 0 1 0 0 1	DADD	Add TOS to NOS double precision and result to NOS. Pop stack.
X 0 1 0 1 0 1 0	DSUB	Subtract TOS from NOS double precision and result to NOS. Pop stack.
X 0 1 0 1 0 1 1	DMUL	Multiply NOS by TOS double precision and result to NOS. Pop stack.
X 0 1 0 1 1 0 0	DDIV	Divide NOS by TOS double precision and result to NOS. Pop stack.

NOTE:
X = Don't care. Operation for bit combinations not listed above is undefined.

Table 2. Execution Times

Command	TOS	NOS	Result	Clock Periods	Time (μs)
SADD	3F800000	3F800000	40000000	58	14.5
SSUB	3F800000	3F800000	00000000	56	14.0
SMUL	40400000	3FC00000	40900000	198	49.5
SDIV	3F800000	40000000	3F000000	228	57.0
CHSS	3F800000	—	BF800000	10	2.5
PTOS	3F800000	—	—	16	4.0
POPS	3F800000	—	—	14	3.5
XCHS	3F800000	40000000	—	26	6.5
CHSD	3FF0000000000000	—	BFF0000000000000	24	6.0
PTOD	3FF0000000000000	—	—	40	10.0
POPD	3FF0000000000000	—	—	26	6.5
CLR	3FF0000000000000	—	—	4	1.0
DADD	3FF00000A0000000	8000000000000000	3FF00000A0000000	578	144.5
DSUB	3FF00000A0000000	8000000000000000	3FF00000A0000000	578	144.5
DMUL	BFF8000000000000	3FF8000000000000	C002000000000000	1748	437.0
DDIV	BFF8000000000000	3FF8000000000000	BFF0000000000000	4560	1140.0

NOTE:
TOS, NOS and result are in hexadecimal; clock period is in decimal.

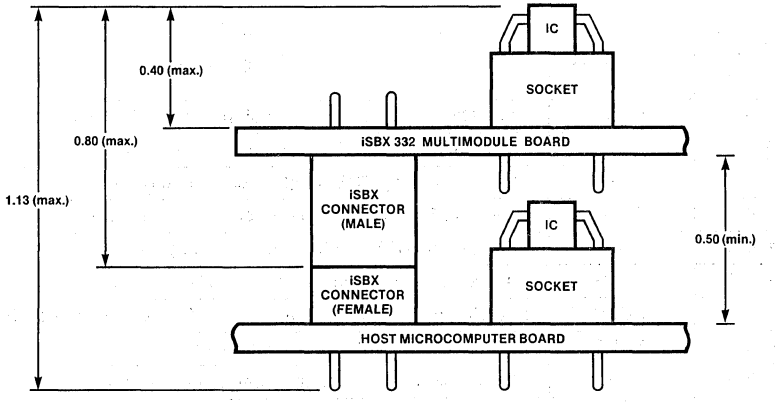


Figure 2. Mounting Clearances (inches)

SPECIFICATIONS

Word Size

Data — 8 Bits

I/O Addressing

Function	Type of Operation	iSBX Connector Port Address
Data Transfer	Read or Write	X0, X2, X4, or X6
Command Transfer	Write	X1, X3, X5, or X7
Status Transfer	Read	X1, X3, X5, or X7
Reset	Write	X8 through XF

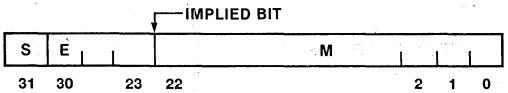
NOTE:
The port addresses are determined on the host iSBC microcomputer. Refer to the Hardware Reference Manual for your host iSBC microcomputer to determine the first digit (X) of the connector port address.

Arithmetic Functions

See Table 1

Floating Point Format

Single Precision Floating Point (32 Bits)

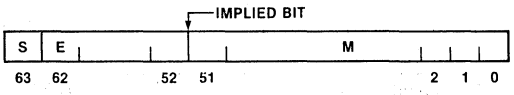


Bit 31: S = Sign of the mantissa. 1 represents negative and 0 represents positive.

Bits 23-30: E = These 8 bits represent a biased exponent. The bias is $2^7 - 1 = 127$.

Bits 0-22: M = 23-bit mantissa. Together with the sign bit, the mantissa represents a signed fraction in sign-magnitude notation. There is an implied 1 beyond the most significant bit (bit 22) of the mantissa. In other words, the mantissa is assumed to be a 24-bit normalized quantity and the most significant bit, which will always be 1 due to normalization, is implied. The FPP restores this implied bit internally before performing arithmetic, normalizes the result, and strips the implied bit before returning the results to the external data bus. The binary point is between the implied bit and bit 22 of the mantissa.

Double Precision Floating Point (64 Bits)



Bit 63: S = Sign of the mantissa. 1 represents negative and 0 represents positive.

Bits 52-62: E = Biased exponent. The bias is $2^{10} - 1 = 1023$.

Bits 0-51: M = 51-bit mantissa. Together with the sign bit, the mantissa represents a signed fraction in sign-magnitude notation. There is an implied 1 beyond the most significant bit (bit 51) of the mantissa. In other words, the mantissa is assumed to be a 53-bit normalized quantity and the most significant bit, which will always be a 1 due to normalization, is implied. The FPP restores this implied bit internally before performing arithmetic, normalizes the result, and strips the implied bit before returning the result to the external data bus. The binary point is between the implied bit and bit 51 of the mantissa.

Status Byte

Contains the following information:

BUSY	SIGN S	ZERO Z	RESERVED	DIVIDE EXCEPTION D	EXPONENT UNDERFLOW U	EXPONENT OVERFLOW V	RESERVED
7	6	5	4	3	2	1	0

Bit 0 Reserved.

Bit 1 Exponent Overflow (V): When 1, this bit indicates that exponent overflow has occurred. Cleared to zero otherwise.

Bit 2 Exponent Underflow (U): When 1, this bit indicates that exponent underflow has occurred. Cleared to zero otherwise.

Bit 3 Divide Exception (D): When 1, this bit indicates that an attempt to divide by zero has been made. Cleared to zero otherwise.

Bit 4 Reserved.

Bit 5 Zero (Z): When 1, this bit indicates that the result returned to TOS after a command is all zeros. Cleared to zero otherwise.

Bit 6 Sign (S): When 1, this bit indicates that the result returned to TOS is negative. Cleared to zero otherwise.

Bit 7 Busy: When 1, this bit indicates the APU is in the process of executing a command. It will become zero after the command execution is complete. All other status bits should be considered to be undefined if this bit is set.

Access Time

Read — 1900 ns (max.)
Write — 1900 ns (max.)

NOTE:

Actual transfer speed is dependent upon the cycle time of the host microcomputer. The listed times assume no operation in progress. If an operation is executing when an access is attempted, the command execution time must be added to the above times for all accesses except status read.

Interrupts

Two interrupt requests may originate from the FPP indicating command completion (END) and error conditions (ERINT).

Interface

iSBX Bus — All signals TTL compatible

Physical Characteristics

Width — 6.35 cm (2.50 in.)
Length — 9.40 cm (3.70 in.)
Height* — 2.04 cm (0.80 in.) iSBX 332 Board
— 2.86 cm (1.13 in.) iSBX 332 Board + Host Board
Weight — 51 gm (1.79 oz)

*See Figure 2

Electrical Characteristics

DC Power Requirements

$V_{CC} = +5V \pm 5\%$ $I_{CC} = 365$ mA max.
$V_{DD} = +12V \pm 5\%$ $I_{DD} = 75$ mA max.

Environmental

Operating Temperature — 0°C to 55°C

Free moving air across the base board and iSBX board.

Reference Manual

9803204-01 — iSBX 332 Floating Point Math MULTIMODULE Board (NOT SUPPLIED)

Reference Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Ave., Santa Clara, California 95051.

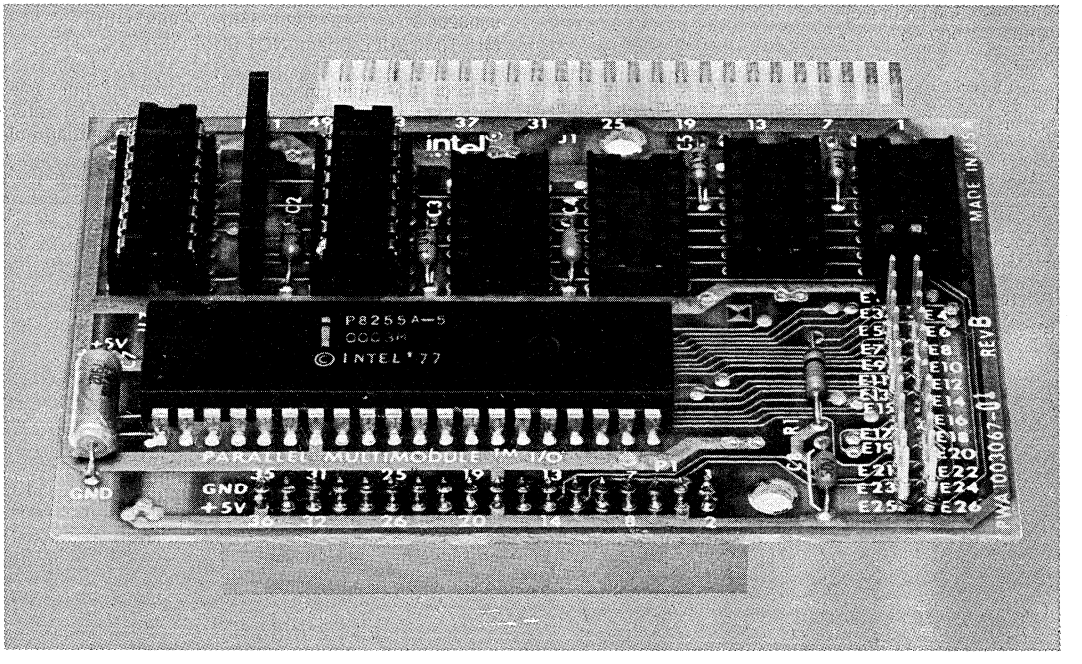
ORDERING INFORMATION

Part Number	Description
SBX 332	Floating Point Math MULTIMODULE Board

iSBX 350
PARALLEL I/O MULTIMODULE BOARD

- **iSBX bus compatible I/O expansion**
- **24 programmable I/O lines with sockets for interchangeable line drivers and terminators**
- **Three jumper selectable interrupt request sources**
- **Accessed as I/O port locations**
- **Single +5V low power requirement**
- **iSBX bus on-board expansion eliminates MULTIBUS system bus latency and increases system throughput**

The Intel® iSBX 350 Parallel I/O MULTIMODULE Board is a member of Intel's new line of iSBX bus compatible MULTIMODULE products. The iSBX MULTIMODULE board plugs directly into any iSBX bus compatible host board offering incremental on-board expansion. The iSBX 350 module provides 24 programmable I/O lines with sockets for interchangeable line drivers and terminators. The iSBX board is closely coupled to the host board through the iSBX bus, and as such, offers maximum on-board performance and frees MULTIBUS system traffic for other system resources. In addition, incremental power dissipation is minimal requiring only 1.6 watts (not including optional driver/terminators).



FUNCTIONAL DESCRIPTION

Programmable Interface

The iSBX 350 module uses an Intel® 8255A-5 Programmable Peripheral Interface (PPI) providing 24 parallel I/O lines. The base-board system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports indicated in Table 1. Therefore, the I/O interface may be customized to meet specific peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and driver/termination characteristics for each application. In addition, inverting bidirectional bus drivers (8226) are provided on sockets to allow convenient optional replacement to non-inverting drivers (8216). The 24 programmable I/O lines, signal ground, and +5 volt power (jumper configurable) are brought to a 50-pin edge connector that mates with flat, woven, or round cable.

Interrupt Request Generation

Interrupt requests may originate from three jumper selectable sources. Two interrupt requests can be automatically generated by the PPI when a byte of information is ready to be transferred to the base board CPU (i.e., input buffer is full) or a byte of information has been transferred to a peripheral device (i.e., output buffer is empty). A third interrupt source may originate directly from the user I/O interface (J1 connector).

Installation

The iSBX 350 module plugs directly into the female iSBX connector on the host board. The module is then secured at one additional point with nylon hardware to insure the mechanical security of the assembly (see Figure 1 and Figure 2).

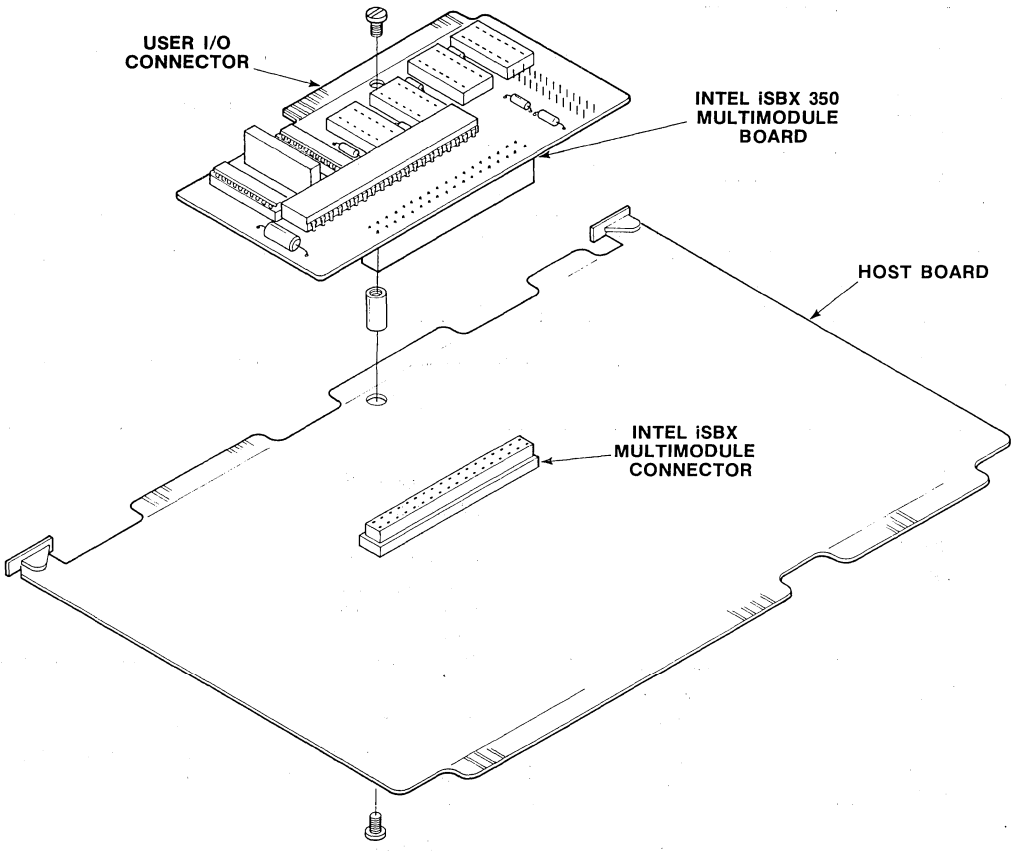


Figure 1. Installation of iSBX 350 Module on a Host Board

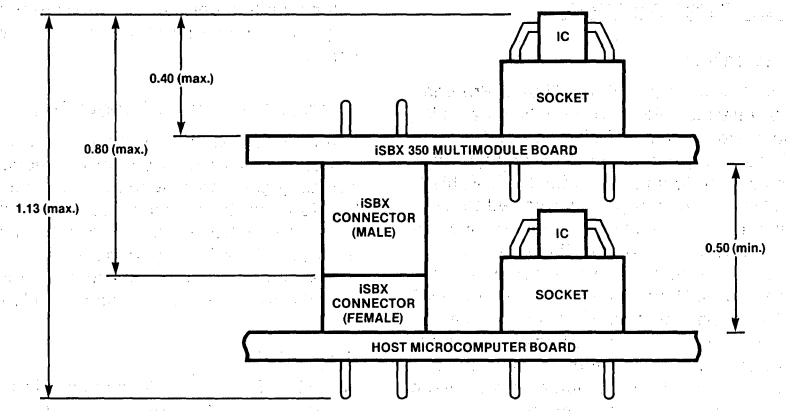


Figure 2. Mounting Clearances (inches)

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
A	8	X	X	X	X	X	
B	8	X	X	X	X		
C	4	X		X			X[1]
	4	X		X			X[1]

NOTE:

1. Part of port C must be used as a control port when either port A or port B are used as a latched and strobed input or a latched and strobed output port or port A is used as a bidirectional port.

SPECIFICATIONS

Word Size

Data — 8 Bits

I/O Addressing

8255A-5 Ports	iSBX 350 Address
Port A	X0 or X4
Port B	X1 or X5
Port C	X2 or X6
Control	X3 or X7
Reserved	X8 to XF

NOTE:

The first digit of each port I/O address is listed as "X" since it will change dependent on the type of host iSBC microcomputer used. Refer to the Hardware Reference Manual for your host iSBC microcomputer to determine the first digit of the port address.

I/O Capacity

24 programmable lines (see Table 1)

Access Time

Read — 250 ns max.
Write — 300 ns max.

NOTE:

Actual transfer speed is dependent upon the cycle time of the host microcomputer.

Interrupts

Interrupt requests may originate from the programmable peripheral interface (2) or the user specified I/O (1).

Interfaces

iSBX™ Bus — All signals TTL compatible
Parallel I/O — All signals TTL compatible

Parallel Interface Connectors

Interface	No. of Pairs/ Pins	Centers (In.)	Connector Type	Vendor	Vendor Part No.
Parallel I/O Connector	25/50	0.1	Female	3M	3415-0001 with Ears
Parallel I/O Connector	25/50	0.1	Female, Soldered	GTE Sylvania	6AD01251A1DD

Note: Connector compatible with those listed may also be used.

Line Drivers and Terminators

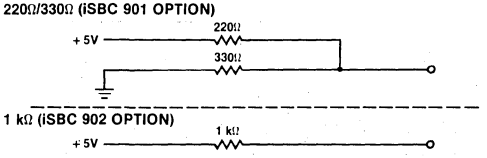
I/O Drivers — The following line drivers and terminators are all compatible with the I/O driver sockets on the iSBX 350.

Driver	Characteristic	Sink Current (mA)
7438	I, OC	48
7437	I	48
7432	NI	16
7426	I, OC	16
7409	NI, OC	16
7408	NI	16
7403	I, OC	16
7400	I	16

Note:
I = Inverting, NI = Non-Inverting, OC = Open Collector

Port 1 has 25 mA totem pole drivers and 1 kΩ terminators.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pull up.



Physical Characteristics

Width — 7.24 cm (2.85 in.)
Length — 9.40 cm (3.70 in.)
Height* — 2.04 cm (0.80 in.) iSBX 350 Board
— 2.86 cm (1.13 in.) iSBX 350 Board + Host Board
Weight — 51 gm (1.79 oz)

*See Figure 2.

Electrical Characteristics

DC Power Requirements

Power Requirement	Configuration
+5V @ 320 mA	Sockets XU3, XU4, XU5, and XU6 empty (as shipped).
+5V @ 500 mA	Sockets XU3, XU4, XU5, and XU6 contain 7438 buffers.
+5V @ 620 mA	Sockets XU3, XU4, XU5, and XU6 contain iSBC 901 termination devices.

Environmental

Operating Temperature — 0°C to 55°C

Reference Manual

9803191-01 — iSBX 350 Parallel I/O MULTIMODULE Manual (NOT SUPPLIED)

Reference Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Ave., Santa Clara, California 95051.

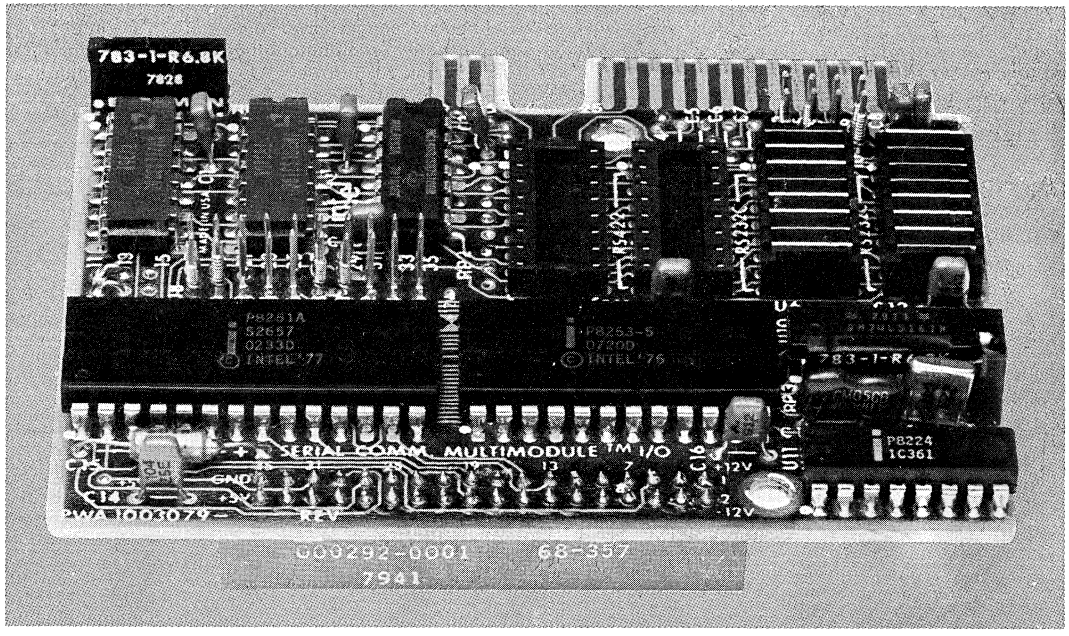
ORDERING INFORMATION

Part Number	Description
SBX 350	Parallel I/O MULTIMODULE Board

iSBX 351
SERIAL I/O MULTIMODULE BOARD

- **iSBX bus compatible I/O expansion**
- **Programmable synchronous/asynchronous communications channel with RS232C or RS449/422 interface**
- **Software programmable baud rate generator**
- **Two programmable 16-bit BCD or binary timers/event counters**
- **Four jumper selectable interrupt request sources**
- **Accessed as I/O port locations**
- **Low power requirements**
- **Single +5V when configured for RS449/422 interface**
- **iSBX bus on-board expansion eliminates MULTIBUS system bus latency and increases system throughput**

The Intel® iSBX 351 Serial I/O MULTIMODULE board is a member of Intel's new line of iSBX bus compatible MULTIMODULE products. The iSBX MULTIMODULE board plugs directly into any iSBX bus compatible host board offering incremental on-board I/O expansion. The iSBX 351 module provides one RS232C or RS449/422 programmable synchronous/asynchronous communications channel with software selectable baud rates. Two general purpose programmable 16-bit BCD or binary timers/event counters are available to the host board to generate accurate time intervals under software control. The iSBX board is closely coupled to the host board through the iSBX bus, and as such, offers maximum on-board performance and frees MULTIBUS system traffic for other system resources. In addition, incremental power dissipation is minimum requiring only 3.0 watts (assumes RS232C interface).



FUNCTIONAL DESCRIPTION

Communications Interface

The iSBX 351 module uses the Intel® 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) providing one programmable communications channel. The USART can be programmed by the system software to individually select the desired asynchronous or synchronous serial data transmission technique (including IBM Bi-Sync). The mode of operation (i.e. synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. The 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The command lines, serial data lines, and signal ground lines are brought out to a double edge connector configurable for either an RS232C or RS449/422 interface (see Figure 3). In addition, the iSBX 351 module is jumper configurable for either point-to-point or multidrop network connection.

16-Bit Interval Timers

The iSBX 351 module uses an Intel 8253 Programmable Interval Timer (PIT) providing 3 fully programmable and independent BCD and binary 16-bit

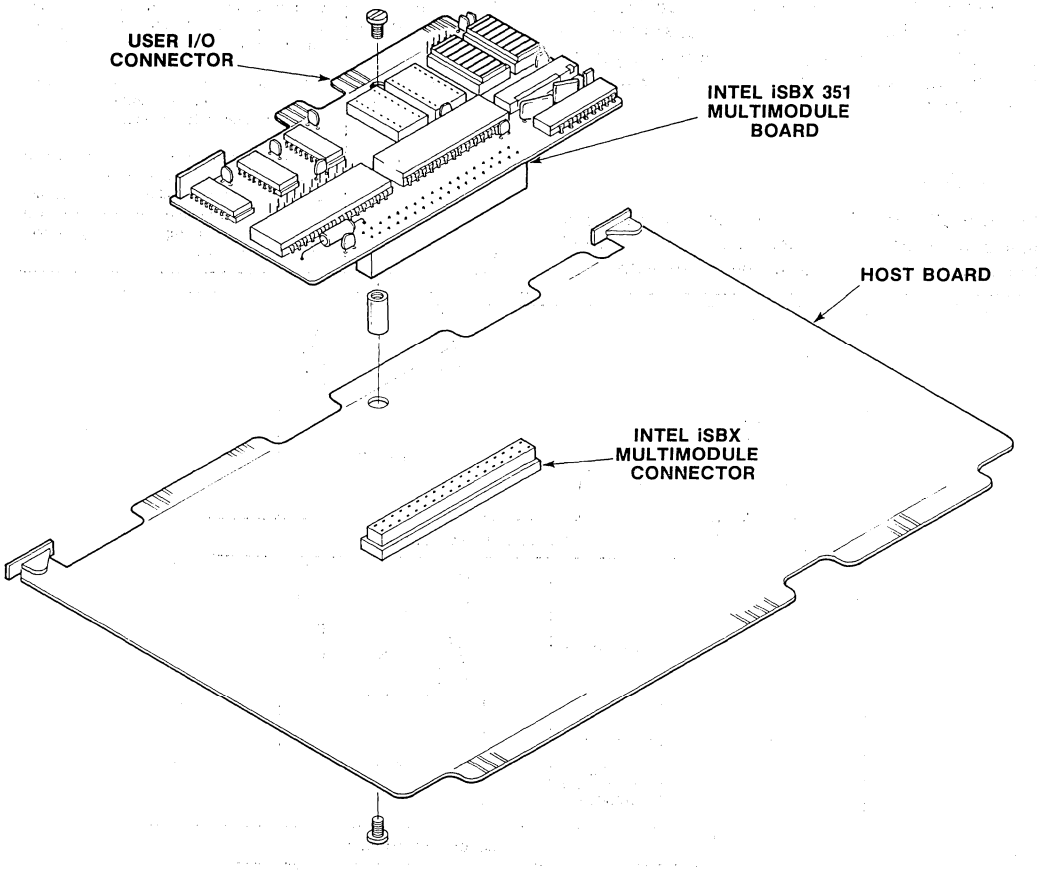


Figure 1. Installation of iSBC 351 Module on a Host Board

AFN-01381A

interval timers. One timer is available to the system designer to generate baud rates for the USART under software control. Routing for the outputs from the other two counters is jumper selectable to the host board. In utilizing the iSBX 351 module, the systems designer simply configures, via software, each timer independently to meet system requirements. Whenever a given baud rate or time delay is needed, software commands the programmable timers to select the desired function. The functions of the timers are shown in Table 1. The contents of each counter may be read at any time during system operation.

Interrupt Request Lines

Interrupt requests may originate from four sources. Two interrupt requests can be automatically generated by the USART when a character is ready to be transferred to the host board (i.e. receive buffer is full) or a character has been transmitted (i.e. transmit buffer is empty). In addition, two jumper selectable requests can be generated by the programmable timers.

Installation

The iSBX 351 module plugs directly into the female iSBX connector on the host board. The module is then secured at one additional point with nylon hardware to insure the mechanical security of the assembly (see Figures 1 and 2).

Table 1. Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated. This function is useful for generation of real-time clocks.
Programmable one-shot	Output goes low upon receipt of an external trigger edge and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counting "window" has been enabled or an interrupt may be generated after N events occur in the system.

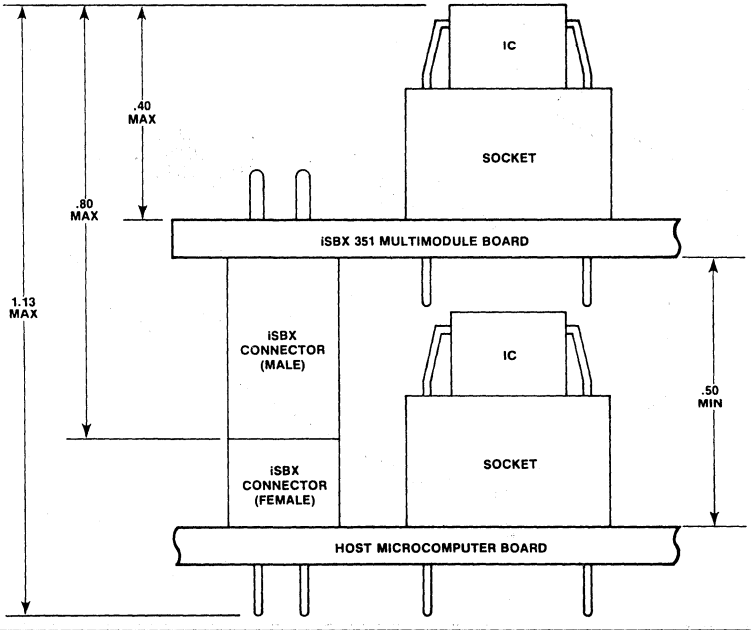


Figure 2. Mounting Clearances (inches)

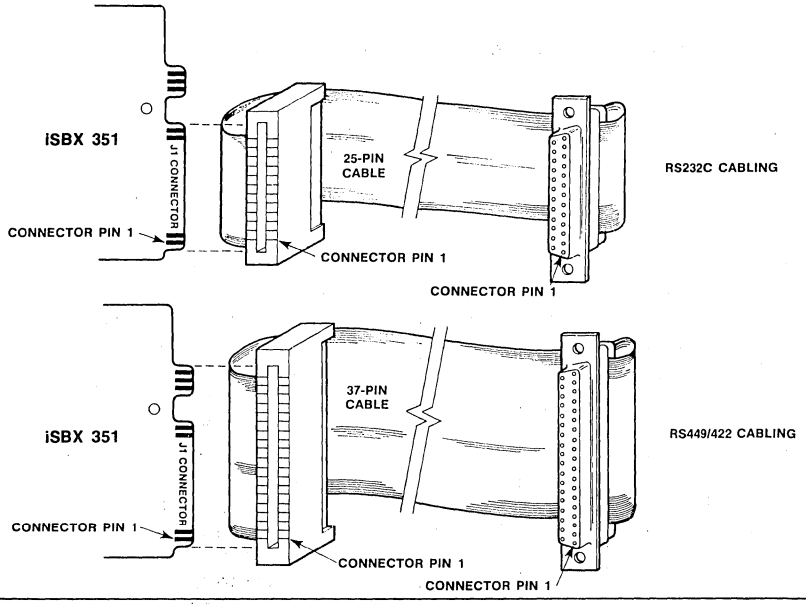


Figure 3. Cable Construction and Installation for RS232C and RS449/422 Interface

SPECIFICATIONS

Word Size

Data — 8 bits

I/O Addressing

I/O Address	Chip Select	Function
X0, X2, X4, or X6	8251A USART	Write: Data Read: Data
X1, X3, X5, or X7		Write: Mode or Command Read: Status
X8 or XC	8253 PIT	Write: Counter 0 (Load Count ÷ N) Read: Counter 0
X9 or XD		Write: Counter 1 (Load Count ÷ N) Read: Counter 1
XA or XE		Write: Counter 2 (Load Count ÷ N) Read: Counter 2
XB or XF		Write: Control Read: None

NOTE: The first digit of each port I/O address is listed as "X" since it will change depending on the type of host iSBC microcomputer used. Refer to the Hardware Reference Manual for your host iSBC microcomputer to determine the first digit of the I/O address.

Access Time

Read — 250 nsec max

Write — 300 nsec max

Note

Actual transfer speed is dependent upon the cycle time of the host microcomputer.

Serial Communications

Synchronous — 5 - 8-bit characters; internal character synchronization; automatic sync insertion; even, odd or no parity generation/detection.

Asynchronous — 5 - 8-bit characters; break character generation and detection; 1, 1½, or 2 stop bits; false start bit detection; even, odd or no parity generation/detection.

Sample Baud Rate:

8253 PIT Frequency[1] (kHz, Software Selectable)	8251 USART Baud Rate (Hz)[2]		
	Synchronous	Asynchronous	
		÷ 16	÷ 64
307.2	—	19200	4800
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
2.4	2400	150	—
1.76	1760	110	—

NOTES: 1. Frequency selected by I/O writes of appropriate 16-bit frequency factor to Baud Rate Register.

2. Baud rates shown here are only a sample subset of possible software-programmable rates available. Any frequency from 18.75 Hz to 614.4 kHz may be generated utilizing on-board crystal oscillator and 16-bit Programmable Interval Timer (used here as frequency divider).

Interval Timer and Baud Rate Generator

Input Frequency (selectable):

1.23 MHz ±0.1% (.813 μsec period nominal)

153.6 kHz ±0.1% (6.5 μsec period nominal)

Output Frequency:

	Rate Generator (Frequency)		Real-Time Interrupt (Interval)	
	Min.	Max.	Min.	Max.
Single Timer[1]	18.75 Hz	614.4 kHz	1.63 μsec	53.3 msec
Single Timer[2]	2.34 Hz	76.8 kHz	13.0 μsec	426.7 msec
Dual Timer[3] (Counters 0 and 1 in series)	0.000286 Hz	307.2 kHz	3.26 μsec	58.25 min
Dual Timer[4] (Counters 0 and 1 in series)	0.0000358 Hz	38.4 kHz	26.0 μsec	7.77 hrs

NOTES: 1. Assuming 1.23 mHz clock input.
2. Assuming 153.6 kHz clock input.
3. Assuming Counter 0 has 1.23 mHz clock input.
4. Assuming Counter 0 has 153.6 kHz clock input.

Interrupts

Interrupt requests may originate from the USART (2) or the programmable timer (2).

Interfaces

iSBX Bus — all signals TTL compatible.

Serial — configurable for EIA Standards RS232C or RS449/422

EIA Standard RS232C signals provided and supported:
- Clear to Send (CTS)
- Data Set Ready (DSR)
- Data Terminal Ready (DTR)
- Request to Send (RTS)
- Receive Clock (RXC)
- Receive Data (RXD)
- Transmit Clock (DTE TXC)
- Transmit Data (TXD)

EIA Standard RS449/422 signals provided and supported:
- Clear to Send (CS)
- Data Mode (DM)
- Terminal Ready (TR)
- Request to Send (RS)
- Receive Timing (RT)
- Receive Data (RD)
- Terminal Timing (TT)
- Send Data (SD)

Physical Characteristics

Width — 7.24 cm (2.85 inches)
Length — 9.40 cm (3.70 inches)
Height* — 2.04 cm (0.80 inches) iSBX 351 Board
— 2.86 cm (1.13 inches) iSBX 351 Board and Host Board
Weight — 51 grams (1.79 ounces)
* (See Figure 2)

Serial Interface Connectors

Configuration	Mode[2]	MULTIMODULE Edge Connector	Cable	Connector[8]
RS232C	DTE	26-pin[5], 3M-3462-0001	3M[3]-3349/25	25-pin[7], 3M-3482-1000
RS232C	DCE	26-pin[5], 3M-3462-0001	3M[3]-3349/25	25-pin[7], 3M-3483-1000
RS449	DTE	40-pin[6], 3M-3464-0001	3M[4]-3349/37	37-pin[1], 3M-3502-1000
RS449	DCE	40-pin[6], 3M-3464-0001	3M[4]-3349/37	37-pin[1], 3M-3503-1000

NOTES: 1. Cable housing 3M-3485-4000 may be used with the connector.
2. DTE — Data Terminal mode (male connector), DCE — Data Set mode (female connector).
3. Cable is tapered at one end to fit the 3M-3462 connector.
4. Cable is tapered to fit 3M-3464 connector.
5. Pin 26 of the edge connector is not connected to the flat cable.
6. Pins 37, 39, and 40 of the edge connector are not connected to the flat cable.
7. May be used with cable housing 3M-3485-1000.
8. Connectors compatible with those listed may also be used.

Electrical Characteristics

DC Power Requirements

Mode	Voltage	Amps (Max.)
RS232C	+5V ±0.25V	460 mA
	+12V ±0.6V	30 mA
	−12V ±0.6V	30 mA
RS449/422	+5V ±0.25V	530 mA

Environmental Characteristics

Temperature — 0 - 55°C, free moving air across the base board and MULTIMODULE board.

Reference Manual

9803190-01 — iSBX 351 Serial I/O MULTIMODULE Manual (NOT SUPPLIED)

Reference Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Ave., Santa Clara, California, 95051.

ORDERING INFORMATION

Part Number	Description
SBX 351	Serial I/O MULTIMODULE Board

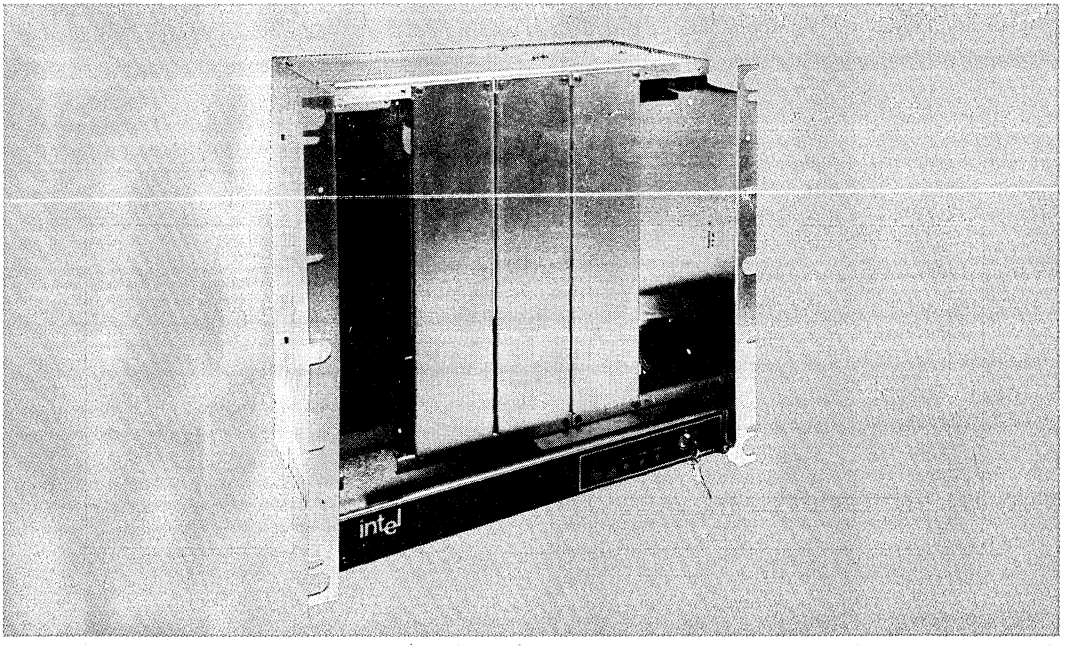
iCS™ Industrial Control Series

4

iCS 80
INDUSTRIAL CHASSIS

- **MULTIBUS standard 4-slot backplane, expandable to 12 slots**
- **Vertical board orientation for convection cooling**
- **19-inch wide RETMA rack mounting or NEMA type backwall mounting brackets**
- **Four fans for forced-air cooling**
- **Submitted for approval as a UL recognized component**
- **110/230V, 50/60 Hz operation**
- **Slide in/out mounting rails for iSBC power supplies**
  - **— Quick disconnect cabling for serviceability**
  - **— Your choice of supply**
- **Lockable service panel**
- **All front access serviceability**
  - **— iSBC boards**
  - **— Power supplies**
  - **— Interrupt and reset buttons**
  - **— Operation indicators and fuse**
- **Recessed mounting space for signal conditioning/wire termination panels**

The iCS 80 Industrial Chassis provides industrially oriented mounting space for Intel single board computer (iSBC) products, associated iSBC power supplies, and related iCS 9XX analog and digital signal conditioning/termination panels. The base unit provides a 4-slot MULTIBUS backplane (iSBC 604) with expansion space and cabling to expand to 12 MULTIBUS backplane slots by adding additional 4-slot iSBC 614s as needed (up to two). All of the 25-plus Intel MULTIBUS bus-compatible iSBC boards can be inserted into any one of the 12 slots. In addition, over 50 products from 30 independent manufacturers have been designed for mounting into the MULTIBUS backplane. Full MULTIBUS compatibility in the iCS 80 chassis also allows configuration of multiple single board computers to share system tasks through communication over the bus (through multimaster bus arbitration built on the multiple iSBC processors).



FUNCTIONAL DESCRIPTION

Self Contained Low Cost Controllers

Small, self contained industrial controllers can be configured with the 4-slot cardcage and iSBC 635 power supply. As shown in Figure 2, this packaging can also accommodate the iCS 9XX series signal conditioning/ termination panels.

Or Large Power and Point Counts in a Small Package

At the high end of performance for the iCS 80 chassis, a user can build a 12-slot configuration with the Intel iSBC 640 Power Supply. This iCS 80 chassis can support the iSBC 86/12A 16-bit computer with 112K bytes memory (96K RAM, 16K ROM), 64 differential analog inputs, 180 digital inputs, 52 isolated digital outputs, and 8 analog outputs (four current loops); in total a 304-channel, mixed analog and isolated digital, input and output controller, large enough for most dedicated applications (see Figure 3 for two other examples).

Engineered for Industrial Applications

The MULTIBUS slots are mounted vertically to improve convection cooling and the top, bottom and sides are engineered to allow maximum air flow over the boards. Four fans are provided as standard to increase air flow, allowing users to eliminate or minimize the need for supplementary fans or air conditioning.

Power Supply Flexibility

To provide a modular base on which to build a variety of configurations, no power supply is provided in the iCS 80 Industrial Chassis. Users choose one of the low cost Intel iSBC 635 (14-amp) or iSBC 640 (30-amp) power supplies based on their application. Slide in/out mounting rails are provided to match the iSBC 635 and iSBC 640 supplies, and quick disconnect cabling and connectors are provided for rapid service replacement. An AC wiring barrier strip allows simple wiring connections for integration into larger systems (see Figure 4).

Industrial Rack Mounting

The chassis mounts directly into 19-inch standard width RETMA (Radio-Electronics-Television Manufacturers Association) customer provided rack. Alternately, mounting brackets and power cabling access are provided for mounting directly on a backwall, such as the backwall panel of a NEMA-type (National Electrical Manufacturers Association), front-access-only cabinet.

Front Access Serviceability

To simplify serviceability, front access is provided for all iSBC boards, the power supply, operation indicator lights, interrupt and reset buttons, and the AC power fuse.

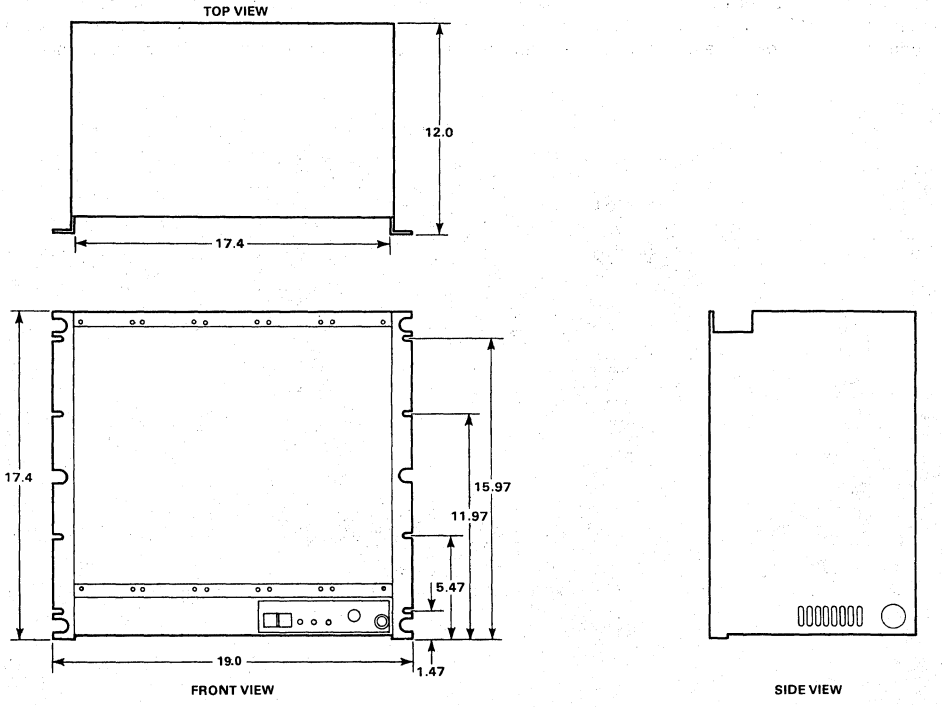
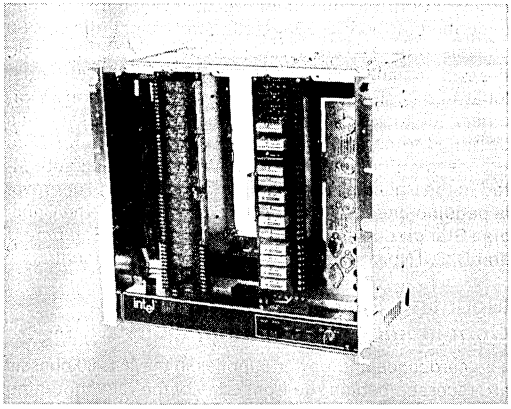


Figure 1. iCS 80 Chassis Dimensions



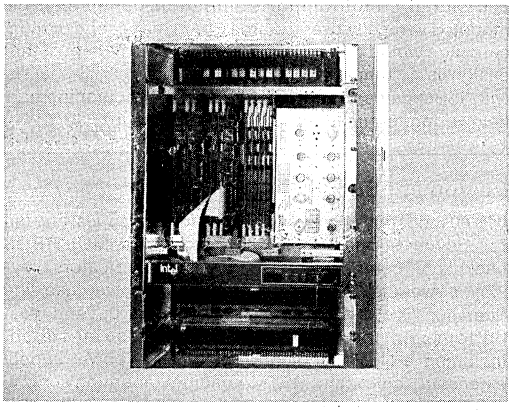
Typical Small Configuration

- 8-bit 8088 processor (iSBC 88/40)
- 4K bytes RAM (8K optional)
- 2K-16K bytes E^2PROM or 8K-128K bytes ROM/EPROM
- 16-64 analog inputs
- 8 analog outputs
- 8 isolated digital inputs
- 8 isolated digital outputs

OR

- 12 TTL outputs
- 12 TTL inputs

Figure 2. Small Configuration iCS 80 with iSBC 635, iCS 910 and iCS 930 Signal Conditioning/Termination Panels



Typical Maximum Configuration

- 16-bit 8086 processor (iSBC 86/30 w/RAM MULTIMODULE)
- 768K bytes RAM (2 - iSBC 056)
- 128K bytes EPROM (or 16K E^2PROM)
- 240 analog inputs (3 - iSBC 88/40 w/2 ea. iSBX 311)
- 24 analog voltage outputs

OR

- 24 analog current outputs (4-20 mA)
- 72 isolated digital inputs/outputs
- 144 TTL digital inputs/outputs (2 - iSBC 519s)

(All iCS 9XX Signal Conditioning/Termination Panels are not shown)

Figure 3. iCS 80 with 12 MULTIBUS Card Slots and iSBC 640 Power Supply, Large Configuration

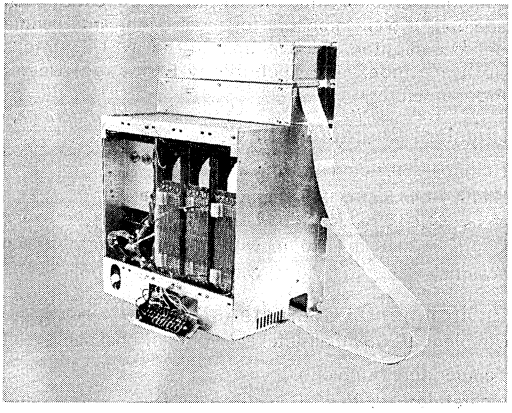


Figure 4. Rear View iCS 80 Chassis Showing Power Distribution Panel, and Cabling from iCS 80 Chassis to iCS 9XX RETMA Mounted Signal Conditioning Panels (Top of iCS 80 Chassis)

Lockable Service Panel

To assist in development, checkout and service, two pushbuttons are provided. The RESET button pulls low the initialize line (INIT) on the MULTIBUS backplane. The INTERRUPT button pulls low one interrupt line on the MULTIBUS backplane (INT1). Logic within the iCS 80 ensures that these buttons function with all versions of Intel single board computers. From the front of the iCS 80 chassis, without a CRT or other panel, an operator or service person can reset or interrupt on-going iCS 80 system operations to get attention, signal an alarm, or start a self-test operation.

A front panel key provides three positions: OFF (AC power off and key removable), ON (AC power on, pushbuttons enabled, key unremovable), and LOCK (AC power on, pushbuttons disabled, key removable).

Three indicator light emitting diodes record basic chassis status. POWER ON (GREEN); RUN (GREEN); and HALT (RED); the RESET or INTERRUPT buttons will remove the HALT state.

U.L. Approved

The iCS 80 chassis has received full Underwriters Laboratory approval (F.6 #E70842) as a U.L. listed component under the Underwriters Laboratories Safety Standard for Process Control Equipment, UL1092. When installed as described in the iCS 80 Installation Manual, the iCS 80 chassis provides adequate protection against shock, fire and casualty hazards, and should comply with most local and regional requirements for installation in ordinary locations. In addition, the iCS 80 chassis was designed to comply with the UL requirements for Data Processing Equipment, UL478. It has also been submitted to the Canadian Standards Association and approval is pending under CSA category C22.2 No. 142, the Canadian Standard for Safety for Process Control Equipment and C22.2 No. 154 for Data Processing Equipment.

Mounting Space for Signal Conditioning/Wire Terminations

The cardcages and power supplies in the iCS 80 chassis are recessed behind the front edge of the rack mounting ears to provide mounting space for the iCS 9XX series signal conditioning/termination panels and field wiring. For smaller systems with only one or two iSBC 604/614 cardcages (4 to 8 slots), up to two iCS 910, iCS 920, or iCS 930 signal conditioning/termination panels can be mounted vertically over the area where the second or third cardcage would mount (see Figure 2). The benefit of this design is a completely self-contained industrial chassis with iSBC cards, power supply, signal conditioning and field wiring terminations, all in one enclosure.

SPECIFICATIONS

Capacity

Four slots for MULTIBUS compatible single board computers, memory, I/O or other expansion boards

Expandable to 12 slots using customer plug-together iSBC 614 cardcages

Front Panel Controls

Pushbuttons

RESET: Connected to Initialize/ on MULTIBUS backplane

INTERRUPT: Connected to Interrupt 1/ line on MULTIBUS backplane.

Panel Indicator Lights (LEDs)

POWER ON (green): +5V power exists on the MULTIBUS backplane

RUN (green): CPU is executing an instruction. Light goes out if CPU is in WAIT or HALT state

HALT (red): CPU has executed a HALT instruction

Keylock

OFF: AC power off, key removable

ON: AC power on, pushbuttons enabled, key unremovable

LOCK: AC power on, pushbuttons disabled, key removable

Fuse — AC power (6A)

Equipment Supplied

iCS 80 industrial chassis, three fans for cardcages, one fan for power supply, 4-slot cardcage with MULTIBUS backplane, control panel with switches, indicators, keylock, power distribution barrier strip, AC power fuse, line filter, 115V power cable, and logic for interrupt and reset buttons. An installation package is also provided, including a NEMA cabinet mounting kit, power supply extension cables, and RETMA cabinet mounting screws, 110/230 VAC operation.

Software

See the RMX/80 Real-time Multitasking Executive specifications for industrial related applications. In addition, system monitors for most of the Intel single board computers are available in the INSITE (Intel's Software Index and Technology Exchange) User's Program Library.

Physical Characteristics

Height — 39.3 cm (15.7 in.)

Width — 48.5 cm (19.0 in.) at front panel
43.5 cm (17.4 in.) behind front panel

Depth — 30.0 cm (12.0 In.) with all protrusions

Weight — 16.8 kg (37.0 lb) without power supplies

Environmental Characteristics

(Ambient at iCS-80 air intake, bottom of chassis)

Temperature (Ambient)

Operating: 0°C to 50°C (32°F to 122°F)

Non-operating: −40°C to +85°C

Humidity — Up to 90% relative, noncondensing at 40°C

Electrical Characteristics

The iCS 80 chassis provides mounting space for either the iSBC 635 or iSBC 640 power supply. Unless otherwise stated, electrical specifications apply to both power supplies when installed by user in iCS 80 chassis.

Input Power

Frequency: 47 to 63 Hz. Voltage (Nominal)
(Single Phase): 100, 115, 215, or 230 VAC +10%, jumper selectable.

	With iSBC 635	With iSBC 640	Input Voltage
Current: (Including fans)	3.0A max	5.6A max	103 VAC
	1.5A max	2.8A max	206 VAC
Power, max:	315 watts	580 watts	

Output Power

Voltage	Output Current (max)		Overvoltage Protection	
	iSBC 635	iSBC 640	iSBC 635	iSBC 640
+ 12V	2.0A	4.5A	+ 14V to + 16V	+ 14V to + 16V
+ 5V	14.0A	30.0A	+ 5.8V to + 6.6V	+ 5.8V to + 6.6V
− 5V	0.9A	1.75A	− 5.8V to − 6.6V	− 5.8V to − 6.6V
− 12V	0.8A	1.75A	− 14V to − 16V	− 14V to − 16V

Combined Line/Load Regulation — ±1% at ±10% static line change and ±50% static load change, measured at the output connector (±0.2% measured at the power supply under the same conditions).

Remote Sensing — Provided for +5 VDC output line regulation.

Output Ripple and Noise — 10 mV (iSBC 635 and iSBC 640 supply) peak-to-peak, max (DC to 500 kHz)

Output Transient Response — Less than 50 μsec for ±50% load change.

Maximum Watts Dissipation (load plus losses) — 500W (iSBC 640 supply), 250W (iSBC 635 supply)

Installation

Complete instructions for installation are contained in the iCS 80 Site Planning and Installation Guide, including RETMA and NEMA cabinet mounting, and field signal, ground wiring and cooling suggestions.

Warranty

The iCS 80 Industrial Chassis is warranted to be free from defects in materials and workmanship under normal use and service for a period of 90 days from date of shipment.

Reference Manuals

9800799A — iCS 80 Industrial Chassis Hardware Reference Manual (SUPPLIES)
9800798 — iCS 80 Industrial Systems Site Planning and Installation Guide (SUPPLIED)
9800708A — iSBC 604/614 Cardcage Hardware Reference Manual (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

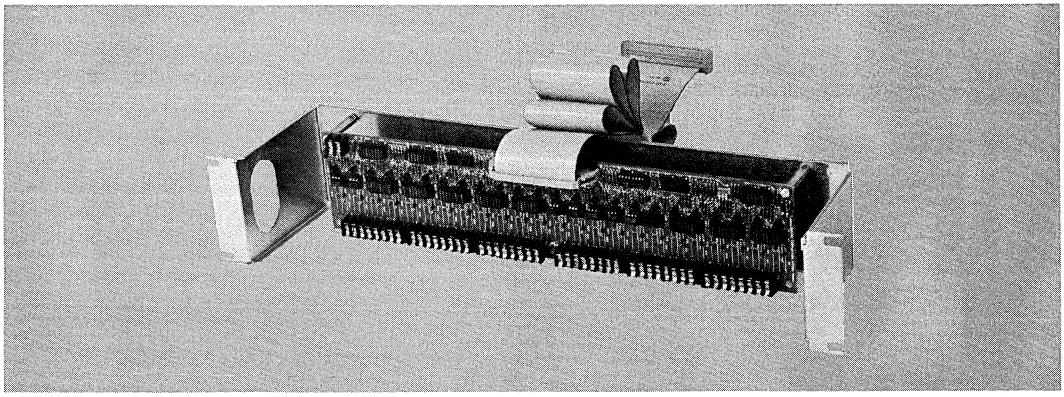
The iCS 80 Industrial Chassis must be ordered as a kit with an Intel power supply of your choice. Ordering as a kit will ensure shipment of the power supply and iCS 80 chassis at the same time. Typical configurations and ordering instructions are:

Part Number	Description
ICS 80 Kit 635	iCS 80 system consisting of: iCS 80 Industrial Chassis iSBC 635 Power Supply
ICS 80, Kit 640	iCS 80 system consisting of: iCS 80 Industrial Chassis iSBC 640 Power Supply

iCS 910/920/930
SIGNAL CONDITIONING/TERMINATION PANELS

- **Interconnects iSBC and digital I/O ports to field signal/control wiring**
- **Ribbon cable connection from panel is pin compatible with iSBC analog, CPU, and digital board I/O ports**
- **Barrier strip screw terminals for**
  - **— iCS 910: 32 single-ended analog inputs (or 16 differential signal plus shield) plus four analog voltage outputs or two analog 4 to 20 mA current outputs**
  - **— iCS 920: 24 medium power digital inputs and/or outputs (55V, 300 mA max)**
  - **— iCS 930: 16 high power AC or DC digital inputs or outputs (280 VAC, 3A max)**
- **Flexible mounting kits for**
  - **— 19″ width RETMA rack**
  - **— NEMA type backwall**
  - **— iCS 80 Industrial Chassis**
- **Digital signal conditioning (iCS 920/930)**
  - **— Sockets for optically isolated input filters and solid state output switches**
  - **— Pad space for transient suppressors, current limiting resistors, and voltage dividers**
  - **— Socketed fuse for overload protection (iCS 930)**
  - **— LED/channel status indicators**
- **Engineering printed circuit mounting space for customer analog input components (iCS 910)**
  - **— Noise fiters**
  - **— Current loop resistors**
  - **— Open circuit detection resistors**
  - **— Voltage divider resistors**
  - **— Thermistor bias current**
- **Submitted for UL recognition**

The iCS 910/920/930 Signal Conditioning/Termination Panels are heavy duty printed circuit boards with screw terminations which allow industrial customers to easily connect their heavier gauge field signal wiring to Intel's line of 8- and 16-bit single board computers, and iSBC analog and digital I/O boards. Flat ribbon cables connect the iCS 910 panels to any of the Intel iSBC 700 series analog input and output board pin-outs. Flat ribbon cables also connect the iCS 920 panels and iCS 930 panel to the 50-pin digital I/O ports (8255 or UPI) on Intel's single board computers and digital I/O boards. Power for opto-isolators or line drivers (+5 VDC) can be supplied via this cable from the iSBC boards. Jumpers and a screw terminal block are provided on the iCS 920/930 panels to allow an external supply of +5V power. A similar jumper/terminal block is provided on the iCS 910 panel to allow users to connect external +15V (or greater) compliance voltage for larger analog output loads.



FUNCTIONAL DESCRIPTION COMMON TO iCS 910/920/930

Large Wire or Spade Lug Connections

The barrier strip screw terminations on the iCS 910/920/930 panels provide familiar connection points for factory electricians to terminate the heavier gauge wiring often pulled through conduits from sensors or control elements. These screw terminals securely connect up to 14 AWG gauge wire size (16-gauge on iCS 910/920 panels). Alternately, spade lugs can be crimped on field wiring and inserted under the screw terminals.

Mounting Flexibility and Serviceability

The iCS 910/920/930 panels were designed to be physically separate from iSBC boards or the iCS 80 chassis to allow maximum mounting flexibility and ease of serviceability. The panels and field wiring can be mounted in one area of the cabinet where electricians have access. Flat ribbon cable can then be run to the area where control electronics technicians have access.

The iCS 910/920/930 panels may be mounted horizontally in a 19" standard width (RETMA) rack using a recessed mounting panel (see Figure 1). Alternately, the panels can be mounted on a cabinet wall (e.g., NEMA cabinet backwall) using standoffs provided (see Figure 2). Or, for the most compact packaging, users can mount up to two iCS 910/920/930 panels vertically, directly on the front of the iCS 80 chassis using standoffs and holes provided (see Figure 3).

A black metal labelling strip is provided with each iCS 910/920/930 panel. White, blank gummed labels are included so that users can custom identify each input or output channel. A clear plastic cover is provided to protect against inadvertent touching or damage to the screw terminals or customer mounted components.

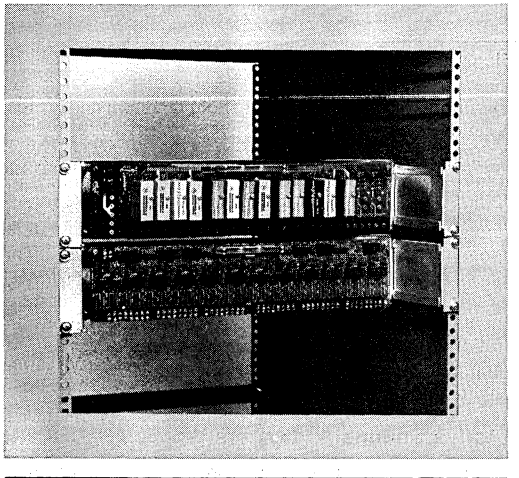


Figure 1. iCS 930 AC and iCS 920 Digital Signal Conditioning/Termination Panel Mounted on a 19" Width RETMA Rack

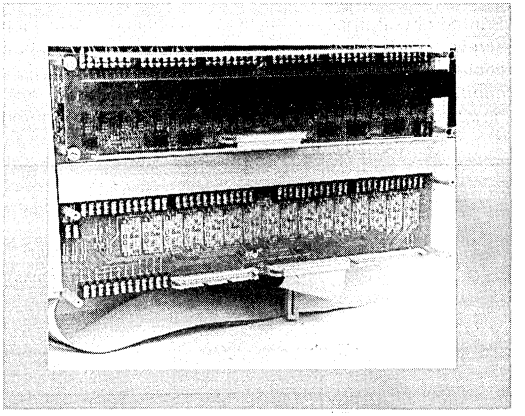


Figure 2. iCS 910/920 Signal Conditioning/Termination Panel Mounted on a NEMA Cabinet Backwall

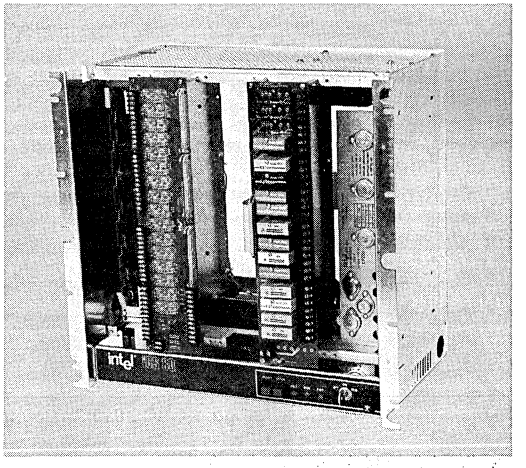


Figure 3. iCS 910/930 Signal Conditioning/Termination Panels Mounted on iCS 80 Industrial Chassis

iCS 910 ANALOG SIGNAL CONDITIONING/TERMINATION PANEL

Mixed Analog Input and Output Signals

A single iCS 910 panel connects up to 32 single ended analog inputs (or 16 differential analog inputs plus shield) to the iSBC 711 or iSBC 732 analog input boards. In addition, the same iCS 910 panels can connect up to four analog output voltages from the iSBC 724 analog output board, or two voltage or 4 to 20 mA current loop outputs from the iSBC 732 combination analog input/output board. Three flat ribbon cables are included in the iCS 910 installation kit (two analog inputs, one analog output) to route signals to iSBC 711/724/732 boards.

Engineered Signal Conditioning Mounting Space

Printed circuit traces on the iCS 910 panel connect each screw terminal analog input channel to the flat ribbon cable connector. Users can jump straight through signal connections if they desire. Each input channel trace, however, passes through a custom engineered printed circuit area onto which users may mount components to signal condition analog input signals. Pad traces and holes are designed to allow easy mounting of R-C noise filters, input voltage resistor/divider networks, current loop input resistors, open circuit detection resistors, or to supply thermistor bias current (see Figure 4 for schematic of a typical analog input channel).

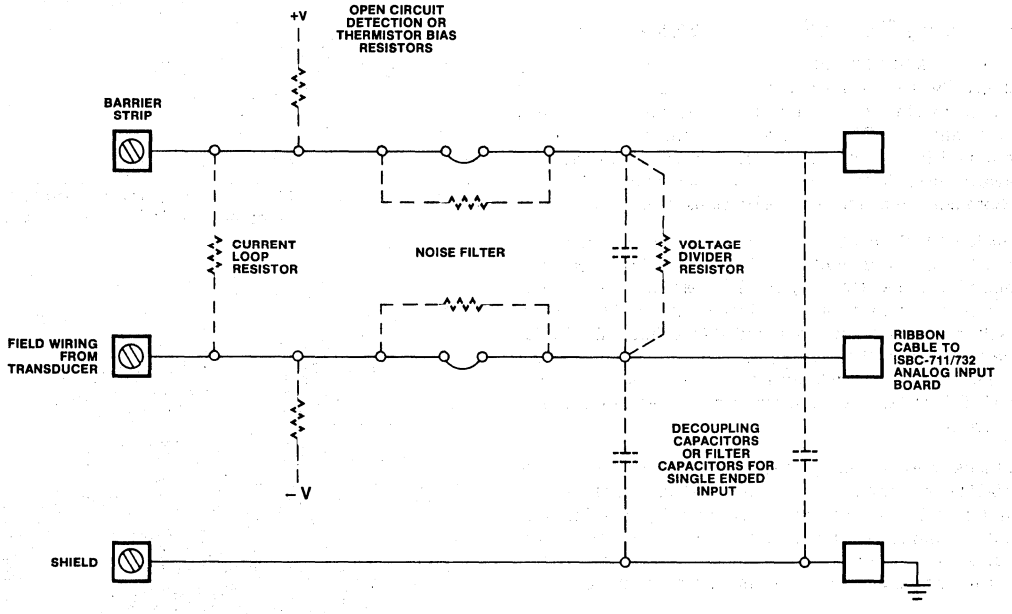


Figure 4. iCS 910 Analog Input Signal Conditining Examples

iCS 920 DIGITAL SIGNAL CONDITIONING/TERMINATION PANEL

The iCS 920 panel interconnects up to 24, 2-wire digital input or output channels from barrier strip screw terminals to the 16- or 24-bit digital I/O ports, standard on many Intel single board computers and digital I/O expansion boards. Screw terminals allow for one each 16 AWG size wire for differential (2-wire) connections or two each AWG 18-gauge wire for daisy chaining grounds or power for external contact sensing.

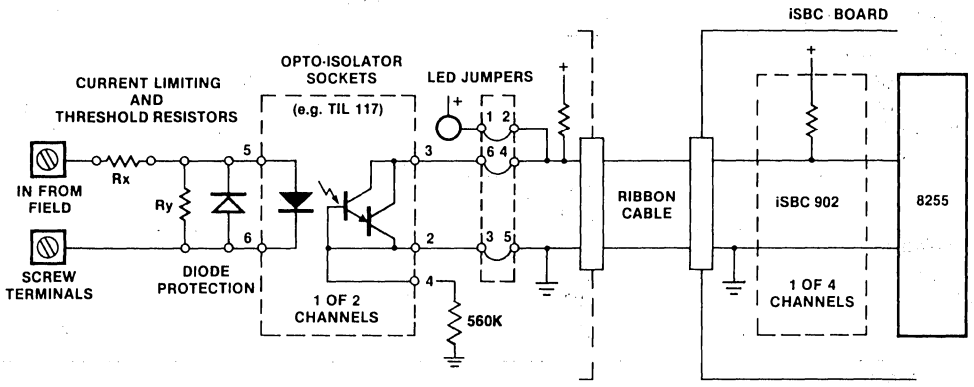
Flexibility in Isolation and Serviceability

Dual-in-line sockets are in series with each channel (see Figure 5) to allow customer jumpering for straight through connections (TTL I/O), or for insertion of popular DIP packaged opto-isolators or digital output high current driver transistors. Circuit pads are available for mounting voltage divider/threshold resistors and protection diodes.

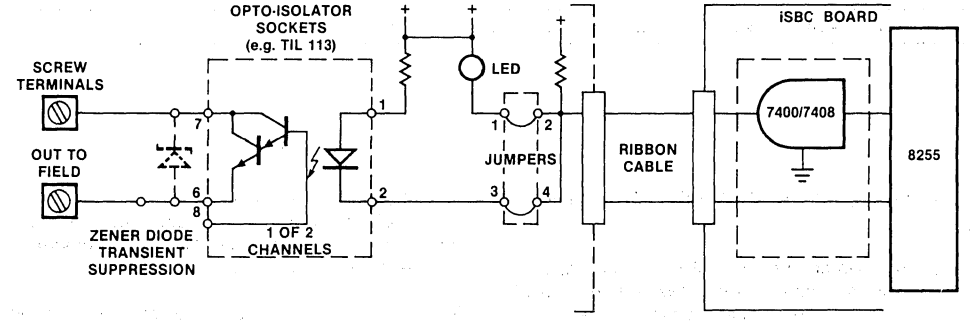
Groups of four inputs can have mixed voltage levels, opto-isolation, or straight through connections in groups of two. Output groups of four can be mixed opto-isolated or high current drive in groups of two. DIP components from a wide variety of vendors are selected and inserted by users based on their application. The iCS 920 manual recommends several alternative components and offers design assistance for your I/O configuration. Digital signal conditioning examples for several common industrial voltages are shown in Table 1 and in the diagrams below (see Figure 5).

Active Channel Indicators

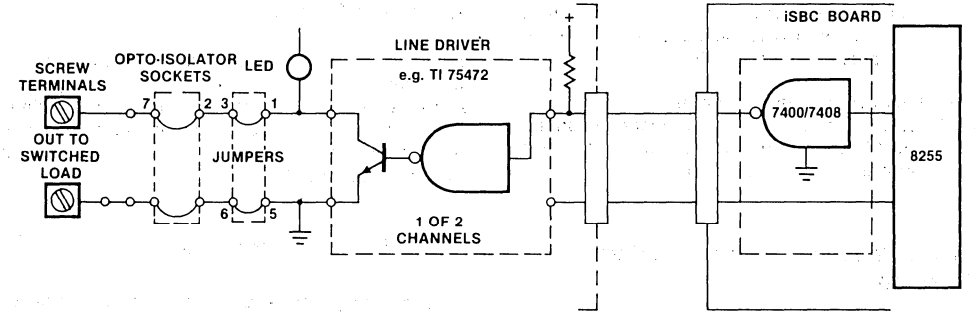
Light emitting diodes (LEDs) are mounted adjacent to each channel's screw terminals and may be jumpered in to indicate the Hi-Lo status of each of the 24 input or output channels.



OPTICALLY ISOLATED DC INPUT EXAMPLE (iCS-920 panel)



OPTICALLY ISOLATED DC OUTPUT EXAMPLE (iCS-920 panel)



CURRENT DRIVER OUTPUT (55V, 300 mA) EXAMPLE (iCS-920 panel)

Figure 5. Digital Signal Conditioning Examples

Table 1. iCS 920 Digital I/O Signal Conditioning Plug-In Component Examples

Digital Voltage Input or Output Load Voltage	Maximum Input Current (mA)	Threshold Voltage (V)	Opto-Isolators*	Diode Protection*
Opto-Isolated Input				
5 VDC	50	3	TIL117	1N4002
12 VDC	50	6	TIL117	1N4002
24 to 26 VDC	40	6	TIL117	1N4002
48 VDC	20	12	4N36	1N4002

	Maximum Output Current (mA)	Line Driver*	Opto-Isolators*
Opto-Isolated Output			
12 VDC	100	—	TIL113
24 VDC	100	—	TIL119
48 VDC	100	—	MCS 2
Current Drivers			
55 VDC	300	TI75472	—
Half Wave Rectifier Outputs			
24 VAC SCR	300	—	GE4N40
115 VAC SCR	150	—	MCS 2

*Example component — alternate source components are listed in the iCS 920 Hardware Reference Manual.

iCS 930 AC Signal Conditioning/Termination Panel

The iCS 930 panel interconnects 16 2-wire digital input or output channels from barrier strip screw terminals to 16 bits of the digital I/O ports available on many Intel single board computers and digital I/O expansion boards. The iCS 930 panel differs from the iCS 920 digital signal conditioning/termination panel in that the iCS 930 panel handles higher AC or DC voltages and currents (up to 280V, 3A), such as those found on many 115 VAC machines, motor starters, and industrial control panels. The iCS 930 panel is also recommended for optically isolated DC outputs greater than 100 mA.

The iCS 930 screw terminals accept up to 14 AWG size wire each for differential (2 wires per channel) connections, or two 14 AWG size wires for daisy chaining grounds or power from external sources.

Modular Isolation/Switching with Easy Serviceability

Each iCS 930 panel accepts up to 16 user supplied, optically isolated input modules or optically isolated solid state switches, for either AC or DC voltages (see Figure 6). Each module is screw mountable/replaceable and can be mixed for AC or DC input, or AC or DC output, in groups of four. Among groups of four inputs (or outputs) each channel can be individually mixed for AC or DC input (or AC or DC output). The user pays only for those channels implemented. User supplied compatible modules are shown in Table 2.

DC and AC input modules are current actuated and thus provide a 5-ms filter against spurious noise spikes or contact bounce. AC solid state output modules provide zero crossing turn on to minimize arcing.

Protection Circuitry

Each of the 16 channels contain a socketed fuse to protect against overload. In addition, mounting pads are available on each channel output for user supplied voltage transient RC "snubber" components or inductive pulse suppression, e.g., metallic-oxide-varistor (MOV) for large motor starting.

Active Channel Indicators

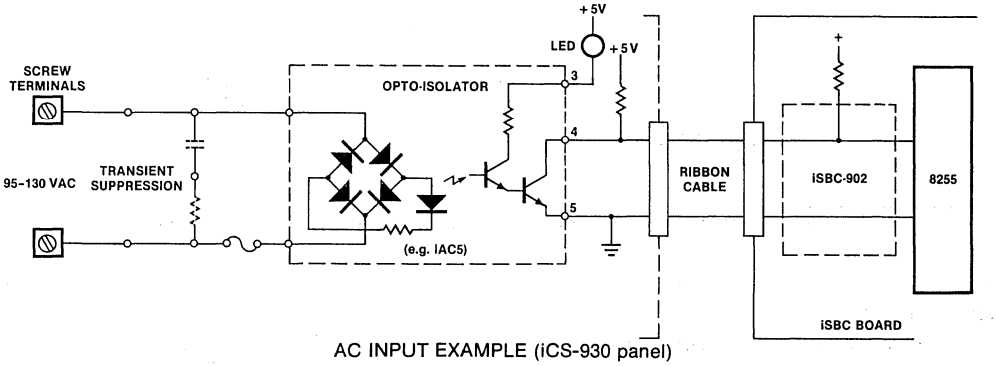
Light emitting diodes (LEDs) are mounted adjacent to each channel's screw terminals and opto-module to indicate Hi-Lo status of that channel and to assist in troubleshooting servicing.

Examples of iCS 930 input and output schematics are shown in Figure 6.

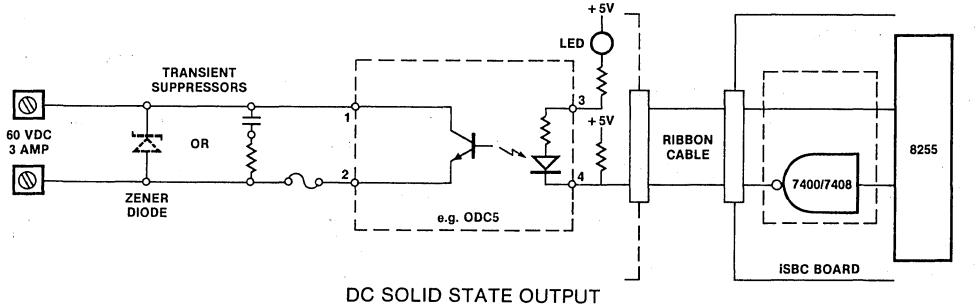
Table 2. Optically Isolated Modules Compatible with iCS 930 Signal Conditioning/Termination Panel

Signal Conditioning Desired	Voltage Rating	Maximum Input Current	Opto-22 Number*	Motorola Number*
AC Input — 115 VAC	95 to 130 VAC	10 mA	IAC5	IAC5
220 VAC	180 to 280 VAC	10 mA	IAC5A	
DC Input — 5 μsec Filter	10 to 32 VDC	32 mA	IDC5	IDC5
Fast, 50 μsec On	4 to 16 VDC	14 mA	IDC5B	
		Output Current Rating		
AC Output	12 to 140 VAC	3A	OAC5	OAC5
	24 to 280 VAC	3A	OAC5A	
DC Output	10 to 60 VDC	3A	ODC5	ODC5
	200 VDC	1A	ODC5A	

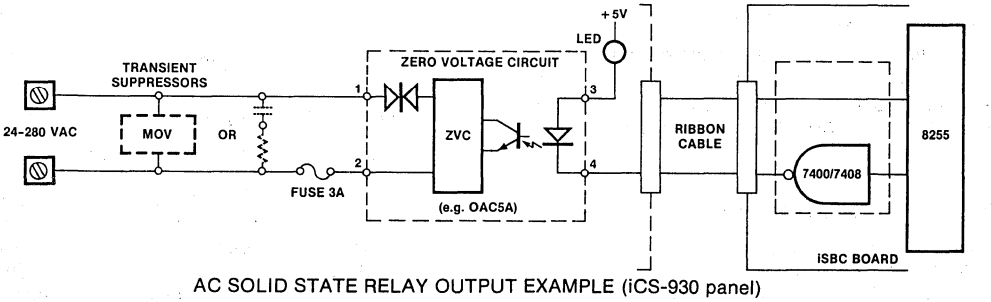
*Motorola and Opto-22 sales offices are located in North America, Europe, and Japan.



AC INPUT EXAMPLE (iCS-930 panel)



DC SOLID STATE OUTPUT



AC SOLID STATE RELAY OUTPUT EXAMPLE (iCS-930 panel)

Figure 6. Typical iCS 930 Signal Conditioning Examples

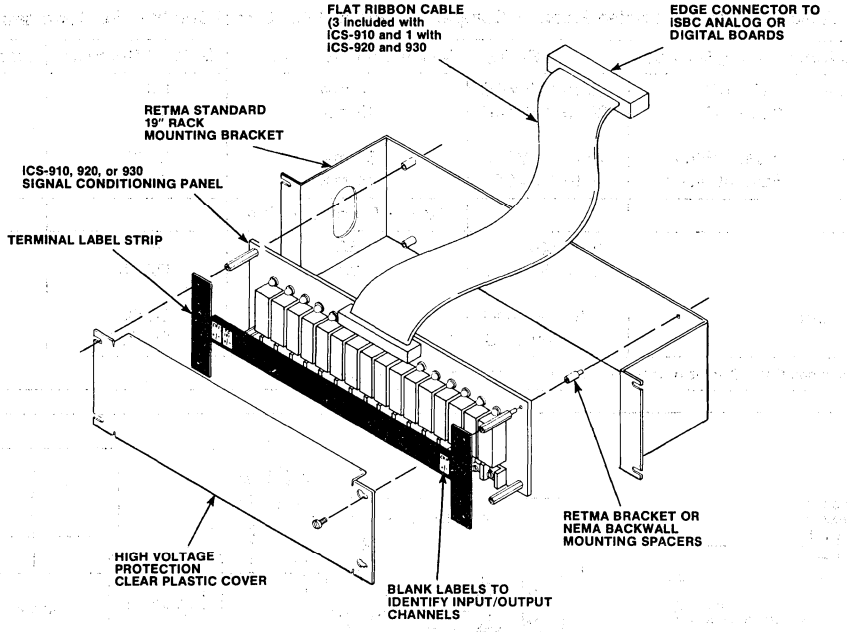


Figure 7. Mounting Arrangements for Signal Conditioning/Terminal Panels

SPECIFICATIONS

(For iCS 910/920/930 panels unless otherwise specified)

Number of Lines

iCS 910 Panel

Analog Inputs — Sixteen 3-wire (differential signal plus shield) or 32 single ended

Analog Outputs — Four 2-wire voltage output (iSBC 724 Analog Board) or two 2-wire current output (iSBC 732 Analog Board)

iCS 920 Panel — Zero to 24 digital inputs or outputs in groups of four

iCS 930 Panel — Zero to 16 digital inputs or outputs in groups of four

Isolation Characteristics

Line-to-Line Isolation — 250 VDC or RMS AC (iCS 910/920 panels), 500 VDC or RMS AC (iCS 930 panel)

Input/Output Isolation — 250 VDC or RMS AC (iCS 920 panel), 500 VDC or RMS AC (iCS 930 panel)

Physical Characteristics

Width: 36.63 cm (14.65 in.)
Height: 8.13 cm (3.25 in.)
Thickness: 0.24 cm (0.093 in.), iCS 910/920 panel
0.32 cm (0.125 in.), iCS 930 panel

	iCS 910	iCS 920	iCS 930
Weight:	(Minimum, PC panel only)		
	455 gm(16 oz)	455 gm (16 oz)	681 gm (24 oz)
	(Maximum with all components and mounting kit installed)		
	1.6 Kg (56 oz)	1.8 Kg (64 oz)	3.4 Kg (120 oz)
Depth:	(With components and clear plastic cover installed)		
	5.08 cm(2.0 in.)	5.08 cm (2.0 in.)	5.08 cm(2.0 in.)
Connectors:	(Barrier strip)		
	2/56 screws 48 AI 12 AO 2 power	2/56 screws 48 DI/DO 2 + 5V power	6/32 screws 32 DI/DO 2 + 5V power
	(J1, J2, J3 to iSBC boards)		
	50-pin 0.1 in. centers (2.54 mm)	50-pin 0.1 in. centers (2.54 mm)	50-pin 0.1 in. centers (2.54 mm)
	(Mating connector: 3M 3415-0000 or TI H3-12125)		

Maximum Distance from iSBC Boards

The iCS 910/920/930 panels are shipped with 4-ft. long cables. With customer provided 50-conductor or twisted pair ribbon cable, however, the iCS 910/920/930 panels can be mounted remote from the iSBC analog or digital I/O boards. In electrically quiet environments using normal iSBC board line driver/receivers, the iCS 910/920/930 panels should be able to operate up to 25 ft. (7.69m) from the iSBC board.

Electrical Characteristics

Power Requirements

iCS 920 panel — +15V ±5%, 25 mA max if iSBC 724 or iSBC 732 ±15V power is used for user mounted open circuit detection, or thermistor bias components. Additional power must be supplied via +15V terminal block.

iCS 920 panel — +5V ±5%, 1.46A max (24 channels high current drive)

ICS 920 Channel Configuration	Maximum per Channel Current (Includes pullups, LEDs, Isolators, drivers)
TTL in	23 mA
TTL out	23 mA
Opto-isolated in	23 mA
Opto-isolated out	41 mA
Open collector driver output	61 mA

Note: Both iCS 920 and iCS 930 panels have jumpered provision for externally supplied +5V power via a screw terminal block.

iCS 930 panel — +5V ±5%, 320 mA max. Output AC or DC channel: 20 mA/chan max; Input AC or DC channel: 12 mA/chan max.

Maximum Power Dissipation

iCS 910 panels — 3 watts with 16 channels analog input signal conditioning and +15V external compliance voltage

iCS 920 panels — 12 watts with 24 channels each containing high current driver outputs

iCS 930 panels — 80 watts with 16 channels of AC or DC output

Underwriters Laboratory (UL) Recognition

The iCS 910/920/930 signal conditioning/termination panels have been submitted to Underwriters Laboratories for approval as a UL recognized component under the UL safety standard for process control equipment, UL 1092.

Environmental Characteristics

Operating Temperature — 0 to 70°C (32°F to 158°F)

Relative Humidity — 0 to 90%, noncondensing

Hardware Supplied

ICS 910 — Analog Signal Conditioning/Terminating Panel, three 4-ft, 50-conductor flat ribbon cables with connectors, and installation kit below

ICS 920 — Digital Signal Conditioning/Termination Panel, one 4-ft, 50-conductor flat ribbon cable with connectors, and installation kit below

ICS 930 — AC Signal Conditioning/Termination Panel, one 4-ft, 50-conductor ribbon cable with connectors, and installation kit below

Installation kit consisting of RETMA (19″ rack) mounting bracket, clear plastic safety cover, labelling strip with blank gummed labels, hex standoffs and mounting screws

Documentation Supplied

A schematic diagram and assembly diagram are supplied with each iCS 910/920/930 panel.

Reference Manuals

9800800A — iCS 910 Analog Signal Conditioning/Termination Panel Hardware Reference Manual (NOT SUPPLIED).

9800801A — iCS 920 Digital Signal Conditioning/Termination Panel Hardware Reference Manual (NOT SUPPLIED)

9800802A — iCS 930 AC Signal Conditioning/Termination Panel Hardware Reference Manual (NOT SUPPLIED)

9800798A — iCS 80 Systems Site Planning and Installation Guide (NOT SUPPLIED), but supplied with iCS 80 Industrial Chassis

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

Installation

Complete instructions for installation and service are contained in the applicable iCS 910/920/930 Hardware Reference Manual. Additional system level information is available in the iCS 80 Systems Site Planning and Installation Guide, including RETMA and NEMA cabinet mounting, field signals, ground wiring and cooling suggestions.

Warranty

The iCS 80 Industrial Chassis is warranted to be free from defects in materials and workmanship under normal use and service for a period of 90 days from date of shipment.

ORDERING INFORMATION

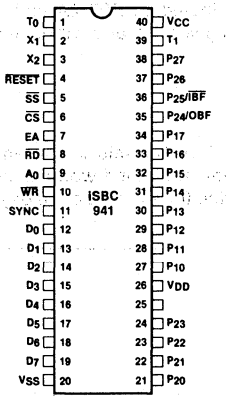
Part Number	Description
iCS 910	Analog signal conditioning/ termination panel
iCS 920	Digital signal conditioning/ termination panel
iCS 930	AC signal conditioning/termination panel

iSBC 941
INDUSTRIAL DIGITAL PROCESSOR

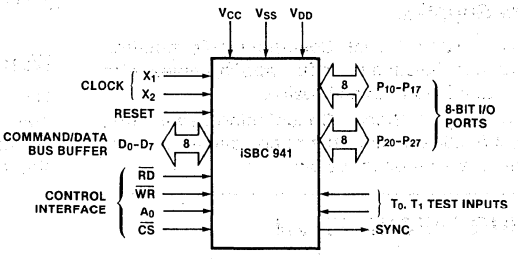
- **Provides measurement and control of common industrial digital input and output signals**
  - **— Sense change state**
  - **— Pulse counting**
  - **— Pulse generation**
  - **— Period measurement**
  - **— Frequency measurement**
- **Off-loads host processor from time-consuming task of digital I/O processing**
- **Simplified command protocol with MCS 80/85/86 "Master" Processor**
- **Compatible with 8041A Universal Peripheral Interface (UPI-41A) sockets such as those provided on Intel iSBC 80/30 and the iSBC 569 Intelligent Digital Processor**
- **Applications include**
  - **— Switch sensing**
  - **— Motor speed control**
  - **— Stepper motor actuation**
  - **— Serial communications**
- **16 programmable I/O lines, TTL compatible**
- **Single +5V supply**

The iSBC 941 Industrial Digital Processor is a 40-pin DIP device which provides the user with easy-to-use processing of digital input and output signals desired in many industrial automation environments. One of nine digital I/O functions can be selected at any one time for measuring, counting, or controlling up to 16 separate I/O lines. Additional utility commands allow reading or setting the condition of unused I/O lines. Simplex serial input and output modes can assemble or disassemble bytes transmitted asynchronously over TTL lines, including insertion and deletion of start/stop bits. The device has two 8-bit, TTL compatible I/O ports.

PIN CONFIGURATION



BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Industrial Digital Processor

Designed to operate as a slave device, the iSBC 941 processor may be requested to implement one of nine Primary Functions. These Primary functions are subroutines which are stored in program memory of the iSBC 941 device. Each Primary Function has a specific I/O task. Available Primary Functions include:

EVENT — Monitors and counts up to eight input lines for event counting or comparison to a preset count for each line. Interrupts may be generated or counter can be read 'on-the-fly' without changing its state.

FCOUNT — Measures frequency of one of eight digital inputs over a programmable period. Inputs may be selected under user program control or iSBC 941 processor may be requested to automatically scan inputs in sequential order, update, and hold or interrupt for reading each 16-bit counter. Input frequencies up to 18 KHz may be measured.

FREQ — Generates up to eight gated frequency outputs with separately programmable pulse width and periods and complementary synchronous outputs.

PERIOD — Measures the period of up to four inputs (single cycle).

SCAN — Monitors up to 16 input lines for change-of-state and direction of change. Change-of-state interrupts may be generated. User can individually disable inputs.

SERIN — Enables simplex reception and 8-bit byte assembling of asynchronously transmitted serial data bits for communications applications. Includes detection/deletion of start/stop bits. Baud rates up to 1200 baud may be programmed.

SEROUT — Enables simplex transmission of asynchronous serial data bits for communications applications. Includes insertion of start/stop bits. Baud rates up to 1200 baud may be programmed.

SHOT1 — Emulates a gated one-shot pulse generator (edge triggered and retriggerable modules) with programmable delay and period. Complimentary, synchronous one-shots can be created on separate output lines, on up to eight lines.

STEPPER — Generates up to eight programmable outputs that may be used for control of stepper motors. Step rate, step count, and step direction are user defined. Stepper motors designed for up to eight phase control may be controlled directly or stepper motor translator signals may be generated.

Any of the sixteen UPI processor I/O lines that are not used by a Primary Function are available for general purpose use; e.g., direct status reads or latched digital outputs, through the use of Utility Commands.

Commands recognized by the iSBC 941 Industrial Digital Processor are defined by one of two categories, Control Commands or Utility Commands. Control Commands are used to start and stop a Primary Function. Utility Commands are typically associated with moving a byte of information or reading the status of the iSBC 941 processor through the COMMAND/DATA Bus Buffer.

Control Commands available are:

ENFLAG — Enables the iSBC 941 processor to send interrupts via its I/O lines to the host processor.

INITPF — Selects the desired Primary Function and initializes parameters used by the Primary Function.

LOOP — Continuously executes the selected Primary Function at a specific rate.

PACIFY — Resets all iSBC 941 processor I/O lines to the input state and clears all control variables.

PAUSE — Commands the iSBC 941 processor to exit the LOOP or INITPF mode.

Utility Commands include:

CLRP1 (CLRP2) — Sets (to logic level 0) selected iSBC 941 processor Port 1 (Port 2) output lines. All other lines are unaffected.

ENP1IN (ENP2IN) — Enables user-defined mask to inhibit the writing of '0's by the iSBC 941 processor to Port 1 (Port 2) input lines. This function is used by SETP1 (SETP2), CLRP1 (CLRP2), STEP and LOOP.

IDEN — Requests the identity code of the iSBC 941 processor accessed.

LATCH — Transfers to holding area for reading all eight counters used by the EVENT Primary Function.

RDEC — Enables host processor to read one of eight user-specified event accumulators used by the EVENT Primary Function.

RDFQ — Reads bytes from the iSBC 941 processor's

first-in-first-out (FIFO) buffer (used by SCAN and SEROUT Primary Functions).

RDHR — Enables the host processor to read the contents of the iSBC 941 holding register specified (used by PULSE, PERIOD, and EVENT Primary Functions).

RDIDV — Enables the host processor to read the iSBC 941 processor's input-data-valid (IDV) flags.

RDLC — Enables host processor to read one of eight user specified EVENT counters previously stored by LATCH command.

RDP1 (RDP2) — Reads data byte present at iSBC 941 processor Port 1 (Port 2) and sends to host processor.

SETOE — Sets output enable parameter bits.

SETP1 (SETP2) — Sets (to logic level 1) selected iSBC 941 processor Port 1 (Port 2) output lines. All other lines are unaffected.

SETSF — Sets the Time Reference Period scale factor for value.

WRP1 (WRP2) — Allows host processor to write to iSBC 941 processor Port 1 (Port 2).

Simple Command Protocol

iSBC 941 functions may be implemented with minimum software overhead required of the host processor. An easy-to-implement protocol ensures that communication between the iSBC 941 processor and host CPU is straightforward and uncomplicated.

Implementing a Primary Function involves simple programming; the host processor transmits to the iSBC 941 processor a command byte followed by parameter bytes (the number of parameter bytes required is dependent upon the Primary Function selected). For example, to execute the Primary Function SCAN:

(1) Initialize iSBC 941 processor (Command byte)

(2) Select SCAN as Primary Function and select internal or external time reference (Parameter byte)

(3) Program Time Reference Period (scan rate — this byte is required only if internal time reference was selected in (2)) (Parameter byte)

(4) Define return message format (Parameter byte)

(5) Enable selected SCAN input lines (enabled in groups of eight) (Parameter byte)

(6) Terminate parameter list (Command byte)

(7) Enable iSBC 941 processor interrupt outputs (Command byte)

(8) Request execution (Command byte)

PIN DESCRIPTION

Signal	Description
D_0-D_7	Three-state, bidirectional COMMAND/DATA BUS BUFFER lines used to interface the iSBC 941 processor to an 8-bit master system data bus.
P_{10}-P_{17}	8-bit, PORT 1 quasi-bidirectional I/O lines.
P_{20}-P_{27}	8-bit, PORT 2 quasi-bidirectional I/O lines. Control can configure P_{24} as OBF (Output Buffer Full), P_{25} as IBF (Input Buffer Full) to send interrupt signals to master CPU.
$\overline{WR}$	I/O write input which enables the master CPU to write data and command words to the iSBC 941 COMMAND/DATA BUS BUFFER.
$\overline{RD}$	I/O read input which enables the master CPU to read data and status words from the COMMAND/DATA BUS BUFFER or status register.
$\overline{CS}$	Chip select input used to select one iSBC 941 processor out of several connected to a common data bus.
A_0	Address input used by the master processor to indicate whether byte transfer is data or command.
T_0, T_1	Input pins used by various iSBC 941 processor routines.
X_1, X_2	Inputs for a crystal, LC or an external timing signal to determine the internal oscillator frequency.
SYNC	Output signal which occurs once per iSBC 941 instruction cycle. SYNC can be used as a strobe for external circuitry.
RESET	Input used to reset status flip-flops and to prepare iSBC 941 processor to receive commands.
V_{CC}	+5V power supply pin.
V_{DD}	+5V during normal operation.
V_{SS}	Circuit ground potential.
EA	Circuit ground potential.
$\overline{SS}$	Connect to +5V through 10K-ohm pull-up resistor

ELECTRICAL CHARACTERISTICS

Absolute Maximum Ratings*

Ambient Temperature Under Bias . . . . . 0°C to 70°C
Storage Temperature . . . . . – 65°C to + 150°C
Voltage on Any Pin With Respect to Ground . . . . . 0.5V to + 7V
Power Dissipation . . . . . 1.5 Watt

*COMMENT: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

D.C. and Operating Characteristics

$T_A = 0°C$ to 70°C, $V_{SS} = 0V$, $V_{CC} = +5V \pm 5\%$

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
V_{IL}	Input Low Voltage (All Except X_1, X_2)	-0.5	0.8	V	
V_{IH1}	Input High Voltage (All except X_1, X_2, $\overline{RESET}$, $\overline{WR}$, $\overline{CS}$)	2.0	V_{CC}	V	
V_{IH2}	Input High Voltage (X_1, X_2, $\overline{RESET}$)	3.0	V_{CC}	V	
V_{IH3}	Input High Voltage ($\overline{WR}$, $\overline{CS}$)	2.2	V_{CC}	V	
V_{OL1}	Output Low Voltage (D_2-D_7, Sync)		0.45	V	$I_{OL} = 2.0$ mA
V_{OL2}	Output Low Voltage (All Other Outputs Except Prog)		0.45	V	$I_{OL} = 1.6$ mA
V_{OH1}	Output High Voltage (D_0-D_7)	2.4		V	$I_{OH} = -400\ \mu A$
V_{OH2}	Output High Voltage (All Other Outputs)	2.4		V	$I_{OH} = -50\ \mu A$
I_{IL}	Input Leakage Current (T_0, T_1, $\overline{RD}$, $\overline{WR}$, $\overline{CS}$, A_0,)		± 10	μA	$V_{SS} \leq V_{IN} \leq V_{CC}$
I_{OZ}	Output Leakage Current (D_0-D_7, High Z State)		± 10	μA	$V_{SS} + 0.45 \leq V_{IN} \leq V_{CC}$
I_{LI1}	Low Input Load Current ($P_{10}P_{17}$, $P_{20}P_{27}$)		0.5	mA	$V_{IL} = 0.8V$
I_{LI2}	Low Input Load Current ($\overline{RESET}$, SS)		0.2	mA	$V_{IL} = 0.8V$
I_{DD}	V_{DD} Supply Current		15	mA	
$I_{CC} + I_{DD}$	Total Supply Current		125	mA	

A.C. Characteristics

$T_A = 0°C$ to 70°C, $V_{SS} = 0V$, $V_{CC} = V_{DD} = +5V \pm 5\%$

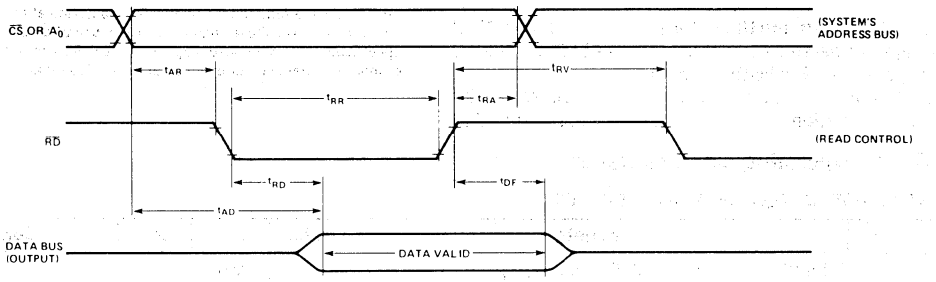
DBB READ

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
t_{AR}	$\overline{CS}$, A_0 Setup to $\overline{RD}$↓	0		ns	
t_{RA}	$\overline{CS}$, A_0 Hold After $\overline{RD}$↑	0		ns	
t_{RR}	$\overline{RD}$ Pulse Width	250		ns	
t_{AD}	$\overline{CS}$, A_0 to Data Out Delay		225	ns	$C_L = 150$ pF
t_{RD}	$\overline{RD}$↓ to Data Out Delay		225	ns	$C_L = 150$ pF
t_{RDF}	$\overline{RD}$↑ to Data Float Delay		100	ns	
t_{RV}	Recovery Time Between Reads And/Or Write	300		ns	
t_{CY}	Cycle Time	2.5	15	μs	

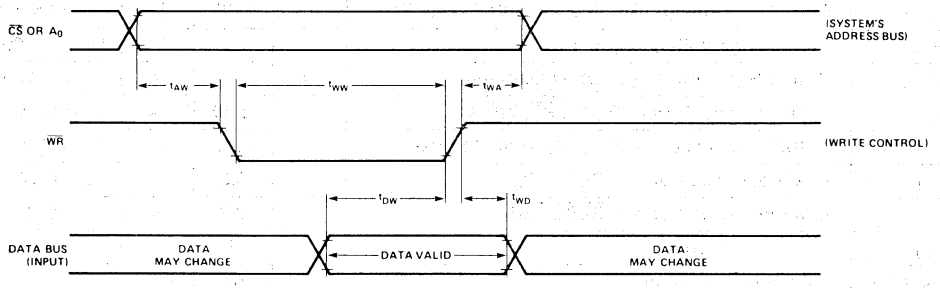
DBB WRITE

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
t_{AW}	CS, A_0 Setup to WR↓	0		ns	
t_{WA}	CS, A_0 Hold After WR↑	0		ns	
t_{WW}	WR Pulse Width	250		ns	
t_{DW}	Data Setup to WR↑	150		ns	
t_{WD}	Data Hold Aftert WR↑	0		ns	

1. READ OPERATION—DATA BUS BUFFER REGISTER.



2. WRITE OPERATION—DATA BUS BUFFER REGISTER.



Reference Manuals

9803077 — ISBC 941 Industrial Digital Processor User's Guide (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
SBC 941	Industrial Digital Processor

Run–Time Systems Software

5

iRMX 80
REAL-TIME MULTI-TASKING EXECUTIVE

- **Designed for Intel® iSBC 80/10B, iSBC 80/20-4, 80/24 and iSBC 80/30 based applications**
- **Completely user configurable through interactive configuration utility**
- **Priority-oriented scheduling**
- **Small, efficient nucleus**
- **Simple user task interface**
- **Comprehensive I/O support**
- **Library of flexible modules**
- **Structured application environment**

The Intel® iRMX 80 Real-Time Multi-Tasking Executive is an easy-to-use, sophisticated software system which operates on Intel's 8-bit single board computers and System 80 packaged systems. The iRMX 80 executive provides real-time facilities for priority-based resource allocation, intertask communications, standard I/O device control, and other features suitable for many applications including medical electronics, industrial process control, instrumentation, test systems, and data communications. The iRMX 80 package provides the framework that allows system developers to begin immediate application software implementation. The implementation and integration process is aided by the Interactive Configuration Utility (ICU80) program, a configuration tool which accelerates the development process.

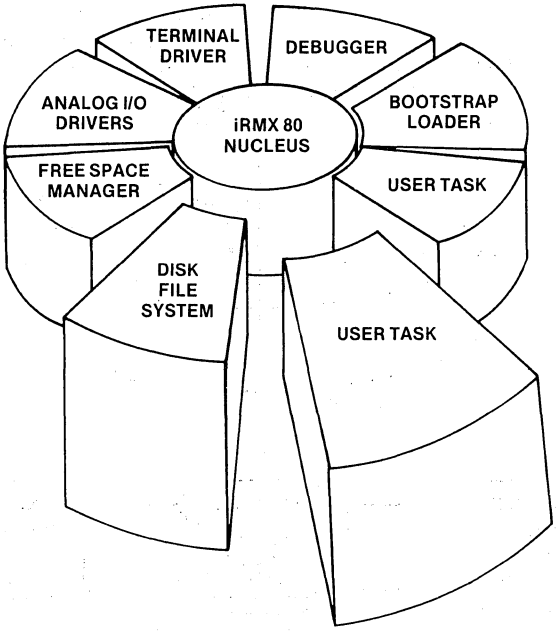


Figure 1. Structure Diagram

FEATURES

The iRMX 80 executive provides users of Intel Single Board Computers simple, easy-to-use tools for creating a wide range of application systems. The most popular features of the iRMX 80 are:

Structured Environment

The iRMX 80 executive provides a consistent structure, as diagrammed in Figure 1, from application to application thus allowing experience gained on one system to be easily transferred to others. Often, entire programs may be used in multiple applications.

Simple Interface

The iRMX 80 executive provides a simple, straightforward program interface for user programs. This interface is consistent throughout the range of facilities offered, reducing the number of concepts which must be learned.

Library Modules

The iRMX 80 executive is constructed in a thoroughly modular manner with the full range of facilities being offered in multiple library modules, see Table 1, allowing easy selection of the exact facilities required.

Small Nucleus

The iRMX 80 nucleus provides a small, efficient foundation upon which application systems may be easily built. A wide range of multi-tasking, real-time facilities such as intertask communication and control are included.

Priority-Oriented Scheduler

The iRMX 80 scheduler insures that the highest priority task which is ready to execute is given system control, allowing the application system to be responsive to its external world.

Comprehensive I/O Support

The iRMX 80 libraries contain support for a wide range of I/O boards supplied by Intel, simplifying the addition of peripherals to an application system. For applications which require custom boards the iRMX 80 device handler philosophy allows easy addition of user written handlers.

Interactively Configurable

The Interactive Configuration Utility (ICU80) program provides the user with an easy-to-use method of configuration of iRMX 80-based applications. Responding to questions from the ICU80 program running on the Intellec Microcomputer Development System, the user tailors the application system by selecting modules, e.g., nucleus flexibility as shown in Figure 2, from the wide variety of iRMX 80 facilities. The resultant system contains only the modules necessary for its use, allowing the iRMX 80 executive to fit a wide range of applications from small special purpose dedicated applications to large general purpose systems.

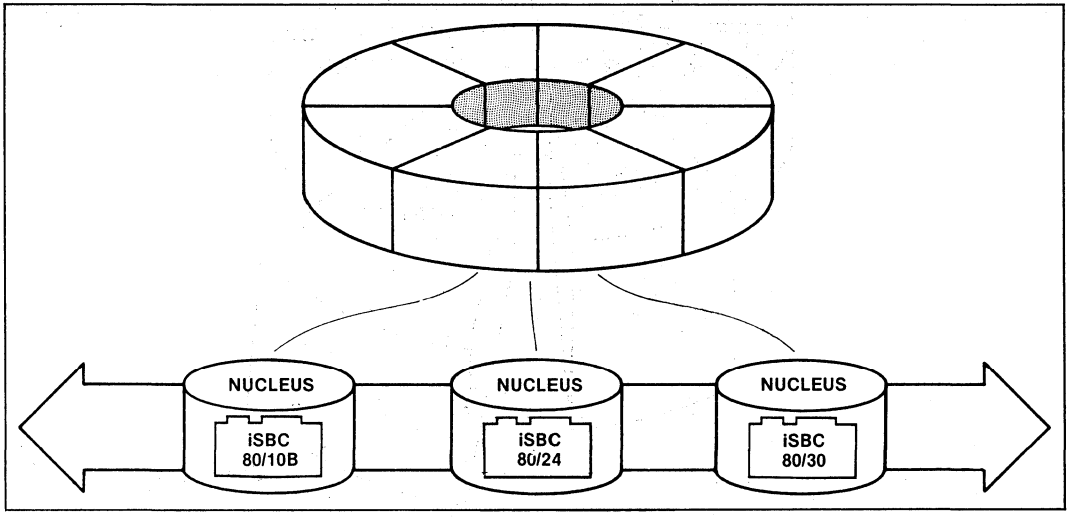


Figure 2. Configuration Flexibility Provides Application Freedom. The iRMX 80 executive allows you the freedom to choose from a wide range of iSBC family processors and peripherals upon which your application may be built. It allows you to break the software "chain" which ties your application to a single processor type and thereby gain application freedom.

Board Technological Support

The iRMX 80 executive provides support for a range of processor technologies from the 8080-based iSBC 80/10 Single Board Computer to the 8085-based iSBC 80/30 Single Board Computer. Applications are offered an easy upgrade path with the iRMX 80 executive which allows greater price/performance to be achieved without expensive software modification.

Extensive Debugging Aids

The iRMX 80 executive provides two user oriented, interactive software debugging aids. The debuggers allow memory examination and modification, execution breakpoints, and automatic stack overflow monitoring. These powerful aids allow simplified task debugging and faster application system development.

FACILITIES

The various facilities offered by the iRMX 80 executive are provided as independent library modules, thus allowing simple inclusion or exclusion, depending on the user's specific requirements. These facilities are described below.

Nucleus

The iRMX 80 executive provides nuclei for operation on various iSBC single board computers. The nuclei provide real-time scheduling, interrupt handling, intertask communications, and task control. The services offered are:

- Send a message from one task to another
- Accept a message from another task
- Wait for a message to be transmitted from another task
- Transmit a special interrupt message to a task
- Suspend execution of a task temporarily
- Continue execution of a previously suspended task.

Disk File System

The iRMX 80 Disk File System (DFS) provides for the filing and retrieving of data using disks. The iRMX 80 DFS allows for either Intellec Development Systems, ISIS-II compatible media format, or a user specified format. iRMX 80 DFS offers the following services in an ISIS-II compatible media format:

- OPEN a file for processing
- READ data from a file
- WRITE data to a file
- SEEK to a specific location within a file
- CLOSE a file to further processing
- RENAME a file
- DELETE a file.

For those applications which require unique media formats the iRMX 80 executive offers a level of processing which allows complete user flexibility in formatting data. The services offered are:

- SEEK to a specific track
- READ a sector
- WRITE a sector
- FORMAT a track
- RECALIBRATE to Track 0
- VERIFY a sector
- DELETE a sector.

Terminal Handler

The iRMX 80 terminal handler provides a data path between a console device (CRT or TTY) and user tasks. Communications between task and device are affected by using the nucleus services SEND and WAIT. Two versions of the terminal handler are offered:

1) *Full Terminal Handler* — The full terminal handler has a built-in interface to the debugger. It also provides for:
   - Correction of data previously input
   - Automatic buffering of data prior to a read request
   - Priority output path which allows "emergency" messages to bypass any other messages which may be queued for output
   - Automatic baud rate search to determine communications terminal speed.

Table 1. iRMX 80 Memory Requirement

Memory Requirements** (Bytes)									Minimum Development System Requirements		
Module	Nucleus	Full Terminal Handler	Minimal Terminal Handler	Free Space Manager	Disk File System	Disk I/O	Analog I/O	Bootstrap Loader & Initializer	Operating System	Memory Size (Bytes)	Minimum Diskette Drives
PROM*	2K	3K	600	1K	5.5K	700	800	600	ISIS-II	64K RAM	2
RAM	250	950	120	250	1.6K	100	50	900			

*Indicates amount of code which can be configured in PROM.
**All figures are approximate.

2) *Minimal Terminal Handler* — The minimal terminal handler provides a limited feature version of the full terminal handler for memory space critical applications. The minimal terminal handler provides for correction of data previously input.

Free Space Manager

The iRMX 80 free space manager provides the capability of dynamically allocating RAM memory based upon user requests. Requests may be made for any size memory blocks and will be accommodated based on memory availability. The free space manager services are:

- Request memory
- Release memory.

Analog Handlers

The iRMX 80 analog handlers provide a convenient mechanism for obtaining and transmitting analog values between user tasks and Intel's iSBC 711, 724 and 732 Analog Boards. The analog handlers offer a full range of services including:

- Repetitive channel input
- Sequential channel/single gain input
- Sequential channel/variable gain input
- Random channel/variable gain input
- Random channel output.

The input and output modules are individually configurable allowing greater application flexibility.

Bootstrap Loader

The iRMX 80 bootstrap loader allows those applications using disk to create essentially a "soft" system that may be loaded into RAM from disk rather than being permanently PROM resident. This provides greater application flexibility in building and supporting disk-based systems.

Interactive Configuration

The Interactive Configuration Utility (ICU80) program provides relief from the burden of manually creating hardware configuration tables and combining selected software components. Using the environment information, the iRMX 80 nucleus effectively controls and orchestrates the application system.

The iRMX 80 package provides two avenues for configuring applications; an effective macro mechanism allows specific detail manipulation of structures for the experienced iRMX 80 user or, secondly, an easy-to-use interactive ICU80 utility program generates the structures automatically. This latter program displays a clear and concise set of questions on the terminal of the Intellec Development System and elicits the configuration information about the application system, e.g. CPU TYPE: *80/30*, or TERM HNDLER: *FULL*. After describing the application system configuration the ICU80 program will initiate the housekeeping chores (linking and locating), thereby supplying the target iRMX 80 application. The result is the rapid development of the target iRMX 80 application system.

The iRMX 80 generation process allows application programs written in PL/M-80, FORTRAN-80, BASIC-80, or 8080/8085 Assembly Language to be merged with the specific iRMX 80 modules desired, see Figure 3. The system may then be debugged using Intel's sophisticated In-Circuit Emulation (ICE™) products or iRMX 80 debugger. The final application system is then available for either PROM or disk-based systems.

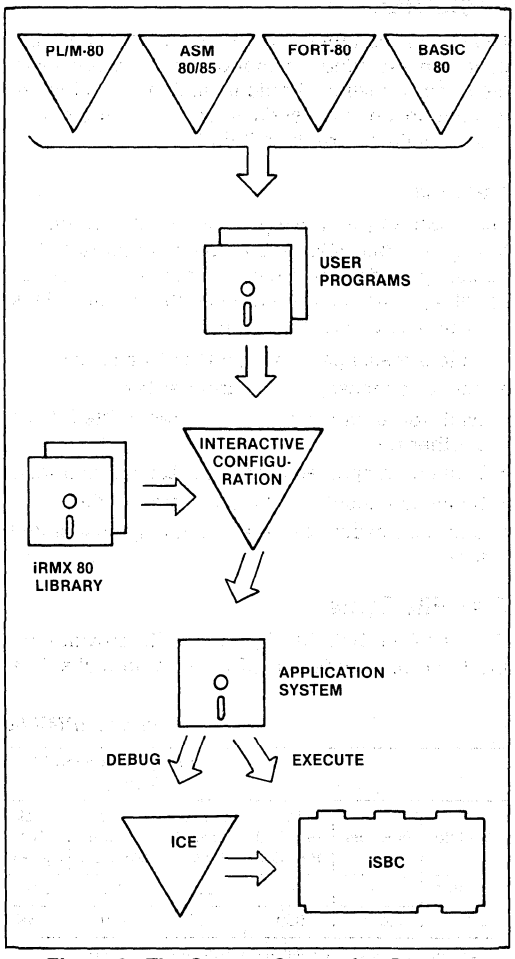


Figure 3. The System Generation Process

SPECIFICATIONS

Supported Hardware

SINGLE BOARD COMPUTERS

iSBC 80/10A
iSBC 80/10B
iSBC 80/20
iSBC 80/20-4
iSBC 80/24
iSBC 80/30

MASS STORAGE CONTROLLERS

iSBC 202
iSBC 204

ANALOG BOARDS

iSBC 711	iSBC 711A
iSBC 724	iSBC 724A
iSBC 732	iSBC 732A

iRMX 80 Executive Shipping Package

Single and double density diskettes containing:

- iSBC 80/10, 80/20, and 80/30 Nuclei
- Terminal Handler
- Minimal Terminal Handler
- Free Space Manager
- Disk File System
- Analog Handlers
- Debuggers
- Bootstrap Loader
- Configuration Macros
- Interactive Configuration Utility Program (ICU80)

Problem Reports

Reference Manuals

Reference Manuals

9800522 — iRMX 80 User's Guide (SUPPLIED)

9803087 — iRMX 80 Installation Instructions (SUPPLIED)

142603 — iRMX 80 Interactive Configuration Utility User's Guide (SUPPLIED)

143238 — Introduction to the iRMX 86/88 Real-Time Multitasking Executives

Reference Manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
RMX 80	Real-Time Multi-Tasking Executive

iMMX 800
MULTIBUS® MESSAGE EXCHANGE SOFTWARE

- **Supports 8 and 16-bit multiprocessing on the MULTIBUS® system bus**
- **Provides a uniform easily-used interface for iRMX™ 80, iRMX™ 86, iRMX™ 88-based distributed microcomputer processing applications**
- **Supports variable-length message transfers, up to 64K bytes, between MULTIBUS® devices**
- **Manages shared address space and transfers 8 and 16-bit data**
- **Meets the Intel MULTIBUS® Interprocessor Protocol (MIP) standard for reliable process-to-process message transfers**
- **Supports iSBC™ 550 Ethernet* Communications Controller with an iRMX™ I/O system driver interface**

The iMMX 800 MULTIBUS Message Exchange Software implements the MIP standard process-to-process message transfer protocol for loosely-coupled multiprocessing on the MULTIBUS bus. The iMMX 800 software reliably transfers a message, using shared memory, between processes based on iRMX 80, iRMX 88 Executives or the iRMX 86 Operating System residing on different iSBC board products (iSBC 80/24, iSBC 80/30, iSBC 86/05, iSBC 86/12A, iSBC 88/25, iSBC 88/40, iSBC 544, iSBC 550, iSBC 569 boards and compatible customer-designed products). The iMMX software provides a standard, easy-to-use system call to implement multiprocessing solutions for greater total application throughput; functions can be assigned to separate iSBC microcomputers, yet the communication and sharing of data remains straight-forward. New functions can be added to an existing system using the standard iMMX routine, thereby extending the useful life of a system.

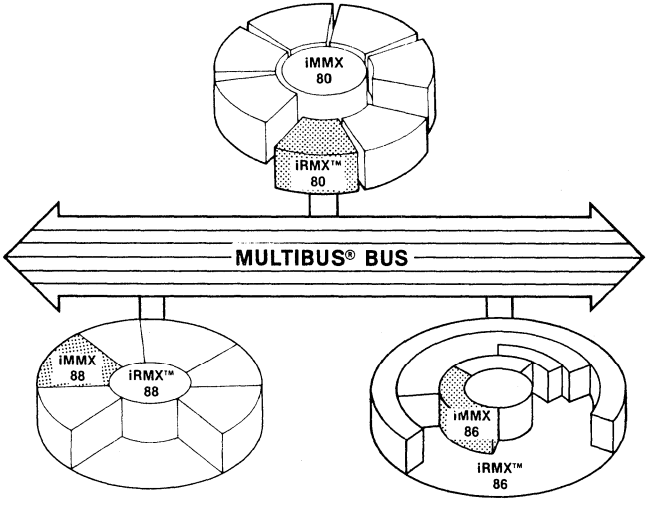


Figure 1. iMMX 800 Real-Time Executive Interface

The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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.

* ETHERNET is a trademark of Xerox Corp.
© INTEL CORPORATION, 1981

October, 1981
Order Number: 143875-002

FUNCTIONAL DESCRIPTION

The iMMX 800 MULTIBUS Message Exchange Software provides the iRMX 80, iRMX 88 Real-Time Executives and the iRMX 86 Operating System a standardized, process-to-process communications protocol. This protocol provides the fundamental capabilities needed to exchange data between multiple 8-bit and 16-bit microcomputers residing on the same MULTIBUS system bus. User-defined application tasks, on different devices, can exchange messages in a consistent manner. iMMX software supports error detection and reporting such that the user task can take appropriate action to ensure reliable message transfers.

MULTIBUS®-Based Networking

The iMMX 800 software supports a loosely-coupled MULTIBUS-based network system. The software interface is composed of simple, easy-to-use, modules (see Table 1). By supporting the addressing, data transfer, control, and memory management functions, the software, as shown in Figure 2, divides the operation into three functions; the virtual interface, the logical protocol, and the physical protocol.

Table 1. iMMX 800 Software Memory Requirements

Executive	K Bytes
iRMX™ 80	3.7
iRMX™ 88	4.5
iRMX™ 86	6.8

The **virtual interface** is the application task's access to the iMMX services. Through this interface, Task A can request connection (CONNECTION-ID) to an identified Task B. Using the CONNECTION-ID for Task B, Task A requests the iMMX memory manager to transfer a message via that CONNECTION-ID.

The **logical protocol** supports a message manager function. The Message Manager prepares the message for delivery to a specific destination port based on the connection specified. In addition, the logical protocol returns status information about the transfer.

The **physical protocol** is totally concerned with the hardware-related elements of message delivery. These elements are data flow control, mutual exclusion mechanics, address recognition and interactive signalling requirements.

Four different inter-device signalling mechanisms are supported. These asynchronous mechanisms include MULTIBUS interrupt-mapped, memory-mapped (i.e., "flag" bytes), polling, and I/O port-mapped.

8 and 16-Bit Distributed Microcomputing

The application solution can be a heterogeneous set of processors and real-time executives as shown in Figure 1. The iMMX 800 software provides a uniform interface to applications and manages a myriad of hardware combinations including multiple-function boards, single-function boards, master boards, slave boards, dual-port,

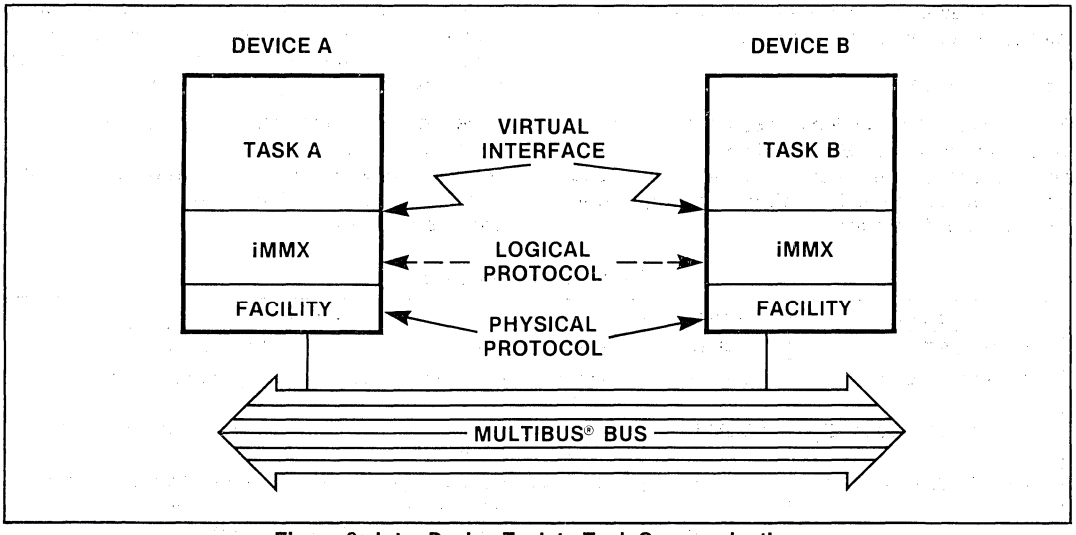


Figure 2. Inter-Device Task-to-Task Communications

private or public memory, 8-bit or 16-bit data transfers, and inter-device interrupt signalling methods. In particular, programs written to access 8 or 16-bit data through iMMX software can do so independently of actual data format.

iRMX™ Uniform Interface

The iMMX software package provides a uniform interface across all iRMX-based software environments. The iMMX software services are provided as a set of tasks, system procedures, and interrupt drivers.

Support is supplied for the iAPX 86/88-based microcomputers that support the iRMX 86 Operating System and the iRMX 88 Executive. In addition, software support is provided for the iRMX 80 Executive for the Intel 8085-based products: the iSBC 80/24 and iSBC 80/30 processor boards, and the iSBC 569 and iSBC 544 intelligent slave boards.

Message Transfer Mechanism

The iMMX software operational model is based on a message-passing model. Based on available memory space, the maximum message size is 64K bytes.

Shown in Table 2 are five different services available: Find Port, Activate Port, Transfer Message, Deactivate Port and Lose Port.

FIND PORT

The Find Port service returns a CONNECTION-ID, which is used by iMMX 800 software to pass messages to the destination port.

ACTIVATE PORT

The Activate Port service allows a task to receive messages sent via iMMX software. The application utilizes a convenient virtual-level interface, and does not have to maintain configuration-dependent information.

TRANSFER MESSAGE

The Transfer Message service causes messages to be delivered via the specified CONNECTION-ID. The competition status on the message's delivery is returned as a status variable.

DEACTIVATE PORT

The Deactivate Port service closes an active system-port. All successive messages sent to that system-port are returned to the sender.

LOSE PORT

The CONNECTION-ID provided by the initial Find Port request loses further reference to the system port. The CONNECTION-ID is no longer usable.

Shared Address Space

The iMMX 800 software package provides a memory management service that supports shared memory address spaces. An application transfers a message through a locally addressed port. The iMMX 800 memory manager maps that message, via a shared memory address, to the receiving port's same or aliased memory space. The receiving port then accesses the port through local addressing.

Interprocessor Protocol Architecture

The Intel MULTIBUS Interprocessor Protocol (MIP) specifies an architecture by which processes executing on different MULTIBUS single board computers can communicate with one another in a reliable, controlled manner within that system. A system can consist of a heterogeneous set of processors, executing a heterogeneous set of real-time executives and application software.

Based on a simple internal structure, the MIP specification defines a functional consistency across several product lines and provides the means to support efficient operation in multiple processor environments.

Table 2. System Calls

Function	Name	Description
FIND PORT	CQFIND	**Find** the system-port and return a CONNECTION-ID.
ACTIVATE PORT	CQACTV	**Activate** a local system-port for receiving messages from other tasks.
TRANSFER MESSAGE	CQXFER	**Transfer** the message to the system-port identified by the CONNECTION-ID.
DEACTIVATE PORT	CQDACT	**Deactivate** an active system-port. Messages addressed to it are no longer delivered, and are returned to the sender.
LOSE	CQLOSE	Loses the active CONNECTION-ID to a port.

iRMX™ I/O System Driver

Since an iRMX-based application may be using the iRMX I/O System's independent device driver interface, the iMMX 800 software is usable by device drivers.

An example of maintaining the iRMX-based application interface, the iMMX 800 package provides an iSBC 550 Ethernet Communications Controller device driver. This device driver uses iMMX 800 routines to communicate to the iSBC 550 controller (see Figure 3).

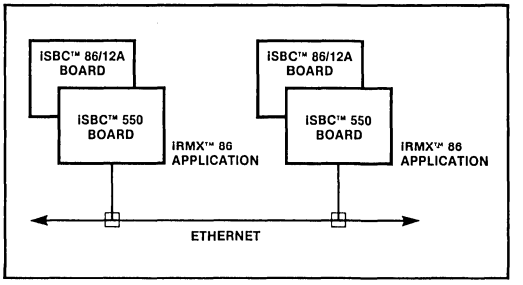


Figure 3. Ethernet Communications

SPECIFICATIONS

iSBC™ Supported Hardware

SINGLE BOARD COMPUTERS

iSBC 80/24
iSBC 80/30
iSBC 86/05
iSBC 86/12A
iSBC 88/25
iSBC 88/40

INTELLIGENT CONTROLLERS

iSBC 544 (Communications)
iSBC 550 (Communications)
iSBC 569 (Digital)

Reference Manual (Supplied)

143808 — iMMX 800 Reference Manual and Users' Guide

ORDERING INFORMATION

Description

The iMMX 800 MULTIBUS Message Exchange Software is a licensed product that provides users of Intel Single Board Computers and the iRMX 80, iRMX 86, and iRMX 88 (Version 1.0) Real-Time Executives software implementing a standardized, memory-based, task-to-task communication protocol. This protocol provides the capabilities needed to exchange data between multiple 8-bit and 16-bit microcomputers residing on the same MULTIBUS system bus.

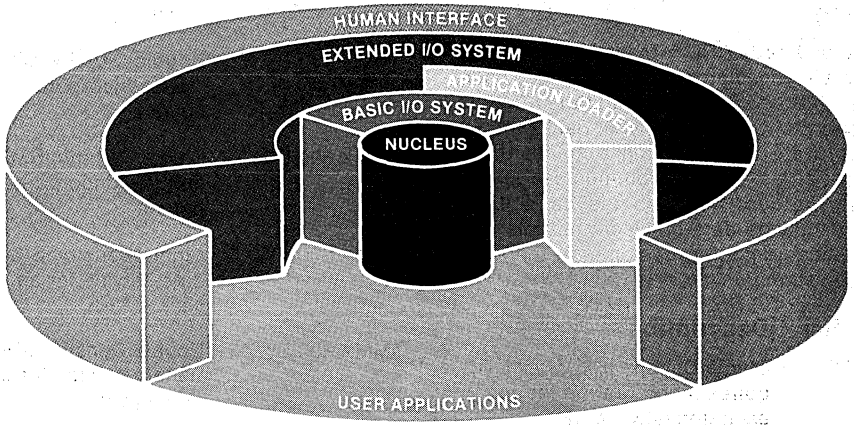
Part Number	Description
MMX 800 ARO	Single Density ISIS Media. Requires incorporation fee for each derivative work.
MMX 800 BRO	Double Density ISIS Media. Requires incorporation fee for each derivative work.
MMX 800 DRO	Single Density iRMX 86 Media. Requires incorporation fee for each derivative work.
MMX 800 ABY	Single Density ISIS Media. Includes incorporation fee buyout.
MMX 800 BBY	Double Density ISIS Media. Includes incorporation fee buyout.
MMX 800 DBY	Single Density iRMX 86 Media. Includes incorporation fee buyout.
MMX 800 AWX	Single Density ISIS Media. Update service for an additional year.
MMX 800 BWX	Double Density ISIS Media. Update service for an additional year.
MMX 800 DWX	Single Density iRMX 86 Media. Update service for an additional year.
MMX 800 LST	Human readable source listings for the iMMX 800 software modules.

iRMX 86™ OPERATING SYSTEM

- **Structured multiple application environment**
- **Object-oriented architecture**
- **(P)ROM or RAM based**
- **User configurable and extensible**
- **Real-Time priority-oriented scheduler**
- **Complete bootstrap and application loaders**
- **Powerful error management**
- **Comprehensive I/O system**
- **Extensive human interface**
- **Interactive system debugger**

The Intel iRMX 86 Operating System is an easy to use, comprehensive multiprogramming software system for Intel iSBC 86 and 88 Single Board Computers and other iAPX 86 and iAPX 88-based microcomputers. The iRMX 86 Operating System extends the architecture of the underlying processor by providing a collection of new operations that act on the Operating System objects provided by the system or user created extensions. The multiprogramming environment of the iRMX 86 Operating System, based on a real time, event-driven scheduler, provides an efficient foundation for applications including process control, intelligent terminals, office systems, data communications and medical electronics.

Each layer of the operating system simplifies user access to the underlying hardware by taking advantage of the mechanisms provided by the layers below. Each layer of the iRMX 86 Operating System is configurable to allow applications to customize the system to particular needs.



iRMX 86™ Operating System Layers

The following are trademarks of Intel Corporation and may be used to describe Intel products: CREDIT, Index, Intel, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS, and the combination of iCE, iSBC, iSBX, iRMX or iCS, and a numerical suffix. 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 CORPORATION, 1981

FUNCTIONAL DESCRIPTION

Services provided by the iRMX 86 Operating System include facilities for executing programs concurrently, sharing resources and information, servicing asynchronous events, and interactively controlling system resources and utilities. In addition, the iRMX 86 Operating System provides all major real time facilities including priority-based system resource allocation, means for concurrently monitoring and controlling multiple external events, real time clock control, interrupt management, and task dispatching. The iRMX 86 Operating System contains the following modules: An object-oriented Nucleus; Device Independent Basic and Extended I/O Systems; Terminal Handler; Bootstrap and Application Loaders; Human Interface with complete command line interpreter; and an interactive, object-oriented Debugger.

Because the modules and services provided by the operating system are user selectable, application specific operating systems can be created by iRMX 86 users. The iRMX 86 Operating System therefore eliminates the need for custom operating system design, thereby reducing development time, cost, effort, and risk.

FEATURE OVERVIEW

The iRMX 86 Operating System provides users with simple, easy-to-use, quality software tools for creating a wide range of application systems. Some important features are:

Structured Application Environment

The iRMX 86 Operating System provides a consistent structure from application to application, and CPU to CPU, thus allowing experience gained on one system to be easily transferred to others. Often entire programs can be ported from one application to another.

Object-Oriented Architecture

The iRMX 86 Operating System extends the capability of the underlying CPU by adding a number of new data structures (objects) and a number of functions to operate on these objects. This architecture provides a simple, symmetric, and easy to learn interface to a comprehensive system. The Nucleus provides the means to create, manipulate, and delete the basic objects necessary for any application. It also provides a mechanism for users to create their own object definitions and use them as part of the basic operating system. Each of the outer layers of the system add to the list of available objects by using this same extension mechanism.

(P)ROM or RAM Based

The iRMX 86 Operating System can be made resident in (P)ROM or can be loaded into RAM from a secondary storage device using one of the supplied Loaders. Being able to place all system software in (P)ROM offers three benefits: 1) Systems may be moved to harsh environments that preclude the use of disks; 2)The overhead expense of providing mass storage devices can be eliminated; and 3) System performance can be increased by eliminating the wait states required for most RAM's, and disk accesses required for most disk-based operating systems.

User Configurable

Users of the iRMX 86 Operating System are able to use a wide range of features or select only those which meet the specific requirements of a particular application. Each system call provided by the operating system may be removed from the system if it is not used. Each task can specify the use of the 8087 Numeric Data Processor. Jobs can be configured with specific running environments. Individual I/O port addresses are also configurable, making the system ideal for component level applications.

This complete modularity along with many other user options allows users to configure systems in a cost-effective manner regardless of the application environment. Each layer of the system may be configured in this manner, or (with the exception of the Nucleus) left out of the system altogether.

Nucleus

The Nucleus of the iRMX 86 Operating System provides the foundation upon which a variety of applications systems can be built. It includes the facilities to manage the basic objects of the system necessary to perform multiprogramming, multitasking, critical section management, and extensive task-to-task communication and control.

Embedded in the Nucleus are the facilities to support concurrent program execution and handling

of simultaneous asynchronous events. These facilities allow interrupts coming from specialized peripheral devices to be serviced in an efficient manner. The iRMX 86 Operating System allows the CPU hardware to be used by multiple applications, thus reducing the overall system size, complexity, and cost. These facilities are built from four key concepts:

OBJECT MANAGEMENT — Just as floating point numbers are data structures using operators such as multiply and subtract to operate on them, iRMX 86 objects are data structures with system calls to manipulate them. Because of the uniform structure of the system, users have a foundation on which to tailor the Nucleus to the application by removing system calls not necessary for the application and by adding objects and system calls customized for the application. The basic objects of the Nucleus are:

- SEGMENTS — Store data in dynamically created RAM buffers with a specified length.
- MAILBOXES — Provide a mechanism for intertask and interprogram object and data transfer. Mailboxes are locations for objects to be sent and received. For example, using a mailbox for intertask communication permits a time-critical task to forward data to a non-time-critical task for processing. Mailboxes are generally used to pass data segments from task to task, although any object (user or system defined) may be transferred.
- SEMAPHORES — Manage mutual exclusion and synchronization. A semaphore is used to signal another task when processing has been completed or when resources are available. A semaphore provides a low-overhead signalling mechanism.
- REGIONS — Control access to critical sections by allowing only one task at any given time to access a portion of code. Examples are a nonreentrant procedure or code for controlling a peripheral device that can only service one request at a time. In addition, regions can be used to protect data structures from being manipulated by more than one procedure at a time.
- TASKS — Perform the actual work of the application by executing software modules. Each task in the system has the characteristics of a unique processor. It has its own code, priority, stack, data area, and status. If the task is designated as using the 8087 Numeric Data Processor, it also has its own copy of the NDP registers, stack, and status. Task execution is based on an event-driven, priority-based scheduling algorithm.
- JOBS — Permit isolation of application tasks, objects, and memory to provide a multiprogramming environment. Jobs encapsulate an application and limit the degree of interaction between sets of tasks.
- COMPOSITE OBJECTS — Permit users to create objects not found in the set of Nucleus objects. These new objects appear to other facilities in the iRMX 86 system as if they are part of the original system. This means that Composite objects can be manipulated using the Mailboxes and Object Directories in exactly the same manner as other objects.

SCHEDULING — The iRMX 86 Nucleus offers a priority-oriented, event-driven scheduling mechanism that supports up to 255 different priority levels. The scheduler uses the task priority to determine which task receives control of the CPU, and to ensure that the highest priority task ready to execute is given control of the system. That task will continue to run until a higher priority interrupt occurs, or until the running task requests resources that are not available. This priority scheduling allows the system to be responsive to the external environment while allocating resources among the application tasks.

INTERRUPT MANAGEMENT — The iRMX 86 Operating System provides two levels of interrupt management: Interrupt Handlers and Interrupt Tasks. The first optimizes response time, the other optimizes response capabilities. Interrupt tasks allow use of all iRMX 86 system calls and mask only lower priority interrupts. Interrupt handlers permit direct control over the CPU's interrupt logic and only allow the use of interrupt system calls. This structure allows users to easily perform buffering of data while leaving complex processing of the data to interrupt tasks.

For systems requiring more than the 8 interrupt levels of the master 8259A interrupt controller, the iRMX 86 Operating System allows applications to configure up to 7 slave 8259A's. Using the slave devices, systems can respond in real time to as many as 57 interrupt sources.

ERROR MANAGEMENT — When a task issues an iRMX 86 system call, the results may not always

be what the task expects to achieve. For example, the task may request memory that is not available, or it may use an invalid parameter. The iRMX 86 Operating System may be configured to provide two levels of comprehensive error management: hierarchical error handling and selective error processing.

Hierarchical error handling permits a task to handle various errors at different levels of the system. Errors common to a number of tasks can be addressed by system-wide error handlers. Application-specific errors can be routed to job-level handlers. In addition, if a task has a need for a unique error handler, an error handler can be specified for that task. This flexibility means global error handlers can be created for the majority of the errors, reducing the amount of error handling software to be written.

In addition, each application can select the type of error to be processed by the error handlers. The errors are divided into two categories: programmer errors such as invalid parameters; and environmental condition errors such as detection of insufficient memory to meet requirements.

Terminal Handler

The Terminal Handler supports real time, asynchronous I/O between the operator's terminal and application tasks. It provides a line buffer which stores ASCII characters as they are input from the console. Special editing characters are used to control the terminal and the buffer contents, and are not entered into the data. The Handler may be configured as an output-only version to support those applications not requiring terminal input.

I/O System

The iRMX 86 I/O System provides an extensive facility for device independent I/O through a series of supplied device drivers, or any number of user supplied device drivers that can be configured to operate at any I/O port address. The Basic I/O System (BIOS) implements an asynchronous interface to the device drivers allowing users to explicitly overlap I/O functions with other operations. The Extended I/O System (EIOS) performs all of the synchronization necessary to do read-ahead and write-behind buffering automatically, and to reference files with logical names. By configuring the appropriate interface, applications can develop an I/O subsystem with the optimum degree of device control while requiring a minimum of design time and effort. Furthermore, the device-independent nature of the system allows use of different devices without redesign.

The I/O System provides access to three types of files:

- NAMED FILES — allow applications to refer to collections of bytes (files) by using a name. These names are cataloged in directories to allow file access by different tasks and jobs. Directories are special named files that store directory and access information about other named files and directory files.
- PHYSICAL FILES — provide a mechanism to make actual physical connections to storage devices. This type of file is typically used to communicate with simple devices such as printers and terminals.
- STREAM FILES — are mechanisms for communicating between tasks and jobs as if the data were written and then read from a FIFO file.

The named files may be organized in a hierarchical structure as shown in Figure 1, where the triangular files are named data files, and the rectangular files represent directories. This hierarchy allows data to be grouped logically and accessed with a minimum of overhead.

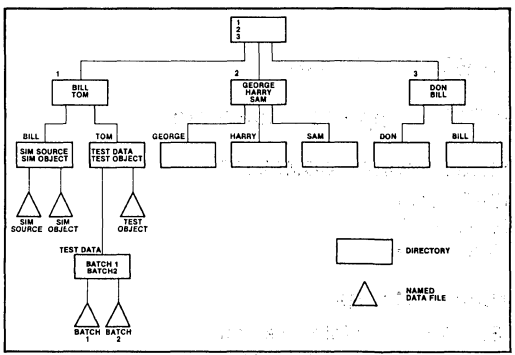


Figure 1. Example of Named-File Tree

Loaders

The iRMX 86 Operating System contains two loaders: A Bootstrap Loader capable of loading a file from mass storage into system RAM; and an Application Loader available to tasks as I/O system calls.

The Bootstrap Loader can be configured to load from a specific device, or to use the first device that becomes ready after the system has been started. It can also be configured to load a file specified by the operator at the system console.

The Application Loader provides a simple mechanism for loading application code and data files into the system. It can be used to load absolute code into a fixed location, or to load relocatable code into dynamically allocated memory locations. It can also be used to support code overlay functions.

Human Interface

The Human Interface is the uppermost layer of the iRMX 86 Operating System. It supports the user by providing a number of utilities useful in typical applications. It also provides the application programmer with a number of tools to generate custom utilities using the basic system utilities or by interfacing directly to the Command Line Interpreter.

Human Interface commands supplied with the iRMX 86 Operating System include commands to perform: creating a directory file; creating, copying, deleting, and renaming files; loading and starting application programs; formatting a device volume; and submitting a command file in a batch mode. The Human Interface also provides some utilities useful in debugging applications: a debug command enabling users to start commands via the system debugger, and a file copying facility used in conjunction with the iSBC 957A Monitor to convert MDS files to and from the iRMX 86 file format.

Interactive System Debugger

The iRMX 86 Operating System provides a comprehensive tool for interactive software debugging. The Debugger has two capabilities that greatly simplify the process of debugging a multitasking system. First, the Debugger allows users to debug several tasks while the balance of the application system continues to run in real-time. Second, the Debugger allows programmers to interactively view and modify system constructs as well as the system RAM and CPU registers. The debugger is structured to enable system designers to track system-wide problems easily. It can also remain in the final application as a continuous maintenance tool.

SPECIFICATIONS

iSBC™ Supported Hardware

SINGLE BOARD COMPUTERS

iSBC 88/40
iSBC 86/05
iSBC 86/12A

MASS STORAGE

iSBC 204	Flexible disk controller
iSBC 206	Hard disk controller
iSBC 208	Flexible disk controller
iSBC 215A	Winchester disk controller
iSBC 215B	Winchester disk controller
iSBC 220	SMD disk controller
iSBC 254	Bubble Memory board

MULTIMODULE™ BOARDS

iSBX 218	Flexible disk controller (when used with the iSBC 215)
iSBC 337	Numeric data processor
iSBX 351	Serial I/O channel

User iAPX 86 and iAPX 88 Based Systems

The iRMX 86 system runs on user designed boards with the following components:

8253	Programmable Interval Timer
8259A	Programmable Interrupt Controller
8251A	USART (When the Terminal Handler is configured into the system)
8087	Numeric Data Processor (when NDP tasks are configured into the system)

ORDERING INFORMATION

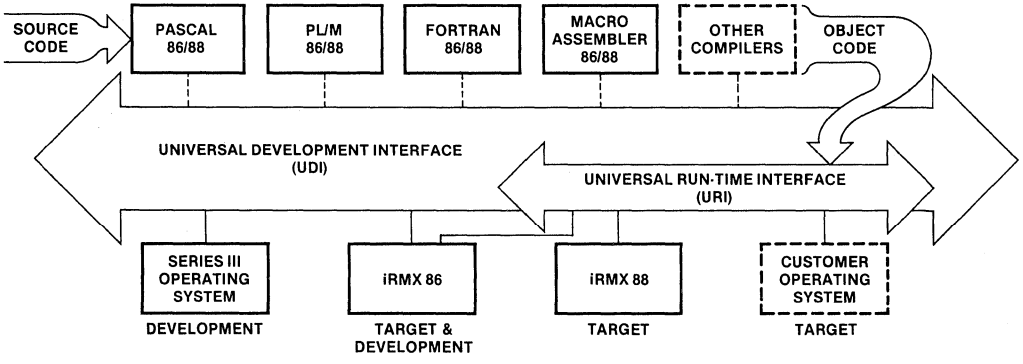
The ordering options for the iRMX 86 Operating System are listed below. All options include a full year of update service. All the options including the word "KIT" are shipped with a complete set of manuals, the iSBC 957B system monitor for the iSBC 86/12A and iSBC 88/40 Single Board Computers, and an iRMX 86 Customer Training Course credit voucher that is valid for 6 months after the date of purchase.

Part Number	Description
RMX 86 KIT ARO, and RMX 86 KIT BRO:	Single and double density OEM license requiring an Incorporation Fee for each derivative work
RMX 86 KIT AST, and RMX 86 KIT BST:	Single and double density license for one additional development site
RMX 86 KIT ABY, and RMX 86 KIT BBY:	Single and double density OEM Buy-out requiring no further Incorporation Fee
RMX 86 AWX, and RMX 86 BWX:	Single and double density update service for an additional year.
RMX 86 LST:	Source listings provided on Microfiche.

iRMX™ 86 LANGUAGES

- **Provide full software development capability for the OEM's iSBC™ 86/88 or iAPX™ 86/88 "target" systems**
- **Programming languages for OEMs to pass through to system end-users**
- **All languages generate Intel® Universal Run-Time Interface (URI) compatible programs in the Intel Standard Object Format (OMF)**
- **All implement the REALMATH™ standard for consistent and reliable results using the iAPX 88/20 or 86/20 Numeric Data Processors**
- **Resident on the iRMX 86 Real-Time Multiprogramming Operating System**
- **Newest technology languages**
  iRMX 861—ISO PASCAL
  iRMX 862—ANS 77 Subset FORTRAN
  iRMX 863—Intel PL/M 86
- **Compilers optimized to increase application performance and decrease application size**
- **Fully compatible with Intel Intellec® development system language products**
- **iRMX 86 utility software, including MACRO ASSEMBLER, EDIT, LINK, LOCATE and LIB**

The iRMX 86 Languages provide users of Intel's iAPX 86/88 or iSBC 86/88 microcomputers with full "on-the-target-system" development capability. This allows OEMs to provide their end-users with the facility to make on-the-spot modifications and add additional capability to their applications. All languages generate code which is compatible with Intel's Universal Run-Time Interface. This ensures users that their application system will run on the iRMX 86 or iRMX 88 Operating System using any iSBC 86/88 or iAPX 86/88-based system supporting the Universal Run-Time Interface. The iRMX 86 Languages are fully compatible with Series III Development System-based language products.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, REALMATH, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE, ICS, iAPX and iMMX. 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 CORPORATION, 1981

FEATURES AND BENEFITS

Major features of the iRMX 86 Languages and their benefits include:

Target System Development

OEMs may accomplish application software development on their target application hardware. This provides the greatest possible utility of the OEM's application system. Additionally, OEMs may provide an entirely new dimension in flexibility — reprogrammability of his system by the end-user.

Since they are resident on the iRMX 86 Operating System, the languages allow users to reduce the learning curve investment necessary for new application systems with only one operating system to learn.

New Technology Languages

The iRMX 86 Languages provide OEMs with the newest programming languages available. iRMX 86 PASCAL is fully compatible with the proposed ISO language standard. iRMX 86 FORTRAN provides the majority of the ANS 77 FORTRAN language features, including IF-THEN-ELSE, Zero Case Do-Loops, and Direct Access I/O. The complex arithmetic capability, the only missing major feature, will be supplied in a later release.

Efficient Application Code

The iRMX 86 Language products are optimized to provide a maximum efficiency in object code generation. This provides users with the smallest, fastest programs which a high-level language can generate.

Full Language Compatibility

The iRMX 86 Language products are compatible in three ways. They provide a Universal Run-Time environment for application programs, allow object modules to be linked together regardless of compiler used, and are fully source and object compatible with the Series III Development System language products.

The Universal Run-Time environment allows users to create their application software without regard for which Intel operating system it will run on. This allows maximum flexibility in allowing applications to easily move from one processor to another, and from one operating system to another.

All iRMX 86 Languages generate code compatible with the Intel Object Module Format (OMF) standard. This provides users with the capability to mix languages on a single application system. In this way a user can select exactly the right language tool for specific parts of the application rather than a project as a whole.

Intel Series III Development System languages provide all of the same benefits described, allowing users to develop their software on a system optimized for program development and then easily move it to the final system for test, debug and minor redevelopment.

Full REALMATH Support

The iRMX 86 Languages support the REALMATH floating point standard. This allows users of all iRMX 86 languages to access the iAPX 86/20 or iAPX 88/20 Numeric Data Processors or iSBC 337 MULTIMODULE™ board. These numeric processors offer over 100 times greater performance than comparible software-implemented algorithms, and reduce the system memory requirements by at least 16 KB. The REALMATH standard (proposed IEEE standard) povides universal consistency in results of numeric computations. The iRMX 86 Languages provide efficient object code generation and access to the highest performance floating point package available on microcomputers.

Complete Set of Languages and Utilities

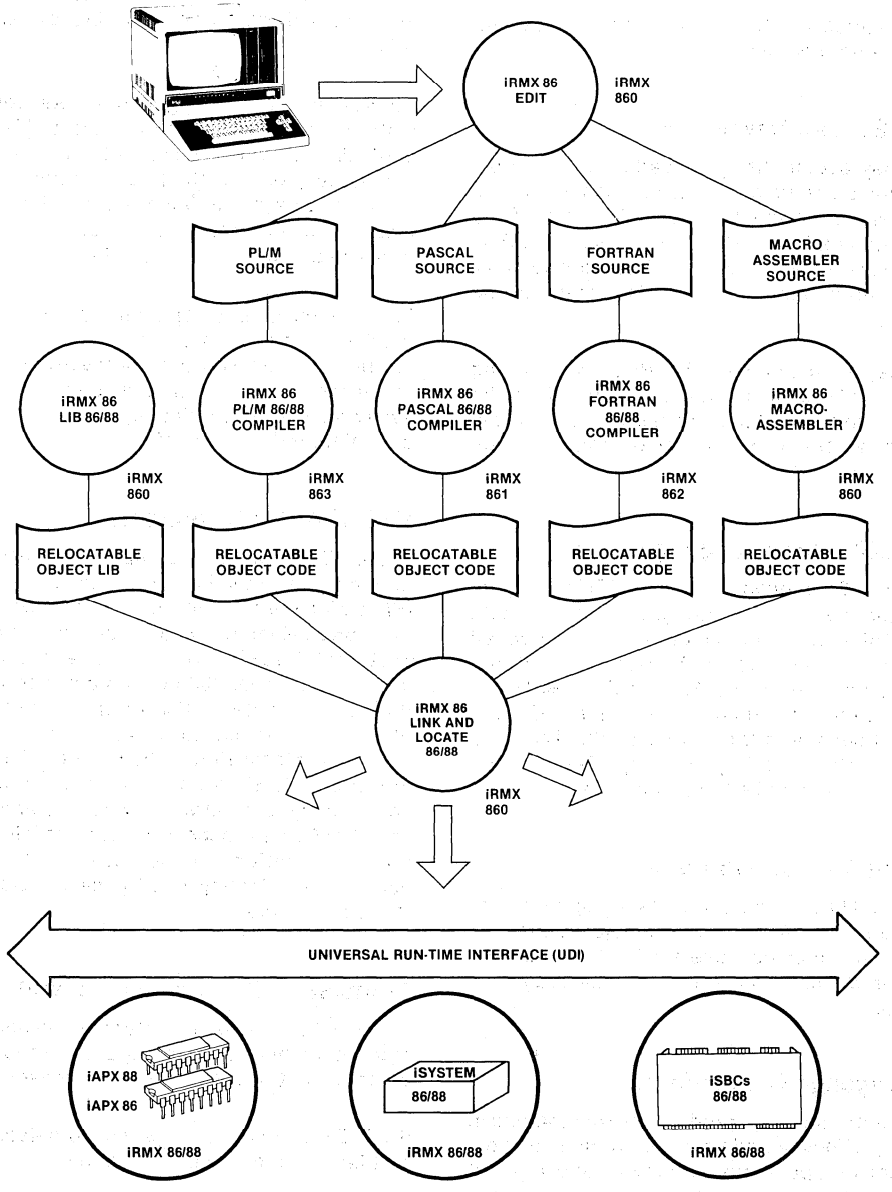
The iRMX 86 Languages offer a broad selection of modern, highly efficient language products and a complete set of target system software.

iRMX 86 Languages allow you to select the correct language for your application.

- Technical — FORTRAN or PASCAL
- Systems Programming — PL/M
- Commercial — PASCAL
- Size Optimized — MACRO ASSEMBLER

All necessary software for development is provided, including EDIT, LINK, LOCATE, and LIB.

Software Development Process



LANGUAGE HIGHLIGHTS

The iRMX 86 Languages allow OEMs to choose from a broad spectrum of specific language features and "on-the-target-system" development utilities.

iRMX 86 PASCAL (iRMX 861)

The iRMX 86 PASCAL compiler provides a strict implementation of the proposed ISO language standard. All source programs are validated by the compiler to ensure its conformance to the standard. Many extensions to the language are avail-

able which allow PASCAL programs to be written specifically for microcomputers. Features such as separate module compilation and iAPX 86/20, 88/20 Numeric Data Processor support are a few of its many. The ISO standard "source evaluator" can be switched off to accept these extensions. For more information on iRMX 86 PASCAL features, see the PASCAL 86/88 Software Package data sheet (121680).

iRMX 86 FORTRAN (iRMX 862)

The iRMX 86 FORTRAN compiler provides users total compatibility with existing FORTRAN 66 language-generated code, plus many new language features provided by the FORTRAN 77 language standard. These new features offer FORTRAN programmers many new capabilities, including "IF-THEN-ELSE", random access I/O and character variables. For a more detailed explanation of iRMX 86 FORTRAN, see the FORTRAN 86/88 Software Package data sheet (400630).

iRMX 86 PL/M (iRMX 863)

The iRMX 86 PL/M compiler provides users with a powerful, microcomputer-oriented system programming language. The PL/M 80 Language was introduced in *1976* by Intel. It was the first microcomputer-oriented, block structured, high-level language available. Since *1976*, thousands of users, shipping over millions of microcomputer-based systems, have generated their system software with PL/M 80 and PL/M 86.

iRMX 86 PL/M 86 is a compatible superset of PL/M 80 which offers easy portability of software across the full range of microcomputers supplied by Intel. For more information about iRMX 86 PL/M, see the PL/M 86/88 Software Package data sheet (402175).

iRMX 86 Utilities (iRMX 860)

iRMX 86 EDIT

The iRMX 86 EDIT program provides users with a powerful, sophisticated, line-oriented editing facility. EDIT delivers a range of capability suitable for novice users as well as advanced capabilities for sophisticated users. Its key features include a macro processor capable of creating and executing complex strings of commands, which ease the editing chore, as well as defining blocks of text which may be included anywhere in the text file. EDIT offers variable command sourcing, symbolic line numbering and reference by symbol.

The facilities of EDIT allow users to create, maintain and manipulate extensive libraries of source code with minimal effort. For more information on iRMX 86 EDIT, see the EDIT Software Package data sheet (143883).

iRMX 86 LINK/LOCATE

The iRMX 86 LINK program connects object modules which have been individually compiled into a single, relocatable object module. The input object code may have been produced by any Object Module Format-compatible compiler. Output object modules may be recombined into larger object modules, allowing work from a large programming staff to be easily integrated into an application system.

The iRMX 86 LOCATE program maps the relocatable object code into the iAPX 86/88 memory segments. Modules may be targeted for specific memory types. For example, those portions of the application which must be PROM resident can be mapped directly to the appropriate address range.

Both iRMX 86 LINK and LOCATE provide listings of resultant memory and symbol maps for easy reference and simplified debug. For more information on iRMX 86 LINK and LOCATE, see the 8086/8088 Software Development Package data sheet (9800757-04).

iRMX 86 LIB

The iRMX 86 LIB "Library manager" allows creation and maintenance of object module libraries. These libraries allow easy collection of related object code to reduce the overhead of maintaining many separate modules. Users may create new libraries, add and delete object modules, as well as list the contents of the library and their public symbols. Individual modules within the library will automatically be included in a total application system by the iRMX 86 LINK program. For more information on iRMX 86 LIB, see the 8086/8088 Software Development Package data sheet (9800757-04).

SPECIFICATIONS

Required Hardware

Any iAPX 86/88 or iSBC 86/88-based system capable of running the iRMX 86 Operating System, version 1.4 or later.

With:

- 96 KB of additional memory for the iRMX 86 Languages and utilities.
- Two iRMX 86-compatible disks (flexible and/or hard) with one flexible disk required for soft-distribution.
- System console device

Optional Hardware

iAPX 86/20, iAPX 88/20 or iSBC 337 Numeric Data Processors for support of the REALMATH standard.

Required Software

iRMX 86 Operating System, version 1.4 or later, including BIOS, EIOS, HI (128 KB of memory). An additional 96 KB is required for dynamic work space for the languages.

Documentation Packages

Package	Manual Included
iRMX 861	PASCAL 86/88 User's Guide (121539)
iRMX 862	FORTRAN 86/88 User's Guide (121539)
iRMX 863	PL/M 86/88 User's Guide (121636)
iRMX 860	— EDIT User's Guide (143587) — Macro Assembly Language Reference Manual (9800640) — Software Development Utilities Manual (980639)

ORDERING INFORMATION

The products listed below require the signing of an Intel® Master Software License Agreement. All DRO products below include 1 year of update service.

Ordering Code	Description
iRMX 86 Utility Package	**(EDIT, LINK, LOCATE, LIB, MACRO ASSEMBLER)**
iRMX 860 DRO	Single density, iRMX 86 compatible diskettes
iRMX 860 DWX	Update service on single density diskettes
iRMX 86 PASCAL 86/88	
iRMX 861 DRO	Single density, iRMX 86 compatible diskettes
iRMX 861 DWX	Update service on single density diskettes
iRMX 86 FORTRAN 86/88	
iRMX 862 DRO	Single density, iRMX 86 compatible diskettes
iRMX 862 DWX	Update service on single density diskettes
iRMX 86 PL/M 86/88	
iRMX 863 DRO	Single density, iRMX 86 compatible diskettes
iRMX 863 DWX	Update service on single density diskettes

iRMX™ 88
REAL-TIME MULTITASKING EXECUTIVE

- **Event-driven multitasking executive software supports iSBC™ 86/05, 86/12A, 88/25, 88/40 or iAPX 86, 88 based applications**
- **Small, high-performance, PROMable executive supports high sample rates**
- **Provides simple, intertask communications and synchronization**
- **Supports the 8087 Numeric Processor Extension (NPX) for arithmetic applications**
- **Supports component or iSBC™-based system generation through Interactive Configuration Utility**
- **I/O system provides compatible iRMX™ 86 files and device independent I/O interface**
- **I/O system supports the User Run-time Interface (URI) for PL/M, PASCAL and FORTRAN coded application tasks**
- **Memory management of full megabyte iAPX 86, 88 memory**

The iRMX 88 Real-Time Multitasking Executive is a small, event-driven single-user executive system. Designed for dedicated computer applications using iSBC 86/05, 86/12A, 88/25, 88/40 or iAPX 86, 88 custom products, the modular software package provides real-time application support for PASCAL, FORTRAN, PL/M, and assembler coded tasks. Application tasks utilize intertask communications, asynchronous I/O control, priority-based resource allocation and file support for the iSBC 204, 208, 215, 215/218, and 220 Disk Controllers, and the iSBC 254 Bubble Memory product.

The small, high performance iRMX 88 Executive can be located in EPROM or bootstrapped into RAM memory. The iRMX 88 Executive offers features that are suitable for performance-critical process control applications, production test stand units, sophisticated laboratory analysis, instrumentation, specialized data acquisition systems or monitoring stations. The iRMX 88 design, based upon the iRMX 80 Real-Time Executive, offers iRMX 80-like interfaces for those 8-bit applications which are upgrading to 16-bit solutions for the 1 Megabyte addressing, expanded application functions, and higher performance data sampling requirements.

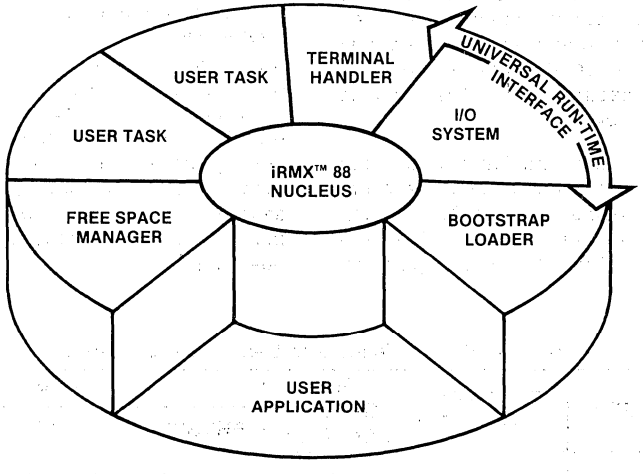


Figure 1. Module Representation

The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1981

October, 1981
Order Number: 143130-002

FUNCTIONAL DESCRIPTION

The iRMX 88 Real-Time Multitasking Executive Software package provides facilities for executing tasks concurrently, managing resources and servicing asynchronous events to users of Intel's single board computers and custom iAPX 86, 88-based products. The foundation modules support real-time dedicated computer applications with priority-based task scheduling, interrupt dispatching, real-time clock control with 1 ms resolution, multiple event monitoring and control, and file services for flexible, hard, Winchester, SMD disk units and bubble memory devices. The software package includes the primary modules: Nucleus, Free Space Manager, Terminal Handler, I/O System and Bootstrap Loader. The Interactive Configuration Utility (ICU) executes on a Series III Intellec System, or iRMX 86 Operating System with a Universal Development Interface (UDI).

FEATURE OVERVIEW

Event-Driven Multitasking

The iRMX 88 Executive provides a control software foundation called a Nucleus. The iRMX 88 Nucleus provides two major functions: first, the facility for concurrent task execution; secondly, the facility for handling simultaneous asynchronous events.

The structured multitasking environment permits segmenting of the application tasks. The number of tasks, managed by the Nucleus, is limited only by the available 1 Megabyte memory space. The tasks are prioritized such that the highest-ranked task is executing, e.g., an alarm event preempts the lower priority executing task. The Nucleus supports 255 priority levels.

Since internal or external events (interrupts) occur randomly, the Nucleus synchronizes the event with a task. The Nucleus supports either an interrupt service routine or an interrupt task. The interrupt service routine offers high-speed performance flexibility since it masks all interrupts and supports burst-rate data sample gathering. The interrupt task is useful for lower frequency interrupts, masking only lower priority interrupts.

Small High-Performance Executive

The iRMX 88 Executive software utilizes a simple, straightforward architecture which minimizes the memory requirements, as shown in Table 1. In addition, the modules are designed to be totally EPROM resident for those systems where mass storage devices cannot be used because of the danger of contamination.

Real-time microcomputer solutions require the recognition of interrupts. The performance of the system is with respect to data sample rates. If there is no activity in progress when an interrupt occurs, the time to handle that interrupt is dependent on the number of instructions executed, e.g., 175 microseconds interrupt latency time on an iSBC 86/12A board. Most real-time solutions have multiple events occurring and background operations in progress. Seldom does a background task have critical sections of code which cannot be interrupted.

Intertask Communications

The iRMX 88 Nucleus provides a simple, easy-to-use intertask communications mechanism based upon a message. Messages are transferred between tasks with two basic procedure calls, a send (RQSEND) and a wait (RQWAIT). Task "A" requests the Nucleus to RQSEND the pointer to a message buffer to Task "B" (see Figure 2). The Nucleus controls the message flow by activating the higher-priority Task B, or queuing the message if a lower-priority Task B is not waiting for the message. The receiving task does an RQWAIT to get the message pointer and can now access the data which may be for synchronization or real-time control operations.

Table 1. iRMX™ 88 Module Memory Requirements

MODULE	NUCLEUS	TERMINAL HANDLER	FREE SPACE MANAGER	I/O SYSTEM		
				PHYSICAL**	NAMED**	BOOTSTRAP
EPROM* (K bytes)	3.0	1.3	0.6	20.0	32.0	0.5

* amount of code configured in EPROM; all numbers are approximate
** includes one 3K byte device driver (named file plus physical file is 34.0K bytes)

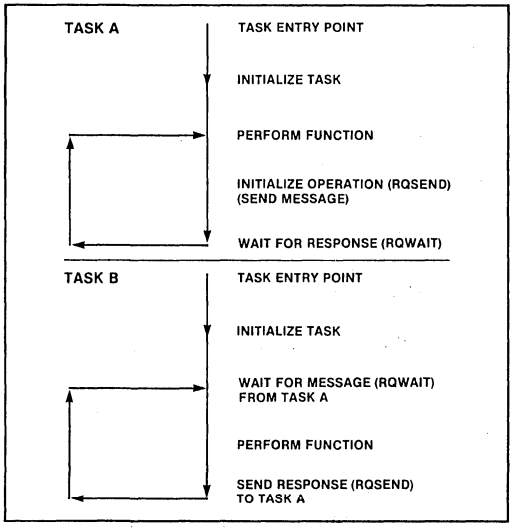


Figure 2. Intertask Communications

Numeric Data Processor

The iRMX 88 Nucleus fully supports the 8087 Numeric Processor Extension (NPX) functions for high-speed arithmetic functions of real-time applications. High-performance numeric processing applications, which utilize 8-, 16-, 32- and 64-bit integers, 32-, 64- and 80-bit floating point or 18-digit BCD operations, are accelerated up to 100 times over a iAPX 86, 88 software solution. The NPX functions, including trigonometric, logarithmic and exponential functionals, are essential in scientific, engineering, navigational or military applications.

Nucleus Primitives

The Nucleus performs other functions as shown in Table 2, in addition to the message communications management. Some primitives like CREATE TASK and DELETE TASK allow dynamic creation/deletion of tasks during run-time. This dynamic capability allows the Nucleus tables to

Table 2. Nucleus Primitives

NAME	FUNCTION
ACCEPT	Accept a message from specified exchange. Returns message address if available, zero otherwise.
CREATE TASK	Create task by building new Task Descriptor based on specified Static Task Descriptor.
CREATE EXCHANGE	Create exchange at specified RAM address.
DISABLE INTERRUPT	Disable specified interrupt level.
DELETE EXCHANGE	Delete specified exchange.
DELETE TASK	Delete the task specified.
ENABLE INTERRUPT	Initialize message portion of the Interrupt Exchange Descriptor associated with the specified interrupt level (the first time called only), and enable specified interrupt level.
END INTERRUPT	Signals specific end-of-interrupt for the specified interrupt exchange in a user-supplied interrupt service routine.
INTERRUPT SEND	Send an interrupt message to the specified interrupt exchange.
RESUME	Resume a task that has previously been suspended.
SEND	Send the message located at "msg-addr" to the exchange specified by "exch-addr."
SET INTERRUPT	Set interrupt vector address. An interrupt is to be serviced by the user-supplied routine starting at the address, thus bypassing Nucleus interrupt software.
SUSPEND TASK	Suspend execution of the task specified by the Task Descriptor.
WAIT	Wait at the specified exchange until a message is available or time limit expires. Return address of system timout message or user message.

expand and accommodate infrequently used tasks which are loaded into memory from a mass storage device.

Interactive System Generation

The iRMX 88 Executive is constructed in a thoroughly modular manner with the full range of facilities being offered in library modules. By selecting the appropriate features and combining them with the user-written application tasks the generated system is tailored to the application's requirements minimizing memory overhead for unused features.

An Interactive Configuration Utility provides a query-based tool that configures the iRMX 88-based application. Responding to questions from the ICU utility program executing on a Series III Intellec Microcomputer Development System or an iRMX 86-based system, the user quickly tailors the real-time application system.

I/O System

The iRMX 88 I/O System provides an extensive facility for device-independent I/O. Through a series of supplied iRMX 86 compatible device drivers, the I/O System supports a wide-range of iSBC peripheral controllers. Custom peripheral controllers are supported through user-written device drivers which are integrated with the I/O System at system configuration time. The device-independent nature of the system allows use of different devices without application redesign.

The I/O System (IOS) procedures manage real-time file operations supporting both sequential and random access (see Table 3). The IOS maximizes system throughput by allowing multiple disk operations to proceed in parallel. For example, files can be "double buffered" so that the task can be processing data in one buffer while the IOS is filing another.

The IOS provides access to two types of files:

- **Named Files** allow applications to refer to collections of bytes (files) by using a name. These names are cataloged in a directory which allows files to be accessed by different tasks.
- **Physical Files** allow applications to make a physical connection to a storage device. Typically used for simple devices such as printers, terminals or sequential data logging where file structures are not necessary.

The file types are a compatible subset of the iRMX 86 Basic I/O System with a flat (non-hierarchical) directory.

Bootstrap Loader

The iRMX 88 IOS has a Bootstrap Loader which loads a file from mass storage into system memory. The configurable Bootstrap Loader loads the file from a specific device, automatically from the first-ready device of a designated device list, or accepts the file name from a terminal. Storing the system software on disk allows easier future changes to the application system.

Run-Time Interface

The iRMX 88 Executive provides the User Run-time Interface (URI). This URI interface, in addition to encompassing the I/O System services, provides additional functionality for tasks. The additional functionality includes a trap function and memory management routines which provide the run-time foundation for PASCAL-86, FORTRAN-86, or PL/M-86 coded application tasks.

Table 3. I/O System Services

	SERVICE	FUNCTION
Data Transfer Services	CLOSE	**Closes** a file connection.
	OPEN	**Opens** a file connection for access.
	READ	**Reads** a number of bytes from a file.
	SEEK	**Seeks** to the indicated position.
	TRUNCATE	**Truncates** a file.
	WRITE	**Writes** a number of bytes to that file.
File Connection Services	ATTACH	**Attaches** to a file connection.
	CREATE	**Creates** a file and returns a file connection.
	CONNECTION STATUS	Returns the file **connection status**.
	DELETE	Marks the file for **deletion**.
	DETACH	**Detaches** a file connection.
	RENAME	**Renames** an existing file.
Volume Preparation	FORMAT	**Formats** the disk for files.

SPECIFICATIONS

Intellec® System Configuration and Generation Requirements

Series III Intellec Microcomputer Development System with UDI support and a minimum of 2 diskette drives.

iRMX™-Based Configuration and Generation Requirements

iRMX 86-based system with UDI support and a minimum of 2 diskette drives.

Supported Hardware

iSBC™ SUPPORTED MICROCOMPUTERS

iSBC 86/05 Board
iSBC 86/12A Board
iSBC 88/25 Board
iSBC 88/40 Board

MASS STORAGE

iSBC 204 Flexible Diskette Controller
iSBC 208 Flexible Disk Controller
iSBC 215A Winchester Disk Controller
iSBC 215B Winchester Disk Controller
iSBC 220 SMD Disk Controller
iSBC 254 Bubble Memory Board

MULTIMODULE™ BOARDS

iSBX 218 Flexible Disk Controller (when used with the iSBC 215 Controller)
iSBC 337 Numeric Data Processor
iSBX 351 Serial I/O Board

CUSTOM iAPX 86, 88-BASED SYSTEMS REQUIREMENTS

8253 or 8254 Programmable Interval Timer
8259A Programmable Interrupt Controller
8251A USART or iSBX 351 board (when the Terminal Handler is configured into the system).
8087 Numeric Processor Extension (when NPX tasks are configured into the system).

Reference Manuals (supplied)

143238 — Introduction to the iRMX 80/88 Real-Time Multitasking Executives

143241 — iRMX 88 Installation Instructions

143232 — iRMX 88 Reference Manual

142603 — iRMX 80/88 Interactive Configuration User's Guide

142926 — Guide to Writing Device Drivers for the iRMX 86 and iRMX 88 I/O Systems

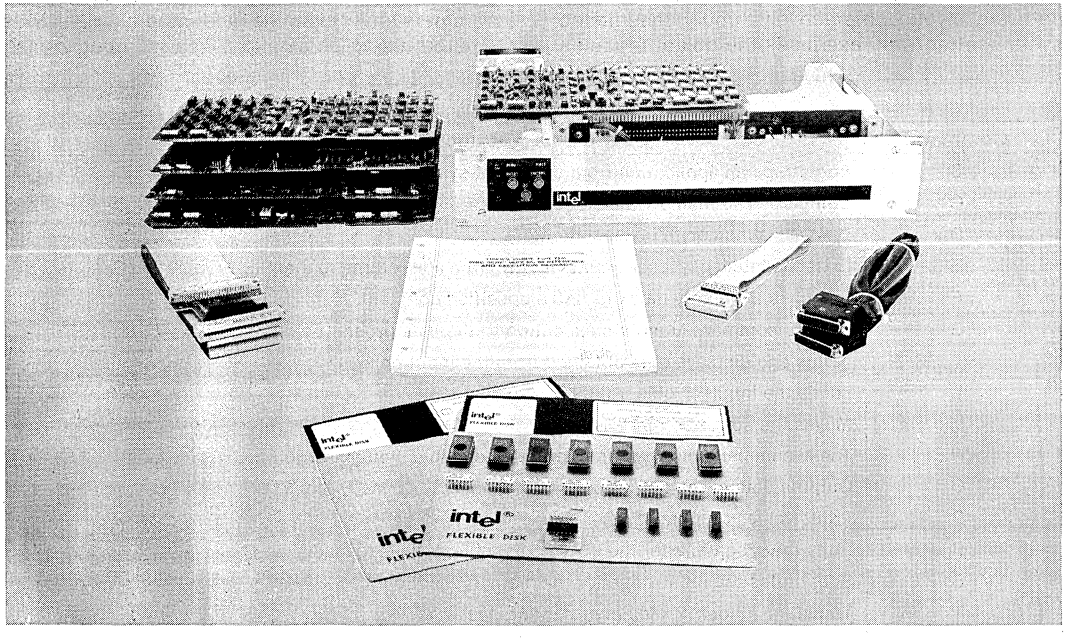
ORDERING INFORMATION

Part Number	Description
RMX 88	A licensed product which includes Nucleus, Terminal Handler, Free Space Manager, and I/O System object modules. Package also includes UDI-compatible Interactive Configuration Utility program for system generation and a complete set of manuals. Purchase price includes an iRMX 88 Customer Training Course credit.
RMX 88 ARO	Single Density ISIS media. Requires derivative work incorporation fee.
RMX 88 BRO	Double Density ISIS media. Requires derivative work incorporation fee.
RMX 88 DRO	Single Density RMX-86 media. Requires derivative work incorporation fee.
RMX 88 ABY	Single Density ISIS media. Includes incorporation fee buyout.
RMX 88 BBY	Double Density ISIS media. Includes incorporation fee buyout.
RMX 88 DBY	Single Density RMX-86 media. Includes incorporation fee buyout.
RMX 88 AWX	One year Single Density ISIS media update service.
RMX 88 BWX	One year Double Density ISIS media update service.
RMX 88 DWX	One year Single Density RMX-88 media update service.
RMX 88 LST	Human readable source listings for iRMX 88 software.
RMX 88 LWX	Update service for human readable source listings.
RMX 88 RF	Incorporation fee.

iSBC™ 957B
iAPX 86, 88 INTERFACE AND EXECUTION PACKAGE

- **Supports target system debugging for iSBC™ 86/05, 86/12A, 88/25, 88/40 or iAPX 86, 88-based applications**
- **Interactively extends the Intellec® development environment to the target system for code execution and symbolic displays of results**
- **Supports custom and iRMX™ operating systems with application access to ISIS-II files**
- **Supports the 8087 Numeric Processor Extension (NPX) functions for high-speed arithmetic applications**
- **Utilizes a parallel or serial connection between the Intellec® Development System and the target system**
- **Provides an applications bootstrap from iRMX™ 86 and 88 file compatible peripherals**

The Intel iSBC 957B package contains the necessary hardware, software, cables, terminator packs and documentation required to interface, through a serial or parallel connection, an iSBC 86/05, 86/12A, 88/25, 88/40 or iAPX 86, 88 target system to an MDS 800, Series II or Series III Intellec Microcomputer Development System for full-speed execution and debugging of application software. The iSBC 957B package supports the OEM's choice of a custom operating system, iRMX 86 Operating System or iRMX 88 Real-Time Multitasking Executive for the target application system. OEM's may utilize any iRMX 86, 88 supported target system peripheral for a bootstrap of the application system or have full access to the ISIS-II files of the Intellec system.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1981

October, 1981
Order Number: 210221-001

FUNCTIONAL DESCRIPTION

Overview

Extending the software development capabilities of the Intellec Microcomputer Development System, the iSBC 957B, iAPX 86, 88 Interface and Execution Package provides a link to executing and debugging in a target system. The application software, developed under the Intellec-resident ISIS-II Operating System, can readily be downloaded to an iSBC 86, 88 Single Board Computer or a custom iAPX 86, 88 system using the included monitor and its powerful, interactive debugging commands via the Intellec console. Programmers can effectively develop applications to ensure timely product availability.

Target System Debugging

The iSBC 957B package includes a communications link and target system resident monitor software for target application debugging. The target system monitor supports debugging through an attached CRT or the Intellec System. The Intellec resident communications software manages the link (serial or parallel) between the Intellec system and the target system. The communications software passes the appropriate console-requested commands to the monitor software. The monitor software, invoked interactively through the Intellec system, effectively exercises the object modules on the target system. Pre-configured EPROM resident monitors are supplied for the iSBC 86/05, 86/12A and 88/40 products. The monitor software is configurable as to the selection of the communication link, processor board, numeric processor extension and the bootstrap loader functions. The OEM would burn the configured monitor into EPROMs for the target system.

The execution command environment (see Table 1) supported by the resident monitor loads object code into memory, executes it at full speed, sets breakpoints and examines the results. The monitor selectively executes portions of program modules based on breakpoints and single stepping requests. The monitor also provides commands to examine memory, manage memory movement by block, search for values, compare contents, and modify its value. Other program debugging information is provided through the displaying, examining or modifying of the iAPX 86, 88 registers.

Numeric Data Processor Support

Arithmetic applications utilizing the 8087 Numeric Processor Extension (NPX) are fully supported by the iSBC 957B Monitor. In addition to executing applications with the full NPX performance, programmers may examine and modify the NPX's registers using decimal and real number format.

Table 1. Monitor Commands

Command	Function
B	**Bootstraps** an application from the target system's peripheral device.
C	**Compares** two memory blocks.
D	**Displays** a memory block's contents.
E	**Exits** the Intellec® -based (APXLOD) program and returns to the ISIS-II Operating System.
F	Searches a memory block to **find** a specified constant.
G	The **GO** command transfers execution to the user program.
I	**Inputs** and displays data obtained from the input port.
L	**Loads** the Intellec® object file into memory.
M	**Moves** the specified memory block.
N	The **N** command displays an instruction's single step execution.
O	**Outputs** data to the output port.
P	**Prints** values of literals.
R	**Runs** the program after the iAPX 86, 88 object file is loaded.
S	**Substitutes** the input value for the memory location.
T	**Transfers** a block of memory to an Intellec® file.
X	Allows iAPX™ 86, 88 or NPX registers to be **examined** or modified.
*	Indicates remainder of the line is a comment.

The programmer feels confident that correct and meaningful numbers are entered for the application without having to encode and decode complex real, integer, and BCD hexadecimal formats.

Easy-To-Use Connection

The physical interface between the Intellec Microcomputer Development System and the target iAPX 86, 88 system is accomplished with the supplied iSBC 957B cables. The cabling arrangement is either a serial or parallel line and varies depending on whether the development system is of the Intellec MDS 800 family or one of the Intellec Series II or III family. All communication, including data transfer and command requests, occurs over this line.

As shown in Figure 1, the connection to the target iSBC 86/12A system is accomplished with the iSBC 86/12A serial port through an RS232C serial line interconnected to Serial port 1 of an Intellec Series II Model 220 or Model 230 or the CRT port of an Intellec 800 Development System. In some target iSBC application systems, the serial I/O port may not be available for debugging. The example connection, shown in Figure 2, shows how the parallel port of the iSBC 86/12A board is connected through the parallel cable to the UPP port of the Intellec system.

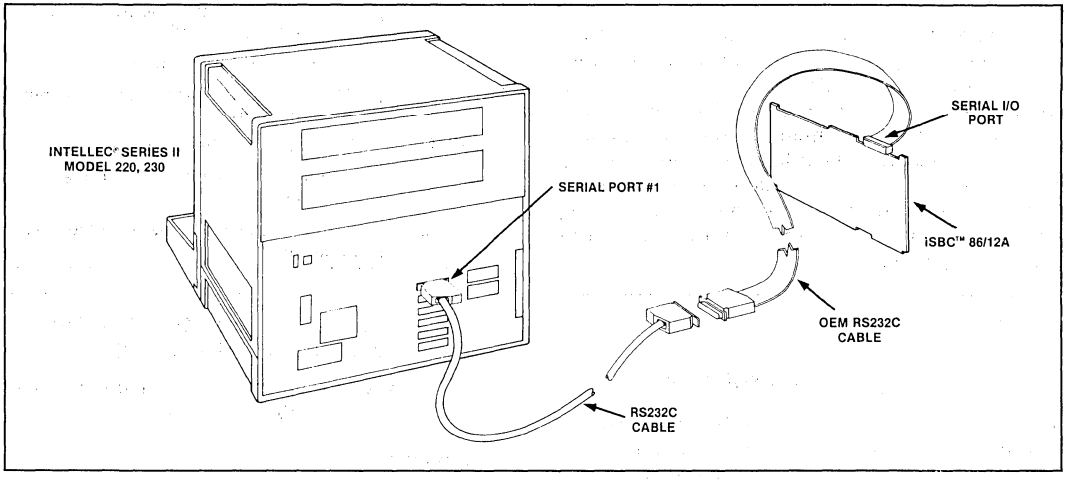


Figure 1. Intellec® Series Models 220, 230 Serial Connection

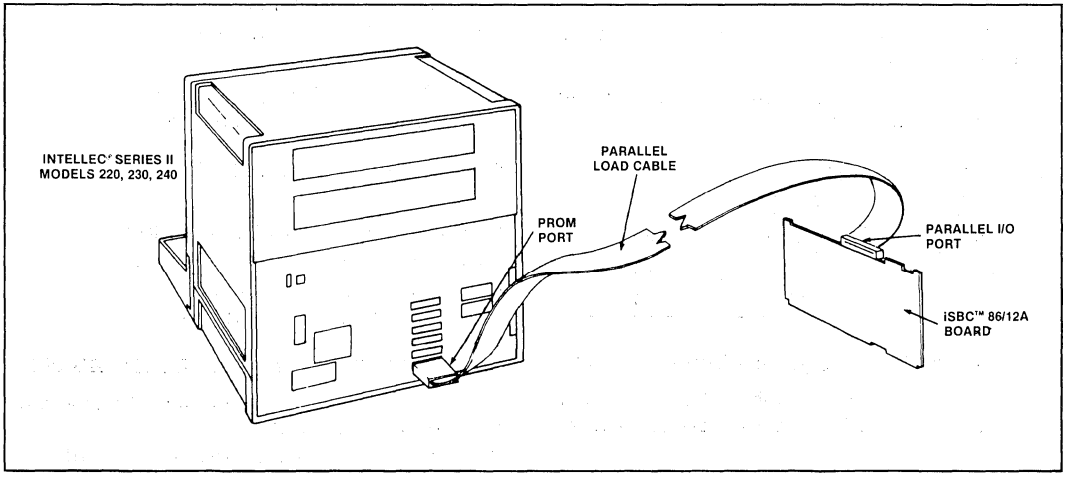


Figure 2. Intellec® Series Models 220, 230, 240 Parallel Connection

Application Bootstrap Loader

The bootstrap loader, invoked from a stand-alone CRT attached to the iSBC product or the Intellec console, dynamically loads the application system into the target system's memory from an attached peripheral through the iRMX 86, 88 compatible bootstrap loader. This configurable function can be included with the iSBC 957B monitor and installed in the iAPX 86, 88 target system. The programmer may load application code modules or an entire system from bubble memory, floppy diskettes, Winchester disks or other iRMX supported mass storage devices.

Application Access to ISIS-II Files

Application programs have read or write access to the ISIS-II file system from the iAPX 86, 88 target system through simple, easy-to-use, ISIS-like routines (as shown in Table 2). The routines are used by applications running in the target system to transfer files from the ISIS environment into the environment of the target system. User written applications, utilizing the target operating system, can manipulate or store data on the target system's peripherals.

Table 2. Routines for ISIS-II Services Available to Target System Applications

Routine	Target System Function
ATTRIB	Changes an ISIS-II file **attribute.**
CI	Returns a character **input** from the **console.**
CLOSE	**Closes** an opened ISIS-II file.
CO	Transfers a character for **console output.**
DELETE	**Deletes** the specified ISIS-II file.
ERROR	Displays an **error** message on the Intellec® console.
EXIT	**Exits** to the target system monitor.
LOAD	**Loads** target system memory with ISIS-II object code file.
OPEN	**Opens** an ISIS-II file for access.
READ	**Reads** up to 4096 bytes from an ISIS-II file to memory.
RENAME	**Renames** an ISIS-II disk file.
SEEK	**Seeks** to the specified ISIS-II file location.
WRITE	**Writes** up to 4096 bytes from memory to an ISIS-II file.

SPECIFICATIONS

Intellec® Configuration Environment

The Intellec Microcomputer Development System is utilized for application program development and requires the following to support the iSBC 957B package:

1. 64K bytes RAM
2. Double density diskette or single density diskette subsystem
3. ISIS-II Operating System and associated language translators

Data Transfer Rates

Intellec® System Family	Serial K Bytes/min	Parallel K Bytes/min
MDS-800	37	96
Series II, III	37	29

iAPX 86, 88 Target System Environment

Supporting the iAPX 86, 88-based final product configuration, the iSBC 957B software package requires:

1. iSBC 957B cable and EPROMmed monitor
2. Serial or parallel interface (8251 USART or 8255 Programmable Peripheral Interface)
3. iSBC 86, 88-based single board computer or custom iAPX 86, 88-based board

Hardware

SUPPORTED iSBC™ MICROCOMPUTERS

iSBC 86/05 Single Board Computer
iSBC 86/12A Single Board Computer
iSBC 88/25 Single Board Computer
iSBC 88/40 Single Board Computer
iSBC 337 Numeric Data Processor

SUPPORTED iSBC™ PERIPHERAL CONTROLLERS

iSBC 204 Flexible Diskette Controller
iSBC 215 Winchester Disk Controller
iSBC 220 SDM Disk Controller
iSBC 254 Bubble Memory Board

SUPPORTED iSBX™ MULTIMODULE™ BOARDS

iSBX 218 Flexible Disk Controller (when mounted on an iSBC 215 controller)
iSBX 350 Parallel I/O MULTIMODULE Board
iSBX 351 Serial I/O MULTIMODULE Board

iSBC™ 957B Package Contents (Supplied)

CABLES

1 — Serial I/O port of iSBC or iAPX 86, 88 compatible product to male RS232C connector

1 — Intellec System RS232C port to female RS232C connector

1 — Parallel load cable to mate between Intellec System UPP port and parallel I/O port on iSBC or iAPX 86, 88 compatible product

PARALLEL INTERFACE ADAPTER

1 — Parallel port status adapter for iSBC products using parallel cable

I/O DRIVER AND TERMINATORS

4 — iSBC 901 - 220 ohm/330 ohm terminator packs

4 — iSBC 902 - 1K ohm terminator packs

4 — 7437 line driver packs

INTERFACE AND EXECUTION SOFTWARE DISKETTES

1 — Single density, ISIS compatible

1 — Double density, ISIS compatible

SYSTEM MONITOR EPROMs

Microcomputer	Address		Supports NPX
	RAM	ROM	
iSBC™ 86/05, 86/12A	00000H-007FFH	FC000H-FFFFFH	Yes
iSBC™ 88/40	00000H-006FFH	FD000H-FFFFFH*	No

* Allows 2816 E^2PROM to be used at FC000H

Reference Manual (Supplied)

143979-002 — User's Guide for the iSBC 957B, iAPX 86, 88 Interface and Execution Package

ORDERING INFORMATION

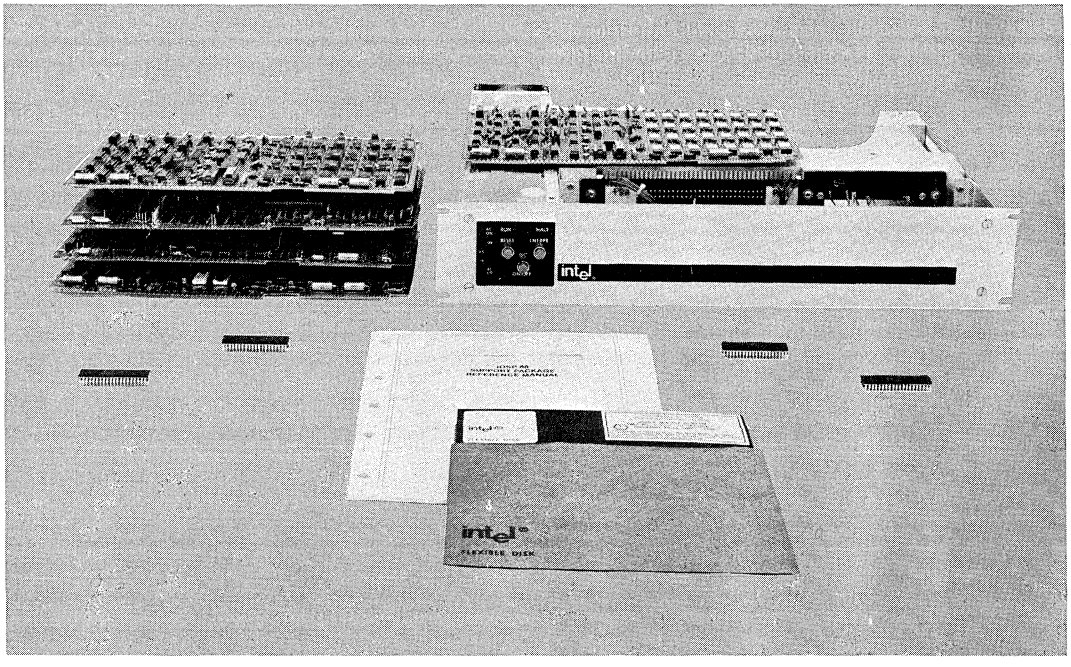
Part Number Description

SBC 957B	Intellec to iAPX 86, 88 Interface and Execution Package including software, cables and EPROMs.

iOSP™ 86
iAPX 86/30 AND iAPX 88/30 SUPPORT PACKAGE

- **Development and run-time support for iAPX 86/30 and 88/30 Operating System Processors**
- **Total iRMX™ 86 Operating System software compatibility**
- **Extendable with iRMX™ 86 Operating System calls**
- **Compatible with Intel® PL/M 86/88, PASCAL 86/88, FORTRAN 86/88, and iAPX 86/88 ASSEMBLER**
- **Supports (P)ROM or RAM based system**
- **Complete system initialization aids**
- **Complete system configuration aids**

The Intel iOSP 86 Support Package for the iAPX 86/30 and 88/30 Operating System Processors contains a comprehensive set of easy-to-use tools necessary to develop (P)ROM or RAM-based applications that use the 80130 Operating System Firmware component. All of the system initialization and run-time facilities are provided in libraries that may be configured to specific requirements, and linked to application programs written in either iAPX 86 or iAPX 88 Assembler or a high level programming language such as PASCAL 86 and PL/M 86. The iOSP 86 Package provides users with the basic initialization and interface routines needed to build application software based on the fundamental operating system functions of the iAPX 86/30 and 88/30 Operating System Processors. The iOSP 86 Package also enables users to add higher level I/O functions from the fully compatible iRMX 86 Operating System, or to form custom, real-time systems.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE, iOSP and iCS. 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 CORPORATION, 1981

October, 1981
Order Number: 210236-001

FUNCTIONAL DESCRIPTION

The iAPX 86/30 and iAPX 88/30 Operating System Processors (OSPs) provide an easy-to-use foundation on which many real-time applications may be built. They provide the functions and system support needed to implement both simple and complex applications that require multiple tasks to run concurrently (see Figure 1). These services are made possible by the addition of the five new data types integrated into the 80130 Operating System Firmware (OSF) component. The 80130 OSF extends the basic data types of the CPU (integer, byte, character, etc.) by adding new system data types (JOB's, TASK's, MAILBOX's, SEGMENT's, and REGION's), and extensive timer, interrupt, memory, and error management designed to give real-time response to multitasking and multiprogramming applications. As shown in the second half of the figure, other operating system functions such as mass storage I/O services and an easy-to-use Human Interface can be added easily, by using modules from the complete operating system services of the iRMX 86 Operating System. The iOSP 86 Support Package provides both an interface between application software and the Operating System Processors, and development tools designed to make the implementation and initialization of real-time, multitasking systems much simpler.

The iOSP 86 Support Package provides system developers with the configuration options necessary to tailor the iAPX 86/30 and 88/30 Operating System Processors to custom applications. Using the Linking and Locating facilities of either an Intel Intellec Development System, or a suitably equipped iRMX 86 system, the interface libraries provided in the package can be added to application software modules to form easy-to-use initialization routines. They also form a simple interface between application software and the operating system primitives of the 80130 OSF component. The various configuration options include:

Memory and I/O Addressing

The 80130 OSF requires a 16K byte block of memory address space to be reserved for accessing internal functions. The iOSP 86 Support Package is used to specify the base address of the 80130 and the beginning of the initialization routines.

All Interrupt and Timer management of the OSF is controlled via a reserved 16-byte I/O address block that may be selected by the user. In addition, from 1 to 7 slave 8259A interrupt controllers can be specified in order to provide the system with up to 57 priority interrupt sources. The OSF baud rate generator may also be configured to support an optional terminal interface.

Extending the 80130 OSF

The 80130 OSF allows users to add their own operating system extensions. These extensions may take advantage of the detailed and efficient intertask communication and synchronization primitives already provided by the 80130, and/or may utilize custom functions tailored to specific applications. The Support Package also enables users to extend the OSF with the extensive services of Intel's iRMX 86 Operating System, thereby allowing applications to grow without having to change or alter application software already written, or having to write other operating system software. Use of the 80130 with the iRMX 86 Operating system greatly reduces the amount of memory needed for the iRMX 86 Nucleus layer, and enables applications to take advantage of the increased

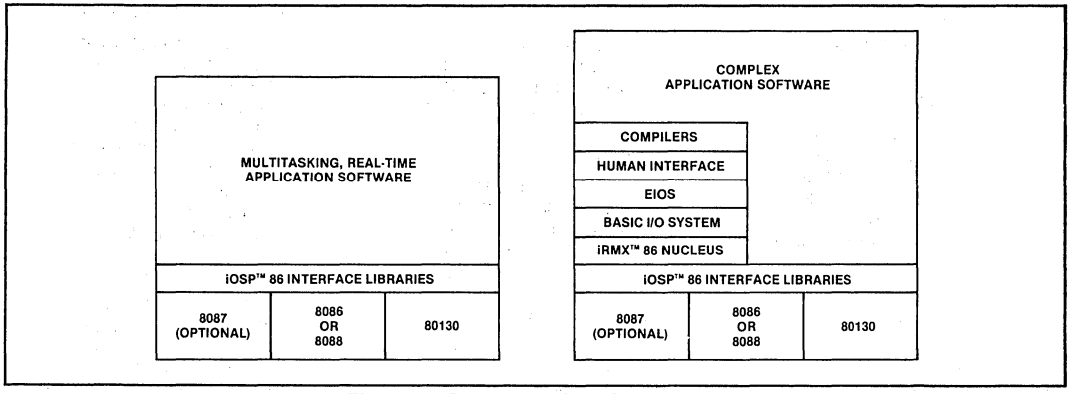


Figure 1. Structure of Typical Systems

performance and reduced size requirements inherent in the iAPX 86/30 and 88/30 VLSI Operating System Processors. As each of the services provided by the 80130 is completely iRMX 86-compatible, applications have an automatic upward path to support complete file systems and multiple processor environments.

Application Interfaces

Two interface libraries are included in the iOSP 86 Support Package. The first allows programmers to write application software modules in the Compact Model of computation supported by Intel's compilers. The second provides an interface to program segments written in either the Medium or Large Models.

The interface libraries provide the means of accessing all of the primitives supported by the Operating System Processors. With this interface, and all the memory management primitives of the OSPs, applications have full access to 1M byte of memory, and all of the addressing modes of the CPU.

The iAPX 86/30 and 88/30 OSPs allow applications to take full advantage of the Compact, Medium, and Large models of computation afforded by the segment model of the CPU's.

These libraries are fully compatible with object modules produced by the MACRO 86/88 Assembler, and the PASCAL 86/88 and FORTRAN 86/88 and PL/M 86/88 Compilers.

Application Initialization

The iOSP 86 Support Package provides for the configuration of the system Root JOB, and all user application JOB's that require initialization when the system is started. The user may also specify the configuration of the interrupt system (including slave 8259A interrupt controllers) and the clock rate used for system timing. These choices are automatically programmed into the various devices when the system is initialized.

Operating System Calls

The 80130 OSF performs a total of 35 operating system primitives all of which are completely compatible with the equivalent iRMX 86 Operating System calls. The iOSP 86 Support Package provides user-level interfaces to these primitives to enable applications to create, delete, control, and exchange the new data types provided by the 80130 OSF. In general, these interfaces allow application software to manage all of the resources of an iAPX 86/30 or 88/30 OSP (and an optional 8087 Numeric Processor Extension) system via any of the 35 normal Assembly Language system calls shown in Figure 2.

Required Development Hardware

Use of the iOSP 86 Support Package requires an Intel MDS Development System supporting either single or double density diskettes, or any iRMX 86 system supporting a standard floppy diskette drive and the iRMX 860 Assembler, Linker, and Locator Package. Use of the 80130 requires only a minimal system including either the iAPX 86/30 or 88/30 Operating System Processor, and enough system memory to contain the application programs and approximately 2K bytes of initialization and interface software provided in the iOSP 86 Support Package.

```
JOB GROUP
    CALL RQ$CREATE$JOB

TASK GROUP
    CALL RQ$CREATE$TASK
    CALL RQ$DELETE$TASK
    CALL RQ$SUSPEND$TASK
    CALL RQ$RESUME$TASK
    CALL RQ$SLEEP
    CALL RQ$GET$TASK$TOKENS
    CALL RQ$SET$PRIORITY

SEGMENT GROUP
    CALL RQ$CREATE$SEGMENT
    CALL RQ$DELETE$SEGMENT

MAILBOX GROUP
    CALL RQ$CREATE$MAILBOX
    CALL RQ$DELETE$MAILBOX
    CALL RQ$SEND$MESSAGE
    CALL RQ$RECEIVE$MESSAGE

REGION GROUP
    CALL RQ$CREATE$REGION
    CALL RQ$DELETE$REGION
    CALL RQ$SEND$CONTROL
    CALL RQ$RECEIVE$CONTROL
    CALL RQ$ACCEPT$CONTROL

DELETION CONTROL GROUP
    CALL RQ$DISABLE$DELETION
    CALL RQ$ENABLE$DELETION

INTERRUPT MANAGEMENT GROUP
    CALL RQ$SET$OS$EXTENSION
    CALL RQ$SET$INTERRUPT
    CALL RQ$ENTER$INTERRUPT
    CALL RQ$EXIT$INTERRUPT
    CALL RQ$WAIT$INTERRUPT
    CALL RQ$SIGNAL$INTERRUPT
    CALL RQ$RESET$INTERRUPT
    CALL RQ$ENABLE
    CALL RQ$DISABLE
    CALL RQ$GET$LEVEL

ERROR CONTROL GROUP
    CALL RQ$SET$EXCEPTION
    CALL RQ$SIGNAL$EXCEPTION
    CALL RQ$GET$EXCEPTION
    CALL RQ$GET$TYPE
```

Figure 2. Operating System Primitives

ORDERING INFORMATION

Each of the ordering options listed below include all the necessary initialization and interface procedures needed to use the iAPX 86/30 and iAPX 88/30 Operating System Processors. Purchase of the iOSP 86 Package requires verification of an Intel Master Software License. Each package also includes an iOSP 86 User's Manual (Document Number 143331), and a one-year update service.

Part Number	Description
OSP 86 A	iOSP 86 Support Package contained on an ISIS-II compatible, single density diskette.
OSP 86 B	iOSP 86 Support Package contained on an ISIS-II compatible, double density diskette.
OSP 86 E	iOSP 86 Support Package contained on an iRMX 86 format, double density diskette.

Peripheral Controllers

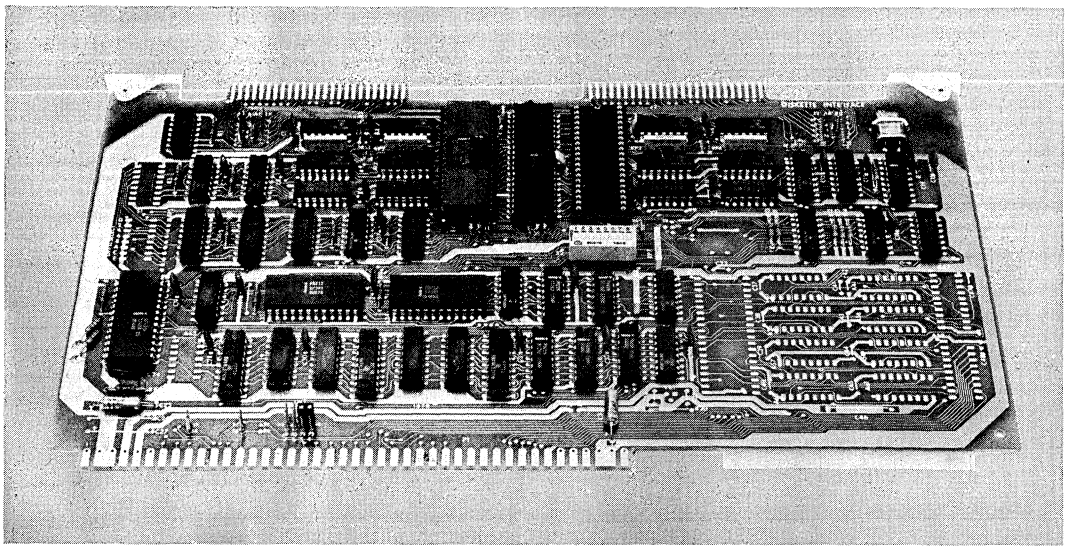
6

intel®

iSBC 204
SINGLE DENSITY FLEXIBLE DISKETTE CONTROLLER

- **Full compatibility with iSBC 80, iSBC 86, and iSBC 88 Single Board Computers**
- **Direct compatibility with most single-density, soft-sectored standard- (8″) and mini-size (5¼″) flexible diskette drives**
- **Software supported by iRMX 80, iRMX 86 and iRMX 88 Real-Time Multi-tasking Executive disk file system**
- **DMA input/output allows single board computers to process in parallel with diskette transfer operations**
- **Programmable track-to-track access, head-settling, and head-load times**
- **On-board data separation logic**
- **Read, write, verify, and search on single or multiple sectors**
- **Single +5V supply**

The Intel iSBC 204 Single Density Flexible Diskette Controller is a single board universal diskette controller capable of supporting virtually any software-sectored, single density diskette drive. The standard iSBC 204 Controller can control two drive surfaces (two single-sided drives or one double-sided drive). With the addition of a second (optional) Intel 8271 component, up to four drives can be supported. In addition to the standard IBM 3740 formats, the controller supports sector lengths of up to 4096 bytes plus mini-size drive formats. The iSBC 204's wide range of drive compatibility is achieved without compromising performance. The operating characteristics (track-to-track access, head-load, and head-settling times) are specified under user program control. The controller can read, write, verify, and search either single or multiple sectors.



FUNCTIONAL DESCRIPTION

Intel's 8271 Floppy Disk Controller (FDC) circuit is the heart of the iSBC 204 Controller. On-board data separation logic performs standard FM encoding and decoding, obviating external separation circuitry at the drive. Diskette data transfers are DMA (direct memory access) through an on-board Intel 8257 DMA controller circuit which manages DMA transfers and signals the master iSBC processor on completion of the transfer. A block diagram of the iSBC 204 Controller is shown in Figure 1.

Universal Drive and MULTIBUS Compatibility

Because the iSBC 204 Controller has universal drive compatibility, it can be used to control virtually any standard- or mini-sized single density diskette drive. Moreover, the iSBC 204 Controller fully supports the microcomputer industry standard MULTIBUS system bus and can be used with any single board computer or system compatible with Intel's bus. Because the iSBC 204 Controller is programmable, its performance is not compromised by its universal drive compatibility. The track-to-track access, head-load, and head-settling characteristics of the selected drive model are program specified. Data may be organized in a fully compatible IBM 3740 sector format, in sectors up to 4096 bytes in length, or in formats compatible with the mini-sized diskette drives.

Interface Characteristics

Expandability — Each standard iSBC 204 Controller includes a single 8271 FDC circuit capable of supporting two drive surfaces. Optionally the iSBC 204 may be expanded to support four single-sided (or two double-sided) drives with the insertion of a second 8271 component into an on-board socket.

Simplified Interface — The cables between the iSBC 204 Controller and the drive(s) may be either low cost, flat ribbon cable with mass termination connectors or twisted pair conductors with individually wired connectors. An on-board, cross-connect matrix allows optional drive control and status signals to be connected while maintaining pin-to-pin compatibility.

Programming

The powerful 8271 FDC circuit is capable of executing high-level commands that simplify system software development. The device can read, write, and verify both single and multiple sectors. CRC characters are generated and checked automatically. Up to two tracks on each surface may be designated "bad" and logically removed from the diskette.

Sector Scanning — Scan commands permit sectors to be searched for a specified data pattern or "key". During scan operations the pattern image from memory is continuously compared with a sector or multiple sectors

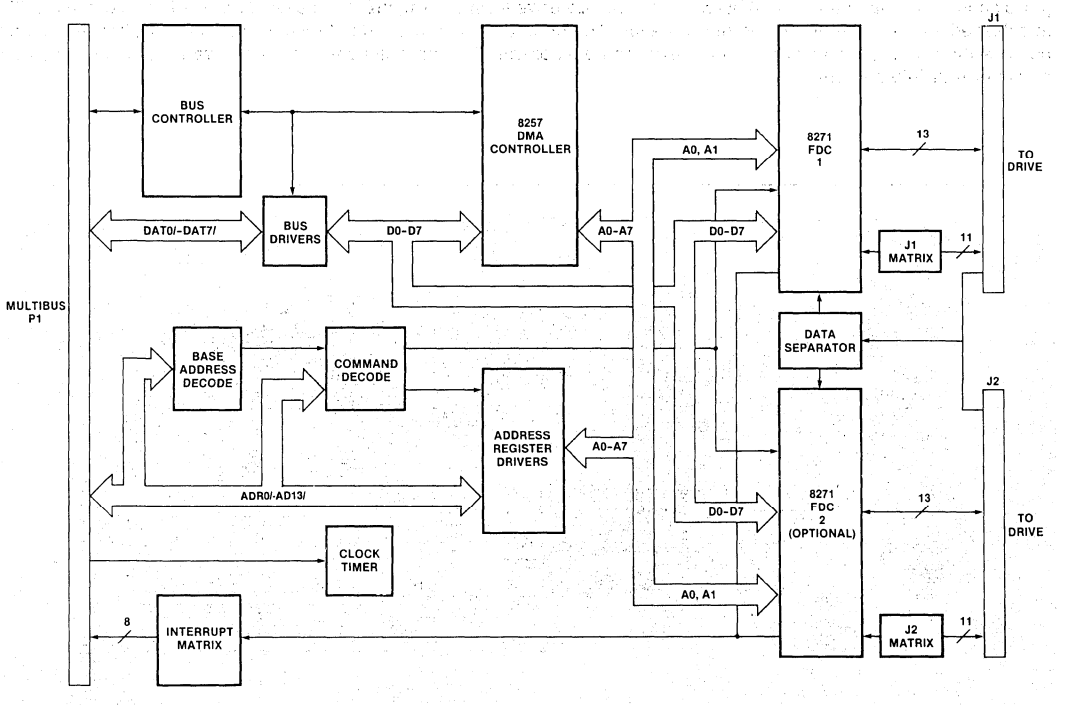


Figure 1. iSBC 204 Single Density Diskette Controller Block Diagram

read from the diskette. No CPU intervention is required until a match is found or all specified sectors have been searched.

Program Initiation — All diskette operations are initiated by standard input/output (I/O) port operations through an iSBC single board computer. System software first initializes the controller with the operating characteristics of the selected drive. The diskette is then formatted under program control. For subsequent transfers, the starting memory address and transfer mode are specified for the DMA controller. Data transfers occur in response to commands output by the CPU.

Data Transfer — Once a diskette transfer operation has been initiated, the controller acts as a bus master and transfers data over the MULTIBUS at high speed. No CPU intervention is required until the transfer is complete as indicated either by the generation of an interrupt on the bus or by examination of a "done" bit by the CPU.

SPECIFICATIONS

Compatibility

CPU — Any iSBC MULTIBUS computer or system mainframe.

Drive — Single density, standard- (8″) and mini-sized (5¼″) diskette drives. The standard iSBC 204 Controller supports two single-sided drives or one double-sided drive. By adding an (optional) 8271 FDC, four single-sided or two double-sided drives may be supported. The following drives are known to be compatible:

Standard Size	Mini Size
CDC 9404	PERTEC FD200
GSI 110	SHUGART SA400
MEMOREX 550	WANGCO 82
MEMOREX 552 (dual-sided)	
SHUGART 800	
SHUGART 850 (dual-sided)	
WANGCO 76S	
PERTEC 650 (SD/DD, DBL. Head)	

Diskette — Unformatted IBM Diskette 1 (or equivalent single-sided); unformatted IBM Diskette 2 (or equivalent double-sided); unformatted Shugart SA104 Diskette (or equivalent mini).

Data Organization and Capacity

(Standard Size Drives)

	IBM Format			Non-IBM Format		
Bytes per sector	128	256	512	1024	2048	4096
Sectors per track	26	15	8	4	2	1
Tracks per diskette	77			Up to 255		
Bytes per diskette (77 tracks)	256,256 (128-byte sector) 295,680 (256-byte sector) 315,392 (512-byte sector)			315,392		

Drive Characteristics

	Standard Size	Mini Size
Transfer rate (KB/sec)	250	125
Disk speed (RPM)	360	300
Track-to-track access (programmable)	1 to 255 ms in 1 ms steps	2 to 510 ms in 2 ms steps
Head settling time (programmable)	0 to 255 ms in 1 ms steps	0 to 510 ma in 2 ms steps
Head load time (programmable)	0 to 60 ms in 4 ms steps	0 to 120 ms in 8 ms steps

Equipment Supplied

iSBC 204 Controller

Reference Schematic

Controller-to-drive cabling and connectors are not supplied with the iSBC 204 Controller. Cables can be fabricated easily using either flat ribbon cable or twisted pair conductors with commercially available connectors as described in the iSBC 204 Hardware Reference Manual.

Optional Equipment

8271 Flexible Diskette Controller Component — Adding a second 8271 device to the fully tested circuit on the iSBC 204 Controller allows four drive surfaces to be supported.

Physical Characteristics

Width — 6.75 in. (17.15 cm)
Height — 0.5 in. (1.27 cm)
Length — 12.0 in. (30.48 cm)
Shipping Weight — 1.75 lb (0.80 kg)
Mounting — Occupies one slot of iSBC system chassis or iSBC 604/614 cardcage.

Electrical Characteristics

Power Requirements — 5.0V (±5%), 2.5A max

Environmental Characteristics

Temperature — 0°C to 55°C (operating); −55°C to +85°C (non-operating)

Humidity — Up to 90% relative humidity without condensation (operating); all conditions without condensation or frost (non-operating)

Reference Manuals

9800568 — iSBC 204 Diskette Controller Hardware Reference Manual (NOT SUPPLIED).

9800522 — RMX/80 User's Guide (NOT SUPPLIED).

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

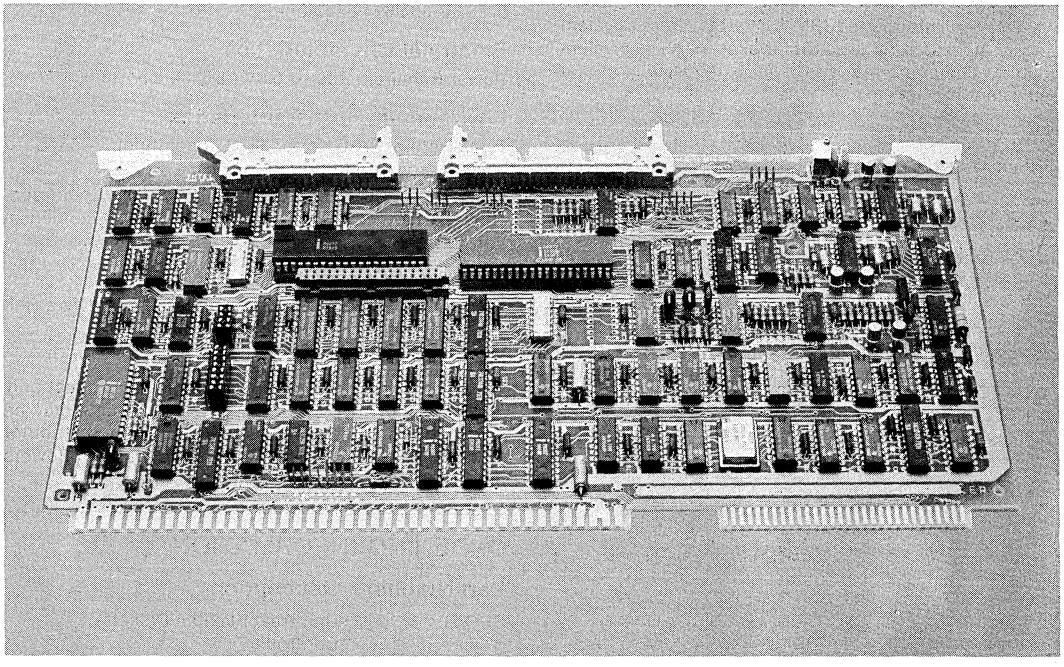
ORDERING INFORMATION

Part Number	Description
SBC 204	Universal Flexible Diskette Controller

iSBC 208
FLEXIBLE DISK CONTROLLER

- **Compatible with all iSBC 80, iSBC 86, and iSBC 88 Single Board Computers**
- **Controls most single and double density diskette drives**
- **On-board iSBX bus for additional functions**
- **User-programmable drive parameters allow wide choice of drives**
- **Phase lock loop data separator assures maximum data integrity**
- **Read and write on single or multiple sectors**
- **Single +5V Supply**
- **Capable of addressing 16M bytes of system memory**

The Intel iSBC 208 Flexible Disk Controller is a diskette controller capable of supporting virtually any soft-sectored, double density or single density diskette drive. The standard controller can control up to four drives with up to eight surfaces. In addition to the standard IBM 3740 formats and IBM System 34 formats, the controller supports sector lengths of up to 8192 bytes. The iSBC 208 board's wide range of drive compatibility is achieved without compromising performance. The operating characteristics are specified under user program control. The controller can read, write, verify, and search either single or multiple sectors. Additional capability such as parallel or serial I/O or special math functions can be placed on the iSBC 208 board by utilizing the iSBX bus connection.



FUNCTIONAL DESCRIPTION

Intel's 8272 Floppy Disk Controller (FDC) circuit is the heart of the iSBC 208 Controller. On-board data separation logic performs standard MFM (double density) and FM (single density) encoding and decoding, eliminating the need for external separation circuitry at the drive. Data transfers between the controller and memory are managed by a DMA device which completely controls transfers over the MULTIBUS system bus. A block diagram of the iSBC 208 Controller is shown in Figure 1.

Universal Drives and the iSBC 208 Controller

Because the iSBC 208 Controller has universal drive compatibility, it can be used to control virtually any standard- or mini-sized diskette drive. Moreover, the iSBC 208 Controller fully supports the iSBX bus and can be used with any iSBX module compatible with this bus. Because the iSBC 208 Controller is programmable, its performance is not compromised by its universal drive compatibility. The track-to-track access, head-load, and head-unload characteristics of the selected drive model are program specified. Data may be organized in sectors up to 8192 bytes in length.

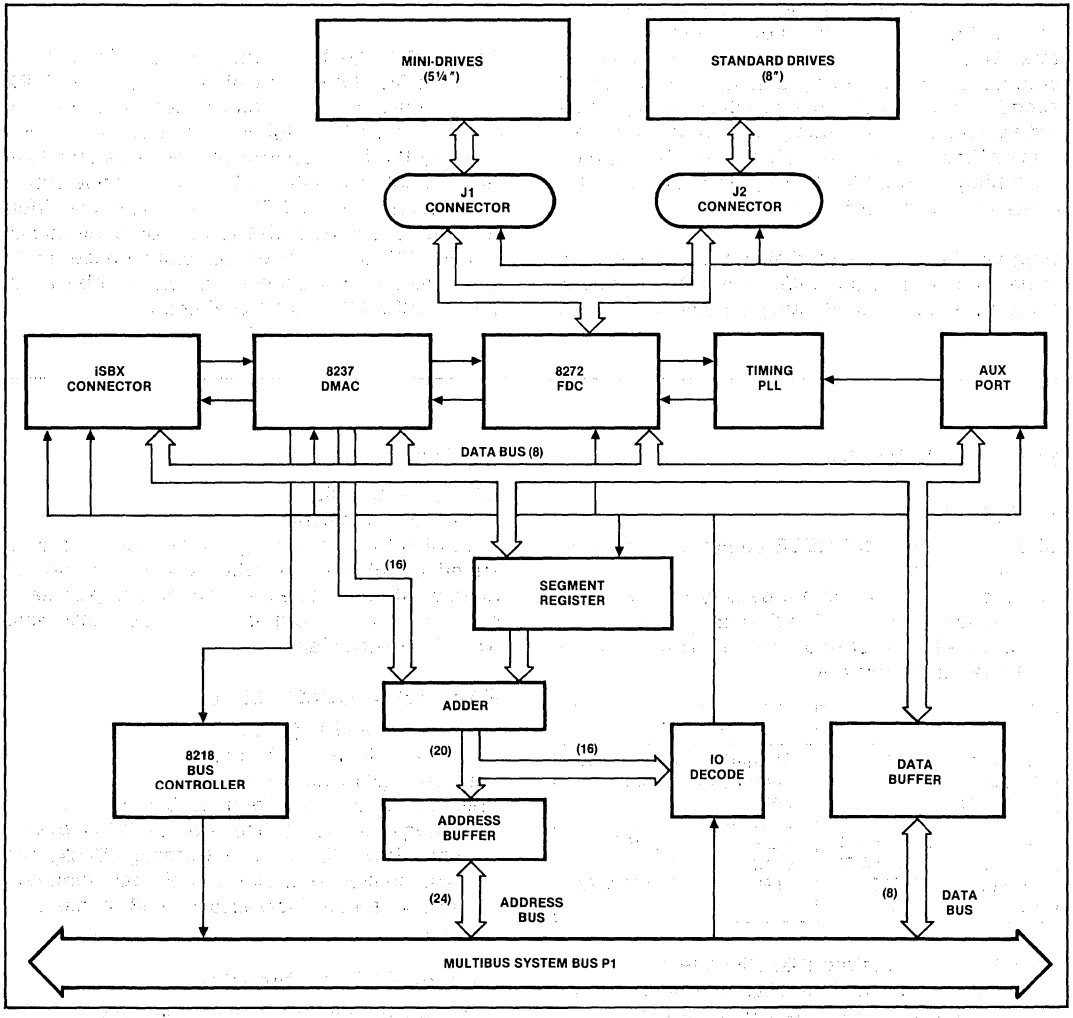


Figure 1. iSBC 208 Flexible Disk Controller Block Diagram

AFN-01629A

Interface Characteristics

The standard iSBC 208 Controller includes an Intel 8272 Floppy Disk Controller chip which supports up to four drives, single or double sided.

SIMPLIFIED INTERFACE—The cables between the iSBC 208 Controller and the drive(s) may be low cost, flat ribbon cable with mass termination connectors. The mechanical interface to the board is a right-angle header with locking tabs for security of connection.

PROGRAMMING — The powerful 8272 FDC circuit is capable of executing high-level commands that simplify system software development. The device can read and write both single and multiple sectors. CRC characters are generated and checked automatically. Recording density is selected at each Read and Write to support the industry standard technique of recording basic media information on Track 0 of Side 0 in single density, and then switching to double density (if necessary) for operations on other tracks.

Program Initiation—All diskette operations are initiated by standard input/output (I/O) port operations through an iSBC single board computer. System software first initializes the controller with the operating characteristics of the selected drive. The diskette is then formatted under program control. For subsequent transfers, the starting memory address and transfer mode are specified for the DMA controller. Data transfers occur in response to commands output by the CPU.

Data Transfer—Once a diskette transfer operation has been initiated, the controller acts as a bus master and transfers data over the MULTIBUS at high speed. No CPU intervention is required until the transfer is complete as indicated either by the generation of an interrupt on the bus or by examination of a "done" bit by the CPU.

iSBX BUS SUPPORT — One connector is available on the iSBC 208 board which supports the iSBX system bus. This connector supports single-byte transfer as well as higher-speed transfers supervised by the DMA controller. Transfers may take place in polled or interrupt modes, user-selected. The presence of the iSBX bus allows many different functions to be added to the board. Serial I/O, parallel I/O and various special-purpose math functions are only a few of the capabilities available on iSBX MULTIMODULE boards.

SPECIFICATIONS

Compatibility

CPU—Any iSBC MULTIBUS computer or system main frame

Devices—Double or single density standard (8″) and mini (5¼″) flexible disk drives. The drives may be single or double sided. Drives known to be compatible are:

Standard (8″)		Mini (5¼″)	
Caldisk	143M	Shugart	450 SA 400
Remex	RFD 4000	Micropolis	1015-IV
Memorex	550	Pertec	250
MFE	700	Siemens	200-5
Siemens	FDD 200-8	Tandon	TM-100
Shugart	SA 850/800	CDC	9409
Pertec	FD 650	MPI	51/52/91/92
CDC	9406-3		

Diskette—Unformatted IBM Diskette 1 (or equivalent single-sided media); unformatted IBM Diskette 2D (or equivalent double-sided).

Equipment Supplied

iSBC 208 Controller
Reference Schematic
Controller-to-drive cabling and connectors are not supplied with the controller. Cables can be fabricated with flat cable and commercially-available connectors as described in the iSBC 208 Hardware Reference Manual

Physical Characteristics

Width—6.75 inches (17.15 cm)
Height—0.5 inches (1.27 cm)
Length—12.0 inches (30.48 cm)
Shipping Weight—1.75 pounds (0.80 Kg)
Mounting—Occupies one slot of iSBC system chassis or iSBC 604/614 Cardcage/Backplane. With an iSBX MULTIMODULE board mounted, vertical height increases to 1.13 inches (2.87 cm).

Electrical Characteristics

Power Requirements— +5 VDC @ 3.0A

Data Organization and Capacity

Standard Size Drives

	Double Density						Single Density					
	IBM System 34			Non-IBM			IBM System 3740			Non-IBM		
Bytes per Sector	256	512	1024	2048	4096	8192	128	256	512	1024	2048	4096
Sectors per Track	26	15	8	4	2	1	26	15	8	4	2	1
Tracks per Diskette	77			256			77			256		
Bytes per Diskette (Formatted, per diskette surface)	512,512 (256 bytes/sector) 591,360 (512 bytes/sector) 630,784 (1024 bytes/sector)			630,784			256,256 (128 byte/sector) 295,680 (256 bytes/sector) 315,392 (512 bytes/sector)			315,392		

Drive Characteristics

	Standard Size	Mini Size
	Double/Single Density	Double/Single Density
Transfer Rate (K bytes/sec)	62.5/31.25	31.25/15.63
Disk Speed (RPM)	360	300
Step Rate Time (Programmable)	1 to 16 msec/track in 1 msec increments	2 to 32 msec/track in 2 msec increments
Head Load Time (Programmable)	2 to 254 msec in 2 msec increments	4 to 508 msec in 4 msec increments
Head Unload Time (Programmable)	16 to 240 msec in 16 msec increments	32 to 480 msec in 32 msec increments

Environmental Characteristics

Temperature—0°C to 55°C (operating); −55°C to +85°C (non-operating)

Humidity—Up to 90% Relative Humidity without condensation (operating); all conditions without condensation or frost (non-operating)

Reference Manual

143078-001—iSBC 208 Flexible Disk Controller Hardware Reference Manual (NOT SUPPLIED). Reference manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051.

ORDERING INFORMATION

Part Number	Description
SBC 208	Flexible Disk Controller

iSBC™ 215A/iSBC™ 215B/iSBC™ 215C WINCHESTER DISK CONTROLLER

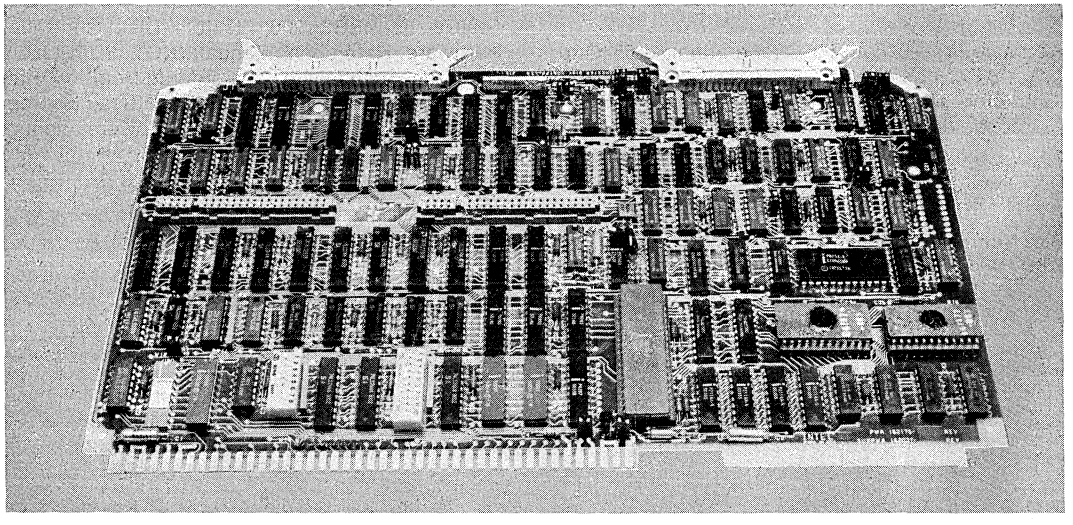
- **Controls up to four 5¼", 8" or 14" Winchester disk drives**
- **Over 100 MB of storage per controller**
- **Two iSBX™ connectors on-board**
- **Removable back-up storage available through the iSBX™ 218 Flexible Disk Controller**
- **Provides ANSI 1226 standard interface**
- **Software drivers available for iRMX™ 86 and iRMX™ 88 operating systems**
- **Intel® 8089 I/O Processor provides DMA channels plus user-programmable intelligence**
- **On-board diagnostics and ECC**
- **Full sector buffering on-board**
- **Capable of addressing 1 MB of system memory**

The iSBC 215 Winchester Disk Controller will enhance the mass storage capabilities of any iSBC 80, iSBC 88, or iSBC 86-based MULTIBUS system. The controller will interface to industry standard Winchester disk drives currently available in formatted capacity from 4.5 to 26.7 MB. Recording densities are expected to increase rapidly in the near future and the iSBC 215 controller has been designed to accommodate these increases.

The iSBC 215 board will control up to four 5¼", 8" or 14" drives. The iSBC 215A board controls open-loop drives; iSBC 215B board controls closed-loop drives; iSBC 215C board controls ANSI standard 1226 drives.

Two iSBX connectors are provided on the board to interface with the iSBX 218 Flexible Disk Controller, providing up to 4 MB of removable storage.

Increased computing power made available in the iSBC board products has led to a requirement for larger, more reliable mass storage subsystems. The Winchester disk controller provides a high capacity, low cost disk solution that is well matched to single board computer applications.



FUNCTIONAL DESCRIPTION

Programming

Programming the iSBC 215 controller is simplified by the use of memory-based parameter blocks. A linked list technique is used, allowing the user to perform multiple disk operations.

Full On-Board Buffer

The iSBC 215 controller contains enough on-board RAM for buffering one full data sector. The controller is designed to make use of this buffer in all transfers. The on-board sector buffer prevents data overrun errors and allows the iSBC 215 Winchester Disk Controller to occupy any priority slot on the MULTIBUS.

ECC

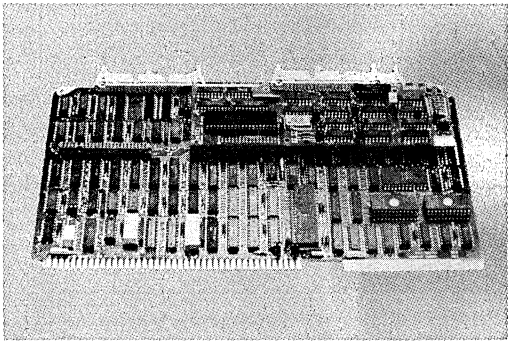
High data integrity is provided by on-board Error Checking Code (ECC) logic. When writing sector ID or data fields, a 32-bit Fire code, for burst error correction, is appended to the field by the controller. During a Read operation, the same logic regenerates the ECC polynomial and compares this second polynomial to the appended ECC. The ECC logic can detect an erroneous data burst up to 32 bits in length and using an 8089 algorithm can correct an erroneous burst up to 11 bits in length.

iSBX Interface

The software interface and data buffering capabilities used for Winchester drives are also available for both iSBX MULTIMODULE interfaces. Software developed for the iSBC 215 controller can also be used to transfer data to and from an iSBX-compatible I/O device.

Expanded I/O Capability

The iSBC 215 controller allows the user to execute user-written 8089 programs located in on-board or MULTIBUS system RAM. Thus the full capability of the 8089 I/O processor can be utilized for customer I/O requirements.



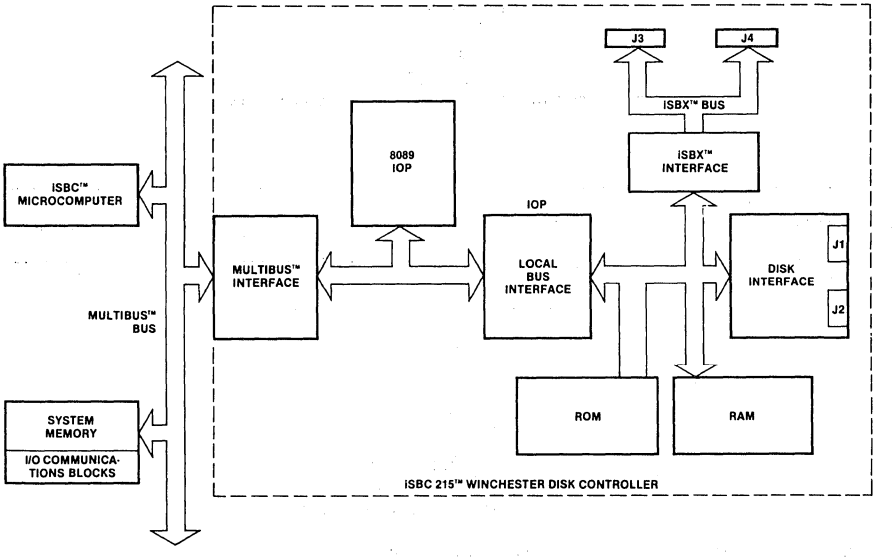
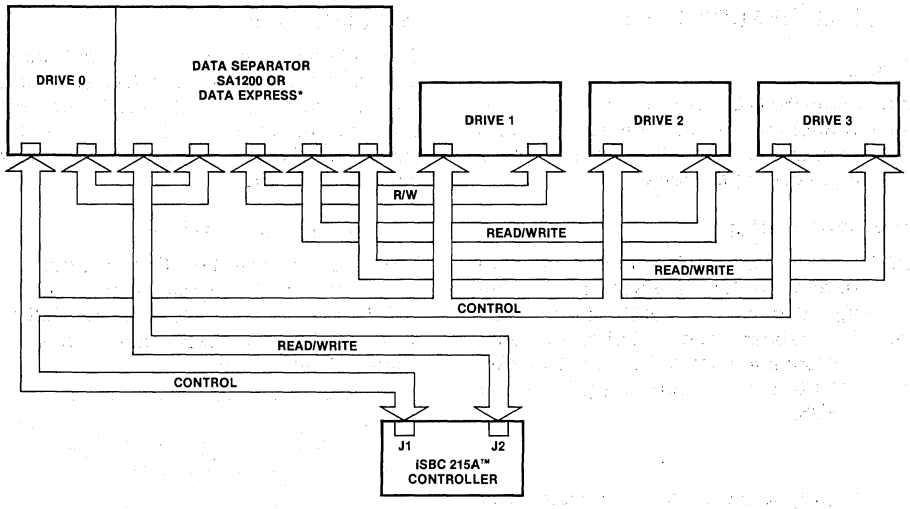
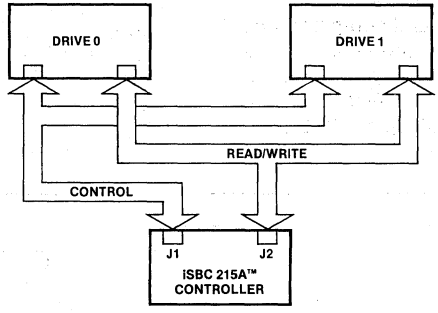


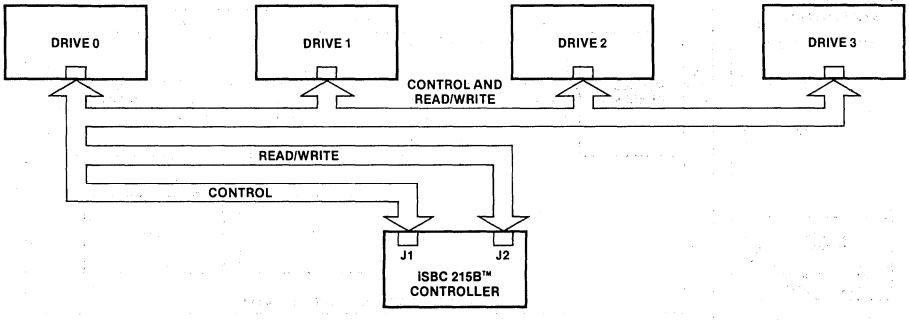
Figure 1. Block Diagram of iSBC 215™ Winchester Disk Controller



INTERFACE WITH SHUGART 8"/QUANTUM/RMS DRIVES



INTERFACE WITH MEMOREX/SHUGART 14"/CDC/FUJITSU DRIVES



INTERFACE WITH PERTEC AND PRIAM DRIVES

* Data Express is a trademark of Rotating Memory Systems Inc.

Figure 2. Controller to Drive Interfacing

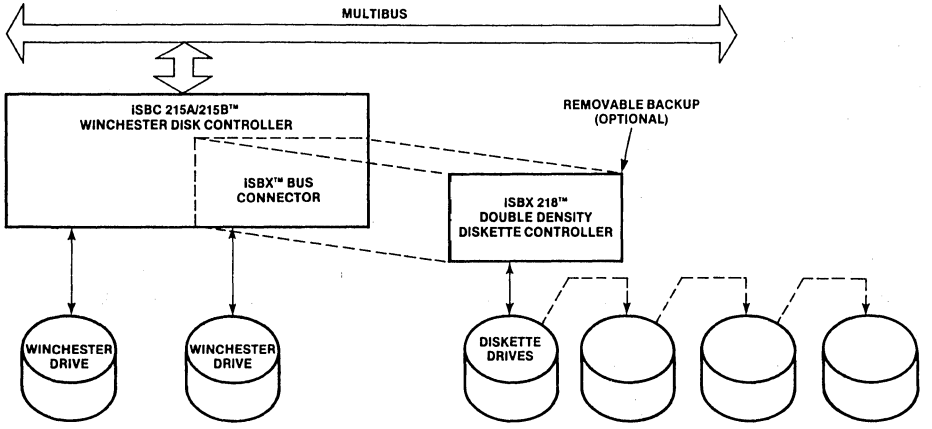


Figure 3. System Configuration (with Optional Diskette Backup)

SPECIFICATIONS

Compatibility

CPU — Any iSBC MULTIBUS computer or system mainframe

Disk Drives — Winchester Disk Drives; both open-loop and closed-loop head positioner types. The following drives are known to be compatible:

Open-Loop (iSBC 215A)
Shugart SA 1000 Series Shugart SA 4000 Series Memorex 100 Series Quantum Q2000 Series Fujitsu 2301, 2302 CDC 9410 RMS 5¼″ Series
Closed-Loop (iSBC 215B)
Pertec D8000 Series Priam 8″ and 14″ Drive Series
ANSI (iSBC 215C)
3M 8430 Series BASF 6170 Series IMI 7700 Series Kennedy 7300 Series Pertec D800 Series Priam 8″ Series SLI Cheyenne
iSBX MULTIMODULE Boards
iSBX 218 Flexible Disk Controller iSBX 350 Parallel I/O iSBX 351 Serial I/O iSBX 311 Analog Input iSBX 328 Analog Output

Equipment Supplied

iSBC 215 Winchester Disk Controller Reference Schematic

Controller-to-drive cabling and connectors are not supplied with the controller. Cables can be fabricated with flat cable and commercially-available connectors as described in the iSBC 215 Hardware Reference Manual.

Physical Characteristics

Width — 6.75 in. (17.15 cm)

Height — 0.5 in. (1.27 cm)

Length — 12.0 in. (30.48 cm)

Shipping Weight — 19 oz (54 kg)

Mounting — Occupies one slot of iSBC system chassis or cardcage/backplane

With an iSBX MULTIMODULE board mounted, vertical height increases to 1.13 in. (2.87 cm).

Electrical Characteristics

Power Requirements

+5 VDC @ 3.25A max
−5 VDC @ 0.15A max[1]
+12 VDC @ 0.15A max[2]
−12 VDC @ 0.03A max[2]

Notes:
1. On-board regulator and jumper allows +12 VDC usage from MULTIBUS.
2. Required for some iSBX MULTIMODULE boards.

Drives per Controller

5¼″ Winchester Disk Drives — Up to four RMS drives.

8″ Winchester Disk Drives — Up to four ANSI, Shugart, Pertec, Quantum or Priam drives; up to two Memorex, CDC, or Fujitsu drives.

14″ Winchester Disk Drives — Up to four Priam drivers; up to two Shugart drives.

Flexible Disk Drives — Up to four drives through the optional iSBX 218 Flexible Disk Controller connected to the iSBC 215 board's iSBX connector.

Environmental Characteristics

Temperature — 0° to 55°C (operating); −55°C to +85°C (non-operating)

Humidity — Up to 90% relative humidity without condensation (operating); all conditions without condensation or frost (non-operating)

Reference Manual

121593-002 — iSBC 215 Winchester Disk Controller Hardware Reference Manual (NOT SUPPLIED)

Reference manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051

Data Organization and Capacity

Sectors/Track[1]

Bytes/Sector	Priam 8″	Priam 14″	RMS/Shugart/Quantum	Memorex	Pertec
128	70	104	54	64	69
256	42	62	31	38	42
512	23	34	17	21	24
1024	12	18	9	11	12

Note 1. Maximum allowable for corresponding selection of bytes per sector.

Formatted Capacity/Drive[2]

Bytes/Sector	Shugart	Quantum	Pertec	Priam	Memorex	RMS
128	7.08 MB	7.08 MB	12.35 MB	23.29 MB	7.99 MB	8.40 MB
256	8.12	8.12	15.03	27.94	9.49	9.65
512	8.91	8.91	17.17	30.62	10.49	10.58
1024	9.43	9.43	17.18	31.95	10.98	11.21

Note 2. Shugart SA 1004, Quantum Q2010, Priam 3450, Pertec D8000, Memorex 101, RMS 512

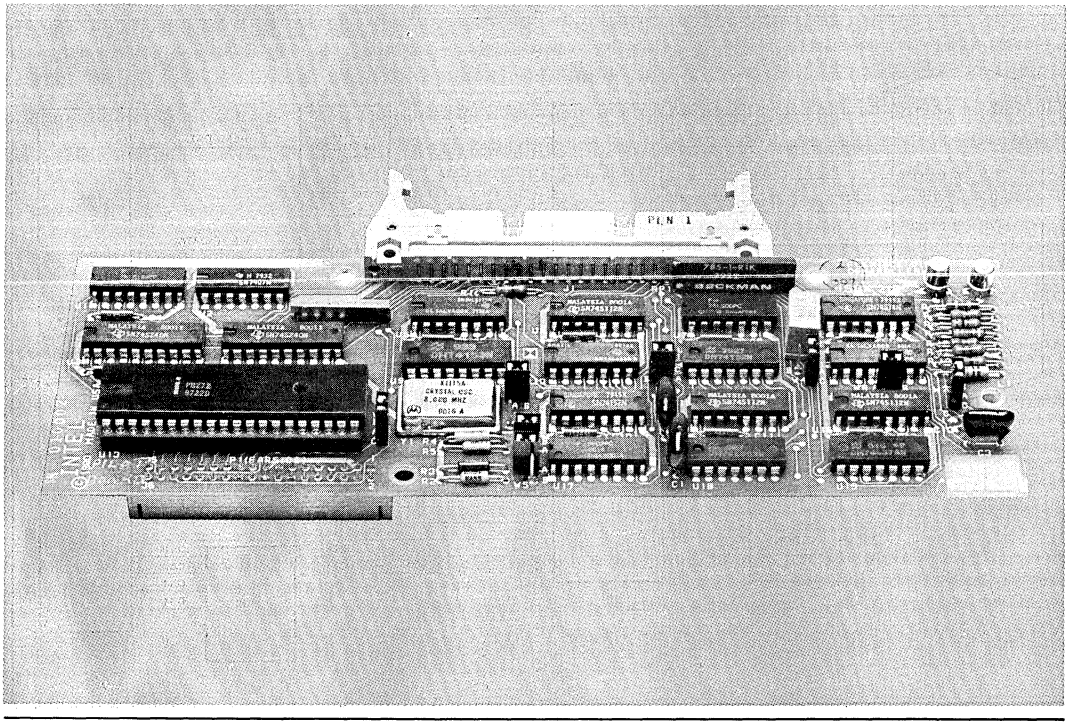
ORDERING INFORMATION

Part Number	Description
SBC 215A	Winchester Disk Controller (open-loop)
SBC 215B	Winchester Disk Controller (closed-loop)
SBC 215C	Winchester Disk Controller (ANSI interface)

iSBX 218
FLEXIBLE DISK CONTROLLER

- **iSBX MULTIMODULE controller provides flexibility at low cost**
- **Controls most single and double density diskette drives**
- **User-programmable drive parameters allow wide choice of drives**
- **Phase lock loop data separator assures maximum data integrity**
- **Read and write on single or multiple sectors**
- **Single +5V supply**

The Intel iSBX 218 Flexible Disk Controller is a double-wide iSBX board diskette controller capable of supporting virtually any soft-sectored, double density or single density diskette drive. The standard controller can control up to four drives with up to eight surfaces. In addition to the standard IBM 3740 formats and IBM System 34 formats, the controller supports sector lengths of up to 8192 bytes. The iSBX 218 board's wide range of drive compatibility is achieved without compromising performance. The operating characteristics are specified under user program control. The controller can read, write, verify, and search either single or multiple sectors.



FUNCTIONAL DESCRIPTION

Intel's 8272 Floppy Disk Controller (FDC) chip is the heart of the iSBX 218 Controller. On-board data separation logic performs standard MFM (double density) and FM (single density) encoding and decoding, eliminating the need for external separation circuitry at the drive. Data transfers between the controller and memory are managed by the intelligent device (usually an Intel 8-bit or 16-bit CPU chip) on the host board. A block diagram of the iSBX 218 Controller is shown in Figure 1.

Universal Drive and iSBX 218 Controller

Because the iSBX 218 Controller has universal drive compatibility, it can be used to control virtually any standard- or mini-sized diskette drive. Moreover, the iSBX 218 Controller fully supports the iSBX bus and can be used with any single board computer which furnishes this bus. Because the iSBX 218 Controller is programmable, its performance is not compromised by its universal drive compatibility. The track-to-track access, head-load, and head-unload characteristics of the selected drive model are program specified. Data may be organized in sectors up to 8192 bytes in length.

Interface Characteristics

The standard iSBX 218 Controller includes an Intel 8272 Floppy Disk Controller chip which supports up to four drives, single or double sided.

SIMPLIFIED INTERFACE—The cables between the iSBX 218 Controller and the drive(s) may be low cost, flat ribbon cable with mass termination connectors. The mechanical interface to the board is a right-angle header with locking tabs for security of connection.

PROGRAMMING — The powerful 8272 FDC circuit is capable of executing high-level commands that simplify system software development. The device can read and write both single and multiple sectors. CRC characters are generated and checked automatically. Recording density is selected at each Read and Write to support the industry standard technique of recording basic media information on Track 0 of Side 0 in single density, and then switching to double density (if necessary) for operations on other tracks.

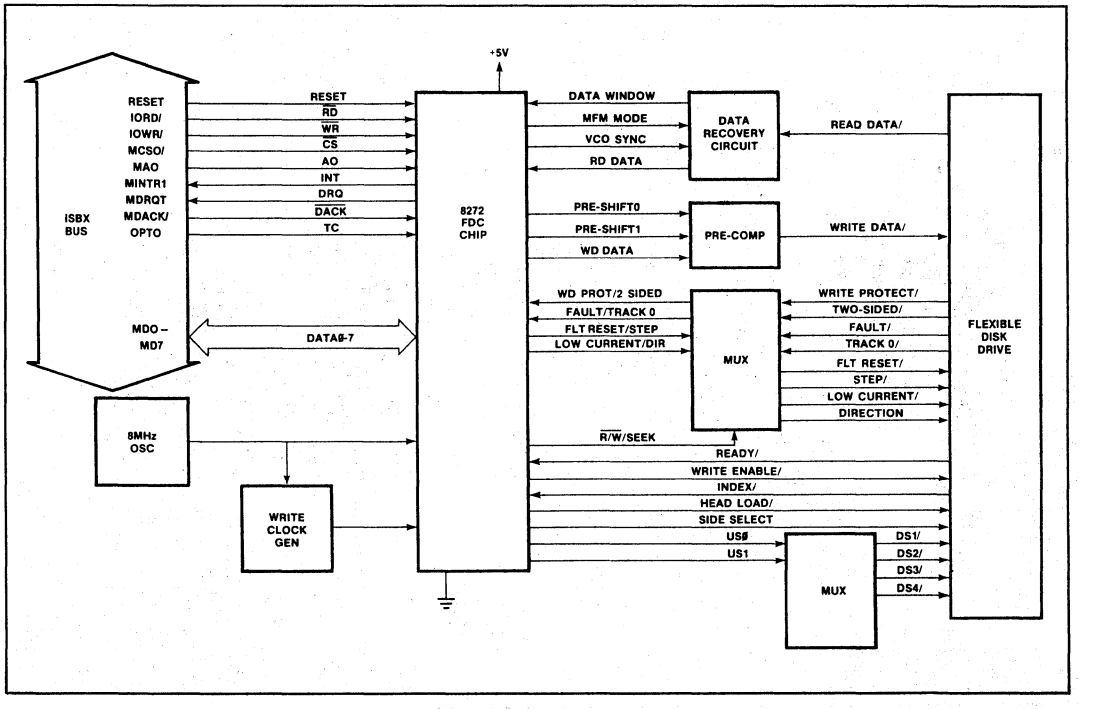


Figure 1. Block Diagram of iSBX 218 Board

PROGRAM INITIATION—All diskette operations are initiated by standard iSBX bus input/output (I/O) operations through the host iSBC single board computer. System software first initializes the controller with the operating characteristics of the selected drive. The diskette is then formatted under program control. Data transfers occur in response to commands output by the CPU.

DATA TRANSFER—Once a diskette transfer operation has been initiated, the controller will require a data transfer every 13 microseconds (double density) or 26 microseconds (single density). Most CPUs will operate in a polled mode, checking controller status and transferring bytes when the controller is ready. Boards utilizing the Intel 8080 chip, such as the iSBC 80/10B board, will be restricted to single density operation with the iSBX 218 Controller, due to these speed requirements. A programming example illustrating the iSBC 80/10B handler is contained in the Hardware Reference Manual.

SPECIFICATIONS

Compatibility

CPU—Any iSBC single board computer or I/O board compatible with the MULTIBUS system bus *and* implementing the iSBX bus and connector.

Devices—Double or single density standard (8″) and mini (5¼″) flexible disk drives. The drives may be single or double sided. Drives known to be compatible are:

Standard (8″)		Mini (5¼″)	
Caldisk	143M	Shugart	450/400
Remex	RFD 4000	Micropolis	1015-IV
Memorex	550	Pertec	250
MFE	700	Siemens	200-5
Siemens	FDD 200-8	Tandon	TM-100
Shugart	SA 850/800	CDC	9409
Pertec	FD 650	MPI	51/52/91/92
CDC	9406-3		

Diskette—Unformatted IBM Diskette 1 (or equivalent single-sided media); unformatted IBM Diskette 2D (or equivalent double-sided).

Equipment Supplied

iSBX 218 Controller
Reference Schematic
Controller-to-drive cabling and connectors are not supplied with the controller. Cables can be fabricated with flat cable and commercially-available connectors as described in the iSBX 218 Hardware Reference Manual.
Nylon Mounting Bolts

Physical Characteristics

Width—2.85 inches (7.24 cm)
Height—0.5 inches (1.27 cm)
Length—7.5 inches (19.05 cm)
Shipping Weight—1 pound (0.46 Kg)
Mounting—Occupies one double-wide iSBX MULTIMODULE position on boards; increases board height (host plus iSBX board) to 1.13 inches (2.87 cm).

Data Organization and Capacity

Standard Size Drives

	Double Density						Single Density					
	IBM System 34			Non-IBM			IBM System 3740			Non-IBM		
Bytes per Sector	256	512	1024	2048	4096	8192	128	256	512	1024	2048	4096
Sectors per Track	26	15	8	4	2	1	26	15	8	4	2	1
Tracks per Diskette	77			256			77			256		
Bytes per Diskette (Formatted, per diskette surface)	512,512 (256 bytes/sector) 591,360 (512 bytes/sector) 630,784 (1024 bytes/sector)			630,784			256,256 (128 byte/sector) 295,680 (256 bytes/sector) 315,392 (512 bytes/sector)			315,392		

Drive Characteristics

	Standard Size	Mini Size
	Double/Single Density	Double/Single Density
Transfer Rate (K bytes/sec)	62.5/31.25	31.25/15.63
Disk Speed (RPM)	360	300
Step Rate Time (Programmable)	1 to 16 msec/track in 1 msec increments	2 to 32 msec/track in 2 msec increments
Head Load Time (Programmable)	2 to 256 msec in 2 msec increments	4 to 512 msec in 4 msec increments
Head Unload Time (Programmable)	0 to 240 msec in 16 msec increments	0 to 480 msec in 32 msec increments

Electrical Characteristics

Power Requirements— +5 VDC @ 0.81A

Environmental Characteristics

Temperature—0°C to 55°C (operating); −55°C to +85°C (non-operating).

Humidity—Up to 90% Relative Humidity without condensation (operating); all conditions without condensation or frost (non-operating).

Reference Manual

121583-001—iSBX 218 Flexible Disk Controller Hardware Reference Manual (NOT SUPPLIED).

Reference manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051.

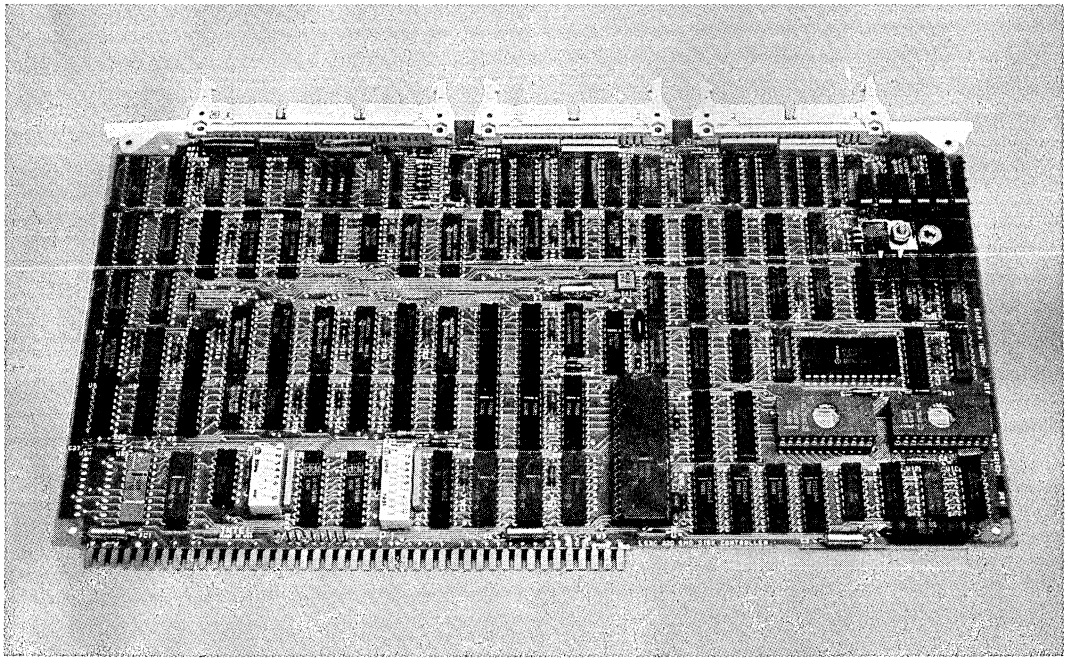
ORDERING INFORMATION

Part Number	Description
SBX 218	Flexible Disk Controller

iSBC™ 220
SMD DISK CONTROLLER

- **Controls up to four SMD interface compatible disk drives**
- **12 MB to 2.4 GB per controller**
- **Compatible with all iSBC™ 80, iSBC™ 88, and iSBC™ 86 Single Board Computers**
- **Intel® 8089 I/O Processor provides two high speed DMA channels as well as controller intelligence**
- **Software drivers available for iRMX™ 86 and iRMX™ 88 operating systems**
- **On-board diagnostic and ECC**
- **Full sector buffering on-board**
- **Capable of addressing 1 MB of system memory**
- **SMD interface available on 14″ Winchester, CMD, SMD and large fixed-media drives**

The iSBC 220 SMD Disk Controller brings very large mass storage capabilities to any iSBC 80, iSBC 88, or iSBC 86 MULTIBUS system. The controller will interface to any disk drive conforming to the industry standard SMD interface. Using simplified cable connections, up to four drives may be connected to the iSBC 220 Controller Board to give a total maximum capacity of 2.4 gigabytes. The Intel 8089 I/O Processor simplifies programming through the use of memory-based parameter blocks. A linked list technique allows the user to perform multiple disk operations.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: CREDIT, Index, Intel, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS, and the combination of MCS, ICE, iSBC, iSBX or iCS, and a numerical suffix. 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 CORPORATION, 1980

FUNCTIONAL DESCRIPTION

Full On-Board Buffer

The iSBC 220 SMD Controller contains enough on-board RAM for one full sector buffering. The controller is designed to make use of this buffer in all transfers. The on-board sector buffer prevents data overrun errors and allows the iSBC 220 SMD Controller to occupy any priority slot on the MULTIBUS.

ECC

High data integrity is provided by on-board Error Checking Code (ECC) logic. When writing sector ID or data fields, a 32-bit Fire code, for burst error correction, is appended to the field by the controller. During a Read operation, the same logic regenerates the ECC polynomial and compares this second polynomial to the appended ECC. The ECC logic can detect an erroneous data burst up to 32 bits in length and using an 8089 alrogithm can correct an erroneous burst up to 11 bits in length.

SMD Interface

High speed, reliable data transfers are a major benefit of using the SMD interface. A data transfer rate of 1.2 MB is accomplished by using separate (radial) differential data line cabling for each drive. Control signals are daisy-chained from drive to drive.

Defective Track Handling

When a track is deemed defective, the host processor reformats the track, giving it a defective track code and enters the address of the next available alternate track. When the controller accesses a track previously marked defective, the controller automatically seeks to the assigned alternate track. The alternate track seek is totally automatic and invisible to the user.

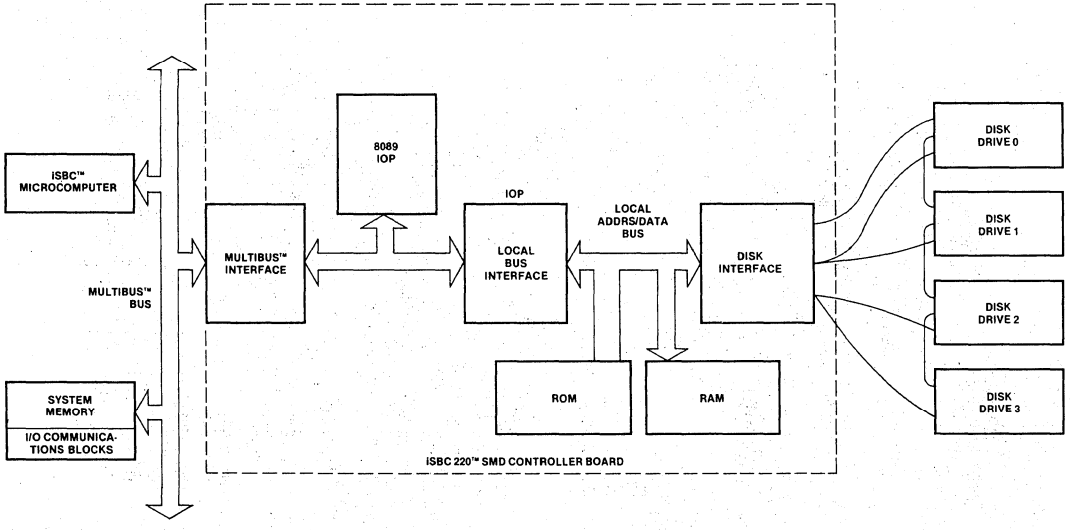


Figure 1. Simplified Block Diagram of iSBC 220™ SMD Disk Controller

AFN-01794A

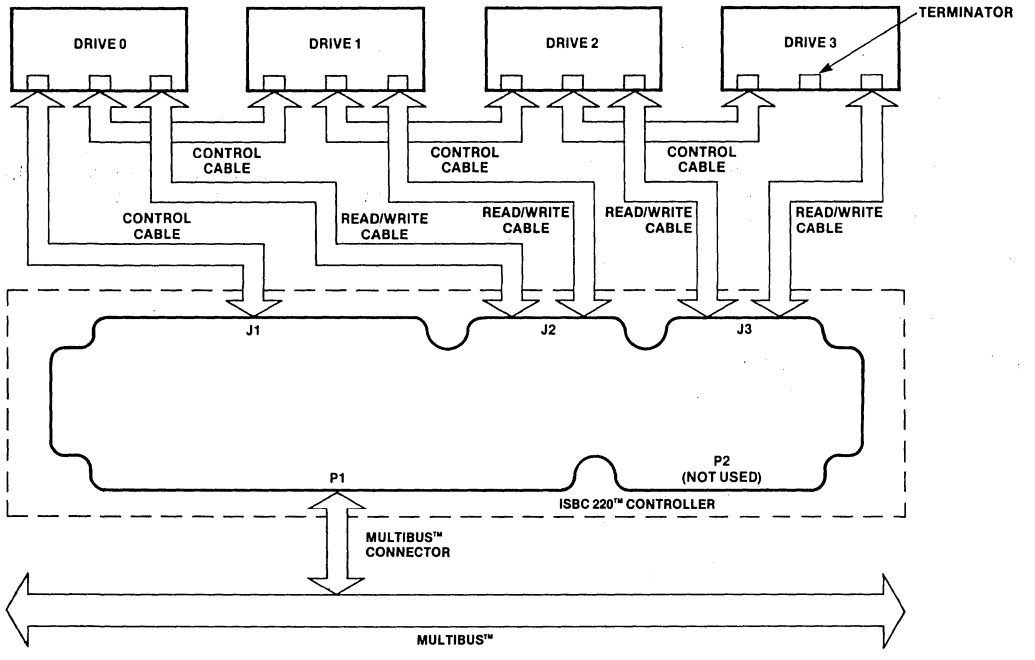


Figure 2. Typical Multiple Drive System

SPECIFICATIONS

Compatibility

CPU — Any iSBC MULTIBUS computer or system mainframe

Disk Drive — Any SMD interface-compatible disk drive

Equipment Supplied

iSBC 220 SMD Disk Controller

Reference schematic

Controller-to-drive cabling and connectors are not supplied with the controller. Cables can be fabricated with flat cable and commercially-available connectors as described in the iSBC 220 SMD Disk Controller Hardware Reference Manual.

Physical Characteristics

Width — 6.75 in. (17.15 cm)

Height — 0.5 in. (1.27 cm)

Length — 12.0 in. (30.48 cm)

Shipping Weight — 19 oz (0.54 kg)

Mounting — Occupies one slot of iSBC system chassis or cardcage/backplane

Electrical Characteristics

Power Requirements

+5 VDC @ 3.25A max

−5 VDC @ 0.75A max[1]

Note 1: On-board voltage regulator allows optional −12 VDC usage from MULTIBUS.

Data Organization and Capacity

Bytes per Sector[2] — 128 256 521 1024

Sectors per Track[2] — 108 64 35 18

Note 2: Software selectable.

Table 1. Drive Characteristics (Typical)

Disk (spindle) Speed	3600 rpm
Tracks per Surface	823
Head Positioning	Closed loop servo type, track following
Access Time	Track to Track 6 ms Average 30 ms Maximum 55 ms
Data Transfer Rate	1.2 megabytes/second
Storage Capacity	12 to 2.4 gigabytes

Environmental Characteristics

Temperature — 0°C to 55°C (operating); −55°C to +85°C (non-operating)

Humidity — Up to 90% relative humidity without condensation (operating); all conditions without condensation or frost (non-operating)

Reference Manual

121597-001 — iSBC 220 SMD Disk Controller Hardware Reference Manual (NOT SUPPLIED)

Reference manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051.

ORDERING INFORMATION

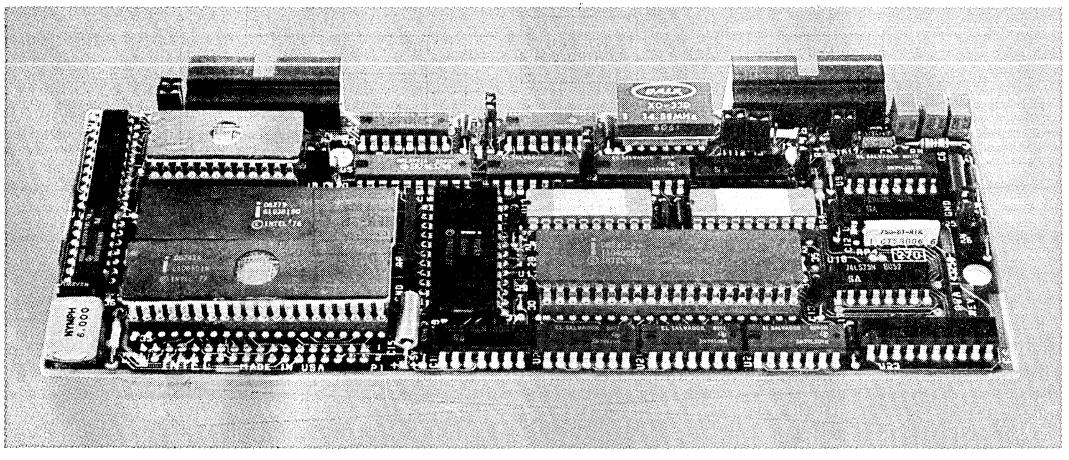
Part Number	Description
SBC 220	SMD Disk Controller

iSBX™ 270
VIDEO DISPLAY CONTROLLER

- **Complete video display controller on a double-wide iSBX™ MULTIMODULE™ board**
- **Interfaces to either black and white or color display monitors**
- **Displays 7 x 9, 5 x 7 or 6 x 8 character fonts**
- **High level software interface via a pre-programmed 8041A UPI**
- **Interchangeable character fonts available in EPROM**
- **Keyboard and light pen interface provided on-board**
- **50 Hz or 60 Hz frame rate operation**
- **Provides cursor control, reverse video, blinking, underline, highlight and page or scroll mode**
- **Compatible with all 8/16 bit iSBC™ boards which support the Intel iSBX™ bus**
- **Graphics capability via pre-defined graphic character fonts**

The iSBX 270 Video Display Controller (VDC) is a complete video controller on a standard double wide Intel iSBX MULTIMODULE board. Providing either black and white (B&W) or eight-color displays, the iSBX 270 VDC brings alphanumeric video control to the iSBX bus. Any computer board or system supporting the Intel iSBX MULTIMODULE bus is compatible with the iSBX 270 VDC, including most board and system products from Intel. Additionally, the iSBX 270 VDC supports keyboard and light pen I/O on-board; this simplifies the design of intelligent terminals.

The iSBX 270 module allows the user to add high level video display capability to his/her computer system with a minimal cost and effort. Typical applications for the iSBX 270 VDC include video displays for industrial operator stations, word processing systems, data base management products and many other uses.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1981

October, 1981
Order Number: 210220-001

FUNCTIONAL DESCRIPTION

iSBX™ Interface

The iSBX 270 VDC interfaces to the Intel iSBX bus via the 8041A Universal Peripheral Interface (UPI) Microcomputer. The 8041A, under firmware control, provides communication between the base board and the iSBX 270 controller circuitry via the iSBX data and control lines. Data may be displayed immediately following power up, using default initialization provided by the 8041A UPI. In addition, eight high-level commands are provided by the iSBX 270 firmware; these eight commands are used to alter the default initialization of the controller and determine status. Following initialization, characters are displayed on the CRT by simply writing to the proper I/O port.

CRT Interface

The iSBX 270 VDC will interface to many B&W and RGB color display monitors. For B&W monitors, the iSBX 270 board provides TTL level signals for video, vertical sync and horizontal sync. Additionally, in B&W, two levels of intensity (normal and highlight) are supported under program control.

When operating in the color mode, the iSBX 270 module provides TTL level 75 ohm line drivers for Red, Green, and Blue Video and sync allowing 8 different colors to be displayed.

Composite video is not provided on the iSBX 270 MULTIMODULE board; however, with minimal external circuitry, composite video can be added (circuit design available; contact the local Intel Sales Office for details).

Table 1 lists several CRT vendors compatible with the iSBX 270 VDC.

Table 1. CRT's (B&W and Color)[1]

TYPE	VENDOR	MODEL #
B&W	Ball Brothers	TTL 120, TV 120, TV 50
	Motorola	M3570
	TSD	MDC-15
	ELSTON	DM30-12B0-51-A04
Color	Ball Brothers	7-015-0131
	IDT	19AC
	CONRAC	5711C13
	NEC	1202DH
	MITSUBISHI	C-3419

[1]**NOTE:** This in no way constitutes an endorsement by Intel Corporation of these companies' products. The companies listed are known to provide products compatible with the iSBX 270 board.

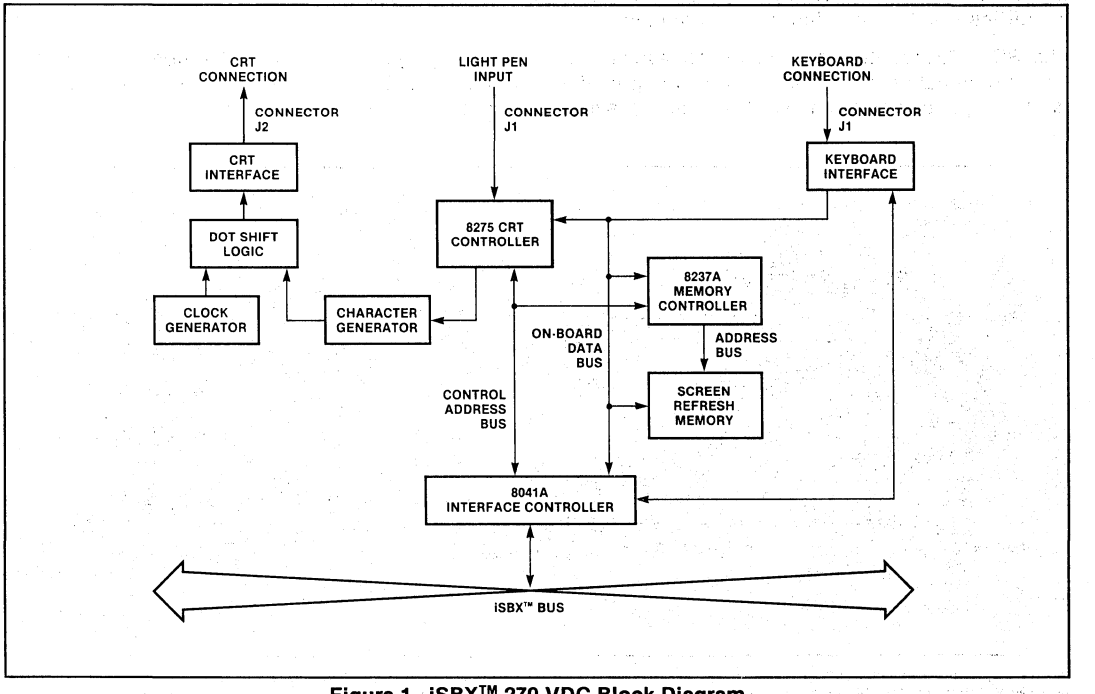


Figure 1. iSBX™ 270 VDC Block Diagram

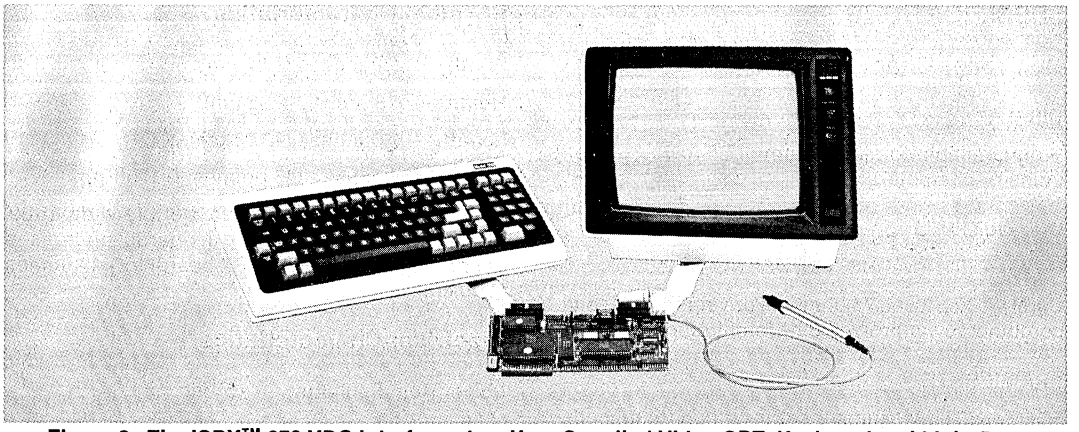


Figure 2. The iSBX™ 270 VDC Interfaces to a User-Supplied Video CRT, Keyboard and Light Pen

CRT Controller

The CRT Controller performs all timing and data buffering functions for the CRT. The iSBX 270 VDC uses the Intel 8275 CRT Controller (for additional details refer to the 8275 data sheet available from Intel).

Screen Refresh

The iSBX 270 VDC contains 4K bytes of high speed static RAM, as well as a high speed DMA controller (8237A). The 8237A, under the control of the 8041A UPI, takes care of both writing data to the screen and refreshing the screen.

Character Generation

The character fonts (128 characters, including alphabetic, numeric, and special characters) that are displayed on the CRT are stored in EPROM. The need may arise to display different character fonts, i.e., those used in international systems or custom symbols which are application specific. With the iSBX 270 VDC the user may modify any or all of the character fonts by simply reprogramming the EPROM. In addition, the user may utilize a larger EPROM to obtain up to 256 characters.

Keyboard Interface

The iSBX 270 VDC also interfaces to a keyboard I/O device via the J1 edge connector. The keyboard interface of the iSBX 270 VDC accepts up to eight TTL parallel data lines and one TTL strobe, either positive or negative. Keyboard input is indicated by a status bit in the 8041A and/or an interrupt. In addition, control lines are provided for visual and/or audible indicators.

Table 2 lists several keyboards that interface to the iSBX 270 VDC.

Table 2. Keyboards[1]

VENDOR	MODEL #
Advanced Input Devices	SK-067
Cherry	B70-05AB
Cherry	CB80-07AA
Chomerics	AN26109/AE26203
Cortron	35-500014
Keytronic	L1648
Keytronic	L1660
Keytronic	L1674-03
Keytronic	L1752
Microswitch	66SD6-7
Microswitch	87SD30-8

[1]**NOTE:** This in no way constitutes an endorsement by Intel Corporation of these companies' products. The companies listed are known to provide products compatible with the iSBX 270 board.

Light Pen Interface

Light pen I/O devices may be directly interfaced to the iSBX 270 VDC. A light pen hit is triggered on the rising edge of the light pen signal and is indicated by a status bit in the UPI 8041A and/or an interrupt.

Table 3 lists a light pen vendor whose product interfaces to the iSBX 270 VDC.

Table 3. Light Pens[1]

VENDOR	MODEL #
Information Control Co.	LP-700

[1]**NOTE:** This in no way constitutes an endorsement by Intel Corporation of this company's products. The company listed is known to provide products compatible with the iSBX 270 board.

SPECIFICATIONS

Controller Characteristics

DISPLAY

Programmable to a maximum of 35 rows x 80 columns of characters.

CRT OUTPUTS

B&W — TTL level HSYNC, VSYNC, Video.

Color — TTL level, 75 ohm line drivers for RGB and combined sync provide 8 different display colors.

FRAME RATE

50 Hz or 60 Hz via jumper settings (non-interlaced).

CHARACTER FONTS

5 x 7, 7 x 9 or 6 x 8 jumperable with appropriate crystal. Character generator uses 2716 EPROM. Also compatible with 2732A EPROM's. For generation of special fonts, please refer to iSBX 270 VDC Hardware Reference Manual.

VIDEO CONTROL

Reverse video, blinking, underline, highlight, cursor control and page or scroll mode.

TV MONITOR

Most video display monitors with a 10 MHz bandwidth or better.

LIGHT PEN INPUT

TTL level pulse, maximum 50 ns rise time, minimum 100 ns hold time.

Compatibility

CPU

Any iSBC single board computer or I/O board compatible with the MULTIBUS system bus and implementing the iSBX bus and connector.

Physical Characteristics

Width — 3.08 inches (7.82 cm)

Height — 0.8 inches (2.05 cm)

Length — 7.5 inches (19.05 cm)

Shipping Weight — 0.5 pounds (0.175 Kg)

Mounting — Occupies one double-wide iSBX MULTIMODULE position on boards; increases board height (host plus iSBX board) to 1.14 inches (2.90 cm).

Electrical Characteristics

Power Requirements +5 Vdc @ 1.3A.

Environmental Characteristics

Temperature — 0°C to 55°C (operating); −55°C to +85°C (non-operating).

Humidity — Up to 90% relative humidity without condensation (operating); all conditions without condensation or frost (non-operating).

Equipment Supplied

iSBX 270 VDC Controller
Reference Schematic

Cabling and connectors from the VDC controller to the CRT, keyboard and light pen are not supplied with the controller. Cables can be fabricated with commercially available cable and connectors as described in the iSBX 270 Hardware Reference Manual.

Reference Manual

143444-001 — iSBX 270 Video Display Controller Hardware Reference Manual (NOT SUPPLIED).

Reference manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051.

ORDERING INFORMATION

Part Number	Description
SBX 270	Video Display Controller MULTIMODULE Board

Memory Expansion Boards

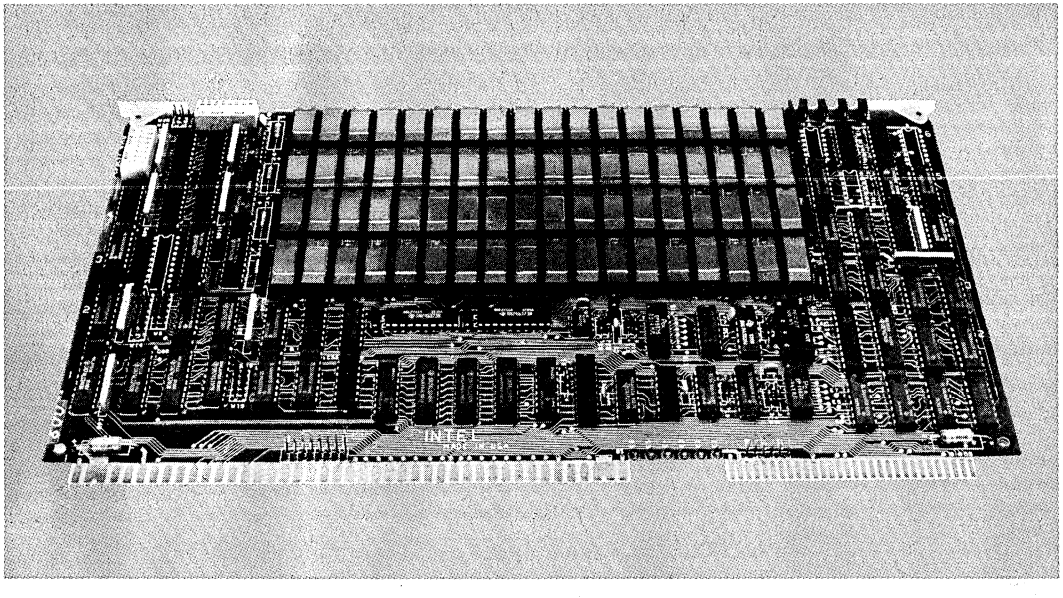
7

iSBC™ 012B
RAM MEMORY BOARDS

- **iSBC 86, iSBC 88 and iSBC 80 board RAM expansion through direct MULTIBUS® interface**
- **512K of read/write memory**
- **On-board parity generator/checker and error status register**
- **Requires a single +5 volt power supply**
- **Assignable anywhere within a 16 megabyte address space**
- **Jumper selectable base address on any 16K byte boundary**
- **Auxiliary power bus and memory protect control logic for battery backup RAM requirements**

The iSBC 012B RAM memory board is a member of Intel's complete line of iSBC memory and I/O expansion boards. The board interfaces directly to any iSBC 86, iSBC 88 or iSBC 80 Single Board Computer via the MULTIBUS interface to expand system RAM capacity. The iSBC 012B board contains 512K bytes of read/write memory implemented using dynamic RAM components. An on-board dynamic RAM controller refreshes a portion of these components every 16 microseconds. Each refresh cycle utilizes memory for 550 nanoseconds (maximum).

The iSBC 012B board generates byte oriented parity during all write operations and performs parity checking during all read operations. When a parity error is detected, the board can generate an interrupt on the MULTIBUS interface. In addition, the row and bank of the RAM array containing the error are stored in a Parity Flag Register. This register is accessible as a MULTIBUS I/O port. An on-board LED also provides a visual indication that a parity error has occurred.



The following are trademarks of Intel Corporation and may be used to describe Intel products: CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE, ICS, iAPX and iMMX. 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 CORPORATION, 1981

SPECIFICATIONS

Word Size

8 bits and 16 bits

Memory Size

524,288 bytes (iSBC 012B)

Access Time

330 nsec (worst case)
300 nsec (typical)

Cycle Times (Worst Case)

Read — 500 ns max.
Write — 500 ns max.
Refresh — 550 ns max.

Interface

All address, data and command signals are TTL compatible.

Address Selection

Memory — Base address is jumper selectable on any 16K byte boundary in a 16 megabyte address space. On-board RAM cannot cross a 4 megabyte address boundary.

Parity Flag Register — The I/O address of the Parity Flag Register is jumper selectable to be between 00H to 0FH or 40H to 4FH.

Connector

Edge connector — 86 pin double-sided PC edge connector with 0.156 in. contact centers.

Mating connector — Viking 3KH43/9AMK12 or equivalent.

Auxiliary Power

An auxiliary power bus is provided to allow separate power to RAM array for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Memory Protect

An active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM on the board. This input is provided for the protection of RAM contents during system power-down sequences.

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 14 oz. (397 gm)

Electrical Characteristics

D.C. POWER REQUIREMENTS

All configurations require only +5 volts ±5%.

Normal System Operation (max.)
4.8A (worst case)
3.46A (typical)

Auxiliary Power No RAM Access (max.)
1.35A (worst case)
0.88A (typical)

Environmental Characteristics

Operating Temperature — 0°C to +55°C

Relative Humidity — to 90% (without condensation)

Reference Manual

143865-001 — iSBC 056B/012B Hardware Reference Manual (not supplied)

Manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Dept., 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number Description

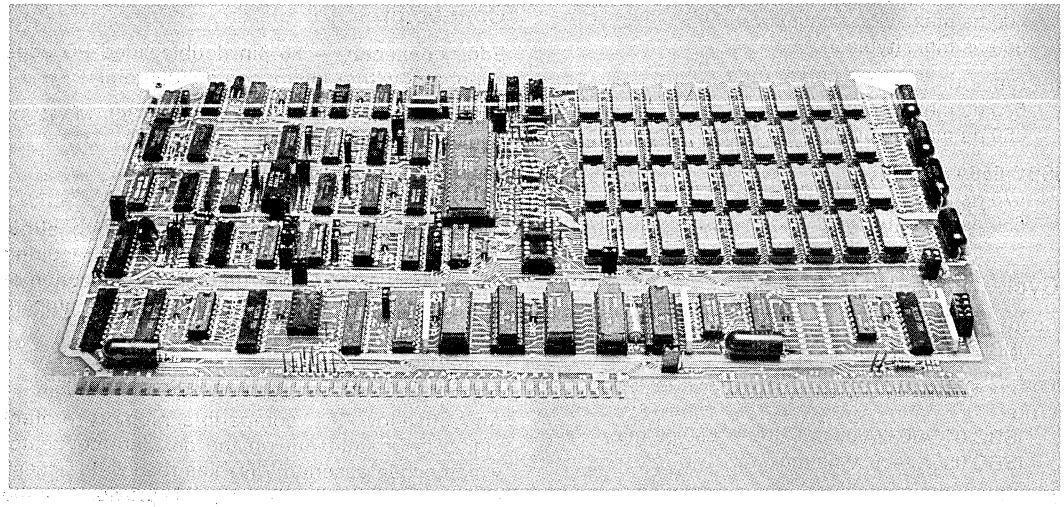
SBC 012B	512K-Byte RAM Board with Parity

iSBC™ 016A/032A/064A/028A/056A RAM MEMORY BOARDS

- iSBC 86, iSBC 88 and iSBC 80 board RAM expansion through direct MULTIBUS® interface
- 16K, 32K, 64K, 128K or 256K bytes of read/write memory
- On-board parity generator/checker and error status register
- Requires a single +5 volt power supply
- Assignable anywhere within a 16 megabyte address space
- Jumper selectable base address on any 4K byte boundary
- Auxiliary power bus and memory protect control logic for battery backup RAM requirements

The iSBC 016A, iSBC 032A, iSBC 064A, iSBC 028A and iSBC 056A RAM memory boards are members of Intel's complete line of iSBC memory and I/O expansion boards. Each board interfaces directly to any iSBC 80, iSBC 88 or iSBC 86 Single Board Computer via the MULTIBUS interface to expand system RAM capacity. The iSBC 016A, iSBC 032A, iSBC 064A, iSBC 028A and iSBC 056A boards contain 16K, 32K, 64K, 128K or 256K bytes of read/write memory implemented using dynamic RAM components. An on-board LSI dynamic RAM controller refreshes a portion of these components every 14 microseconds. Each refresh cycle utilizes memory for 480 nanoseconds (maximum).

The iSBC 032A, iSBC 064A, iSBC 028A and iSBC 056A boards generate byte oriented parity during all write operations and perform parity checking during all read operations. When a parity error is detected, these boards can generate an interrupt on the MULTIBUS interface. In addition, the row and bank of the RAM array containing the error are stored in a Parity Flag Register (see Figure 1). This register is accessible as a MULTIBUS I/O port. An on-board LED also provides a visual indication that a parity error has occurred. To facilitate testing of these boards, parity generation and checking can be changed from even to odd under software control.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE, ICS, iAPX and iMMX. 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 CORPORATION, 1981

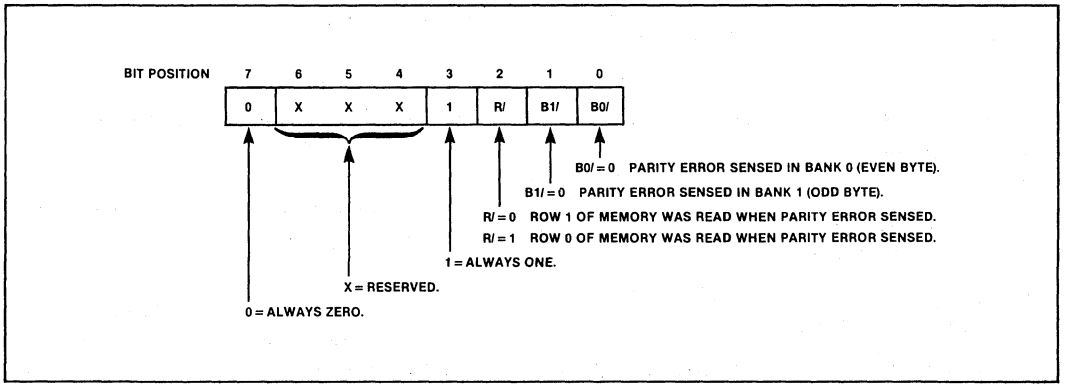


Figure 1. Parity Flag Register Format

SPECIFICATIONS

Word Size

8 bits and 16 bits

Memory Size

16,384 bytes (iSBC 016A); 32,768 bytes (iSBC 032A); 65,536 bytes (iSBC 064A); 131,072 bytes (iSBC 028A); or 262,144 bytes (iSBC 056A)

Access Time

iSBC 016A/032A/064A
400 ns max. (worst case)
360 ns max. (typical)

iSBC 028A
500 ns max. (worst case)
460 ns max. (typical)

iSBC 056A
570 ns max. (worst case)
530 ns max. (typical)

Cycle Times (Worst Case)

Read
iSBC 016A/032A/064A/028A — 600 ns max.
iSBC 056A — 650 ns max.

Write
iSBC 016A/032A/064A/028A — 600 ns max.
iSBC 056A — 650 ns max.

Refresh
iSBC 016A/032A/064A/028A — 480 ns max.
iSBC 056A — 600 ns max.

Interface

All address, data and command signals are TTL compatible.

Address Selection

Memory — Base address is jumper selectable on any 4K byte boundary in a 16 megabyte address space. On-board RAM cannot cross a megabyte address boundary.

Parity Flag Register — The I/O address of the Parity Flag Register is jumper selectable to be between 00H to 0FH or 40H to 4FH.

Connector

Edge connector — 86 pin double-sided PC edge connector with 0.156 in. contact centers.

Mating connector — Viking 3KH43/9AMK12 or equivalent.

Auxiliary Power

An auxiliary power bus is provided to allow separate power to RAM array for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Memory Protect

An active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM on the board. This input is provided for the protection of RAM contents during system power-down sequences.

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 14 oz. (397 gm)

Electrical Characteristics

D.C. POWER REQUIREMENTS

All configurations require only +5 volts ±5%.

Normal System Operation (max.)

iSBC 016A — 3.3A (worst case)
2.65A (typical)
iSBC 032A/064A — 4.0A (worst case)
3.20A (typical)
iSBC 028A/056A — 4.57A (worst case)
3.66A (typical)

Auxiliary Power No RAM Access (max.)

iSBC 016A — 0.37A (worst case)
0.30A (typical)
iSBC 032A/064A — 0.41A (worst case)
0.33A (typical)
iSBC 028A/056A — 0.55A (worst case)
0.45A (typical)

Environmental Characteristics

Operating Temperature — 0°C to +55°C
Relative Humidity — to 90% (without condensation)

Reference Manual

143572-001 — iSBC 016A/032A/064A/028A/056A Hardware Reference Manual (not supplied)

Manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Dept., 3065 Bowers Avenue, Santa Clara, California 95051.

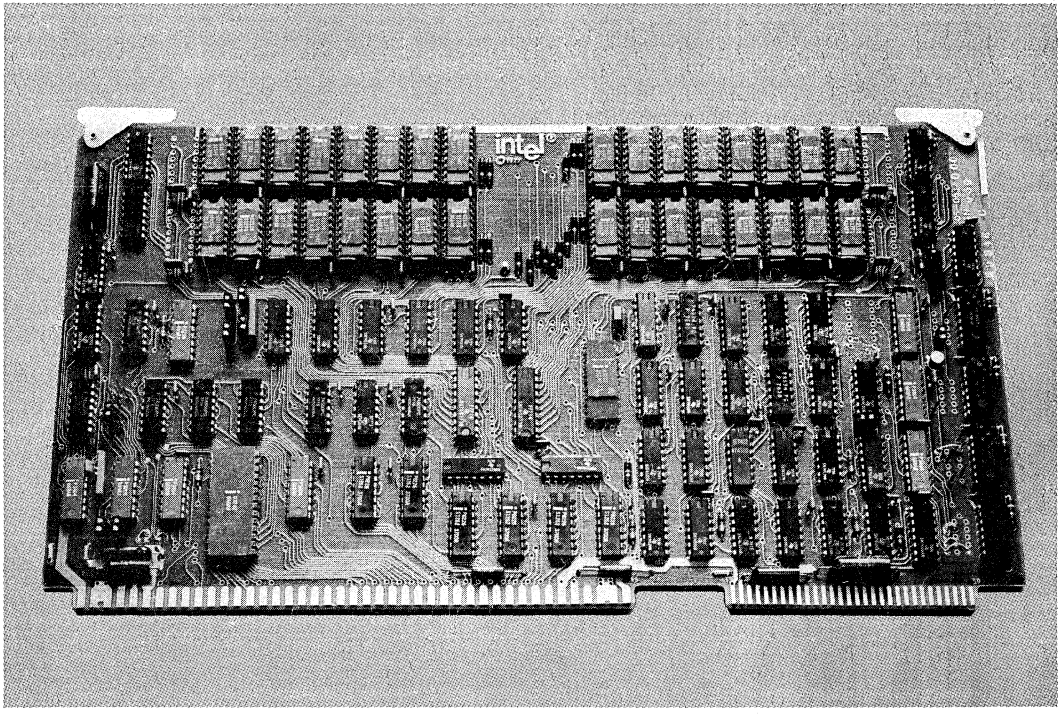
ORDERING INFORMATION

Part Number	Description
SBC 016A	16K-Byte RAM Board
SBC 032A	32K-Byte RAM Board with Parity
SBC 064A	64K-Byte RAM Board with Parity
SBC 028A	128K-Byte RAM Board with Parity
SBC 056A	256K-Byte RAM Board with Parity

iSBC™ 064
RAM MEMORY BOARDS

- **iSBC™ 80, iSBC™ 86 and iSBC™ 88 RAM memory expansion through direct MULTIBUS® interface**
- **64K bytes of read/write memory**
- **On-board hardware for refresh of all dynamic memory elements**
- **Read/write data buffers**
- **Auxiliary power bus and memory protect control logic provided for battery backup RAM requirements**
- **Jumper selectable starting address on any 64K boundary in a megabyte address space**
- **TTL compatible data, address, and command signal interface**

The iSBC 064 RAM Memory Board is a member of Intel's complete line of iSBC memory and I/O expansion boards. Each board interfaces directly to any Intel iSBC 80, iSBC 86 or iSBC 88 single board computer via the MULTIBUS interface to expand RAM memory capacity. The iSBC 064 contains 64K bytes of read/write memory implemented using dynamic RAM memory components. On-board refresh hardware refreshes a portion of RAM memory every 14 microseconds. Each refresh cycle utilizes memory for 585 nanoseconds. If a read or write cycle is in progress when a refresh cycle is scheduled to begin, the refresh cycle is postponed until the end of the cycle. Read/write buffers reside on each board to buffer all data written into or read from the memory array. All data, address, and command signals on the bus interface are TTL compatible.



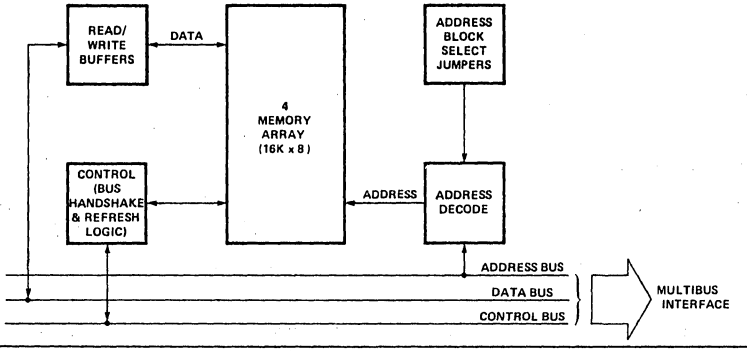


Figure 1. RAM Memory Expansion Boards Block Diagram

SPECIFICATIONS

Word Size
8 bits and 16 bits

Memory Size
65,536 bytes

Access Time
450 ns max

Cycle Times
Read Cycle — 700 ns max
Write Cycle — 600/1240 ns max
Refresh Cycle — 700 ns max

Interface
All address, data, and command signals TTL compatible.

Address Selection
Jumper selection of a 64K byte page in a megabyte address space.

Connectors
Edge Connectors — 86-pin double-sided PC edge connector with 0.156-in. contact centers.

Mating Connector — Viking 3KH43/9AMK12

Auxiliary Power
An auxiliary power bus is provided to allow separate power to RAM for systems requiring battery back up of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Memory Protect
An active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences.

Physical Characteristics
Width — 12.00 in. (30.48 cm)
Height — 6.76 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 14 oz (415.2 gm)

Electrical Characteristics
DC Power Requirements

Voltage	Normal System Operation (max)[1]	AUX Power RAM Access (max)[2]	AUX Power No RAM Access (max)
V_{CC} = +5V ±5%	I_{CC} = 3.2A	1.7A	1.7A
V_{DD} = +12V ±5%	I_{DD} = 600 mA	600 mA	120 mA
V_{BB} = −5V ±5%	I_{BB} = 10 mA	10 mA	3 mA

Notes
1. All current values include AUX power.
2. RAM chips and RAM control logic powered via auxiliary power bus.
3. Power necessary to refresh RAMs and maintain data, as after system power failure.

Environmental Characteristics
Operating Temperature — 0 °C to +55 °C

Reference Manual
9800488B — iSBC 032/048/064 Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
SBC 064	64K-Byte RAM Board

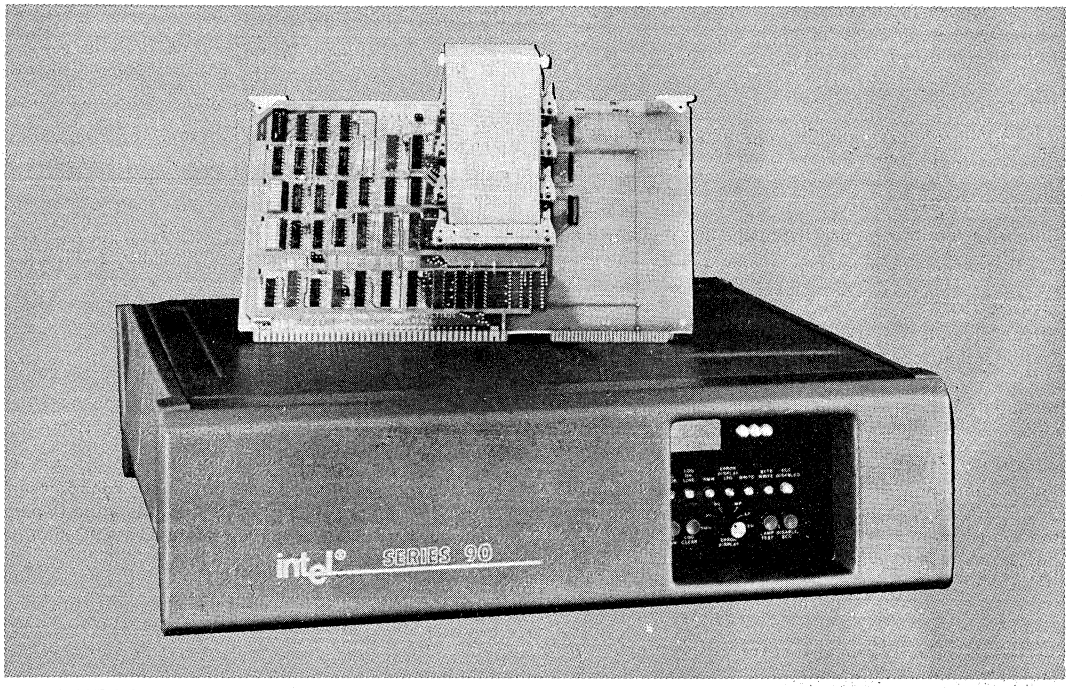
iSBC 090™ MEMORY SYSTEM

- **512K, 768K or 1024 K-byte capacities**
- **Compatible with Intellec Model 286 Series III development system**
- **Error check and correcting (ECC) circuitry provides single-bit correction and double-bit detection**
- **8-bit or 16-bit word transfer**
- **Error logger and display isolates single bit errors to the memory device level**
- **Switch-selectable starting address on any 4K byte boundary**
- **Field expandability**

The Intel® iSBC 090 Memory System is a random-access dynamic memory system to be used with Intel's MULTIBUS™ System and iSBC 80/86™ product line. The iSBC 090 Memory System can provide up to 1 Megabyte of memory in 256K-byte increments or up to 512K-bytes in 128K-byte increments. It consists of a MULTIBUS Interface Board and a Series 90 Random Access Dynamic Memory. The Interface Board plugs directly into the MULTIBUS backplane, and through four ten-foot cables interfaces the MULTIBUS system and the memory.

The Interface Board is a 12.0 inch by 6.75 inch printed circuit board that occupies any one slot in the MULTIBUS backplane. It allows the user to select the beginning and ending addresses to which the iSBC 090 Memory System will respond; it permits either 8-bit or 16-bit bus masters, or mixes of both, as well as supporting the normal read, write, refresh, and inhibit RAM cycles of the MULTIBUS system. Operating power for the interface is supplied by the MULTIBUS chassis and is not affected by increases in memory capacity.

The memory is a self-contained unit measuring 5.21 inches high by 19.00 inches wide and by 19.5 inches deep. It includes a memory storage area, which provides up to 1 Megabyte of memory. In addition, its control interface provides single-bit error detection and correction, double-bit error detection, and refresh arbitration. The memory also includes an error logger and error display, as well as its own power supplies and blower assembly. It is available as a table-top system with either 115 or 220 VAC input power.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Index, Intel, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, RMX, UPI, μScope, Promware, MCS, ICE, iSBC, iSBX, MULTIMODULE and iCS, and the combination of MCS, RMX, ICE, iSBC, iSBX or iCS and a numerical suffix. 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 CORPORATION, 1961.

FUNCTIONAL DESCRIPTION

The memory portion of the iSBC 090 Memory System (Figure 1) provides the memory storage area and the error detection and correction functions for the system.

The MULTIBUS Interface Board provides address compatibility between the MULTIBUS system and the memory; supports the four types of data transfer across the MULTIBUS interface; and generates the required control and status signals. Interface Board address circuits permit the iSBC 090 System to fit any address space from 4K-bytes to one megabyte, starting at any 4K boundary.

The Error Logger and Display records error, syndrome, and status signals, then on request displays them for quick location of memory errors.

Capacity

The iSBC 090 Memory System provides up to one megabyte of memory in 256K increments. Each 256K-byte increment is provided by adding one memory module in the memory. Three standard system capacities are offered: 512K, 768K and 1024K-bytes. The 768K-byte and 512K systems can be expanded to 1024K-bytes by the addition of one or two memory modules respectively.

Addressability

The address space which the iSBC 090 Memory System will occupy in the MULTIBUS addressing scheme is determined by two eight-position DIP switches on the Interface Board. The switches allow the memory to be any size, ranging from 4K-bytes to one megabyte, starting and ending on any 4K boundary.

Error Logger and Display

The Error Logger is a random-access memory which stores (logs) ECC syndrome bits that identify the failing bit and its location. The logger memory can store a maximum of 4096 single and double-bit errors. A display panel shows error information for user reference.

The error logger operates in three modes: log (write), scan (read), and clear. In normal system operation, the logger is operated in log mode and accessed only when one or more of the syndrome bits go active, indicating an error condition. The syndrome pattern identifying the error and the address of the error location are stored in the logger memory.

To look at the stored error information, the logger is placed in scan mode, taking it off-line. The logger memory is sequentially scanned until error information is reached. Scanning stops and the memory card identification, the card row, the data byte and the data bit are displayed on the logger display panel. The scan can be resumed using a scan control button.

INSTALLATION

The MULTIBUS Interface Board plugs directly into the MULTIBUS backplane. All operating power is furnished through the MULTIBUS connectors. Interface to the memory portion of the iSBC 090 Memory System is made using four 50-pin flat-ribbon cables, supplied with the System.

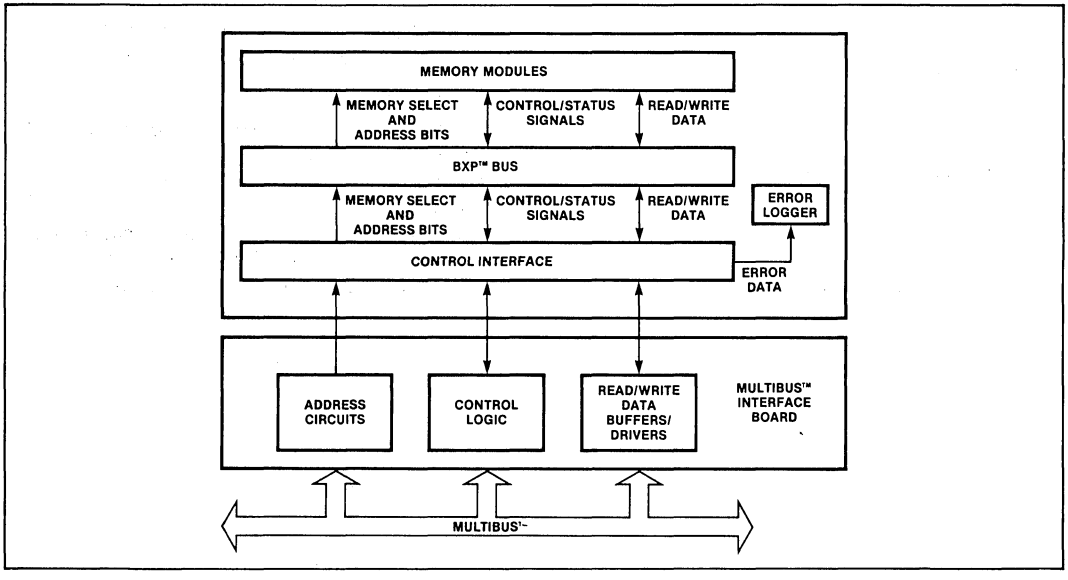


Figure 1. iSBC 090™ Memory System, Block Diagram

The iSBC 090 is available as a 19-inch rack-mounted unit with slide attachments or as a table top configuration with side covers instead of slides.

Depending on the configuration selected, the memory is connected to either 115 VAC, 50/60 Hz, single-phase, or 220 VAC, 50/60 Hz, single-phase power.

SPECIFICATIONS

Storage Capacity

512K, 768K, 1024K bytes

Word Length

8/16 bits plus 6 bits for ECC

Operating Cycles

Read Cycle
Write Cycle
Inhibit RAM Cycle
Refresh Cycle

Read Access Time (8- or 16-Bit Transfer)

450 nsec max (MRDC* to XACK*)

Write Access Time

8-Bit Transfer — 485 nsec max (MWTC* to XACK*)
16-Bit Transfer — 150 nsec max (MWTC* to XACK*)

*MWTC = Memory Write Command
XACK = Transfer Acknowledge
MRDC = Memory Read Command

(Reference MULTIBUS Manual 9800683)

Read Cycle Time (8- or 16-Bit Transfer)

485 nsec max

Write Cycle Time

8-Bit Transfer — 700 nsec max
16-Bit Transfer — 400 nsec max

Refresh Cycle Time (8- or 16-Bit Transfer)

450 nsec max

Logic Levels Input

Logic High — +2.0V to 5.25V
Logic Low — −0.5V to ±0.80V

Logic Levels Output

Logic High — +2.5V to +5.25V at 0.1 mA
Logic Low — 0.0V to +0.5V at 16.0 mA

Physical Characteristics

iSBC Interface Board

Height — 6.75 in. (17.15 cm)
Width — 12.0 in. (30.48 cm)
Thickness — 0.5 in. (1.27 cm)
Weight — 3 lbs. (1.4 Kg) (with interface cables)

Series 90 Memory System

Height — 5.21 in. (13.2 cm)
Width — 19.0 in. (48.25 cm)
Depth — 19.5 in. (49.53 cm)
Weight — 30 lbs. (13.5 Kg) max.
Mounting — table top

Electrical Characteristics

MULTIBUS Interface Board

+5V ± 6.0%
2 Amps typical
3 Amps worst case

Series 90 Memory System

115 VAC, 50/60 Hz, 2.8A
220 VAC, 50/60 Hz, 1.6A

Environmental Requirements

Ambient Operating Temperature — 0°C to 50°C
Relative Humidity — 10 to 90% without condensation

REFERENCE MANUAL

The iSBC 090 Memory System is supported by a full line of documentation, as listed below:

111710, Technical Manual for iSBC 090™ Memory System

111784, Technical Manual for Series 90 Random Access Memory Systems with CI-9000 Control Interface

111764, Technical Manual for Series 90, CM-90 Dynamic Memory Module

ORDERING INFORMATION

Model Number	Description
SBC-090-x1K	512K Byte Multibus-compatible dynamic RAM memory system with ECC. Expandable to 1024K Bytes using CM-90100-H22 memory module (below). "x" in model number must be specified per NOTE below to define input voltage and chassis configurations.
SBC-090-x1L	768K Byte memory system. Otherwise identical to SBC-090-x1K.
SBC-090-x1M	1024K Byte memory system. Not expandable. Otherwise identical to above SBC-090-x1K.
CM-90100-H22	256K Byte memory module for expansion of SBC-090-x1K or L.
SBC-090/556 Kit	768K Byte memory system—provides an enhanced compile environment when used with the Model 556 Functional Series III Upgrade Package for Intellec Series II/80, Series II/85 Microcomputer Development System (110V/60Hz or 220V/60Hz).
SBC-090/556I Kit	768K Byte memory system—provides an enhanced compile environment when used with the Model 556 Functional Series III Upgrade Package which consists of the Model 556 software and hardware performance package and the integrated 8085 processor board (IPC-85). This upgrade package is for Intellec Series II/80 Development System (110V/60Hz or 220V/50Hz).

NOTE

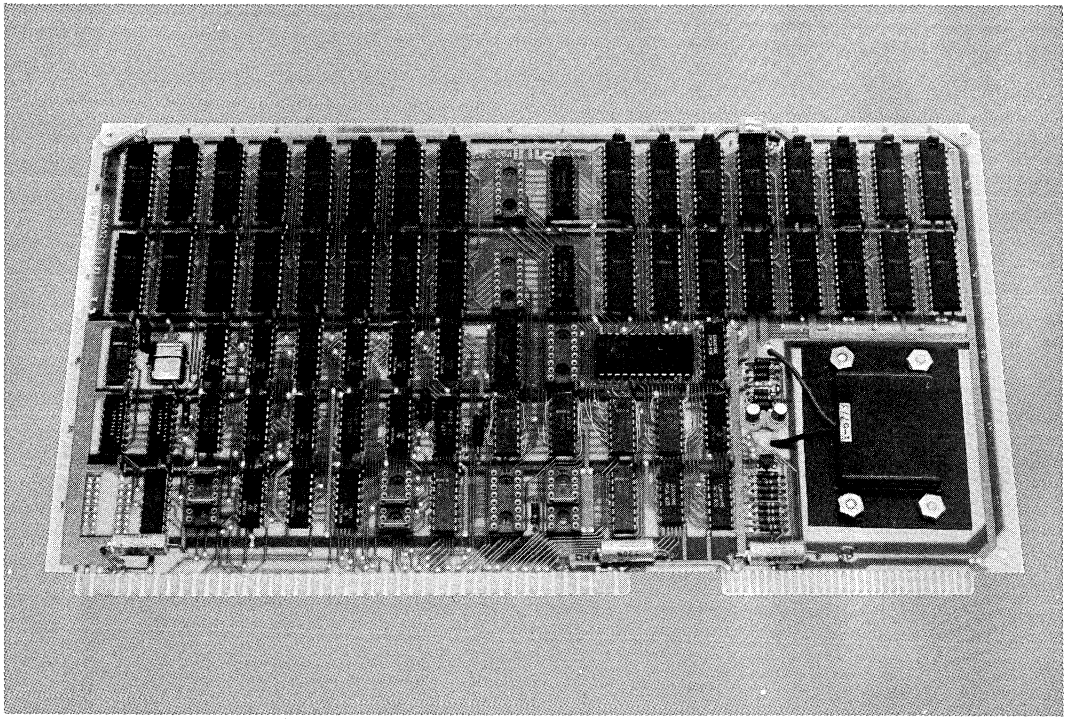
To order SBC 090, the "x" in model no. *must* be specified as shown below to define input voltage and chassis configuration.

x = 2 for table-top unit, 110 VAC
x = 3 for table-top unit, 220 VAC

iSBC 094
4K-BYTE CMOS RAM MEMORY BATTERY BACKUP BOARD

- **iSBC 80, iSBC 86 and iSBC 88 non-volatile RAM memory expansion through the MULTIBUS**
- **4K bytes of low power static CMOS RAM memory**
- **On-board power-fail interface logic**
- **Base address selectable to start on any 4K memory address boundary**
- **On-board rechargeable batteries and charging circuitry for 96-hour data retention**
- **Single +5V power requirement**

The iSBC 094 4K-Byte CMOS RAM Memory/Battery Backup Board is a member of Intel's complete line of iSBC memory and I/O expansion boards. The iSBC 094 interfaces directly to iSBC single board computer via the system bus to expand RAM memory capacity. The board contains 4K bytes of read/write memory, implemented using 32 Intel 5101 CMOS RAM memory components. On-board rechargeable batteries and charging circuitry insure that data contained in RAM will be retained for at least 96 hours after system bus power (+5V) is removed. Critical system parameters stored in the iSBC 094 RAM will thus be saved during temporary system power failures. Full power-fail interface logic is provided on the board to generate a CPU interrupt when system power fails. Orderly system shutdown procedures may then be executed and critical system parameters may be retrieved and stored. The use of CMOS RAM on the iSBC 094 also reduces power dissipation during normal system operation. The iSBC 094 contains jumpers for use in selecting a contiguous 4K-byte address segment beginning on any 4K memory address boundary (0000H, 1000H, 2000H, etc.). Read/write buffers reside on the board to buffer all data written into or read from the memory array. All address, data, and command signals on the bus interface are TTL compatible.



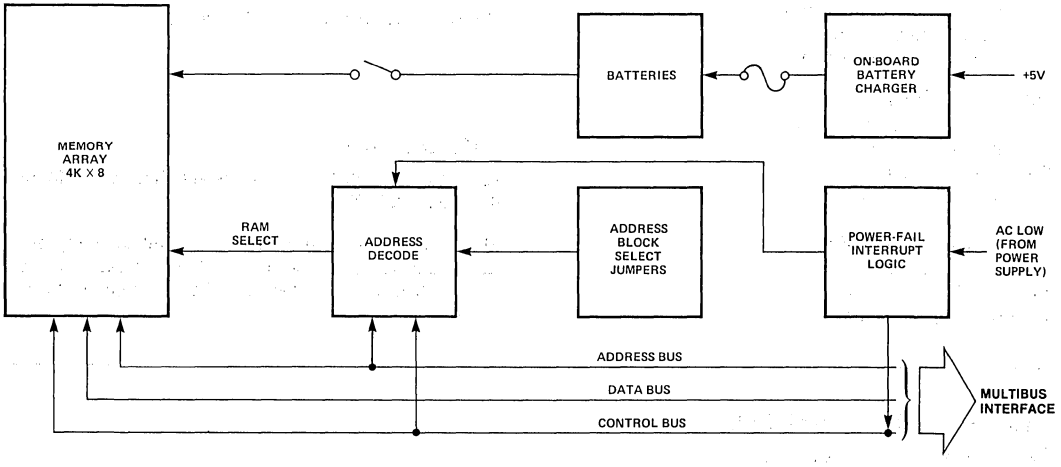


Figure 1. iSBC 094 Memory Backup Board Block Diagram

SPECIFICATIONS

Word Size
8 bits and 16 bits

Memory Size
4096 bytes

Memory Response Time

Operation	Access (ns, max)	Cycle (ns, max)
Read	750	900
Write	—	900

Interface
All address, data, and command signals are TTL compatible.

Power Fail Interrupt
Control logic is also included for generation of a power-fail interrupt to the MULTIBUS interface, which works in conjunction with the AC low signal from the Intel iSBC 635 Power Supply or equivalent.

Memory Protect
An on-board memory protect signal disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences. This signal is automatically asserted by the power-fail interface logic 3.6 ms after the AC low signal is received from the system power supply to signify that system power is beginning to fail.

Address Selection
4K segments starting at any jumper selectable base address on a 4K-byte boundary (e.g., 0000_H, 1000_H, ... $F000_H$). The memory will appear in every 64K-byte memory page.

Mating Connectors

Interface	Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	Viking 3KH43/9AMK12
Auxiliary[1]	60	0.1	AMP PE5-14559 or TI H311130

Note
1. Connector Dimensions vary from vendor to vendor. Review vendor specifications to ensure that connector heights and wire-wrap pin lengths to conform to your system packaging requirements.

Data Retention
96 hours minimum after +5V bus power is removed.

Battery Characteristics
Type — Nickel-Cadmium, rechargeable
Capacity — 150 mA hr
Voltage — 3.6V nominal

Battery Charger Characteristics

Charge Time
14 hours for full charge (150 mA hr)
Full overcharge protection
Full short-circuit protection

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.60 in. (1.27 cm)
Weight — 12 oz (340.5 gm)

Electrical Characteristics

Average DC Current
V_{CC} = +5V DC ±5%
I_{CC} = 0.8A typ, 1.7A max

Environmental Characteristics

Operating Temperature — 0 °C to 55 °C

Reference Manual

9800449B — iSBC 094 Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
SBC 094	4K-Byte CMOS RAM Memory Battery Backup Board

iSBC 108A/116A™ COMBINATION MEMORY AND I/O EXPANSION BOARDS

8K or 16K bytes of read/write memory (iSBC 108A™, iSBC 116A™ boards respectively).

Sockets for up to 32K bytes of EPROM.

Auxiliary power bus and memory protect control logic provided for battery backup RAM requirements.

RAM and EPROM assignable anywhere within a one megabyte address space.

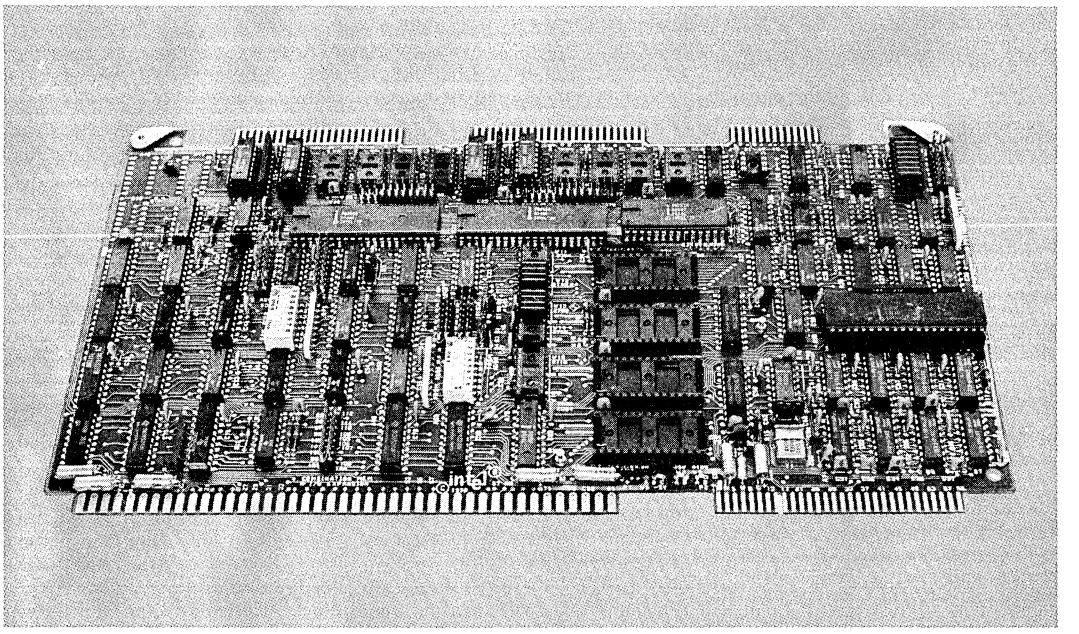
48 programmable I/O lines with sockets for interchangeable line drivers and terminators.

Synchronous/asynchronous communications interface with RS232C drivers and receivers.

Eight maskable interrupt request lines with a pending interrupt register.

1 msec interval timer.

The iSBC 108A and iSBC 116A Combination Memory and I/O Boards are members of Intel's complete line of iSBC memory and I/O expansion boards. Both boards interface directly with any iSBC 80, iSBC 86 or iSBC 88 single board computer via the MULTIBUS interface to expand RAM, EPROM, serial I/O and parallel I/O capacity. This mixture makes the iSBC 108A and 116A combination boards ideal for small microcomputer systems where the on-board resources of a single board computer are insufficient for incrementing the memory and I/O capacities of larger multiple board systems.



FUNCTIONAL DESCRIPTION

Memory Capabilities

The iSBC 108A board contains 8K bytes and the iSBC 116A board contains 16K bytes of RAM implemented with eight dynamic RAM components. An Intel 8202A dynamic RAM controller is used to provide all timing, control and refresh signals. Starting on a 4K byte boundary, RAM may be located anywhere in the MULTIBUS one megabyte memory address space.

Both combination boards contain four 28-pin sockets for adding up to 4K bytes (using Intel 2708 or 2758 EPROMs), 8K bytes (using Intel 2716 EPROMs), 16K bytes (using Intel 2732 or 2732A EPROMs), or 32K bytes (using Intel 2764 EPROMs) of non-volatile read-only-memory.

Parallel I/O Interface

Each combination board contains 48 programmable I/O lines implemented using two Intel 8255A programmable peripheral interfaces. The system software is used to configure the I/O lines in any combination of unidirectional input/output, and bi-directional ports indicated in Table 1. Therefore, the I/O interface may be customized to meet specified peripheral requirements. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 48 programmable I/O lines and signal ground lines are brought out to two 50-pin edge connectors that mate with flat, round, or woven cable.

Communications Interface

A programmable communications interface using Intel's 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on each board. A jumper selectable baud rate generator provides the USART with all common communications frequencies between 75 Hz and 38.4 kHz. The USART can be programmed by the system software to select the desired asynchronous or synchronous serial data transmission technique (including IBM Bi-Sync). The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and asynchronous serial transmission rate are all under program con-

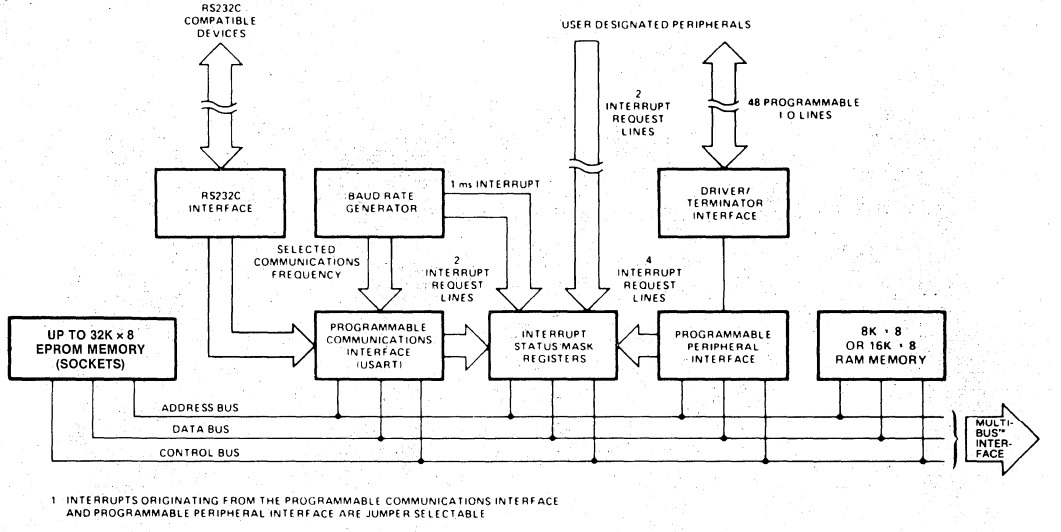


Figure 1. iSBC 108A/116A™ Combination Memory and I/O Expansion Board Block Diagram

trol. The 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The inclusion of a comprehensive RS232C interface on the boards in conjunction with the USART provides a direct interface to CRTs, RS232C compatible cassettes, and asynchronous and synchronous modems. The RS232C, serial data lines, and signal ground lines are brought out to a 26-pin edge connector which mates with RS232C compatible flat or round cables.

Interrupt Request Lines

Interrupt requests may originate from eight sources. Four jumper selectable interrupt requests can be automatically generated by the programmable peripheral interfaces when a byte of information is ready to be transferred to the CPU (i.e., input buffer is full) or a character has been transmitted (i.e., output data buffer is empty). Two jumper selectable interrupt requests can be automatically generated by the USART when a character is ready to be transferred to the CPU (i.e., receive buffer is full) or a character has been transmitted (transmit buffer is empty). Two interrupt request lines may be interfaced directly from user designated peripheral devices via the I/O edge connector. An on-board register contains the status of all eight interrupt request lines, and may be interrogated by the CPU. Each interrupt request line is maskable under program control. Routing for the eight interrupt request lines is jumper selectable. They may be ORed to provide a single interrupt request line for the iSBC 80/10A, or they may be individually provided to the MULTIBUS interface for use by the other iSBC single board computers.

Interval Timer

Each board contains a jumper selectable 1 ms interval timer. The timer is enabled by jumpering one of the interrupt request lines from the I/O edge connector to a 1 ms interval interrupt request signal originating from the baud rate generator.

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]
4	8	X	X	X	X	X	
5	8	X	X	X	X		
6	4	X		X			X[2]
	4	X		X			X[2]

Notes

1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output or port 1 is used as a bidirectional port.

2. Part of port 6 must be used as a control port when either port 4 or port 5 are used as a latched and strobed input or a latched and strobed output or port 4 is used as a bidirectional port.

SPECIFICATIONS

Memory Word Size
8 bits only. 16-bit single board computers may use this memory only for the storage of 8-bit data.

Memory Addressing
EPROM — Up to 4K, 8K, 16K or 32K bytes of read-only-memory may be located anywhere within a one megabyte address range. The base address must be located on a 4K byte boundary. EPROM addresses may not cross 32K byte boundaries.

RAM — 8K (iSBC 108A) or 16K (iSBC 116A) bytes of RAM may be located anywhere in a one megabyte address range. The base address must be located on a 4K byte boundary. RAM addresses may not cross 32K byte boundaries.

Memory Response Time

Memory	Access (ns)	Cycle (ns)
RAM	450 max*	580 max*
EPROM/ROM	450 max	635 max

* Without refresh contention.

I/O Transfer Rate
Parallel — Read or write acknowledge time 575 ns max.

Serial — (USART)

Frequency (kHz) (Jumper Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous (Program Selectable)	
		÷ 16	÷ 64
307.2	—	19200	4800
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
6.98	6980	—	110

Serial Communications Characteristics
Synchronous — 5 — 8 bit characters; internal or external character synchronization; automatic sync insertion.

Asynchronous — 5 — 8 bit characters; break characters generation; 1, 1½, or 2 stop bits; false start bit detectors.

Interrupts
Eight interrupt request lines may originate from the programmable peripheral interface (4 lines), the USART (2 lines) or user specified devices via the I/O edge connector (2 lines), or interval timer.

Interrupt Register Addresses
XX1 Interrupt mask register
XX0 Interrupt status register

Note
XX is any two hex digits assigned by jumper selection.

Timer Interval
1.003 ms ±0.1% when 110 baud rate is selected.
1.042 ms ±0.1% for all other baud rates.

Interfaces
Bus — All signals TTL compatible
Parallel I/O — All signals TTL compatible
Serial I/O — RS232C
Interrupt Requests — All TTL compatible

Connectors

Interface	No. of Pins	Centers (in.)	Mating Connectors
Bus (P1)	86	0.156	Viking 3KH43/9AMK12
Parallel I/O	50	0.1	3M 3415-000 or TI H312125
Serial I/O	26	0.1	3M 3462-000 or TI H312113
Aux. Power (P2)	60	0.1	AMP PE5-14559 or TI H311130

NOTE: Connector heights and wire-wrap pin lengths are not guaranteed to conform to Intel OEM packaging.

I/O Addressing

Port	1	2	3	4	5	6	8255A No. 1 Control	8255A No. 2 Control	USART Data	USART Control
Address	XX4	XX5	XX6	XX8	XX9	XXA	XX7	XXB	XXC or XXE	XXD or XXF

Note
XX is any two hex digits assigned by jumper selection.

Auxiliary Power
An auxiliary power bus is provided to allow separate power to RAM for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Memory Protect
An active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during system power-down sequences.

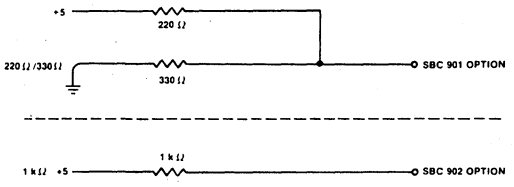
Line Drivers and Terminators

I/O Drivers — The following line drivers and terminators are all compatible with the I/O driver sockets on the iSBC 108A/116A board. Ports 1 and 4 have 25 mA totem-pole drivers and 1 kΩ terminators.

Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note
I = inverting; NI = non-inverting; OC = open collector.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pullup.



Bus Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-State	32
Commands	Tri-State	32

Physical Characteristics
Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 14 oz (397.3 gm)

Electrical Characteristics
Average DC Current

	V_{DD} = +12±5%	V_{CC} = +5±5%	V_{BB} = -5±5%	V_{AA} = -12±5%
No EPROM or Terminators	250 mA	2.9 A	—	70 mA
4 2708s and 8 Terminators	520 mA	3.6 A	180 mA	70 mA
4 2716s and No Terminators	250 mA	3.3 A	—	70 mA
4 2732s and No Terminators	250 mA	3.5 A	—	70 mA
Aux. Power RAM Accessed	175 mA	0.45 A	3 mA	—
Aux. Power No RAM Access	20 mA	0.45 A	3 mA	—

Environmental Characteristics
Operating Temperature — 0°C to +55°C.

Reference Manuals
9800862 — iSBC 108A/116A Board Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

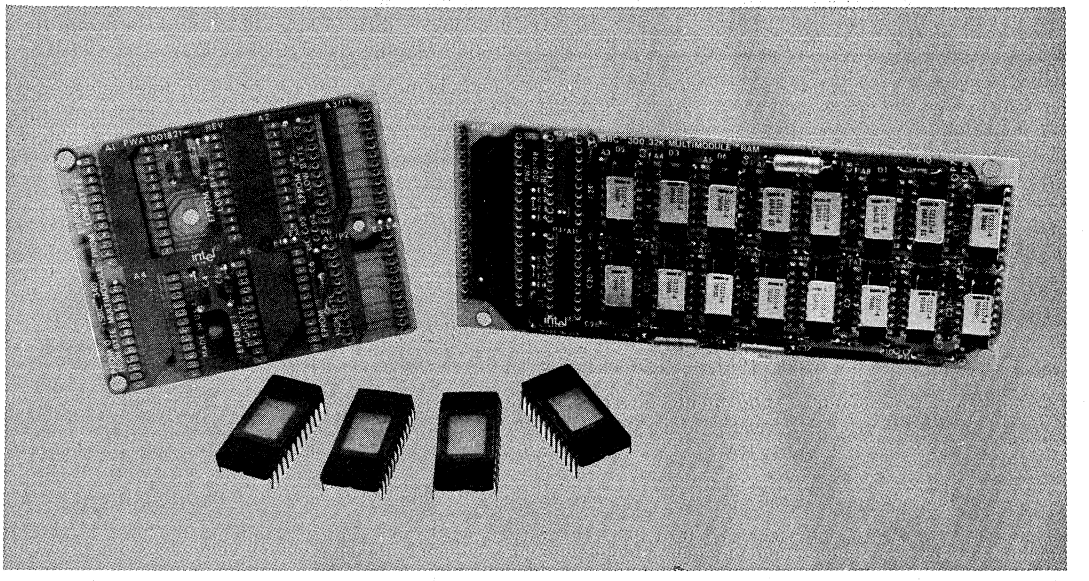
ORDERING INFORMATION

Part Number	Description
SBC 108A	Combination Memory and I/O Expansion Board with 8K bytes RAM
SBC 116A	Combination Memory and I/O Expansion Board with 16K bytes RAM

iSBC 300 or (pSBC 300*)
32K-BYTE RAM EXPANSION MODULE
iSBC 340 or (pSBC 340*)
16K-BYTE EPROM EXPANSION MODULE

- **On-board memory expansion for iSBC 86/12A Single Board Computer**
- **iSBC 300 module provides 32K bytes of dual port dynamic RAM and plugs directly into the iSBC 86/12A board**
- **iSBC 340 module provides sockets for up to 16K bytes of additional EPROM and plugs directly into the iSBC 86/12A board**
- **On-board memory expansion eliminates MULTIBUS system bus latency and increases system throughput**
- **Low power requirements**
- **Simple, reliable mechanical and electrical interconnection**

The iSBC 300 32K-byte RAM expansion module and the iSBC 340 16K-byte EPROM expansion module provide simple, low cost expansion of the memory complement available on the iSBC 86/12A single board computer. Each module utilized individually or together can double the iSBC 86/12A board's on-board RAM and EPROM memory capacity. The iSBC 300 32K-byte RAM expansion module and the iSBC 340 16K-byte EPROM expansion module options for the iSBC 86/12A board offer system designers a new level of flexibility in defining and implementing Intel® single board computer systems. These options allow the systems designer to double the memory complement of an iSBC 86/12A board with a minimum of system implications. Because they expand the memory configuration on-board, they can be accessed as quickly as the existing iSBC 86/12A memory by eliminating the need for accessing the additional memory via the MULTIBUS system bus. With the iSBC 86/12A board mounted in the top slot of an iSBC 604 or iSBC 614 cardcage, sufficient clearance exists for mounting both the iSBC 300 and/or the iSBC 340 expansion module option(s). If the iSBC 86/12A board is inserted into some other slot, the combination of boards will physically (but not electrically) occupy two cardcage slots. Incremental power required by the options is minimal; for instance, only 305 mW is needed for the iSBC 300 RAM expansion module.



*Same product, manufactured by Intel Puerto Rico, Inc.

FUNCTIONAL DESCRIPTION

iSBC 300 32K-Byte MULTIMODULE RAM

The iSBC 300 module contains sixteen 16K-byte dynamic RAM devices, sockets for the Intel® 8202A Dynamic computer. It expands the iSBC 86/12A board's on-board dual port RAM capacity from 32K bytes to 64K bytes. The iSBC 300 module contains sixteen 16K-byte dynamic RAM devices, sockets for the Intel® 8202 Dynamic RAM Controller and memory interface latching. To install the iSBC 300 module, the latches and controller from the iSBC 86/12A board are removed and inserted into the sockets on the iSBC 300 module. The add-on board is then mounted onto the iSBC 86/12A board. Pins extending from the controller's and latches' sockets mate with the devices' sockets underneath (see Figure 1). Additional pins mate to supply power and other signals to complete the electrical interface. The module is then secured at three additional points with nylon hardware to insure the mechanical security of the assembly.

To complete the installation, two socketed PROMs are replaced on the iSBC 86/12A board with those supplied with the iSBC 300 kit. These are the on-board memory and MULTIBUS address decode PROMs which allow the iSBC 86/12A board logic to recognize its expanded on-board memory complement.

iSBC 340 16K-byte MULTIMODULE EPROM

The iSBC 340 module expands the iSBC 86/12A Single Board Computer's on-board EPROM capacity from 16K bytes to 32K bytes. It measures 3.3" by 2.8" and consists of a PC board with six 24-pin special sockets. Two of the sockets have extended pins which mate with two of the EPROM sockets on the iSBC 86/12A board. Two of the EPROMs which would have been inserted on the iSBC 86/12A board are then reinserted in the iSBC 340 module. Additional pins also mate for bringing chip selects for the remaining EPROM devices (see Figure 2). The mechanical interface is similar to that used on the iSBC 300 RAM module and consists of two additional mounting holes and the necessary mounting hardware.

The iSBC 340 module supports Intel® 2732 EPROM. One section of the iSBC 86/12A on-board memory and MULTIBUS address decode PROMs (the same decode PROMs mentioned for the iSBC 300 module) is already preprogrammed to support the iSBC 340 module with Intel® 2732 EPROMs. This section is selected through the EPROM configuration switches on the iSBC 86/12A board. The iSBC 340 board can optionally be configured by the user to support Intel® 2758 or 2761 EPROMs by programming new iSBC 86/12A decode PROMs to support these devices. Necessary documentation and PROM map listings are in the iSBC 86/12A Hardware Reference Manual (order number 9803074-01).

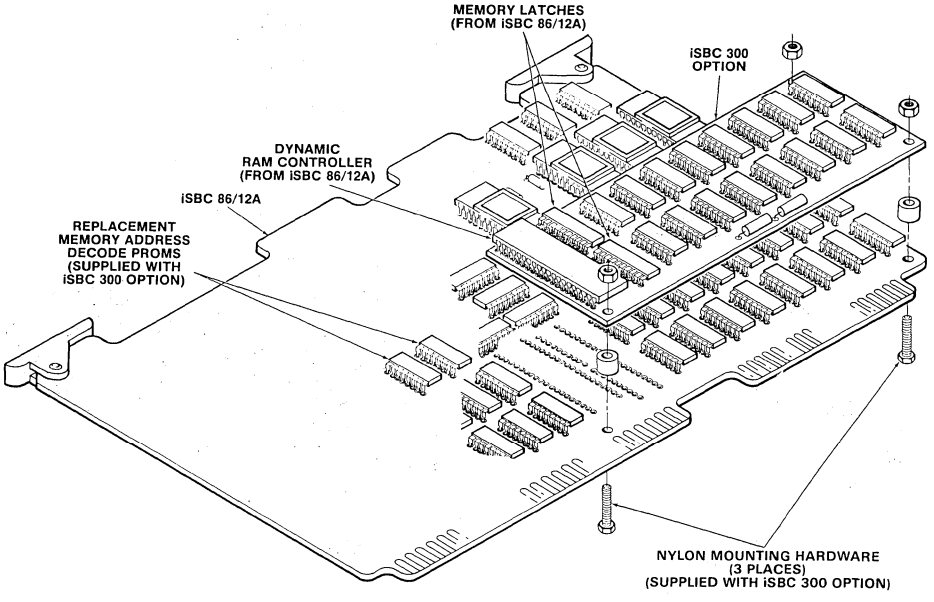


Figure 1. Installation of iSBC 300 MULTIMODULE RAM on iSBC 86/12A Single Board Computer

SPECIFICATIONS

Word Size
8 or 16 bits (16-bit data paths)

Memory Size
iSBC 300 Module — 32,768 bytes of RAM
iSBC 340 Module — 16,384 bytes (max) of EPROM

Access Time
iSBC 300 Module — Read: 1 μsec, write: 1.2 μsec
iSBC 340 Module — Standard EPROMs (450 nsec): 1 μsec, fast EPROMs (350 or 390 nsec): 800 nsec

Interface
The interface for the iSBC 300 and iSBC 340 module options is designed only for Intel's iSBC 86/12A Single Board Computer.

Memory Addressing
On-board RAM

CPU Access

iSBC 86/12A board only (32K bytes) — 00000-07FFFH.
iSBC 86/12A board + iSBC 300 module (64K bytes) — 00000-0FFFFH.
MULTIBUS Access — Jumper selectable for any 8K-byte boundary, but not crossing a 128K-byte boundary.

On-board EPROM

iSBC 86/12A board only (16K-bytes max.) — FF000-FFFFFH (using 2758 EPROMs); FE000-FFFFFH (using 2316E ROMs or 2716 EPROMs); and FC000-FFFFFH (using 2332A ROMs or 2732 EPROMs).

iSBC 86/12A board + iSBC 340 module (32K bytes max) — FE000-FFFFFH (using 2758 EPROMs); FC000-FFFFFH (using 2716 EPROMs); F8000-FFFFFH (using 2732 EPROMs).

On-board EPROM/ROM is not accessible via the MULTIBUS interface.

Auxiliary Power/Memory Protection
The low power memory protection option included on the iSBC 86/12A boards supports the iSBC 300 RAM module.

"Local Only" Memory Protection
The iSBC 86/12A Single Board Computer supports dedication of on-board RAM for on-board CPU access only in 8K, 16K, 24K, or 32K-byte segements. Installation of the iSBC 300 option allows protection of 16K, 32K, 48K, or 64K-byte segments.

Physical Characteristics

	iSBC 300	iSBC 340
Width	5.75"	3.3"
Length	2.35"	2.8"
Height of iSBC 86/12A plus mounted option	.718	.718*
Weight	13 oz.	5 oz.

*Includes EPROMs

All necessary mounting hardware (nylon, screws, spacers, nuts) are supplied with each kit.

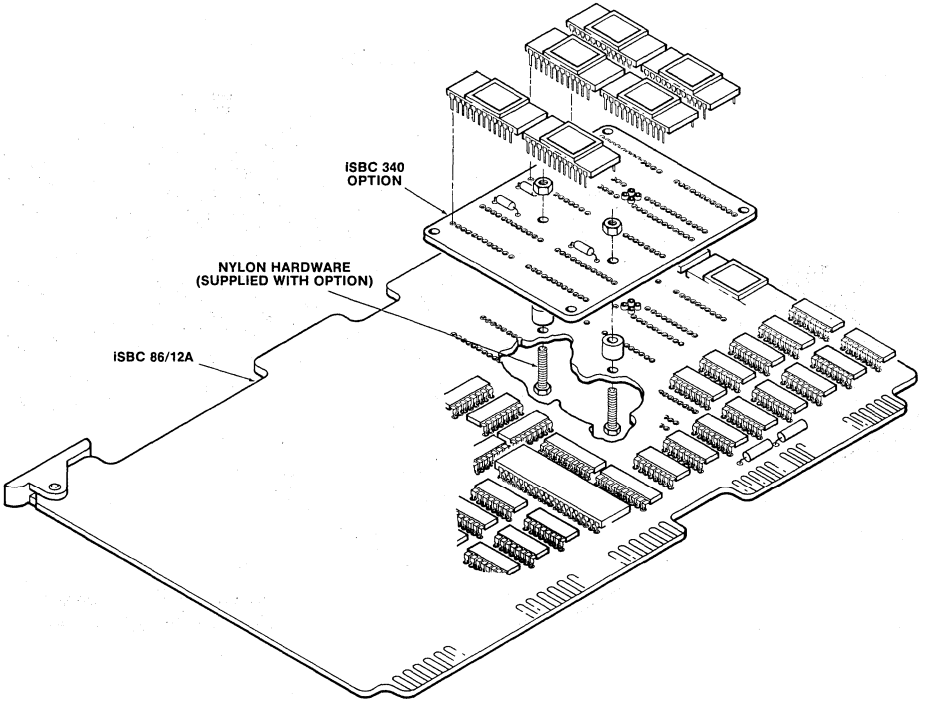


Figure 2. Installation of iSBC 340 MULTIMODULE EPROM Option on iSBC 86/12A Single Board Computer

Electrical Characteristics

DC power requirements:

Voltage	ISBC 300	ISBC 340
+5 ±5%	1 mA	120 mA[1]
+12 ±5%	24 mA	—
-12 ±5%	1 mA	—

Note:
1. Loaded with Intel 2732 EPROMs.

Environmental Characteristics

Operating Temperature — 0° to +55°C
Relative Humidity — to 90% (without condensation)

Reference Manuals

All necessary documentation for the iSBC 300 MULTIMODULE RAM and iSBC 340 MULTIMODULE EPROM/ROM is included in the iSBC 86/12A Hardware Reference Manual; order #9803074-01. (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051.

ORDERING INFORMATION

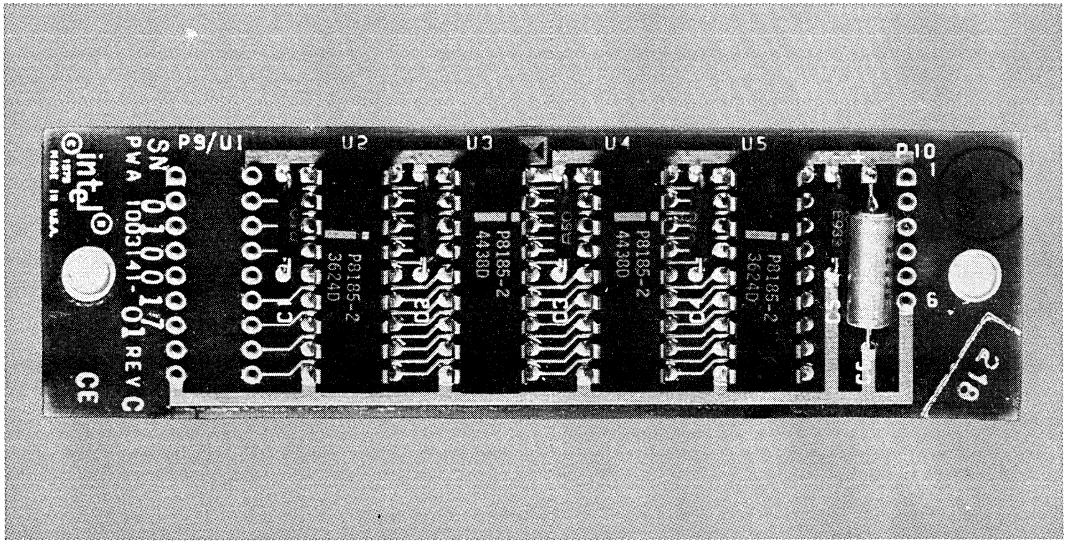
Part Number Description

SBC 300	32K byte MULTIMODULE RAM
SBC 340	16K byte MULTIMODULE EPROM

iSBC™ 301
4K-BYTE RAM
MULTIMODULE™ BOARD

- **On-board memory expansion to 8K bytes for iSBC™ 80/24 and iSBC™ 88/40 Single Board Computers**
- **Provides 4K bytes of static RAM directly on-board**
- **Uses 5 MHz (8185-2) RAMs**
- **Single +5V supply**
- **0.5 watts incremental power dissipation**
- **On-board memory expansion eliminates MULTIBUS system bus latency and increases system throughput**
- **Reliable mechanical and electrical interconnection**

The Intel iSBC 301 4K-Byte RAM MULTIMODULE Board provides simple, low cost expansion to double the RAM capacity on the iSBC 80/24 or iSBC 88/40 Single Board Computer to 8K bytes. This offers system designers a new level of flexibility in defining and implementing system memory requirements. Because memory is configured on-board, it can be accessed as quickly as the existing iSBC 80/24 or iSBC 88/40 memory, eliminating the need for accessing the additional memory via the MULTIBUS system bus. As a result, the iSBC 301 board provides a high speed, cost effective solution for systems requiring incremental RAM expansion. Incremental power required by the iSBC 301 module is minimal, dissipating only 0.5 watts.



FUNCTIONAL DESCRIPTION

The iSBC 301 board measures 3.95″ by 1.20″ and mounts above the RAM area on the iSBC 80/24 or iSBC 88/40 single board computer. It expands the on-board RAM capacity from 4K bytes to 8K bytes. The iSBC 301 MULTIMODULE board contains four 1K byte static RAM devices and a socket for one of the RAM devices on the iSBC 80/24 or iSBC 88/40 board. To install the iSBC 301 MULTIMODULE board, one of the RAMs is removed from the host board and inserted into the socket on the iSBC 301 board. The add-on board is then mounted into the vacated RAM socket on the host board. Pins extending from the RAM socket mate with the device's socket underneath (see Figure 1). Additional pins mate to the power supply and chip select lines to complete the electrical interface. The MULTIMODULE board is then secured at two additional points with nylon hardware to insure mechanical security of the assembly. With the iSBC 80/24 or iSBC 88/40 board mounted in the top slot of an iSBC 604 or iSBC 614 cardcage, sufficient clearance exists for mounting the iSBC 301 option. If the iSBC 80/24 or iSBC 88/40 board is inserted into some other slot, the combination of boards will physically (but not electrically) occupy two cardcage slots.

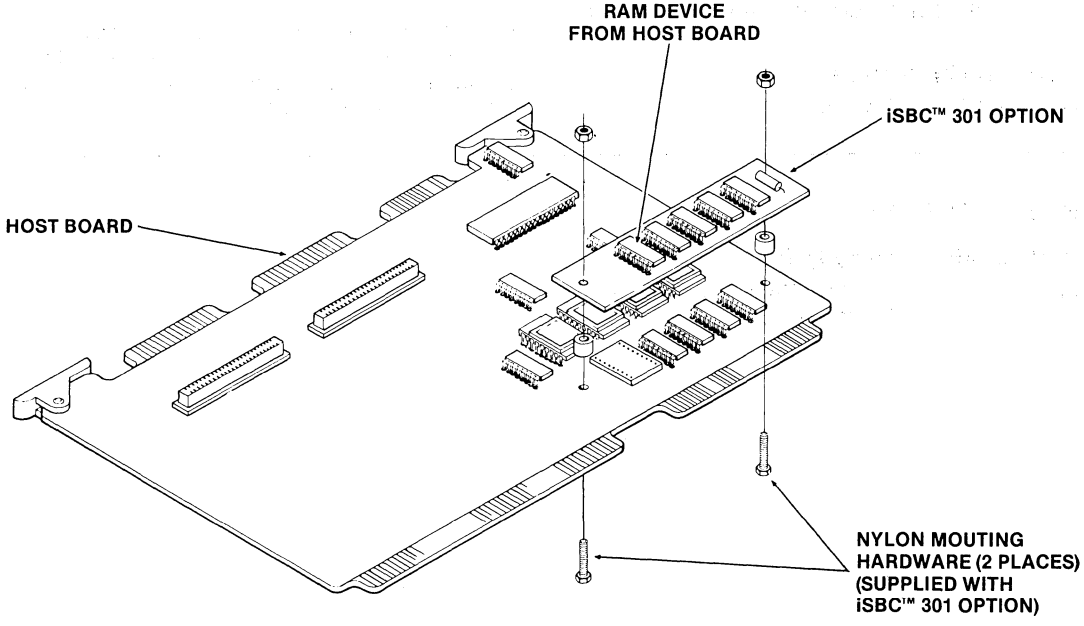


Figure 1. Installation of iSBC™ 301 4K-Byte RAM MULTIMODULE™ Board

SPECIFICATIONS

Word Size
8 bits

Memory Size
4096 bytes of RAM

Access Time
Read: 140 ns (from READ command)
200 ns (from ALE)
Write: 150 ns (from READ command)
190 ns (from ALE)

Memory Addressing
Memory addressing for the iSBC 301 4K-Byte RAM MULTIMODULE Board is controlled by the host board via the address and chip select signal lines and is contiguous with the host board RAM.

iSBC 80/24 and iSBC 301 board: 02000-02FFF
iSBC 88/40 and iSBC 301 board: 00000-01FFF

Physical Characteristics
Width — 1.20 in. (3.05 cm)
Length — 3.95 in. (10.03 cm)
Height — .44 in. (1.12 cm) iSBC 301 Board
.56 in. (1.42 cm)
iSBC 301 Board + host board
Weight — .69 oz. (19 gm)

Electrical Characteristics
DC Power Requirements:
10 mA at +5 Volts incremental power

Environmental Characteristics
Operating Temperature — 0° to +55° C
Relative Humidity — to 90% (without condensation)

Reference Manuals
All necessary documentation for the iSBC 301 MULTIMODULE board is included in the CPU board Hardware Reference Manual (NOT SUPPLIED)

iSBC 80/24 — Order No. 142648-001
iSBC 88/40 — Order No. 124978-001

Manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

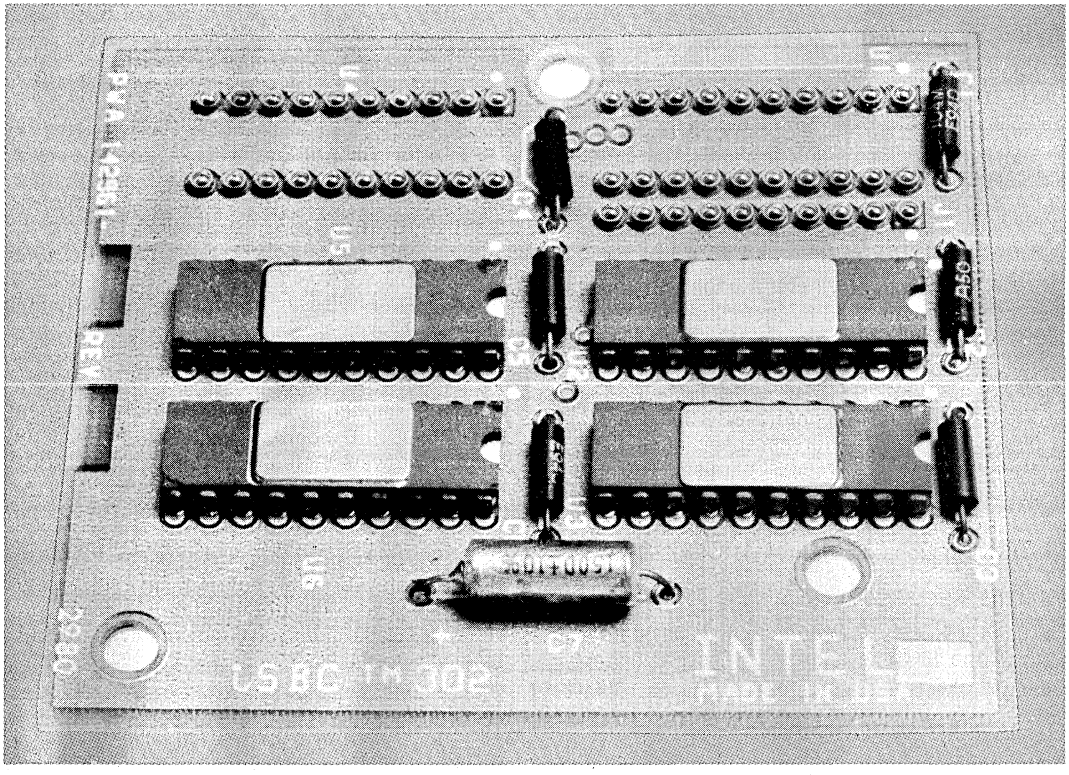
SPECIFICATIONS

Part Number	Description
SBC 301	4K Byte RAM MULTIMODULE Board

iSBC™ 302
8K-BYTE MULTIMODULE™ RAM

- **Expands on-board memory of the iSBC™ 86/05 and iSBC™ 88/25 Single Board Computers**
- **Uses four Intel® 2168 static RAMs**
- **Single + 5V supply**
- **On-board memory expansion eliminates system bus latency and increases system throughput**
- **Reliable mechanical and electrical interconnection**

The Intel iSBC 302 8K-Byte MULTIMODULE RAM provides simple, low-cost expansion to double the RAM capacity on the iSBC 86/05 Single Board Computer to 16K bytes or increase RAM capacity on the iSBC 88/25 Single Board Computer to 12K bytes. This offers system designers a new level of flexibility in implementing system memory. Because the MULTIMODULE memory is configured on-board, it can be accessed as quickly as the standard on-board iSBC 86/05 or iSBC 88/25 memory, eliminating the need for accessing additional memory via the MULTIBUS system bus. As a result, the iSBC 302 board provides a high-speed, cost-effective solution for systems requiring incremental RAM expansion.



FUNCTIONAL DESCRIPTION

The iSBC 302 board measures 2.60″ by 2.30″ and mounts above the RAM area on the iSBC 86/05 or iSBC 88/25 Single Board Computer. The iSBC 302 MULTIMODULE board contains four 4K × 4 static RAM devices and sockets for two of the RAM devices on the iSBC 86/05 board. With the iSBC 302 module mounted on the iSBC 88/25 board, the two sockets on the iSBC 302 module may be filled with 4K × 4 static RAMs. The two sockets on the iSBC 302 module have extended pins which mate with two sockets on the base board. Additional pins mate to the power supply and chip select lines to complete the electrical interface. The mechanical integrity of the assembly is assured with nylon hardware securing the module in two places. With the iSBC 86/05 or iSBC 88/25 board mounted in the top slot of an iSBC 604/614 cardcage, sufficient clearance exists for the mounted iSBC 302 option. If the iSBC 86/05 or iSBC 88/25 board is inserted into some other slot, the combination of boards will physically (but not electrically) occupy two cardcage slots.

SPECIFICATIONS

Word Size

8/16 bits

Memory Size

16,384 bytes of RAM

Cycle Time

Provides "no wait state" memory operations on the iSBC 86/05 board at 5 MHz or 8 MHz or the iSBC 88/25 board at 5 MHz.

5 MHz cycle time — 800 ns
8 MHz cycle time — 500 ns

Memory Addressing

Memory addressing for the iSBC 302 MULTIMODULE board is controlled by the host board via the address and chip select signal lines.

With the iSBC 86/05 board:

The 8K bytes of RAM on the iSBC 302 board occupy the 8K-byte address space immediately after that of the iSBC 86/05 board's 8K RAM (i.e., default configuration —

iSBC 86/05 board's RAM — $00000\text{-}01FFF_H$
iSBC 302 board's RAM — $02000\text{-}03FFF_H$).

With the iSBC 88/25 board:

The 8K bytes of RAM on the iSBC 302 board occupy the 8K byte address space immediately after that of the iSBC 88/25 board's 4K RAM (i.e., default configuration —

iSBC 88/25 board's RAM — $0\text{-}0FFF_H$
iSBC 302 board's RAM — $01000_H\text{-}02FFF_H$).

Physical Characteristics

WIDTH — 2.6 in. (6.60 cm)

LENGTH — 2.3 in. (5.84 cm)

HEIGHT — 0.56 in. (1.42 cm) iSBC 302 board + iSBC 86/05 or iSBC 88/25 board

WEIGHT — 1.25 oz (35 gm)

Electrical Characteristics

DC POWER REQUIREMENTS — 720 mA at +5V incremental power

Environmental Characteristics

OPERATING TEMPERATURE — 0°C to +55°C

RELATIVE HUMIDITY — to 90% (without condensation)

Reference Manuals

All necessary documentation for the iSBC 302 MULTIMODULE board is included in the CPU board Hardware Reference Manuals (NOT SUPPLIED).

iSBC 86/05 — Order No. 143153-001
iSBC 88/25 — Order No. 143825-001

Manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

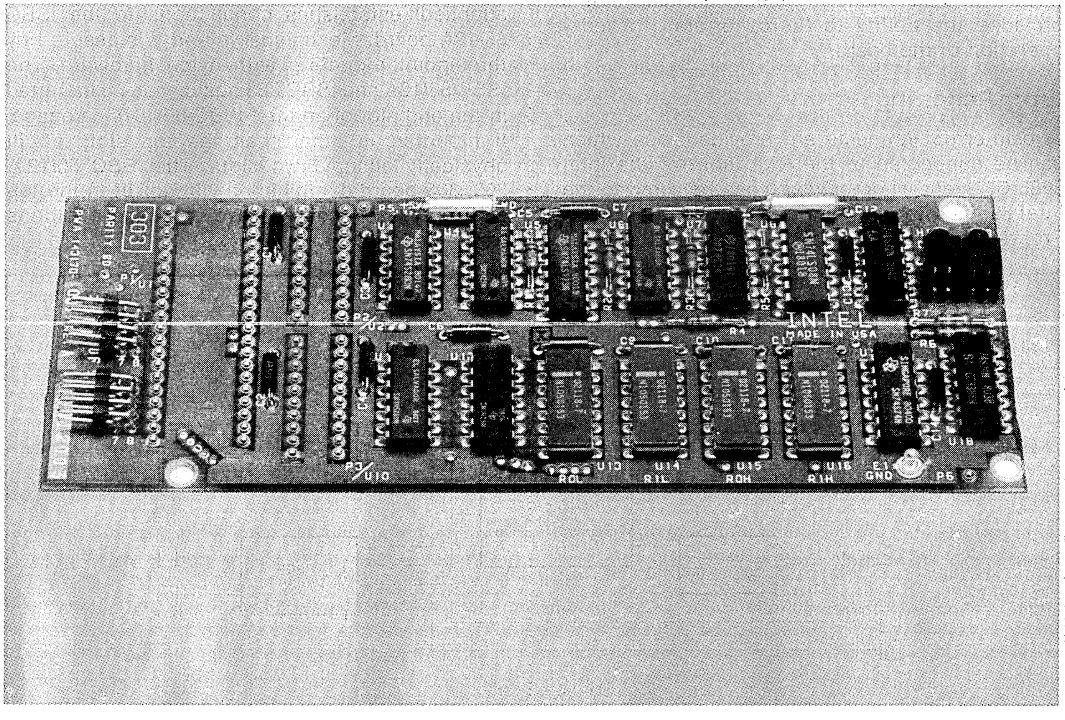
ORDERING INFORMATION

Part Number	Description
SBC 302	8K-Byte MULTIMODULE RAM

iSBC™ 303
MULTIMODULE™ PARITY

- **Add-on parity option for iSBC 86/12A Single Board Computer**
- **Supports 32KB or 64KB (with iSBC 300 MULTIMODULE RAM) on-board RAM**
- **Byte parity with programmable odd/even detection/generation**
- **Two LED error indicators**
- **Two interrupt requests for error reporting**
- **No degradation of memory performance**
- **Memory diagnostic capability**

The iSBC 303 MULTIMODULE Parity option provides on-board parity support for the on-board RAM of the iSBC 86/12A Single Board Computer. Memory parity generation/detection is invaluable for those applications in which execution of erroneous instructions or data must be prevented, as in critical process control, medical and financial transaction systems. When used on single board computers in conjunction with MULTIBUS®-based, parity-protected RAM expansion boards in large memory configurations, the iSBC 303 MULTIMODULE Parity option provides consistent protection throughout all system RAM.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE, ICS, iAPX and iMMX. 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 CORPORATION, 1981

FUNCTIONAL DESCRIPTION

The iSBC 303 MULTIMODULE Parity option is a complete subsystem, providing all necessary logic and display to support the iSBC 86/12A on-board RAM. Operation of the iSBC 303 option, once initialized, is transparent to the system, as the option causes no memory performance degradation. If an error is detected and interrupts are enabled, an interrupt request will be issued to the iSBC 86/12A board. Included on-board is the parity generator/checker and parity memory RAM, the interrupt request logic, error display, command register and the necessary interface for address, data and control (inputs) and interrupt requests (outputs) (See Figure 1). The parity generator/checker is a 74S280 MSI device. The memory devices are four Intel® 2118 dynamic RAMs. Parity is generated/detected on a byte basis; for a 32KB RAM configuration, only two of the parity RAMs are used. When the iSBC 303 board is used in conjunction with the iSBC 300 32KB MULTIMODULE RAM option (which provides a total of 64KB on-board RAM), all four parity RAMs are used. Odd or even parity may be programmatically selected through the command register. This feature also provides the ability to "force errors" to verify operation of the board.

Error Reporting

Two interrupt requests are provided: one which can be connected to the NMI input of the Intel 8086 CPU, and another which can be wired to the Intel 8259A interrupt controller and/or Intel 8255A Programmable Peripheral Interface on the iSBC 86/12A board. The 'NMI request' can be enabled/disabled via the command register or optionally with the NMIMASK signal which can be controlled by the 8255A PPI. Additionally, a power fail request signal originating from the system power supply can be "OR'ed" with the parity NMI request, allowing both signals to be issued to the 8086 CPU.

The error display logic contains two LEDs which indicate errors for high and low bytes. High and low byte error signals can also be connected to the 8255A, to read error status under program control. The LEDs and error signals are controlled by the command register.

Base-board Interface

The address, control and data signals are generated by the iSBC 86/12A RAM logic, and are connected to the iSBC 303 module through the sockets of the Intel 8202A RAM controller and the data latches (see Figure 2). The other I/O signals to/from the iSBC 303 board are brought out to an 8-pin connector from which a cable assembly (not supplied) establishes connections to the iSBC 86/12A board. Mechanical integrity is assured by three-point mounting with nylon hardware. The iSBC 86/12A and iSBC 303 boards can be mounted in the end slot of an Intel iSBC 604/614 cardcage. If mounted in other than the top slot, it will physically occupy two slots. The iSBC 86/12A, 300, 303 board combination will not fit in the top slot, and also occupies two cardcage positions when mounted.

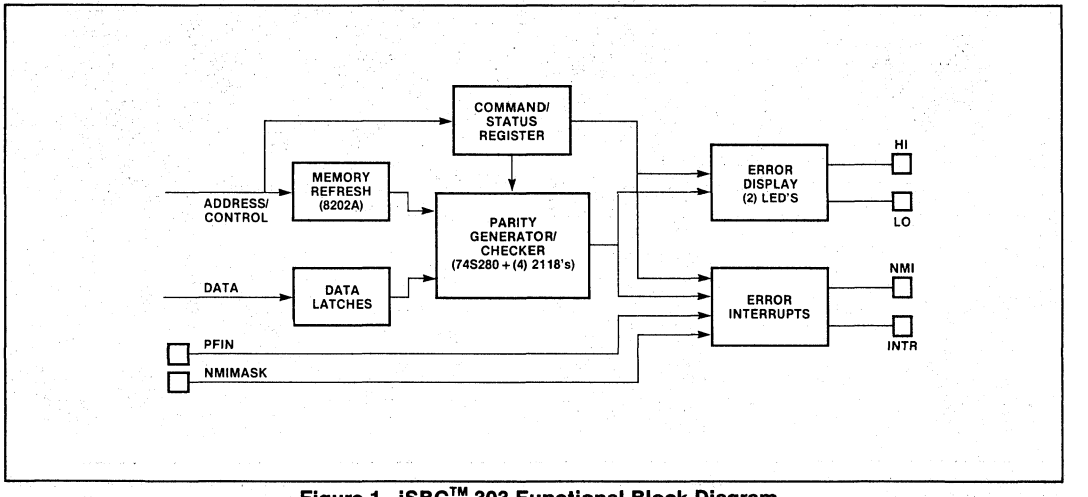


Figure 1. iSBC™ 303 Functional Block Diagram

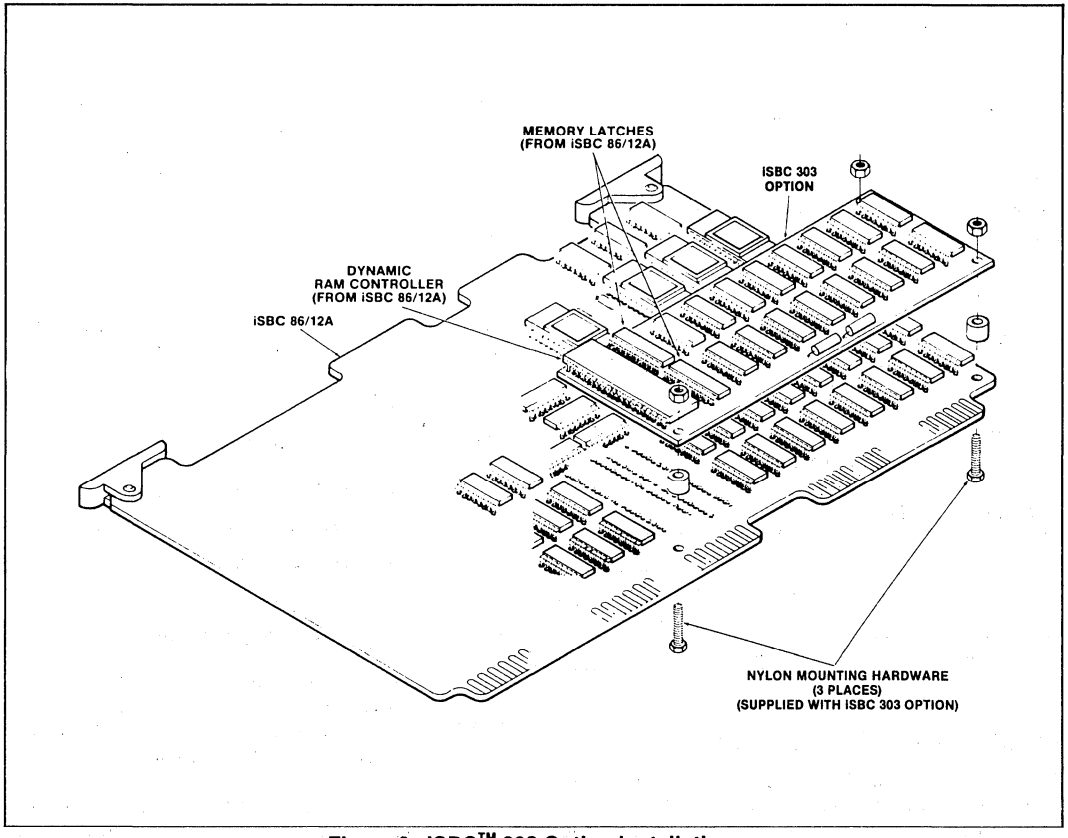


Figure 2. iSBC™ 303 Option Installation

Programmatic Control

The command register is a 4-bit, memory mapped register through which the operation of the iSBC 303 module is controlled (see Figure 3). Read/write access to all bits is available through either location 0H or 400H (jumper selectable). Initialization software should enable the NMI interrupt and the maskable interrupt and LED, as required. It should also select odd or even parity generation (odd is recommended for detection of catastrophic RAM failure).

Diagnostic Capability

Operation of the iSBC 303 module and the integrity of the RAM can be checked with a diagnostic software routine which can 'force errors' in parity memory. This is accomplished by writing data in one parity mode (e.g. odd), toggling the odd/even bit and then reading data with the complementary parity mode (e.g. even). These reads should cause detectable errors and interrupts will be generated, verifying the error detecting and reporting capability.

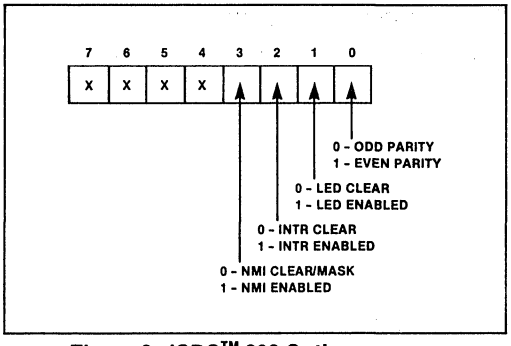


Figure 3. iSBC™ 303 Option Command/Status Register

SPECIFICATIONS

Word Size

Byte parity on 8 and 16-bit words.

Cycle Time

Supports iSBC 86/12A memory cycle times with no additional wait states.

Memory Capacity

Supports 32KB of iSBC 86/12A on-board RAM or 64KB with iSBC 300 MULTIMODULE RAM option.

Interface

All signals TTL compatible.

Connector (Not Supplied)

Shells	Pins	
	Reeled	Loose
AMP 87456-4	86491-3	87045-2
Berg 65043-033	47564	47744

Auxiliary Power/Memory Protection

The auxiliary power (i.e. battery backup operation) and memory protection (i.e. write disable on out-of-limits power supply voltages) features of the iSBC 86/12A Single Board Computer are supported on the iSBC 303 module.

Physical Characteristics

Width — 6.375 in. (16.2 cm)

Height — Height of iSBC 86/12A board + iSBC 303 module: 0.718 in. (1.82 cm)
Height of iSBC 86/12A board + iSBC 300 module + iSBC 303 module: 1.05 in. (2.66 cm)

Depth — 2.40 in. (6.1 cm)

Weight — 2.3 oz. (28.5 gm)

All necessary mounting hardware (screws, nuts, spacers) are supplied with each kit.

Electrical Characteristics

605 mA max., +5 VDC (±5%)

Environmental Characteristics

Operating Temperature — 0°C to +55°C

Relative Humidity — To 90% (without condensation)

Reference Manual (Not Supplied)

All necessary documentation for the iSBC 303 MULTIMODULE Parity option is contained in the iSBC 86/12A Single Board Computer Hardware Reference Manual; order #9803074-02.

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051.

ORDERING INFORMATION

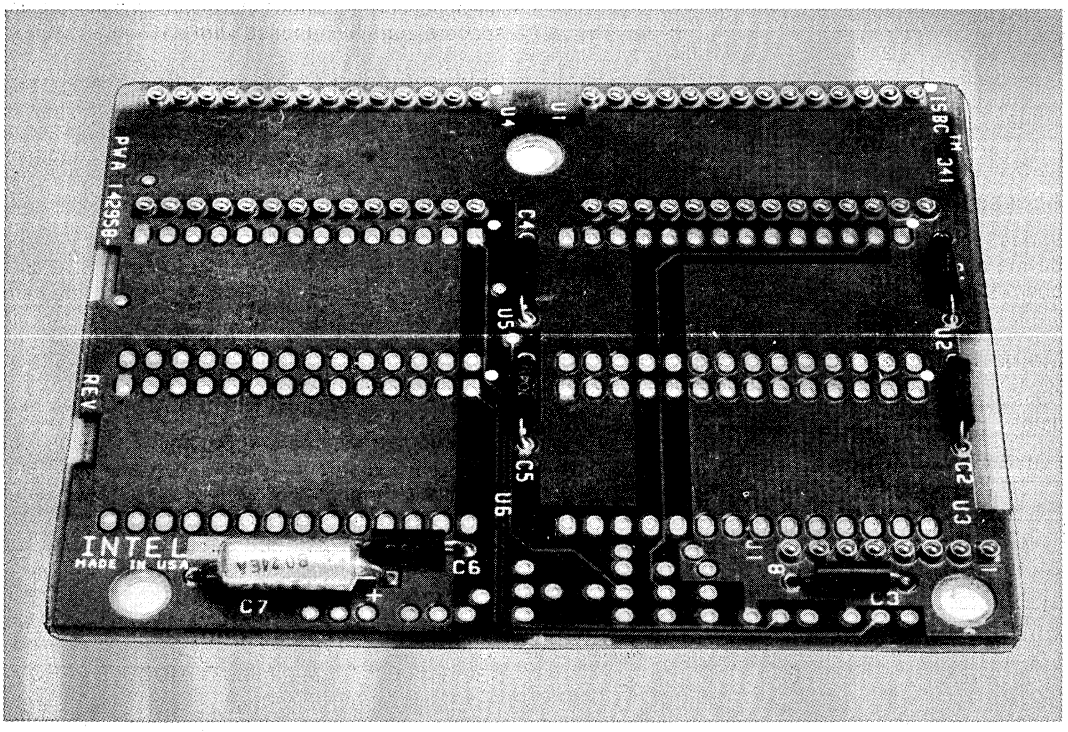
Part Number	Description
SBC 303	MULTIMODULE Parity

iSBC 341™
28-PIN MULTIMODULE EPROM

- **On-board memory expansion for iSBC 86/05, iSBC 88/25, and iSBC 88/40 microcomputers**
- **Supports JEDEC 24/28-pin standard memory devices, including EPROMs, byte-wide RAMs, and E²PROMs**
- **Sockets for up to 64K bytes of expansion with Intel® 27128 EPROMs**
- **On-board expansion provides "no wait state" memory access with selected devices**
- **Simple, reliable mechanical and electrical interface**

The iSBC 341 28-pin MULTIMODULE EPROM board provides simple, low-cost expansion of the on-board EPROM capacity of the iSBC 86/05 Single Board Computer, the iSBC 88/25 Single Board Computer and the iSBC 88/40 Measurement and Control Computer. Four additional 28-pin sockets support JEDEC 24/28-pin standard devices, including EPROMs, byte-wide static and pseudo-static RAMs.

The MULTIMODULE expansion concept provides the optimum mechanism for incremental memory expansion. Mounting directly on the microcomputer, the benefits include low cost, no additional power requirements beyond the memory devices, and higher performance than MULTIBUS-based memory expansion.



FUNCTIONAL DESCRIPTION

The iSBC 341 28-pin MULTIMODULE EPROM option effectively doubles the number of sockets available for EPROM on the base microcomputer board on which it is mounted. The iSBC 341 board contains six 28-pin sockets. Two of the sockets have extended pins which mate with two of the sockets on the base board. Two of the EPROMs which would have been inserted in the base board are then reinserted in the iSBC 341 sockets. Additional interface pins also connect chip select lines and power. The mechanical integrity of the assembly is assured with nylon hardware securing the unit in two places.

Through its unique interface, the iSBC 341 board can support 8 or 16-bit data paths. The data path width is determined by the base board — being 8 bits for the iSBC 88/40 and iSBC 88/25 microcomputers, and 8/16 bits for the iSBC 86/05 board.

SPECIFICATIONS

Word Size

8 or 8/16 bits (determined by data path width of base board).

Memory Size

32K bytes with available technology (JEDEC standard defines device pin-out to 128K-bit devices).

Device Size (Bytes)	EPROM Type	Max. iSBC 341 Capacity (Bytes)
2K x 8	2716	8K
4K x 8	2732	16K
8K x 8	2764	32K
16K x 8	27128	64K

Access Time

Varies according to base board and memory device access time. Consult data sheet of base board for details.

Memory Addressing

Consult data sheet of base board for addressing data.

POWER REQUIREMENTS

Devices[1]	Max. Current @ 5V ±5%
2716	420 mA
2732, 2732A	600 mA
2764	600 mA

NOTE:
1. Incremental power drawn from host board for four additional devices.

Auxiliary Power

There are no provisions for auxiliary power (battery backup) on the iSBC 341 option.

Physical Characteristics

WIDTH — 3.4 in. (8.64 cm)

LENGTH — 2.7 in. (6.86 cm)

HEIGHT — 0.78 in. (1.98 cm))*

WEIGHT — 5 oz (141.5 gm)

*Includes height of mounted memory devices and base board.

All necessary mounting hardware (nylon screws, spacers, nuts) is supplied with each kit.

Environmental Characteristics

OPERATING TEMPERATURE — 0°C to +55°C

RELATIVE HUMIDITY — to 90% (without condensation)

Reference Manuals

All necessary documentation for the iSBC 341 module is included in the CPU board Hardware Reference Manuals (NOT SUPPLIED)

iSBC 86/05 — Order No. 143153-001
iSBC 88/25 — Order No. 143825-001
iSBC 88/40 — Order No. 124978-001

Manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

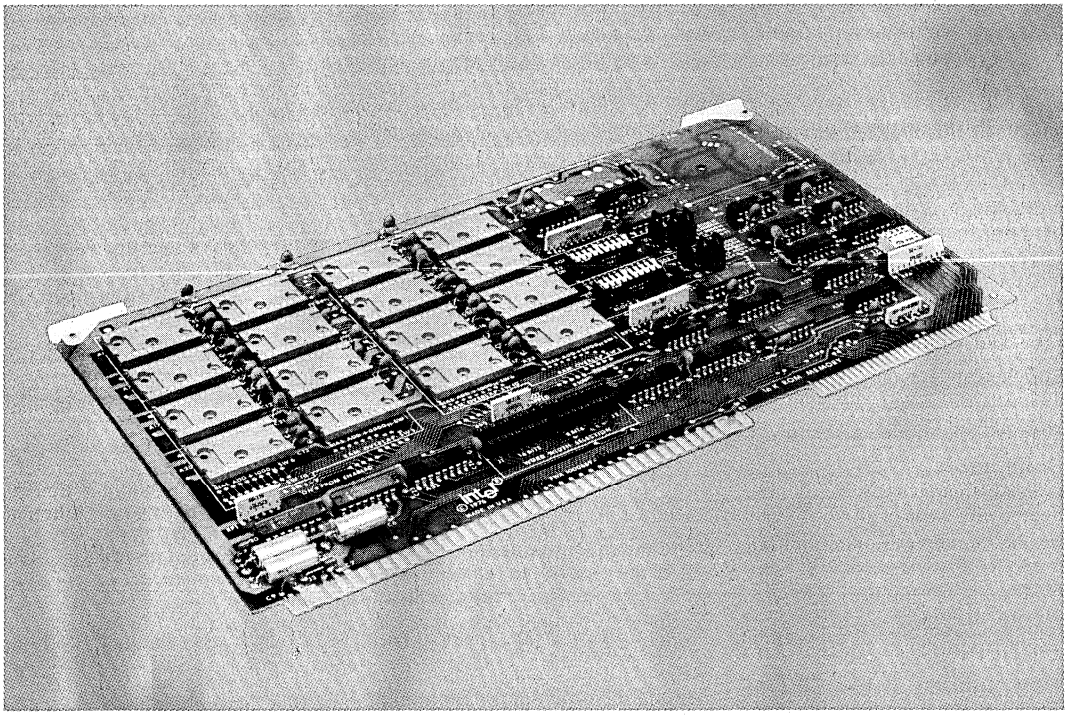
ORDERING INFORMATION

Part Number	Description
SBC 341	28-Pin MULTIMODULE EPROM

iSBC 416
16K EPROM EXPANSION BOARD

- **Allows iSBC 80 EPROM/ROM expansion through the MULTIBUS interface**
- **Sockets for up to 16K bytes of Intel 2708 programmable and erasable PROM**
- **Switches enable or disable each memory block**
- **Jumper selectable addresses for each 8K block**
- **Buffered address and data lines**

The iSBC 416 16K EPROM Expansion Board is a member of Intel's complete line of iSBC memory and I/O expansion boards. The iSBC 416 interfaces directly to any iSBC 80 single board computer via the system bus to expand EPROM memory capacity. The board contains 16 sockets that can house Intel 2708 programmable and erasable EPROMs. EPROM memory can be added in 1K-byte increments. The iSBC 416 contains a set of jumpers allowing the selection of the base address of independent 8K memory blocks, to begin on any 8K boundary. Switches are used to enable on-board memory in 1K block increments.



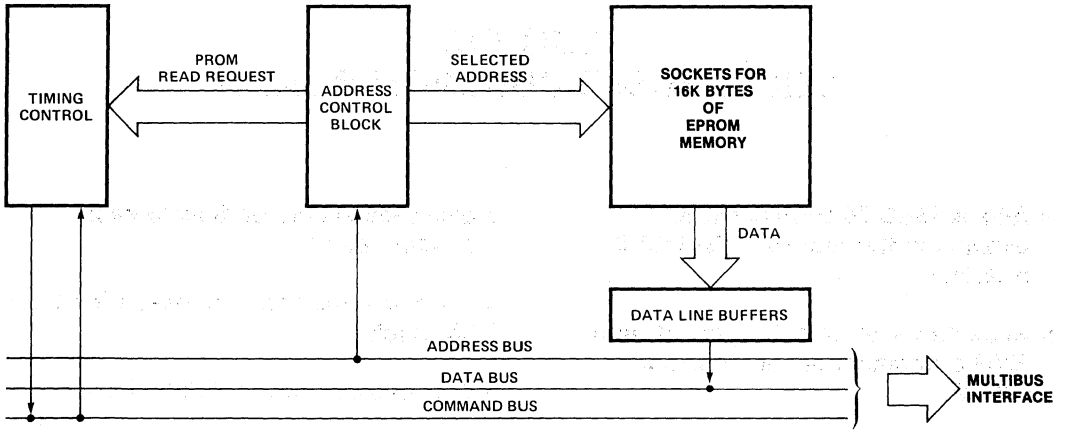


Figure 1. iSBC 416 PROM Expansion Board Block Diagram

SPECIFICATIONS

Word Size
8 bits

Memory Size
Sockets for up to 16K bytes. Memory may be added in 1K-byte increments.

Compatible Intel Memory
EPROM — 2708

Interface
All address, data, and command signals are TTL compatible and iSBC 80 bus compatible.

Address Selection
Switches and jumpers allowing the selection of a base address for each independent 8K block of memory, on any 8K boundaries

Connectors
Edge Connector — 86-pin double-sided PC edge connector with 0.156-in. (0.40 cm) contact centers.
Mating Connector — Viking 3KH43/9AMK12

Physical Characteristics
Width — 12.00 in. (30.40 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 12 oz (340.5 gm)

Electrical Characteristics
DC Power Requirements

	Without Memory	With 2608		With 2708	
		Typ	Max	Typ	Max
+5V	0.75A	0.77A	0.79A	0.85A	0.91A
−5V	—	0.001A	0.010A	0.48A	0.75A
+12V	—	0.58A	0.96A	0.80A	1.04A

Environmental Characteristics
Operating Temperature — 0°C to 55°C

Reference Manuals
9800265A — iSBC 416 Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

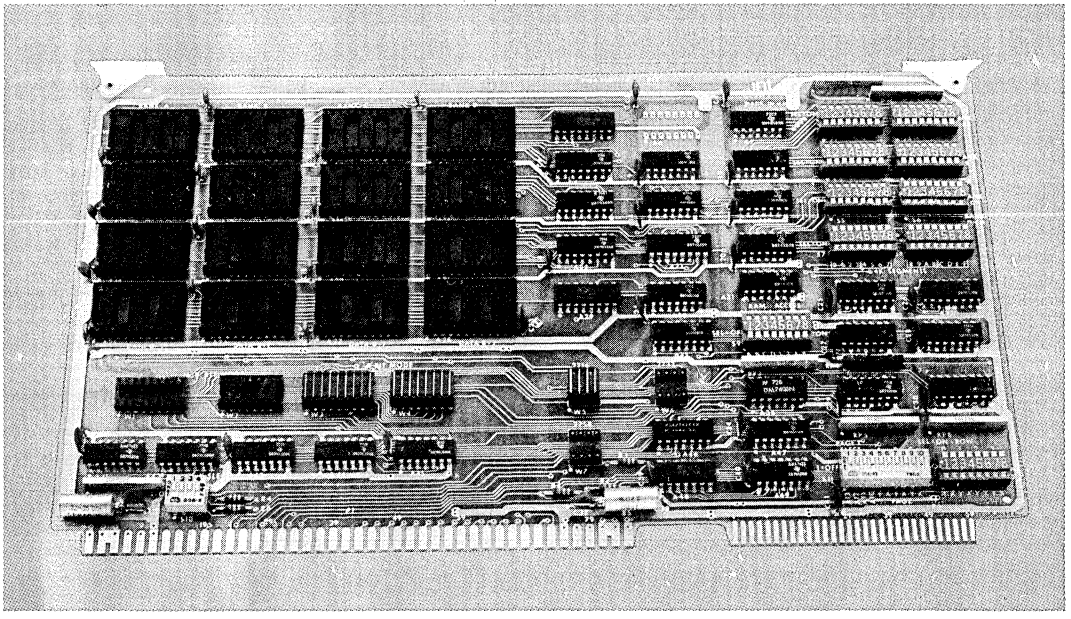
ORDERING INFORMATION

Part Number	Description
SBC 416	16K EPROM Expansion Board

iSBC™ 464
64K BYTE EPROM EXPANSION BOARD

- **Provides EPROM/ROM expansion of iSBC™ 80, iSBC™ 86 and iSBC™ 88 systems via direct MULTIBUS interface**
- **Sockets for up to 64K bytes of EPROM**
- **Compatible with Intel® 2758, 2716 or 2732/2732A erasable PROMs**
- **Switch selectable base address on 4K byte boundaries for each memory bank**
- **Assignable anywhere within a 1 megabyte address space**
- **EPROM components which are not enabled are placed in standby power mode**
- **Requires a single +5V power supply**

The iSBC 464 is a member of Intel's complete line of iSBC memory and I/O expansion boards. The iSBC 464 board interfaces directly to the iSBC 80, iSBC 86 or iSBC 88 single board computers via the MULTIBUS system bus, to expand system EPROM memory capacity.



FUNCTIONAL DESCRIPTION

Memory Configuration

The iSBC 464 board contains sixteen sockets which provide a maximum of 64K bytes of memory expansion. The actual capacity of the board is determined by the type and quantity of EPROM components installed by the user. The board is compatible with three different sizes of Intel EPROM devices. These are the 1K byte 2758 EPROM, the 2K byte 2716 EPROM, and the 4K byte 2732 EPROM.

Mode of Operation — The iSBC 464 board can operate in one of two modes: the 8 bit only mode or the 16/8 bit mode. The 8 bit mode provides the most efficient memory configuration for systems handling 8 bit data. The 16/8 bit mode allows 16 bit words to be accessed by 16 bit processors. In the 16/8 bit mode, 16 bit and 8 bit microprocessors may also access either the high order byte or the low order byte of a 16 bit word. The mode of operation is selected by placing two option jumper blocks in the appropriate sockets.

Memory Banks — When used in the 8 bit mode, the iSBC 464 board is organized into four banks (labeled A-D) of four sockets each. Depending on the type of memory components used, each bank may contain a maximum of 4K, 8K or 16K bytes of memory. Unused memory sockets may be deselected by bank or individually in bank D. Deselecting a bank or individual socket frees that address space for use elsewhere in the system. In the 16/8 bit mode, banks A & B and C & D are paired together to form two banks (labeled AB, CD) which are 16 bits wide. Each of these banks has four socket pairs. Bank AB may be deselected as a single unit. Socket pairs in bank CD may be deselected individually. Thus, board configurations using fewer than 16 memory components do not fill memory address space with unused sockets. Selection/deselection is accomplished by setting switches on the board.

Memory Access Time — The iSBC 464 board operates with one of 15 switch selectable memory access times ranging from 35 to 1435 nanoseconds. This feature allows the board to be tailored to the performance of the installed components and the system CPU.

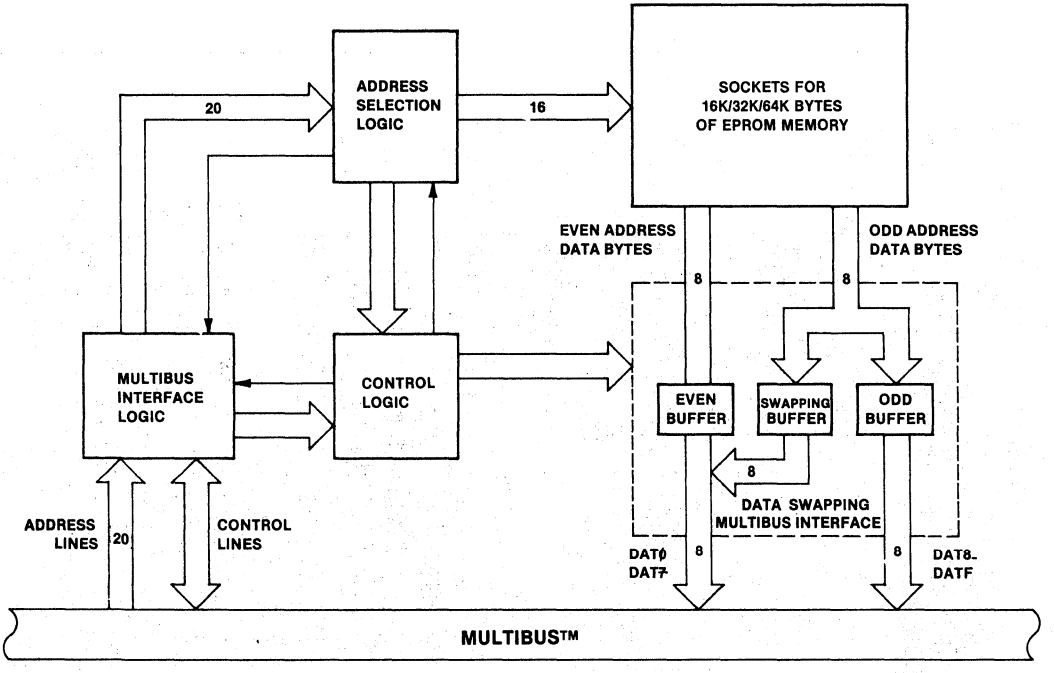


Figure 1. iSBC 464 Block Diagram

Memory Addresses

Switch selectable options on the iSBC 464 board allow the board to be assigned anywhere within a 1 megabyte address space. In either operating mode, the base address of each memory bank may be set to any 4K byte boundary within a 64K byte memory page. There is one exception. If the 4K byte devices are used in the 16/8 bit mode, then base addresses are restricted to 8K byte boundaries. If the board is used in a system with an address range greater than 64K bytes, memory on the iSBC 464 board may reside in one or two 64K byte memory pages. Any two pages out of a possible 16 may be chosen by setting switches on the board.

Standby Power Operation

The iSBC 464 board takes advantage of the standby modes of the Intel 2758, 2716 and 2732. When they are not enabled, these components draw as little as 25% of their active level power with no degradation in access time. The iSBC 464 board is designed so that only two memory components are enabled during a read operation.

RAM Overlap

Memory banks of the iSBC 464 board can be overlapped with the addresses of system RAM by setting on-board switches. The process of addressing a memory bank will drive the Inhibit RAM (INH1/) signal true. This signal is issued to the MULTIBUS system bus in order to prevent any MULTIBUS accessable RAM in the system from responding to the current address. If an EPROM is addressed which has its corresponding RAM overlap switch on, an access time of 15 clock cycles is imposed. This allows overlapped dynamic RAM to refresh before the address on the MULTIBUS is changed. The RAM overlap feature does not apply to RAM which is not on the MULTIBUS system bus.

SPECIFICATIONS

Word Size

8 bits or 16 and 8 bits

Memory Size

Sockets are provided for up to 16K bytes in 1K increments or 32K bytes in 2K increments or 64K bytes in 4K increments

Compatible Intel® Memory

EPROM — 2758 or 2716 or 2732

INTERFACE — All 20 address, 16 data, and 6 control signals are TTL compatible and Intel MULTIBUS compatible

Electrical Characteristics

DC Power (max)

V_{CC}: +5V DC ±5%
I_{CC}: 1.1 amps without EPROMs
I_{CC}: 1.6 amps with (16) 2716s or 2758s
I_{CC}: 1.3 amps with (16) 2732s or 2732As

Connectors

Bus — 86-pin double-sided PC edge connector with 0.40 cm (0.156 in.) contact centers

Mating Connector — Viking 3KH43/9AMK12 or compatible connector

Physical Characteristics

Length — 30.48 cm (12 in.)
Height — 17.15 cm (6.75 in.)
Depth — 1.27 cm (0.5 in.)
Weight — 294 gm (10.5 oz) without EPROM

Environment

Operating Temperature — 0°C to +55°C
Relative Humidity Limits — <90% non-condensing

Reference Manual

9800643A — iSBC 464 Memory Expansion Board Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
SBC 464	64K EPROM Expansion Board

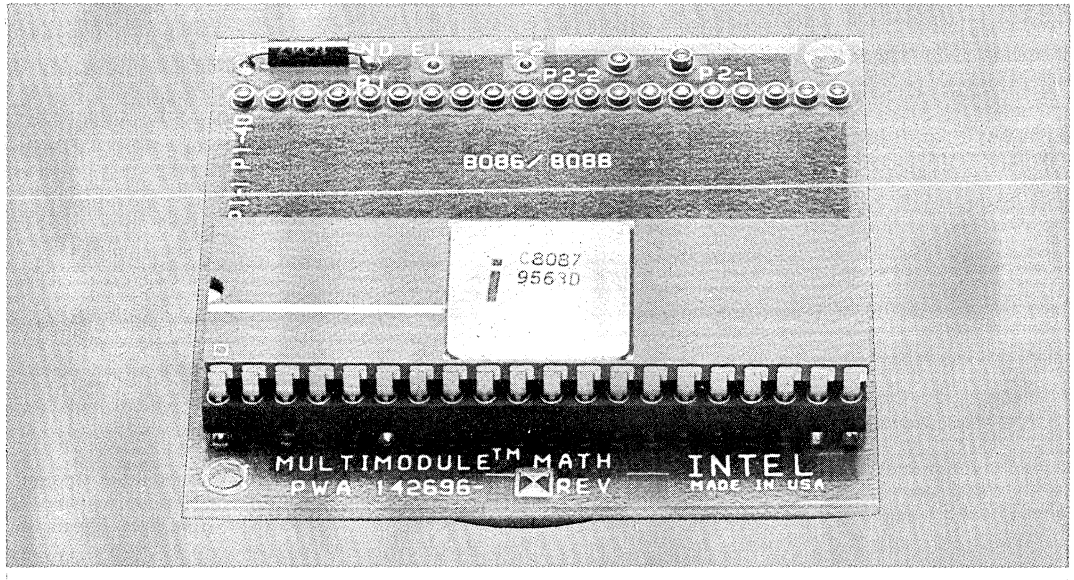
8

Digital I/O Expansion and Signal Conditioning Boards

iSBC™ 337 MULTIMODULE™ NUMERIC DATA PROCESSOR

- **High speed fixed and floating point functions for iSBC 86, 88 and iAPX 86, 88 systems**
- **MULTIMODULE option containing 8087 Numeric Data Processor**
- **Supports seven data types including single and double precision integer and floating point**
- **Implements proposed IEEE Floating Point Standard for high accuracy**
- **Extends host CPU instruction set with arithmetic, logarithmic, transcendental and trigonometric instructions**
- **50 × performance improvements in Whetstone benchmarks over iAPX 86/10 performance**
- **Software support through ASM-86/88 Assembly Language and High Level Languages**

The Intel iSBC 337 MULTIMODULE Numeric Data Processor offers high performance numerics support for iSBC 86 and iSBC 88 Single Board Computer users, for applications including simulation, instrument automation, graphics, signal processing and business systems. The coprocessor interface between the 8087 and the host CPU provides a simple means of extending the instruction set with over 60 additional numeric instructions supporting six additional data types. The data formats conform to the proposed IEEE Floating Point Standard insuring highly accurate results. The MULTIMODULE implementation allows the iSBC 337 module to be used on all iSBC 86 and iSBC 88 Microcomputers and can be added as an option to custom iAPX 86 and iAPX 88 board designs.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Index, Intel, MULTIBUS, RMX, iRMX, UPI, ICE, iSBC, iSBX, MULTIMODULE, iAPX and iCS. 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 CORPORATION, 1981

June, 1981
143087.01

OVERVIEW

The iSBC 337 MULTIMODULE Numeric Data Processor provides arithmetic and logical instruction extensions to the 8086 and 8088 CPU's of the iAPX 86 and iAPX 88 families, to provide iAPX 86/20 and iAPX 88/20 Numeric Data Processors. The instruction set consists of arithmetic, transcendental, logical, trigonometric and exponential instructions which can all operate on seven different data types. The data types are 16, 32, and 64 bit integer, 32 and 64 bit floating point, 18 digit packed BCD and 80 bit temporary.

Coprocessor Interface

The coprocessor interface between the host CPU (8086 and 8088) and the iSBC 337 processor provides easy to use and high performance math processing. Installation of the iSBC 337 processor is simply a matter of removing the host CPU from its socket, installing the iSBC 337 processor into the host's CPU socket, and reinstalling the host CPU chip into the socket provided for it on the iSBC 337 processor (see Figure 1). All synchronization and timing signals are provided via the coprocessor interface with the host CPU. The two processors also share a common address/data bus. (See Figure 2.) The 8087 Numeric Data Processor (NDP) component is capable of recognizing and executing 8087 numeric instructions as they are fetched by the host CPU. This interface allows concurrent processing by the host CPU and the 8087. It also allows 8087 and host CPU instructions to be intermixed in any fashion to provide the maximum overlapped operation and the highest aggregate performance.

High Performance and Accuracy

The 80-bit wide internal registers and data paths contribute significantly to high performance and

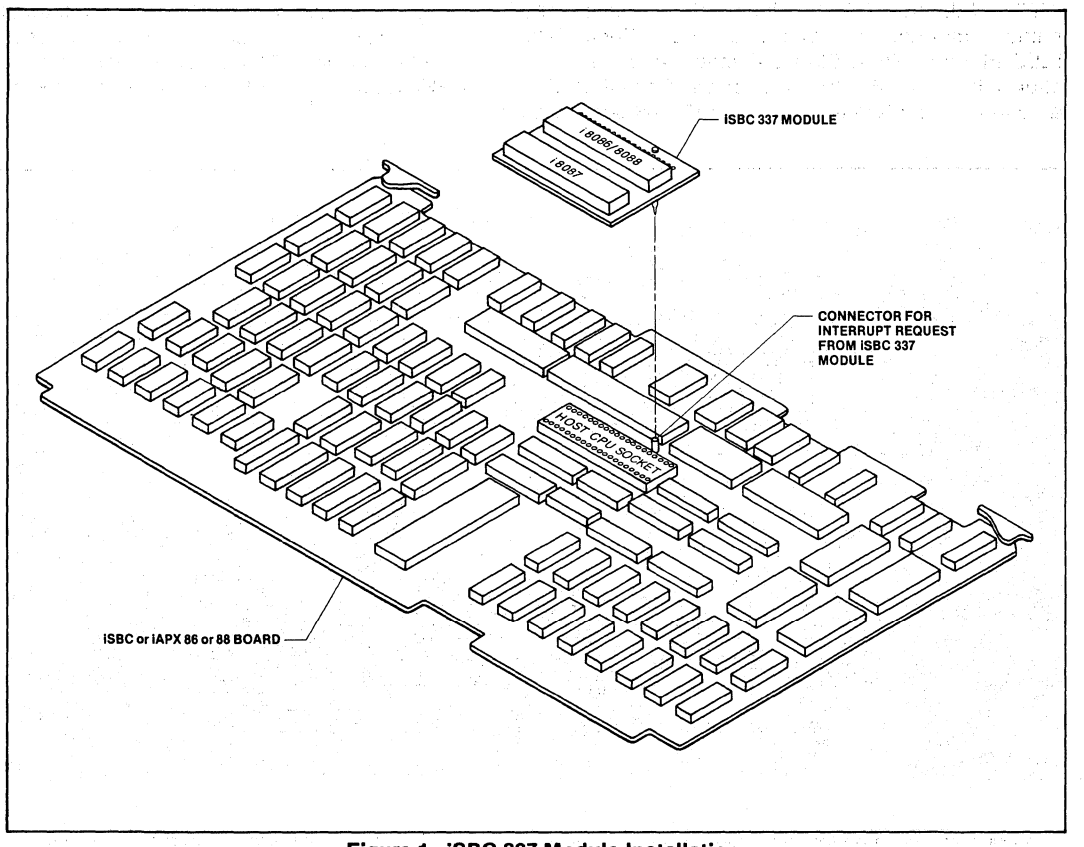


Figure 1. iSBC 337 Module Installation

minimizes the execution time difference between single and double precision floating point formats. This 80-bit architecture, in conjunction with the use of the proposed IEEE Floating Point Standard provides very high resolution and accuracy. This precision is complemented by extensive exception detection and handling. Six different types of exceptions can be reported and handled by the 8087. The user also has control over internal precision, infinity control and rounding control.

SYSTEM CONFIGURATION

As a coprocessor to an 8086 or 8088, the 8087 is wired in parallel with the CPU as shown in Figure 2. The CPU's status and queue status lines enable the NDP to monitor and decode instructions in synchronization with the CPU and without any CPU overhead. Once started, the 8087 can process in parallel with and independent of the host CPU. For resynchronization, the NDP's BUSY signal informs the CPU that the NDP is executing an instruction and the CPU WAIT instruction tests this signal to insure that the NDP is ready to execute subsequent instructions. The NDP can interrupt the CPU when it detects an error or exception. The interrupt request line is routed to the CPU through an 8259A Programmable Interrupt Controller. This interrupt request signal is brought down from the iSBC 337 module to the iSBC 86, 88 Single Board Computer through a single pin connector (see Figure 1). The signal is then routed to the interrupt matrix for jumper connection to the 8259A Interrupt Controller. Other iAPX 86 and 88 designs may use a similar arrangement, or by masking off the 8086's "READ" pin from the iSBC 337 socket, provisions are made to allow the now vacated pin of the host's CPU socket to be used to bring down

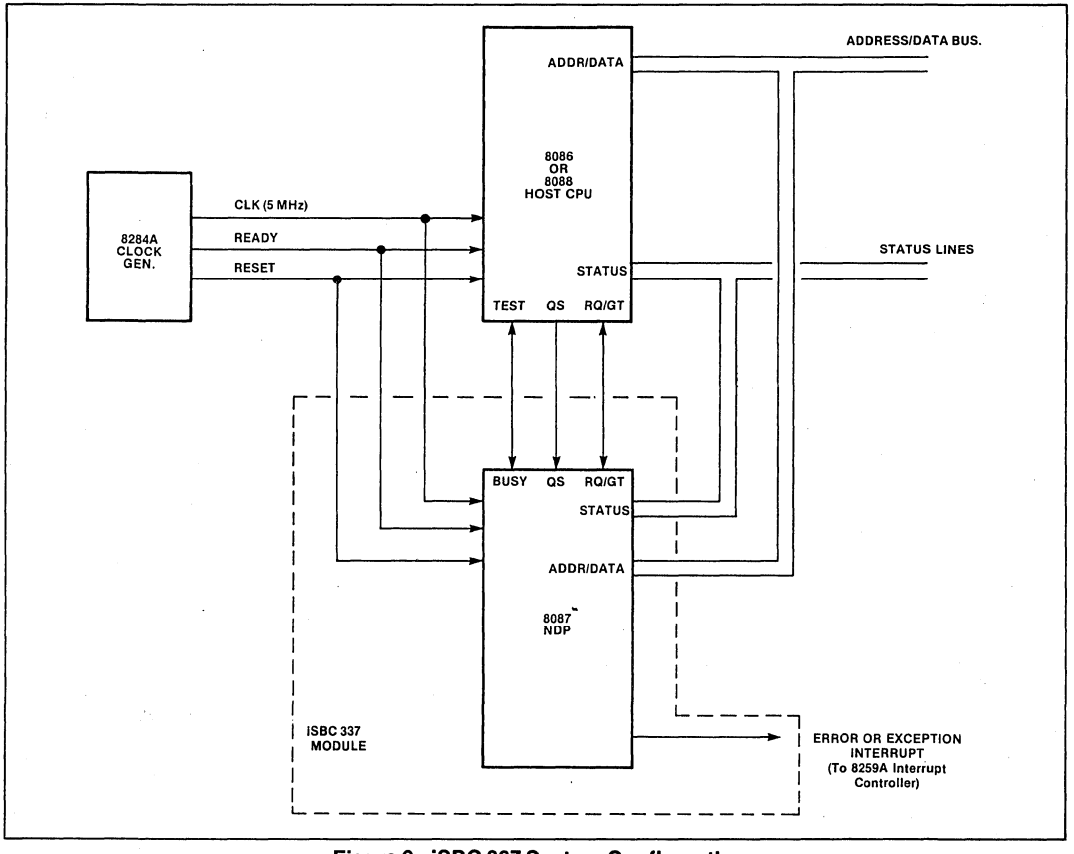


Figure 2. iSBC 337 System Configuration

Table 1. 8087 Datatypes

Data Formats	Range	Precision	Most Significant Byte (7 07 07 07 07 07 07 07 07 07 0)
Word Integer	10^4	16 Bits	I_{15} ... I_0 Two's Complement
Short Integer	10^9	32 Bits	I_{31} ... I_0 Two's Complement
Long Integer	10^{19}	64 Bits	I_{63} ... I_0 Two's Complement
Packed BCD	10^{18}	18 Digits	S — D_{17} D_{16} ... D_1 D_0
Short Real	$10^{\pm 38}$	24 Bits	S E_7 E_0 F_1 ... F_{23} F_0 Implicit
Long Real	$10^{\pm 308}$	53 Bits	S E_{10} E_0 F_1 ... F_{52} F_0 Implicit
Temporary Real	10^{+4932}	64 Bits	S E_{14} E_0 F_0 ... F_{63}

Note:

Integer: I **Sign: S** **Packed BCD:** $(-1)^S(D_{17}\ldots D_0)$ **Bias** = 127 for Short Real

Fraction: F **BCD Digit (4 Bits): D** **Real:** $(-1)^S(2^{E-BIAS})(F_0F_1\ldots)$ 1023 for Long Real

Exponent: E 16i383 for Temp Real

the interrupt request signal for connection to the base board and then to the 8259A. Another alternative is to use a wire to establish this connection.

PROGRAMMING INTERFACE

Table 1 lists the seven data types the 8087 supports and presents the format for each type. Internally, the 8087 holds all numbers in the temporary real format. Load and store instructions automatically convert operands represented in memory as 16-, 32-, or 64-bit integers, 32- or 64-bit floating point numbers or 18-digit packed BCD numbers into temporary real format and vice versa.

Computations in the 8087 use the processor's register stack. These eight 80-bit registers provide the equivalent capacity of 40 16-bit registers. The 8087 register set can be accessed as a stack, with instructions operating on the top stack element, or as a fixed register set, with instructions operating on explicitly designated registers.

Table 3 lists the 8087's instructions by class. Assembly language programs are written in ASM 86/88, the iAPX 86, 88 assembly language. Table 2 gives the execution times of some typical numeric instructions and their equivalent time on a 5 MHz 8086.

Table 2. Execution Time for Selected 8087 Actual and Emulated Instructions

Floating Point Instruction	Approximate Execution Time (microseconds)	
	8087 (5 MHz Clock)	8086 Emulation
Add/Subtract Magnitude	14/18	1,600
Multiply (single precision)	19	1,600
Multiply (extended precision)	27	2,100
Divide	39	3,200
Compare	9	1,300
Load (double precision)	10	1,700
Store (double precision)	21	1,200
Square Root	36	19,600
Tangent	90	13,000
Exponentiation	100	17.100

FUNCTIONAL DESCRIPTION

The NDP is internally divided into two processing elements, the control unit (CU) and the numeric execution unit (NEU), providing concurrent operation of the two units. The NEU executes all numeric instructions, while the CU receives and decodes instructions, reads and writes memory operands and executes processor control instructions.

Control Unit

The CU keeps the 8087 operating in synchronization with its host CPU. 8087 instructions are intermixed with CPU instructions in a single instruc-

Table 3. 8087 Instruction Set

Data Transfer Instructions

Real Transfers	
FLD	Load real
FST	Store real
FSTP	Store real and pop
FXCH	Exchange registers
Integer Transfers	
FILD	Integer load
FIST	Integer store
FISTP	Integer store and pop
Packed Decimal Transfers	
FBLD	Packed decimal (BCD) load
FBSTP	Packed decimal (BCD) store and pop

Comparison Instructions

FCOM	Compare real
FCOMP	Compare real and pop
FCOMPP	Compare real and pop twice
FICOM	Integer compare
FICOMP	Integer compare and pop
FTST	Test
FXAM	Examine

Transcendental Instructions

FPTAN	Partial tangent
FPATAN	Partial arctangent
F2XM1	2^X-1
FYL2X	$Y \bullet \log_2 X$
FYL2XP1	$Y \bullet \log_2 (X+1)$

Arithmetic Instructions

Addition	
FADD	Add real
FADDP	Add real and pop
FIADD	Integer add
Subtraction	
FSUB	Subtract real
FSUBP	Subtract real and pop
FISUB	Integer subtract
FSUBR	Subtract real reversed
FSUBRP	Subtract real reversed and pop
FISUBR	Integer subtract reversed
Multiplication	
FMUL	Multiply real
FMULP	Multiply real and pop
FIMUL	Integer multiply
Division	
FDIV	Divide real
FDIVP	Divide real and pop
FIDIV	Integer divide
FDIVR	Divide real reversed
FDIVRP	Divide real reversed and pop
FIDIVR	Integer divide reversed
Other Operations	
FSQRT	Square root
FSCALE	Scale
FPREM	Partial reminder
FRNDINT	Round to integer
FXTRACT	Extract exponent and significand
FABS	Absolute value
FCHS	Change sign

Processor Control Instructions

FINIT/FNINIT	Initialize processor
FDISI/FNDISI	Disable interrupts
FENI/FNENI	Enable interrupts
FLDCW	Load control word
FSTCW/FNSTCW	Store control word
FSTSW/FNSTSW	Store status word
FCLEX/FNCLEX	Clear exceptions
FSTENV/FNSTENV	Store environment
FLDENV	Load environment
FSAVE/FNSAVE	Save state
FRSTOR	Restore state
FINCSTP	Increment stack pointer
FDECSTP	Decrement stack pointer
FFREE	Free register
FNOP	No operation
FWAIT	CPU wait

tion stream. The CPU fetches all instructions from memory; by monitoring the status signals emitted by the CPU, the NDP control unit determines when an 8086 instruction is being fetched. The CU taps the bus in parallel with the CPU and obtains that portion of the data stream.

After decoding the instruction, the host executes all opcodes but ESCAPE (ESC), while the 8087 executes only the ESCAPE class instructions. (The first five bits of all ESCAPE instructions are identical). The CPU does provide addressing for ESC instructions, however.

An 8087 instruction either will not reference memory, will require loading one or more operands from memory into the 8087, or will require storing one or more operands from the 8087 into memory. In the first case a non-memory reference escape is used to start 8087 operation. In the last two cases, the CU makes use of a "dummy read" cycle initiated by the CPU, in which the CPU calculates the operand address and initiates a bus cycle, but does not capture the data. Instead, the CPU captures and saves the address which the CPU places on the bus. If the instruction is a load, the CU additionally captures the data word when it becomes available on the local data bus. If data required is longer than one word, the CU immediately obtains the bus from the CPU using the request/grant protocol and reads the rest of the information in consecutive bus cycles. In a store operation, the CU captures and saves the store address as in a load, and ignores the data word that follows in the "dummy read" cycle. When the 8087 is ready to perform the store, the CU obtains the bus from the CPU and writes the operand starting at the specified address.

Numeric Execution Unit

The NEU executes all instructions that involve the register stack; these include arithmetic, logical, transcendental, constant and data transfer instructions. The data path in the NEU is 80 bits wide (64 fraction bits, 15 exponent bits and a sign bit) which allows internal operand transfers to be performed at very high speeds.

When the NEU begins executing an instruction, it activates the 8087 BUSY signal. This signal is

used in conjunction with the CPU WAIT instruction to resynchronize both processors when the NEU has completed its current instruction.

Register Set

The 8087 register set is shown in Figure 3. Each of the eight data registers in the 8087's register stack is 80 bits wide and is divided into "fields" corresponding to the NDP's temporary real data type.

The register set may be addressed as a push down stack, through a top of stack pointer or any register may be addressed explicitly relative to the top of stack.

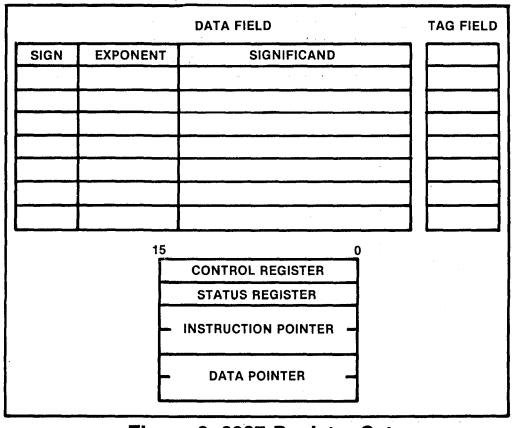


Figure 3. 8087 Register Set

Status Word

The status word shown in Figure 4 reflects the overall state of the 8087; it may be stored in memory and then inspected by CPU code. The status word is a 16-bit register divided into fields as shown in Figure 4. The busy bit (bit 15) indicates whether the *NEU* is executing an instruction (B = 1) or is idle (B = 0). Several instructions which store and manipulate the status word are executed exclusively by the CU, and these do not set the busy bit themselves.

The four numeric condition code bits (C_0-C_3) are similar to the flags in a CPU: various instructions update these bits to reflect the outcome of NDP operations.

Bits 14-12 of the status word point to the 8087 register that is the current top-of-stack (TOP).

Bit 7 is the interrupt request bit. This bit is set if any unmasked exception bit is set and cleared otherwise.

Bits 5-0 are set to indicate that the NEU has detected an exception while executing an instruction.

Tag Word

The tag word marks the content of each register as shown in Figure 5. The principal function of the tag word is to optimize the NDP's performance. The tag word can be used, however, to interpret the contents of 8087 registers.

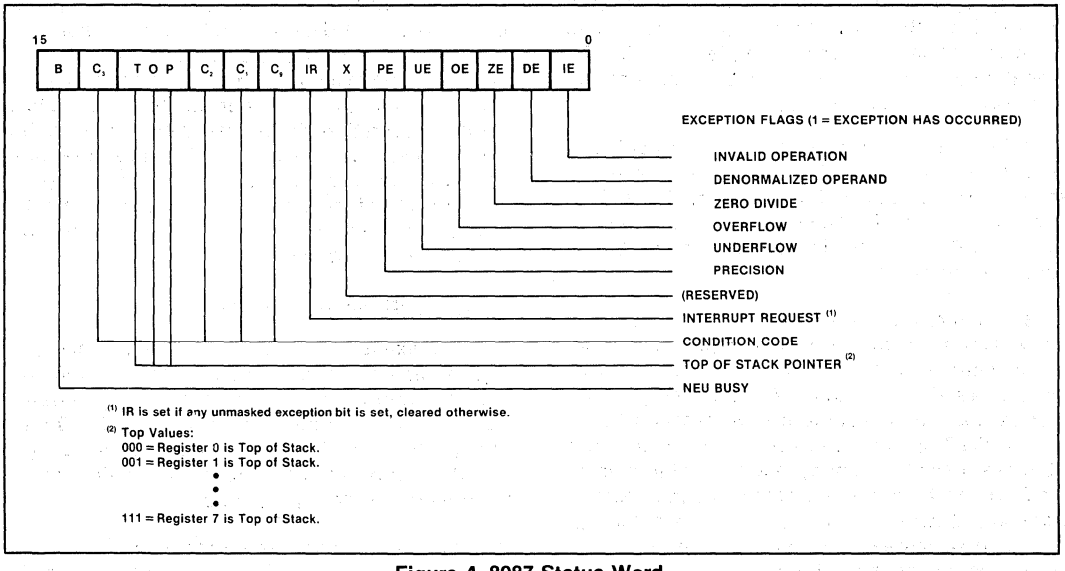


Figure 4. 8087 Status Word

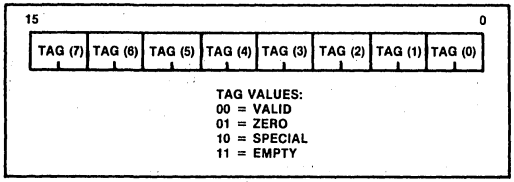


Figure 5. 8087 Tag Word

Instruction and Data Pointers

The instruction and data pointers (see Figure 6) are provided for user-written error handlers. Whenever the 8087 executes an NEU instruction, the CU saves the instruction address, the operand address (if present) and the instruction opcode. The 8087 can then store this data in memory.

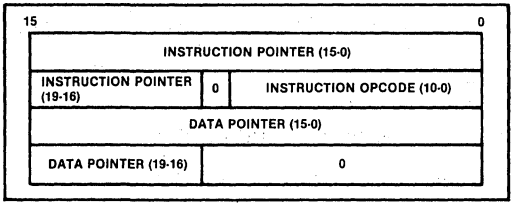


Figure 6. 8087 Instruction and Data Pointers

Control Word

The NDP provides several processing options which are selected by loading a word from memory into the control word. Figure 7 shows the format and encoding of the fields in the control word.

Exception Handling

The 8087 detects six different exception conditions that can occur during instruction execution. Any or all exceptions will cause an interrupt if unmasked and interrupts are enabled.

If interrupts are disabled the 8087 will simply suspend execution until the host clears the exception. If a specific exception class is masked and that exception occurs, however, the 8087 will post the exception in the status register and perform an on-chip default exception handling procedure, thereby allowing processing to continue. The exceptions that the 8087 detects are the following:

1. **INVALID OPERATION:** Stack overflow, stack underflow, indeterminate form (0/0, — , etc.) or the use of a Non-Number (NAN) as an operand. An exponent value is reserved and any bit pattern with this value in the exponent field is termed a Non-Number and causes this exception. If this exception is masked, the 8087's default response is to generate a specific NAN called INDEFINITE, or to propagate already existing NANs as the calculation result.

2. **OVERFLOW:** The result is too large in magnitude to fit the specified format. The 8087 will generate the code for infinity if this exception is masked.

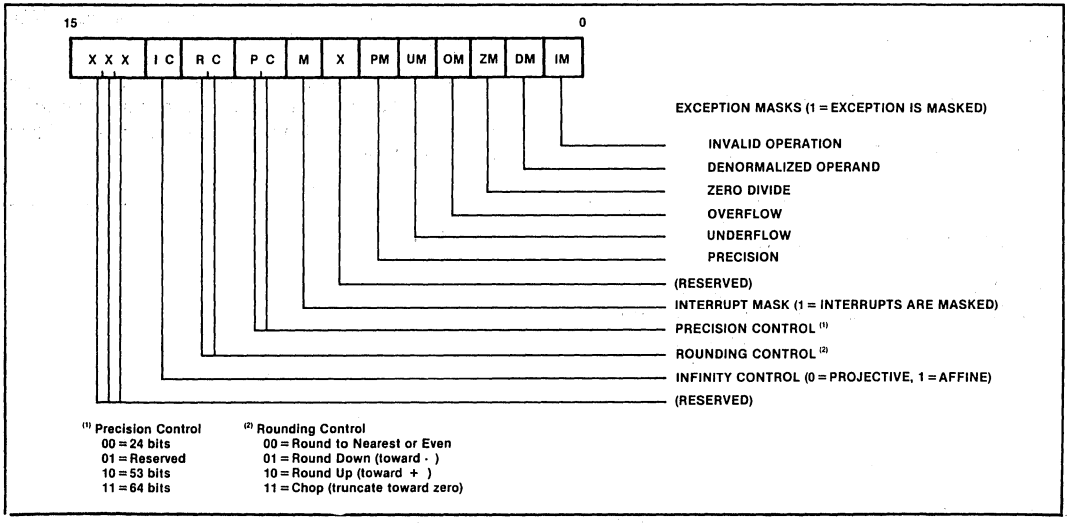


Figure 7. 8087 Control Word

3. **ZERO DIVISOR:** The divisor is zero while the dividend is a non-infinite, non-zero number. Again, the 8087 will generate the code for infinity if this exception is masked.

4. **UNDERFLOW:** The result is non-zero but too small in magnitude to fit in the specified format. If this exception is masked the 8087 will denormalize (shift right) the fraction until the exponent is in range. This process is called gradual underflow.

5. **DENORMALIZED OPERAND:** At least one of the operands or the result is denormalized; it has the smallest exponent but a non-zero significand. Normal processing continues if this exception is masked off.

6. **INEXACT RESULT:** If the true result is not exactly representable in the specified format, the result is rounded according to the rounding mode, and this flag is set. If this exception is masked, processing will simply continue.

SOFTWARE SUPPORT

The iSBC 337 module is supported by Intel's ASM-86/88 Assembly Language and PL/M-86/88 Systems Implementation Language. In addition to the instructions provided in the languages to support the additional math functions, a software emulator is also available to allow the execution of iAPX 86/20 instructions without the need for the iSBC 337 module. This allows for the development of software in an environment without the iAPX 86/20 processor and then transporting the code to its final run time environment with no change in mathematical results.

SPECIFICATIONS

Physical Characteristics

Width — 5.33 cm (2.100″)

Length — 5.08 cm (2.000″)

Height — 1.82 cm (.718″)
iSBC 337 board +
host board

Weight — 17.33 grams (.576 oz.)

Electrical Characteristics

DC Power Requirements (8087 only)

$V_{cc} = 5V \pm 5\%$ $I_{cc} = 475$ mA max.

Environmental Characteristics

Operating Temperature — 0°C to 55°C

Free air moving across base board and iSBC 337 module.

Relative Humidity — Up to 90% R.H. without condensation.

Reference Manual

142887-001 — iSBC 337 MULTIMODULE Numeric Data Processor Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

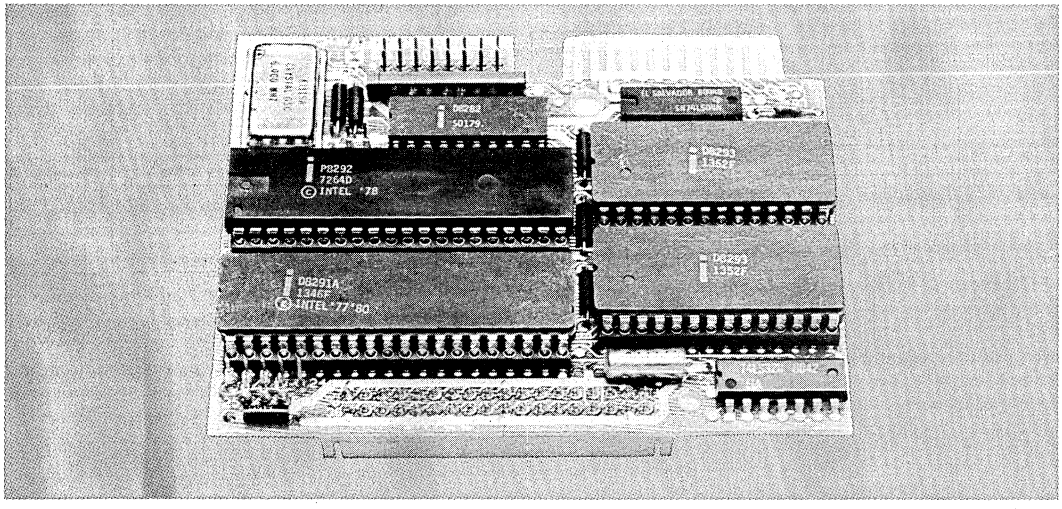
Part Number	Description
SBC 337	MULTIMODULE Numeric Data Processor

iSBX 488™
GPIB MULTIMODULE BOARD

- **Complete IEEE 488-1978 talker/listener functions including:**
  - **— Addressing, handshake protocol, service request, serial and parallel polling schemes**
- **Complete IEEE 488-1978 controller functions including:**
  - **— Transfer control, service requests and remote enable**
- **Simple read/write programming**
- **Software functions built into VLSI hardware for high performance, low cost and small size**
- **Standard iSBX™ Bus interface for easy connection to Intel iSBC™ boards**
- **IEEE 488-1978 standard electrical interface transceivers**
- **Five volt only operation**

The Intel iSBX 488 GPIB Talker/Listener/Controller Multimodule board provides a standard interface from any Intel iSBC board equipped with an iSBX connector to over 600 instruments and computer peripherals that employ the use of the IEEE 488-1978 standard (General Purpose Interface Bus). The single-wide iSBX 488 Multimodule board implements the complete IEEE 488-1978 Standard Digital Interface for Programmable Instrumentation by taking full advantage of Intel's VLSI technology. The iSBX 488 Multimodule board incorporates the 8291A GPIB Talker/Listener, 8292 GPIB Controller and two 8293 GPIB Transceiver devices on a single low cost 3.7″ by 2.85″ iSBX Multimodule board. The iSBX 488 board represents a significant step forward in joining microcomputers and instrumentation using industry standards such as the Multibus system bus, iSBX bus and IEEE 488-1978. The high performance iSBX 488 Multimodule board mounts easily on Intel iSBX bus compatible single board computers and provides functions which previously required a board eight times its size.

The iSBX 488 board provides a simple programming interface to the user for easy reading, writing and monitoring of all GPIB functions. The intelligent interface provided by the iSBX 488 board minimizes the impact of the host processor bandwidth.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, MULTIBUS, iRMX, iSBC, iSBX, MULTIMODULE, iCS and iEBC, and the combination of MCS, ICE, iRMX, iSBC, iSBX, iCS or iEBC, and a numerical suffix. 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 CORPORATION, 1981

April 1981
143580

FUNCTIONAL DESCRIPTION

The iSBX 488 Multimodule board is a single-wide iSBX bus compatible I/O expansion board that provides a complete implementation of the IEEE 488-1978 Standard Digital Interface for Programmable Instrumentation. The iSBX 488 Multimodule board may be configured to be a GPIB controller, talker, listener or talker/listener. The hardware implementation of the iSBX 488 board takes full advantage of Intel's VLSI capability by using the Intel 8292 GPIB controller, 8291A talker/listener and two (2) 8293 bus transceivers. All communication between the host iSBC board and the iSBX 488 Multimodule board is executed via the Intel standard iSBX connector. Many of the functions that previously were performed by user software have been incorporated into VLSI hardware for high performance and simple programming. Both the Intel 8291A GPIB Talker/Listener device and the 8292 device can each communicate independently with the host processor on the iSBC board depending on configuration. Communication from the host iSBC board to either device on the iSBX 488 board is flexible and may be either interrupt or poll driven depending on user requirements. Data transfers to or from the GPIB may be executed by the host processor's I/O Read and I/O Write commands or with DMA handshaking techniques for very high speed transfers.

GPIB Talker/Listener Capabilities

The Intel 8291A device on the iSBX 488 Multimodule board handles all talker/listener communications between the host iSBC processor board and the GPIB. Its capabilities include data transfer, bus handshake protocol, talker/listener addressing procedures, device clearing and triggering, service requests, and both serial and parallel polling schemes. In executing most procedures the iSBX 488 board does not interrupt the microprocessor on the iSBC processor board unless a byte of data is waiting on input or a byte is sent to an empty output buffer, thus offloading the host CPU of GPIB overhead chores.

SIMPLE PROGRAMMING INTERFACE — The GPIB talker/listener functions can be easily programmed using the high level commands made available by the Intel 8291A on the iSBX 488 Multimodule board. The 8291A device architecture includes eight registers for input and eight registers for output. One each of these read and write registers is used for direct data transfers. The remaining write registers are used by the pro-

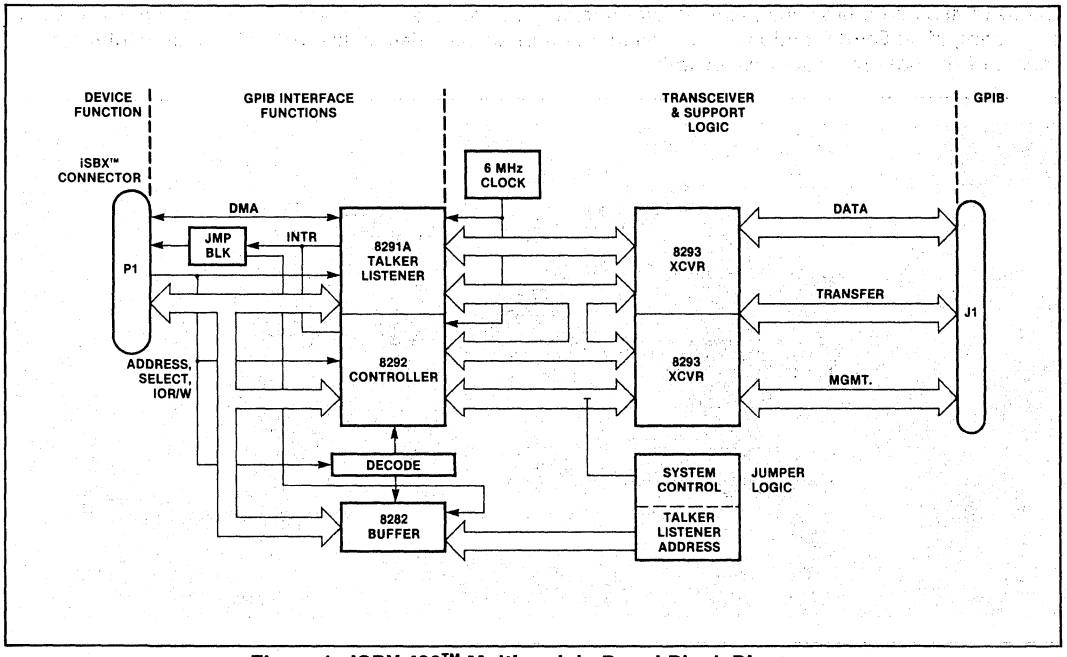


Figure 1. iSBX 488™ Multimodule Board Block Diagram

grammer to control the various interface features of the Intel 8291A device. The remaining read registers provide the user with a monitor of GPIB states, bus conditions and device status.

SOFTWARE FUNCTIONS BUILT INTO VLSI HARDWARE — Additional features that have migrated from discrete logic and software into Intel VLSI include programmable data transfer rate and three addressing modes that allow the iSBX board to be addressed as either a major or a minor talker/listener with primary or secondary addressing. The iSBX 488 Multimodule board can be programmatically configured into almost any bus talker, listener, or talker/listener configuration. Writing software to control these and other iSBX 488 board functions is simply a matter of reading or writing the control registers.

IEEE 488-1978 Functions[1]

Function	iSBX 488™ Supported IEEE Subsets
Source Handshake (SH)	SH0, SH1
Acceptor Handshake (AH)	AH0, AH1
Talker (T)	T0 through T8
Extended Talker (TE)	TE0 through TE8
Listener (L)	L0 through L4
Extended Listener (LE)	LE0 through LE4
Service Request (SR)	SR0, SR1
Remote Local (RL)	RL0, RL1
Parallel Poll (PP)	PP0, PP1, PP2
Device Clear (DC)	DC0 through DC2
Device Trigger (DT)	DT0, DT1
Controller (C)	C0 through C28

[1] For detailed information refer to IEEE Standard Digital Interface for Programmable Instrumentation published by The Institute of Electrical and Electronics Engineers, Inc., 1978.

Controller Capabilities

The GPIB controller functions supplied by the iSBX 488 board are provided by the Intel 8292 GPIB controller device. The 8292 is actually an Intel 8041A eight bit microcomputer that has been preprogrammed to implement all IEEE 488-1978 controller functions. The internal RAM in the 8041A is used as a special purpose register bank for the 8292 GPIB Controller. Just as with the 8291A GPIB Talker/Listener device, these registers are used by the programmer to implement controller monitor, read and write commands on the GPIB.

When configured as a bus controller the iSBX 488 board will respond to Service Requests (SRQ) and will issue Serial Polls. Parallel Polls are also issued to multiple GPIB instrument devices for receiving simultaneous responses. In applications requiring multiple bus controllers, several iSBX 488 boards may each be configured as a controller and pass the active control amongst each other. An iSBX 488 board configured for a System Controller has the capability to send Remote Enable (REN) and Interface Clear (IFC) for initializing the bus to a known state.

GPIB Physical Interface

The iSBX 488 Multimodule board interfaces to the GPIB using two Intel 8293 bidirectional transceivers. The iSBX 488 board meets or exceeds all of the electrical specifications defined in IEEE 488-1978 including the required bus termination specifications. In addition, for direct connection to the GPIB, the iSBC 988 cable, a 26 conductor 0.5 meter GPIB interface cable is also available from Intel. The cable is terminated with a 26-pin edge connector at the iSBX end and a 24-pin GPIB connector at the other. The cable is also supplied with shield lines for simple grounding in electrically noisy environments.

Installation

The iSBX 488 Multimodule board plugs directly onto the female iSBX connector available on many Intel iSBC boards. The Multimodule board is then secured at one additional point with nylon hardware (supplied) to insure the mechanical security of the assembly.

SPECIFICATIONS

Interface Information

iSBX™ Bus — All signals TTL compatible
26-pin edge connector — Electrical levels compatible with IEEE 488-1978.

Physical Characteristics

Width — 3.70 in (.94 cm)
Length — 2.85 in (7.24 cm)
Height — 0.8 in (2.04 cm)
Weight — 3.1 oz (87.8 gm)

GPIB Data Rate*

300K bytes/sec transfer rate with DMA host iSBC board

50K bytes/sec transfer rate using programmed I/O

730 nsec Data Accept Time

* Data rates are iSBX board maximum. Data rates will vary and can be slower depending on host iSBC board and user software driver.

Electrical Characteristics

DC power requirements —

V_{CC} = +5 Vdc ±5%

I_{CC} = 600 milliamps maximum

GPIB Electrical and Mechanical Specifications

Conforms to IEEE 488-1978 standard electrical levels and mechanical connector standard when purchased with the iSBC 988 GPIB cable.

Environmental Characteristics

Operating Temperature — 0° to 60°C (32° to 140°F)

Relative Humidity — Up to 90% R.H. without condensation.

Reference Manual

143154-001 — iSBX 488 GPIB Talker/Listener/Controller Multimodule Board Hardware Reference Manual (not supplied).

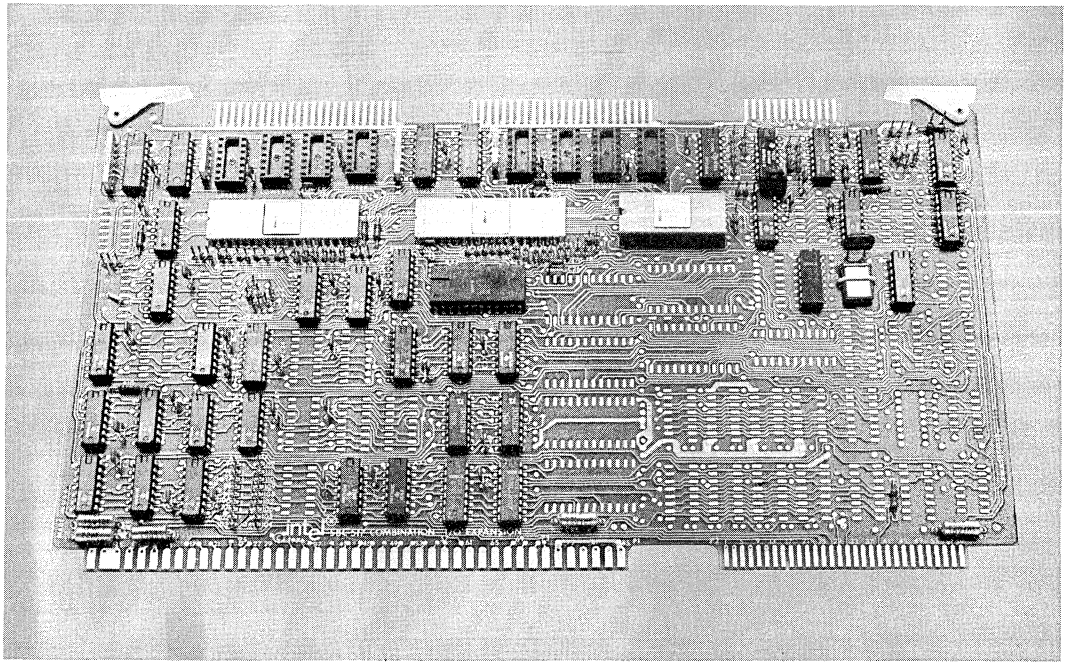
ORDERING INFORMATION

Part Number	Description
SBX 488	GPIB Multimodule
SBC 988	0.5 meter GPIB cable for iSBX 488 Multimodule Board

iSBC 517
COMBINATION I/O EXPANSION BOARD

- **48 programmable I/O lines with sockets for interchangeable line drivers and terminators**
- **Synchronous/asynchronous communications interface with RS232C drivers and receivers**
- **Eight maskable interrupt request lines with a pending interrupt register**
- **1 ms interval timer**

The iSBC 517 Combination I/O Expansion Board is a member of Intel's complete line of iSBC memory and I/O expansion boards. The board interfaces directly with any iSBC single board computer via the system bus to expand serial and parallel I/O capacity. The combination I/O board contains 48 programmable parallel I/O lines. The system software is used to configure the I/O lines to meet a wide variety of system peripheral requirements. The flexibility of the I/O interface is significantly enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. A programmable RS232C communications interface is provided on the iSBC 517. This interface may be programmed by the system software to provide virtually any asynchronous or synchronous serial data transmission technique (including IBM Bi-Sync). A comprehensive RS232C interface to CRTs, RS232C compatible cassettes, and asynchronous and synchronous modems is thus on the board. An on-board register contains the status of eight interrupt request lines which may be interrogated from the system bus, and each interrupt request line is maskable under program control. The iSBC 517 also contains a jumper selectable 1 ms interval timer and interface logic for eight interrupt request lines.



FUNCTIONAL DESCRIPTION

Programming Flexibility

The 48 programmable I/O lines on the iSBC 517 are implemented utilizing two Intel 8255 programmable peripheral interfaces. The system software is used to configure these programmable I/O lines in any of the combinations of unidirectional input/output, and bi-directional ports indicated in Table 1. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. The 48 programmable I/O lines and signal ground lines are brought out to two 50-pin edge connectors that mate with flat, round, or woven cable. Typical I/O read access time is 280 nanoseconds. Typical I/O read cycle time is 600 nanoseconds.

Communications Interface

The programmable communications interface on the iSBC 517 is provided by an Intel 8251 Universal Synchronous/Asynchronous Receiver/Transmitter (USART). The USART can be programmed by the system software to select the desired asynchronous or synchronous serial data transmission technique (including IBM Bi-Sync). The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and asynchronous serial transmission rate are all under program control. The 8251 provides full duplex, double-buffered transmit and receive capability, and parity, overrun, and framing error detection are all incorporated in the USART. The comprehensive RS232C interface on the board provides a direct interface to RS232C compatible equipment. The RS232C serial data lines and signal ground lines are brought out to a 26-pin edge connector that mates with RS232C compatible flat or round cables.

Interrupt Request Lines

Interrupt requests may originate from eight sources. Four jumper selectable interrupt requests can be automatically generated by the programmable peripheral interface when a byte of information is ready to be transferred to the CPU (i.e., input buffer is full) or a character has been transmitted (i.e., output data buffer is empty). Two jumper selectable interrupt requests can be automatically generated by the USART when a character is ready to be transferred to the CPU (i.e., receive buffer is full) or a character has been transmitted (transmit buffer is empty). These six interrupt request lines are all maskable under program control. Two interrupt request lines may be interfaced directly from user designated peripheral devices via the I/O edge connector. An on-board register contains the status of all eight interrupt request lines, and may be interrogated by the CPU. Each interrupt request line is

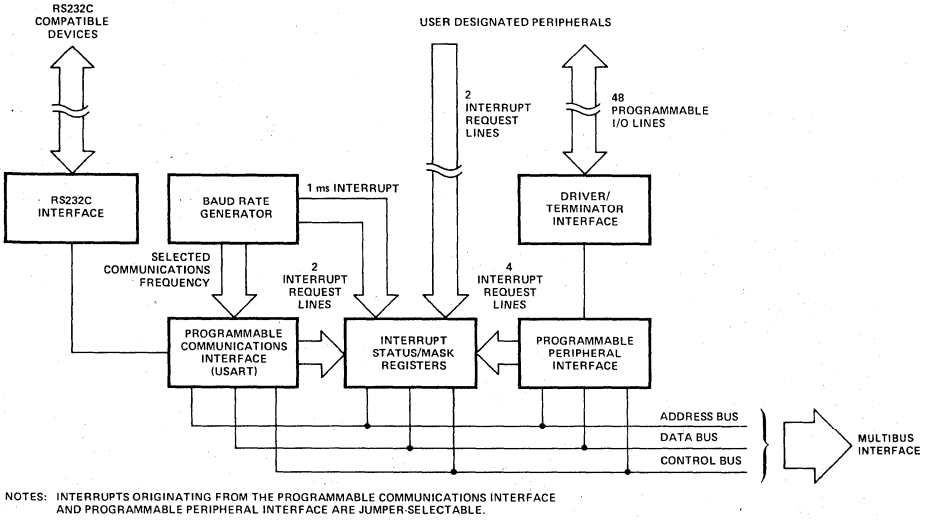


Figure 1. iSBC 517 Combination I/O Expansion Board Block Diagram

Table 1. Input/Output Port Modes of Operation

Ports	Lines (qty)	Mode of Operation					
		Unidirectional				Bidirectional	Control
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X	X	
2	8	X	X	X	X		
3	4	X		X			X[1]
	4	X		X			X[1]
4	8	X	X	X	X	X	
5	8	X	X	X	X		
6	4	X		X			X[2]
	4	X		X			X[2]

Notes

1. Part of port 3 must be used as control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.
2. Part of port 6 must be used as a control port when either port 4 or port 5 are used as a latched and strobed input or a latched and strobed output port or port 4 is used as a bidirectional port.

maskable under program control. Routing for the eight interrupt request lines is jumper selectable. They may be ORed to provide a single interrupt request line for the iSBC 80/10B, or they may be individually provided to the system bus for use by other iSBC single board computers.

Interval Timer

Each board contains a jumper selectable 1 ms interval timer. The timer is enabled by jumpering one of the interrupt request lines from the I/O edge connector to a 1 ms interval interrupt request signal originating from the baud rate generator.

SPECIFICATIONS

I/O Addressing

Port	1	2	3	4	5	6	8255 No. 1 Control	8255 No. 2 Control	USART Data	USART Control
Address	X4	X5	X6	X8	X9	XA	X7	XB	XC	XD

Note

X is any hex digit assigned by jumper selection.

I/O Transfer Rate

Parallel — Read or write cycle time 760 ns max
Serial — (USART)

Frequency (kHz) (Jumper Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous (Program Selectable)	
		÷16	÷64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
6.98	6980	—	110

Serial Communications Characteristics

Synchronous — 5-8 bit characters; internal or external character synchronization; automatic sync insertion.

Asynchronous — 5-8 bit characters; peak characters generation; 1, 1½, or 2 stop bits; false start bit detectors.

Interrupts

Eight interrupt request lines may originate from the programmable peripheral interface (4 lines), the USART (2 lines), or user specified devices via the I/O edge connector (2 lines) or interval timer.

Interrupt Register Address

X1 Interrupt mask register
X0 Interrupt status register

Note

X is any hex digit assigned by jumber selection.

Timer Interval

1.003 ms ±0.1% when 110 baud rate is selected
1.042 ms ±0.1% for all other baud rates

Interfaces

Bus — All signals TTL compatible
Parallel I/O — All signals TTL compatible
Serial I/O — RS232C
Interrupt Requests — All TTL compatible

Connectors

Interface	Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	CDC VPB01E43A00A1
Parallel I/O	50	0.1	3M 3415-000 or TI H312125
Serial I/O	26	0.1	3M 3462-000 or TI H312113
Auxiliary[1]	60	0.1	AMP PE5-14559 or TI H311130

Note

1. Connector heights and wire-wrap pin lengths are not guaranteed to conform to Intel OEM or system packaging. Auxiliary connector is used for test purposes only.

Line Drivers and Terminators

I/O Drivers — The following line drivers and terminators are compatible with all the I/O driver sockets on the iSBC 517:

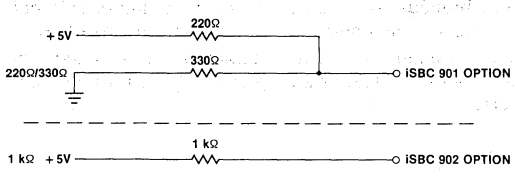
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note

I = inverting; NI = non-inverting; OC = open-collector.

Ports 1 and 4 have 25 mA totem-pole drivers and 1 kΩ terminators.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pullup



Bus Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-state	50
Commands	Tri-state	25

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 14 oz (397. 3 gm)

Electrical Characteristics

Average DC Current

V_{CC} = +5V ±5%
V_{DD} = +12V ±5%
V_{AA} = −12V ±5%
I_{CC} = 2.4 mA max
I_{DD} = 40 mA max
I_{AA} = 60 mA max

Note

Does not include power required for optional I/O drivers and I/O terminators. With eight 220Ω/330Ω input terminators installed, all terminator inputs low.

Environmental Characteristics

Operating Temperature — 0 °C to +55 °C

Reference Manual

9800388B — iSBC 517 hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

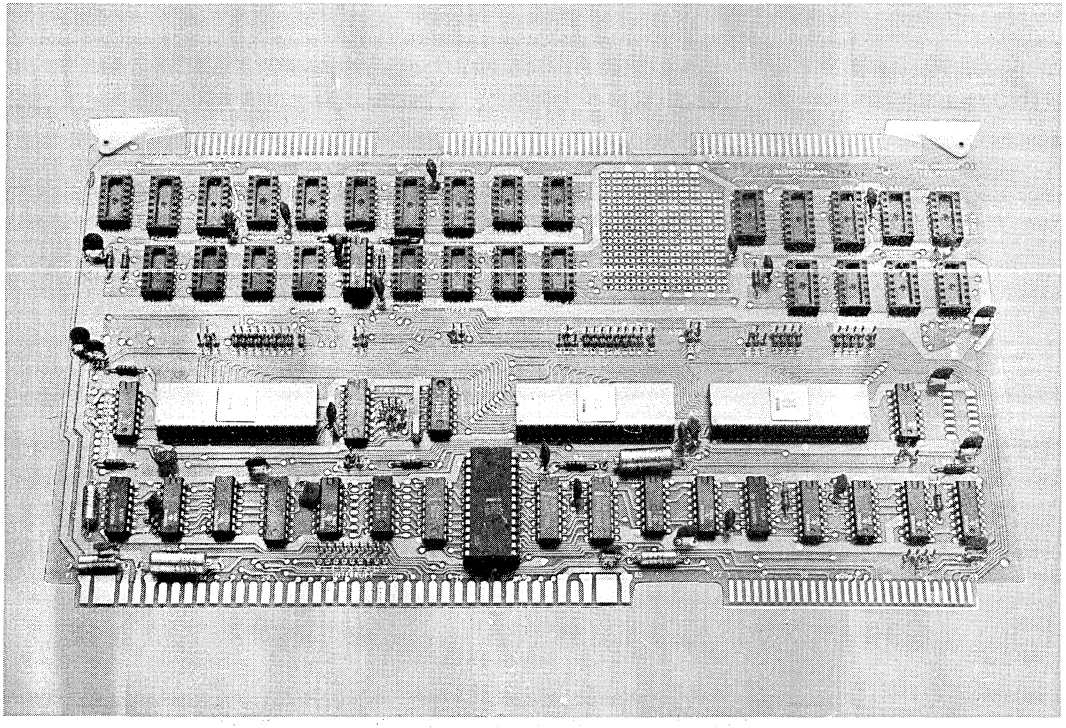
ORDERING INFORMATION

Part Number	Description
SBC 517	Combination I/O Expansion Board

iSBC 519 (or pSBC 519*)
PROGRAMMABLE I/O EXPANSION BOARD

- **iSBC I/O expansion via direct MULTIBUS Interface**
- **72 programmable I/O lines with sockets for interchangeable line drivers and terminators**
- **Jumper selectable I/O port addresses**
- **Jumper selectable 0.5, 1.0, 2.0, or 4.0 ms interval timer**
- **Eight maskable interrupt request lines with priority encoded and programmable interrupt algorithms**

The iSBC 519 Programmable I/O Expansion Board is a member of Intel's complete line of iSBC memory and I/O expansion boards. The iSBC 519 interfaces directly to any iSBC single board computer via the system bus to expand input and output port capacity. The iSBC 519 provides 72 programmable I/O lines. The system software is used to configure the I/O lines to meet a wide variety of peripheral requirements. The flexibility of the I/O interface is further enhanced by the capability of selecting the appropriate combination of optional line drivers and terminators to provide the required sink current, polarity, and drive/termination characteristics for each application. Address selection is accomplished by using wire-wrap jumpers to select one of 16 unique base addresses for the input and output ports. The board operates with a single +5V power supply.



*Same product, manufactured by Intel Puerto Rico, Inc.

FUNCTIONAL DESCRIPTION

The 72 programmable I/O lines on the iSBC 519 are implemented utilizing three Intel 8255 programmable peripheral interfaces. The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports indicated in Table 1. In order to take full advantage of the large number of possible I/O configurations, sockets are provided for interchangeable I/O line drivers and terminators. The 72 programmable I/O lines and signal ground lines are brought out to three 50-pin edge connectors that mate with flat, round, or woven cable.

Interval Timer

Typical I/O read access time is 350 nanoseconds. Typical I/O read/write cycle time is 450 nanoseconds. The interval timer provided on the iSBC 519 may be used to generate real time clocking in systems requiring the periodic monitoring of I/O functions. The time interval is derived from the constant clock (BUS CCLK) and the timing interval is jumper selectable. Intervals of 0.5, 1.0, 2.0, and 4.0 milliseconds may be selected when an iSBC single board computer is used to generate the clock. Other timing intervals may be generated if the user provides a separate constant clock reference in the system.

Eight-Level Vectored Interrupt

An Intel 8259 programmable interrupt controller (PIC) provides vectoring for eight interrupt levels. As shown in Table 2, a selection of three priority processing algorithms is available to the system designer so that the

Table 1. Input/Output Port Modes of Operation

Ports	Lines (qty)	Mode of Operation					Control
		Unidirectional				Bidirectional	
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1,4,7	8	X	X	X	X	X	
2,5,8	8	X	X	X	X		
3,6,9	4	X		X			X[1,2,3]
	4	X		X			X[1,2,3]

Notes

1. Part of port 3 must be used as a control port when either port 1 or port 2 are used as a latched and strobed input or a latched and strobed output port or port 1 is used as a bidirectional port.
2. Part of port 6 must be used as a control port when either port 4 or port 5 are used as a latched and strobed input or a latched and strobed output port or port 4 is used as a bidirectional port.
3. Part of port 9 must be used as a control port when either port 7 or port 8 are used as a latched and strobed input or a latched and strobed output port or port 7 is used as a bidirectional port.

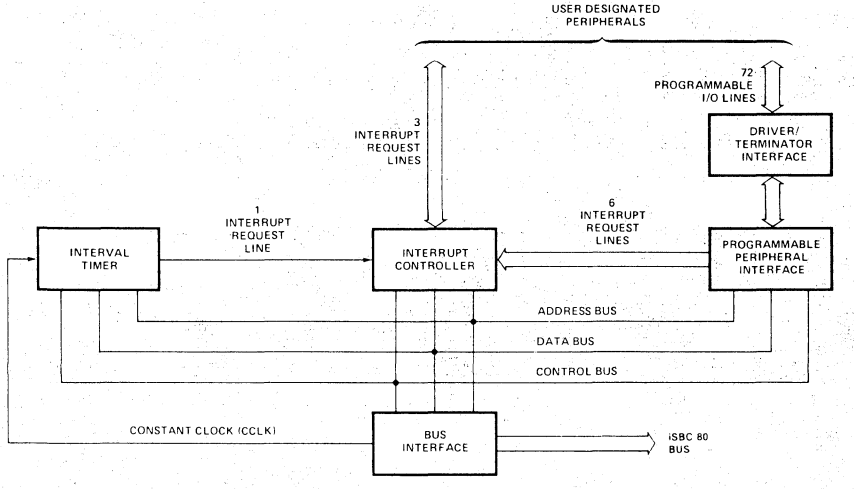


Figure 1. iSBC 519 Programmable I/O Expansion Board Block Diagram

Table 2. Interrupt Priority Options

Algorithm	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels are based in sequence numerically on this assignment.

manner in which requests are serviced may be configured to match system requirements. Priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts interrupt requests from the programmable parallel I/O interfaces, the interval timer, or direct from peripheral equipment. The PIC then determines which of the incoming requests is of the highest priority, determines whether this request is of higher priority than the level currently being serviced, and if appropriate, issues an interrupt to the system master. Any combination of interrupt levels may be masked through storage, via software, of a single byte to the interrupt mask register of the PIC.

Interrupt Request Generation — Interrupt requests may originate from 10 sources. Six jumper selectable interrupt requests can be automatically generated by the programmable peripheral interfaces when a byte of information is ready to be transferred to the system master (i.e., input buffer is full) or a character has been transmitted (i.e., output data buffer is empty). Three interrupt request lines may be interfaced to the PIC directly from user designated peripheral devices via the I/O edge connectors. One interrupt request may be generated by the interval timer.

Bus Line Drivers — The PIC interrupt request output line may be jumper selected to drive any of the nine interrupt lines on the MULTIBUS. Any of the on-board request lines may also drive any interface interrupt line directly via jumpers and buffers on the board.

SPECIFICATIONS

Addressing

Port	1	2	3	8255 No. 1 Control	4	5	6	8255 No. 2 Control	7	8	9	8255 No. 3 Control
Address	X0	X1	X2	X3	X4	X5	X6	X7	X8	X9	XA	XB

Interrupts

Register Addresses (hex notation, I/O address space)

XD Interrupt request register
XC In-service register
XD Mask register
XC Command register
XD Block address register
XC Status (polling register)

Note
Several registers have the same physical address; sequence of access and one data bit of control word determines which register will respond.

Ten interrupt request lines may originate from the programmable peripheral interface (6 lines), or user specified devices via the I/O edge connector (3 lines), or interval timer (1 line).

Interval Timer

Output Register — Timer interrupt register output is cleared by an output instruction to I/O address XE or XF[1].

Timing Intervals — 500, 1,000, 2,000, and 4,000 ms ± 1%; jumper selectable[2].

Notes
1. X is any hex digit assigned by jumper selection.
2. Assumes constant clock (CCLK) frequency of 9.216 MHz ± 1%.

Interfaces

Bus — All signals TTL compatible
Parallel I/O — All signals TTL compatible
Interrupt Requests — All TTL compatible

Connectors

Interface	Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	Viking 3KH43/9AMK12
Parallel I/O	50	0.1	3M 3415-000 or TI H312125
Serial I/O	26	0.1	3M 3462-000 or TI H312113
Auxiliary[1]	60	0.1	AMP PE5-14559 or TI H311130

Note
1. Connector heights and wirewrap pin lengths are not guaranteed to conform to Intel OEM or System packaging.

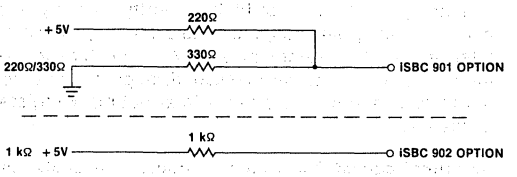
Line Drivers and Terminators

I/O Drivers — The following line drivers and terminators are compatible with all the I/O driver sockets on the iSBC 519:

Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note
I = inverting; NI = non-inverting; OC = open-collector.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pullup



Ports 1, 4, and 7 may use any of the drivers or terminators shown above for unidirectional (input or output) port configurations. Either terminator and the following bidirectional drivers and terminators may be used for ports 1, 4, and 7 when these ports are used as bidirectional ports.

Bidirectional Drivers

Driver	Characteristic	Sink Current (mA)
Intel 8216	NI, TS	25
Intel 8226	I, TS	50

Note

I = inverting; NI = non-inverting; TS = three-state.

Terminators (for ports 1, 4, and 7 when used as bidirectional ports)

Supplier	Product Series
CTS	760-
Dale	LDP14k-02
Beckman	899-1

Bus Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-state	50
Commands	Tri-state	25

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 14 oz (397.3 gm)

Electrical Characteristics

Average DC Current

Voltage	Without Termination[1]	With Termination[2]
V_{CC} = +5V ±5%	I_{CC} = 1.5A max	3.5A max

Note

1. Does not include power required for optional I/O drivers and I/O terminators.
2. With 18 220Ω/330Ω Input terminators installed, all terminator inputs low.

Environmental Characteristics

Operating Temperature — 0°C to +55°C

Reference Manual

9800385B — iSBC 519 hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
SBC 519	Programmable I/O Expansion Board

iSBC 556
OPTICALLY ISOLATED I/O BOARD

- **iSBC 80 and MULTIBUS compatible**
- **Up to 48 digital optically isolated input/output data lines**
- **Choice of**
  - **— 24 fixed input lines**
  - **— 16 fixed output lines**
  - **— 8 programmable lines**
- **Provisions for plug-in, optically isolated receivers, drivers, and terminators**
- **Voltate/current levels**
  - **— Input up to 48V**
  - **— Output up to 30V, 60 mA**
- **Common interrupt for up to 8 sources**
- **+5V supply only**

The iSBC 556 Optically Isolated I/O Board provides 48 digital input/output lines with isolation between process application or peripheral device and the iSBC 80 series single board computers. The iSBC 556 contains two 8255A programmable interface devices, and sockets for user supplied optically isolated drivers, receivers, and input resistor terminators, together with common interrupt logic and iSBC 80 bus interface logic. Input signals can be single-ended or differential types with user defined input range (resistor terminator and opto-isolated receiver selection), allowing flexibility in design of voltage and threshold levels. The output allows user selection of Opto-Isolated Darlington Pair which can be used as an output driver either as an open collector or current switch.

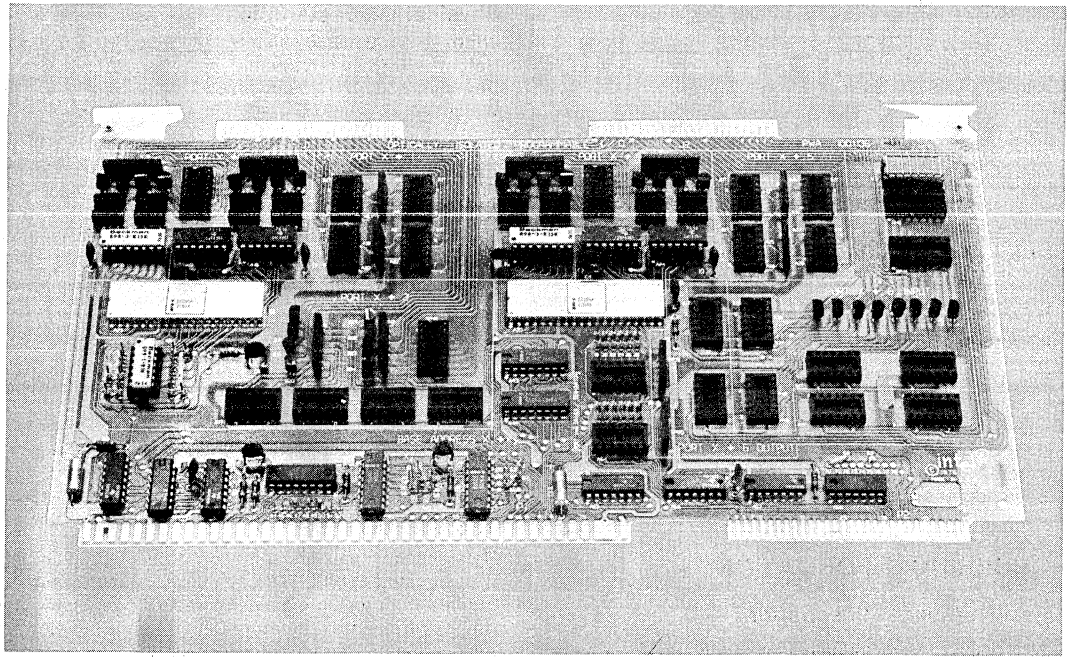


Table 1. I/O Ports Opto-Isolator Receivers, Drivers, and Terminators

Port No. X = I/O Base Address	Type of I/O	Lines (qty)	Resistor Terminator Pac Rp 16-Pin DIP Bourns 4116R-00 or Equivalent	Dual Opto-Isolator 8-Pin Dip Monsanto MC T66 or Equivalent	Dual Opto-Isolator Darlington Pair 6-Pin DIP Monsanto 4N29, 30 31, 32 or Equivalent	Driver 7438 or Equivalent	Pull-Up iSBC 902
X+0	Input	8	1	4	—	—	
X+1	Output	8	—		8	—	
X+2	Input/ Control	8	1		8	—	
X+4	Input	8	1	4	—	—	
X+5	Output	8			8	—	
X+6	Input/ Output	8	1 if input		8	2 if output	2 if input
X+7	Control						

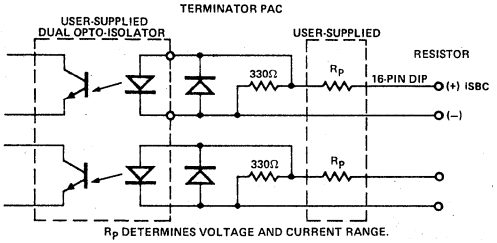
SPECIFICATIONS

Number of Lines

24 input lines
16 output lines
8 programmable lines: 4 input — 4 output

I/O Interface Characteristics

Line-to-Line Isolation — 235V DC or peak AC
Input/Output Isolation — 500V DC or peak AC



Bus Interface Characteristics

All data address and control commands are iSBC 80 bus compatible.

I/O Addressing

Port	8255 #1			Control	8255 #2			Control
	A	B	C		A	B	C	
Address	X+0	X+1	X+2	X+3	X+4	X+5	X+6	X+7

Where:
base address is from 00H to 1FH (jumper selectable)

ORDERING INFORMATION

Part Number	Description
SBC 556	Optically Isolated I/O Board

Connectors

Interface	Pins (qty)	Centers		Mating Connectors
		in.	cm	
P1 iSBC bus	86	0.156		Viking 3KH43/9AMK12
J1 16 fixed input & 8 fixed output lines	50	0.1		3M 3415-000 or TI M312125
J2 8 fixed output, 8 fixed output, & 8 programmable input/output lines	50	0.1		3M 3415-000 or TI M312125

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 12 oz (397.3 gm)

Electrical Characteristics

Average DC Current

$V_{CC} = +5V \pm 5\%$, 1.0A without user supplied isolated receiver/driver

$I_{CC} = 1.6A$ max with user supplied isolator receiver/driver

Environmental Characteristics

Temperature — 0°C to 55°C
Relative Humidity — 0 to 90%, non-condensing

Reference Manual

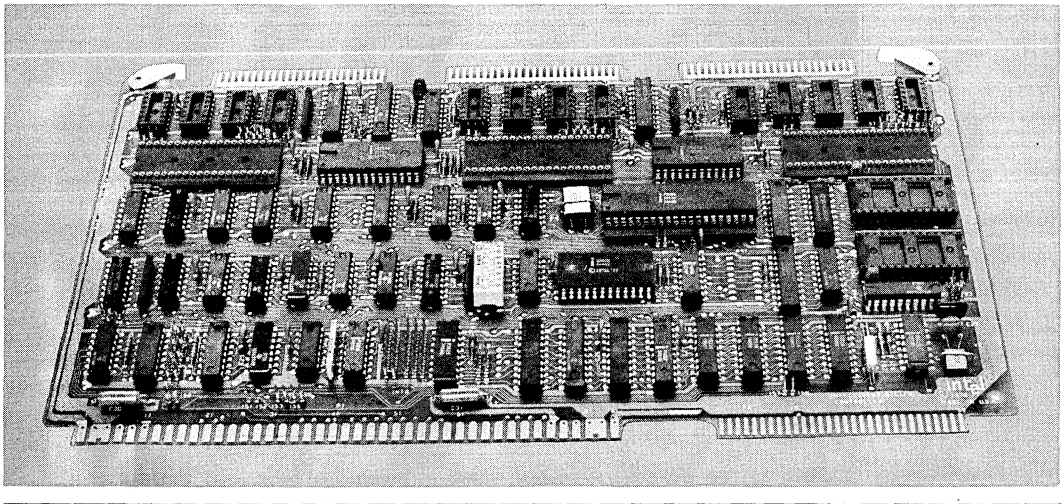
9800489-02 — iSBC 556 Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

iSBC 569
INTELLIGENT DIGITAL CONTROLLER

- **Single board digital I/O controller with up to four microprocessors to share the digital input/output signal processing**
- **3 MHz 8085A central control processor**
- **Three sockets for 8041/8741A Universal Peripheral Interface (UPI-41A) for distributed digital I/O processing, such as:**
  - **— Industrial signal processor (iSBC 941)**
  - **— Custom programmed 8041A/8741A**
- **Three operational modes**
  - **— Stand-along digital controller**
  - **— MULTIBUS master**
  - **— Intelligent slave (slave to MULTIBUS master)**
- **2K bytes of dual port static read/write memory**
- **Sockets for up to 8K bytes of Intel 2758, 2716, 2732 erasable programmable read only memory**
- **48 programmable parallel I/O lines with sockets for interchangeable line drivers or terminators**
- **Three programmable counters**
- **12 levels of programmable interrupt control**
- **Single +5V supply**
- **MULTIBUS standard control logic compatible with optional iSBC 80 and iSBC 86 CPU, memory, and I/O expansion boards**

The Intel iSBC 569 Intelligent Digital Controller is a single board computer (8085A based) with sockets for three 8041A/8741A Universal Peripheral Interface chips (UPI-41A). The I/O processing algorithm may be tailored to application requirements using designer selected combinations of standard Intel industrial signal processors (e.g., iSBC 941) or user programmed UPI-41A processors. These devices may be used to offload the 8085A processor from time consuming tasks such as pulse counting, event sensing, and parallel or serial digital I/O data formatting with error checking and handshaking. The iSBC 569 board is a complete digital controller with up to four processors on a single 6.75 inches x 12.00 inches (17.15cm x 30.48cm) printed circuit board. The 8085A CPU, system clock, read/write memory, non-volatile memory, priority interrupt logic, programmable timers, MULTIBUS control and interface logic, optional UPI processors and optional line driver and terminators all reside on one board.



FUNCTIONAL DESCRIPTION

Intelligent Digital Controller

Three modes of operation — the iSBC 569 Intelligent Digital Controller is capable of operating in one of three modes; stand alone controller, bus master, or intelligent slave.

Stand alone controller — the iSBC 569 board may function as a stand alone, single board controller with CPU, memory, and I/O elements on a single board. Five volt (+5VDC) only operation allows configuration of low cost controllers with only a single power supply voltage. The on-board 2K bytes RAM and up to 16K bytes ROM/EPROM, as well as the assistance of three UPI-41A processors, allow significant digital I/O control from a single board.

Bus master — in this mode of operation, the iSBC 569 controller may interface with and control iSBC expansion memory and I/O boards, or even other iSBC 569 Intelligent Digital Controllers configured as intelligent slaves (but no additional bus masters).

Intelligent slave — the iSBC 569 controller can perform as an intelligent slave to any 8- or 16-bit MULTIBUS master CPU by offloading the master of digital control related tasks. Preprocessing of data for the master is controlled by the on-board 8085A CPU which coordinates up to three UPI-41A processors. Using the iSBC 569 board as an intelligent slave, multi-channel digital control can be managed entirely on-board, freeing a system master to perform other system functions. The dual port RAM memory allows the iSBC 569 controller to process and store data without MULTIBUS memory contention.

Simplified Programming

By using Intel UPI-41A processors for common tasks such as counting, sensing change of state, printer control and keyboard scanning/debouncing, the user frees up time to work on the more important application programming of machine or process optimization. Controlling the Intel UPI-41A processors becomes a simple task of reading or writing command and data bytes to or from the data bus buffer register on the UPI device. Programming the iSBC 941 Industrial Digital Processor to produce a pulse output, for example, is as simple as sending command and parameter bytes indicating initialization, pulse output selection, period and delay parameters, followed by a command to begin execution.

Central Processing Unit

A powerful Intel 8085A 8-bit CPU, fabricated on a single LSI chip, is the central processor for the iSBC 569 ™ controller. The six general purpose 8-bit registers may be addressed individually or in pairs, providing both single and double precision operations. The program counter can address up to 64K bytes of memory using iSBC expansion boards. The 16-bit stack pointer controls the addressing of an external stack. This stack provides sub-routine nesting bounded only by memory size. The minimum instruction execution time is 1.30 microseconds. The 8085A CPU is software compatible with the Intel 8080A CPU.

Bus Structure

The iSBC 569 Intelligent Digital Controller utilizes a triple bus architecture concept. An internal bus is used for on-board memory and I/O operations. A MULTIBUS interface is available to provide access for all external memory and I/O operations. A dual port bus with controller enables access via the third bus to 2K bytes of static RAM from either the on-board CPU or a system master. Hence, common data may be stored in on-board memory and may be accessed either by the on-board CPU or by system masters. A block diagram of the iSBC 569 functional components is shown in Figure 1.

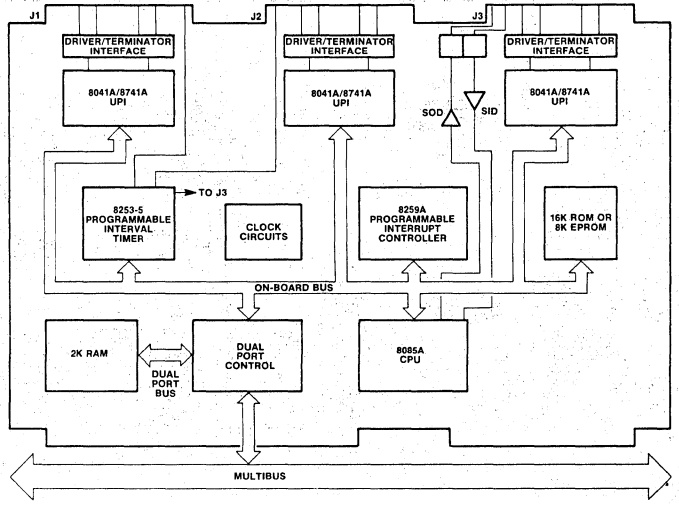


Figure 1. iSBC 569 Intelligent Digital Controller Block Diagram

RAM Capacity

The iSBC 569 board contains 2K bytes of read/write memory using Intel 2114 static RAMs. RAM accesses may occur from either the iSBC 569 controller or from any other bus master interfaced via the MULTIBUS system bus. The iSBC 569 board provides addressing jumpers to allow the on-board RAM to reside within a one megabyte address space when accessed via the system bus. In addition, a switch is provided which allows the user to reserve a 1K byte segment of on-board RAM for use by the 8085A CPU. This reserved RAM space is not accessible via the system bus and does not occupy any system address space.

EPROM/ROM Capacity

Two sockets for up to 16K bytes of nonvolatile read only memory are provided on the iSBC 569 board. Nonvolatile memory may be added in 1K-byte increments up to a maximum of 2K bytes using Intel 2758 erasable and electrically reprogrammable ROMs (EPROMs); in 2K-byte increments up to a maximum of 4K bytes using Intel 2316 ROMs or 2716 EPROMs; in 4K byte increments up to 8K bytes maximum using Intel 2732 EPROMs; or in 8K-byte increments up to 16K bytes maximum using Intel 2364 ROMs (both sockets must contain same type ROM/EPROM). All on-board ROM/EPROM operations are performed at maximum processor speed.

Universal Peripheral Interfaces (UPI-41A)

The iSBC 569 Intelligent Digital Controller board provides three sockets for user supplied Intel 8041A/8741A Universal Peripheral Interface (UPI-41A) chips. Sockets are also provided for the associated line drivers and terminators for the UPI I/O ports. The UPI-41A processor is a single chip microcomputer containing a CPU, 1K bytes of ROM (8041A) or EPROM (8741A), 64 bytes of RAM, 16 programmable I/O lines, and an 8-bit timer/event counter. Special interface registers included in the chip allow the UPI-41A processor to function as a slave processor to the iSBC 569 controller board's 8085A CPU. The UPI processor allows the user to specify algorithms for controlling peripherals directly thereby freeing the 8085A for other system functions. For additional information, including UPI-41A instructions, refer to the UPI-41 User's Manual (Manual No. 9800504).

Industrial Digital Processor (iSBC 941)

The iSBC 941 Industrial Digital Processor is a 40-pin DIP device which provides the user with easy-to-use processing of digital input and output signals desired in many industrial automation environments. One of nine digital I/O functions can be selected at any one time for measuring, counting, or controlling up to 16 separate I/O lines. An additional eight utility commands allow reading or setting the condition of unused I/O lines. Simplex serial input and output modes can assemble or disassemble bytes transmitted asynchronously over TTL lines, including insertion and deletion of start/stop bits. The iSBC 941 processor plugs into any of the three UPI-41A sockets on the iSBC 569 board. Simple programming commands from the master 8085A processor can thus implement up to 48 lines of preprocessed digital I/O signals. For specific commands and further information, refer to the iSBC 941 Data Sheet in this document.

Programmable Timers

The iSBC 569 Intelligent Digital Controller board provides three independently programmable interval timer/counters utilizing one Intel 8253 Programmable Interval Timer (PIT). The Intel 8253 PIT provides three 16-bit BCD or binary interval timer/counters. Each timer may be used to provide a time reference for each UPI™ processor or for a group of UPI processors. The output of each timer also connects to the 8259A Programmable Interrupt Controller (PIC) providing the capability of timed interrupts. All gate inputs, clock inputs, and timer outputs of the 8253 PIT are available at the I/O ports for external access.

Timer Functions — In utilizing the iSBC 569 controller, the systems designer simply configures, via software, each timer to meet systems requirements. The 8253 PIT modes are listed in Table 1. The contents of each counter may be read at any time during system operation with simple read operations for event counting applications. The contents of each counter can be read "on-the-fly" for time stamping events or time clock referenced program initiations.

Table 1. 8253 Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached, an interrupt request is generated.
Programmable one-shot	Output goes low upon receipt of an external trigger edge or software command and returns high when terminal count is reached. This function is retriggerable.
Rate generator	Divide by N counter. The output will go low for one input clock cycle, and the period from one low-going pulse to the next is N times the input clock period.
Square-wave rate generator	Output will remain high until one-half the count has been completed, and go low for the other half of the count.
Software triggered strobe	Output remains high until software loads count (N). N counts after count is loaded, output goes low for one input clock period.
Hardware triggered strobe	Output goes low for one clock period N counts after rising edge on counter trigger input. The counter is retriggerable.
Event counter	On a jumper selectable basis, the clock input becomes an input from the external system. CPU may read the number of events occurring after the counting "window" has been enabled or an interrupt may be generated after N counts occur in the system.

Interrupt Capability

The iSBC 569 Intelligent Digital Controller provides interrupt service for up to 12 interrupt sources. Any of the 12 sources may interrupt the on-board processor. Four interrupt levels are handled directly by the 8085A CPU and eight levels are serviced from an Intel 8259A Programmable Interrupt Controller (PIC) routing an interrupt request output to the INTR input of the 8085A.

8085A Interrupt — Each of four direct 8085A interrupt inputs has a unique vector memory address. An 8085A jump instruction at each of these addresses then provides software linkage to interrupt service routines located independently anywhere in the memory.

8259A Interrupts — The eight interrupt sources originate from both on-board controller functions and the system bus:

UPI-41A Processors — one interrupt from each of three UPI processor sockets.

8253 PIT — one interrupt from each of three timer outputs.

MULTIBUS System Bus — one of eight MULTIBUS interrupt lines may be jumpered to either of two 8259A PIC interrupt inputs.

Programmable Reset — The iSBC 569 Intelligent Digital Controller board has a programmable output latch used to control on-board functions. Three of the outputs are connected to separate UPI-41A RESET inputs. Thus, the user can reset any or all of the UPI-41A processors under software control. A fourth latch output may be used to generate an interrupt request onto the MULTIBUS interrupt lines. A fifth latch output is connected to a light-emitting diode which may be used for diagnostic purposes.

Expansion Capabilities

When the iSBC 569 controller is used as a single board digital controller, memory and I/O capacity may be expanded using Intel MULTIBUS compatible expansion boards. In this mode, no other bus masters may be in the system. Memory may be expanded to a 64K byte capacity by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be increased by adding I/O expansion boards. Multiple iSBC 569 boards may be included in an expanded system using one iSBC 569 Intelligent Digital Controller as the system master and additional controllers as intelligent slaves.

Intelligent Slave Programming

When used as an intelligent slave, the iSBC 569 controller appears as an additional RAM memory module. System bus masters communicate with the iSBC 569 board as if it were just an extension of system memory. To simplify this communication, the user has been given some specific tools:

Flag Interrupt — The Flag Interrupt is generated any time a write command is performed by an off-board CPU to the first location of iSBC 569 RAM. This interrupt provides a means for the master CPU to notify the iSBC 569 controller that it wished to establish a communications sequence. The flag interrupt is cleared when the on-board processor reads the first location of its RAM. In systems with more than one intelligent slave, the flag interrupt provides a unique interrupt to each slave outside the normal MULTIBUS interrupt lines (INT0/-INT7/).

RAM — The on-board 2K byte RAM area that is accessible to both an off-board CPU and the on-board 8085A may be configured for system access on any 2K boundary.

MULTIBUS Interrupts — The third tool to improve system operation as an intelligent slave is access to the MULTIBUS interrupt lines. The iSBC 569 controller can both respond to interrupt signals from an off-board CPU, and generate an interrupt to the off-board CPU via the system bus.

System Development Capability

Software development for the iSBC 569 Intelligent Digital Controller board is supported by the Intellec® Microcomputer Development System including a resident macroassembler, text editor, system monitor, a linker, object code locator, and Library Manager. In addition, both PL/M and FORTRAN language programs can be compiled to run on the iSBC 569 board. A unique in-circuit emulator (ICE-85™) option provides the capability of developing and debugging software directly on the iSBC 569 board. This greatly simplifies the design, development, and debug of iSBC 569 system software.

SPECIFICATIONS

8085A CPU

Word Size — 8, 16 or 24 bits

Cycle Time — 1.30 μsec ± .1% for fastest executable instruction; i.e., four clock cycles.

Clock Rate — 3.07 MHz ± .1%

System Access Time

Dual port memory — 725 nsec

Memory Capacity

On-board ROM/EPROM — 2K, 4K, 8K, or 16K bytes of user installed ROM or EPROM

On-board RAM — 2K bytes of static RAM. Fully accessible from on-board 8085A. Separately addressable from system bus.

Off-board expansion — up to 64K bytes of EPROM/ROM or RAM capacity.

I/O Capacity

Parallel-Timers — Three timers, with independent gate input, clock input, and timer output user-accessible. Clock inputs can be strapped to an external source or to an on-board 1.3824 MHz reference. Each timer is connected to a 8259A Programmable Interrupt Controller and may also be optionally connected to UPI processors.

UPI-I/O — Three UPI-41A interfaces, each with two 8-bit I/O ports plus the two UPI Test Inputs. The 8-bit ports are user-configurable (as inputs or outputs) in groups of four.

Serial — 1 TTL compatible serial channel utilizing SID and SOD lines of on-board 8085A CPU

On-Board Addressing

All communications to the UPI-41A processors, to the programmable reset latch, to the timers, and to the interrupt controller are via read and write commands from the on-board 8085A CPU.

Memory Addressing

On-board ROM/EPROM — 0-07FF (using 2758 EPROMs); 0-OFFF (using 2716 EPROMs or 2316 ROMs); 0-1FFF (using 2732 EPROMs); 0-3FFF (using the 2364 ROMs)

On-board RAM — 8000-87FF System access — any 2K increment 00000-FF800 (switch selectable); 1K bytes may be disabled from bus access by switch selection.

I/O Addressing

Source	Addresses
8253	0E0H-0E3H
UPI0	0E4H-0E5H
UPI1	0E6H-0E7H
UPI2	0E8H-0E9H
PROGRAMMABLE RESET	0EAH-0EBH
8259A	0ECH-0EDH

Timer Specifications

Input frequencies — jumper selectable reference

Internal: 1.3824 MHz ± .1% (.723 μsec, nominal)

External: User supplied (2 MHz maximum)

Output Frequencies (at 1.3824 MHz)

Function	Min[1]	Max[1]
Real-time interrupt interval	1.45 μsec	47.4 msec
Rate Generator (frequency)	21.09 Hz	691.2 KHz
1. Single 16-bit binary count		

Interfaces

MULTIBUS™ Interface — All signals compatible with iSBC and MULTIBUS architecture

Parallel I/O — All signals TTL compatible

Interrupt Requests — All TTL compatible

Timer — All signals TTL compatible

Serial I/O — All signals TTL compatible

Connectors

Interface	Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	Viking 3KH43/9AMK12
Parallel I/O	50	0.1	3M 3415-000 or TI H312125

Physical Characteristics

Width — 30.48 cm (12.00 inches)

Depth — 17.15 cm (6.75 inches)

Thickness — 1.27 cm (0.50 inch)

Weight — 3.97 gm (14 ounces)

Electrical Characteristics

DC Power Requirements — +5V @ 2.58A with no optional devices installed. For each 8741A add 135 mA. For each 220/330 resistor network, add 60 mA. Add the following for each EPROM/ROM installed.

Type	+5.0V Current Requirement	
	1ROM	2ROMS
2758	100 mA	125 mA
2716	100 mA	125 mA
2316E	120 mA	240 mA
2732	40 mA	55 mA
2364	40 mA	55 mA

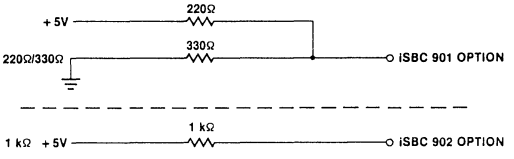
Line Drivers and Terminators

I/O Drivers — The following line drivers are all compatible with the I/O driver sockets on the iSBC 569 Intelligent Digital Controller.

Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

Note I = inverting; NI = non-inverting; OC = open collector.

I/O Terminators — 220Ω/330Ω divider or 1 kΩ pullup



Environmental Characteristics

Operating Temperature — 0°C to 55°C (32°F to 131°F)

Relative Humidity — To 90% without condensation

Reference Manuals

9800845-01 — iSBC 569 Intelligent Digital Controller Board Hardware Reference Manual (NOT SUPPLIED)

9803077 — iSBC 941 Digital Signal Processor User's Guide (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
SBC 569	Intelligent Digital Controller

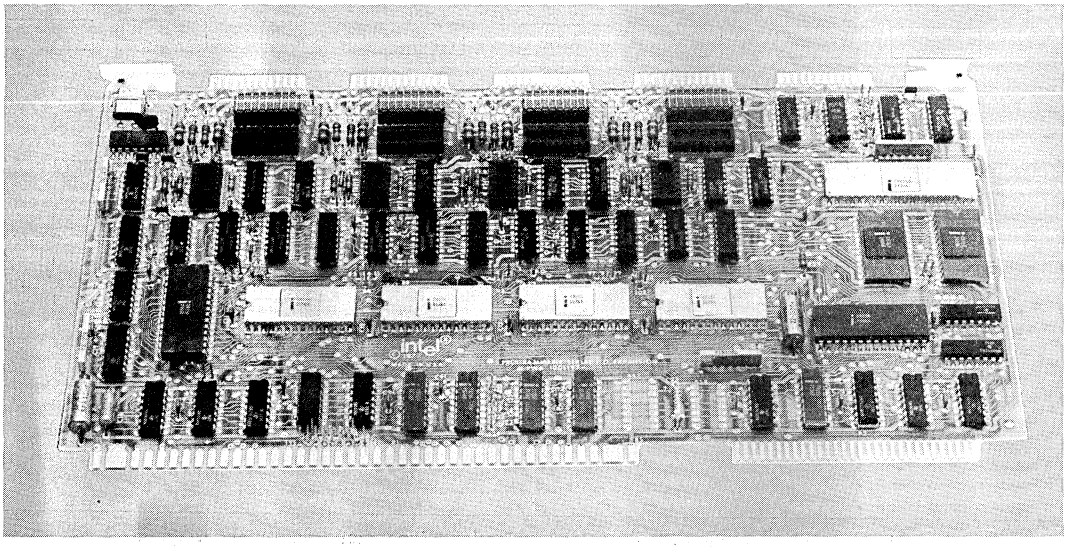
Communication Controllers

9

iSBC™ 534 (or pSBC 534*)
FOUR CHANNEL COMMUNICATION EXPANSION BOARD

- **Serial I/O expansion through four programmable synchronous and asynchronous communications channels**
- **Individual software programmable baud rate generation for each serial I/O channel**
- **Two independent programmable 16-bit interval timers**
- **Sixteen maskable interrupt request lines with priority encoded and programmable interrupt algorithms**
- **Jumper selectable interface register addresses**
- **16-bit parallel I/O interface compatible with Bell 801 automatic calling unit**
- **RS232C/CCITT V.24 interfaces plus 20 mA optically isolated current loop interfaces (sockets)**
- **Programmable digital loopback for diagnostics**
- **Interface control for auto answer and auto originate modems**

The iSBC 534 Four Channel Communication Expansion Board is a member of Intel's complete line of memory and I/O expansion boards. The iSBC 534 interfaces directly to any single board computer via the MULTIBUS to provide expansion of system serial communications capability. Four fully programmable synchronous and asynchronous serial channels with RS232C buffering and provision for 20 mA optically isolated current loop buffering are provided. Baud rates, data formats, and interrupt priorities for each channel are individually software selectable. In addition to the extensive complement of EIA Standard RS232C signals provided, the iSBC 534 provides 16 lines of RS232C buffered programmable parallel I/O. This interface is configured to be directly compatible with the Bell Model 801 automatic calling unit. These capabilities provide a flexible and easy means for interfacing Intel iSBC based systems to RS232C and optically isolated current loop compatible terminals, cassettes, asynchronous and synchronous modems, and distributed processing networks.



*Same product, manufactured by Intel Puerto Rico, Inc.

FUNCTIONAL DESCRIPTION

Communications Interface

Four programmable communications interfaces using Intel's 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) are contained on the board.* Each USART can be programmed by the system software to individually select the desired asynchronous or synchronous serial data transmission technique (including IBM Bisync). The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. Each 8251A provides full duplex, double buffered transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in each USART. Each set of RS232C command lines, serial data lines, and signal ground lines are brought out to 26-pin edge connectors that mate with RS232C flat or round cables.

16-Bit Interval Timers

The iSBC 534 provides six fully programmable and independent BCD and binary 16-bit interval timers utilizing two Intel 8253 programmable interval timers.* Four timers are available to the systems designer to generate baud rates for the USARTs under software control. Routing for the outputs from the other two counters is jumper selectable. Each may be independently routed to the programmable interrupt controller to provide real time clocking or to the USARTs (for applications requiring different transmit and receive baud rates). In utilizing the iSBC 534, the systems designer simply configures, via software, each timer independently to meet system requirements. Whenever a given baud rate or time delay is needed, software commands to the programmable timers select the desired function. Three functions of these timers are supported on the iSBC 534, as shown in Table 1. The contents of each counter may be read at any time during system operation.

Table 1. Programmable Timer Functions

Function	Operation
Interrupt on terminal count	When terminal count is reached an interrupt request is generated. This function is used for the generation of real-time clocks.
Rate generator	Divide by N counter. The output will go low for one input clock cycle and high for N – 1 input clock periods.
Square wave rate generator	Output will remain high for one-half the count and low for the other half of the count.

Interrupt Request Lines

Two independent Intel 8259A programmable interrupt controllers (PIC's) provide vectoring for 16 interrupt levels.* As shown in Table 2, a selection of three priority processing algorithms is available to the system designer. The manner in which requests are serviced may thus be configured to match system requirements. Priority assignments may be reconfigured dynamically via software at any time during system operation. Any combination of interrupt levels may be masked through storage, via software, of a single byte to the interrupt mask register of each PIC. Each PIC's interrupt request

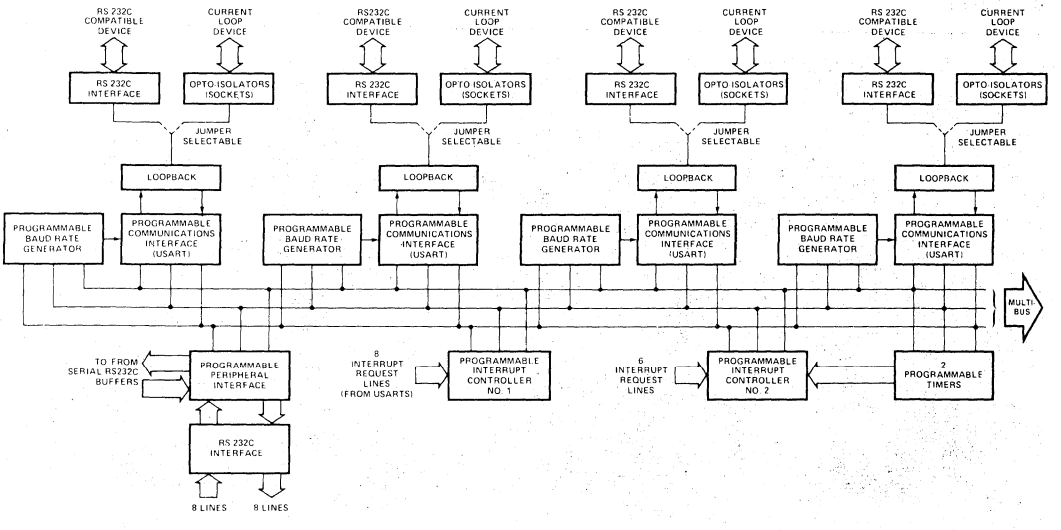


Figure 1. iSBC 534 Four Channel Communications Expansion Board Block Diagram

output line may be jumper selected to drive any of the nine interrupt lines on the MULTIBUS.

Table 2. Interrupt Priority Options

Algorithm	Operation
Fully nested	Interrupt request line priorities fixed at 0 as highest, 7 as lowest.
Auto-rotating	Equal priority. Each level, after receiving service, becomes the lowest priority level until next interrupt occurs.
Specific priority	System software assigns lowest priority level. Priority of all other levels based in sequence numerically on this assignment.

Interrupt Request Generation — As shown in Table 3, interrupt requests may originate from 16 sources. Two jumper selectable interrupt requests (8 total) can be automatically generated by each USART when a character is ready to be transferred to the MULTIBUS system bus (i.e., receive buffer is full) or a character has been transmitted (transmit buffer is empty). Jumper selectable requests can be generated by two of the programmable timers (PITs), and six lines are routed directly from peripherals to accept carrier detect (4 lines), ring indicator, and the Bell 801 present next digit request lines.

Table 3. Interrupt Assignments

Interrupt Request Line	PIC 0	PIC 1
0	PORT 0 R_X RDY	PIT 1 counter 1
1	PORT 0 T_X RDY	PIT 2 counter 2
2	PORT 1 R_X RDY	Ring indicator (all ports)
3	PORT 1 T_X RDY	Present next digit
4	PORT 2 R_X RDY	Carrier detect port 0
5	PORT 2 T_X RDY	Carrier detect port 1
6	PORT 3 R_X RDY	Carrier detect port 2
7	PORT 3 T_X RDY	Carrier detect port 3

Systems Compatibility

The iSBC 534 provides 16 RS232C buffered parallel I/O lines implemented utilizing an Intel 8255A programmable peripheral interface (PPI) configured to operate in mode 0.* These lines are configured to be directly compatible with the Bell 801 automatic calling unit (ACU). This capability allows the iSBC 534 to interface to Bell 801 type ACUs and up to four modems or other serial communications devices. For systems not requiring interface to an ACU, the parallel I/O lines may also be used as general purpose RS232C compatible control lines in system implementation.

*Complete operational details on the Intel 8251A USART, the Intel 8253 Programmable Interval Timer, the Intel 8255 Programmable Peripheral Interface, and the Intel 8259A Programmable Interrupt Controller are contained in the Intel 8080 Microcomputer System User's Manual and 8085 Microcomputer System User's Manual.

SPECIFICATIONS

Serial Communications Characteristics

Synchronous — 5-8 bit characters; internal or external character synchronization; automatic sync insertion.
Asynchronous — 5-8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detection.

Sample Baud Rates[1]

Frequency[2] (kHz, Software Selectable)	Baud Rate (Hz)		
	Synchronous	Asynchronous	
		÷ 16	÷ 64
153.6	—	9600	2400
76.8	—	4800	1200
38.4	38400	2400	600
19.2	19200	1200	300
9.6	9600	600	150
4.8	4800	300	75
6.98	6980	—	110

Notes:
1. Baud rates shown here are only a sample subset of possible software-programmable rates available.Any frequency from 18.75 Hz to 614.4 kHz may be generated utilizing on-board crystal oscillator and 16-bit programmable interval timer (used here as frequency divider).
2. Frequency selected by I/O writes of appropriate 16-bit frequency factor to Baud Rate Register.

Interval Timer and Baud Rate Generator Frequencies

Input Frequency (On-Board Crystal Oscillator) — 1.2288 MHz ± 0.1% (0.813 μs period, nominal)

Function	Single Timer		Dual/Timer Counter (Two Timers Cascaded)	
	Min	Max	Min	Max
Real-Time Interrupt Interval	1.63 μs	53.3 ms	3.26 μs	58.25 minutes
Rate Generator (Frequency)	18.75 Hz	614.4 kHz	0.0029 Hz	307.2 kHz

Interfaces — RS232C Interfaces
EIA Standard RS232C Signals provided and supported:

Carrier detect	Receive data
Clear to send	Ring indicator
Data set ready	Secondary receive data
Data terminal ready	Secondary transmit data
Request to send	Transmit clock
Receive clock	Transmit data

Parallel I/O — 8 input lines, 8 output lines, all signals RS232C compatible
Bus — All signals MULTIBUS system bus compatible

I/O Addressing

The USART, interval timer, interrupt controller, and parallel interface registers of the iSBC 534 are configured as a block of 16 I/O address locations. The location of this block is jumper selectable to begin at any 16-byte I/O address boundary (i.e., 00H, 10H, 20H, etc.).

I/O Access Time

400 ns USART registers
400 ns Parallel I/O registers
400 ns Interval timer registers
400 ns Interrupt controller registers

Compatible Connectors/Cable

Interface	Pins (qty)	Centers (in.)	Mating Connectors	Cable
Bus	86	0.156	Viking 2KH43/9AMK12	N/A
Serial and parallel I/O	26	0.1	3M 3462-0001 or TI H312113	Intel iSBC 955

Compatible Opto-Isolators

Function	Supplier	Part Number
Driver	Fairchild General Electric Monsanto	4N33
Receiver	Fairchild General Electric Monsanto	4N37

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 14 oz (398 gm)

Electrical Characteristics

Average DC Current

Voltage	Without Opto-Isolators	With Opto-Isolators[1]
V_{CC} = +5V	1.9 A, max	1.9 A, max
V_{DD} = +12V	275 mA, max	420 mA, max
V_{AA} = −12V	250 mA, max	400 mA, max

Note

1. With four 4N33 and four 4N37 opto-isolator packages installed in sockets provided to implement four 20 mA current loop interfaces.

Environmental Characteristics

Operating Temperature — 0 °C to +55 °C

Reference Manual

9800450-02 — iSBC 534 Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

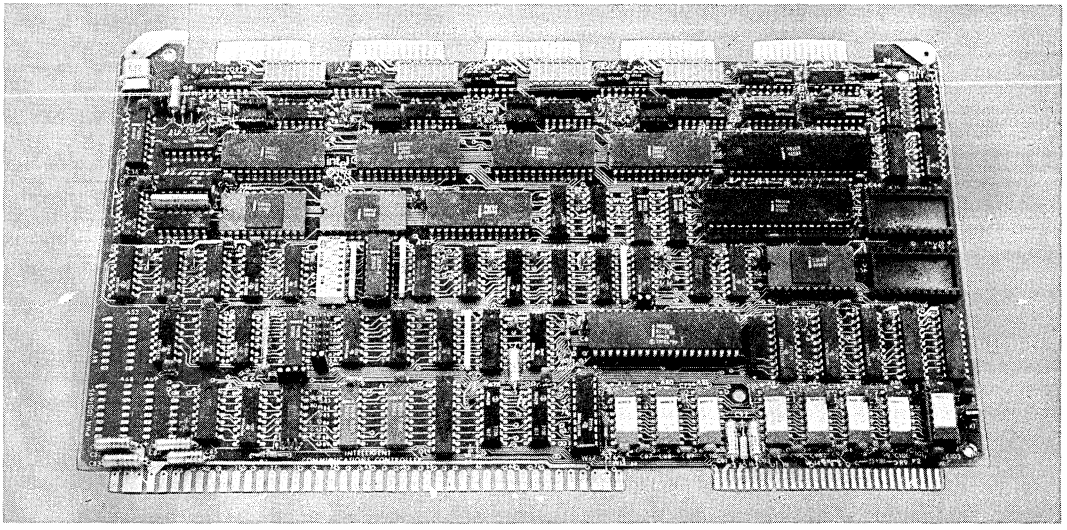
ORDERING INFORMATION

Part Number	Description
SBC 534	Four Channel Communication Expansion Board

iSBC 544
INTELLIGENT COMMUNICATIONS CONTROLLER

- iSBC Communications Controller acting as a single board communications computer or an intelligent slave for communications expansion
- On-board dedicated 8085A Microprocessor providing communications control and buffer management for four programmable synchronous/asynchronous channels
- Sockets for up to 8K bytes of read only memory
- 16K bytes of dual port dynamic read/write memory with on-board refresh
- Extended MULTIBUS addressing permits iSBC 544 board partitioning into 16K-byte segments in a 1-megabyte address space
- Ten programmable parallel I/O lines compatible with Bell 801 Automatic Calling Unit
- Twelve levels of programmable interrupt control
- Individual software programmable baud rate generation for each serial I/O channel
- Three independent programmable interval timer/counters
- Interface control for auto answer and auto originate modem

The iSBC 544 Intelligent Communications Controller is a member of Intel's family of single-board computers, memory, I/O, and peripheral controller boards. The iSBC 544 board is a complete communications controller on a single 6.75 × 12.00 inch printed circuit card. The on-board 8085A CPU may perform local communications processing by directly interfacing with on-board read/write memory, nonvolatile read only memory, four synchronous/asynchronous serial I/O ports, RS232/RS366 compatible parallel I/O, programmable timers, and programmable interrupts.



FUNCTIONAL DESCRIPTION

Intelligent Communications Controller

Two Mode Operation — The iSBC 544 board is capable of operating in one of two modes: 1) as a single board communications computer with all computer and communications interface hardware on a single board; 2) as an "intelligent bus slave" that can perform communications related tasks as a peripheral processor to one or more bus masters. The iSBC 544 may be configured to operate as a stand-alone single board communications computer with all MPU, memory and I/O elements on a single board. In this mode of operation, the iSBC 544 may also interface with expansion memory and I/O boards (but no additional bus masters). The iSBC 544 performs as an intelligent slave to the bus master by performing all communications related tasks. Complete synchronous and asynchronous I/O and data management are controlled by the on-board 8085A CPU to coordinate up to four serial channels. Using the iSBC 544 as an intelligent slave, multichannel serial transfers can be managed entirely on-board, freeing the bus master to perform other system functions.

Architecture — The iSBC 544 board is functionally partitioned into three major sections: I/O, central computer, and shared dual port RAM memory (Figure 1). The I/O hardware is centered around the four Intel 8251A USART devices providing fully programmable serial interfacing. Included here as well is a 10-bit parallel interface compatible with the Bell 801 automatic calling unit, or equivalent. The I/O is under full control of the on-board CPU and is protected from access by system bus masters. The second major segment of the intelligent communications controller is a central computer, with an 8085A CPU providing powerful processing capability. The 8085A together with on-board EPROM / ROM, static RAM, programmable timers/counters, and program-

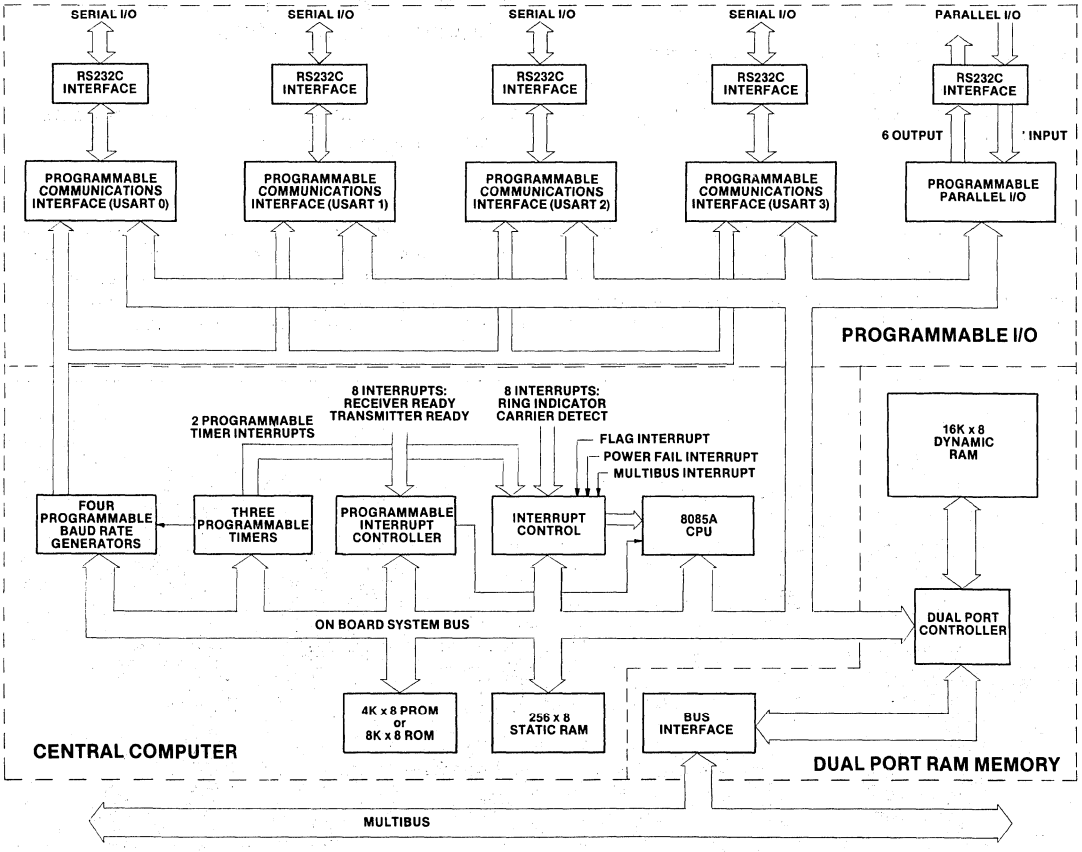


Figure 1. iSBC 544 Intelligent Communications Controller Block Diagram

mable interrupt control provide the intelligence to manage sophisticated communications operations on-board the iSBC 544 board. The timer/counters and interrupt control are also common to the I/O area providing programmable baud rates to the USARTs and prioritizing interrupts generated from the USARTs. The central computer functions are protected for access only by the on-board 8085A. Likewise, the on-board 8085A may not gain access to the system bus when being used as an intelligent slave. When the iSBC 544 is used as a bus master, the on-board 8085A CPU controls complete system operation accessing on-board functions as well as memory and I/O expansion. The third major segment, dual port RAM memory, is the key link between the iSBC 544 intelligent slave and bus masters managing the system functions. The dual port concept allows a common block of dynamic memory to be accessed by the on-board 8085A CPU and off-board bus masters. The system program can, therefore, utilize the shared dual port RAM to pass command and status information between the bus masters and on-board CPU. In addition, the dual port concept permits blocks of data transmitted or received to accumulate in the on-board shared RAM, minimizing the need for a dedicated memory board.

Serial I/O

Four programmable communications interfaces using Intel's 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART) are contained on the board and controlled by the on-board CPU in combination with the on-board interval timer/counter to provide all common communication frequencies. Each USART can be programmed by the system software to individually select the desired asynchronous or synchronous serial data transmission technique (including IBM Bisync). The mode of operation (i.e., synchronous or asynchronous), data format, control character format, parity, and baud rate are all under program control. Each 8251A provides full duplex, double-buffered, transmit and receive capability. Parity, overrun, and framing error detection are all incorporated in each USART. Each channel is fully buffered to provide a direct interface to RS232C compatible terminals, peripherals, or synchronous/asynchronous modems. Each channel of RS232C command lines, serial data lines, and signal ground lines are brought out to 26-pin edge connectors that mate with RS232C flat or round cable.

Parallel I/O Port

The iSBC 544 provides a 10-bit parallel I/O interface controlled by an Intel 8155 Programmable Interface (PPI) chip. The parallel I/O port is directly compatible with an Automatic Calling Unit (ACU) such as the Bell Model 801, or equivalent, and can also be used for auxiliary functions. All signals are RS232C compatible, and the interface cable signal assignments meet RS366 specifications. For systems not requiring an ACU interface, the parallel I/O port can be used for any general purpose interface requiring RS232C compatibility.

Central Processing Unit

Intel's powerful 8-bit n-channel 8085A CPU, fabricated on a single LSI chip, is the central processor for the iSBC 544. The 8085A CPU is directly software compatible with the Intel 8080A CPU. The 8085A contains six 8-bit general purpose registers and an accumulator. The six general purpose registers may be addressed individually or in pairs, providing both single and double precision operators. The minimum instruction execution time is 1.45 microseconds. The 8085A CPU has a 16-bit program counter. An external stack, located within any portion of iSBC 544 read/write memory, may be used as a last-in/first-out storage area for the contents of the program counter, flags, accumulator, and all of the six general purpose registers. A 16-bit stack pointer controls the addressing of this external stack. This stack provides subroutine nesting bounded only by memory size.

EPROM/ROM Capacity

Sockets for up to 8K bytes of nonvolatile read only memory are provided on the iSBC 544 board. Read only memory may be added in 2K-byte increments up to a maximum of 4K bytes using Intel 2716 EPROMs or masked ROMs; or in 4K-byte increments up to 8K bytes maximum using Intel 2732 EPROMs. All on-board EPROM/ROM operations are performed at maximum processor speed.

RAM Capacity

The iSBC 544 contains 16K bytes of dynamic read/write memory using Intel 2117 RAMs. Power for the on-board RAM may be provided on an auxiliary power bus, and memory protect logic is included for RAM battery back-up requirements. The iSBC 544 contains a dual port controller, which provides dual port capability for the on-board RAM memory. RAM accesses may occur from either the on-board 8085A CPU or from another bus master, when used as an intelligent slave. Since on-board RAM accesses do not require the MULTIBUS, the bus is available for concurrent bus master use. Dynamic RAM refresh is accomplished automatically by the iSBC 544 for accesses originating from either the CPU or from the MULTIBUS.

Addressing — On board RAM, as seen by the on-board 8085A CPU, resides at address 8000-BFFF. On-board RAM, as seen by an off-board CPU, may be placed on any 4K-byte address boundary. The iSBC 544 provides extended addressing jumpers to allow the on-board RAM to reside within a one megabyte address space when accessed via the MULTIBUS. In addition, jumper options are provided which allow the user to protect 8K- or 12K-bytes of on-board RAM for use by the on-board 8085 CPU only. This reserved RAM space is not accessible via the MULTIBUS and does not occupy any system address space.

Static RAM — The iSBC 544 board also has 256 bytes of static RAM located on the Intel 8155 PPI. This memory is only accessible to the on-board 8085A CPU and is located at address $7F00_H$-$7FFF_H$.

Programmable Timers

The iSBC 544 board provides seven fully programmable and independent interval timer/counters utilizing two Intel 8253 Programmable Interval Timers (PIT), and the Intel 8155. The two Intel 8253 PITs provide six independent BCD or binary 16-bit interval timer/counters and the 8155 provides one 14-bit binary timer/counter. Four of the PIT timers (BDGO-3) are dedicated to the USARTs providing fully independent programmable baud rates.

Three General Use Timers — The fifth timer (BDG4) may be used as an auxiliary baud rate to any of the four USARTs or may alternatively be cascaded with timer six to provide extended interrupt intervals. The sixth PIT timer/counter (TINT1) can be used to generate interrupt intervals to the on-board 8085A. In addition to the timer/counters on the 8253 PITs, the iSBC 544 has a 14-bit timer available on the 8155 PPI providing a third general use timer/counter (TINT0). This timer output is jumper selectable to the interrupt structure of the on-board 8085A CPU to provide additional timer/counter capability.

Timer Functions — In utilizing the iSBC 544 board, the systems designer simply configures, via software, each timer independently to meet systems requirements. Whenever a given baud rate or interrupt interval is needed, software commands to the programmable timers select the desired function. The on-board PITs together with the 8155 provide a total of seven timer/counters and six operating modes. Mode 3 of the 8253 is the primary operating mode of the four dedicated USART baud rate generators. The timer/counters and useful modes of operation for the general use timer/counters are shown in Table 1.

Interrupt Capability

The iSBC 544 board provides interrupt service for up to 21 interrupt sources. Any of the 21 sources may interrupt the intelligent controller, and all are brought through the interrupt logic to 12 interrupt levels. Four interrupt levels are handled directly by the interrupt processing capability of the 8085A CPU and eight levels are serviced from an Intel 8259A Programmable Interrupt Controller (PIC) routing an interrupt request output to the INTR input of the 8085A (see Table 2).

Interrupt Sources — The 21 interrupt sources originate from both on-board communications functions and the Multibus. Two interrupts are routed from each of the four USARTs (8 interrupts total) to indicate that the transmitter and receiver are ready to move a data byte to or from the on-board CPU. The PIC is dedicated to accepting these 8 interrupts to optimize USART service request. One of eight interrupt request lines are jumper selectable for direct interface from a bus master via the system bus. Two auxiliary timers (TINT0 from 8155 and TINT1 from 8253) are jumper selectable to provide general purpose counter/timer interrupts. A jumper selectable Flag Interrupt is generated to allow any bus master to interrupt the iSBC 544 by writing into the base address of the shared dual port memory accessable to the system. The Flag Interrupt is then cleared by the iSBC 544 when the on-board processor reads the base address. This interrupt provides an interrupt link between

Table 1. Programmable Timer Functions

Function	Operation	Counter
Interrupt on Terminal Count (Mode 0)	When terminal count is reached, an interrupt request is generated. This function is useful for generation of real-time clocks.	8253 TINT1
Rate Generator (Mode 2)	Divide by N counter. The output will go low for one input clock cycle and high for N-1 input clock periods.	8253 BDG4 *
Square-Wave Rate Generator (Mode 3)	Output will remain high until one-half the TC has been completed, and go low for the other half of the count. This is the primary operating mode used for generating a Baud rate clocked to the USARTs.	8253 BDG0-4 TINT1
Software Triggered Strobe (Mode 4)	When the TC is loaded, the counter will begin. On TC the output will go low for one input clock period.	8253 BDG4 * TINT1
Single Pulse	Single pulse when TC reached.	8155 TINT0
Repetitive Single Pulse	Repetitive single pulse each time TC is reached until a new command is loaded.	8155 TINT0

* BDG4 is jumper selectable as an auxiliary baud rate generator to the USARTs or as a cascaded output to TINT1. BDG4 may be used in modes 2 and 4 only when configured as a cascaded output.

Table 2. Interrupt Vector Memory Locations

Interrupt Source		Vector Location	Interrupt Level
Power Fail	TRAP	24_H	1
8253 TINT1 8255 TINT0	RST 7.5	$3C_H$	2
Ring Indicator (1) Carrier Detect	RST 6.5	34_H	3
Flag Interrupt INT0/-INT7/ (1 of 8)	RST 5.5	$2C_H$	4
RXRDY0 TXRDY0 RXRDY1 TXRDY1 RXRDY 2 TXRDY2 RXRDY3 TXRDY3	INTR	Programmable	5-12

(1) Four ring indicator interrupts and four carrier detect interrupts are summed to the RST 6.5 input. The 8155 may be interrogated to inspect any one of the eight signals.

a bus master and intelligent slave (See System Programming). Eight inputs from the serial ports are monitored to detect a ring indicator and carrier detect from each of the four channels. These eight interrupt sources are summed to a single interrupt level of the 8085A CPU. If one of these eight interrupts occur, the 8155 PPI can then be interrogated to determine which port caused the interrupt. Finally, a jumper selectable Power Fail Interrupt is available from the Multibus to detect a power down condition.

8085 Interrupt — Thirteen of the twenty-one interrupt sources are available directly to four interrupt inputs of the on-board 8085A CPU. Requests routed to the 8085A interrupt inputs, TRAP, RST 7.5, RST 6.5 and RST 5.5 have a unique vector memory address. An 8085A jump instruction at each of these addresses then provides software linkage to interrupt service routines located independently anywhere in the Memory. All interrupt inputs with the exception of the TRAP may be masked via software.

8259A Interrupts — Eight interrupt sources signaling transmitter and receiver ready from the four USARTs are channeled directly to the Intel 8259A PIC. The PIC then provides vectoring for the next eight interrupt levels. Operating mode and priority assignments may be reconfigured dynamically via software at any time during system operation. The PIC accepts transmitter and receiver interrupts from the four USARTs. It then determines which of the incoming requests is of highest priority, determines whether this request is of higher priority than the level currently being serviced, and, if appropriate, issues an interrupt to the CPU. The output of the PIC is applied directly to the INTR input of the 8085A. Any combination of interrupt levels may be masked, via software, by storing a single byte in the interrupt mask register of the PIC. When the 8085A responds to a PIC interrupt, the PIC will generate a CALL instruction for each interrupt level. These addressses are equally spaced at intervals of 4 or 8 (software selectable) bytes. Interrupt response to the PIC is software programmable to a 32- or 64-byte block of memory. Interrupt sequences may be expanded from this block with a single 8085A jump instruction at each of these addresses.

Interrupt Output — In addition, the iSBC 544 board may be jumper selected to generate an interrupt from the on-board serial output data (SOD) of the 8085A. The SOD signal may be jumpered to any one of the 8 MULTIBUS interrupt lines (INT0/-INT7/) to provide an interrupt signal directly to a bus master.

Power-Fail Control

Control logic is also included to accept a power-fail interrupt in conjunction with the AC-low signal from the iSBC 635 Power Supply or equivalent.

Expansion Capabilities

When the iSBC 544 board is used as a single board communications controller, memory and I/O capacity may be expanded and additional functions added using Intel MULTIBUS™ compatible expansion boards. In this mode, no other bus masters may be configured in the system. Memory may be expanded to a 65K byte capacity by adding user specified combinations of RAM boards, EPROM boards, or combination boards. Input/output capacity may be increased by adding digital I/O and analog I/O expansion boards. Furthermore, multiple iSBC 544 boards may be included in an expanded system using one iSBC 544 board as a single board communications computer and additional controllers as intelligent slaves.

System Programming

In the system programming environment, the iSBC 544 board appears as an additional RAM memory module when used as an intelligent slave. The master CPU communicates with the iSBC 544 board as if it were just an extension of system memory. Because the iSBC 544 board is treated as memory by the system, the user is able to program into it a command structure which will allow the iSBC 544 board to control its own I/O and memory operation. To enhance the programming of the iSBC 544 board, the user has been given some specific tools. The tools are: 1) the flag interrupt, 2) an on-board RAM memory area that is accessible to both an off-board CPU and the on-board 8085A through which a communications path can exist, and 3) access to the bus interrupt line.

Flag Interrupt — The Flag Interrupt is generated anytime a write command is performed by an off-board CPU to the base address of the iSBC 544 board's RAM. This interrupt provides a means for the master CPU to notify the iSBC 544 board that it wishes to establish a communications sequence. In systems with more than one intelligent slave, the flag interrupt provides a unique interrupt to each slave outside the normal eight MULTIBUS interrupt lines (INT0/-INT7/).

On-Board RAM — The on-board 16K byte RAM area that is accessible to both an off-board CPU and the on-board 8085A can be located on any 4K boundary in the system. The selected base address of the iSBC 544 RAM will cause a flag interrupt when written into by an off-board CPU.

Bus Access — The third tool to improve system operation as an intelligent slave is access to the Multibus interrupt lines. The iSBC 544 board can both respond to interrupt signals from an off-board CPU, and generate an interrupt to the off-board CPU via the MULTIBUS.

System Development Capability

The development cycle of iSBC 544 board based products may be significantly reduced using the Intellec series microcomputer development systems. The Intellec resident macroassembler, text editor, and system monitor greatly simplify the design, development and debug of iSBC 544 system software. An optional ISIS-II diskette operating system provides a linker, object code locater, and library manager. A unique in-circuit emulator (ICE-85) option provides the capability of developing and debugging software directly on the iSBC 544 board.

SPECIFICATIONS

Serial Communications Characteristics

Synchronous — 5-8 bit characters; automatic sync insertion; parity.

Asynchronous — 5-8 bit characters; break character generation; 1, 1½, or 2 stop bits; false start bit detection; break character detection.

Baud Rates

Frequency (KHz)[1] (Software Selectable)	Baud Rate (Hz)[2]	
	Synchronous	Asynchronous
		÷16 ÷64
307.2		19200 4800
153.6	—	9600 2400
76.8	—	4800 1200
55.8	55800	3500 870
38.4	38400	2400 600
19.2	19200	1200 300
9.6	9600	600 150
4.8	4800	300 75
6.98	6980	— 110

Notes:
1) Frequency selected by I/O writes of appropriate 16-bit frequency factor to Baud Rate Register.
2) Baud rates shown here are only a sample subset of possible software programmable rates available. Any frequency from 18.75 Hz to 614.4 KHz may be generated utilizing on-board crystal oscillator and 16-bit Programmable Interval Timer (used here as a frequency divider).

8085A CPU

Word Size — 8, 16 or 24 bits/instruction; 8 bits of data

Cycle Time — 1.45/usec ± .1% for fastest executable instruction; i.e. four clock cycles.

Clock Rate — 2.76 MHz ± .1%

System Access Time

Dual port memory — 740 nsec

Note: Assumes no refresh contention

Memory Capacity

On-Board ROM/PROM — 4K, or 8K bytes of user installed ROM or EPROM.

On-Board Static RAM — 256 bytes on 8155.

On-Board Dynamic RAM (on-board access) — 16K bytes. Integrity maintained during power failure with user-furnished batteries (optional).

On-Board Dyanmic RAM (MULTIBUS access) — 4K, 8K, or 16K-bytes available to bus by switch selection.

Memory Addressing

On-Board ROM/PROM — 0-0FFF (using 2716 EPROMs or masked ROMs); 0-1FFF (using 2732 EPROMs)

On-Board Static Ram — 256 bytes: 7F00-7FFF

On-Board Dynamic RAM (on-board access) — 16K bytes: 8000-BFFF.

On-Board Dynamic RAM (MULTIBUS access) — any 4K increment 00000-FF000 which is switch and jumper selectable. 4K- 8K- or 16K-bytes can be made available to the bus by switch selection.

I/O Capacity

Serial — 4 programmable channels using four 8251A USARTs.

Parallel — 10 programmable lines available for Bell 801 ACU, or equivalent use. Two auxiliary jumper selectable signals.

I/O Addressing

On-Board Programmable I/O

Port	Data	Control
USART 0	D0	D1
USART 1	D2	D3
USART 2	D4	D5
USART 3	D6	D7
8155 PPI	E9 (Port A)	E8
	EA (Port B)	
	EB (Port C)	

Interrupts

Addresses for 8259A Registers (Hex notation, I/O address space)

E6 Interrupt request register
E6 In-service register
E7 Mask register
E6 Command register
E7 Block address register
E6 Status (polling register)

Note: Several registers have the same physical address: Sequence of access and one data bit of the control word determines which register will respond.

Interrupt levels routed to the 8085 CPU automatically vector the processor to unique memory locations:

24 TRAP
3C RST 7.5
34 RST 6.5
2C RST 5.5

Timers

Addresses for 8253 Registers (Hex notation, I/O address space)

Programmable Interrupt Timer One

D8	Timer 0	BDG0
D9	Timer 1	BDG1
DA	Timer 2	BDG2
DB	Control register	

Programmable Interrupt Timer Two

DC	Timer 0	BDG3
DD	Timer 1	BDG4
DE	Timer 2	TINT1
DF	Control register	

Address for 8155 Programmable Timer

E8 Control
EC Timer (LSB) TINT0
ED Timer (MSB) TINT0

Input frequencies — Jumper selectable reference 1.2288 MHz ± .1% (.814 usec period nominal) or 1.843 MHz ± .1% crystal (0.542 usec period, nominal)

Output Frequencies (at 1.2288 MHz)

Function	Single timer/counter		Dual timer/counter (two timers cascaded)	
	Min	Max	Min	Max
Real-time interrupt interval	1.63 usec	53.3 usec	3.26 usec	58.25 min
Rate Generator (frequency)	18.75 Hz	614.4 KHz	0.00029 Hz	307.2 KHz

Interfaces

Serial I/O — EIA Standard RS232C signals provided and supported:

Carrier Detect
Clear to Send
Data Set Ready
Data Terminal Ready
Request to Send
Receive Clock
Receive Data
Ring Indicator
Secondary Receive Data *
Secondary Transmit Data *
Transmit Clock
Transmit Data
DTE Transmit Clock

* Optional if parallel I/O port is not used as Automatic Calling Unit.

Parallel I/O — Four inputs and eight outputs (includes two jumper selectable auxiliary outputs). All signals compatible with EIA Standard RS232C. Directly compatible with Bell Model 801 Automatic Calling Unit, or equivalent.

MULTIBUS — Compatible with iSBC MULTIBUS.

On-Board Addressing

All communications to the parallel and serial I/O ports, to the timers, and to the interrupt controller, are via read and write commands from the on-board 8085A CPU.

Auxiliary Power

An auxiliary power bus is provided to allow separate power to RAM for systems requiring battery backup of read/write memory. Selection of this auxiliary RAM power bus is made via jumpers on the board.

Connectors

Interface	Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	Viking 2KH43/9AMK12
Parallel I/O	50	0.1	3M 3415-000 or AMP 88083-1
Serial I/O	26	0.1	3M 3462-000 or AMP 88373-5

Memory Protect

An active-low TTL compatible memory protect signal is brought out on the auxiliary connector which, when asserted, disables read/write access to RAM memory on the board. This input is provided for the protection of RAM contents during the system power-down sequences.

Bus Drivers

Function	Characteristic	Sink Current (mA)
Data	Tri-state	50
Address	Tri-state	15
Commands	Tri-state	32

Note: Used as a master in the single board communications computer mode.

Physical Characteristics

Width: 30.48 cm (12.00 inches)
Depth: 17.15 cm (6.75 inches)
Thickness: 1.27 cm (0.50 inch)
Weight: 3.97 gm (14 ounces)

Electrical Characteristics

DC Power Requirements

	Current Requirements			
Configuration	V_{CC} = +5V ±5% (max)	V_{DD} = ±12V ±5% (max)	V_{BB} = −5V(3) ±5% (max)	V_{AA} = −12V ±5% (max)
With 4K EPROM (using 2716)	I_{CC} = 3.4 max max	I_{DD} = 350mA max	I_{BB} = 5mA max	I_{AA} = 200mA max
Without EPROM	3.3A max	350 mA max	5 mA max	200 mA max
RAM only (1)	390 mA max	176 mA max	5 mA max	—
RAM(2) refresh only	390 mA max	20 mA max	5 mA max	

Notes: 1. For operational RAM only, for AUX power supply rating.
2. For RAM refresh only. Used for battery backup requirements. No RAM accessed.
3. V_{BB} is normally derived on-board from V_{AA}, eliminating the need for a V_{BB} supply. If it is desired to supply V_{BB} from the bus, the current requirement is as shown.

Environmental Characteristics

Operating Temperature: 0°C to 55°C (32°F to 131°F)
Relative Humidity: To 90% without condensation

Reference Manual

9800616B — iSBC 544 Intelligent Communication Controller Board Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

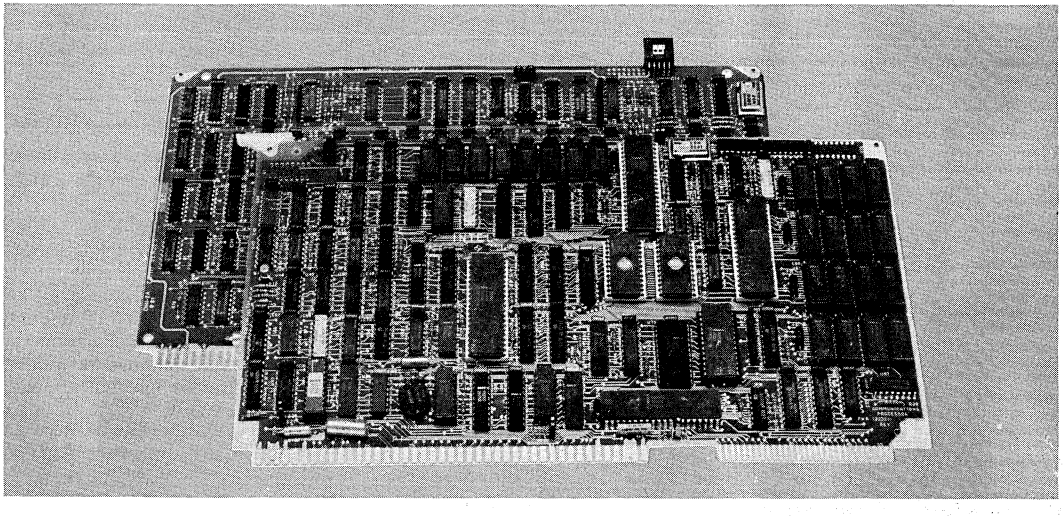
ORDERING INFORMATION

Part Number	Description
iSBC 544	Intelligent Communications Controller

iSBC™ 550
ETHERNET™ COMMUNICATIONS CONTROLLER

- **Meets the version 1.0 tri-corporate Ethernet specification**
- **Ethernet data link layer support**
  - **Data encapsulation**
  - **Framing and packet control**
  - **Buffer management**
- **Ethernet physical link layer support**
  - **Serial/deserialization**
  - **10 Mbits per second data rate**
  - **CRC generation/check**
  - **Carrier-sense multiple-access with collision detection (CSMA/CD)**
  - **Transceiver interface compatibility**
- **Easy-to-use MULTIBUS® interprocessor protocol supported in firmware**
- **Power-up confidence test assures integrity of on-board memory and programmable LSI**
- **Traffic, errors and collision information maintained for network management**
- **Excellent foundation for Ethernet local area end-to-end network**

The iSBC 550 Ethernet Communications Controller meets the tri-corporate (DEC, Xerox, Intel) specification for Ethernet local area networks. All the functions of the Ethernet data link layer and physical link layer are provided on two 6.75 × 12″ circuits boards and associated firmware. The MULTIBUS compatible controller can be utilized as the foundation for a single board computer (iSBC)-based Ethernet local area network or as a prototype for Intel® 8085, iAPX™ 88, or iAPX 86 component-based Ethernet applications. The iSBC 550 controller's firmware (supplying the Ethernet and system interface) has an easy-to-use MULTIBUS Interprocessor Protocol (MIP) facility, which is readily accessed from another iSBC Board using a custom run-time software system or Intel's iRMX™ 80/88/86 Real-Time Executive software and the iMMX™ 800 (MULTIBUS Message Exchange) software package. The Ethernet data link functions are divided between the processor board which provides the data link layer's software to control the data encapsulation and the link management, and the serial/deserialization (SerDes) board which provides the 10-MBit per second serial interface to the Ethernet transceiver.



The following are trademarks of Intel Corporation and may be used to describe Intel products: CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE, ICS, iAPX and iMMX. 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 CORPORATION, 1981
*EthernetTM is a trademark of Xerox Corporation

June, 1981
143882

FUNCTIONAL DESCRIPTION

The iSBC 550 Ethernet Communications Controller is a two-board MULTIBUS-compatible set that offers high-speed Ethernet-compatible data transfer between digital devices operating at a 10-Mbit per sec data rate. The iSBC 550 controller can effectively support the needs of local area network applications, such as office automation, distributed data processing, factory data collection, research data collection, intelligent terminal and other EDP-related products.

Ethernet Specification

The Ethernet network is a local area network concept that is jointly being supported by Intel Corporation, Digital Equipment Corporation, and Xerox Corporation. The network is designed to link systems over a distance of up to 2500 meters using an available 50-ohm coaxial cable. Several hundred stations may be connected to the cable which supports a data rate of 10 Megabits per second. The data is encapsulated in a packet message format. The data signal is a base-band, Manchester-encoded type that is self-synchronizing.

The jointly developed Ethernet specification, "The Ethernet, A Local Area Network Data Link Layer and Physical Link Layer Specification, Version 1.0, September 30, 1980", precisely defines the two lower layers of a local area network architecture where the system is a series of independent layers. The lowest layer, the physical link layer, is concerned with coaxial cable interface. The data link layer supports the peer protocol's statistical contention resolution (CSMA/CD) and link management functions. All additional network layers are defined by the user during the implementation of the application-specific layers.

Ethernet Data Link Layer Support

The iSBC 550 processor board provides the data link layer's software to control the data encapsulation and the link management, including frame delimitation, address handling, error detection, and collision handling. After the iSBC 550 processor board is initialized upon system start-up or reset, the data link firmware is ready to service the local area network commands. An example of a command structure sent the iSBC controller to receive a packet of data from the Ethernet link is shown in Figure 1. The message passed via the MIP (MULTIBUS Interprocessor Protocol) interface is composed of two parts, the iSBC 550 controller information (including the command and associated data), and the required Ethernet information.

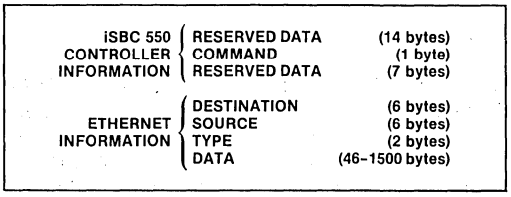


Figure 1. Data Link for SUPPLYBUF Command Format

Shown in Table 1 are eight external Ethernet controller commands available to a user's application via the MIP interface. The commands manage the Ethernet multicast address recognition, message type connection, message flow, and overall network statistics.

Table 1. External Controller Commands

Command	Function
CONNECT	Indicates the data link message TYPE to be *connected* to user program.
DISCONNECT	*Disconnects* the data link TYPE from the user's application.
ADDMCID	*Adds* a multicast ID for recognition.
DELETEMCID	*Delete* the specified multicast ID.
TRANSMIT	*Transmit* a data packet to the Ethernet link.
SUPPLYBUF	*Supplies* a buffer for packet reception from the Ethernet link ("receive" function).
READ	*Read* the statistical variables maintained by data link layer.
READC	*Read* and *clear* the statistical variables.

Ethernet Physical Link Layer Support

The Serialization/Deserialization (SerDes) board provides the required electrical characteristics of the physical link layer of the Ethernet architecture for a transceiver interface. The transceiver is a device physically attached to the coax cable which does signal conditioning for transmitting and receiving.

Many major functions are controlled by the SerDes board. These functions include serialization/deserialization, packet framing, Manchester encoding/decoding, transmit data flow control, receive data flow control, destination address decoding for received message, CRC generation and

checking, and diagnostics for CRC error, loopback, transmit timeout, and CSMA/CD (Carrier-Sense Multiple-Access with Collision-Detection).

Easy-To-Use Interface

One of the iSBC 550 controller boards is an iAPX 88-based processor board which has firmware support for the user's application interface. The programmatic interface utilizes the MULTIBUS Interprocessor Protocol (MIP) interface to the processor board. This interface is concerned with the message-passing protocol between multiple-processors. The iMMX 800 (MULTIBUS Message Exchange) software supports the MIP interface and offers a convenient quick-start method for users of Intel's iRMX 80, iRMX 88 executives and iRMX 86 operating system products for an Ethernet-based application.

Confidence Test

An effective diagnostic function is implemented in firmware on the processor board. This function is invoked at system initialization during both power-up and system reset time. These functions include: packet CRC checking, memory test, controller loopback, and other error tests. The tests provide a fundamental level of controller integrity.

Network Statistics

Statistics maintained by the data link firmware include packet traffic counts, collision information and error totals. This information can be effectively utilized by the user's application to understand the network's operation.

End-To-End Networking Foundation

The iSBC 550 controller provides the foundation data link layer and the physical link layer for a local area network architecture. Typically, the higher levels are user-defined and include the transport and the session control layers. The transport control layer is concerned with the end-to-end communications and the virtual channel connection via a port-to-port address. The session control layer provides the process-to-process control function which includes symbolic name binding and the establishment of the virtual connection via the transport control layer. In addition, the session control provides the specific error and recovery control responsible for message delivery.

The higher levels of the local area network architecture (see Figure 2) which use the data link layer are outside of the Ethernet standard, but can be implemented quickly on companion iSBC boards (e.g., iSBC 80/24, iSBC 88/25, iSBC 86/12A) running under the iRMX 80/88/86 Real-Time Multitasking Executives, respectively, and associated iMMX MULTIBUS Message Exchange (iMMX 800) software. Special iSBC 550 device driver software compatible with the iRMX 86 and iRMX 88 file systems is provided in the iMMX 800 package.

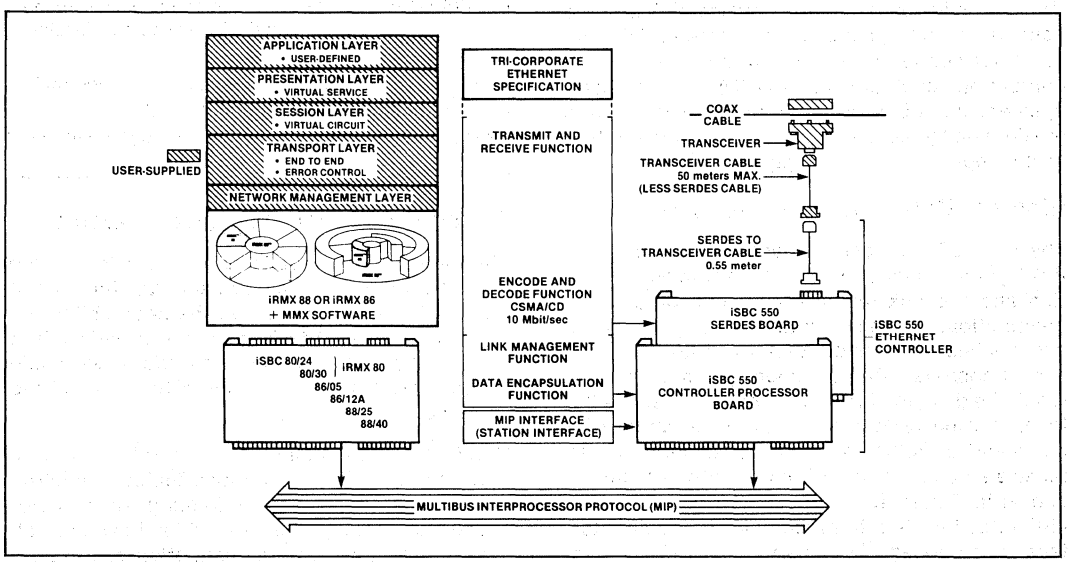


Figure 2. Ethernet Architecture and Implementation

SPECIFICATIONS

Memory Addressing Capability

MULTIBUS System Bus — (00000-EFFFF)

Ethernet I/O Channels

One Ethernet electrically-compatible transceiver line on the SerDes board.

Interface Specifications

MULTIBUS System Bus — All signals TTL compatible.

Transceiver — All signals Ethernet specifications transceiver compatible.

Serial Communications Characteristics

Bit Serial Frame — Provides 64-bit *preamble*, 48-bit *destination* address, 48-bit *source* address, 16-bit *type*, 46-1500 bytes for *data*, and a *frame* check sequence of 32 bits.

Ethernet Network Specifications Supported

Coax Cable Length — 500-meter max.
Transceiver Cable Length — 50-meter max.
Number of Stations — 100 max.
Baud Rate — 10-Mbit/sec

System Clock

5.00 MHz, ± 0.1%

Physical Characteristics (Both Boards)

Width — 12.00 in. (30.48 cm) (each board)
Height — 6.75 in. (17.15 cm) (each board)
Depth — 0.5 in. (1.27 cm) (each board)
Weight — 3.5 lb (1.6 kg) (both boards)

SerDes to Transceiver Cable

Length — 0.55 meter (22 in.). Four pair twisted-wire cable with SerDes connector and transceiver interface connector.

Electrical Characteristics

Power requirements for both boards
+ 5 VDC @ 9.0A max.
+ 12 VDC @ 0.5A max.

Environmental Characteristics

Operating Temperature — 0°C to 55°C

Relative Humidity — To 90% (without condensation)

Connectors

Interface	Pins (qty)	Centers (in.)	Mating Connectors
MULTIBUS System	86	0.156	Viking 2KH43/9AMK12
SerDes Edge Connector	10	0.1	AMP 87631-5 Housing AMP 87195-9 Pins
Transceiver	15	0.1	Cinch Type DA 51220-1

Reference Manuals

121746 — iSBC 550 Ethernet Communications Controller Hardware Reference Manual (NOT SUPPLIED)

121769 — The Ethernet Communications Controller Programmer's Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051.

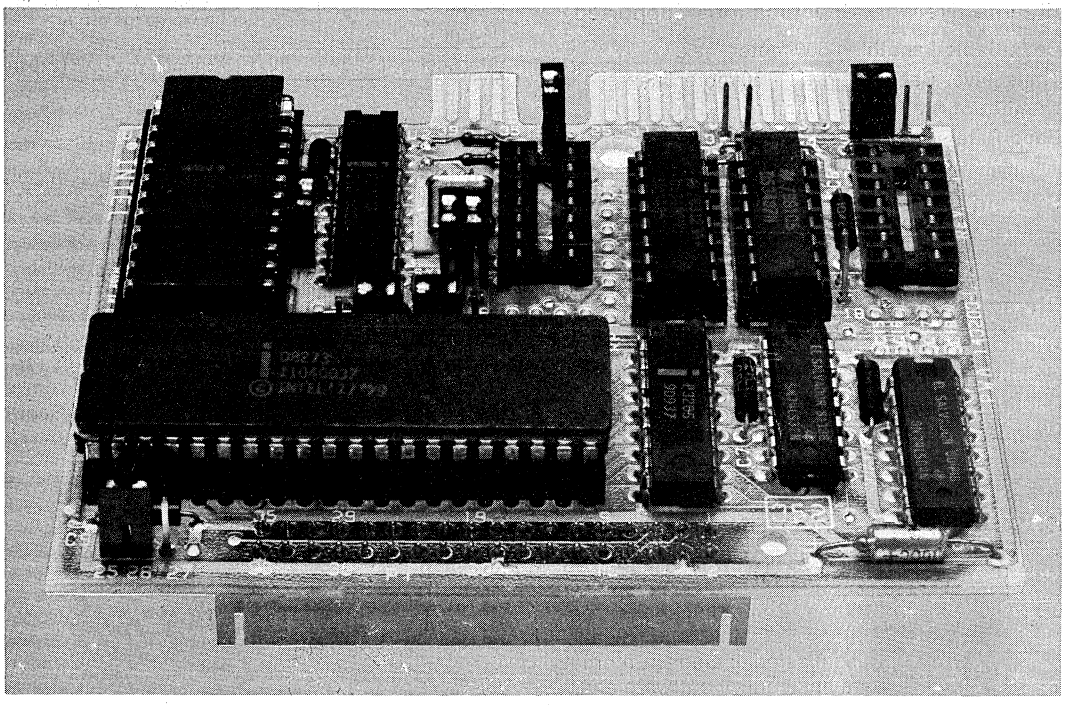
ORDERING INFORMATION

Part Number	Description
SBC 550	Ethernet Communications Controller for 10 Mbit/sec coaxial transmission. Includes Ethernet data link control software and cable to transceiver.

iSBX™ 352
BIT SERIAL COMMUNICATIONS MULTIMODULE™ BOARD

- **Provides an HDLC/SDLC half/full-duplex communications channel for iSBX™ bus compatible micro-computers**
- **Supports RS232C (including modem support) or RS449/422A interface**
- **Single +5V when configured for RS449/422A interface**
- **Software programmable baud rate generation up to 64K baud synchronous and 9.6K baud self-clocking**
- **Supports synchronous or self-clocking NRZI point-to-point, multidrop and self-clocking NRZI SDLC loop data link interfaces**

The Intel iSBX 352 Bit Serial Communications MULTIMODULE board offers incremental on-board I/O expansion support for ISO/CCITT's HDLC or IBM's SDLC communication. Plugging directly into any iSBX bus compatible host board, the iSBX 352 module provides one RS232C or RS449/422A programmable bit serial communications channel with software selectable baud rates (up to 64K baud for half-duplex synchronous operations). Data link interfaces supported are: synchronous point-to-point, multidrop and SDLC loop. The phase lock loop feature provides NRZI self-clocking 9.6K baud operation.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1981

October, 1981
Order Number: 210218-001

FUNCTIONAL DESCRIPTION

Communications Interface

The iSBX 352 module uses the Intel 8273 Programmable HDLC/SDLC Protocol Controller. The iSBX 352 module provides one bit-serial communications channel for iSBX bus compatible host microcomputers. (See Figure 1.) An iSBC microcomputer or MULTIBUS-based application is easily connected to an HDLC/SDLC point-to-point, multidrop, or an SDLC loop configuration.

The High-Level Data Link Control (HDLC) is the International Standards Organization (ISO) standard discipline used to implement X.25 packet switching communications. The Synchronous Data Link Control (SDLC) is an IBM communication protocol used to implement the System Network Architecture (SNA). Both protocols, HDLC and SDLC, are bit oriented, code independent, and support full-duplex operations.

Data Link Interface

The control lines, serial data lines and signal ground lines are brought out to the double edge connector of the iSBX 352 module and are configurable for RS232C or RS449/422A interface (see Figure 2).

Addressing an iSBX 352 board by using a port address, the program performs the 8-bit data transfer required, using buffered or non-buffered transmit/receive and abort sequences.

Serial data transfer control is provided by the 8273 controller of the iSBX 352 module which interfaces the parallel iSBX bus to the serial channel. During a transmit sequence, the iSBX 352 module accepts data and commands from the iSBX bus interface, translates and formats the data into HDLC/SDLC protocol formats, provides the proper RS232C or RS422A interface control signals, and passes data onto the serial channel. The receive operation is the inverse of the previous sequence.

Data Link Configurations

The supported data link configurations are shown in Table 1. The following example configurations provide an overview and a figure for five typical data link configurations:

Table 1. iSBX™ 352 Supported Configurations

Connection	Synchronous		Asynchronous	
	Modem	Direct	Modem*	Direct
point-to-point	X	X	X	X
multidrop	X	X	X	X
loop	NA	NA	X	X

* Modem should not respond to a break.

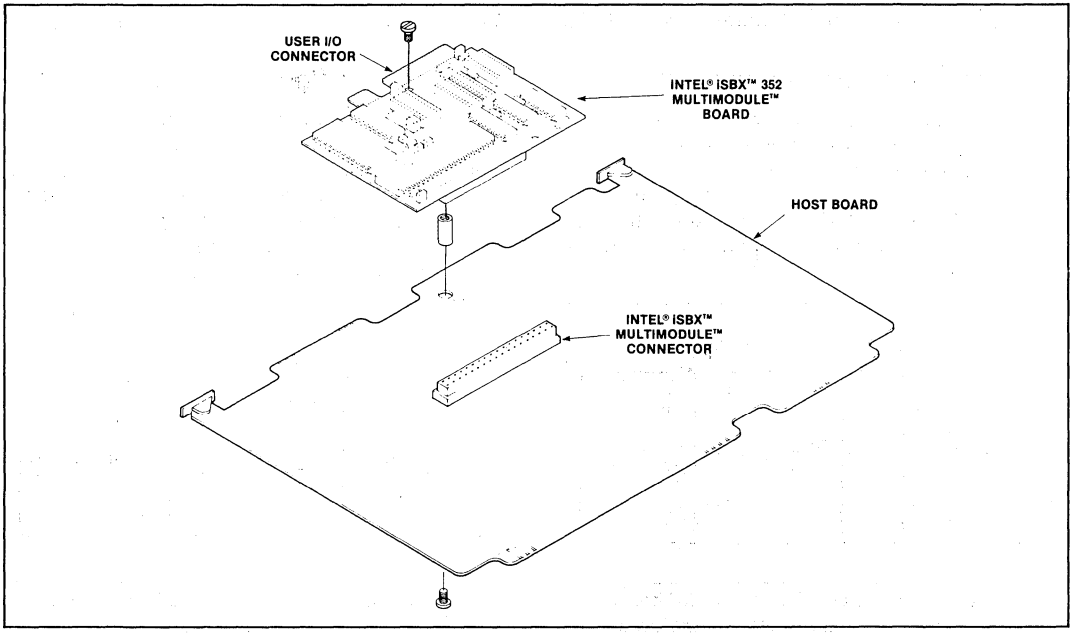


Figure 1. Installation of iSBX™ 352 MULTIMODULE™ Board on a Host Board

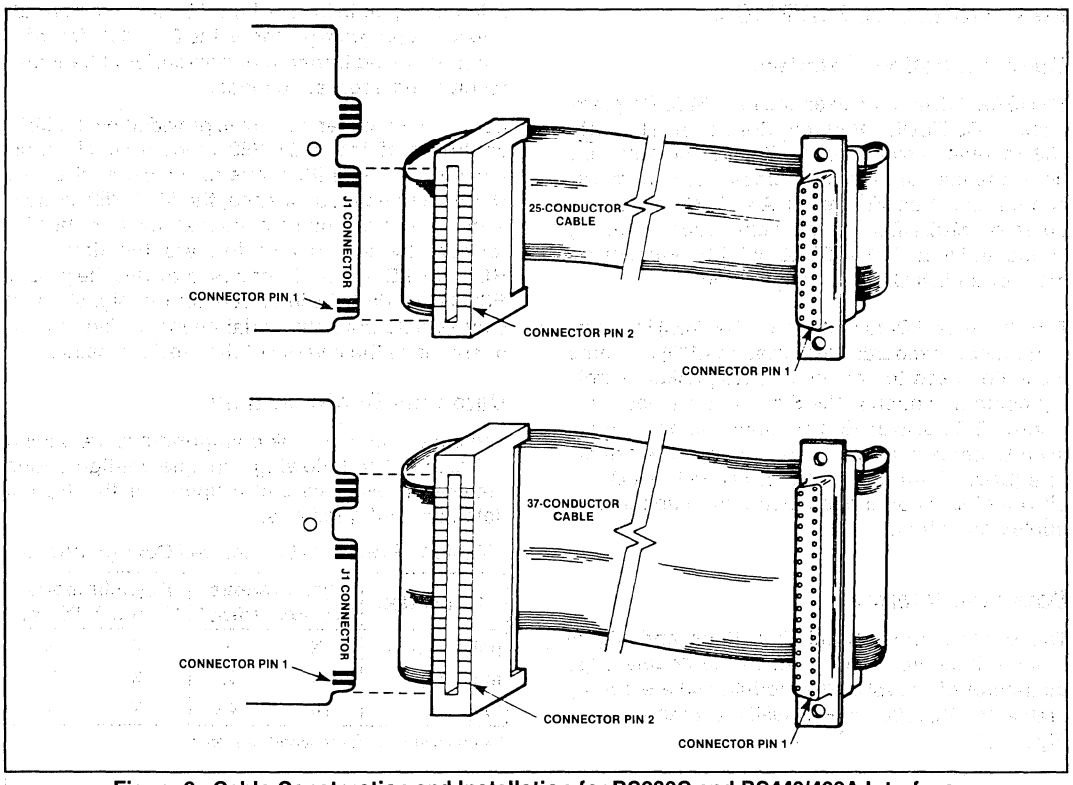


Figure 2. Cable Construction and Installation for RS232C and RS449/422A Interface

SYNCHRONOUS POINT-TO-POINT INTERFACE

Figure 3 shows a synchronous point-to-point mode of operation for the iSBX 352 module. This RS232C example uses a modem for generation of the receive clock for coordination of the data transfer. The iSBX 352 module generates the transmit synchronizing clock for synchronous transmission.

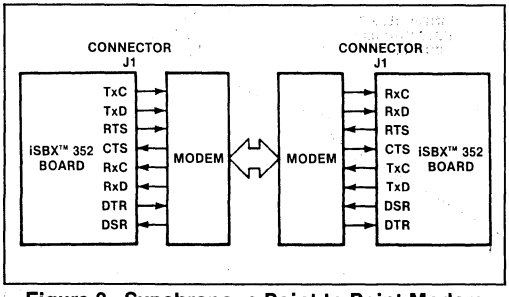


Figure 3. Synchronous Point-to-Point Modem Interface Configuration Example - RS232C

SELF-CLOCKING POINT-TO-POINT INTERFACE

The iSBX 352 module is used in an asynchronous mode interface when configured as shown in Figure 4. The point-to-point RS232C example uses the self-clocking mode interface for NRZI encoding/decoding of data. The digital phase lock loop allows operation of the interface in either half-duplex or full-duplex implementation with or without modems.

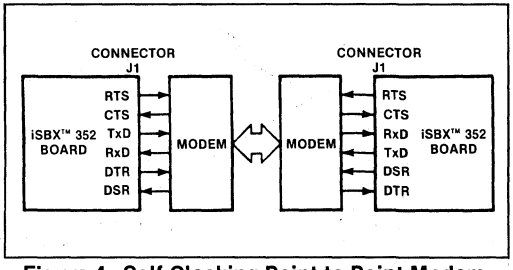


Figure 4. Self-Clocking Point-to-Point Modem Interface Configuration Example - RS232C

SYNCHRONOUS MULTIDROP

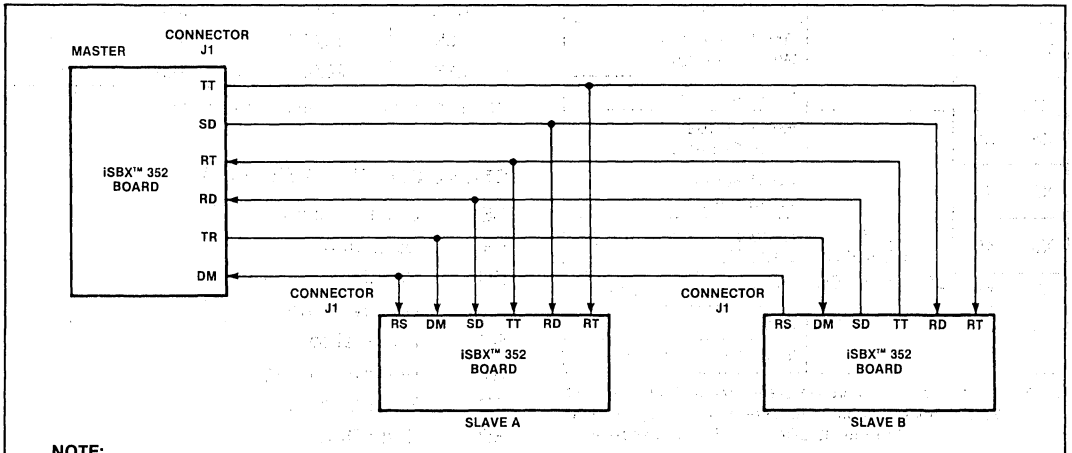
The iSBX 352 MULTIMODULE is used as both a master and a slave node in the RS449/422A example shown in Figure 5. This synchronous multidrop application is effective for high-speed data transfers between slave stations and a central master station.

ASYNCHRONOUS SELF-CLOCKING MULTIDROP

The iSBX 352 MULTIMODULE example in Figure 6 shows a master and multiple slaves in a multidrop configuration. This self-clocking example uses the 8273 digital phase lock loop and NRZI data encoding.

SDLC Loop

The SDLC self-clocking loop configuration shown in Figure 7 permits longer networks since each secondary slave station is a repeater set in one-bit-delay mode. The data sent out by the primary station (the loop controller) are relayed bit-for-bit through each secondary station and finally back to the master station.



NOTE:
The last slave device in the system must contain termination resistors on all signal lines received by the slave board. The master device must contain resistors on all received signal lines.

Figure 5. Synchronous Multidrop Network Configuration Example - RS422A

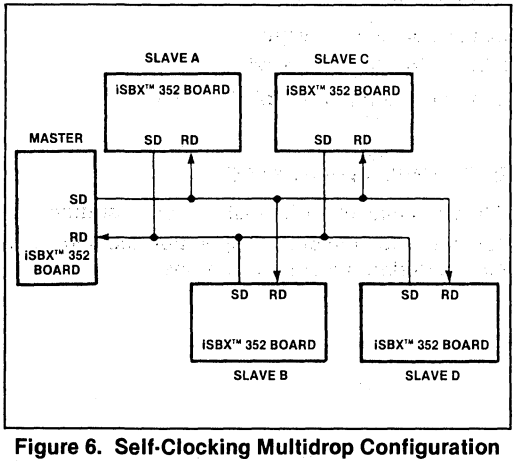


Figure 6. Self-Clocking Multidrop Configuration Example - RS422A

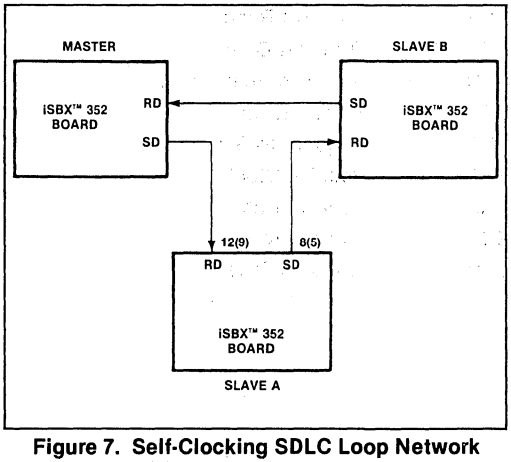


Figure 7. Self-Clocking SDLC Loop Network Configuration Example

SPECIFICATIONS

Data Size

8 Bits

I/O Port Addresses

Port Address		Device Selected	Function Performed
8-bit	16-bit		
X0	X0	8254-2 PIT	Read Counter 0 Write Counter 0
X1	X2		Read Counter 1 Write Counter 1
X2	X4		Read Counter 2 Write Counter 2
X3	X6		Write Control
X4	X8	8273 HDLC/SDLC CONTROLLER	Read Status Write Command
X5	XA		Read Result Write Parameter
X6	XC		Read Transmit Interrupt Write Reset
X7	XE		Read Receive Interrupt
Y0	Y0		Read Receive Data
Y4	Y8		Write Transmit Data

NOTE: Refer to the Hardware Reference Manual for your host iSBC™ microcomputer to determine the upper digit (either X or Y) of the MULTIMODULE™ port address.

Interfaces

iSBX™ BUS — All signals TTL compatible

SERIAL RS232C SIGNALS

CTS	Clear to Send
DSR	Data Set Ready
DTE TXC	Transmit Clock
DTR	Data Terminal Ready
FG	Frame Ground
RTS	Request to Send
RXC	Receive Clock
RXD	Receive Data
SG	Signal Ground
TXD	Transmit Data

RATE GENERATOR FREQUENCIES

Baud Rate bits/sec	8254-2 Divide Count		
	Synchronous	Self-Clocking	
64K	125	TX Clock	32X Clock
56K	143	—	—
48K	167	—	—
19.2K	417	—	—
9.6K	833	833	26
4.8K	1,667	1,667	52
2.4K	3,333	3,333	104
1.2K	6,667	6,667	208
0.6K	13,333	13,333	417
0.3K	26,667	26,667	833

NOTE: All numbers are in decimal notation.

SERIAL RS449/422A SIGNALS

CS	Clear to Send
DM	Data Mode
RC	Receive Common
RD	Receive Data
RS	Request to Send
RT	Receive Timing
SC	Send Common
SD	Send Data
SG	Signal Ground
TR	Terminal Ready
TT	Terminal Timing

OPERATING SPEEDS

24 MHz on-board crystal
8 MHz clocking of the 8254-2 PIT
4 MHz clocking of the 8273 Device

DATA THROUGHPUT SPEED

64K baud maximum for half-duplex operation

48K baud for full-duplex operation issuing commands during transmit operations

SERIAL INTERFACE CONNECTORS

Configuration	Mode[2]	MULTIMODULE™ Edge Connector	Cable	Connector
RS232C	DTE	26-pin[5], 3M-3462-0001	3M[3]-3349/25	25-pin[7], 3M-3482-1000
RS232C	DCE	26-pin[5], 3M-3462-0001	3M[3]-3349/25	25-pin[7], 3M-3483-1000
RS449	DTE	40-pin[6], 3M-3464-0001	3M[4]-3349/37	37-pin[1], 3M-3502-1000
RS449	DCE	40-pin[6], 3M-3464-0001	3M[4]-3349/37	37-pin[1], 3M-3503-1000

NOTES:
1. Cable housing 3M-3485-4000 may be used with the connector.
2. DTE - Data Terminal Equipment mode (male connector); DCE - Data Set Equipment mode (female connector).
3. Cable is tapered at one end to fit the 3M-3462 connector.
4. Cable is tapered to fit 3M-3464 connector.
5. Pin 26 of the edge connector is not connected to the flat cable.
6. Pins 38, 39, and 40 of the edge connector are not connected to the flat cable.
7. May be used with the cable housing 3M-3485-1000.

Electrical Characteristics

DC POWER REQUIREMENTS

Interface	Voltage	Current (max)	Total Power
RS 232C	+5 ± 0.25V	595 mA	3.8 watts
	−12 ± 0.6V	30 mA	
	+12 ± 0.6V	30 mA	
RS 449/422A	+5 ± 0.25V	775 mA	4.1 watts

Environmental Characteristics

Temperature — 0 - 55°C, free moving air across base board and MULTIMODULE board

Humidity — to 90%, without condensation

Physical Characteristics

Width — 7.27 cm (2.85 inches)

Length — 9.40 cm (3.70 inches)

Height — 1.40 cm (0.56 inches)

Weight — 72 gm (2.53 ounces)

Reference Manual (Not Supplied)

143983 — iSBX 352 Bit Serial Communications MULTIMODULE Board Hardware Reference Manual.

Reference manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Ave., Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
SBX 352	HDLC/SDLC Serial I/O MULTIMODULE Board

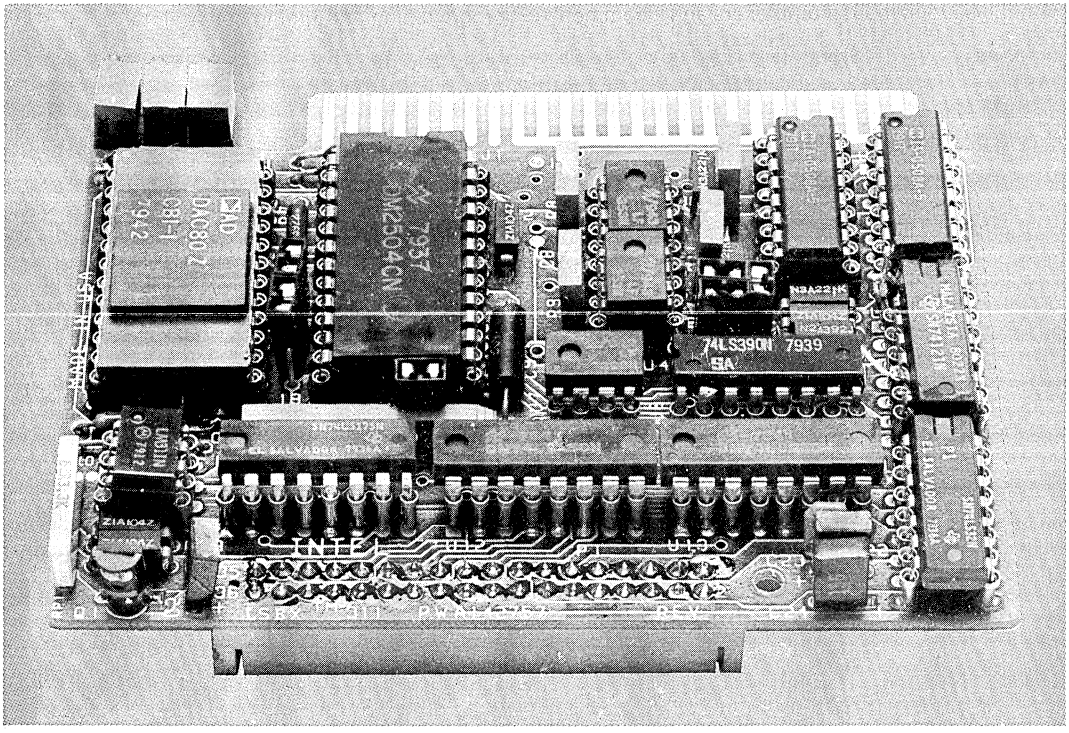
Analog I/O Expansion and Signal Conditioning Boards

10

iSBX 311
ANALOG INPUT MULTIMODULE BOARD

- **Low cost analog input for iSBX MULTIMODULE compatible iSBC boards**
- **8 differential/16 single-ended, fault protected inputs**
- **20 mV to 5V full scale input range, resistor gain selectable**
- **Unipolar (0 to +5V) or bipolar (−5V to +5V) input, jumper selectable**
- **12-bit resolution analog-to-digital converter**
- **0.035% full scale accuracy (11 bits) at 25°C**
- **18 kHz samples per second throughput to memory**
- **Connector compatible with iCS 910 Analog Termination Panel**

The Intel iSBX 311 Analog Input MULTIMODULE board provides simple interfacing of non-isolated analog signals to any iSBC board which has an iSBX compatible bus and connectors. The single-wide iSBX 311 plugs directly onto the iSBC board, providing data acquisition of analog signals from eight differential or sixteen single-ended voltage inputs, jumper selectable. The iSBX 311 MULTIMODULE is connector and pinout compatible with the Intel iCS 910 Analog Signal Conditioning/Termination panel so that field wiring can easily be terminated and current loop-to-voltage conversion resistors can be mounted for current loop analog signal monitoring. Resistor gain selection is provided for both low level (20mv full scale range) and high level (5 volt FSR) signals. Incorporating the latest high quality IC components, the iSBX 311 MULTIMODULE board provides 12 bit resolution, 11 bit accuracy, and a simple programming interface, all on a low cost iSBX MULTIMODULE board.



FUNCTIONAL DESCRIPTION

The iSBX 311 Analog Input MULTIMODULE board is a member of Intel's growing family of MULTIMODULE expansion boards, designed to allow quick, easy, and inexpensive expansion for the Intel single board computer product line. The iSBX 311 Analog Input MULTIMODULE Board shown in figure 1, is designed to plug onto any host iSBC microcomputer that contains an iSBX bus connector (P1). The board provides 8 differential or 16 single-ended analog input channels that may be jumper-selected as the application requires. The MULTIMODULE board includes a user-configurable gain, and a user-selectable voltage input range (0 to +5 volts, or −5 to +5 volts). The MULTIMODULE board receives all power and control signals through the iSBX bus connector to initiate channel selection, sample and hold operation, and analog-to-digital conversion.

Input Capacity

Sixteen separate analog signals may be randomly or sequentially sampled in single-ended mode with the sixteen input multiplexers and a common ground. For noisier environments, differential input mode can be configured to achieve 8 separate differential signal inputs, or 16 pseudo-differential inputs.

Resolution

The iSBX 311 MULTIMODULES provide 12-bit resolution with a successive approximation analog-to-digital converter. For bipolar operation (−5 to +5 volts) it provides 11 bits plus sign.

Speed

The A-to-D converter conversion speed is 35 microseconds (28KHZ samples per second). Combined with the sample and hold, settling times and the programming interface, maximum throughput via the iSBX bus and into memory will be 54 microseconds per sample, or 18 KHZ samples per second, for a single channel, a random channel, or a sequential channel scan. A-to-D conversion is initiated via the iSBX connector and programmed command from the iSBC base board. Interrupt on end-of-conversion is a standard feature to ease programming and timing constraints.

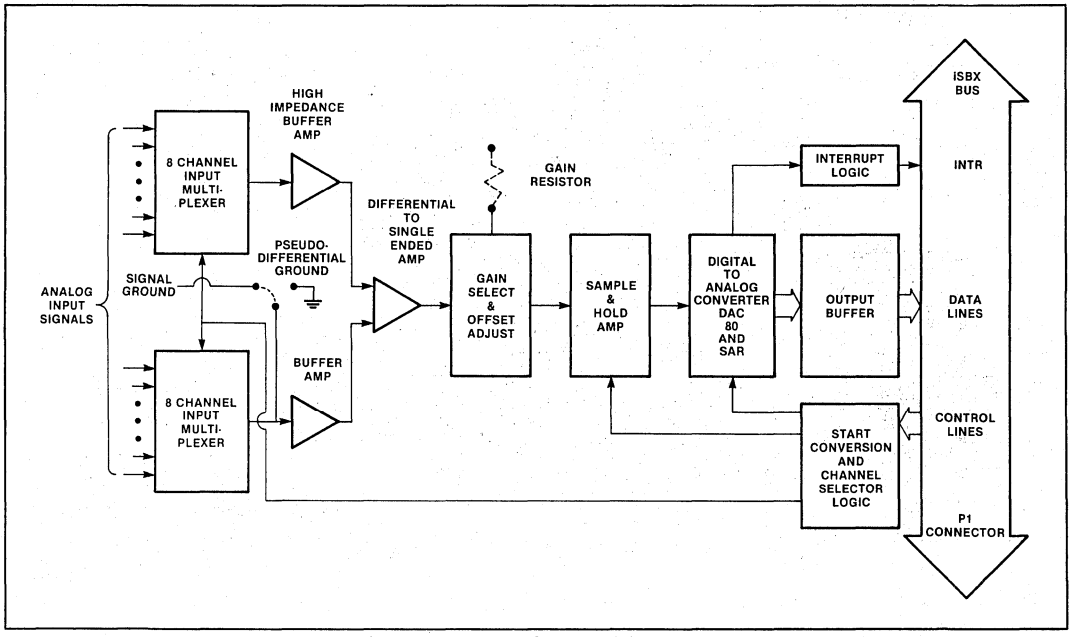


Figure 1. iSBX 311 Analog Input MULTIMODULE Board

Accuracy

High quality components are used to achieve 12 bits resolution and accuracy of .035% full scale range ± ½ LSB. Offset and gain are adjustable to ±0.024% FSR ± ½ LSB accuracy at any fixed temperature between 0°C (gain = 1). See specifications for other gain accuracies.

Gain

To allow sampling of millivolt level signals such as strain gauges and thermocouples, gain is made configurable via user inserted gain resistors up to 250× (20 millivolts, full scale input range). User can select any other gain range from 1 to 250 to match his application.

OPERATIONAL DESCRIPTION

The host iSBC microcomputer addresses the iSBX 311 MULTIMODULE board by executing IN or OUT instructions to the iSBX 311 MULTIMODULE as one of the legal port addresses. Analog-to-digital conversions can be programmed in either of two modes: 1. start conversion and poll for end-of-conversion (EOC), or 2. start conversion and wait for interrupt (INTRO/) at end of conversion. When conversion is complete as signaled by one of the above techniques, INput instructions read two bytes (low and high bytes) containing the 12 bit data word plus status information as shown below.

OUTput Command — Select input channel and start conversion.

Bit Position	7	6	5	4	3	2	1	0
Input Channel					C3	C2	C1	C0

INput Data — Read converted data and status (low byte) or Read converted data (high byte). Reads can be with or without reset of interrupt request line (INTRO/).

Bit Position	7	6	5	4	3	2	1	0
Low/status Byte	D3	D2	D1	D0		start/	busy/	EOC/
High Byte	D11	D10	D9	D8	D7	D6	D5	D4

Fastest data conversion and transfer to memory can be obtained by dedicating the microcomputer to setting the channel address/starting conversion, polling the status byte for EOC/, and when it comes true, read the two bytes of the conversion and send the start conversion/next channel address command. For multitasking situations it may be more convenient to use the interrupt mode, reading in data only after an interrupt signals end of conversion.

SPECIFICATIONS

Inputs — 8 differential. 16 single-ended. Jumper selectable.

Full Scale Input Voltage Range — −5 to +5 volts (bipolar). 0 to +5 volts (unipolar). Jumper selectable.

Gain — User-configurable through installation of two resistors. Factory-configured for gain of X1; gains above 250 not recommended.

Resolution — 12 bits over full scale range (1.22 mv at 0-5 v, 5 μv at 0-20 mv)

Accuracy —

Gain	Accuracy at 25°C
1	±0.035% ± ½ LSB
5	±0.035% ± ½ LSB
50	±0.035% ± ½ LSB
250	±0.035% ± ½ LSB

NOTE:
Figures are in percent of full scale reading. At any fixed temperature between 0° and 60°C, the accuracy is adjustable to ±0.035% of full scale.

Dynamic Error — ±0.015% FSR for transitions

Gain TC (at Gain = 1): 30 PPM per degree centigrade (typical); 56 PPM per degree centigrade (max).

Offset TC (in percent of FSR/°C):

Gain	Offset
1	.0018
5	.0036
50	.024
250	.116

Offset is measured with user-supplied 10 PPM/°C gain resistors installed.

Input Protection — ±30 volts.

Input Impedance — 20 megohms (minimum).

Conversion Speed — 50 microseconds (nominal).

Common Mode Rejection Ratio — 60 db (minimum).

Sample and hold — sample time 15 microseconds.

Aperature — hold aperature time: 120 nanoseconds.

Connectors —

Interface	Pins (Qty)	Centers in	Centers cm	Mating Connectors
P1 iSBX Bus	36	0.1	0.254	iSBC iSBX connector
J1 8/16 channels analog	50	0.1	0.254	3m 3415-000 or T1 H312125 or iCS 910 cable

Physical Characteristics

Width — 9.40 cm (3.7 inches)

Length — 6.35 cm (2.5 inches)

Height — 2.03 cm (0.80 inch) MULTIMODULE board only

2.82 cm (1.13 inches) MULTIMODULE and iSBC board

Weight — 68.05 gm (2.4 ounces)

Electrical Characteristics (from iSBX connector)

Vcc = ±5 volts (±0.25V), Icc = 250 mAmax

Vdd = +12 volts (±0.6V), Idd = 50 mAmax

Vss = −12 volts (±0.6V), Iss = 55 mAmax

Environmental Characteristics

Operating Temperature — 0° to 60°C (32° to 140°C)

Relative Humidity — to 90% (without condensation)

Shock Tested At — Class B Specification

Reference Manuals

142913-001 — iSBX 311 Analog Input MULTIMODULE Board Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

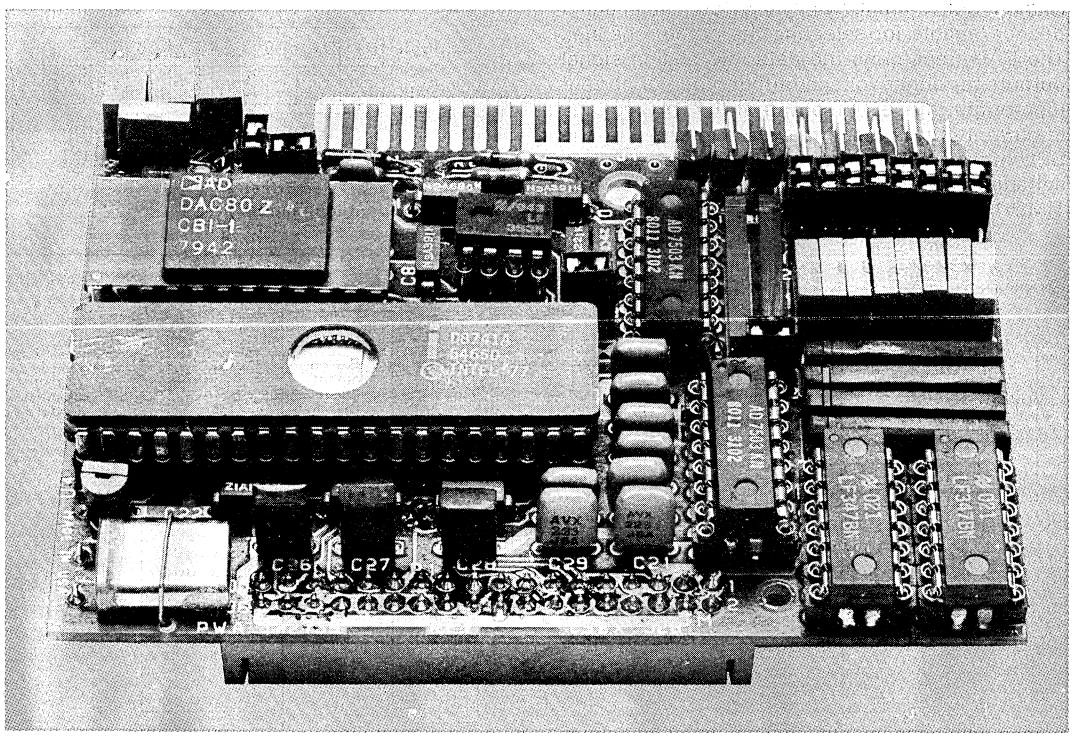
ORDERING INFORMATION

Part Number	Description
SBX 311	Analog Input MULTIMODULE Board

iSBX 328
ANALOG OUTPUT MULTIMODULE EXPANSION BOARD

- **Low cost analog output for iSBX MULTIMODULE compatible iSBC Boards**
- **8 channels output, current loop or voltage in any mix**
- **4–20 mA current loop; 5V unipolar or bipolar voltage output**
- **12-bit resolution**
- **0.035% full scale volage accuracy @ 25°C**
- **Connector compatible with iCS 910 Analog Termination Panel**
- **Intel design based on UPI control for high density and low cost**
- **Programmable offset adjust in current loop mode**

The Intel iSBX 328 MULTIMODULE board provides analog signal output for any iSBC board which has an iSBX compatible bus and connectors. The single-wide iSBX 328 plugs directly onto the iSBC board, providing eight independent output channels of analog voltage for meters, CRT control, programmable power supplies, etc. Voltage output can be mixed with current loop output for control of popular 4-20ma industrial control elements. By using an Intel single chip computer LSI (8041) for refreshing separate sample-hold amplifiers through a single 12 bit DAC, eight channels can be contained on a single MULTIMODULE board, for high density and low cost per channel. High quality analog components provide 12 bit resolution, 11 bit accuracy, and slew rates per channel of 0.1 volt per microsecond. Programming the iSBX 328 MULTIMODULE board is done via a simple two byte protocol over the iSBX bus. Maximum channel update rates are 5KHZ on a single channel to 1 KHZ on all eight channels. Outputs are compatable for screw termination of field wiring on the iCS 910 Analog Signal Conditioning/Termination Panel.



FUNCTIONAL DESCRIPTION

The iSBX 328 MULTIMODULE board, shown in figure 1 is designed to plug onto any host iSBC microcomputer that contains an iSBX bus connector. The board uses an 8041 UPI device to control eight analog output channels that may be user-configured through jumpers to operate in either bipolar voltage output mode (−5 to +5 volts), unipolar voltage output mode (0 to +5 volts), or current loop output mode (4 to 20 mA) applications. Channels may be individually wired for simultaneous operation in both current loop output and voltage output applications. The outputs from 50-pin edge connector J1 on the MULTIMODULE board are pin-compatible with the iCS 910 Signal Conditioning/Termination Panel.

Interfacing Through the Intel iSBX Bus

All data to be output through the MULTIMODULE board is transferred from the host iSBC microcomputer to the MULTIMODULE board via the iSBX bus connector. The UPI device on the MULTIMODULE board accepts the binary digital data and generates a 12-bit data word for the Digital-to-Analog Converter (DAC) and a four bit channel decode/enable for selecting the output channel. The DAC transforms the data into analog signal outputs for either voltage output mode or current loop output mode. Offsetting of the DAC voltage in current output mode may be performed by the UPI software offset routine or by the hardware offset adjustments included on the board. The MULTIMODULE board status is available via the iSBX bus connector, to determine if the UPI is ready to receive updates to analog output channels.

OPERATIONAL DESCRIPTION

The host iSBC microcomputer addresses the MULTIMODULE board by executing IN or OUT instructions specifying the iSBX 328 MULTIMODULE as a port address. The UPI on the iSBX 328 is initialized to select whether software or hardware offset is to be used and how many channels will be active. Then a 2 byte transfer to each active channel sets the 12 bit output value, the channel selected and the current or voltage mode.

Commands

OUTput Command — Initialization of UPI/iSBX 328

7							0 Bit
			N	N	D2	D1	D0

last channel to be output

NN: 0,0 = unipolar configuration software current offset
0,1 = no mixing
1,0 = bipolar configuration software current offset

OUTput Command — Data Bytes

	7							0 Bit
High Byte	D11	D10	D9	D8	D7	D6	D5	D4
Low Byte	D3	D2	D1	D0	A3	A2	A1	V/C

DAC Data — DAC channel to receive data

0 = UPI generates offset
1 = SBC generates offset in current loop mode

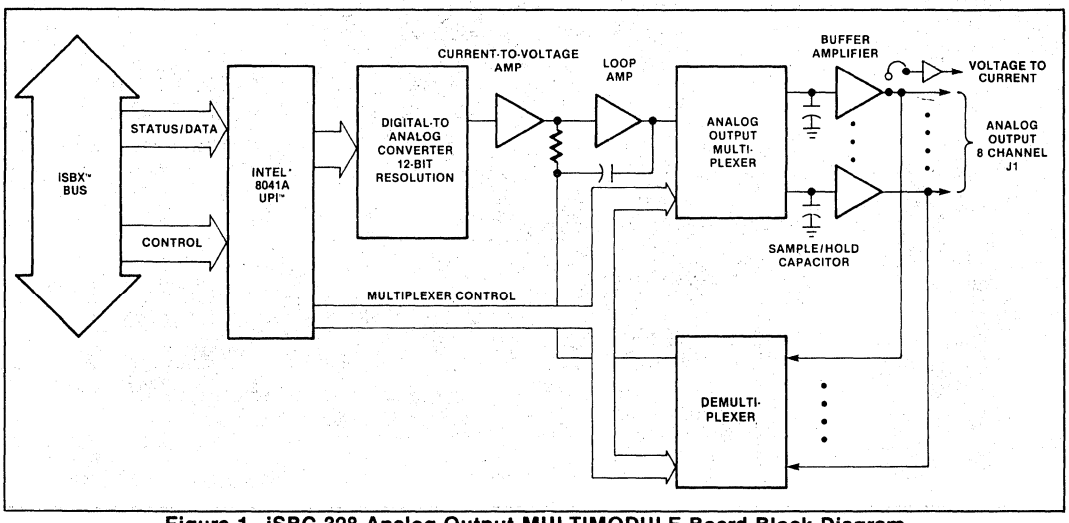
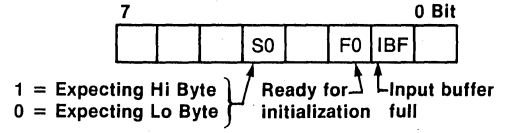


Figure 1. iSBC 328 Analog Output MULTIMODULE Board Block Diagram

INput Command — Status Buffer Read



Interrupts

No interrupts are issued from the iSBX 328 to the host iSBC microcomputer. Data coordination is handled via iSBC software polls of the status buffer.

SPECIFICATIONS

Outputs — 8 non-isolated channels, each independently jumpered for voltage output or current loop output mode.

Voltage Ranges — 0 to +5 volts (unipolar operation)
−5 to +5 volts (bipolar operation)

Current Loop Range — 4 to 20 mA (unipolar operation only)

Output Current — ±5 mA maximum (voltage mode-bipolar operation)

Load Resistance — 0 to 250 ohms with on-board iSBX power. 1000 ohms minimum with 30 VDC max. external supply

Compliance Voltage — 12 V using on-board iSBX power. If supplied by user, up to 30 VDC max

Resolution — 12 bits bipolar or unipolar

Slew Rate — 0.1 volt per microsecond minimum

Single Channel Update Rate — 5KHz

Eight Channel Update Rate — 1KHz

Accuracy —

Mode	Accuracy	Ambient Temp
Voltage-Unipolar, typical	±0.025% FSR	@ 25°C
Voltage-Unipolar, maximum	±0.035% FSR	@ 25°C
Voltage-Unipolar, typical	±0.08% FSR	@ 0° to 60°C
Voltage-Unipolar, maximum	±0.19% FSR	@ 0° to 60°C
Voltage-Bipolar, typical	±0.025% FSR	@ 25°C
Voltage-Bipolar, maximum	±0.035% FSR	@ 25°C
Voltage-Bipolar, typical	±0.09% FSR	@ 0° to 60°C
Voltage-Bipolar, maximum	±0.17% FSR	@ 0° to 60°C
Current Loop, typical	±0.07% FSR	@ 25°C
Current Loop, maximum	±0.08% FSR	@ 25°C
Current Loop, typical	±0.17% FSR	@ 0° to 60°C
Current Loop, maximum	±0.37% FSR	@ 0° to 60°C

Output Impedance — 0.1 ohm. Drives capacitive loads up to 0.05 microfarads. (approx. 1000 foot cable)

Temperature Coefficient — 0.005%/°C

Connectors —

Interface	Pins (Qty)	Centers in	Centers cm	Mating Connectors
P1 iSBX Bus	36	0.1	0.254	iSBC iSBX connector
J1 8/16 channels analog	50	0.1	0.254	3m 3415-000 or T1 H312125 or iCS 910 cable

Physical Characteristics

Width — 9.40 cm (3.7 inches)

Length — 6.35 cm (2.5 inches)

Height — 1.4 cm (0.56 inch) MULTIMODULE board only
2.82 cm (1.13 inches) MULTIMODULE and iSBC board.

Weight — 85.06 gm (3.0 ounces)

Electrical Characteristics

Vcc = ±5 volts (±0.25V), Icc = 140 ma max

Vdd = ±12 volts (±0.6V), Idd = 45 ma max (voltage mode)
= 200 ma max (current loop mode)

Vss = −12 volts (±0.6V), Iss = 55 ma max

Environmental Characteristics

Operating Temperature — 0° to 60°C (32° to 140°C)

Relative Humidity — to 90% (without condensation)

Shock Tested At — Class B specifications

Reference Manuals

142914-001 — iSBX 328 Analog Output MULTIMODULE Board Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051

ORDERING INFORMATION

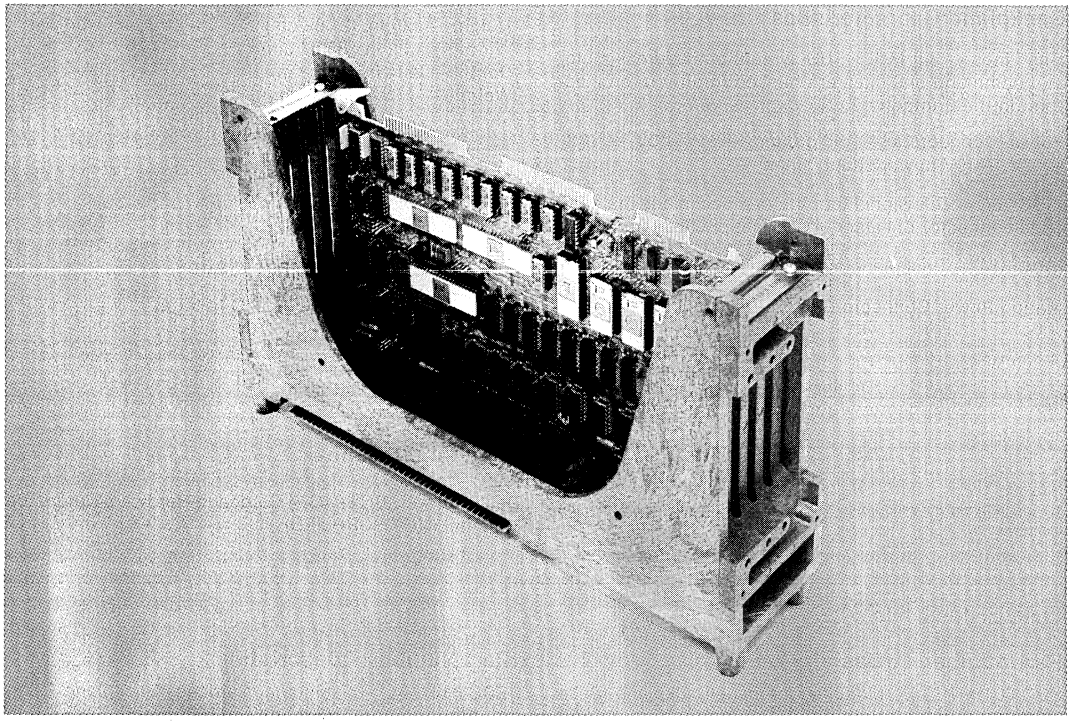
Part Number	Description
SBX 328	Analog Output MULTIMODULE Board

11 System Packaging and Power Supplies and Service

iSBC 604/614 or (pSBC 604/614*) MODULAR CARDCAGE/BACKPLANE

- **Interconnection on MULTIBUS system bus and housing for up to four Intel iSBC boards**
- **Strong cardcage structure helps protect installed iSBC single board computers and expansion boards against warping and physical damage**
- **Connectors allow interconnection of two or more cardcage/backplane assemblies**
- **Cardcage mounting holes facilitate interconnection of units**
- **Compatible with 3.5-inch RETMA rack mount increments**
- **Dual backplane power supply connectors and signal line termination circuits on iSBC 604 Cardcage/ Backplane**

The iSBC 604 and iSBC 614 Modular Cardcage/Backplane units provide low-cost, off-the-shelf housing for OEM products using two or more Intel single board computers. Each unit interconnects and houses up to four boards. The base unit, the iSBC 604 Cardcage/Backplane, contains a male backplane PC edge connector and bus signal termination circuits, plus power supply connectors. It is suitable for applications requiring a single unit, or may be interconnected with the iSBC 614 Cardcage/Backplane when more then one cardcage/backplane unit is needed. The iSBC 614 Cardcage/Backplane contains both male and female backplane connectors, and may be interconnected with iSBC 604/614 Cardcage/Backplane units. Both units are identical, with the exception of the power connectors and bus signal terminator features. A single unit may be packaged in a 3.5-inch RETMA rack enclosure, and two interconnected units may be packaged in a 7-inch enclosure. The units are mountable in any of three planes.



*Same product, manufactured by Intel Puerto Rico, Inc.

AFN-00282A

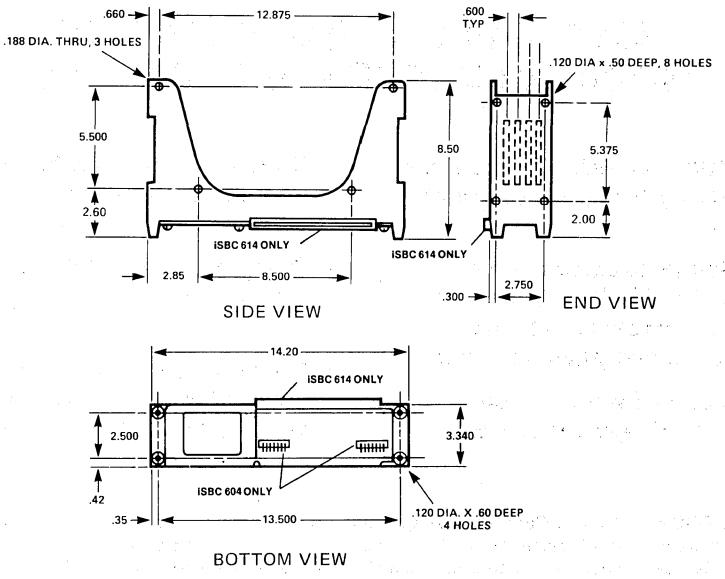


Figure 1. iSBC 604/614 Modular Backplane and Cardcage Dimensions

SPECIFICATIONS

Backplane Characteristics

Bus Lines — All MULTIBUS system bus address, data, and command bus lines are bussed to all four connectors on the printed circuit backplane

Power Connectors — for ground, +5V, −5V, +12V, −12V, −10V power supply lines

iSBC 604 — Bus signal terminators, backplane male PC edge connector only, and power supply headers

iSBC 614 — Backplane male and female connectors

Mating Power Connectors

AMP	Connector	87159-7
	Pin	87023-1
	Polarizing key	87116-2
Molex	Connector	09-50-7071
	Pin	08-50-0106
	Polarizing key	15-04-0219

Note
1. Pins from a given vendor may only be used with connectors from the same vendor.

Physical Dimensions

Height — 8.5 in. (21.59 cm)
Width — 14.2 in. (36.07 cm)
Depth — 3.34 in. (8.48 cm)
Weight — 35 oz (992.23 gm)

Environmental Characteristics

Operating Temperature — 0°C to 55°C

Reference Manual

9800708 — iSBC 604/614 Cardcage Hardware Reference Manual (NOT SUPPLIED)

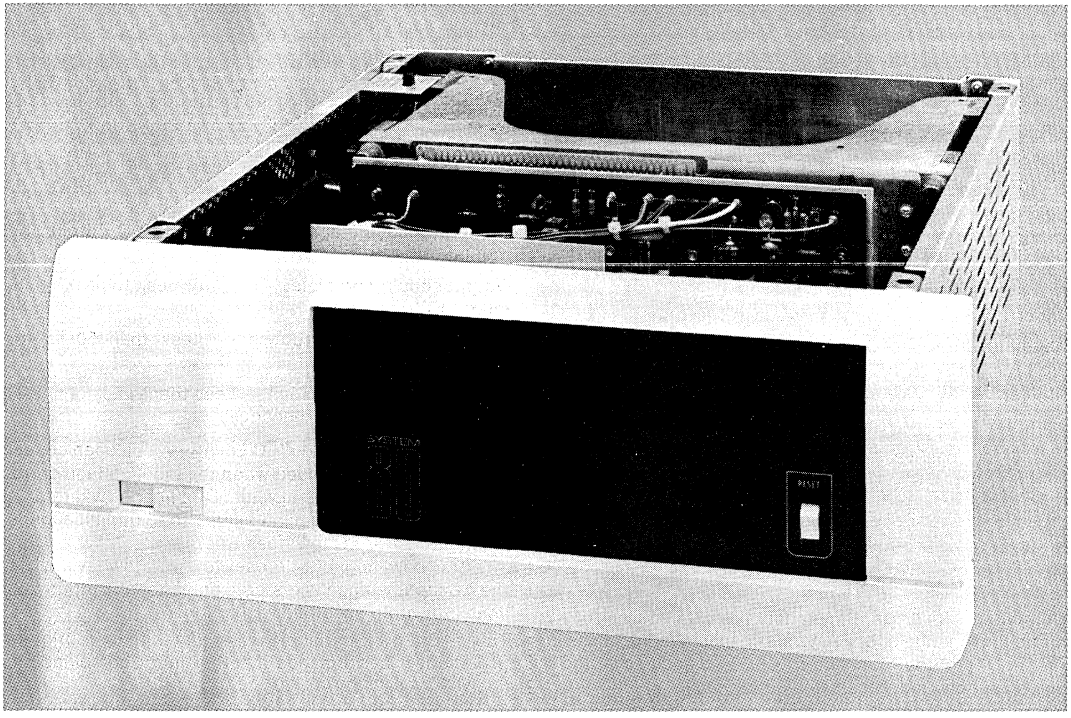
ORDERING INFORMATION

Part number	Description	Part Number	Description
SBC 604	Modular Cardcage/Backplane (Base Unit)	SBC 614	Modular Cardcage/Backplane (Expansion Unit)

iSBC 660
SYSTEM CHASSIS

- **Eight-slot cardcage and backplane for iSBC computers and expansion boards**
- **Heavy duty power supply with all standard iSBC voltages**
- **Compatible with all Intel single board computers**
- **Forced-air cooling**
- **Attractive, versatile pop-off front panel**
- **19-inch wide rack mountable chassis**
- **Horizontal board mounting for compactness**
- **110/220V, 50/60 Hz operation**

The iSBC 660 System Chassis is an attractive, 7-inch high system chassis designed for use with Intel OEM computers. It has eight slots for single board computers, memory, I/O, or other expansion modules. The iSBC 660 is ideal for applications requiring multiple board solutions. DC power output is provided at +12V, +5V, −12V, and −5V levels. The current capabilities of each of these output levels have been chosen to provide power over a 0°C to 50°C temperature range for the majority of applications requiring combinations of computers, memories, peripherals, and other I/O capabilities. Current limiting and over-voltage protection is provided at all outputs. Standard logic recognizes a system AC power failure and generates a TTL signal for use in power-down control. For user convenience, a reset switch is provided on the front panel. The reset signal generated and sent to the system bus can be used for external system control.



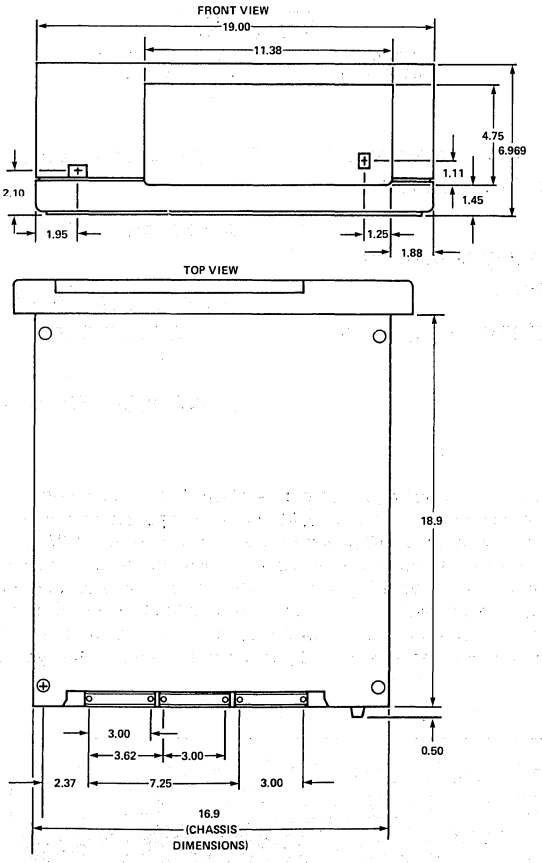


Figure 1. iSBC System Chassis Dimensions

SPECIFICATIONS

Electrical Characteristics

Input Power

Frequency: 50 Hz ±5%, 60 Hz ±5%

Voltage: 115V ±10%, 230V ±10%, 215 VAC ±10%, 100 VAC ±10% via user configured wiring options

Output Power

Power	Output Current (Max)	Current Limit (Amps)	Over-Voltage Protection
+12V	4.5A	5.4	15V ±1V
+5V	30A	3.6	6.2V ±0.4V
−5V	1.75A	2.1	-6.2V ±0.4V
−12V	1.75A	2.1	-15V ±1V

Combined Line/Load Regulation — ±1% at ±10% static line change and ±50% static load change, measured at the output connector (±0.2% measured at the power supply under the same conditions).

Remote Sensing — Provided for +5 VDC output line regulation.

Output Ripple and Noise — 10 mV peak-to-peak maximum (DC to 500 kHz).

Output Transient Response — Less than 50 μs for ±50% load change.

Output Transient Deviation — Less than ±5% of linitial voltage for ±50% load change.

Power Failure Indication (AC Low) — A TTL open collector high signal is provided when the input voltage drops below 90% of its nominal value. DC voltages will remain within 5% of their nominal values for 3.0 milliseconds (minimum) after AC low goes true.

The "AC Low" signal will reset to a TTL low level when the AC input voltage is restored and after all output voltages are within specified regulation.

The "AC Low" threshold is adjustable for optimum power-down performance at other input combinations (i.e. 100 VAC, 215 VAC, 50 Hz).

Humidity — Up to 90% relative, non-condensing

Physical Characteristics

Height — 7 in. (17.8 cm)

Width

At Front Panel: 19 in. (48.3 cm)

Behind Front Panel: 17 in. (43.2 cm)

Depth — 20 in. (50.8 cm) with all protrusions

Environmental Characteristics

Temperature

Operating: 0°C to 50°C

Non-Operating: −40°C to +85°C

Equipment Supplied

iSBC 660 System Chassis with iSBC 640 Power Supply, iSBC 604/614 Cardcage/Backplane, dual fans, pop-off front panel

Connector pack with RS232C cable (terminal/modem interface to single board computers), two 50-pin parallel I/O connectors for single board computers

Schematics for cardcage/backplane, chassis

Outline drawing

Reference Manuals

9800505A — iSBC 660 Hardware Reference Manual (NOT SUPPLIED)

9800505 — iSBC 660 System Chassis Hardware Reference Manual (NOT SUPPLIED)

9800803 — iSBC 640 Power Supply Hardware Reference Manual (NOT SUPPLIED)

9800708 — iSBC 604/614 Cardcage Hardware Reference Manual (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

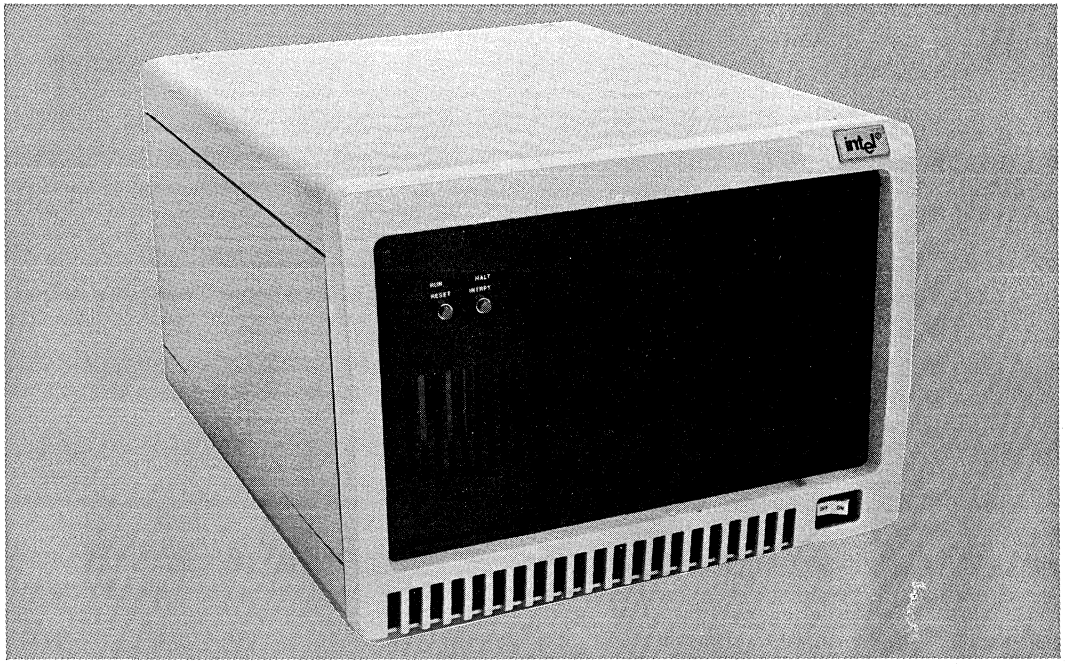
ORDERING INFORMATION

Part Number	Description
SBC 660	System Chassis

iSBC™ 680/iSBC 681
MULTISTORE™ USER SYSTEM PACKAGE

- **Complete system package for user-selected Intel® iSBC boards and up to two 8″ peripheral drives**
- **Holds up to six iSBC boards compatible with the MULTIBUS® system bus**
- **Supplies ±5, ±12, ±24 VDC to power boards and peripheral drives**
- **Available in table-top (iSBC 680 package) or rack-mount (iSBC 681 package) configurations**
- **Designed to meet UL safety requirements and FCC/VDE EMI limits**
- **Power supply provides 8 ms of power-fail warning, plus a real-time clock**

The iSBC 680/iSBC 681 Multistore User System Package products make available to the OEM a new way to assemble his systems for those applications requiring rotating memory or other peripherals built in the 8″ industry-standard form factor. The Multistore package allows the OEM to select the iSBC boards required for the job, and to independently choose from a wide variety of peripherals to complete the system. The switching power supply provides sufficient current at all voltage levels to power most manufacturers' drives, as well as furnishing the standard MULTIBUS system bus voltages to the iSBC boards in the package's cardcage. The appearance of the packages provides an attractive addition to the OEM's system, while the construction allows easy access to the interior for service.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, MULTIBUS, iRMX, iSBC, iSBX, MULTIMODULE, Multistore and iCS. 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 CORPORATION, 1981

July, 1981
143884

FUNCTIONAL DESCRIPTION

Physical Packaging

PACKAGE CONSTRUCTION — The Multistore package is constructed entirely of metal, and all cover pieces are gasketed to completely contain high-frequency noise from the power supply and the system boards within the package.

MOUNTING OPTIONS — The iSBC 680 package is a table-top structure; the iSBC 681 package is the rack-mount version with slides attached to the side panels and a wider trim bezel to cover the mounting rails.

COOLING — The boards and peripherals installed in the package are cooled by air brought in at the bottom of the front panel and drawn through the power supply, with the heated air discharged to the rear of the package.

CARDCAGE/BACKPLANE — The cardcage/backplane accepts up to six iSBC boards compatible with the Intel MULTIBUS system bus (Figure 1).

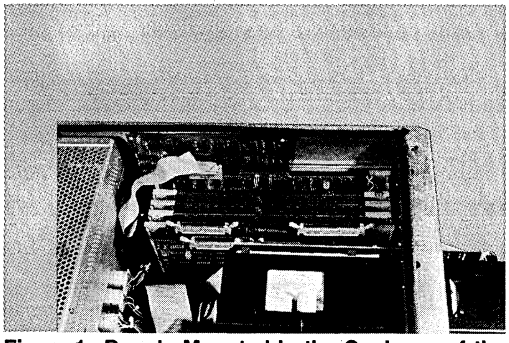


Figure 1. Boards Mounted in the Cardcage of the iSBC 680 Package

PARALLEL PRIORITY — Up to six bus masters may be installed in the package because of the parallel priority logic which is an integral part of the backplane.

ENHANCED NOISE IMMUNITY — The integrity of the package is enhanced by a new backplane for the system boards, which offers improved noise immunity through advanced design techniques.

PERIPHERAL MOUNTING — Two positions are provided for mounting of peripheral drives conforming to the de facto industry standard for size and mounting points on 8″ peripherals. The mounting system provides slides for the bases of the drives, allowing the drives to be installed/removed from the front of the package (Figure 2).

Blank covers with ventilation slots are provided with the package for the unused peripheral positions and for those peripherals not furnished with a cosmetic cover.

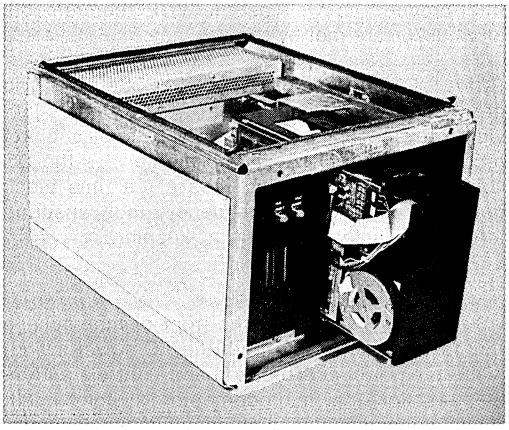


Figure 2. iSBC 680 Package with Winchester Drive Pulled Out for Servicing

System Power

BASIC POWER SUPPLY — The supply provided with the package is an advanced-technology switching power supply, offering large current capabilities over the six DC voltages supported. Sockets for drive power are located on the power supply bulkhead at the rear of the peripheral cavity (Figure 3).

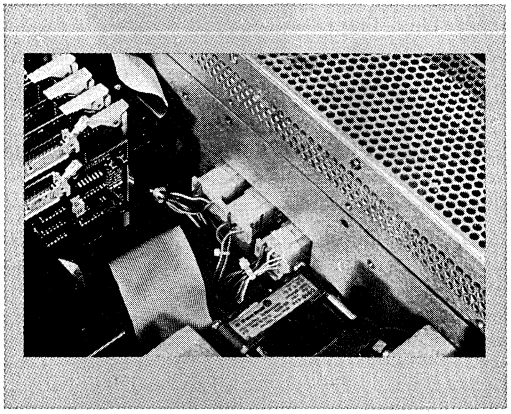


Figure 3. Power Supply Showing Three Power Connectors (Two for Peripherals, One for the Cardcage/Backplane)

INTERNATIONAL ACCEPTANCE — The package is a UL-recognized component, and it has been designed to meet the safety requirements of CSA and VDE.

MEETS EMI STANDARDS — The FCC standards for conducted and radiated EMI (electromagnetic interference), as well as the VDE requirements (0871/0875), are met by the package.

POWER-FAIL/AUTO-RESTART SYSTEM — The package gives the user a set of logic signals providing advanced warning of power failures and protection for battery backed-up memory as the DC voltages fall. It also furnishes a real-time clock derived from the line frequency, thus ensuring long-term stability in user time-keeping.

User Interface

USER CONTROLS — At the front of the package the user has access to both the AC power switch (with integral circuit breaker) and controls and indicators for the microcomputer system itself. "RESET" and "INTERRUPT" switches are provided, along with "RUN" and "HALT" LED indicators.

DEVICE INTERFACE — With the package designed for maximum suppression of both EMI and ESD (electrostatic discharge) the preferred interface between the installed boards and external devices is with shielded cables isolated through user-supplied connectors installed in the panel provided at the rear of the package (Figure 4). The six cut-outs, sized for 50-pin connectors, are furnished with individual cover plates.

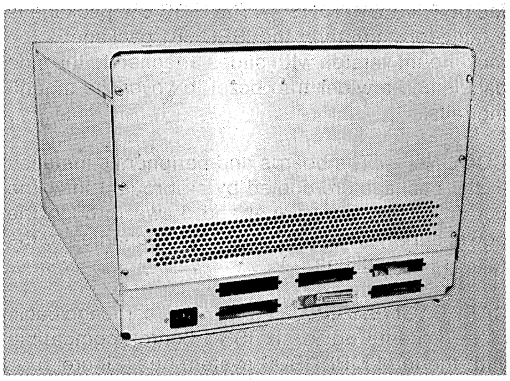


Figure 4. iSBC 680 Package Showing External Device Connection Panel

SPECIFICATIONS

Input Power

Frequency — 47-66 Hz

Voltage — 90-126 VAC/180-252 VAC, single phase (user-selectable).

Periodic and Random Deviation (PARD) — 50 mV peak-to-peak, all outputs.

Output Regulation (Combined Line and Load) — ±1% under any conditions of AC input voltage variation (within operational range) and output load change.

Line Transient Tolerance — A signal of up to 1000 VDC, with a pulse width of up to 50 μs, will have no affect on system operation.

Output Power

Power Fail Indication — PFIN/ is generated approximately 8 ms after the input drops below 90/180 VAC. PFIN/ is available on the P2 connector of the cardcage/backplane for generation of an interrupt. The ± 24 VDC outputs will go to zero within 1 ms of issuance of PFIN/; the ±5 and ±12 VDC outputs will remain within specification for at least 8 ms, after which MPRO/ will go true to protect non-volatile memories from being written into as DC power fails.

System Clock — The power supply provides a 2× line frequency clock output, available on the P2 connector of the cardcage/backplane.

Nominal Voltage	Current[1] (Max Amps)	Typical Peripheral Power Requirements[2]	
		8" Winchester	Diskette Drive
+5	30.0	5.1	1.0
−5	2.0	0.25	0.07
+12	2.9	—	—
−12	3.0	0.7	—
+24	7.8	4.0	1.8
−24	1.6	—	—

NOTES:
1. The maximum power available from the supply, from all outputs, is 300 watts.
2. These are worst-case data, drawn from manufacturers' data sheets.

Physical Characteristics (Figure 5)

Width — 16.8/19.0 in. (42.5/48.3 cm)
iSBC 680/iSBC 681 Packages

Length — 21.5 in. (54.6 cm)

Height — 12.2 in. (31.1 cm)

Weight — 40 lb (18.2 kg) (approximate)

Board Slots — Six @0.665 in. on centers between boards. The board in the top slot may contain any iSBX and/or iSBC MULTIMODULE boards.

Peripheral Size — The drives must fit within an envelope 8.55 in. high by 14.25 in. deep by 4.65 in. wide.

Data Separator Board Location — A space is provided within the package to secure a Winchester drive data separator board, if required.

Environmental Characteristics

Ambient (Inlet) Air Temperature — The inlet air temperature, with peripheral drives installed, may not exceed 35°C. This is for the protection of the peripherals, as both diskette and Winchester drives have ambient maximums of 40°C in most instances.

Humidity — 20% to 80% RH, non-condensing for the chassis and typical peripheral content.

NOTE: The photos of the Multistore packages in this data sheet show boards, an 8″ Winchester hard disk drive, and an 8″ flexible disk drive installed. The packages *do not* include these boards and peripherals; they are shown in the photographs to illustrate physical arrangements in the Multistore package.

Equipment Supplied

iSBC 680 Chassis — Includes table-top package with aluminum sheet metal, 6-slot cardcage/backplane, combination On/Off switch and circuit breaker, peripheral mounting hardware (user must supply power and signal cabling for peripherals), and power supply with AC power cord.

iSBC 681 Chassis — Same as iSBC 680 chassis, plus rack-mount chassis slides and wider bezel.

Reference Manual

162432 — iSBC™ 680/681 Multistore™ Chassis Hardware Reference Manual (NOT SUPPLIED)

Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051.

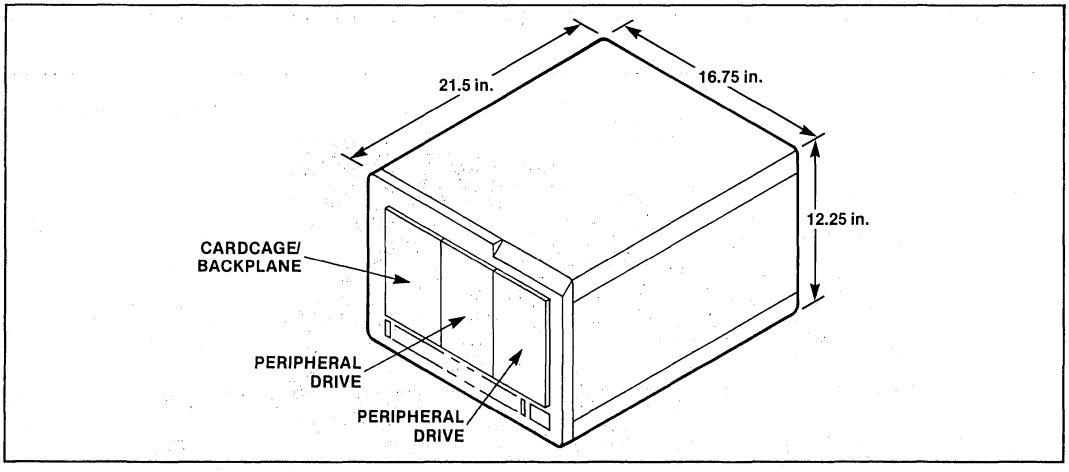


Figure 5. Physical Dimensions of the iSBC 680 Multistore User System Package

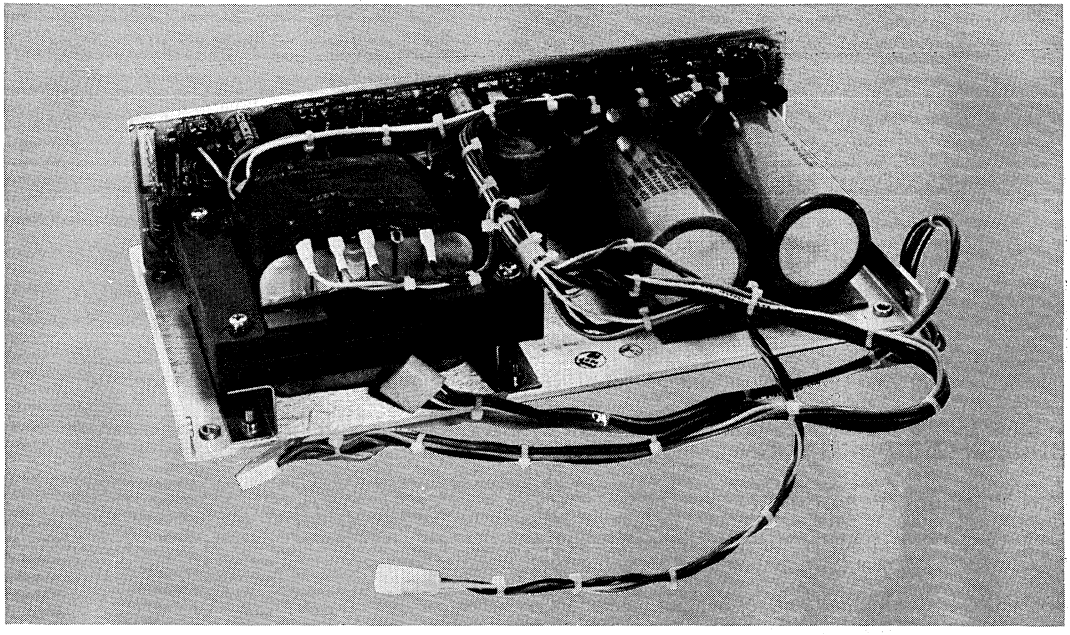
ORDERING INFORMATION

Part Number	Description
SBC 680	Multistore User System Package (Table-Top)
SBC 681	Multistore User System Package (Rack-Mount)

iSBC 635
POWER SUPPLY

- **Compact single chassis**
- **± 5V and ± 12V iSBC 80 and iSBC 86 system power**
- **Sufficient power for one fully loaded Intel single board computer plus residual power for up to three Intel iSBC expansion boards**
- **Current limiting and overvoltage protection on all outputs**
- **DC power cables and connectors mate directly to iSBC 604 Modular Cardcage/ Backplane assembly**
- **"AC low" power failure TTL logic level output provided for system power-down control**
- **100V, 115V, 215V, and 230V AC operation**
- **50 Hz or 60 Hz input**

The iSBC 635 Power Supply provides low cost, off-the-shelf, single chassis power generation for OEM products using Intel single board computers. The iSBC 635 supply provides regulated DC output power at + 12V, + 5V, − 12V, and − 12V levels. The current capabilities of each of these output levels have been chosen to provide power over a 0 °C to + 55 °C temperature range for one Intel single board computer fully loaded with I/O line terminators and drivers and EPROMs, plus residual capability for most combinations of up to three iSBC memory, I/O or combination expansion boards. Current limiting and overvoltage protection is provided on all outputs. Access for AC input is provided via a standard 4-pin keyed connector. DC output power levels are provided on cables with keyed connectors directly compatible with the iSBC 604 Modular Cardcage/Backplane assembly. The iSBC 635 supply includes logic whose purpose is to sense system AC power failure and generate a TTL signal for clean system power-down control.



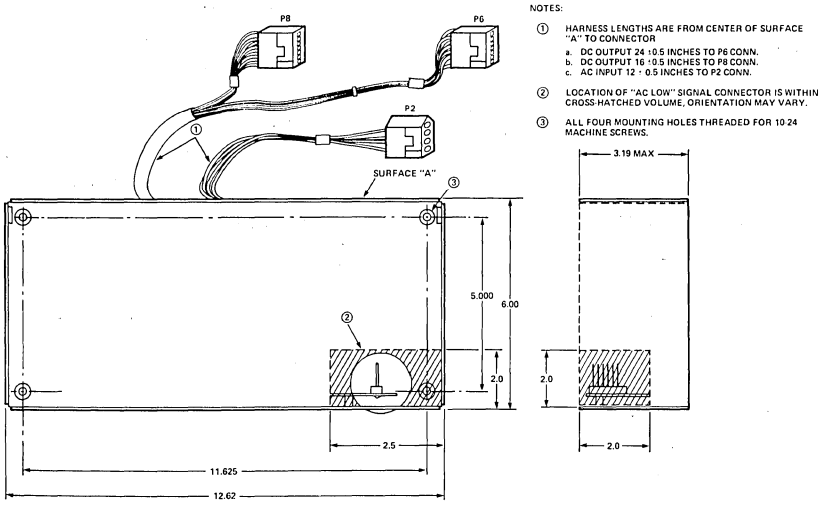


Figure 1. iSBC 635 Mounting Information

SPECIFICATIONS

Mating Connectors[1]

AC Input

Connector	Molex	03-09-1042 or equivalent
Pin	Molex	02-09-1118 or equivalent (18 to 22 gauge wire)

DC Output[2]

Header	Molex	09-66-1071
	AMP	87194-6

"AC Low" Control

Connector	Molex	09-50-7071
	AMP	87159-7
Polarizing key	Molex	15-04-0219
	AMP	87116-2
Pin	Molex	08-50-0106 (18 to 22 gauge wire)
	AMP	87023-1 (18 to 22 gauge wire)

Notes

1. Pins from a given vendor may only be used with connectors from the same vendor.

2. iSBC 635 DC output connectors are directly compatible with power input power connectors on iSBC 604 Modular Cardcage/Backplane assembly. Two connectors are provided.

Physical Characteristics

Height — 3.19 in. max (8.11 cm)
Width — 6.03 in. max (15.32 cm)
Depth — 12.65 in. max (32.12 cm)
Weight — 13 lb (5.90 kgm)

Electrical Characteristics

Input Power — Frequency: 47 - 63 Hz. Voltage (Nominal) (Single Phase): 100, 115, 215, or 230 VAC +10%

Output Power:

Nominal Voltage	Current (AMPS)(MAX)	Current Limit Range (AMPS)	Max Short Circuit (AMPS)	Over-Voltage Protection
+12	2.0	2.1-3.0	1.0 (Foldback)	+14 to +16 V
+ 5	14.0	14.7-21.0	7.0 (Foldback)	+5.8 to +6.6 V
- 5	0.9	0.9-1.4	1.4	-5.8 to -6.6 V
-12	0.8	0.8-1.2	1.2	-14 to -16V

Combined Line/Load Regulation — ±1% at ±10% static line change and ±50% static load change, measured at the output connector (±0.2% measured at the power supply under the same conditions).

Remote Sensing — Provided for +5VDC output line regulation.

Output Ripple and Noise — 10 mV peak-to-peak maximum (DC to 500 KHz)

Output Transient Response — Less than 50 μsec for ±50% load change

Output Transient Deviation — Less than ±5% of initial voltage for ±50% load change.

Power Failure Indication (AC Low) — A TTL open collector high signal is provided when the input voltage drops below 90% of its nominal value. DC voltages will remain within 5% of their nominal values for 3.0 milliseconds (minimum) after AC low goes true.

The "AC Low" signal will reset to a TTL low level when the AC input voltage is restored and after all output voltages are within specified regulation.

The "AC Low" threshold is adjustable for optimum powerdown performance at other input combinations (i.e. 100 VAC, 215 VAC, 50 Hz).

Environmental Characteristics

Operating Temperature — 0°C to +55°C with 35 CFM moving air

Non-Operating — −40°C to +85°C

Equipment Supplied

iSBC 635 Power Supply with AC and DC cables and connectors attached as shown in Figure 1.

Reference Manual

9800298C — iSBC 635 Power Supply Hardware Reference Manual (includes schematics) (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

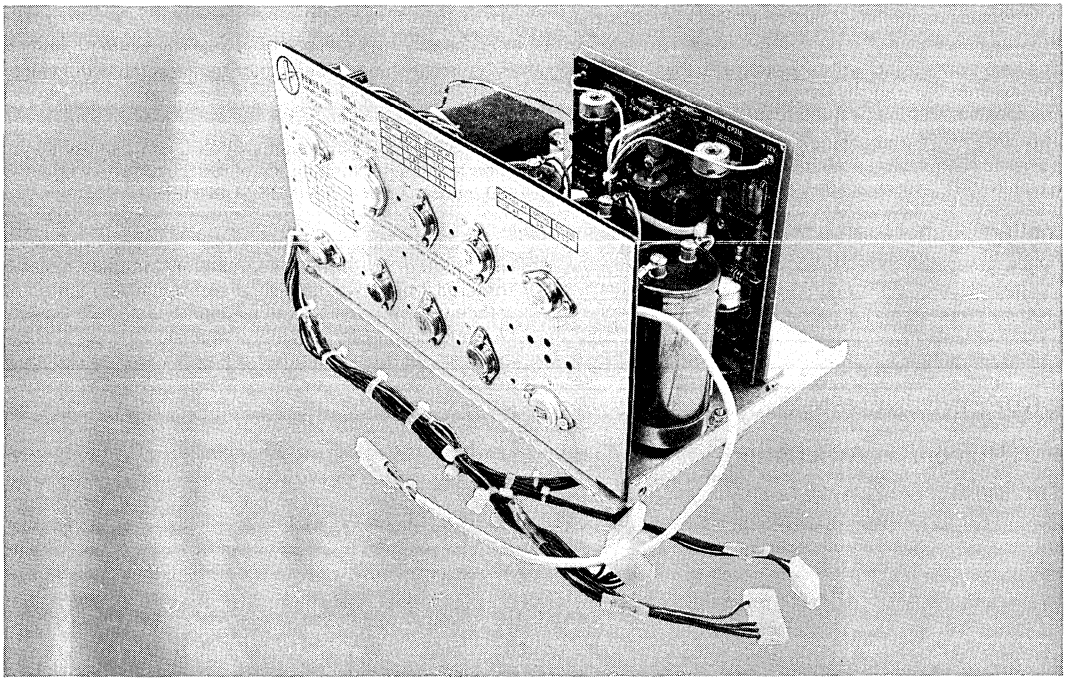
ORDERING INFORMATION

Part Number	Description
SBC 635	Power Supply

iSBC 640
POWER SUPPLY

- ± 5V and ± 12V iSBC 80/86 power
- Sufficient power for 8-12 MULTIBUS computer, memory, and peripheral boards
- Current limiting and overvoltage protection on all outputs
- "AC low" power failure TTL logic level output provided for system power-down control
- Compact single chassis/slide rail mounts in iCS 80 Industrial Chassis or OEM environments
- DC power cables and connectors mate directly to iSBC 604 Modular Cardcage/ Backplane assembly
- 100, 115, 215, and 230V AC operation
- 50 Hz or 60 Hz input

The iSBC 640 Power Supply provides low cost, off-the-shelf, single chassis power generation for OEM and industrial system products using Intel single board computers. The iSBC 640 supply provides regulated DC output power at + 12V, + 5V, − 5V and − 12V levels. The current capabilities of each of these output levels have been chosen to provide power over a 0 °C to + 55 °C temperature range for one fully loaded Intel single board computer, plus residual capability for most combinations of up to eleven iSBC memory, I/O, or combination expansion boards. Current limiting and overvoltage protection is provided on all outputs. Access for AC input is provided via a standard 4-pin keyed connector. DC output power levels are provided on cables with keyed connectors directly compatible with the iSBC 604/614 Modular Backplane/Cardcage assemblies. The iSBC 640 supply includes logic whose purpose is to sense system AC power failure and generate a TTL signal for clean system power-down control.



SPECIFICATIONS

Electrical Characteristics

Input Power
Frequency: 50 Hz ± 5%, 60 Hz ± 5%
Voltage: 115V ± 10%, 230V ± 10%, 215VAC ± 10%, 100VAC ± 10%
Via user configured wiring options

Output Power

Nominal Voltage	Current (Amps)(Max)	Current Limit Range (Amps)	Short Circuit (Amps)(Max)	Overvoltage Protection
+ 12V	4.5A	4.7- 6.8	2.3	15V ± 1V
+ 5V	30A	31.5-45.0	15.0	6.2V ± 0.4V
- 5V	1.75A	1.8- 3.2	0.9	- 6.2V ± 0.4V
- 12V	1.75A	1.8- 3.2	0.9	- 15V ± 1V

Combined Line/Load Regulation — ±1% at ±10% static line change and ±50% static load change, measured at the output connector (±0.2% measured at the power supply under the same conditions).

Remote Sensing — Provided for +5 VDC output line regulation.

Output Ripple and Noise — 10 mV peak-to-peak maximum (DC to 500 KHz)

Output Transient Response — Less than 50 μsec for ±50% load change.

Output Transient Deviation — Less than ± 10% of initial voltage for ± 50% load change.

Power Failure Indication (AC Low) — A TTL open collector high signal is provided when the input voltage drops below 90% of its nominal value. DC voltages will remain within 5% of their nominal values for 3.0 milliseconds (minimum, 7.5 ms typical) after AC Low goes true.

The "AC Low" signal will reset to a TTL low level when the AC input voltage is restored and after all output voltages are within specified regulation.

The "AC Low" threshold is adjustable for optimum powerdown performance at other input combinations (i.e. 100 VAC, 215 VAC, 50 Hz).

Mating Connectors[1]

AC Input

Housing	Molex	03-09-2042 or equivalent
Pin	Molex	02-09-2118 or equivalent (18 to 22 gauge wire)

DC Output[2]

Housing	Molex	26-03-3071
	Amp	3-87025-3
Pins	Molex	08-50-0187 or 08-50-0189
	Amp	87023-1
Key	Molex	15-04-9209
	Amp	87116-2

Compatible with Molex 09-66-1071 Header

Notes
1. Pins from given vendor may only be used with connectors from the same vendor.
2. iSBC 640 DC output connectors are directly compatible with input power connectors on iSBC 604 Modular Cardcage/Backplane assembly. Four connectors are provided.

Physical Characteristics

Height — 6.66 in. max. (16.92 cm)
Width — 8.19 in. max. (20.80 cm)
Depth — 12.65 in. max. (32.12 cm)
Weight — 30 lbs. max (13.63 kg)

Environmental Characteristics

Temperature — 0°C to 55°C with 55 CFM moving air
Non-Operating — - 40°C to + 85°C

Equipment Supplied

iSBC 640 Power Supply with AC and DC cables with keyed connectors.

Reference Manuals

9800803 — iSBC 640 Power Supply Hardware Reference Manual (includes schematic and assembly drawings) (NOT SUPPLIED)

9800798 — iCS 80 Systems Site Planning and Installation Manual (for installation of iSBC 640 supply into iCS 80 Industrial Chassis) (NOT SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

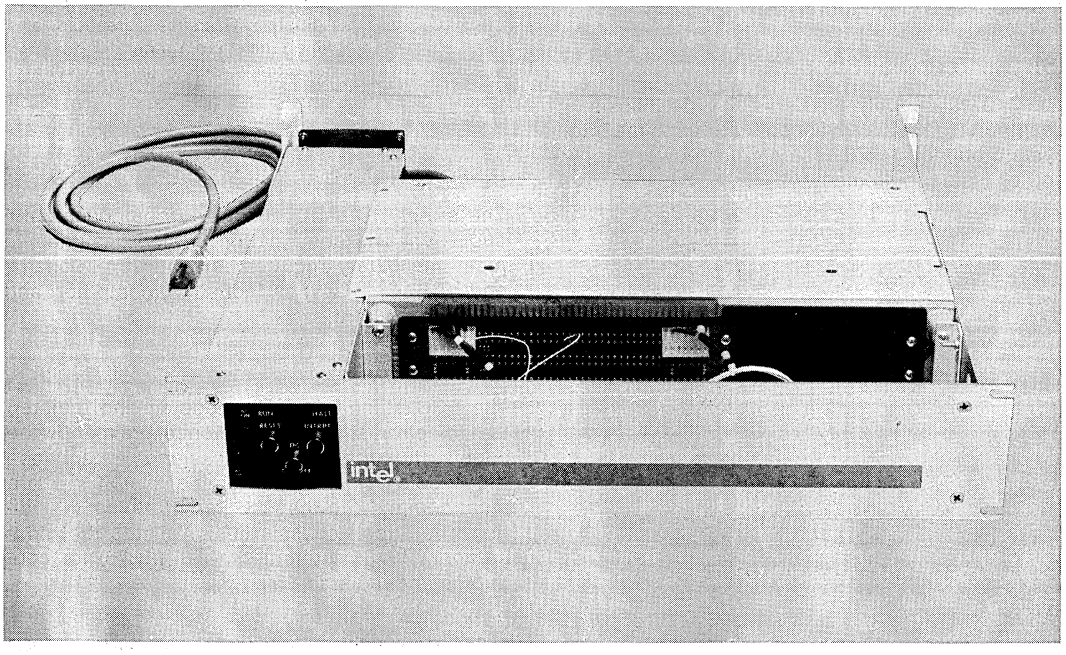
ORDERING INFORMATION

Part Number	Description
SBC 640	Power Supply

iSBC 665/iSBC 645™ SYSTEM CHASSIS AND POWER SUPPLY

- **Intel MULTIBUS™ system bus 4-slot packaging**
- **Complete package of rack-mounting, cooling, controls, and power**
- **Advanced 110 watt switching power supply generates ±5, ±12 VDC**
- **Meets U.S. and International EMI and safety requirements**
- **Wide AC voltage margins keep systems running during "brownouts"**
- **Front panel switches, indicators, and adjustments for operational and service convenience**
- **Power sense circuitry interrupts system 6 msec prior to power failure**

The Intel iSBC 665/iSBC 645 Chassis system provides the MULTIBUS system bus user with a compact set of products offering new standards in 4-slot rack-mount packaging. A high-efficiency switching power supply allows use of 115/230 VAC (+15/−20%), with large surge and noise components, to deliver smooth, stable DC power to the OEM board load. Advanced power-fail sense and restart logic gives the user sufficient time to bring the system to an orderly shutdown in the event of AC mains power failure. Mechanical design features include EMI suppression and a retainer/cover for system boards and I/O edge connectors.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: CREDIT, Index, Intel, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS, and the combination of MCS, ICE, iSBC, iSBX, iRMX or iCS, and a numerical suffix. 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 CORPORATION, 1981

FUNCTIONAL DESCRIPTION

iSBC 665™ System Chassis

The iSBC 665 Chassis is a complete microcomputer package providing four board slots in a 3.5″ vertical space.

RACK MOUNT PACKAGE

The iSBC 665 Chassis mounts in a 19″ EIA standard rack, using its front panel and hangers at the rear of the chassis to secure it to both sets of rails in the cabinet. If slide mounting is preferred, a tray with slides should be used as a platform for the chassis. The physical integrity of the system is enhanced by addition of a connector retainer at the (rear-facing) opening of the cardcage.

INTEGRAL COOLING

The fan on the power supply is utilized to draw ambient air across the boards prior to its being used to cool the supply.

FRONT PANEL

The front panel of the iSBC 665 Chassis forms a complete control center for the system installed in the chassis (see Figure 1).

iSBC 645 Power Supply

The 110-watt supply of the iSBC 665 Chassis is designed to provide advanced features to the Intel system builder who faces complex power supply and chassis requirements.

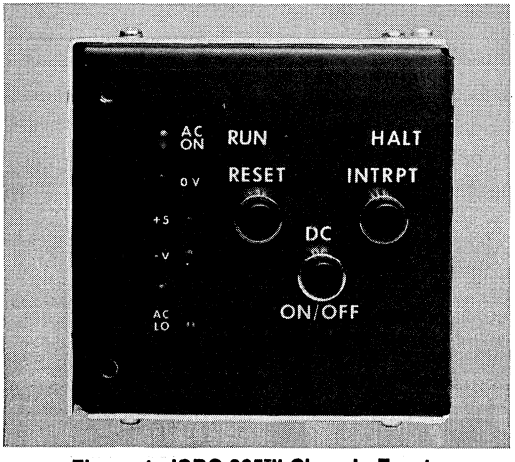


Figure 1. iSBC 665™ Chassis Front Panel Controls

Table 1. iSBC 665™ Chassis/iSBC 645™ Power Supply Control Panel Functions

Label	Function
Controls	
DC ON/OFF	Controls all DC power to the chassis.
RESET	Generates RESET/ signal to pin 14 of P1 (MULTIBUS system bus) backplane.
INTRPT	Generates INT/ signal to pin 42 of P1.
Indicators	
HALT, RUN	Indicate status of system CPU board.
AC ON	Indicates AC power present in supply (AC power switch is located at the rear of the supply).
OV	Indicates power supply shut-down due to an overvoltage condition on +5 or ± 12 VDC outputs.
AC LO	Indicates that AC voltage is below the operating range and the supply has shut down.
Adjustments	
+5	Adjusts +5 VDC output voltage.
−V	Adjusts negative adjustable voltage; set to −5 VDC at the factory.
AC LO	Adjusts AC sense threshold at which the system generates power-fail signals; set to 88/176 VAC at factory.

INTERNATIONAL ACCEPTANCE

The supply is a UL-recognized component; the supply/chassis combination is also designed to meet the safety requirements of CSA (Canada) and VDE (Germany) as industrial, computer, and office equipment.[1]

EMI STANDARDS

The FCC standards for conducted and radiated EMI (electromagnetic interference) are met by the supply, thus the chassis packaging will enhance the OEM's efforts to assemble systems which must comply with the FCC Part 15 Rules. In addition, the supply/chassis design meets the most stringent VDE requirements (0871/0875) for conducted and radiated EMI.[1]

NOTE:
1. CSA and VDE testing have not been officially completed; testing at independent laboratories indicates that these safety and EMI requirements are met. Official testing should be completed in late 1981.

BROWNOUT PROTECTION

The wide AC voltage input range allows microcomputer systems packaged in the chassis to function normally at extremely low AC voltage supply levels.

POWER-FAIL WARNING AND RECOVERY

In the event of a complete power failure, an interrupt is generated 6 ms prior to the supply's issuing a subsequent memory protect signal, giving sufficient time for execution of a user program to bring the entire system to an orderly shut-down.

POWER TRANSIENT TOLERANCE

The supply provides immunity for the system from the high-voltage transient surges and spikes seen in AC power systems. The supply itself provides this isolation with a metal-oxide varistor (MOV) and line filter in the input circuitry.

POWER LINE CLOCK

A clock signal is developed from the AC line at twice the line frequency; this gives the system user an extremely accurate time base.

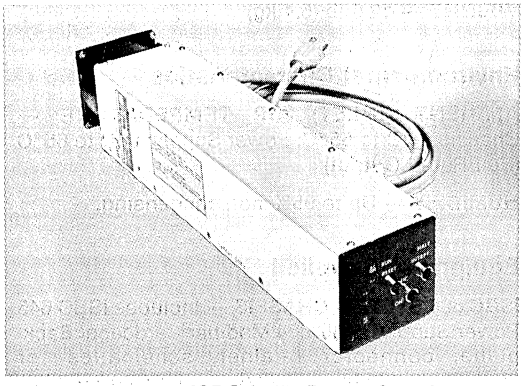


Figure 2. iSBC 645™ Power Supply

SPECIFICATIONS

Electrical Characteristics

INPUT POWER

Frequency: 47-66 Hz
Voltage: 115/230 VAC Single Phase
Range: 90 to 126 VAC/180 to 252 VAC
Consumption (Max.): 230 watts

OUTPUT POWER

Nominal Voltage	Current [2] (Max. Amps)	Current Limit Point (Amps)	Overvoltage Protection[3]
+5	15	18.75	5.25 to 6.25
+12	3	3.75	12.6 to 15.0
−12	1	1.25	−12.6 to −15.0
−Adjustable[4]	1	1.25	N/A

NOTES:
2. Total output power is 110 watts; a maximum of 128 watts is available, but proper operation of the power-fail circuitry is not guaranteed above 110 watts.
3. A minimum load is required on the +5 VDC output; this load must be at least ½ the sum of the loads (in watts) of the remaining three outputs.
4. −2.5 to −12 VDC; factory set to −5 VDC.

OUTPUT REGULATION (COMBINED LINE AND LOAD) — ±1% under any conditions of AC mains voltage variation (within operational range) and output load change.

PERIODIC AND RANDOM DEVIATION (PARD) — 50 millivolts peak-to-peak, all outputs.

LINE TRANSIENT TOLERANCE — A signal of up to 1000 VDC, with a pulse width of up to 50 microseconds, will have no affect on operation.

POWER FAIL INDICATION — An AC low condition generates ACLO and PFIN/ after AC voltage drops below the allowed voltage range. These signals are available on the P2 connector to generate interrupts. The DC voltages will remain within specifications for 6 milliseconds (worst case) following these interrupts, after which Memory Protect (MPRO/) will go true.

OUTPUT VOLTAGE TEMPERATURE COEFFICIENT — 0.03% per °C over the operating range.

SYSTEM CLOCK — 2× line frequency clock signal available on P2 connector.

Physical Characteristics (See Figure 3)

WIDTH — 19.0 in. (48.3 cm)
LENGTH — 16.25 in. (41.3 cm)
HEIGHT — 3.5 in. (8.9 cm)
WEIGHT — 12.0 lb (5.4 kg)

Environmental Characteristics

AMBIENT (INLET) AIR TEMPERATURE — Chassis: 0°C to 55°C; Power Supply: 0°C to 65°C (Full Rated Output)

HUMIDITY — Up to 95% non-condensing.

Equipment Supplied

iSBC 665 SYSTEM CHASSIS — Includes iSBC 645 Power Supply, iSBC 604 Modular Cardcage/ Backplane, connector retainer, schematics for cardcage/backplane, chassis, and power supply.

iSBC 645 POWER SUPPLY — Includes power supply, schematics for supply.

Reference Manuals (Not Supplied)

142836 — iSBC 665™ System Chassis Hardware Reference Manual

142918 — iSBC 645™ Power Supply Hardware Reference Manual

9800708 — iSBC 604/614™ Modular Cardcage/ Backplane Hardware Reference Manual

9088683 — Intel MULTIBUS Specification

Manuals may be ordered from any Intel sales representative, distribution office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

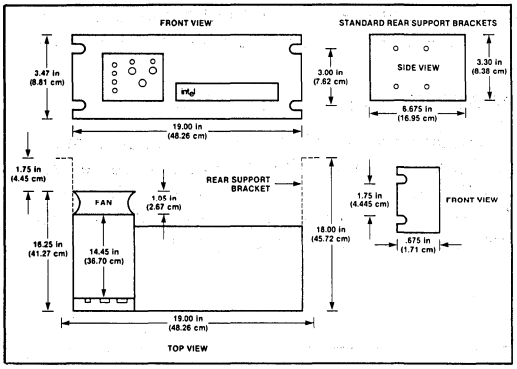


Figure 3. iSBC 665™ System Chassis Physical Dimensions

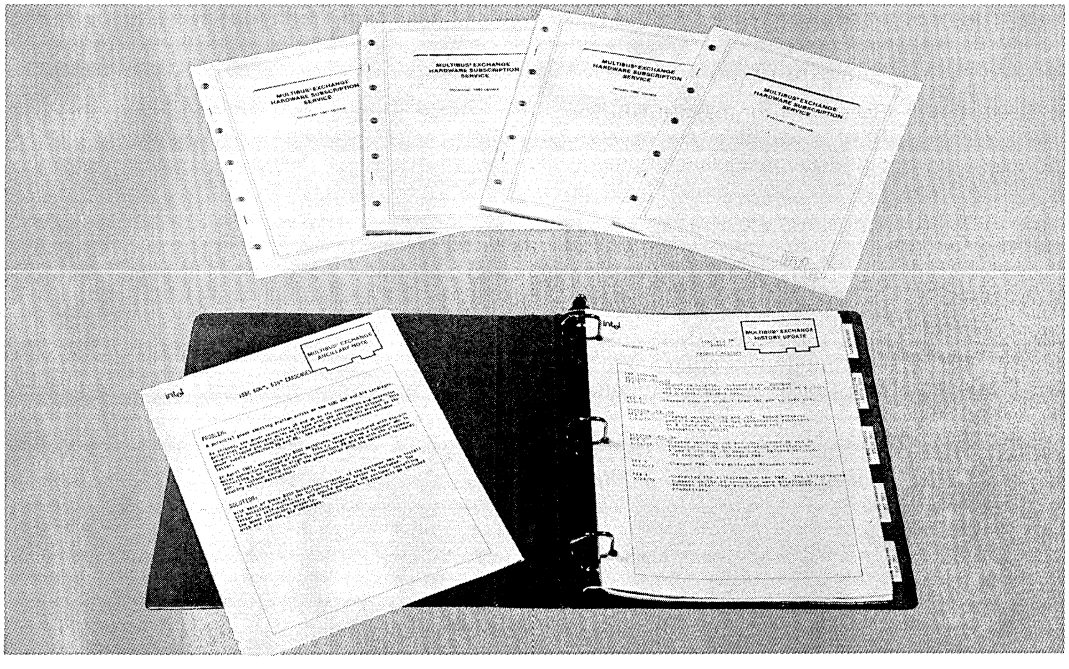
ORDERING INFORMATION

Part Number	Description
SBC 665	System Chassis
SBC 645	Power Supply (110 Watt)

iMBX 100/110/120
MULTIBUS® EXCHANGE
HARDWARE SUBSCRIPTION SERVICE

- **Monthly product updates**
- **History of Engineering Change Orders**
- **Customized to the subscriber's selection of Intel® board level products**
- **Ancillary Notes clarifying and correcting product documentation**
- **Offers full update alternatives — documentation, parts kit, and factory update**
- **Timely announcements of technological and product developments**
- **Available for iSBC,™ iCS,™ and iSBX™ products**

The Intel MULTIBUS EXCHANGE hardware subscription service consists of a monthly update publication which provides summary descriptions of Engineering Change Orders that alert and inform subscribers of all the latest developments and/or improvements applicable to their Intel board level products. In addition, the update contains order numbers and prices for optional update parts kits. This timely service allows users to closely track product changes and, by evaluating or implementing these changes, to take advantage of technological and product developments.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, CREDIT, Index, Insite, Intellec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, UPI, μScope, Promware, MCS, ICE, iRMX, iSBC, iSBX, MULTIMODULE and iCS. 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 CORPORATION, 1981

October, 1981
Order Number: 210276-001

PRODUCTS

The MULTIBUS EXCHANGE hardware subscription service supports Intel manufactured Single Board Computer, Industrial Control Series and MULTIMODULE products. Users can subscribe to any combination of over 60 products.

NEW SUBSCRIBERS

New subscribers receive a history (back to the earliest purchase date of each product) for each subscribed-to product and a handsome notebook in which future updates may be orderly stored.

MONTHLY UPDATES

Each month MULTIBUS EXCHANGE subscribers receive an update containing summaries of Engineering Change Orders and documentation clarification articles relating to each subscribed-to product. Each change description includes the change made, the rationale behind the change, and the user's alternatives concerning updating his product. Subscribers receive the monthly update even if no changes have occured to their products.

ORDERS

Orders for the MULTIBUS EXCHANGE are a combination of a base (iMBX 100) order, plus orders for each product to be included (iMBX 110s and/or iMBX 120s). For each MULTIBUS board, power supply, or chassis, an iMBX 110 is ordered. For each MULTIMODULE board, an iMBX 120 is ordered.

CHARGES

For an annual subscription fee, subscribers receive twelve monthly updates. All new subscribers will also receive a history for each subscribed-to product at no extra charge. For additional specified charges, subscribers can purchase the associated documentation and parts kits listed in the MULTIBUS EXCHANGE updates.

ORDERING INFORMATION

Part Number	Description
MBX 100	Base order for the MULTIBUS EXCHANGE hardware subscription service for one year
MBX 110	MULTIBUS EXCHANGE subscription covering one MULTIBUS board, power supply or chassis for one year
MBX 120	MULTIBUS EXCHANGE subscription covering one MULTIMODULE board for one year

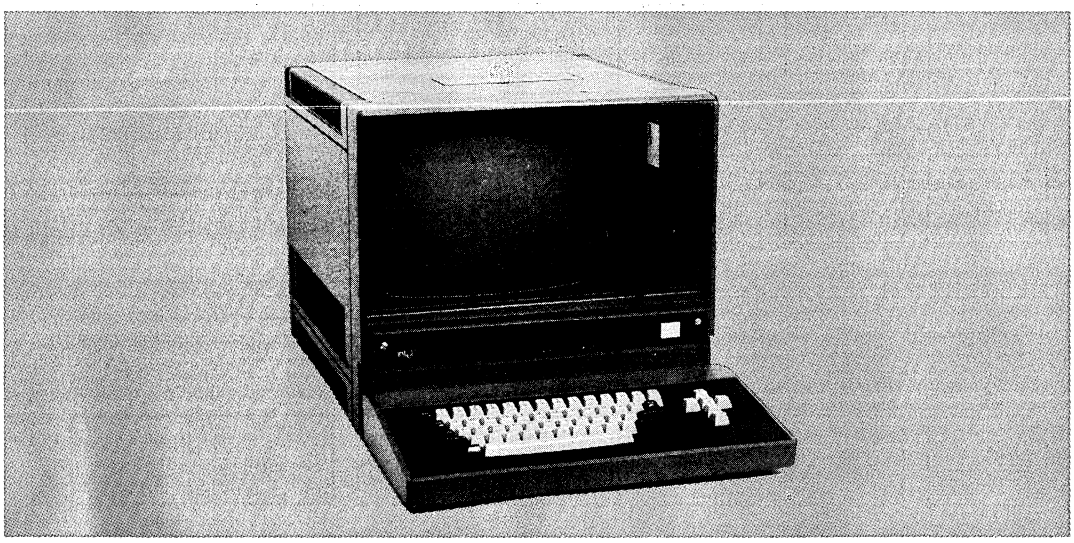
Microcomputer Development Systems

12

MODEL 225
INTELLEC® SERIES II/85
MICROCOMPUTER DEVELOPMENT SYSTEM

- **Complete microcomputer development system for MCS®-86, MCS®-85, MCS®-80, MCS®-48, and MCS®-51 microprocessor families**
- **High performance 8085A-2 CPU, 64K bytes RAM memory, and 4K bytes ROM memory**
- **Self-test diagnostic capability**
- **Built-in interfaces for high speed paper tape reader/punch, printer, and universal PROM programmer**
- **Integral 250K byte floppy disk drive with total storage capacity expandable to over 2M bytes of floppy disk storage and 7.3M bytes of hard disk storage**
- **Powerful ISIS-II Disk Operating System with relocating macroassembler, linker, locater, and CRT based editor CREDIT**
- **Supports PL/M, FORTRAN, BASIC, PASCAL and COBOL high level languages**
- **Software compatible with previous Intellec® systems**

The Intellec Series II/85 Model 225 Microcomputer Development System is a performance enhanced, complete microcomputer development system integrated into one compact package. The Model 225 includes a CPU with 64K bytes of RAM, 4K bytes of ROM, a 2000-character CRT, detachable full ASCII keyboard with cursor controls and upper/lower case capability, and a 250K-byte floppy disk drive. Powerful ISIS-II Disk Operating System software allows the Model 225 to be used quickly and efficiently for assembling and debugging programs for Intel's MCS-86, MCS-85, MCS-80, MCS-48, or MCS-51 microprocessor families. ISIS-II performs all file handling operations for the user, leaving him free to concentrate on the details of his own application. When used with an optional in-circuit emulator (ICE™) module, the Model 225 provides all of the hardware and software development tools necessary for the rapid development of a microcomputer-based product. Optional storage peripherals provide over 2 million bytes of floppy disk, and 7.3 million of hard disk storage capacity.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, Intellec, Multibus, i, iSBC, Multimodule, ICE, iSBX, PROMPT, iCS, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Micromap, μScope and the combination of ICE, iCS, iSBC, iSBX, MCS, or RMX and a numerical suffix.
© Intel Corporation 1980

121599-001 Rev. A

FUNCTIONAL DESCRIPTION

Hardware Components

The Intellec Series II/85 Model 225 is a highly-integrated microcomputer development system consisting of a CRT chassis with a 6-slot cardcage, power supply, fans, cables, single floppy disk drive, and two printed circuit cards. A separate, full ASCII keyboard is connected with a cable. A block diagram of the Model 225 is shown in Figure 1.

CPU Cards — The master CPU card contains its own microprocessor, memory, I/O, interrupt and bus interface circuitry implemented with Intel's high technology LSI components. Known as the integrated processor card (IPC), it occupies the first slot in the cardcage. A second slave CPU card is responsible for all remaining I/O control including the CRT and keyboard interface. This card, mounted on the rear panel, also contains its own microprocessor, RAM and ROM memory, and I/O interface logic, thus, in effect, creating a dual processor environment. Known as the I/O controller (IOC), the slave CPU card communicates with the IPC over an 8-bit bidirectional data bus.

Expansion — Five remaining slots in the cardcage are available for system expansion. Additional expansion of 4 slots can be achieved through the addition of an Intellec Series II expansion chassis.

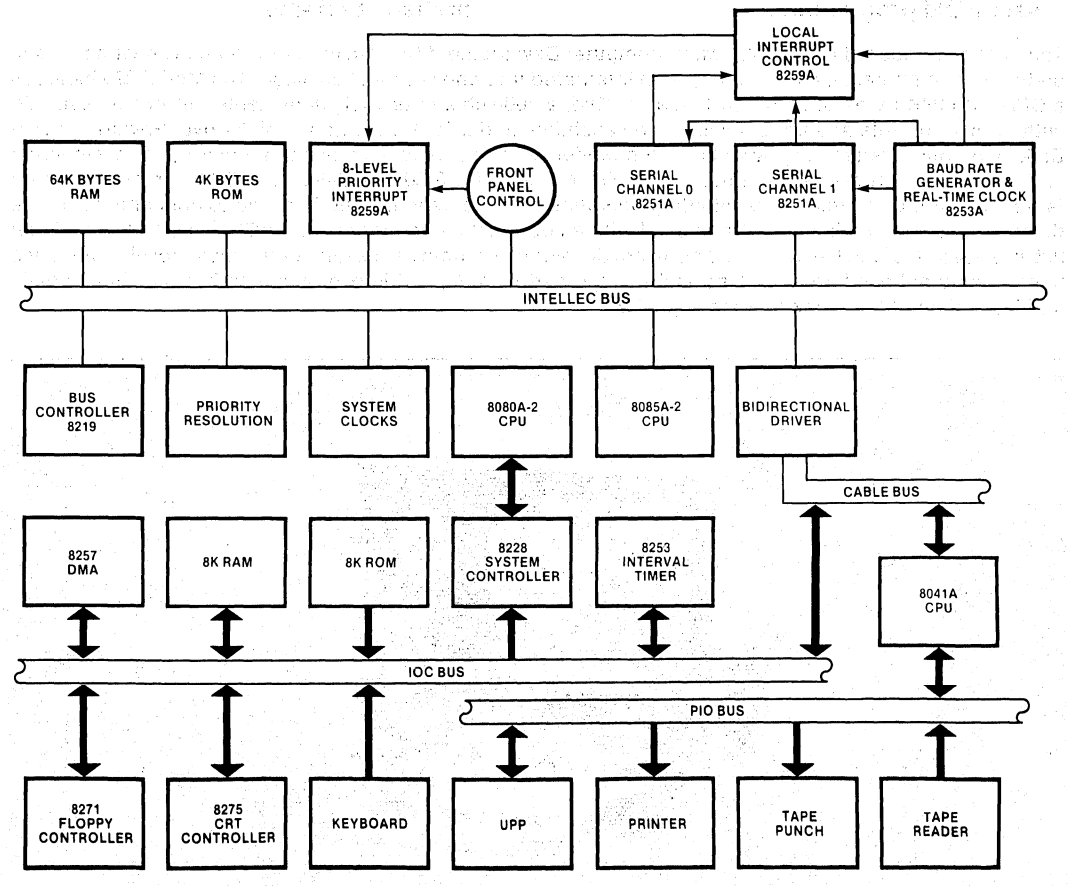


Figure 1. Intellec Series II/85 Model 225 Microcomputer Development System Block Diagram

System Components

The heart of the IPC is an Intel NMOS 8-bit microprocessor, the 8085A-2, running at 4.0 MHz. 64K bytes of RAM memory are provided on the board using 16K RAMs. 4K of ROM is provided, preprogrammed with system bootstrap "self-test" diagnostics and the Intellec Series II/85 System Monitor. The eight-level vectored priority interrupt system allows interrupts to be individually masked. Using Intel's versatile 8259A interrupt controller, the interrupt system may be user programmed to respond to individual needs.

Input/Output

IPC Serial Channels — The I/O subsystem in the Model 225 consists of two parts: the IOC card and two serial channels on the IPC itself. Each serial channel is RS232 compatible and is capable of running asynchronously from 110 to 9600 baud or synchronously from 150 to 56K baud. Both may be connected to a user defined data set or terminal. One channel contains current loop adapters. Both channels are implemented using Intel's 8251A USART. They can be programmed to perform a variety of I/O functions. Baud rate selection is accomplished through an Intel 8253 interval timer. The 8253 also serves as a real-time clock for the entire system. I/O activity through both serial channels is signaled to the system through a second 8259A interrupt controller, operating in a polled mode nested to the primary 8259A.

IOC Interface — The remainder of system I/O activity takes place in the IOC. The IOC provides interface for the CRT, keyboard, and standard Intellec peripherals including printer, high speed paper tape reader/punch, and universal PROM programmer. The IOC contains its own independent microprocessor, an 8080A-2. The CPU controls all I/O operations as well as supervising communications with the IPC. 8K bytes of ROM contain all I/O control firmware. 8K bytes of RAM are used for CRT screen refresh storage. These do not occupy space in Intellec Series II main memory since the IOC is a totally independent microcomputer subsystem.

Integral CRT

Display — The CRT is a 12-inch raster scan type monitor with a 50/60 Hz vertical scan rate and 15.5kHz horizontal scan rate. Controls are provided for brightness and contrast adjustments. The interface to the CRT is provided through an Intel 8275 single-chip programmable CRT controller. The master processor on the IPC transfers a character for display to the IOC, where it is stored in RAM. The CRT controller reads a line at a time into its line buffer through an Intel 8257 DMA controller and then feeds one character at a time to the character generator to produce the video signal. Timing for the CRT control is provided by an Intel 8253 interval timer. The screen display is formatted as 25 rows of 80 characters. The full set of ASCII characters is displayed, including lower case alphas.

Keyboard — The keyboard interfaces directly to the IOC processor via an 8-bit data bus. The keyboard contains an Intel UPI-41™ Universal Peripheral Interface, which scans the keyboard, encodes the characters, and buffers the characters to provide N-key rollover. The keyboard itself is a high quality typewriter style keyboard containing the full ASCII character set. An upper/lower case switch allows the system to be used for document preparation. Cursor control keys are also provided.

Peripheral Interface

A UPI-41 Universal Peripheral Interface on the IOC board provides interface for other standard Intellec peripherals including a printer, high speed paper tape reader, high speed paper tape punch, and universal PROM programmer. Communication between the IPC and IOC is maintained over a separate 8-bit bidirectional data bus. Connectors for the four devices named above, as well as the two serial channels, are mounted directly on the IOC itself.

Control

User control is maintained through a front panel, consisting of a power switch and indicator, reset/boot switch, run/halt light, and eight interrupt switches and indicators. The front panel circuit board is attached directly to the IPC, allowing the eight interrupt switches to connect to the primary 8259A, as well as to the Intellec Series II bus.

Integral Floppy Disk Drive

The integral floppy disk is controlled by an Intel 8271 single chip, programmable floppy disk controller. It transfers data via an Intel 8257 DMA controller between an IOC RAM buffer and the diskette. The 8271 handles reading and writing of data, formatting diskettes, and reading status, all upon appropriate commands from the IOC microprocessor.

MULTIBUS™ Interface Capability

All Intellec Series II/85 models implement the industry standard MULTIBUS protocol. The MULTIBUS protocol enables several bus masters, such as CPU and DMA devices, to share the bus and memory by operating at different priority levels. Resolution of bus exchanges is synchronized by a bus clock signal derived independently from processor clocks. Read/write transfers may take place at rates up to 5 MHz. The bus structure is suitable for use with any Intel microcomputer family.

SPECIFICATIONS

Host Processor (IPC)

8085A-2 based, operating at 4.0 MHz.
RAM — 64K on the CPU card
ROM — 4K (2K in monitor, 2K in boot/diagnostic)
Bus — MULTIBUS™ bus, maximum transfer rate of 5 MHz
Clocks — Host processor, crystal controlled at 4.0 MHz, bus clock, crystal controlled at 9.8304 MHz

I/O Interfaces

Two Serial I/O Channels, RS232C, at 110-9600 baud (asynchronous) or 150-56K baud (synchronous). Baud rates and serial format fully programmable using Intel 8251A USARTs. Serial Channel 1 additionally provided with 20 mA current loop. Parallel I/O interfaces provided for paper tape punch, paper tape reader, printer, and UPP-103 Universal PROM Programmer.

Interrupts

8-level, maskable, nested priority interrupt network initiated from front panel or user selected devices.

Direct Memory Access (DMA)

Standard capability on MULTIBUS interface; implemented for user selected DMA devices through optional DMA module—maximum transfer rate of 5 MHz.

Memory Access Time

RAM — 470 ns max
PROM — 540 ns max

Integral Floppy Disk Drive

Floppy Disk System Capacity —
250K bytes (formatted)
Floppy Disk System Transfer Rate —
160K bits/sec
Floppy Disk System Access Time —
Track to Track: 10 ms max
Average Random Positioning: 260 ms
Rotational Speed: 360 rpm
Average Rotational Latency: 83 ms
Recording Mode: FM

Physical Characteristics

CHASSIS

Width — 17.37 in. (44.12 cm)
Height — 15.81 in. (40.16 cm)
Depth — 19.13 in. (48.59 cm)
Weight — 73 lb. (33 kg)

KEYBOARD

Width — 17.37 in. (44.12 cm)
Height — 3.0 in. (7.62 cm)
Depth — 9.0 in. (22.86 cm)
Weight — 6 lb. (3 kg)

Electrical Characteristics

DC POWER SUPPLY

Volts Supplied	Amps Supplied	Typical System Requirements
+ 5 ± 5%	30.0	17.0
+12 ± 5%	2.5	1.1
−12 ± 5%	0.3	0.1
−10 ± 5%	1.0	0.08
+15 ± 5%*	1.5	1.5
+24 ± 5%*	1.7	1.7

*Not available on bus.

AC REQUIREMENTS FOR MAINFRAME

110V, 60 Hz — 5.9 Amp
220V, 50 Hz — 3.0 Amp

Environmental Characteristics

Operating Temperature — 16°C to 32°C (61°F to 90°F)
Humidity — 20% to 80%

Equipment Supplied

Model 225 Chassis including:
- Integrated Processor Card (IPC)
- I/O Controller Board (IOC)
- CRT
- ROM-Resident System Monitor

Detachable keyboard
ISIS-II System Diskette with MCS-80/MCS-85 Macroassembler
ISIS-II CREDIT Diskette CRT-Based Text Editor

Documentation Supplied

A Guide to Microcomputer Development Systems, 9800558

Intellec® Series II Model 22X/23X Installation Manual, 9800559

ISIS-II System User's Guide, 9800306

Intellec® Series II Hardware Reference Manual, 9800556

8080/8085 Assembly Language Programming Manual, 9800301

ISIS-II 8080/8085 Assembler Operator's Manual, 9800292

Intellec® Series II Systems Monitor Source Listing, 9800605

Intellec® Series II Schematic Drawings, 9800554

ISIS-II CREDIT (CRT-Based Text Editor) User's Guide, 9800902

Additional manuals may be ordered from any Intel sales representative or distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
MDS-225*	Intellec® Series II/85 Model 225 Microcomputer Development System (110V/60 Hz)
MDS-226*	Intellec® Series II/85 Model 226 Microcomputer Development System (220V/50 Hz)

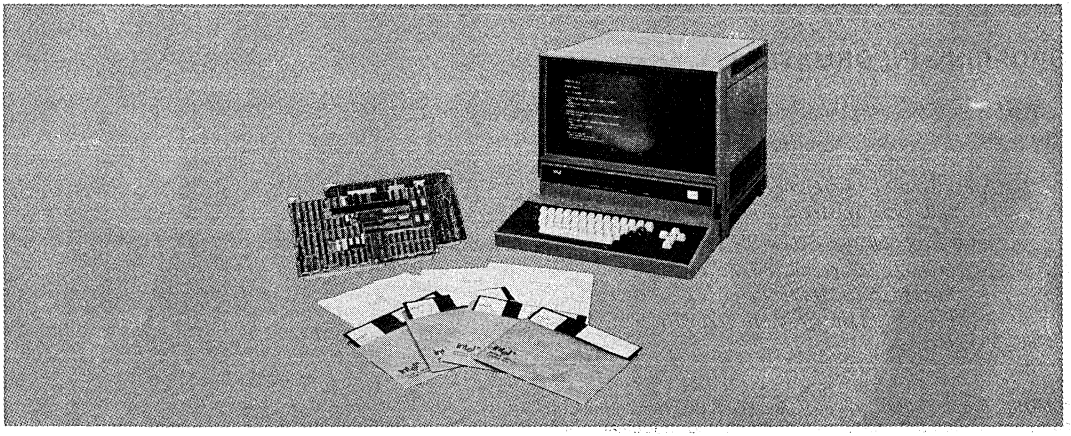
*"***MDS***"* is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.

MODEL 286
INTELLEC® SERIES III
MICROCOMPUTER DEVELOPMENT SYSTEM

- **Supports Intellec 432/100 Evaluation and Educational System**
- **Compatible with iSBC-090 Series 90 Memory System Upgrade: 512K Byte to 1M Byte**
- **Complete 16-bit High Performance, Microcomputer Development Solution for Intel iAPX 86,88 Applications. Also Supports MCS-85™, MCS-80 and MCS-48 Families**
- **Supports Full Range of iAPX 86,88-Resident, High-Level Languages: PL/M 86/88, PASCAL 86/88, and FORTRAN 86/88**
- **2 Host CPUs—iAPX 86 and 8085A—for Enhanced System Performance and Two Native Execution Environments**
- **96K Bytes of User Program RAM Memory Available for iAPX 86,88 Programs**
- **Series II/80 and Series II/85 Upgradeable to 8085/iAPX 86 Series III Functionality**
- **Intellec Model 800 Upgradeable to 8080/iAPX 86 Series III Functionality**
- **Compatible with Intellec Distributed Development Systems**
- **Compatible with Previous Intellec Systems**
- **Software Applications Debugger for User iAPX 86,88 Programs**
- **Upgradeable to a Complete Ethernet* Communications Development System Environment, Using the Model 677 Upgrade**

The Intellec Series-III Microcomputer Development System is a high-performance system solution designed specifically for iAPX 86,88 microprocessor development. It contains two host CPUs, an iAPX 86 and an 8085, that provide two native execution environments for optimum performance and compatibility with the Intellec software packages for both CPUs. The basic system includes 96K bytes of iAPX 86,88 user RAM memory and a 250K byte floppy disk drive. The powerful Disk Operating System maximizes system processing by utilizing the power of both host processors. Standard software includes a full range of iAPX 86,88 resident software. The high-level languages PL/M 86/88, PASCAL 86/88, and FORTRAN 86/88 are also available. A ROM resident software debugger not only provides self-test diagnostic capability, but also gives the user a powerful iAPX 86,88 applications debugger.

*Ethernet is a trademark of Xerox Corporation.



The following are trademarks of Intel Corporation and its affiliates and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{l}m, iCS, ICE, UPI, BXP, iSBC, iSBX, Insite, iRMX, System 2000, CREDIT, iRMX/80, MULTIBUS, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTIMODULE, and the combination of MCS, ICE, iSBC, iRMX or iCS and a numerical suffix. © INTEL CORPORATION, 1981.

AFN-01588B
121670-002

FUNCTIONAL DESCRIPTION

Hardware Components

The Intellec Series III is contained in a single package consisting of a CRT chassis with a 6-slot card cage, power supply, fans, cables, single floppy disk drive, detachable upper/lower case full ASCII keyboard, and four printed circuit cards. A block diagram of the system is shown in Figure 1.

System Components

Two CPU cards reside on the Intellec MULTIBUS bus, each containing its own microprocessor, memory, I/O, interrupt and bus interface circuitry implemented with Intel's high technology LSI components. The integrated processor card (IPC-85), occupies the first slot in the cardcage. A second CPU card, the resident processor board (RPB-86) contains Intel's 16-bit HMOS microprocessor. These CPUs provide the dual processor environment.

A third CPU card performs all remaining I/O including interface to the CRT, integral floppy disk, and keyboard. This card, mounted on the rear panel, contains its own microprocessors, RAM and ROM memory, and I/O interface logic. Known as the I/O controller (IOC), this slave CPU card communicates with the IPC-85 over an 8-bit bidirectional data bus. A 64K byte RAM expansion memory board is also included.

Expansion

Two additional slots in the system cardcage are available for system expansion. The Intellec expansion chassis Model 201 is available to provide 4 additional expansion slots for either memory or I/O expansion.

THE INTELLEC DEVELOPMENT SYSTEM FOR ETHERNET (DS/E)

The Intellec Series III can be expanded to provide the user with the tools necessary to develop and test

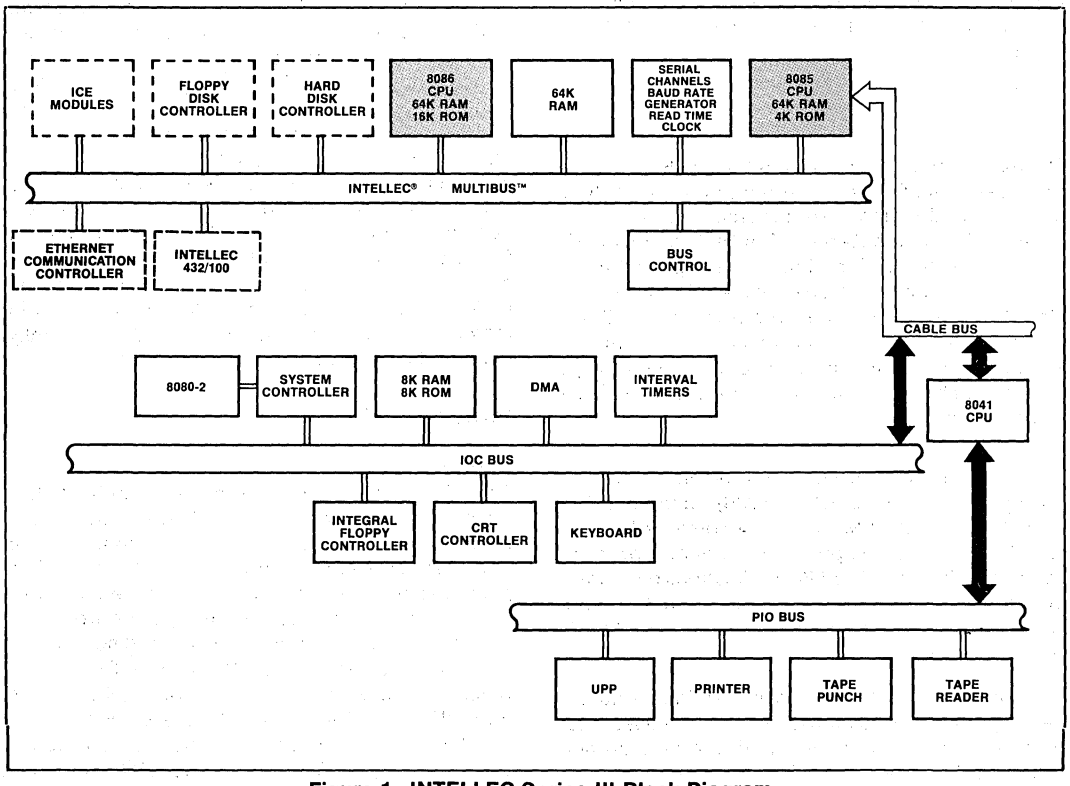


Figure 1. INTELLEC Series III Block Diagram

AFN-01588B

communications software and applications that will use Ethernet as a communications subsystem. The power of the Intellec Series III combined with Model 677 allows the user to develop either 8- or 16-bit Ethernet-based applications.

THE INTELLEC 432/100 EVALUATION AND EDUCATIONAL SYSTEM

The Intellec Series III provides a complete system environment necessary for evaluation of the Intel iAPX 432 32-bit micromainframe. The iSBC 432/100 board plugs into a Multibus slot in the Intellec Series III, sharing system memory and resources. A comprehensive set of documentation, system software and hardware provides the evaluation and educational environment for the powerful iAPX 432.

iAPX 286 Evaluation System

The Intellec Series III provides a complete system environment for evaluation of the iAPX 286 microprocessor's architecture and its instruction set, segmentation timing, memory mapping and protection features. A user can begin the development of complex iAPX 286 programs, systems and operating system nuclei with the Intellec Series III and iAPX 286 evaluation package.

CPU Cards

IPC-85

The heart of the IPC-85 is an Intel NMOS 8-bit microprocessor, the 8085A-2, running at 4.0 MHz. 64K bytes of RAM memory are provided on the board using 16K dynamic RAMs. 4K of ROM is provided, preprogrammed with system bootstrap "self-test" diagnostics and the Intellec System Monitor. The eight-level vectored priority interrupt system allows interrupts to be individually masked. Using Intel's versatile 8259A interrupt controller, the interrupt system may be user programmed to respond to individual needs.

RPB-86

The heart of the RPB-86 is an Intel HMOS 16-bit microprocessor, the iAPX 86 (8086), running at 5.0 MHz. 64K bytes of RAM memory are provided on the board. 16K of ROM is provided on board, preprogrammed with an iAPX 88/86 applications debugger which provides features necessary to debug and execute application software for the iAPX 88/86 microprocessors.

The 8085A-2 and iAPX 86 access two independent memory spaces. This allows the two processors to execute concurrently when an iAPX 88/86 program is run. In this mode, the IPC-85 becomes an intellegent I/O processor board to the RPB-86.

Input/Output

IPC-85 SERIAL CHANNELS

The I/O subsystem in the Series III consists of two parts: the IOC card and two serial channels on the IPC-85 itself. Each serial channel is independently configurable. Both are RS232-compatible and is capable of running asynchronously from 110 to 9600 baud or synchronously from 150 to 56K baud. Both may be connected to a user defined data set or terminal. One channel contains current loop adapters. Both channels are implemented using Intel's 8251A USART. They can be programmed to perform a variety of I/O functions. Baud rate selection is accomplished through an Intel 8253 interval timer. The 8253 also serves as a real-time clock for the entire system. I/O activity through each serial channel is independently signaled to the system through a second 8259A (slave) interrupt controller, operating in a polled mode nested to the master 8259A.

IOC INTERFACE

The remainder of the system I/O activity is handled by the IOC. The IOC provides the interface and control for the keyboard, CRT, integral floppy disk drive, and standard Intellec-compatable peripherals including printer, high speed paper tape reader/punch, and universal PROM programmer. The IOC contains its own independent microprocessor, an 8080A-2. This CPU issues commands, receives status, and controls all I/O operations as well as supervising communications with the IPC-85. The IOC contains interval timers, its own IOC bus system controller, and 8K bytes of ROM for all I/O control firmware. The 8K bytes of RAM are used for CRT screen refresh storage. Neither the ROM nor the RAM occupy space in the Intellec Series III main memory address range because the IOC is a totally independent microcomputer subsystem.

Integral CRT

DISPLAY

The CRT is a 12-inch raster scan type monitor with a 50/60 Hz vertical scan rate and 15.5 kHz horizontal scan rate. Controls are provided for brightness and contrast adjustments. The interface to the CRT is provided through an Intel 8275 single chip programmable CRT controller. The master processor on the IPC-85 transfers a character for display to the IOC, where it is stored in RAM. The CRT controller reads a line at a time into its line buffer through an Intel 8257 DMA Controller. It then feeds one character at a time to the character generator to produce the video signal. Timing for the CRT control is provided by an Intel 8253 programmable interval

timer. The screen display is formatted as 25 rows of 80 characters. The full set of ASCII characters are displayed, including lower case alphas.

KEYBOARD
The keyboard interfaces directly to the IOC processor via an 8-bit data bus. The keyboard contains an Intel UPI-41A Universal Peripheral Interface, which scans the keyboard and encodes the characters to provide N-key roll over. The keyboard itself is a typewriter style keyboard containing the full ASCII character set. An upper/lower case switch allows the system to be used for document preparation. Cursor control keys are also provided.

Peripheral Interface

A UPI-41A Universal Peripheral Interface on the IOC board provides built-in interface for standard Intellec-compatable peripherals including a printer, high speed paper tape reader, high speed paper tape punch, and universal PROM programmer. Communication between the IPC-85 and IOC is maintained over a separate 8-bit bidirectional data bus. Connectors for the four devices named above, as well as the two serial channels, are mounted directly on the IOC itself.

Control

User control is maintained through a front panel, consisting of a power switch and indicator, reset/boot switch, run/halt light and eight interrupt switches and LED indicators. The front panel circuit board is attached directly to the IPC-85, allowing the eight interrupt switches to connect the master 8259A, as well as to the Intellec Series III bus.

User program control in the iAPX 88/86 environment of the Intellec Series III is also directed through keyboard control sequences to transfer control to the iAPX 88/86 applications debugger, abort a user program or translator and returning control to the IPC-85.

DISK SYSTEM

Integral Floppy Disk Drive

The integral floppy disk is controlled by an Intel 8271 single chip, programmable floppy disk controller. The disk provides capacity of 250K bytes. It transfers data via an Intel 8257 DMA Controller between an IOC RAM buffer and the diskette. The 8271 handles reading and writing of data, formatting diskettes, and reading status, all upon appropriate commands from the IOC microprocessor.

Dual Drive Floppy Disk System (Option)

The Intellec Series III Double Density Diskette System provides direct access bulk storage, intelligent controller and two diskette drives. Each drive provides 1/2 million bytes of storage with a data transfer of 500,000 bits/second. The controller is implemented with Intel's powerful Series 3000 Bipolar Microcomputer Set and supports up to four diskette drives to allow more than 2 million bytes of on-line storage.

The diskette controller consists of two boards, the channel board and the interface board. These two PC boards reside in the Intellec Series III system chassis. The channel board receives, decodes and responds to channel commands from the 8085A-2 CPU on the IPC-85. The interface board provides the diskette controller with a means of communication with the disk drives and with the Intellec system bus. The interface board also validates data during disk transactions.

An additional cable and connectors are also supplied to optionally convert the integral floppy disk from single density to double density.

Hard Disk System (Option)

The Intellec Series III Hard Disk System provides direct access bulk storage, intelligent controller and a disk drive containing one fixed platter and one removable cartridge. Each provides approximately 3.65 million bytes of storage with a data transfer rate of 2.5 Mbits/second. The controller is implemented with Intel's Series 3000 Bipolar Microcomputer Set. The controller provides an interface to the Intellec Series III system bus, as well as supporting up to 2 disk drives. The disk system records all data in Double Frequency (FM) on 2 surfaces per platter. Each platter can be write protected by a front panel switch.

HARD DISK CONTROLLER BOARDS
The disk controller consists of two boards which reside in the Intellec Series III system chassis. The disk system is capable of performing six operations: recalibrate, seek, format track, write data, read data, and verify CRC. In addition to supporting a second drive, the disk controller may co-exist with the double-density diskette controller to allow up to 17 million bytes of on-line storage.

MULTIBUS Interface Capability

All models of the Intellec Series III implement the industry standard MULTIBUS protocol. The MULTIBUS architecture allows several bus masters, such as CPU and DMA devices, to share the bus and memory by operating at different priority levels. Resolution of bus exchanges is synchronized by a bus clock signal derived independently from processor clocks. Read/write transfers may take place at rates up to 5 MHz. The bus structure is suitable for use with any Intel microcomputer family.

System Software Features

The Model 286 offers many key advantages for iAPX 86,88 applications and Intellec Development Systems: enhanced system performance through a dual host CPU environment, a full spectrum of iAPX 86,88-resident high-level languages, expanded user program space for iAPX 86,88 programs, and a powerful high-level software applications debugger for iAPX 86,88 microprocessor software.

Dual Host CPU—The addition of a 16-bit 8086 to the existing 8-bit host CPU increases iAPX 86,88 compilation speeds and provides for iAPX 86,88 code execution. When the 8086 is executing a program, the 8-bit CPU off-loads all I/O activity and operates as an intelligent I/O controller to double buffer data to and from the 8086. The 8086 also provides an execution vehicle for 8086 and 8088 object code. An added benefit of two host microprocessors is that 8-bit translations and applications are handled by the 8-bit CPU, and 16-bit translations and applications are handled by the 8086. This feature provides complete compatibility for current systems and means that software running on current Intellec Development Systems will run on the new system.

High-Level Languages for iAPX 86,88—The Model 286 allows the current Intellec system user to take advantage of a breadth of new resident iAPX 86,88 high-level languages: PL/M 86/88, PASCAL 86/88, and FORTRAN 86/88. The iAPX 86,88 Resident Macro Assembler and these high-level language compilers execute on the 8086 host CPU, thereby increasing system performance.

Expanded Program Memory—By adding a Model 286 to an existing Intellec Development System, 96K bytes of user program RAM memory are made available for iAPX 86,88 programs. System memory is expandable by adding additional RAM memory modules. This, combined with the two host CPU system architecture, dramatically increases the processing power of the system.

Software Applications Debugger—The RPB-86 contains the applications debugger which allows iAPX 86,88 programs to be developed, tested, and debugged within the Intellec system. The debugger provides a subset of In-Circuit Emulator commands such as symbolic debugging, control structures and compound commands specifically oriented toward software debug needs.

SPECIFICATIONS

Host Processor Boards

INTEGRATED PROCESSOR CARD
—(IPC-85) 8085A-2 based, operating at 4 MHz
—64K RAM, 4K ROM (2K in monitor and 2K in boot/diagnostic)

RESIDENT PROCESSOR BOARD
—(RPB-86) 8086 based, operating at 5 MHz, 64K RAM, 16K ROM (applications debugger)

BUS
—MULTIBUS bus, maximum transfer rate of 5 MHz

DIRECT MEMORY ACCESS
—(DMA) Standard capability on the MULTIBUS bus; implemented for user selected DMA devices through optional DMA module
—Maximum transfer rate of 2 MHz

Integral Floppy Disk

Capacity—250K bytes (formatted)
Transfer Rate—160K bits/sec
Access Time—
Track to Track: 10 ms max.
Average Random Positioning: 260 ns
Rotational Speed: 360 rpm
Average Rotational Latency: 83 ms
Recording Mode: FM

Dual Floppy Disk Option

Capacity—
Per Disk: 4.1 megabits (formatted)
Per Track: 53.2 kilobits (formatted)
Transfer Rate—500 kilobits/sec
Access Time—
Track to Track: 10 ms
Head Setting Time: 10 ms
Average Random Positioning Time—260 ms

Rotational Speed—360 rpm
Average Rotational Latency: 83 ms
Recording Mode: M^2 FM

Hard Disk Drive Option

Type—5440 top loading cartridge and one fixed platter
Tracks per Inch—200
Mechanical Sectors per Track—12
Recording Technique—double frequency (FM)
Tracks per Surface—400
Density—2,200 bits/inch
Bits per Track—62,500
Recording Surfaces per Platter—2
Capacity—
Per Surface—15M bits
Per Platter—29M bits
Per Drive—59M bits
Per Drive—7.3M bytes (formatted)
Transfer Rate—2.5M bits/sec
Access Time—
Track to Track: 13 ms max
Full Stroke: 100 ms
Rotational Speed: 2,400 rpm

Physical Characteristics

Width—17.37 in. (44.12 cm)
Height—15.81 in. (40.16 cm)
Depth—19.13 in. (48.59 cm)
Weight—81 lb. (37 kg)

KEYBOARD
Width—17.37 in. (44.12 cm)
Height—3.0 in. (7.6 cm)
Depth—9.0 in. (22.86 cm)
Weight—6 lb. (3 kg)

DUAL FLOPPY DRIVE SYSTEM (OPTION)
Width—16.88 in. (42.88 cm)
Height—12.08 in. (30.68 cm)
Depth—1.0 in. (48.26 cm)
Weight—64 lb. (29 kg)

HARD DISK DRIVE SYSTEM (OPTION)
Width—18.5 in. (47.0 cm)
Height—34.0 in. (86.4 cm)
Depth—29.75 in. (75.6 cm)
Weight—202 lb. (92 kg)

ELECTRICAL CHARACTERISTICS

DC Power Supply

Volts Supplied	Amps Supplied	Typical System Requirements
+ 5 ± 5%	30.0	17.0
+12 ± 5%	2.5	1.1
−12 ± 5%	0.3	0.1
−10 ± 5%	1.0	0.08
+15 ± 5%*	1.5	1.5
+24 ± 5%*	1.7	1.7

*Not available on bus

AC Requirements for Mainframe

110V, 60 Hz—5.9 Amp
220V, 50 Hz—3.0 Amp

ENVIRONMENAL CHARACTERISTICS

System Operating Temperature—0° to 35°C (32°F to 95°F)
Humidity—20% to 80%

DOCUMENTATION SUPPLIED

Intellec Series III Microcomputer Development System Product Overview, 121575

A Guide to Intellec Series III Microcomputer Development Systems, 121632-001

Intellec Series III Microcomputer Development System Console Operating Instructions, 121609

Intellec Series III Microcomputer Development System Pocket Reference, 121610

Intellec Series III Microcomputer Development System Programmer's Reference, 121618

iAPX 88/86 Family Utilities User's Guide for 8086-Based Development Systems, 121616

8086/8087/8088 Macro Assembly Language Reference Manual for 8086-Based Development Systems, 121627

8086/8087/8088 Macro Assembly Language Pocket Reference, 9800749

8086/8087/8088 Macro Assembler Operating Instructions for 8086-Based Development Systems, 121628

Intellec Series III Microcomputer Development System Installation and Checkout Manual, 121612

Intellec Series III Microcomputer Development System Schematic Drawings, 121642

ISIS-II CREDIT (CRT-Based Text Editor) User's Guide, 9800902

ISIS-II CREDIT (CRT-Based Text Editor) Pocket Reference, 9800903

The 8086 Family User's Manual, 9800722

The 8086 Family User's Manual, Numeric Supplement, 121586

For Series III Plus Hard Disk Systems Only:

Model 740 Hard Disk Subsystem Operation and Checkout, 9800943

ORDERING INFORMATION

Part Number	Description
DS286 KIT	Intellec Series III Model 286 Microcomputer Development System (110V/60Hz)
DS287 KIT	Intellec Series III Model 287 Microcomputer Development System (220V/50Hz)
DS286FD KIT	Intellec Series III Model 286 Microcomputer Development System with Dual Double Density Flexible Disk System (110V/60Hz)
DS287FD KIT	Intellec Series III Model 287 Microcomputer Development System with Dual Double Density Flexible Disk System (220V/50Hz)
DS286HD KIT	Intellec Series III Model 286 Microcomputer Development System with Pedestal Mounted Hard Disk. (110V/60Hz)
DS287HD KIT	Intellec Series III Model 287 Microcomputer Development System with Pedestal Mounted Hard Disk. (220V/50Hz)

Requires Software License

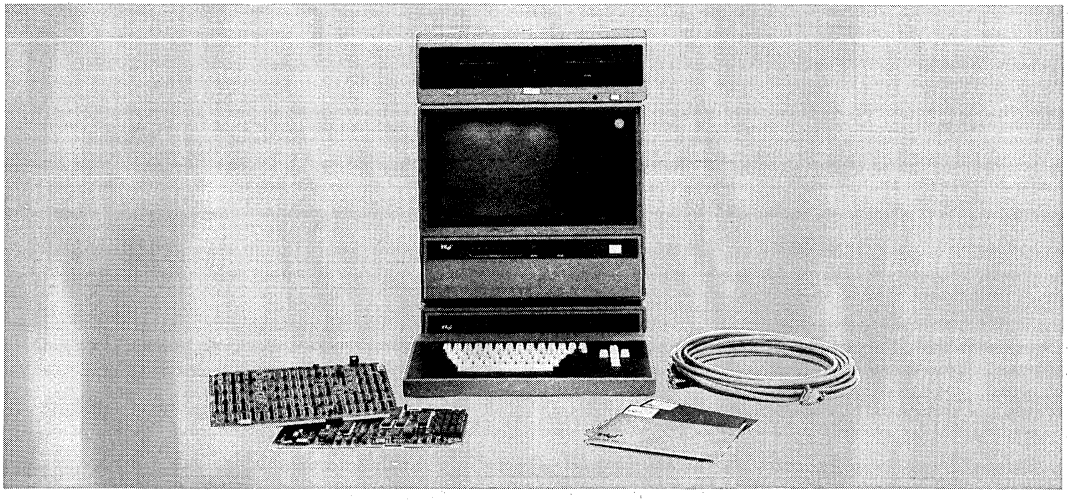
MODEL 675
INTELLEC® DEVELOPMENT SYSTEM
FOR ETHERNET* DS/E

- **Complies with the Intel, DEC, and Xerox Tricompany Ethernet Specification**
- **Provides a complete Ethernet Communications Development System Environment**
- **Includes an Ethernet Data Link Layer Software Library to allow user programs to access the Ethernet Data Link from the 8085 in Model 675 systems**
- **Supports the Ethernet Data Link and Physical Link Control via the MULTIBUS Ethernet Communications Controller**
- **Includes a special 10 meter Interconnect Cable for connecting two Model 675s for Ethernet software prototyping**
- **Supports the full range of iAPX 86, 88-resident, High-Level Languages: PL/M 86/88; PASCAL 86/88; and FORTRAN 86/88**
- **Includes a Software Applications Debugger for iAPX 86, 88 user programs**
- **Upgradable from Intellec Series II/85 and Series III**

The Intellec Development System for Ethernet (DS/E) provides the user with the tools necessary to develop and test communication software and applications that will use Ethernet as a communication subsystem. It combines the power of the Intellec Microcomputer Development System with a dual board Ethernet Communications Controller for microprocessor development. This combination allows the user to develop either 8- or 16-bit Ethernet-based applications.

The Ethernet Communications Controller incorporates the Ethernet Data Link and Physical Link Control to meet the Ethernet specification for 10 Mbit per second data transmission rate over coaxial cable. The controller consists of two MULTIBUS compatible boards, the Processor board and the Serialization/Deserialization (SerDes) board. The Processor board provides the function of packet buffering, processing, and transferring of the processed packets to system memory. The SerDes board performs the serialization/deserialization, framing, CRC generation and checking, Manchester encoding/decoding, and destination address recognition. A 10 meter interconnect cable is included to permit two Model 675s to be connected in a functioning Ethernet environment without the need for Ethernet transceivers and coaxial cable.

*Ethernet is a trademark of Xerox Corporation.



Intel Corporation Assumes No Responsibilty for the Use of Any Circuitry Other Than Circuitry Embodied in an Intel Product. No Other Circuit Patent Licenses are Implied.
© INTEL CORPORATION, 1981.

COMPONENTS

Hardware Components

The Model 675 consists of a CRT chassis with a 6-slot cardcage, power supply, fans, cables, single floppy diskette drive, a detachable upper/lower case full ASCII keyboard, and six printed circuit boards. A block diagram of the Model 675 is shown in Figure 1.

System Components

Two CPU boards, the IPC-85 and the RPB-86, reside on the Model 675 MULTIBUS system bus, each containing its own microprocessor, memory, I/O, interrupt and bus interface circuitry implemented with Intel's high technology LSI components. The IPC-85 integrated processor board consists of an Intel NMOS 8-bit microprocessor, the 8085A-2, and 64 Kbytes of on board memory. The RPB-86 resident processor board contains Intel's HMOS 16-bit microprocessor, the iAPX 86 (8086), and 64 Kbytes of on board memory.

A third CPU board, known as the I/O controller (IOC) performs all remaining I/O including the interface to the CRT, integral floppy disk, and keyboard. The IOC contains its own microprocessor, RAM and ROM memory, and I/O interface logic. It is a slave CPU board which communicates with the IPC-85 over an 8-bit bidirectional data bus.

The Model 675 also includes an additional 64 Kbyte MULTIBUS compatible RAM board supplying the user with a total of 192 Kbytes of RAM. The two remaining boards are the Ethernet Communications Controller.

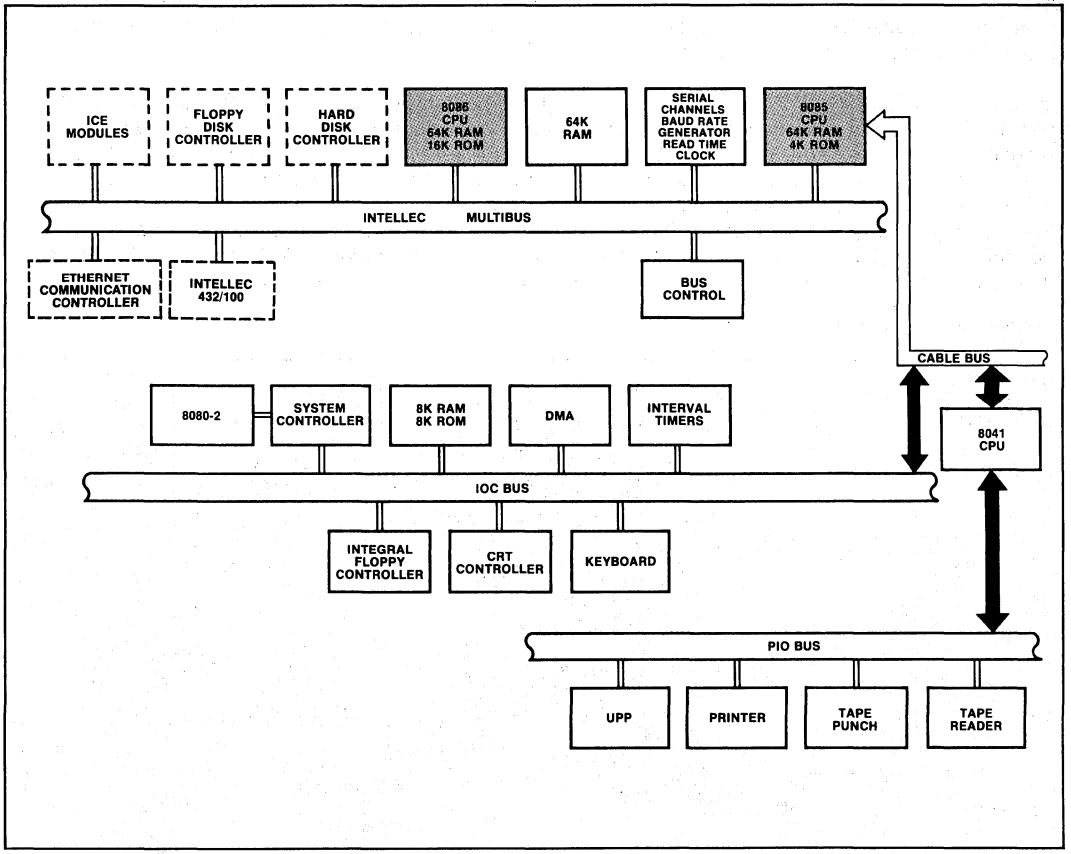


Figure 1. Model 675 DS/E Block Diagram

Expansion

The Intellec expansion chassis Model 201 is included in the basic Model 675. All Model 675 systems contain 10 MULTIBUS board slots—6 in the main system and 4 in the Model 201 chassis. The table below indicates the number of slots available for expansion in each.

Model	Available Expansion Slots
675	4
675FD	2
675HD	2

Ethernet Communications Controller

PROCESSOR BOARD

The processor board interfaces to the MULTIBUS system bus and contains the Intel 5 MHz 8088 CPU, 16 Kbytes dynamic RAM for Ethernet and host interface program execution, 8K EPROM pre-programmed for the Data Link Control, and an 8K static RAM for transmit and receive channel DMA data buffering. The Processor board provides the necessary commands and control to the SerDes board and receives status and data from it.

SerDes BOARD

The SerDes board meets the required electrical specification to the transceiver and provides the Data Link Layer of the Ethernet architecture. The major functions of the SerDes board include serialization/deserialization, framing, Manchester encoding/decoding, transmit data flow control, receive data flow control, destination address decoding for received message, CRC generation and checking, and diagnostic for CRC error, loopback, transmit timeout, and if used with transceivers, CSMA/CD (carrier-sense multiple-access with collision-detection).

DATA BUFFERING

All data transfers from the Ethernet Data Link control of the SerDes board are buffered through the 8K static RAM. This minimizes the amount of bus time used by the Ethernet Communications Controller by eliminating the possibility of data overruns and subsequent repeated I/O operations. In addition, the buffer allows the Ethernet Communications Controller to have a bus priority below higher-priority, time-critical parts of the system.

DIAGNOSTICS

Diagnostic functions are resident on the SerDes board and can be invoked on demand by the program on the Processor board. These functions incude: transmitting packets with a bad CRC; receiving all packets regardless of address; reading data received in error; SerDes loop-back, this function allows data from static RAM to be transmitted and received simultaneously (the received data is verified but not written to the Static RAM). The CRC generation and checking can be used to verify transmit and receive data.

When the 10 meter interconnect cable between two Model 675s is used, point-to-point diagnostics are available to verify station-to-station communications, cable data sensitivity, CRC, packet length errors, and external carrier sense. In addition to these functions, a loop-back diagnostic is provided for use with transceivers. This function is similar to the SerDes loop-back diagnostic except the transceiver cable is also verified.

Figure 2 is a block diagram of the Ethernet Communications Controller.

Software

Three levels of program interface are provided to users of the Model 675 for Ethernet software prototyping and evaluation.

MULTIBUS MESSAGE EXCHANGE (MMX)-ISIS-II

MULTIBUS Message Exchange (MMX)-ISIS-II is a simple processor-to-processor protocol which permits ISIS-II user programs to communicate with the software residing on the Ethernet Communications Controller.

The model of use of MMX is as follows: The calling program will allocate a segment of memory in the 64 Kbyte segment accessible by ISIS-II, fill in application defined fields, and transmit via MMX to a socket on the Ethernet Communications Controller. The program then waits for the application level response to the command just given. There is an MMX module in ROM on the Ethernet Communications Controller, and another in the HOST RAM.

EXTERNAL DATA LINK (EDL)

The External Data Link (EDL) presents a subset of the Data Link Layer interface to users at the MMX-ISIS-II interface level. This allows a user to write Ethernet application programs to access the Ethernet Data Link on the 8085A-2 in ISIS-II systems. The External Data Link resides in the ROM on the Ethernet Communications Controller.

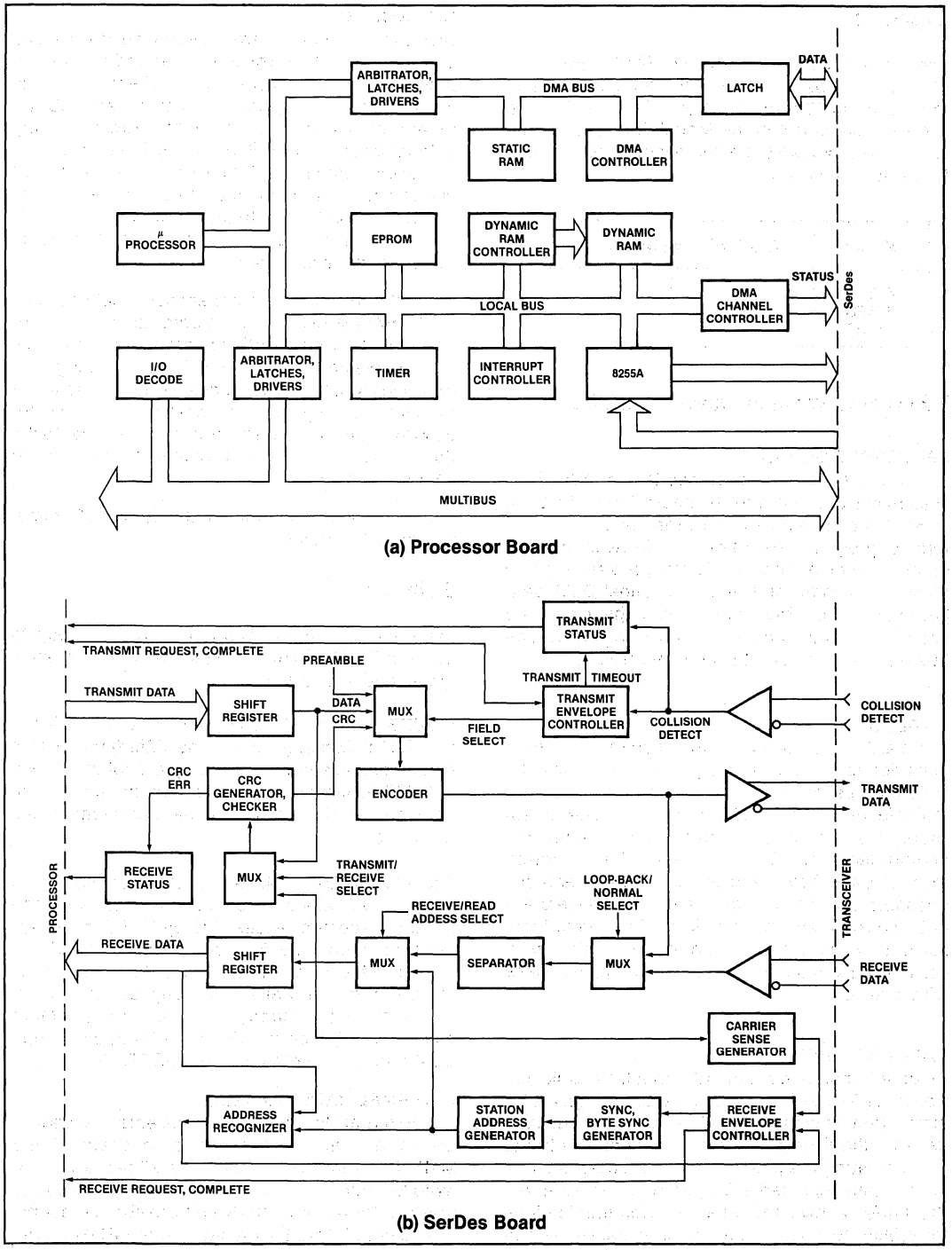


Figure 2. Block Diagram of the Ethernet Communications Controller

ETHERNET DATA LINK LIBRARY

The Ethernet Data Link Library is provided to simplify the External Data Link (EDL), MMX-ISIS-II interface. This library provides synchronous interface procedures for the EDL functions. These procedures allow the user to simply call a subroutine without being aware of the MMX-ISIS-II transaction which is made with the Ethernet Communications Controller. The Ethernet Data Link Library is provided on a diskette. It is designed to be linked with the user's software residing in the HOST RAM.

About the Ethernet

The Ethernet local area network provides a communication facility for high speed data exchange among digital devices located within a moderate sized geographic area. The Ethernet architecture defines the system as a series of independent layers.

The lowest layer, the Physical Link Layer, is concerned with the coaxial cable interface. It completely specifies the essential physical characteristics of the Ethernet, such as data encoding, timing, and voltage levels.

The Data Link defines a medium-independent link level communication facility, built on the medium-dependent physical channel provided by the Physical Layer. It supports the peer protocol statistical contention resolution (CSMA/CD), variable size frames, and link management functions.

The higher levels of the overall network architecture, which use the Data Link Layer, are collectively referred to as the Client Layer. The identity and function of this layer are user specific. The intent, however, is that the Ethernet Physical and Data Link Layers support the higher layers of the ISO model (Network Layer, Transport Layer, Session Layer, etc.).

The overall structure of the layered architecture is shown in Figure 3.

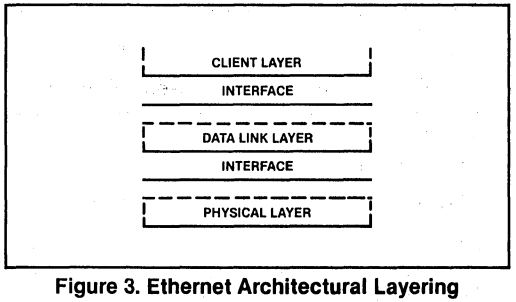


Figure 3. Ethernet Architectural Layering

DISK SUBSYSTEMS (OPTIONAL)

Dual Drive Floppy Disk Subsystem

The Model 675FD Double Density Diskette System provides direct access bulk storage, intelligent controller and two diskette drives. Each drive provides 1/2 Mbytes of storage with a data transfer of 500,000 bits/second. The controller provides an interface to the Model 675 system bus, as well as supporting up to four diskette drives to allow more than 2 Mbytes of on-line storage.

Hard Disk Subsystem

The Model 675HD Hard Disk System provides direct access bulk storage, intelligent controller and a disk drive containing one fixed platter and one removable cartridge. Each provides approximately 3.65 Mbytes of storage with a data transfer rate of 2.5 Mbits/second. The controller provides an interface to the system bus, as well as supporting up to 2 disk drives. The disk system records all data in Double Frequency (FM) on 2 surfaces per platter. Each platter can be write protected by a front panel switch.

SPECIFICATIONS

Host Processor Boards

INTEGRATED PROCESSOR CARD

—(IPC-85) 8085A-2 based, operating at 4 MHz
—64K RAM, 4K ROM (2K in monitor and 2K in boot/diagnostic)

RESIDENT PROCESSOR BOARD

—(RPB-86) 8086 based, operating at 5 MHz
—64K RAM, 16K ROM (application debugger)

BUS

—MULTIBUS system bus, maximum transfer rate of 5 MHz

DIRECT MEMORY ACCESS

—(DMA) standard capability on the MULTIBUS system bus; implemented for user selected DMA devices through optional DMA module
—Maximum transfer rate of 2 MHz

Ethernet Communications Controller

PROCESSOR
—8088 based, operating at 5 MHz
—16K dynamic RAM for program execution
—8K EPROM for program execution
—8K Static RAM for data buffer
—3 DMA channels for receive data
—1 DMA channel for transmit data

SerDes
—Serialization/Deserialization
—Framing
—CRC generation and checking
—Manchester encoding/decoding
—Destination address recognition

Integral Floppy Disk

Capacity—250 Kbytes (formatted)
Transfer Rate—160 Kbits/sec
Access Time—
- Track to Track: 10 ms max
- Average Random Positioning: 260 ns
- Rotational Speed: 360 rpm
- Average Rotational Latency: 83 ms
- Recording Mode: FM

Dual Floppy Disk Option

Capacity—
- Per Disk: 4.1 Mbits (formatted)
- Per Track: 53.2 Kbits (formatted)

Transfer Rate— 500 Kbits/sec
Access Time—
- Track to Track: 10 ms
- Head Setting Time: 10 ms
- Average Random Positioning: 260 ms
- Rotational Speed: 360 rpm
- Recording Mode: M^2 FM

Hard Disk Drive Option

Type—5440 top loading cartridge and one fixed platter
Tracks per Inch—200
Mechanical Sectors per Track—12
Recording Technique—double frequency (FM)
Tracks per Surface—400 bits/inch
Density—2,200 bits/inch
Bits per Track—62,500
Recording Surfaces per Platter—2
Capacity—
- Per Surface—15 Mbits
- Per Platter—29 Mbits
- Per Drive—59 Mbits
- Per Drive—7.3 Mbytes (formatted)

Transfer Rate—2.5 Mbits/sec
Access Time—
- Track to Track: 13 ms max
- Full Stroke: 100 ms
- Rotational Speed: 2,400 rpm

Physical Characteristics

Width—17.37 in. (44.12 cm)
Height—15.81 in. (40.16 cm)
Depth—19.13 in. (48.59 cm)
Weight—84.5 lb. (38.6 kg)

Expansion Chassis

Width—17.37 in. (44.12 cm)
Height—4.81 in. (17.22 cm)
Depth—19.13 in. (48.59 cm)
Weight—42 lb. (19 kg)

Keyboard

Width—17.37 in. (44.12 cm)
Height: 3.0 in. (22.86 cm)
Depth—9.0 in. (22.86 cm)
Weight—6 lb. (3 kg)

Dual Floppy Drive System (Option)

Width—16.88 in. (42.88 cm)
Height—12.08 in. (30.68 cm)
Depth—1.0 in. (48.26 cm)
Weight—64 lb. (29 kg)

Hard Disk Drive System (Option)

Width—18.5 in. (47.0 cm)
Height—34.0 in. (86.4 cm)
Depth—29.75 in. (75.6 cm)
Weight—202 lb. (92 kg)

ELECTRICAL CHARACTERISTICS

DC Power Supply for Mainframe

Volts Supplied	Amps Supplied	Typical System Requirements
+5 ± 5%	30.0	17.0
+12 ± 5%	2.5	1.1
−12 ± 5%	0.3	0.1
−10 ± 5%	1.0	0.08
+15 ± 5%*	1.5	1.5
+24 ± 5%*	1.7	1.7

*Not available on bus.

AC Requirements for Mainframe

110V, 60 Hz—5.9 Amp
220V, 50 Hz—3.0 Amp

DC Power Supply for Expansion Chassis

Volts Supplied	Amps Supplied	Typical System Requirements
+5 ± 5%	24	None
+12 ± 5%	2.0	None
−12 ± 5%	0.3	None
−10 ± 5%	1.0	None

AC Requirements for Expansion Chassis

50-60 Hz, 115/230V AC

ENVIRONMENTAL CHARACTERISTICS

System Operating Temperature—16°C to 32°C (61°F to 90°F)
Humidity—20% to 80%

DOCUMENTATION SUPPLIED

Ethernet Communications Controller Programmer's Reference Manual, 121769.

Ethernet Development System Upgrade Kit Installation and Checkout Manual, 121778.

Intellec Series III Microcomputer Development System Product Overview, 121575.

Intellec Series III Microcomputer Development System Console Operating Instructions, 121609.

Intellec Series III Microcomputer Development System Pocket Reference, 121610.

Intellec Series III Microcomputer Development System Programmer's Reference, 121618.

iAPX 88/86 Family Utilities User's Guide for 8086-Based Development Systems, 121616.

8086/8087/8088 Macro Assembly Language Reference Manual for 8086-Based Development Systems, 121627.

8086/8087/8088 Macro Assembly Language Pocket Reference, 9800/49.

8086/8087/8088 Macro Assembler Operating Instructions for 8086-Based Development Systems, 121628.

Intellec Series III Microcomputer Development System Installation and Checkout Manual, 121612.

Intellec Series III Microcomputer Development System Schematic Drawings, 121642.

ISIS-II CREDIT (CRT-Based Test Editor) User's Guide, 9800902.

ISIS-II CREDIT (CRT-Based Test Editor) Pocket Reference, 9800903.

The 8086 Family User's Manual, 9800722.

The 8086 Family User's Manual, Numeric Supplement, 121586.

For Series III Plus Hard Disk System Only:

Model 740 Hard Disk Subsystem Operation and Checkout, 9800943.

ORDERING INFORMATION

Part Number	Description
DS675A KIT	Intellec Model 675 Development System for Ethernet (110V/60 Hz)
DS675B KIT	Intellec Model 675 Development System for Ethernet (220V/50 Hz)
DS675AFD KIT	Intellec Model 675 Development System for Ethernet with Dual Density Flexible Disk System (110V/60 Hz)
DS675BFD KIT	Intellec Model 675 Development System for Ethernet with Dual Density Flexible Disk System (220V/60 Hz)
DS675AHD KIT	Intellec Model 675 Development System for Ethernet with Pedestal Mounted Hard Disk (110V/60 Hz)
DS675BHD KIT	Intellec Model 675 Development System for Ethernet with Pedestal Mounted Hard Disk (220V/50 Hz)

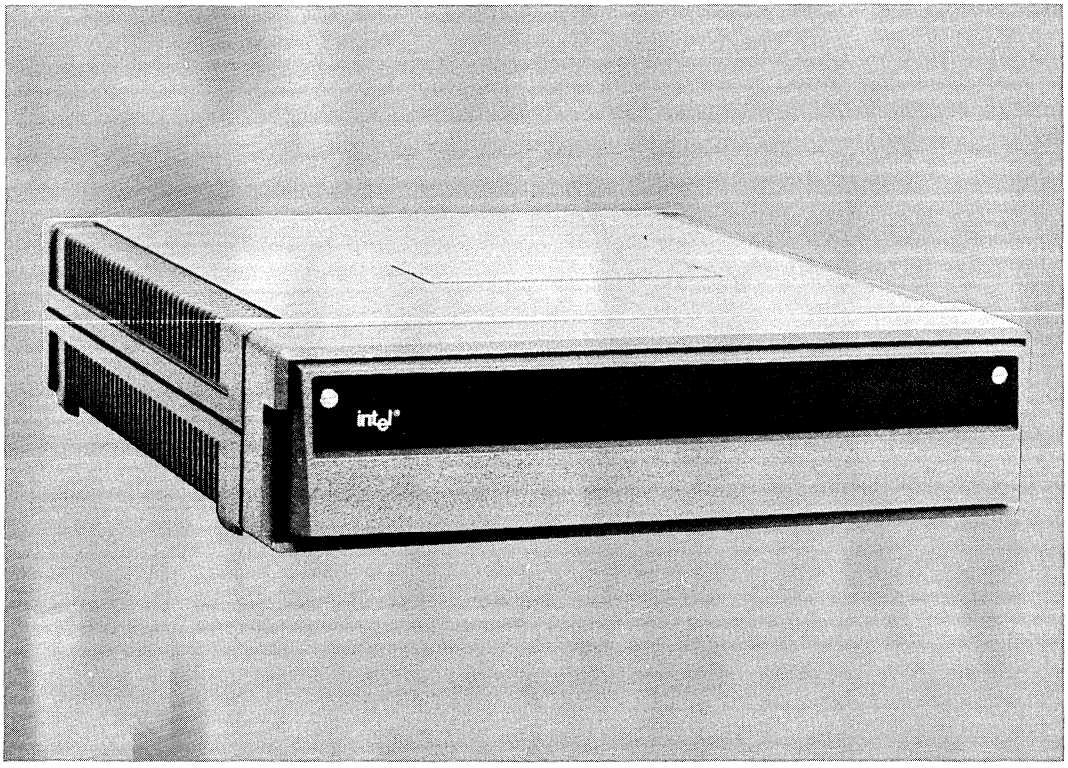
All models listed above require a Software License.

13 Microcomputer Development Systems Options

EXPANSION CHASSIS
INTELLEC® SERIES II
MICROCOMPUTER DEVELOPMENT SYSTEM

- **Four Expansion Slots for Intellec® Series II Systems**
- **Internal Power Supply**
- **Snug Fit Beneath All Intellec® Series II Units**
- **Cable Connectable to Main Intellec® Bus**
- **Standard Intellec MULTIBUS™ with Multi-Processor and DMA Capability**
- **Compatible with Standard Intellec/ iSBC™ Expansion Modules**

The Intellec Series II Expansion Chassis provides four expansion slots for use with Intellec Series II microcomputer development systems. With its own separate power supply, the expansion chassis may be fully loaded with any boards needed to expand a user's Intellec Series II system. With the addition of the expansion chassis, Intellec Series II Models 220 and 230 contain a total of ten slots, sufficient for any configuration Intellec Series II system. The Intellec Series II Expansion Chassis is a compact chassis with a four slot cardcage, power supply, fans, and cable assemblies. It is designed to fit under any Intellec Series II system, connect directly to the system bus through an opening in the top of the chassis, and provide additional slots for the system users. The power supply is linked directly to the main chassis power supply, allowing power to flow to both chassis when the main power is turned on.



SPECIFICATIONS

Physical Characteristics

Width — 17.37 in. (44.12 cm)
Height — 4.81 in. (17.22 cm)
Depth — 19.13 in. (48.59 cm)
Weight — 42 lb. (19 kg)

Electrical Characteristics

DC Power Supply

Volts Supplied	Amps Supplied	System Requirements
+ 5 ± 5%	24	None
+ 12 ± 5%	2.0	None
− 12 ± 5%	0.3	None
− 10 ± 5%	1.0	None

AC Requirements — 50-60 Hz, 115/230V AC

Environmental Characteristics

Operating Temperature — 0° to 35°C (95°F)

Equipment Supplied

Expansion chassis
Cables

Reference Manuals

9800550 — Intellec Series II Installation and Service Guide (SUPPLIED)

9800554 — Intellec Series II Schematic Drawings (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
MDS-201*	Intellec® Series II Expansion Chassis

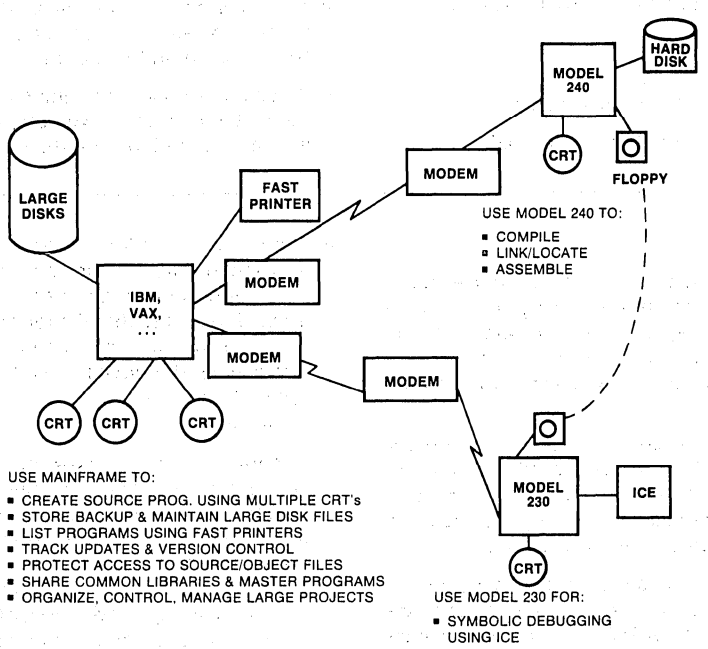
*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Science.

MAINFRAME LINK FOR DISTRIBUTED DEVELOPMENT

- **Integrates user mainframe resources with Intellec® Development Systems.**
- **Uses IBM 2780/3780 standard BISYNC protocol supported by a majority of mainframes and minicomputers.**
- **Protocol supports full error detection with automatic retry.**
- **Software runs under ISIS-II on any Intellec® Development System.**
- **Communicates with remote systems on dedicated or switched (dial-up) telephone lines.**
- **Package also includes tests and a connector for loop-back self-test capability.**

The Mainframe Link consists of software, modem cable to connect the development system to the modem and a loopback connector for diagnostic testing. The software runs under ISIS-II on Intellec Development Systems. It emulates the operation of an IBM 2780 or 3780 Remote Job Entry (RJE) terminal to (1) transmit ISIS-II files to a remote system or (2) receive files from a remote system using standard BISYNC 2780/3780 protocol. The remote system can be any mainframe or minicomputer which supports the IBM 2780 or 3780 communications interface standard. Files may contain ASCII or binary data so that either program source files (ASCII) or program object files (binary) may be transmitted.

The Mainframe Link allows the user to integrate in-house mainframe resources with Intellec Microcomputer Development resources. The mainframe can be used for storage, maintenance and management of program source and object files. The program source can be downloaded to a development system for compilation, assembly, linkage, and/or location. The linked modules can be transmitted and saved on the mainframe to be shared by all programmers. The linked program can then be downloaded to a development system for debugging using ICE emulation.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{1}m, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

121633-01A

FEATURES

- Runs under ISIS-II on any Intellec® Microcomputer Development System.
- Communicates with a remote system using IBM 2780/3780 standard BISYNC protocol, which is supported by a majority of minicomputers and mainframes, on dedicated or switched (dial-up) telephone lines.
- The modem cable supplied with the package can be used to connect the Intellec® Development System to the modem (or modem eliminator) using the standard RS232C port.
- Supports user selectable data transmission rates of up to 9600 baud.
- Package includes diagnostic tests used to verify the operation of the Intellec® Development System using the loop-back connector supplied and data transmission up to the modem using the analog loop-back feature.
- System can be configured to match the requirements of the installation, i.e., using modem eliminators for connections up to fifty (50) feet, or by using modems and telephone lines.
- Software can be configured from several configuration options such as:

  2780, 3780 or Intel Mode

  Transparent mode for binary data

  Non-transparent mode for ASCII data

  Automatic translation from ASCII to EBCDIC and vice versa

  Receive chaining for receiving multiple files
- Intel mode is used mainly for file transfers between two Intellec® Development Systems. The files are duplicated exactly.
- Console commands support all standard features including:

  SEND data in Transparent or Non-transparent mode, with or without translation to EBCDIC

  RECEIVE in Transparent or Non-transparent mode, with or without translation to EBCDIC.

  Support for an IBM RJE console (such as HASP)
- Special utility programs are provided. STRZ strips extra binary zero's from the end of object files. CONSOL assigns system console input to an ISIS-II disk file.
- Can process commands interactively from the console or sequentially from an ISIS-II file under the SUBMIT facility for semi-automatic batch operation.
- Error detection in line transmission and error recovery by automatic retransmission.
- A special command such as DIAGNOSE, allows logging of all data activity on the line, during transmission and reception.
- When not used for communicating with the mainframe, the Intellec® Development System is available as a complete, stand-alone system.

BENEFITS

- Allows the customer to use an in-house mainframe or minicomputer for program source-preparation, editing, back-up and maintenance using inexpensive CRT's and multi-terminal access. The common files may be shared and others protected.
- Many programmers can use and share the high-performance devices normally available on large computer systems, e.g., fast printers to reduce listing time, the large capacity disks with their fast access time to store large program files.
- The source files can be downloaded using the Mainframe Link to an Intellec Development System (e.g., Model 240 or 245) for compilation, linking and locating.
- The compiled and/or linked object files may be transmitted back to the remote for storage. Updates and version numbers and dates can be tracked to ensure that the latest version is always used and back-up files are available. Binary object files can be later downloaded to an Intellec Development System for debugging using an ICE emulator.
- In short, provides a powerful and flexible tool combining the best of both micro and mainframe worlds, i.e., powerful CPU with large disk capacity, file sharing, multi-terminal access, etc., from a mainframe or minicomputer with Intel's versatile and compatible software support systems (including PL/M, PASCAL, FORTRAN, Assembler, R & L) and sophisticated debugging tools such as ICE emulators.

SPECIFICATIONS

Operating Environment

Required Hardware:

Intellec® Microcomputer Development System
Model 800
Models 220, 225, 230, 235, 240 or 245

64KB of Memory

One Diskette Drive
Single or Double Density

System Console
Intel CRT or non-Intel CRT

Recommended Hardware for Compilation:

Hard Disk (Models 240, 245, or Model 740 Upgrade)

Additional Hardware Required for Model 800
iSBC-955™, iSBC-534™

Required Software:

ISIS-II Diskette Operating System
Single or Double Density

Documentation Package

Mainframe Link User's Guide (121565-001)

Shipping Media

Flexible Diskettes
Single and Double Density

Remote System Requirements

- IBM 2780/3780 BISYNC protocol as supported by a majority of mainframes and minicomputers including: all IBM-360/370 Systems, PDP-11/70, VAX-11/780, Data General ECLIPSE.
- Users should purchase this standard software package from the remote system vendor and any additional required hardware such as a synchronous communications interface.
- The operating system at the remote must be configured (SYSGEN'ed) with correct options such as line address, 2780 or 3780, . . .

Communication Equipment Requirements

The Intellec Development System may be connected to the remote system using any one of the following methods:

- For short distances (up to 50 feet), use a synchronous modem eliminator (e.g., SPECTRON ME-81FS-2).
- For distances up to four miles, use short haul synchronous modems and telephone lines.
- For distances greater than four miles, use synchronous modems and telephone lines. The following BELL modems or their equivalents are recommended:

BELL 201C	2400 bits/second (half duplex, switched line)
BELL 208A	4800 bits/second (full duplex, leased line)
BELL 208B	4800 bits/second (half duplex, switched line)
BELL 209A	9600 bits/second (full duplex, leased line)

- Modems at either end must be compatible.

ORDERING INFORMATION

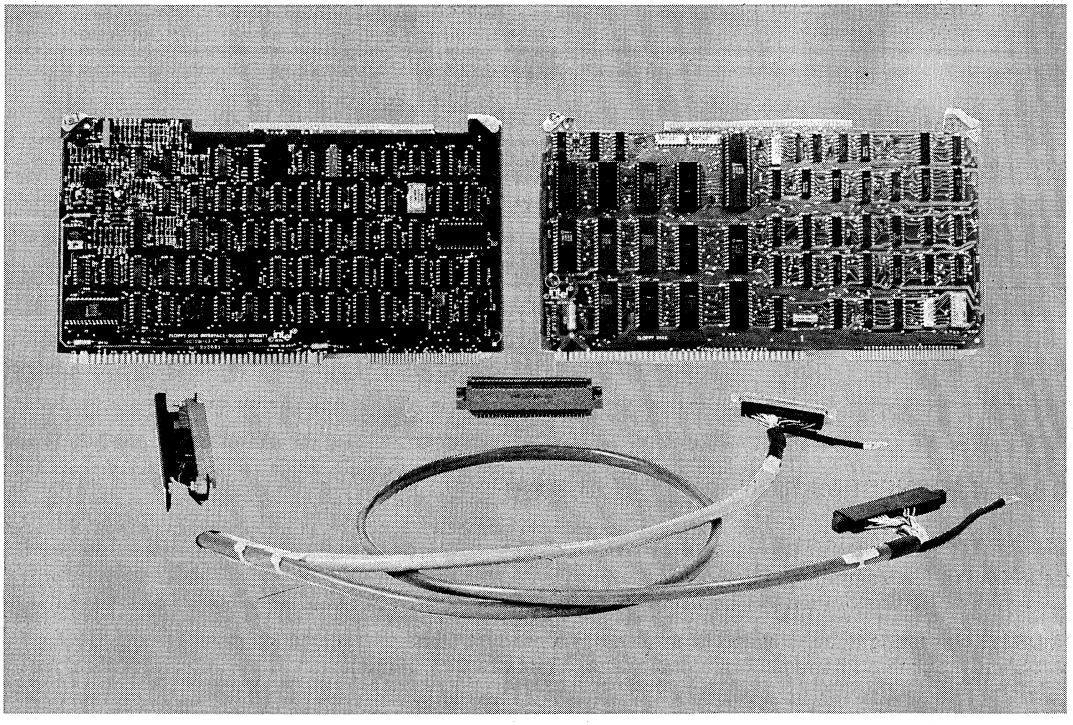
Part Number	Description
*MDS-384 Kit	Mainframe Link for Distributed Development

*MDS is an ordering code only and is not used as a product name or trademark.
MDS® is a registered trademark of Mohawk Data Sciences Corporation.

MODEL 503
DOUBLE DENSITY UPGRADE KIT FOR INTELLEC® MICROCOMPUTER DEVELOPMENT SYSTEM

- **Converts integral single density drive of Model 220 or Model 240 system to double density, doubling the data capacity from ¼ to ½ million bytes**
- **Data recorded on double density flexible disk is in soft sectored format which allows ½ million bytes data capacity with up to 200 files per flexible disk**
- **Associated software and hardware supports up to three double density drives, providing up to 1½ million bytes in one system**
- **Provides total data compability with other Intellec® Double Density Flexible Disk Systems**

The Double Density Upgrade Kit Model 503 provides an easy, cost effective method to convert the integral single density drive of Model 220 or Model 240 system to double density. In addition to doubling the data capacity, the upgrade kit maximizes the ease of data transportability between Intellec® Double Density Flexible Disk Systems.



i, ICE, iCS, Insite, Intel, Intellec, iSBC, Library Manager, MCS, Megachassis, Micromap, Multibus, Multimodule, PROMPT, Promware, RMX, UPI, μScope, and the combination of ICE, iCS, iSBC, MCS, or RMX and a numerical suffix are trademarks of Intel Corporation.
© Intel Corporation 1980

121536-001 Rev. A

SPECIFICATIONS

Equipment Supplied

— Floppy Disk Controller Channel Board

— Double Density Floppy Disk Interface Board

Dual Auxiliary Board Connector
Double Density Controller Cable
Double Density Integral Drive Cable
ISIS-II Double Density System Disk

Hardware

Double Density Specified Flexible Disk
One Recording Surface
Soft Sector Format M²FM
77 Tracks/Diskette
52 Sectors/Track
128 Bytes/Sector

Physical Characteristics

Mounting—Requires two slots of system cardcage

Height— 6.75 in. (17.15mm)
Width —12.00 in. (30.48mm)
Depth — 0.50 in. (1.27mm)

Electrical Characteristics

Channel Board
5V @ 3.75 (typ), 5A (max)

Interface Board
5V @ 1.5A (typ), 2.5A (max)
−10V @ 0.1A (typ), 0.2A (max)

Environmental Characteristics

Controller Boards

Temperature:
Operating: 0 to 55°C
Non-Operating: −55°C to 85°C

Humidity:
Operating: Up to 95% relative humidity without condensation

Non-Operating: All conditions without condensation of water or frost.

Reference Manuals

DOS Hardware Reference Manual	9800422
Reference Schematics	9800425
Installation Instructions	121505

ORDERING INFORMATION

Part Number	Description
MDS-503*	Integral drive single density to double density upgrade kit

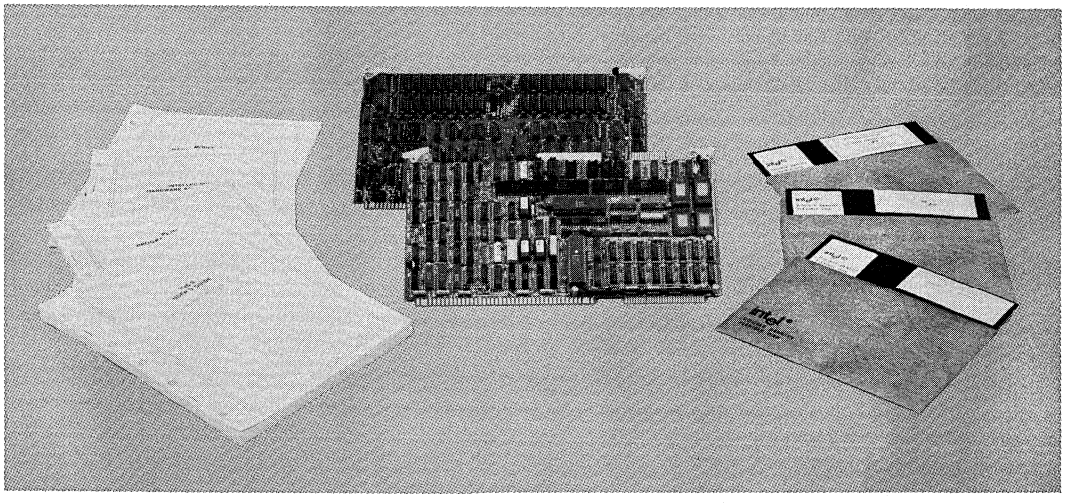
***"MDS"** is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.

MODEL 556
iAPX 86 RESIDENT PROCESSOR BOARD PACKAGE

- **Supports Intellec 432/100 Evaluation and Educational System**
- **Compatible with iSBC-090 Series 90 Memory System Upgrade: 512K Byte up to 1M Byte**
- **High Performance 8086-Based CPU Board for Increased Intellec® Development System Performance and iAPX 86/88 Development Environment**
- **Upgrades Intellec Series II/80, Series II/85, Model 800 Microcomputer Development Systems to the Functionality of Series III Systems**
- **96K Bytes of User Program RAM Memory Available for iAPX 86/88 User Programs**
- **Software Applications Debugger for iAPX 86/88 User Programs**
- **Supports Full Range of iAPX 86/88-resident, High-level Languages: PL/M-86/88, PASCAL-86/88, and FORTRAN-86/88**
- **Includes iAPX 86/88 Resident Relocating Macro Assembler, Linker, Locater and Librarian**
- **Dual-Processor Disk Operating System Software with CRT-based Editor**
- **In Conjunction with Model 677 Provides a Complete Ethernet* Communications Development System Environment**

The Model 556 is a performance enhancement package for Intellec Series II/80, Series II/85 and Model 800 Development Systems, specifically designed for iAPX 86,88 microprocessor development. The Model 556 includes two printed circuit boards (an iAPX 86-based CPU board and a 64K memory board), dual-processor disk operating system software, CRT-based editor, iAPX 86,88 Resident Relocating Macro Assembler, Linker, Locater and Librarian; software applications debugger for iAPX 86,88 user programs; and complete user documentation.

*Ethernet is a trademark of Xerox Corporation.



The following are trademarks of Intel Corporation and its affiliates and may be used only to identify Intel products: i, Int_el, INTEL, INTELLEC, MCS, ^{l}m, iCS, ICE, UPI, BXP, iSBC, iSBX, Insite, iRMX, System 2000, CREDIT, iRMX/80, MULTIBUS, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTIMODULE, and the combination of MCS, ICE, iSBC, iRMX or iCS and a numerical suffix. © INTEL CORPORATION, 1981.

121696-002
AFN-0172B

FUNCTIONAL DESCRIPTION

Hardware Components

Resident Processor Board (RPB-86)—The heart of the RPB-86 is an Intel 8086 16-bit HMOS microprocessor, running at 5.0 MHz. 64K bytes of RAM memory is provided on the board with transparent refresh from the Intel 8202 dynamic RAM controller. 16K bytes of ROM is on board, preprogrammed with an iAPX 86/88 applications debugger. The debugger provides features necessary to debug and control execution of application software for the iAPX 86/88 microprocesors. The RPB-86 occupies two card slots in an Intellec cardcage. The processors use interrupts for interprocessor communications.

RAM Memory Board—The memory board contains 64K bytes of read/write RAM memory and interfaces directly to the Intellec system bus. Refresh hardware is provided onboard for all the dynamic memory elements. Data buffering occurs for all data written to or read from the 64K memory array.

SYSTEM FEATURES

The Model 556 offers many key advantages for iAPX 86/88 applications and Intellec Development Systems: enhanced system performance through a dual host CPU environment, a full spectrum of iAPX 86/88-resident high-level languages, expanded user program space for iAPX 86/88 programs, and a powerful high-level software applications debugger for iAPX 86/88 microprocessor software.

Dual Host CPU—The addition of a 16-bit 8086 to the existing 8-bit host CPU increases iAPX 86/88 compilation speeds and provides for iAPX 86/88 code execution. When the 8086 is executing a program, the 8-bit CPU offloads all I/O activity and operates as an intelligent I/O controller to double buffer data to and from the 8086. The 8086 also provides an execution vehicle for 8086 and 8088 object code. An added benefit of two host microprocessors is that 8-bit translations and applications are handled by the 8-bit CPU, and 16-bit translations and applications are handled by the 8086. This feature provides complete compatibility for current systems and means that software running on current Intellec Development Systems will run on the new system.

High-Level Languages for iAPX 86/88—The Model 556 allows the current Intellec system user to take advantage of a breadth of new resident iAPX 86/88 high-level languages: PL/M 86/88, PASCAL 86/88, and FORTRAN 86/88. The iAPX 86/88 Resident Macro Assembler and these high-level language compilers execute on the 8086 host CPU, thereby increasing system performance.

Expanded Program Memory—By adding a Model 556 to an existing Intellec Development System, 96K bytes of user program RAM memory are made available for iAPX 86/88 programs. System memory is expandable by adding additional RAM memory modules. This, combined with the two host CPU system architecture, dramatically increases the processing power of the system.

Software Applications Debugger—The RPB-86 contains the applications debugger which allows iAPX 86/88 programs to be developed, tested, and debugged within the Intellec system. The debugger provides a subset of In-Circuit Emulator commands such as symbolic debugging, control structures and compound commands specifically oriented toward software debug needs.

SPECIFICATIONS

Resident Processor Board (RPB-86):
8086 based, operating at 5.0 MHz
2 RAM — 64K bytes on the CPU board
ROM — 16K bytes (applications debugger)
Bus — MULTIBUS architecture; 5 MHz maximum transfer rate

Electrical Characteristics

DC POWER SUPPLY

Voltage Requirements	Current Requirements (Amperes Max.)
+5±5% Volts	8.6 A
+12±5% Volts	1.0 A
−12±5% Volts	0.05 A

Environmental Characteristics

Operating Temperature: 0° to 35°C (32°F to 95°F)
Relative Humidity: To 90% without condensation

Equipment Supplied

iAPX 86 Resident Processor Board (RPB-86)
64K Byte RAM Memory Board
iAPX 86/88 Applications Debugger
Self-test Diagnostics
iAPX 86/88 Resident Macro Assembler and Utilities
Dual Processor Disk Operating System Software
CREDIT™ CRT-based text editor

Documentation Supplied

A Guide to Intellec Series III Microcomputer Development Systems, 121632-001

Intellec Series III Microcomputer Development System Product Overview, 121575

Intellec Series III Microcomputer Development System Console Operating Instructions, 121609

Intellec Series III Microcomputer Development System Pocket Reference, 121610

Intellec Series III Microcomputer Development System Programmer's Reference, 121618

iAPX 86/88 Family Utilities User's Guide for 8086-Based Development Systems, 121616

8086/8087/8088 Macro Assembly Language Reference Manual for 8086-Based Development Systems, 121627

8086/8087/8088 Macro Assembly Language Pocket Reference, 9800749

8086/8087/8088 Macro Assembler Operating Instructions for 8086-Based Development Systems, 121628

Intellec Series III Microcomputer Development System Installation and Checkout Manual, 121612

Intellec Series III Microcomputer Development System Schematic Drawings, 121642

ISIS-II CREDIT (CRT-Based Text Editor) User's Guide, 9800902

ISIS-II CREDIT (CRT-Based Text Editor) Pocket Reference, 9800903

The 8086 Family User's Manual, 9800722

The 8086 Family User's Manual, Numerics Supplement, 121586

Additional manuals may be ordered from any Intel sales representative or distributor office, or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
MDS-556*	Model 556 performance upgrade package for Intellec Series II/85 and Model 800 Microcomputer Development Systems (110V/60 Hz or 220V/50 Hz). Specifically designed for iAPX 86/88 microprocessor development. Upgrades all Intellec Series II/80 and Series II/85 models to the functionality of an Intellec Series III Development System.
DS556I Kit	Performance package for Intellec Series II/80 Microcomputer Development Systems. Specifically designed for iAPX 86/88 microprocessor development. The 556I package consists of the Model 556 software and hardware performance package, and the integrated 8085 processor board (IPC-85). This upgrade package is for Intellec Series II/80 Development Systems (110V/60 Hz or 220V/50 Hz) and upgrades all Intellec Series II/80 Models to the full performance and functionality of an Intellec Series III Development System.
DS556432 Kit	Model 556 combined with the Intellec 432/100 Evaluation and Educational System. This package provides all necessary hardware, software and literature to support iAPX 432 evaulation.
DS556I432 Kit	Upgrades Intellec Series II/80 users with all necessary hardware, software and literature to evaluate the iAPX 432. This package includes the DS556I and the Intellec 432/100 Evaluation and Educational System.

* MDS is an ordering code only, and is not used as a product name or trademark. MDS is a registered trademark of Mohawk Data Sciences Corp.

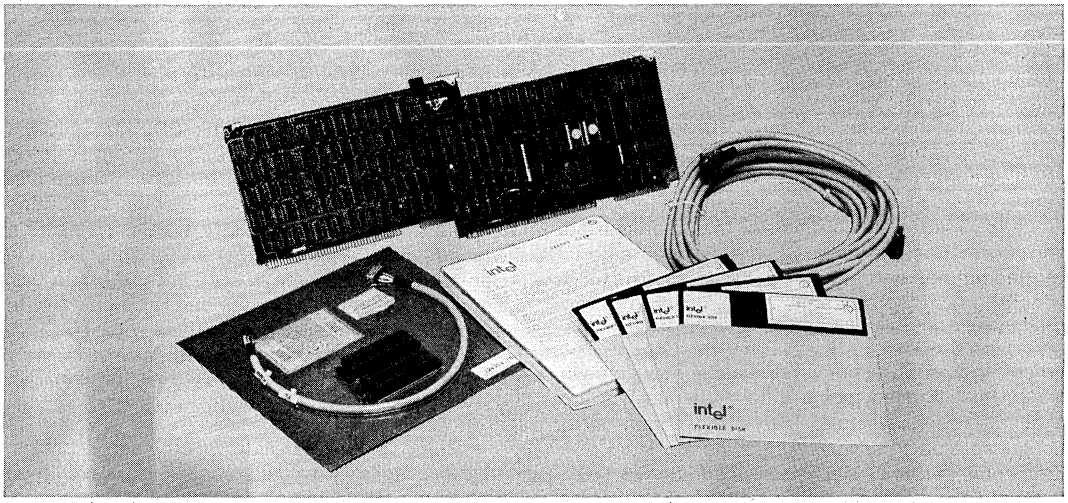
MODEL 677 DS/E ETHERNET* UPGRADE KIT FOR INTELLEC® DEVELOPMENT SYSTEMS

- **Complies with the Intel®, DEC, and Xerox Tricompany Ethernet Specification**
- **Includes an Ethernet Data Link Layer Software Library to allow user programs to access the Ethernet Data Link from the 8085 in Series II and Series III**
- **Supports the Ethernet Data Link and Physical Link Control over the MULTIBUS**
- **Includes a special 10 meter Interconnect Cable for connecting two DS/E units for Ethernet software prototyping**
- **Upgrades Intellec® Series II/85 and Series III Development Systems**

The DS/E Ethernet Upgrade Kit provides the Series II/85 or Series III user with the additional tools necessary to develop and test communication software and applications that will use Ethernet as a communication subsystem. It combines the power of the Intellec Microcomputer Development System with its dual board Ethernet Communications Controller for microprocessor development. This combination allows the user to develop either 8- or 16-bit Ethernet-based applications.

The Ethernet Communications Controller incorporates the Ethernet Data Link and Physical Link Control to meet the Ethernet specification for 10 Mbit per second data transmission rate over coaxial cable. The controller consists of two MULTIBUS compatible boards, the Processor board and the Serialization/Deserialization (SerDes) board. The Processor board provides the function of packet buffering, processing, and transferring of the processed packets to system memory. The SerDes board performs the serialization/deserialization, framing, CRC generation and checking, Manchester encoding/decoding, and destination address recognition. A 10 meter interconnect cable is included to permit two DS/E units to be connected in a functioning Ethernet environment without the need for Ethernet transceivers and coaxial cable.

*Ethernet is a trademark of Xerox Corporation.



Intel Corporation Assumes No Responsibilty for the Use of Any Circuitry Other Than Circuitry Embodied in an Intel Product. No Other Circuit Patent Licenses are Implied.
© INTEL CORPORATION, 1981.

ORDER NUMBER: 210279-001

Ethernet Communications Controller

PROCESSOR BOARD

The processor board interfaces to the MULTIBUS system bus and contains the Intel 5 MHz 8088 CPU, 16 Kbytes dynamic RAM for Ethernet and host interface program execution, 8K EPROM pre-programmed for the Data Link Control, and an 8K static RAM for transmit and receive channel DMA data buffering. The Processor board provides the necessary commands and control to the SerDes board and receives status and data from it.

SerDes BOARD

The SerDes board meets the required electrical specification to the transceiver and provides the Data Link Layer of the Ethernet architecture. The major functions of the SerDes board include serialization/deserialization, framing, Manchester encoding/decoding, transmit data flow control, receive data flow control, destination address decoding for received message, CRC generation and checking, and diagnostic for CRC error, loop-back, transmit timeout, and if used with transceivers, CSMA/CD (carrier-sense multiple-access with collision-detection).

DATA BUFFERING

All data transfers from the Ethernet Data Link control of the SerDes board are buffered through the 8K static RAM. This minimizes the amount of bus time used by the Ethernet Communications Controller by eliminating the possibility of data overruns and subsequent repeated I/O operations. In addition, the buffer allows the Ethernet Communications Controller to have a bus priority below higher-priority, time-critical parts of the system.

DIAGNOSTICS

Diagnostic functions are resident on the SerDes board and can be invoked on demand by the program on the Processor board. These functions incude: transmitting packets with a bad CRC; receiving all packets regardless of address; reading data received in error; SerDes loop-back, this function allows data from static RAM to be transmitted and received simultaneously (the received data is verified but not written to the Static RAM). The CRC generation and checking can be used to verify transmit and receive data.

When the 10 meter interconnect cable between two DS/E Units is used, point-to-point diagnostics are available to verify station-to-station communications, cable data sensitivity, CRC, packet length errors, and external carrier sense. In addition to these functions, a loop-back diagnostic is provided for use with transceivers. This function is similar to the SerDes loop-back diagnostic except the transceiver cable is also verified.

Figure 1 is a block diagram of the Ethernet Communications Controller.

Software

Three levels of program interface are provided to users of the Model 675 for Ethernet software prototyping and evaluation.

MULTIBUS MESSAGE EXCHANGE (MMX)-ISIS-II

MULTIBUS Message Exchange (MMX)-ISIS-II is a simple processor-to-processor protocol which permits ISIS-II user programs to communicate with the software residing on the Ethernet Communications Controller.

The model of use of MMX is as follows: The calling program will allocate a segment of memory in the 64 Kbyte segment accessible by ISIS-II, fill in application defined fields, and transmit via MMX to a socket on the Ethernet Communications Controller. The program then waits for the application level response to the command just given. There is an MMX module in ROM on the Ethernet Communications Controller, and another in the HOST RAM.

EXTERNAL DATA LINK (EDL)

The External Data Link (EDL) presents a subset of the Data Link Layer interface to users at the MMX-ISIS-II interface level. This allows a user to write Ethernet application programs to access the Ethernet Data Link on the 8085A-2 in ISIS-II systems. The External Data Link resides in the ROM on the Ethernet Communications Controller.

ETHERNET DATA LINK LIBRARY

The Ethernet Data Link Library is provided to simplify the External Data Link (EDL), MMX-ISIS-II interface. This library provides synchronous interface procedures for the EDL functions. These procedures allow the user to simply call a subroutine without being aware of the MMX-ISIS-II transaction which is made with the Ethernet Communications Controller. The Ethernet Data Link Library is provided on a diskette. It is designed to be linked with the user's software residing in the HOST RAM.

About the Ethernet

The Ethernet local area network provides a communication facility for high speed data exchange among digital devices located within a moderate

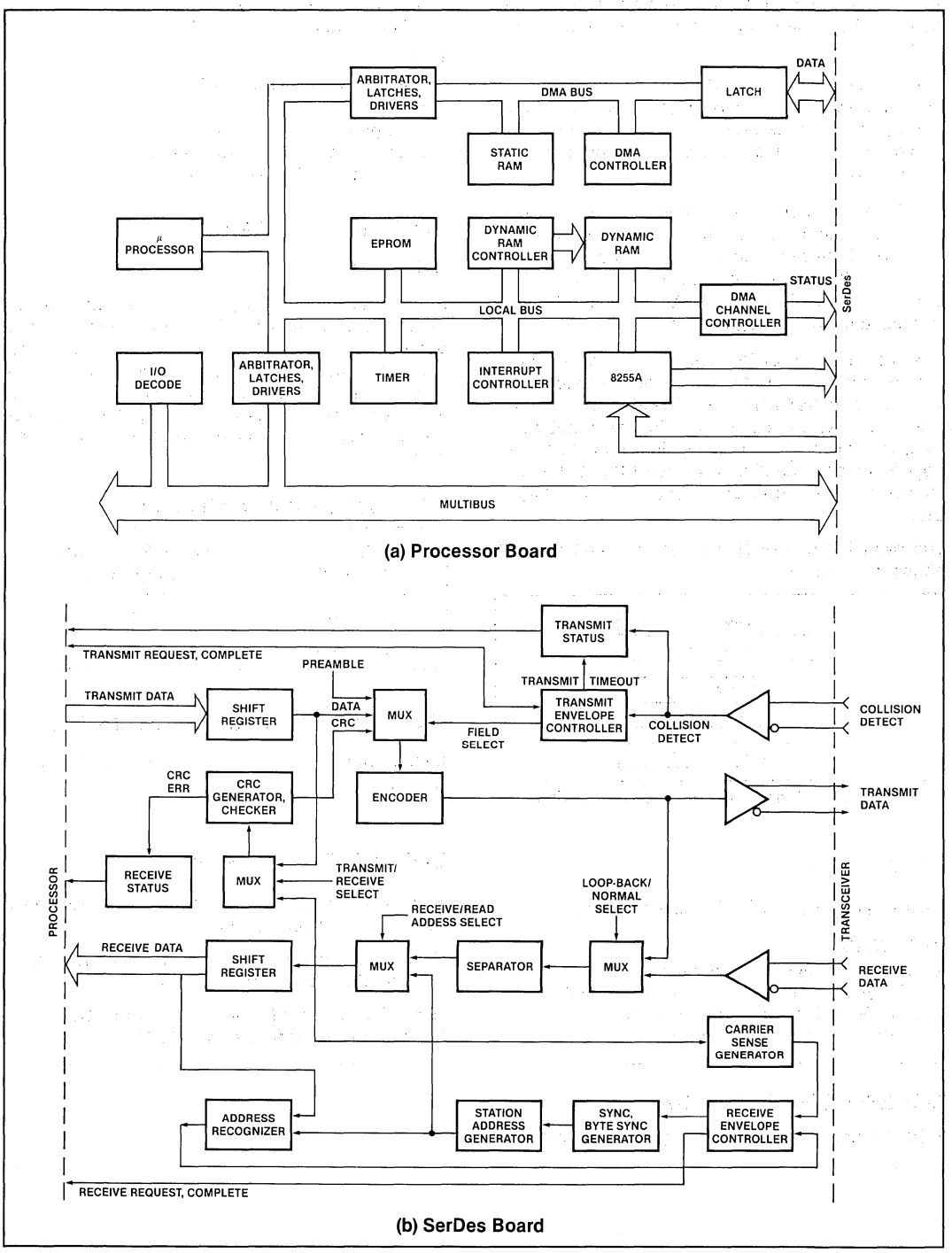


Figure 1. Block Diagram of the Ethernet Communications Controller

sized geographic area. The Ethernet architecture defines the system as a series of independent layers.

The lowest layer, the Physical Link Layer, is concerned with the coaxial cable interface. It completely specifies the essential physical characteristics of the Ethernet, such as data encoding, timing, and voltage levels.

The Data Link defines a medium-independent link level communication facility, built on the medium-dependent physical channel provided by the Physical Layer. It supports the peer protocol statistical contention resolution (CSMA/CD), variable size frames, and link management functions.

The higher levels of the overall network architecture, which use the Data Link Layer, are collectively referred to as the Client Layer. The identity and function of this layer are user specific. The intent, however, is that the Ethernet Physical and Data Link Layers support the higher layers of the ISO model (Network Layer, Transport Layer, Session Layer, etc.).

The overall structure of the layered architecture is shown in Figure 2.

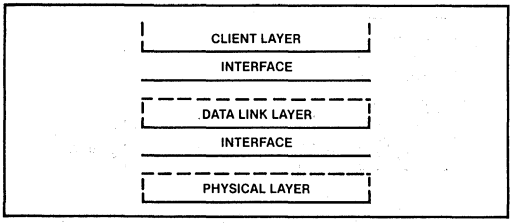


Figure 2. Ethernet Architectural Layering

SPECIFICATIONS

Ethernet Communications Controller

PROCESSOR
—8088 based, operating at 5 MHz
—16K dynamic RAM for program execution
—8K EPROM for program execution
—8K Static RAM for data buffer
—3 DMA channels for receive data
—1 DMA channel for transmit data

SerDes
—Serialization/Deserialization
—Framing
—CRC generation and checking
—Manchester encoding/decoding
—Destination address recognition

ELECTRICAL CHARACTERISTICS

DS677 Kit

Ethernet Development Upgrade Kit includes the Ethernet Communication Controller board set (2 boards), top assembly hardware, 10m interconnect cable, data link interface library, diagnostic test programs, manuals and documentation. Will give Ethernet development capability to any Series II/85 or Series III development system. (Shipping weight 10.5 lbs.)

DC Power Supply for Mainframe

Volts Supplied	Current Requirements Amps Max.
+5±5%	9.0
+12±5%	0.5

ENVIRONMENTAL CHARACTERISTICS

System Operating Temperature—16°C to 32°C (61°F to 90°F)
Humidity—20% to 80%

DOCUMENTATION SUPPLIED

Ethernet Communications Controller Programmer's Reference Manual, 121769.

Ethernet Development System Upgrade Kit Installation and Checkout Manual, 121778.

iSBC-550 Ethernet Communications Controller Hardware Reference Manual 121746.

ORDERING INFORMATION

DS677 Ethernet Upgrade Kit requires a Software License.

ISIS-II SOFTWARE TOOLBOX

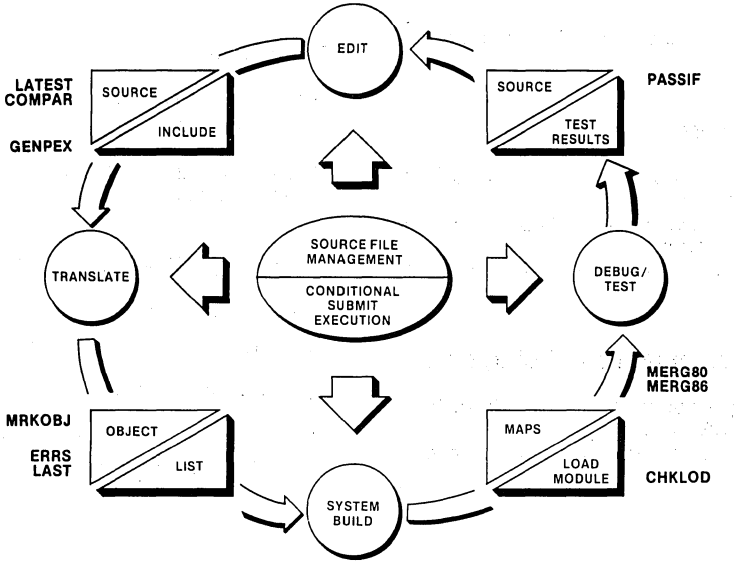
- **Significantly Improves Programmer Productivity**
- **Collection of Utilities that Speed Up Software Design**
- **Enhances Capabilities of ISIS-II Operating System**
- **Most Utilities will Operate on NDS-I Workstations, and Remote Hard Disks**
- **Provides Source File Management, Showing Source Changes, and Performing Version Control**
- **Provides Conditional Control and "Structured Programming" to Submit Files**
- **Runs on Model 800, Series II, and Series III Intellec® Development Systems**

The ISIS-II Software Toolbox is a collection of system utilities that perform a variety of "productivity-oriented" functions. There are two major subsets of Toolbox tools, in addition to numerous ad hoc utilities. These subsets provide Conditional Submit File Control and Source File Management.

The Conditional Submit File Control tools provide "structured programming" at the ISIS-II command level. Jumps, Calls, Returns, etc. are supported, as well as conditional command execution, based on assertions such as file existence, program errors, file matching, and string matching.

The Source Management Tools support version number tracking, and allow users to identify which versions of each source module were used to create a load module. There is also a tool which compares source files and reports all differences.

The tools outside of the two major subsets assist the programmer in some very specific development and debugging tasks. One tool manages all PUBLIC/EXTERNAL declarations in a system. Another merges the locate maps into a program listing, giving absolute symbolic debugging information. There's a directory sorter, a file compactor, and a tool to display just the last block of a file.



MANY TOOLS IN THE TOOLBOX ENHANCE SPECIFIC PHASES OF THE DEVELOPMENT CYCLE. OTHERS IMPROVE PRODUCTIVITY IN ALL PHASES.

The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, CREDIT, i, ICE, iCS, im, Insite, Intel, intel, Intelevision, Intellec, iRMX, iSBC, iSBX, Library Manager, MCS, Megachassis, Micromap, Multibus, Multimodule, PROMPT, Promware, RMX/80, System 2000, UPI, μScope, and the combination of ICE, iCS, iRMX, iSBC, iSBX, MCS, or RMX and a numerical suffix.
© Intel Corporation 1981

FUNCTIONAL DESCRIPTION

Submit File Execution Control

IF/ELSE/ENDIF—conditional submit file execution based on file existence, program errors, pattern matching, plus several other conditions
GOTO—causes submit execution to resume at a specified label
RETURN—causes execution to return to the "submitter" (calling file)
EXIT—halts submit file execution
LOOP—forces execution to resume at the beginning of the submit file
RESCAN—allows submit execution to begin anywhere in file
NOTE—allows "progress report" notes to be placed in submit files
WAIT—displays a message, and waits for user input to continue or abort
STOPIF—halts submit file execution if specified listing contains errors

Source Management

XLATE2—submit-like tool with intelligent parameter substitution (for version control)
MRKOBJ—"marks" object modules with source version information
CHKLOD—lists source version data put in load modules by MRKOBJ
CLEAN—deletes all old versions off a specified disk
LATEST—displays latest version numbers of specified files

Operating System Functions

CONSOL—reassigns console input and console output as directed
DSORT*—alphabetically sorts floppy disk and hard disk directories
RELAB*—changes disk name to any other specified name

Program Development and Debugging

ERRS—fast display of program errors in PL/M 80, PL/M 86, and ASM 86 listings
MERG80—merges debug data from locate maps into PL/M 80 listings
MERG86—merges debug data from symbol maps into PL/M 86 and Pascal 86 listings
GENPEX—produces include file for PL/M external declarations (source level)
PASSIF—general purpose assertion checking, testing, and reporting tool

Text Processing

COMPAR—performs line-oriented text file comparison (shows source changes)
UPPER—changes all letters in an ASCII text file to uppercase
LOWER—changes all letters in an ASCII text file to lowercase
LAST—displays the last 512 bytes of a file
SORT—sophisticated line-oriented text file sorting tool

Disk Backup and File Processing

DCOPY—fast track-by-track diskette copying
HDBACK*—sophisticated hard disk to floppy disk backup program
PACK—compacts text files by removing strings of blanks
UNPACK—reconstitutes "packed" files

Disk Recovery

GANEF*—interactively reads and writes floppy or hard disk data blocks

Program Identification

WHICH—displays version number of Software Toolbox Programs

*These programs will not operate on the NDS-I remote hard disks.

ORDERING INFORMATION

Product Code	Description
MDS-363†	ISIS-II SOFTWARE TOOLBOX

Requires software license.

†MDS is an ordering code only and is not used as a product name or trademark. MDS is a registered trademark of Mohawk Data Science.

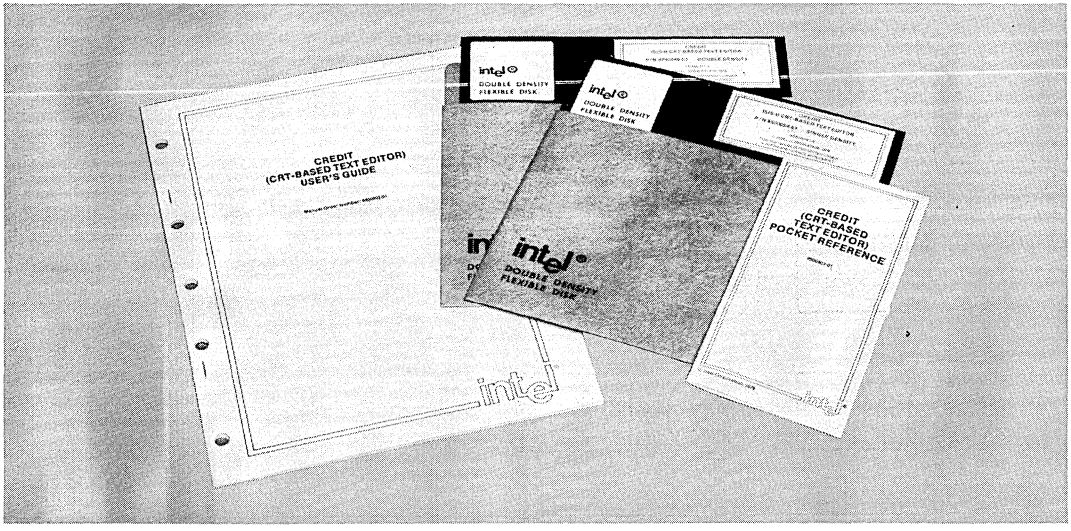
CREDIT™
CRT-BASED TEXT EDITOR
MICROCOMPUTER DEVELOPMENT SYSTEMS

- **Provides Interactive Editing of ASCII Text Files**
- **CRT Screen Display with Cursor-Based Editing Using Single Character Commands for Insertion, Deletion, Page Forward and Backward**
- **Command Line Editing with String Search, Deletion, Insertion and Move**
- **Displays Full Page of Text**
- **Dynamic Macro Command Definition**
- **Operates Under the ISIS-II Operating System on Intellec® and Intellec® Series II Microcomputer Development Systems**

CREDIT is a CRT-based text editor that aids in the creation and editing of ASCII text files on Intellec Microcomputer Development Systems. Once the text has been edited to the programmer's satisfaction, it can be directed to the appropriate language processor for compilation, assembly or interpretation. CREDIT features are easy to use and simplify the change or rearrangement of text files. CREDIT runs under ISIS-II on any Intellec or Intellec Series II Microcomputer Development System with an Intel supplied CRT, disk drive(s) and 64K bytes of memory. Alternatively, it may be configured to run with non-Intel CRTs supporting cursor controls.

There are two editing modes in CREDIT: a screen mode and a command line mode. The screen mode makes full use of the display characteristics of the CRT. The cursor position is visible on the screen and can be positioned by use of the cursor control keys. Display text can be corrected in two ways—either by simply retyping the text, or by using the single-stroke control keys. Specifically, the single-character control keys are used for change, deletion, insertion and paging forward and page backward.

In addition to screen editing, there is command line editing, which includes commands for more powerful and complex editing tasks. Some examples of functions available in the command line mode are search, block move and copy, macro definition and manipulation of external files. These easily used, high-level commands facilitate complex editing and speed microcomputer development.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, CREDIT, Intellec, Multibus, i, iSBC, Multimodule, ICE, iSBX, PROMPT, iCS, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Micromap, μScope and the combination of ICE, iCS, iSBC, iSBX, MCS, or RMX and a numerical suffix.
© Intel Corporation 1980

9800926-02 Rev. B

CREDIT™ EDITOR FEATURES

- Two editing modes: cursor-driven screen editing and command line context editing

CRT Editing Includes:

- Displays full page of text
- Single control key commands for insertion, deletion, page forward and backward
- Type-over correction and replacement
- Immediate feedback of the results of each operation
- The current state of the text is always represented on the display

Command Line Editing Includes:

- String search and substitute
- String delete, change or insert
- Block move
- Block copy
- User-defined macros
- External file handling
- Change CREDIT features with ALTER command
- Conditional iteration
- User-defined tab settings
- Symbolic tag positions
- Automatic disk full warning
- Runs under ISIS-II SUBMIT facility
- Option to exit at any time with original file intact
- HELP command

BENEFITS OF CREDIT™ EDITOR

- Speeds source program creation and editing—lowers the cost of these functions
- Easy to learn and use—
  —source text is clearly displayed
  —single command keys used for CRT editing
  —HELP command is available for easy reference when needed
- Complements existing software - source text used for PL/M, PASCAL, FORTRAN, BASIC, and Assemblers
- Aids in the management of source file libraries
- Offers full use of Intel supplied CRT cursor functions

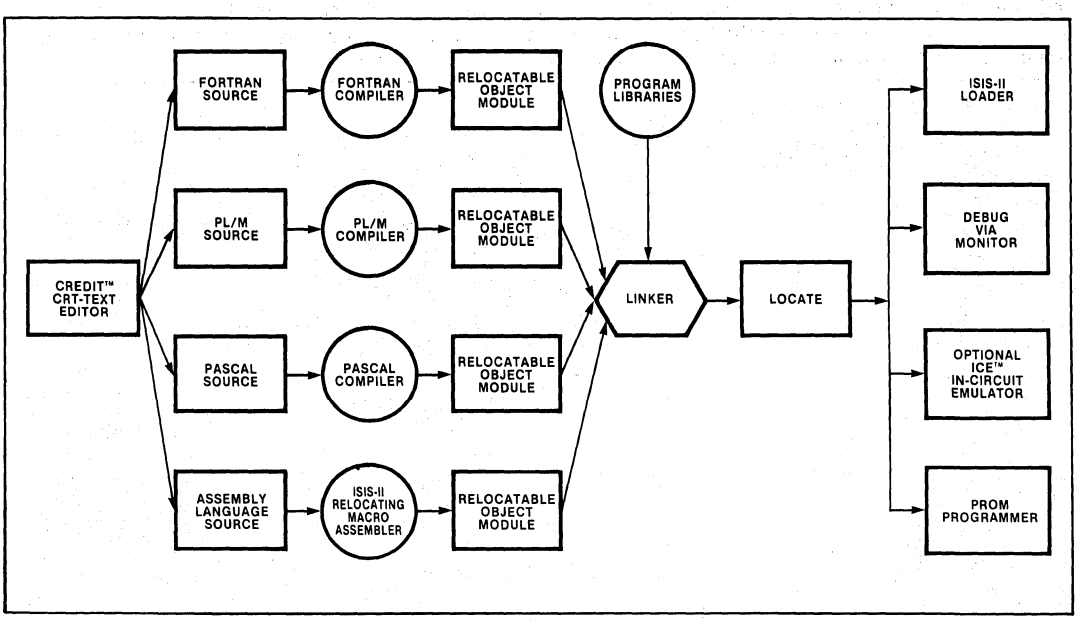
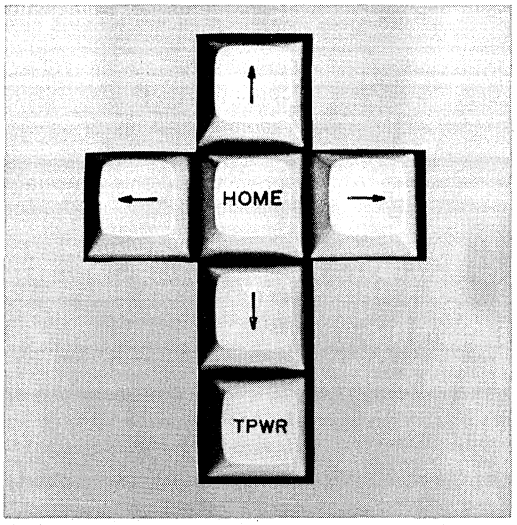


Figure 1. Microcomputer Program Development

SCREEN MODE COMMANDS

MOVE CURSOR:	Use the directional arrow keys on the keyboard.
REPLACE:	Type over existing text with replacement new text.
INSERT:	Insert one character.
	Insert more than one character.
DELETE:	Delete one character.
	Set boundaries and delete all text between them.
PAGE:	Next Page: Get next screenful of text.
	Previous Page: Get previous screenful.
	View Page: Rewrite current page with possible reframing.



COMMAND MODE COMMANDS

HELP:	Display summary of commands.
PRINT:	Print n lines or up to tag.
JUMP:	Move cursor position n characters or to tag.
	Move cursor position n lines forward or backward.
MOVE:	Transfer Copy block of text from tag1, for n lines or through tag2, to cursor position.
	Transfer move: like Transfer Copy but the old copy is deleted.
TAGS:	Set tag n, n = 0-9. Tag n is referenced as Tn.
EXIT:	Normal exit.
	Exit Quit: Abandon any changes to edit file.
INSERT:	Insert before CP all text between delimiters.
DELETE:	Delete n characters, or characters up to tag.
	Delete n lines forward or backward.
FIND:	Search for text; move pointer if found.
SUBST:	newtext replaces oldtext if oldtext is found. (Optional query to user before replacement.)

ADVANCED EDITING COMMANDS

MACROS:	Define a macro.
	Delete a macro, or all macros if name=*.
	Expand and execute macro contents, command mode.
	Expand and execute macro contents, screen mode.
	Print names and definitions of all macros.
FILES:	Open file "filename" for Reading or Writing.
	Close the current external Read (Write) file.
	Go to beginning of current Read file.
	Read and insert n lines from the Read file.
	Write n lines to the external Write file.

GET:	Get contents of file into command line.
QUERY:	Query User: set Query Flag accordingly.
	Do command only if Query Flag is True.
	Do command only if Query Flag is False.
YES:	Do command only if Yes (Search) Flag is False.
	Do command only if Yes (Search) Flag is False.
LOOP:	Exit current iteration loop.
ALTER:	Configure the command input keys to work with alternative CRTs.
USER:	Copy text to the console.
HELP:	Display summary of commands.

SPECIFICATIONS

Operating Environment

Required Hardware

Intellec® Microcomputer Development System
—Model 800 or Series II with 64k bytes of RAM memory
—Series III

Diskette Drive(s)
—Single or double density

System Console
—Intel supplied CRT or alternative CRT supporting cursor controls

Optional Hardware

Line Printer
Additional diskette drive(s)

Required Software

ISIS-II Diskette Operating System
—Single or double density

Documentation Package

CREDIT® (CRT-Based Text Editor) User's Guide (9800902)
CREDIT® Pocket Reference (9800903)

Shipping Media

Flexible Diskettes
—Single or double density

ORDERING INFORMATION

Part Number	Description
MDS-360*	ISIS-II CREDIT CRT-Based Text Editor

Requires Software License

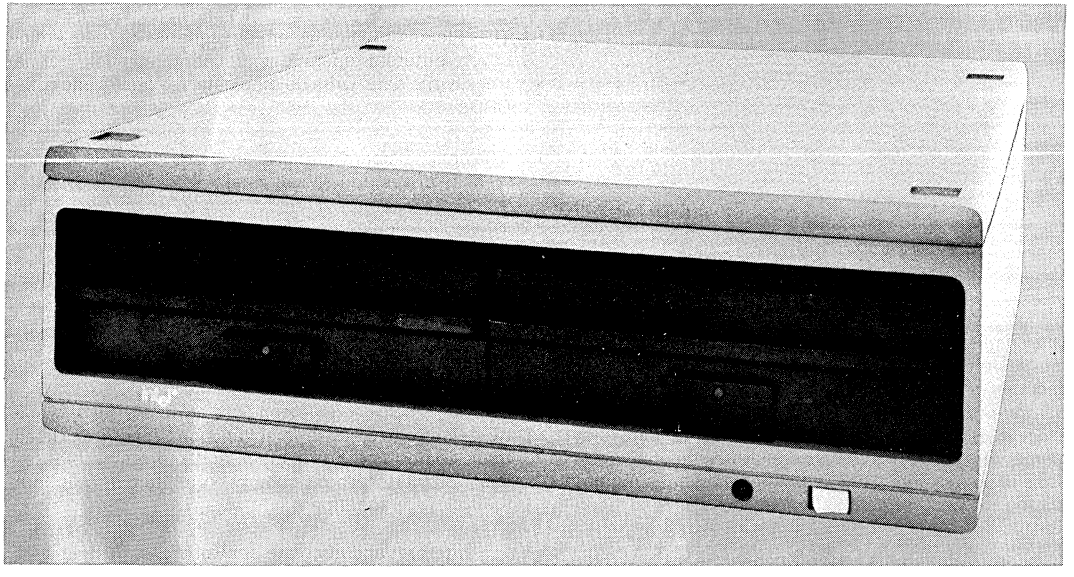
***MDS** is an ordering code only, and is not used as a product name or trademark.
MDS® is a registered trademark of Mohawk Data Sciences Corporation.

Flexible Disk Systems 14

INTELLEC® DOUBLE DENSITY FLEXIBLE DISK SYSTEM

- **Flexible Disk System Providing High Speed Input/Output and Data Storage for Intellec® Microcomputer Development Systems**
- **Data Recorded on Double Density Flexible Disk is in Soft-Sectored Format Which Allows ½ Million Byte Data Capacity with Up to 200 Files Per Flexible Disk**
- **Associated Software Supports Up to Four Double Density Drives and Two Single Density Drives, Providing Up to 2.5 Megabytes of Storage in One System**
- **Dynamic Allocation and Deallocation of Flexible Disk Sectors for Variable Length Files**

The Intellec Flexible Disk System is a sophisticated, general purpose, bulk storage peripheral for use with the Intellec Microcomputer Development System. The use of a flexible disk operating system significantly reduces program development time. The software system known as ISIS-11 (Intel System Implementation Supervisor), provides the ability to edit, assemble, compile, link, relocate, execute and debug programs, and performs all file management tasks for the user.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{i}m, ICS, ICE, UPI, BXP, iSBC, iSBX, iNSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

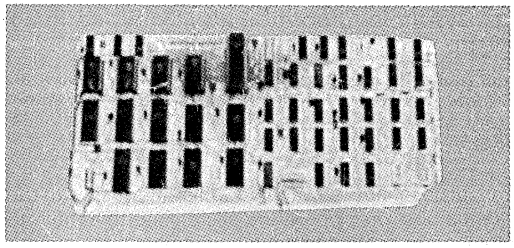
© Intel Corporation 1980

AFN-01050 B

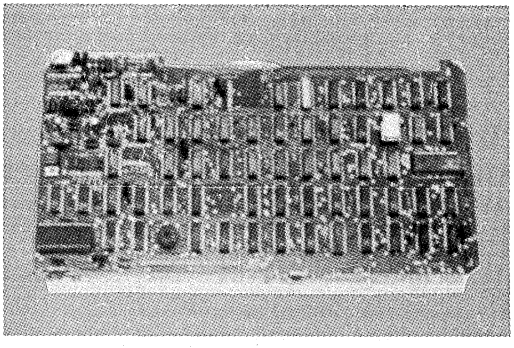
HARDWARE

The Intellec® flexible disk system provides direct access bulk storage, intelligent controller, and two flexible disk drives. The double density drive provides ½ million bytes of storage with a data transfer rate of 500,000 bits/second. The controllers are implemented with Intel's powerful Series 3000 Bipolar Microcomputer Set. The controllers provide interface to the Intellec System bus. Each double density controller will support up to four drives. The flexible disk system records all data in soft sector format.

The double density flexible disk controllers each consists of two boards, the Channel Board and the Interface Board. These two printed circuit boards reside in the Intellec System chassis. The boards are shown in the photograph, and are described in more detail in the following paragraphs.



DOUBLE DENSITY CHANNEL BOARD



DOUBLE DENSITY INTERFACE BOARD

CHANNEL BOARD

The *Channel Board* is the primary control module within the flexible disk system. The Channel Board receives, decodes, and responds to channel commands from the Central Processor Unit (CPU) in the Intellec system. The Channel Board can access a block of Intellec system memory to determine the particular flexible disk operations to be performed and fetch the parameters required for the successful completion of the specified operation.

The control functions of the Channel Board have been achieved with an 8-bit microprogrammed processor, designed with Intel's Series 3000 Bipolar Microcomputer Set. This 8-bit processor includes four 3002 Central Processing Elements (2-bit slice per CPE), a 3001 Microprogram Control Unit, and 512 x 32 bits of 3604 programmable-read-only-memory (PROM) which stores the microprogram. It is the execution of the microprogram by the microcomputer set which actually effects the control capability of the Channel Board.

INTERFACE BOARD

The *Interface Board* provides the flexible disk controller with a means of communication with the flexible disk drives, as well as with the Intellec system bus. Under control of the microprogram being executed on the Channel Board, the Interface Board generates those signals which cause the read/write head on the selected drive to be loaded (i.e., to come in contact with the flexible disk platter), cause the head to move to the proper track and verify successful operation. The Interface Board accepts the data being read off the flexible disk, interprets synchronizing bit patterns, checks the validity of the data using a cyclic redundancy check (CRC) polynomial, and then transfers the data to the Channel Board.

During write operations, the Interface Board outputs the data and clock bits to the selected drive at the proper times, and generates the CRC characters which are then appended to the data.

When the flexible disk controller requires access to Intellec system memory, the Interface Board requests the DMA master control of the system bus, and generates the appropriate memory command. The Interface Board also acknowledges I/O commands as required by the Intellec bus.

The Flexible Disk System is capable of performing seven different operations: recalibrate, seek, format track, write data, write deleted data, read data, and verify CRC.

The channel board supports up to two sets of dual double density drives (four drives total).

The double density controller may co-reside with the Intel single density controller to allow conversion of single density flexible disk to double density format, and provide up to 2.5M bytes of storage.

FLEXIBLE DISK DRIVE MODULES

Each flexible disk drive consists of read/write and control electronics, drive mechanisms, read/write head, track positioning mechanism, and the removable flexible disk platter. These components interact to perform the following functions:

- Interpret and generate control signals
- Move read/write head to selected track
- Read and write data

ASSOCIATED SOFTWARE — INTEL SYSTEMS IMPLEMENTATION SUPERVISOR (ISIS-II)

The Flexible Disk Drive System is to be used in conjunction with the ISIS-II Operating System. ISIS-II provides total file management capabilities, file editing, library management, run-time supports, and utility management.

ISIS-II provides automatic implementation of random access disk files. Up to 200 files may be stored on each ¼ million byte flexible disk for single density system or on each ½ million byte flexible disk for double density system. For more information, see the ISIS-II data specification sheet.

ISIS-II OPERATIONAL ENVIRONMENTAL

ISIS-II

32K bytes RAM memory
48K bytes when using Assembler Macro feature
64K bytes when using PL/M or Fortran
System Console
Double density Flexible Disk Drive

HARDWARE SPECIFICATIONS

MEDIA

Double Density
Double Density Specified Flexible Disk
One Recording Surface
Soft Sector Format
77 Tracks/Diskette
52 Sectors/Track
128 Bytes/Sector

PHYSICAL CHARACTERISTICS

CHASSIS AND DRIVES

Mounting:	Table-Top
Height:	5.7 in. (14.5 cm)
Width:	17.6 in. (44.7 cm)
Depth:	19.4 in. (49.3 cm)
Weight:	43.0 lb. (19.5 kg)

ELECTRICAL CHARACTERISTICS

CHASSIS

DC Power Supplies
Supplied Internal to the Cabinet

AC Power Requirements
3-wire input with center conductor (earth ground) tied to chassis

Single-phase, 115 VAC; 60 Hz; 1.2 Amp Maximum (For a Typical Unit)
230 VAC; 50 Hz; 0.7 Amp Maximum (For a Typical Unit)

FLEXIBLE DISK OPERATING SYSTEM CONTROLLER

DC Power Requirements (All power supplied by Intellec Development System)

CHANNEL BOARD

Double Density
5V @ 3.75A (typ), 5A (max)

Double Density
5V @ 1.5A (typ), 2.5A (max) – 10V @ 0.1A (typ), 0.2A (max)

FLEXIBLE DISK DRIVE PERFORMANCE SPECIFICATION

	Double Density
Capacity (Unformatted):	
Per Disk	6.2 megabits
Per Track	82 kilobits
Capacity (Formatted):	
Per Disk	4.10 megabits
Per Track	53.2 kilobits
Data Transfer Rate	500 kilobits/sec
Access Time:	
Track-to-Track	10 ms
Head Settling Time	10 ms
Average Random Positioning Time	260 ms
Rotational Speed	360 rpm
Average Latency	83 ms
Recording Mode	M^2FM

ENVIRONMENTAL CHARACTERISTICS

MEDIA

Temperature:	
Operating:	15.6°C to 51.7°C
Non-Operating:	5°C to 55°C
Humidity:	
Operating:	8 to 80% (Wet bulb 29.4°C)
Non-Operating:	8 to 90%

DRIVES AND CHASSIS

Temperature:	
Operating:	10°C to 38°C
Non-Operating:	– 35°C to 65°C
Humidity:	
Operating:	20% to 80% (Wet bulb 26.7°C)
Non-Operating:	5% to 95%

CONTROLLER BOARDS

Temperature:	
Operating:	0 to 55°C
Non-Operating:	– 55°C to 85°C
Humidity:	
Operating:	Up to 95% relative humidity without condensation
Non-Operating:	All conditions without condensation of water or frost

EQUIPMENT SUPPLIED

DOUBLE DENSITY

Cabinet, Power Supplies, Line Cord, Two Drives
Double Density FDC Channel Board
Double Density FDC Interface Board
Dual Auxiliary Board Connector
Flexible Disk Controller Cable
Flexible Disk Peripheral Cable
Hardware Reference Manual
Reference Schematics
ISIS-II Double Density System Disk
ISIS-II System User's Guide

OPTIONAL EQUIPMENT

MDS-BLD*	10 Blank Flexible Disks
MDS-730/731*	Second Drive Cabinet with two additional drives

ORDERING INFORMATION

Part Number	Description
MDS-720-110V* 721/220V	Flexible Disk drive unit with two drives, double density drive controller, software, and cables.
MDS-730/110V* 731/220V	Add-on drive unit with two drives and double density cable, without controller and software. Can be used with double density controller.

*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

MCS–80/85™ Development Systems and Options

15

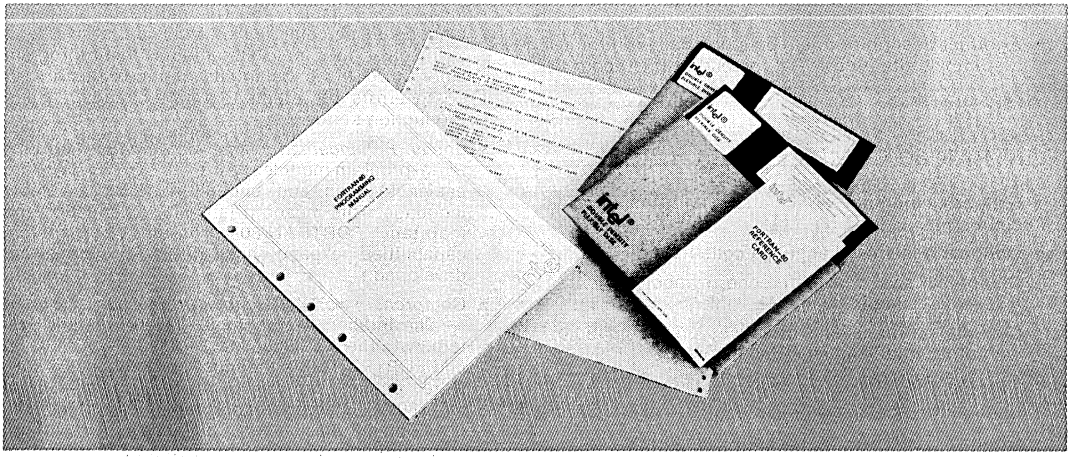
FORTRAN 80
8080/8085 ANS FORTRAN 77
INTELLEC® RESIDENT COMPILER

- **Meets ANS FORTRAN 77 Subset Language Specification plus adds Intel microprocessor extensions**
- **Supports Intel Floating Point Standard with the FORTRAN 80 software routines, the iSBC-310™ High Speed Mathematics Board, or the iSBC-332™ math multimodule**
- **Executes on Intellec Microcomputer Development System and Intellec Series II Microcomputer Development System**
- **Supports full symbolic debugging with ICE-80™ and ICE-85™**
- **Produces relocatable and linkable object code compatible with resident PL/M 80 and 8080/8085 Macro Assembler**
- **Provides optional run-time library to execute in RMX-80™ environment**
- **Has well defined I/O interface for configuration with user-supplied drivers**

FORTRAN 80 is a computer industry-standard, high-level programming language and compiler that translates FORTRAN statements into relocatable object modules. When the object modules are linked together and located into absolute program modules, they are suitable for execution on Intel 8080/8085 Microprocessors, iSBC-80 OEM Computer Systems, and Intellec Microcomputer Development Systems. FORTRAN 80 meets the ANS FORTRAN 77 Language Subset Specification[1]. In addition, extensions designed specifically for microprocessor applications are included. The compiler operates on the Intellec Microcomputer Development System under the ISIS-II Disk Operating Systems and produces efficient relocatable object modules that are compatible for linkage with PL/M 80 and 8080/8085 Macro Assembler modules.

The ANS FORTRAN 77 language specification offers many powerful extensions to the FORTRAN language that are especially well suited to Intel 8080/8085 Microprocessor software development. Because FORTRAN 80 conforms to the ANS FORTRAN 77 standard, the user is assured of compatibility with existing FORTRAN software that meets the standard as well as a guarantee of upward compatibility to other computer systems supporting an ANS FORTRAN 77 Compiler.

[1]ANSI X3J3/90



9800668-03 Rev. C

FORTRAN 80 LANGUAGE FEATURES

Major ANS FORTRAN 77 features supported by the Intel® FORTRAN 80 Programming Language include:

- Structured Programming is supported with the IF ... THEN ... ELSE IF ... ELSE ... END IF constructs.
- CHARACTER data type permits alphanumeric data to be handled as strings rather than characters stored in array elements.
- Full I/O capabilities include:
  - Sequential and Direct Access files
  - Error handling facilities
  - Formatted, Free-formatted, and Unformatted data representation
  - Internal (in-memory) file units provide capability to format and reformat data in internal memory buffers
  - List Directed Formatting
- Supports arrays of up to seven dimensions.
- Supports logical operators
  .EQV. — Logical equivalence
  .NEQV. — Logical nonequivalence

Major extensions to FORTRAN 77 in Intel FORTRAN-80 include:

- Direct 8080/8085 port I/O supported by intrinsic subroutines.
- Binary and Hexadecimal integer constants.
- Well defined interface to FORTRAN-80 I/O statements (READ, OPEN, etc.), allowing easy use of user-supplied I/O drivers.
- User-defined INTEGER storage lengths of 1, 2 or 4 bytes.
- User-defined LOGICAL storage lengths of 1, 2 or 4 bytes.
- REAL STORAGE lengths of 4 bytes.
- Bitwise Boolean operations using logical operators on integer values.
- Hollerith data constants.
- Implicit extension of the length of an integer or logical expression to the length of the left-hand side in an assignment statement.
- A format descriptor to suppress carriage return on a terminal output device at the end of the record.

FORTRAN 80 COMPILER FEATURES

- Supports multiple compilation units in single source file.
- Optional Assembly Language code listing.
- Comprehensive cross-reference, symbol attribute and error listing.
- Compiler controls and directives are compatible with other Intel language translators.
- Optional Reentrancy.
- User-defined default storage lengths.
- Optional FORTRAN 66 Do Loop semantics.
- Source files may be prepared in free format.
- The INCLUDE control permits specified source files to be combined into a compilation unit at compile time.
- Transparent interface for software and hardware floating point support, allowing either to be chosen at time of linking.

FORTRAN 80 BENEFITS

FORTRAN 80 provides a means of developing application software for Intel® MCS-80/85 products in a familiar, widely accepted, and computer industry-standardized programming language. FORTRAN 80 will greatly enhance the user's ability to provide cost-effective solutions to software development for Intel microprocessors as illustrated by the following:

- *Completely Complementary to Existing Intel Software Design Tools* — Object modules are linkable with new or existing Assembly Language and PL/M Modules.
- *Incremental Runtime Library Support* — Runtime overhead is limited only to facilities required by the program.
- *Low Learning Effort* — FORTRAN 80, like PL/M, is easy to learn and use. Existing FORTRAN software can be ported to FORTRAN 80, and programs developed in FORTRAN 80 can be run on any other computer with ANS FORTRAN 77.
- *Earlier Project Completion* — Critical projects are completed earlier than otherwise possible because FORTRAN 80 will substantially increase programmer productivity, and is complementary to PL/M Modules by providing comprehensive arithmetic, I/O formatting, and data management support in the language.
- *Lower Development Cost* — Increases in programmer productivity translates into lower software development costs because less programming resources are required for a given function.
- *Increased Reliability* — The nature of high-level languages, including FORTAN 80, is that they lend themselves to simple statements of the program algorithm. This substantially reduces the risk of costly errors in systems that have already reached production status.
- *Easier Enhancements and Maintenance* — Like PL/M, program modules written in FORTRAN 80 are easier to read and understand than assembly language. This means it is easier to enhance and maintain FORTRAN 80 programs as system capabilities expand and future products are developed.
- *Comprehensive, Yet Simple Project Development* — The Intellec Microcomputer Development System, with the 8080/8085 Macro Assembler, PL/M 80 and FORTRAN 80 is the most comprehensive software design facility available for the Intel MCS-80/85 Microprocessor family. This reduces development time and cost because expensive (and remote) timesharing or large computers are not required.

SAMPLE FORTRAN-80 SOURCE PROGRAM LISTING

```
*    **  THIS PROGRAM IS AN EXAMPLE OF ISIS-II FORTRAN-80 THAT
*    **  CONVERTS TEMPERATURE BETWEEN CELSIUS AND FARENHEIT

      PROGRAM CONVRT

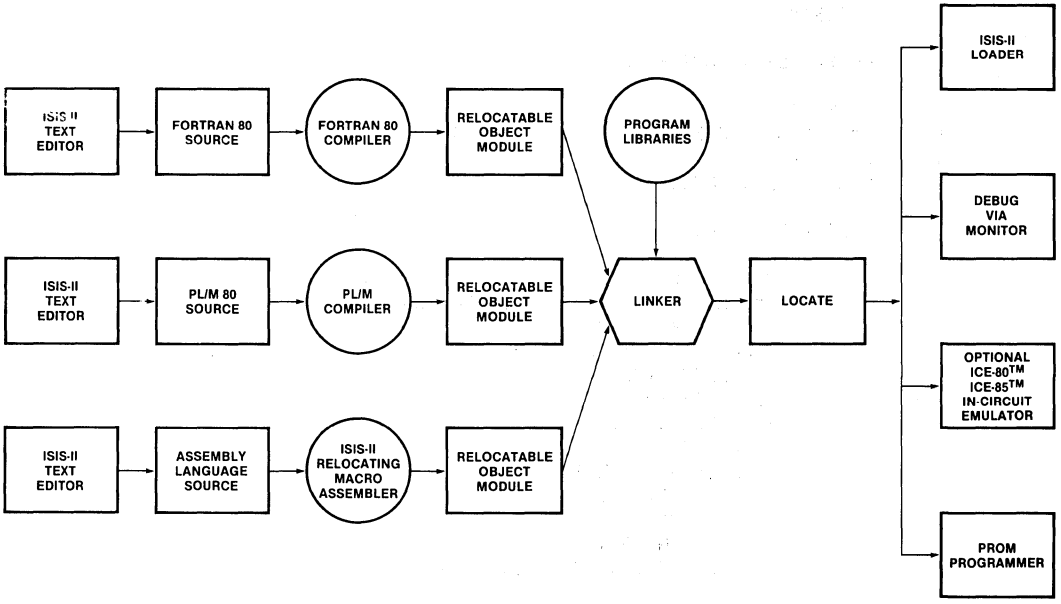
      CHARACTER*1 CHOICE, SCALE

      PRINT 100
*     **  ENTER CONVERSION SCALE (C OR F)
10    PRINT 200
      READ (5,300) SCALE

        IF (SCALE .EQ. 'C')
     +     THEN
              PRINT 400
*             **  ENTER THE NUMBER OF DEGREES FARENHEIT
              READ (5,*) DEGF
              DEGC = 5./9.*(DEGF-32)
*             **  PRINT THE ANSWER
              WRITE (6,500) DEGF,DEGC
*             **  RUN AGAIN?
20            PRINT 600
              READ (5,300) CHOICE
                 IF (CHOICE .EQ. 'Y')
     +              THEN
                       GOTO 10
                 ELSE IF (CHOICE .EQ. 'N')
     +              THEN
                       CALL EXIT
                 ELSE
                    GOTO 20
                 END IF
        ELSE IF (SCALE .EQ. 'F')
     +     THEN
*             **  CONVERT FROM FARENHEIT TO CELSIUS
              PRINT 700
              READ (5,*) DEGC
              DEGF = 9./5.*DEGC+32.
*             **  PRINT THE ANSWER
              WRITE (6,800) DEGC,DEGF
              GOTO 20
        ELSE
*          **  NOT A VALID ENTRY FOR THE SCALE
           WRITE (6,900) SCALE
           GOTO 10
        END IF
100   FORMAT(' TEMPERATURE CONVERSION PROGRAM',//,
     +' TYPE C FOR FARENHEIT TO CELSIUS OR',/,
     +' TYPE F FOR CELSIUS TO FARENHEIT',//)
200   FORMAT(/,' CONVERSION? ',$)
300   FORMAT(A1)
400   FORMAT(/,'ENTER DEGREES FARENHEIT: ',$)
500   FORMAT(/,F7.2,' DEGREES FARENHEIT = ',F7.2,' DEGREES CELSIUS')
600   FORMAT(/,' AGAIN (Y OR N)? ',$)
700   FORMAT(/,' ENTER DEGREES CELSIUS: ',$)
800   FORMAT(/,F7.2,' DEGREES CELSIUS = ',F7.2,' DEGREES FARENHEIT',/)
900   FORMAT(/,1H ,A1,' NOT A VALID CHOICE - TRY AGAIN!',/)
      END
```

FORTRAN 80

The FORTRAN 80 Compiler is an efficient, multiphase compiler that accepts source programs, translates them into relocatable object code, and produces requested listings. After compilation, the object program may be linked to other modules, located to a specific area of memory, then executed. The diagram shown below illustrates a program development cycle where the program consists of modules created by FORTRAN 80, PL/M 80 and the 8080/8085 Macro Assembler.



SPECIFICATIONS

OPERATING ENVIRONMENT

Required Hardware:

Intellec® Microcomputer Development System
- — MDS-800, MDS-888
- — Series II Model 220, Model 230

64K bytes of RAM memory

Dual diskette drives
- — Single or Double Density

System console
- — CRT or hardcopy interactive device

Optional Hardware:

Line Printer
ICE-80™, ICE-85™

Required Software:

ISIS-II Diskette Operating System
- — Single or Double Density

Optional Software:

iSBC-801 FORTRAN-80 Run-Time Software Package for RMX-80

DOCUMENTATION PACKAGE

FORTRAN-80 Programming Manual (9800481)

ISIS-II FORTRAN-80 Compiler Operator's Manual (9800480)

FORTRAN-80 Programming Reference Card (9800547)

SHIPPING MEDIA

Flexible Diskettes
- — Single and Double Density

ORDERING INFORMATION

PART NO.	DESCRIPTION
*MDS-301	FORTRAN 80 Compiler for Intellec Microcomputer Development Systems

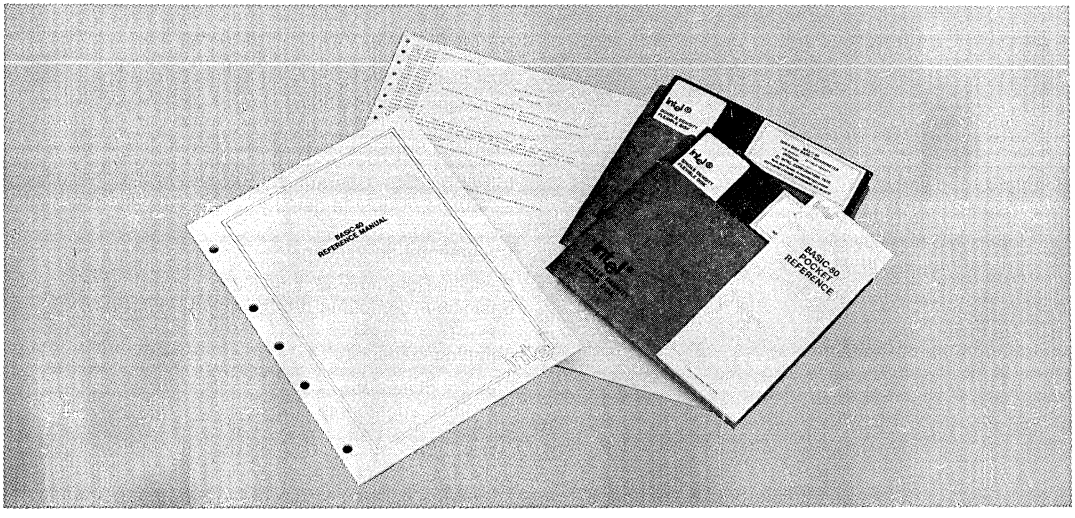
***"MDS"** is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.

BASIC-80
EXTENDED ANS 1978 BASIC
INTELLEC® RESIDENT INTERPRETER

- **Meets ANS 1978 Standard for Minimal BASIC and Adds Many Powerful Extensions**
- **Operates Under the ISIS-II Operating System on Intellec® and Intellec® Series-II Microcomputer Development Systems**
- **Full Sequential and Random Disk File I/O with ISIS-II**
- **Applications Range from Prototyping Microcomputer Software to Inexpensive Engineering and Management Problem Solving on the Intellec® Systems**
- **Supports the Intel® Floating Point Standard and Provides Integer and String Data Types**
- **Can Call User Subroutines Written in FORTRAN 80, PL/M 80, and 8080/85 Macro Assembler that are Resident in the Intellec® Memory**
- **Easily Learned Language and Interactive Environment Combine to Provide a Flexible and Powerful Facility for Developing Programs to Run on the Intellec® Microcomputer Development Systems**

BASIC is an industry standard, high-level programming language which is designed to be easily learned and used by novices and experienced programmers alike. The interpreter provides an interactive environment which allows fast and easy program development, testing, and debugging. BASIC is widely used for problem solving in engineering and management; extensive software exists for business applications such as order entry, accounts receivable, accounts payable, and inventory control, and engineering applications such as numeric and statistical analysis.

Intel's BASIC-80 meets the standards of ANS 1978 BASIC and extends them to take advantage of the software development capabilities of the Intellec Microcomputer Development Systems. The matching of these resources with the ease of programming in BASIC-80 provides a very effective tool for both microprocessor systems development and inexpensive applications programming and problem solving on the Intellec systems.



BASIC-80 LANGUAGE FEATURES

Standard ANS 78 BASIC features, all supported by BASIC-80, include:

- String and numeric constants, variables, and arrays.
- FOR . . . TO . . . STEP . . . NEXT statements for loop execution.
- IF . . . THEN statements for conditional execution.
- ON . . . GOTO statements for computed branching.
- GOSUB/RETURN subroutine calls and returns.
- Built in scientific functions:

ABS	RND	TAN
EXP	SGN	COS
INT	SQR	SIN
LOG	ATN	

- User defined single statement functions.

Major extensions to ANS 78 BASIC which BASIC-80 provides include:

- Support for the Intel single and double precision floating point standard.
- Disk file I/O, supporting both random access and sequential access files.
- Direct read and write to CPU I/O ports through the INP and OUT functions.
- Direct memory read and write through the PEEK and POKE functions.
- Calls to user-supplied external subroutines, which may have been written in FORTRAN-80, PL/M-80, or 8080/8085 Assembly Language and have been located at absolute memory locations using the ISIS-II facilities.
- User directed error trapping and handling functions.
- Program execution trace command.
- Formatted print statement with the PRINT USING function.
- ELSE clause for IF . . . THEN statements.
- Matrices with up to 110 dimensions.
- Extensive string manipulation functions.
- Boolean operators.
- Type conversion functions—integer, floating point, and character.

BENEFITS OF BASIC-80

- Added Value to the Intellec Systems—with BASIC-80 the Intellec Microcomputer Development Systems can be effectively used in many engineering and management applications.
- Inexpensive and Accessible Computational Facility—the ease of use and flexibility inherent in BASIC-80 and its interpretive environment fit well with the "at hand" computational resources of the Intellec systems. The combination is a particularly useful tool for obtaining fast and accurate results.
- Easy to Learn—the language is designed to be easily understood and learned. Results are obtained faster and people who may benefit from using the system can do so easily.
- Aid in Microcomputer Software Design—microcomputer software can be prototyped in BASIC-80 to inexpensively develop and test program logic.
- Complemented by Existing Software—subroutines written in PL/M-80, FORTRAN-80, and ASM 8080/85 can be called from BASIC-80 programs.
- Easy to Enhance and Maintain—BASIC-80, being straightforward and easily understood, provides for programs that are easy to maintain and modify in the future.

SPECIFICATIONS

Operating Environment

Required Hardware:

Intellec Microcomputer Development System
— Models 800 and 888
— Series-II Model 220, Model 230

48K bytes of RAM memory

Diskette drive
— Single or double density

System console
— CRT or hard copy interactive device

Optional Hardware:

Line printer
Additional diskette drive

Required Software:

ISIS-II Diskette Operating System
— Single or double density

Documentation Package:

Basic-80 Reference Manual (9800758A)
Basic-80 Programming Reference Card (9800774)

Shipping Media:

Flexible diskettes
— Single and double density

EXAMPLE BASIC-80 PROGRAM

```
list
10 PRINT "THIS PROGRAM CALCULATES THE MEAN AND STANDARD"
20 PRINT " DEVIATION OF INPUT DATA"
30 S=0:V=0
40 INPUT "NUMBER OF VALUES";N
50 FOR I=1 TO N
60 INPUT A(I)
70 S=S+A(I)
80 NEXT
90 S=S/N
100 REM CALCULATION OF VARIANCE
110 FOR I=1 TO N
120 V=V+(A(I)-S)^2/N
130 NEXT
140 SD=SQR(V)
150 PRINT "MEAN=";S
160 PRINT "STANDARD DEVIATION IS=";SD
Ok


run
THIS PROGRAM CALCULATES THE MEAN AND STANDARD
 DEVIATION OF INPUT DATA
NUMBER OF VALUES? 6
? 34.7
? 32.9
? 38.2
? 35
? 37.6
? 40.9
MEAN= 36.55
STANDARD DEVIATION IS= 2.642442
Ok
```

ORDERING INFORMATION

Product Code	Description
MDS-320*	ISIS-II BASIC-80 Disk-Based Interpreter

*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Science Corporation.

PASCAL 80
SOFTWARE PACKAGE

- **Offers a Superset of Standard Pascal**
- **Provides Highly Structured Language with Powerful Data Type Definitions to Suit Applications**
- **Compiles Pascal Source Code into Intermediate Code to Optimize Execution Speed and Storage**
- **Executes Compiler and Interprets the Intermediate Code on Intellec® Microcomputer Development Systems**
- **Provides a Utility to Produce Relocatable Object Modules Compatible with Other Intel® Languages**
- **Can Call Routines Written in PL/M 80, FORTRAN 80, or 8080/8085 Macro Assembler**
- **Allows Modular Breakdown of Large Programs and Separate Compilation of Individual Modules**
- **Gives Application Control Over Run-Time Errors by Providing User-Declared Error Procedures**

PASCAL 80 Software Package consists of a compiler and an interactive Run-Time System designed to provide the Pascal programming language as a software development tool for Intellec Development System Users.

Pascal is a highly-structured, block-oriented programming language that is now gaining wide acceptance as a powerful software development tool. Its rigid structure encourages and enforces good programming techniques, which, combined with a high level of readability, helps produce more reliable software.

Standard Intel development tools, such as CREDIT editor can be used to create and modify Pascal source programs. The compiler compiles this source and creates a P-Code file. The Run-Time System executes this P-Code in an interpretive manner under ISIS-II.

*Pascal language as defined in *PASCAL User Manual and Report*, Second Edition, Kathleen Jenson and Niklaus Wirth.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Intel, INTEL, INTELLEC, MCS, Im, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

LANGUAGE FEATURES

Data Structures

Pascal allows the user to define labels, constants, data types, variables, procedures, and functions.

Variable Types

Variables can be defined according to the following system-defined data types: boolean, integer, real, character, array, record, string, set, file, and pointer.

User-Defined Types

New types can be defined by the user for added flexibility.

File Handling Procedures

Pascal provides procedures to allow a user's program to interface with the ISIS-II file manager. Routines provided are: RESET, REWRITE, CLOSE, PUT, GET, SEEK, and PAGE.

Input/Output Procedures

Routines are provided to interact with the console or an ISIS file. These procedures are: READ, WRITE, READLN, WRITELN, plus BUFFER and BLOCK Read and Write.

Dynamic Memory Allocation

The procedures NEW, MARK, and RELEASE allow the user to obtain and release memory space at run-time for dynamically allocating variable storage.

String Handling

Pascal provides powerful tools for defining and manipulating strings and character arrays. These facilities enable concatenation of strings, character and pattern scans, insertion, deletion, and pointer manipulation.

Recursion

Pascal allows a PROCEDURE definition to include a call to itself, a powerful construct in many mathematical algorithms.

PROGRAM TRACING FACILITY

The PASCAL 80 System incorporates a program tracing facility which allows for selectively monitoring the execution of a Pascal program. When the TRACE flag is set, the line number of each program statement being executed is output to the console.

The TRACE flag may be manipulated in two ways:

—The TRACEON command (of the Run-Time System) will set the flag, and the TRACEOFF command will reset the flag.

—Pressing the Interrupt 4 switch on the Intellec System front panel will toggle the TRACE flag; i.e., the flag will be set if it was reset, and vice-versa.

COMPILER DIRECTIVES (PARTIAL LIST)

Compiler Command Line Directives

NOLIST
No list file is produced; used for fast compilation of "clean" programs.

NOCODE
No code file is produced; used for syntax error checking.

ERRLIST
List file is limited to only those Pascal lines that contain errors, along with the error messages produced.

LIST (file-name)
Specifies the name of the list file.

CODE (file-name)
Specifies the name of the code file.

NOECHO
Error lines are echoed on the console unless this directive is specified.

Embedded Compiler Directives

$C text
Causes text to appear in code file (allows for comments, copyrights, etc.).

$I+
Causes checking for I/O completion after each I/O transfer. Failure results in a run-time error. ($I− causes no checking, and no errors on I/O failure.)

$R+
Causes Range Checking to occur, so that an out-of-range value causes a Run-Time error. ($R− suppresses generation of code for Range Checking.)

$O+
Causes the compiler to operate in overlay mode. Overlays allow less source code to reside in memory. ($O− causes no overlays, which decreases compile time, since there are fewer disk accesses.)

$T+
Causes the compiler to generate tracing instructions to be used by the TRACE facility. ($T− suppresses tracing instructions.)

BENEFITS

Brings Pascal to Intellec Microcomputer Development Systems:

—Pascal is a block-structured, highly-readable programming language, suitable for a wide-range of applications.

—Pascal is being acclaimed as the programming language of the future; it is being taught in many colleges and universities around the country.

—PASCAL 80 Run-Time System provides great ease in programming formatted I/O operations.

PASCAL 80 provides a portable language for application programs running under ISIS-II.

PASCAL 80 can be used to evaluate complicated algorithms using a natural language.

PASCAL 80 compiler generates intermediate Pseudo-code.

—P-code is optimized for speed and storage space.

—P-code is approximately 50% to 70% smaller than corresponding machine code.

—P-code is machine independent, providing code portability to any CPU.

Makes the Intellec Development System a more valuable tool. Extension of software support to include Pascal makes software development and resource management more flexible.

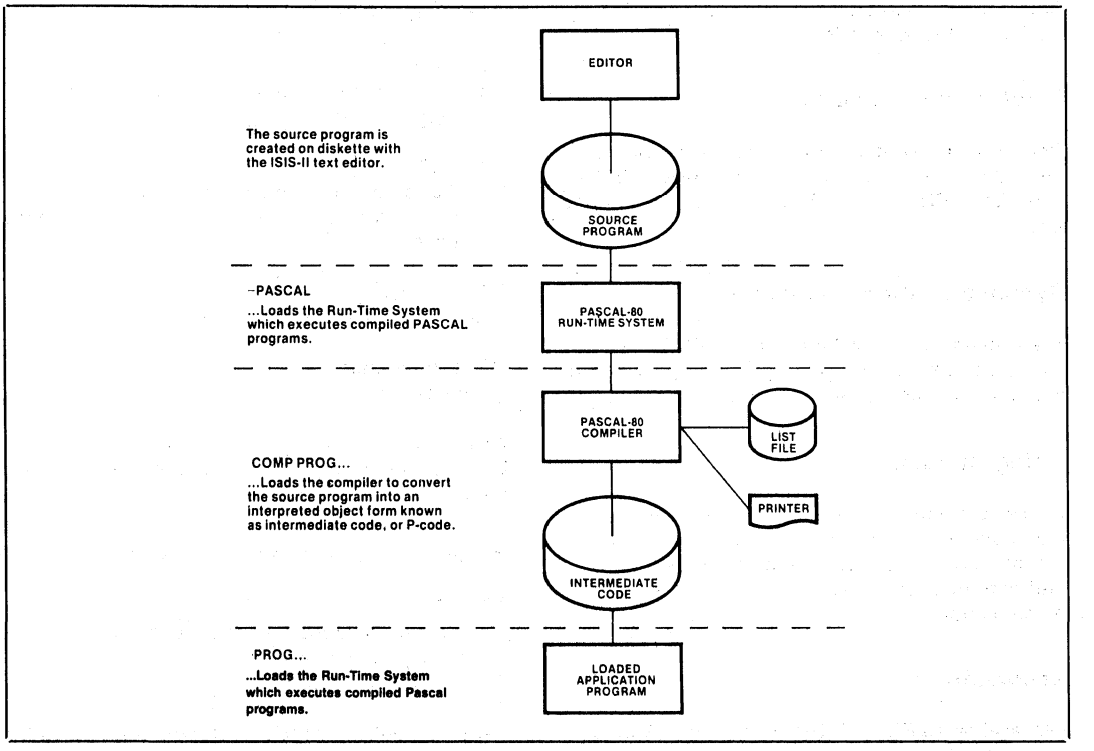


Figure 1. Program Development Cycle

Table 1. Sample Program Listing Showing Nesting Levels

BUFFER.PAS Program Listing

```
Line  Seg  Proc  Lev  Disp

   1    1     1          1   program example;
   2    1     1          3
   3    1     1          3   { Example using bufferread and bufferwrite with break characters }
   4    1     1          3
   5    1     1          3   var buffer: string;
   6    1     1         44       disk  storage: file;
   7    1     1         64       break: char;
   8    1     1         65       new  len, len: integer;
   9    1     1         67       buff  array: packed array[0..80] of char;
  10    1     1        108
  11    1     1    0    0   begin
  12    1     1    1    0     rewrite (disk  storage, 'data');
  13    1     1    1   27     writeln('Input a line of text: ');
  14    1     1    1   68     readln (buffer);
  15    1     1    1   87     len := bufferwrite(disk  storage, buffer[1], length(buffer));
  16    1     1    1  109     repeat
  17    1     1    2  109       reset(disk  storage);
  18    1     1    2  116       writeln; writeln;
  19    1     1    2  132       write('Input break char [cntrl Z to stop]: ');
  20    1     1    2  179       readln(break);
  21    1     1    2  197       if not eof(input) then
  22    1     1    3  208         begin
  23    1     1    4  208           new  len := bufferread(disk  storage, buff  array, len, ord(break));
  24    1     1    4  226           writeln('The buffer read: ');
  25    1     1    4  262           writeln(copy(buffer, 1, abs(new  len)));
  26    1     1    4  292           writeln('Length: ', abs(new  len):0);
  27    1     1    4  331           if new  len < 0 then writeln('(Break char not found)');
  28    1     1    3  378         end;
  29    1     1    1  378     until eof(input);
  30    1     1    0  388   end.
```

SPECIFICATIONS

Operating Environment

REQUIRED HARDWARE
Intellec® Microcomputer Development System
—Model 800
—Series II Model 220, Model 230, Model 240
64KB of Memory
Dual-Diskette Drives
—Single- or Double-Density*
System Console
—Intel® CRT or non-Intel® CRT

*Recommended.

REQUIRED SOFTWARE
ISIS-II Diskette Operating System
—Single- or Double-Density

OPTIONAL SOFTWARE
ISIS-II CREDIT™ (CRT-Based Text Editor)

Documentation Package

PASCAL 80 User's Guide (9801015-01)

PASCAL User Manual and Report, Second Edition, Kathleen Jensen and Niklaus Wirth

Shipping Media

Flexible Diskettes
—Single- and Double-Density

ORDERING INFORMATION

Part Number	Description
MDS-381*	PASCAL 80 Software Package

Requires Software License

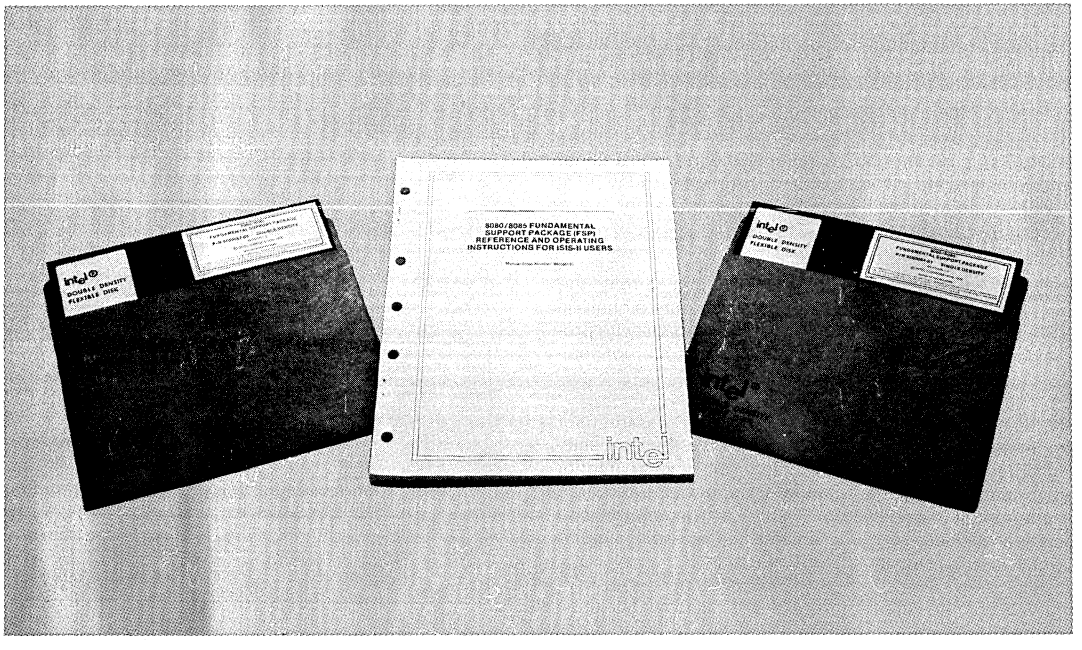
*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

8080/8085 FUNDAMENTAL SUPPORT PACKAGE (FSP)

- **Comprehensive Extension of the 8080 CPU Performance Features Through Standardized Software Building Blocks**
- **Hardware-Independent Modules Selectable by the User for a Wide Variety of Frequently Needed Program Functions**
- **High Standard of Reliability Achieved Through Use of Proven Algorithms and Meticulous Testing of All Functions**
- **Library Routines Provide Extended Math Capabilities of ASM 80, PL/M 80, and FORTRAN 80**
- **Available on Diskette as an ISIS-II Library File**
- **Efficient Integration with the User's Program Using the ISIS Utility Programs LINK and LOCATE**
- **Lowers Software Development Costs**

FSP is a mathematical and technical applications library. It is a comprehensive collection of frequently needed program functions that complement and augment the capabilities of the 8080/8085 microprocessor. The inclusion of these functions eliminates the need to code and debug complicated math routines and permits the user to concentrate on his own application. The use of FSP functions thus makes a noteworthy contribution to the economy of software development.

FSP consists of a collection of subroutines stored as relocatable modules on a diskette. These modules can, with the help of ISIS utilities, LINK and LOCATE, be linked to user programs in assembly language, PL/M or FORTRAN.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{l}m, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

121526-001

PACKAGE FEATURES

The Fundamental Support Package consists of nine individual libraries that lie in a hierarchical structure, as shown in figure 1. These nine sections are:

- The *FSP Machine* (primitive subroutine) package performs fast string handling and binary and decimal integer arithmetic without error reporting.
- The *binary integer arithmetic* routines provide operations on signed and unsigned integers of various formats in binary representation.
- The *floating-point arithmetic* section provides operations on floating-point (real) numbers in four formats: single precision, single-precision extended, double precision, and double-precision extended.
- The *decimal arithmetic* routines provide integer and fixed-point arithmetic on numbers in decimal representation—i.e., stored as strings of ASCII characters.
- The *string handling* section contains routines to transform strings and to extract and insert substrings. A routine for scanning of general input and one for formatting of general output are included.
- The routines for *number conversion and numeric I/O* do transformation of numeric data from one internal format to another, input scanning of numeric strings and formatting of numeric strings for output.
- The *floating-point transcendental function* section provides trigonometric, exponential, and other transcendental functions for single precision, single-precision extended, double precision, and double-precision extended floating-point arguments.
- The *statistics* routines compute the mean, variance, and standard deviation of one group of statistical data, and the covariance and correlation factor of two groups of data.
- The *P.I.D. process control* routines direct the production of an appropriate output signal in response to an input signal, using a formula with proportional, integral, and/or derivative terms, for real-time process control applications.

In linking modules to an application program, the user must note the hierarchical structure of FSP and specify in order the lower level packages on which a higher level package must rely. Figure 1 shows the required subordination of subroutines; for example, the statistical package relies on the floating-point library, which in turn relies on the FSP machine.

All FSP routines are reentrant; that is, all local data used by each routine is stored on the stack. These routines may thus be interrupted, and during the interrupt the same routine or other routines may be called, without affecting the results of the interrupted routine.

DESCRIPTION OF THE LIBRARIES

FSP Machine (FSLMCH.LIB)

The routines in this library, together with the 8080/8085 hardware, constitute the FSP Machine. The routines fall into two categories: the first comprises a group of pseudo-operations that complement and augment the 8080/8085 instruction set. These pseudo-operations are accessible only from assembly language programs and resemble assembler commands. The second category consists of a set of routines that work on variable-length operands; the integer and decimal arithmetic and string-handling capabilities of the FSP are based on this foundation.

The routines in the FSP Machine do not return messages in the event of an error; they are optimized for speed of execution.

String Handling (FSLSTR.LIB)

This library contains routines for manipulating strings and for the processing of character input and output. The string manipulation routines fall into two groups. The first is character-oriented, while the second handles groups of string variables. In both groups, there are functions necessary for retrieval and manipulation of data.

The input scanner recognizes simple symbols like alphabetic names, numbers, boundaries, and gaps in input strings. The routines work with one of the user-defined tables and returns as a result an operand that can be utilized by the other string manipulation routines.

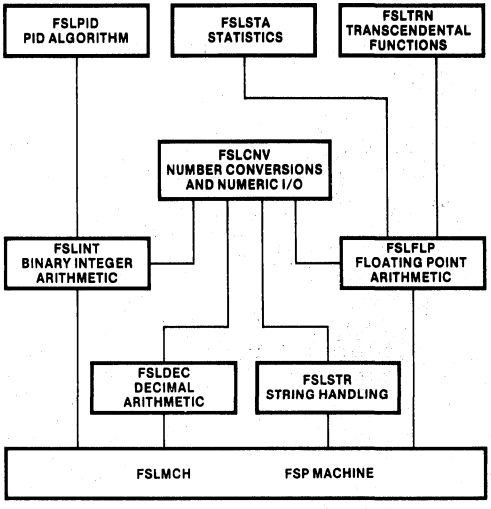


Figure 1. Hierarchy of FSP Modules

The output formatter arranges data into a form required for further manipulation. For example, it permits:

- Copying a character from the input string into the output buffer
- Including a literal in a given position in the output string
- Collapsing leading blanks
- Inserting the sign (i.e., of a number) in a given position in the output string

Decimal Arithmetic (FSLDEC.LIB)

The routines available in this library operate directly on signed decimal numbers without converting them internally into binary. Only integer operands are permitted. These are represented as ASCII strings up to 32 characters long. Addition, subtraction, and multiplication are accomplished by means of one operation each. For division there are two functions; one to calculate the integer quotient, one to calculate a remainder. Additional functions allow for negation, absolute value, and comparison of operands. Also available are utilities to limit the length of operands and to scale decimal variables.

Integer Arithmetic (FSLINT.LIB)

In this library there are routines for unsigned (8- and 16-bit) and signed (8-, 16-, and 32-bit) decimal arithmetic. For each of the fundamental operations—addition, subtraction, multiplication and division, as well as for the comparison of two integer operands—there is one routine. Additional functions provide for manipulation of the signs of operands and conversion of operands between different internal storage formats.

For signed integer operands, the following formats are possible:

FORMAT	RANGE
1 byte, including sign	-2^7 to 2^7-1
2 bytes, including sign	-2^{15} to $2^{15}-1$
4 bytes, including sign	-2^{31} to $2^{31}-1$
8 bytes, including sign	-2^{63} to $2^{63}-1$

For unsigned operands, there are three possible formats:

1 byte, without sign	0 to 2^6-1
2 bytes, without sign	0 to $1^{16}-1$
4 bytes, without sign	0 to $2^{32}-1$

The 8-byte signed format and the 4-byte unsigned format are limited to retrieval of results.

Floating-Point (FSLFLP.LIB)

The routines in this library provide an extensive range of floating-point arithmetic functions. In addition to addition, subtraction, multiplication, and division, there is a module function, a square root function and a routine to compare two floating point operands. Floating-point operands can be represented in any of four formats:

FORMAT	STORAGE	MANTISSA	EXPONENT
Single Precision	4 Byte	24 bits	8 bits
Extended Single Precision	6 Byte	32 bits	15 bits
Double Precision	8 Byte	53 bits	11 bits
Extended Double Precision	10 Byte	64 bits	15 bits

Additional functions provide for truncation, rounding and conversion of floating-point operands from one of the above formats to another.

Conversion and I/O (FSLCNV.LIB)

This library consists of routines for input and output of floating-point numbers and for the conversion of numeric data between different internal formats. There is an input scanner to read numeric data; there is an output formatter that writes output data to a buffer. There are various possible formats for input and output data; numeric data are handled as ASCII strings, as follows:

- Integers (single, decimal mantissa)
- Scaled integers (sign, decimal mantissa, decimal point)
- Floating point numbers (sign, decimal mantissa, decimal point, decimal exponent)

There are also other routines necessary for conversion of binary, decimal and floating point numbers. By means of the transformation of decimal into floating-point numbers and vice-versa, single and double precision are achieved.

Statistics (FSLSTA.LIB)

This library makes available routines that provide elementary statistical functions. The calculation of means, variance, and standard deviation employs one-dimensional arrays of data, whereas the calculation of covariance and correlations presumes two-dimensional arrays. In both cases, data are supplied as single precision floating-point numbers; the statistical routines make use of the floating-point library. Results of calculations are returned as single precision floating point numbers.

Process Control (FSLPID.LIB)

The routines in this library support digital process control using the 8080/8085. The PID algorithm

$$M(t) = B + \frac{100}{P}\left(E(t) + \frac{1}{R}\int_{to}^{t} E(s)\,ds + D\frac{dE}{dt}\right)$$

calculates an output signal M(t) as a function of an input signal E(t). The input quantity is a measure of the deviation of a controlled variable from a set point. The parameters B, P, R, and D are supplied by the user. The implementation is such that the user can select to exclude or include any combination of the terms in the PID equation (i.e., the proportional term, the integral term, the derivative term). There are also routines to initialize the control function and to change the measurement interval.

Transcendental Functions (FSLTRN.LIB)

This library includes routines to calculate elementary mathematical functions in single, extended single, double, and extended double precision. The following functions are available:

- sin, cos, tan
- exp, ln, $\log_{10}$, y^x
- sinh, cosh, tanh
- arc sin, arc cos, arc tan, arc tan (y/x)

SPECIFICATIONS

DEVELOPMENT ENVIRONMENT

Required Hardware:

Intellec Microcomputer Development System
—Model 800
—Series II Model 220, 230, or 240

64KB of Memory
—Single or double density diskette drive

System Console
—Intel or Non-Intel CRT

Required Software:

ISIS-II Diskette Operating System

LINK, LOC Utilities

ASM80, FORTRAN-80, PL/M-80

DOCUMENTATION PACKAGE

8080/8085 Fundamental Support Package (FSP) Reference and Operating Instructions for ISIS-II Users Order Number 9800887

Shipping Media

Flexible Diskettes
—Single and double density

ORDERING INFORMATION:

Product Code	Description
MDS-318*	8080/8085 Fundamental Support Package (FSP)

*"*MDS*" is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.

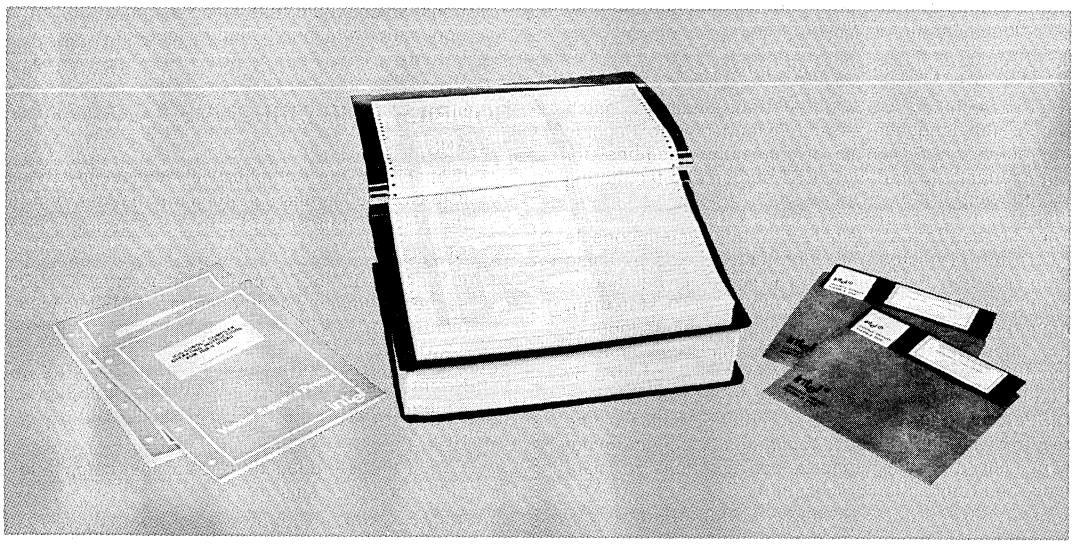
iCIS-COBOL SOFTWARE PACKAGE

- **Meets and Exceeds Minimum ANSI Level 1 Standard for COBOL (X3.23-1974)**
- **Runs Under ISIS-II on Intellec® or Intellec® Series II Microcomputer Development Systems**
- **Compiler Compiles COBOL Source Programs into an Intermediate Code Which is Optimized for Speed and Memory Space**
- **Includes Execution Run-Time Interpreter and an Interactive Debugger**
- **Powerful Extensions for Interactive Programming**
- **Can Link/Call Routines Written in PL/M 80, FORTRAN 80 and 8080/8085 Assembly Language**
- **FORMS Utility Program Allows the User to Design and Test CRT Screen Format Input by Generating COBOL Source Code for the Data Descriptions Defining that CRT Screen Format**
- **Compile-Time Option Available to Flag Any Non-ANSI Standard Features for Portability**
- **Tested Using U.S. Dept. of Navy COBOL Validation System**

iCIS-COBOL, an acronym for Intel's Compact Interactive Standard COBOL, is a package designed to provide a powerful interactive business language to users of Intel's Intellec and Intellec Series II Microcomputer Development Systems. iCIS-COBOL contains the most relevant parts of the ANSI 74 standard plus extra extensions to make this product especially useful to Intellec users. The compiler provides a feature to optionally disallow the iCIS-COBOL extensions and rigidly enforce the ANSI 74 specification. This will prove beneficial to users who may need to port COBOL programs from the Intellec system to any other ANS Level 1 COBOL compiler.

iCIS-COBOL Compiler generates object code for a COBOL "virtual machine." This code is designed for optimum representation of COBOL verbs and data types. The code generated is interpreted by a Run-Time System. This consists of an interpreter which emulates the COBOL virtual machine and interfaces to the ISIS-II operating system and the CRT.

After an application program has been tested and is ready for production use, it is possible to link it permanently to the Run-Time System to form a free-standing ISIS-II loadable program.



LANGUAGE FEATURES

COBOL consists of twelve different modules implemented either to Level 1 or Level 2 as defined in the ANSI specification X3.23. iCIS-COBOL includes the following modules implemented to Level 1:

Nucleus
Table Handling
Sequential I/O
Relative I/O
Indexed I/O
Library
Interprogram Communication

Extensions to ANSI Specification:

- *Advanced screen formatting and data entry facilities.* These include protected and unprotected data, cursor manipulation, and numeric vet.
- *Run time input of filenames.* The actual value of the external filename may be moved to a file identifier location prior to OPENing the file, avoiding the need for an external linking mechanism.
- *Line sequential files.* Variable length records separated by carriage return/line feed saves space on disk and allows iCIS-COBOL programs to process files output by a text editor.
- *Hexadecimal literals.* These may be used to define control characters to output to special peripheral devices.
- *Rapid development facilities.* During development, compiled programs may be loaded directly by the Run-Time System "fast load" facility, thus avoiding the time otherwise spent in linking.
- *Interactive debugging.* Interactive debugging permits the setting of breakpoints, examination and modification of store, etc., at run time. Each COBOL statement is identified by a four-digit hexadecimal number.
- *Lower case.* This is permitted in COBOL words and comments, thus helping to produce easy to read documentation in the program.

INTERACTIVE CRT HANDLING

Intel has taken COBOL — traditionally a batch processing language — and extended it to become interactive. iCIS-COBOL offers many facilities for automatically formatting a CRT screen and facilitating input keying.

The user can format the screen of any system console (CRT) into protected and unprotected fields by using standard COBOL statements. The screen layout may be defined in the DATA DIVISION. An ACCEPT statement nominates a record description which permits input to the character positions corresponding to variables identified by data-names. These may be separated by FILLERs to position them on the screen. Conversely, a DISPLAY outputs only from non-FILLER fields in the record description which it nominates. The programmer can easily build up complex conversations for data entry and transaction processing.

When data is being keyed in, the operator has full cursor manipulation facilities, each variable acting as a tab stop. Non-numeric digits may not be keyed into fields defined as PIC 9. Finally, when the operator has checked that the data is correct, the RETURN key is pressed and processing continues.

SCREEN LAYOUT AND FORMAT FACILITIES	CURSOR CONTROL FACILITIES
Screen as a record description	HOME to the start of the first data field on the screen
FILLER	→ Forward space
REDEFINES	← Backward space
AT line:column	↓ Forward field
Character highlighting	↑ Backward field
Clear screen	C/R Release the screen of data
Numeric vet for PIC fields	L/F Left Fill numeric field
	(The actual keys used vary according to CRT keyboard)

FORMS UTILITY

A majority (up to 80%) of debugging time can be spent in designing, coding and testing the screen form input/output of a COBOL program. The FORMS utility included in the iCIS-COBOL package significantly reduces this debugging time.

Using the FORMS program, the user may:

- Store an image copy on disk of the form he has defined for subsequent use.
- Generate iCIS-COBOL source code for the data descriptions required to define the form just created. This may then be included in an iCIS-COBOL program using COPY.
- Choose to generate a checkout program which allows duplication of the many machine conversations which would take place during a run of the application which is being designed.

COMPILE TIME DIRECTIVES

- ANS

  If specified, the Compiler will accept only those iCIS-COBOL language statements that conform to the ANS 74 standard.
- RESEQ

  If specified, the Compiler generates COBOL sequence numbers, renumbering each line in increments of 10.
- NOINT

  No intermediate code file is output. The Compiler is, in effect, used for syntax checking only.
- NOLIST

  No list file is produced; used for fast compilation of "clean" programs.

- COPYLIST

  The contents of the file(s) nominated in COPY statements are listed.

- NOFORM

  No form feed or page headings are to be output by the Compiler in the list file.

- ERRLIST

  The listing is limited to those COBOL lines containing syntax errors together with the associated error message(s).

- INT (external-file-name)

  Specifies the file to which the intermediate code is to be directed.

- LIST (external-file-name)

  Specifies the file to which the listing is to be directed.

- FORM (integer)

  Specifies the number of COBOL lines per page of listing.

- NOECHO

  Error lines are echoed on the console unless this directive is specified.

BENEFITS

- Brings COBOL to Intellec Microcomputer Development Systems.
  - — COBOL is the industry standard high-level language for business-oriented applications.
  - — Most COBOL programs are self-documenting.
  - — Conversational verbs and phrases and common business terminology make COBOL easy to learn, use and maintain.
  - — More business and application programs are written in COBOL than any other language.
- Meets and exceeds ANS Level 1 COBOL standard.
  - — Assures portability to and from all computers supporting ANS Level 1 COBOL.
  - — Extensive testing and validation using U.S. NAVY COBOL VALIDATION SYSTEM assures functionality for all Level 1 features.
- iCIS-COBOL software package provides an easy to use, efficient and friendly environment for COBOL program development.
  - — CONFIGURATOR program allows the user to reconfigure the software for any non-standard, non-Intel CRT.
  - — Interactive debugger provides features aimed at a CRT based system (rather than batch-oriented).
  - — FORMS utility program reduces total program development time by 30%.
  - — All iCIS-COBOL utilities make use of CRT cursor control.
- Adds value to an Intellec development system.
  - — COBOL applications programs developed using iCIS-COBOL software package will increase utilization of Intellec development systems.

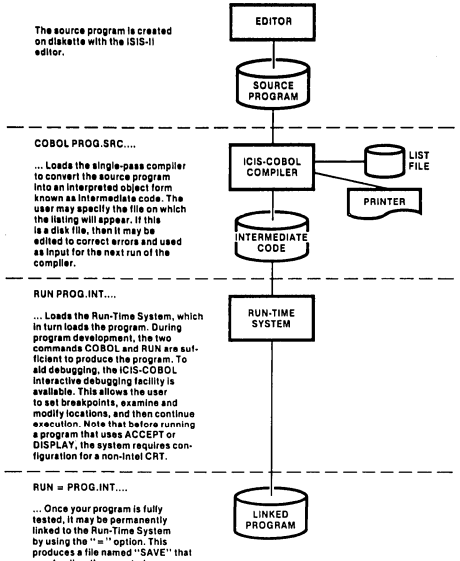


Figure 1. Program Development Cycle

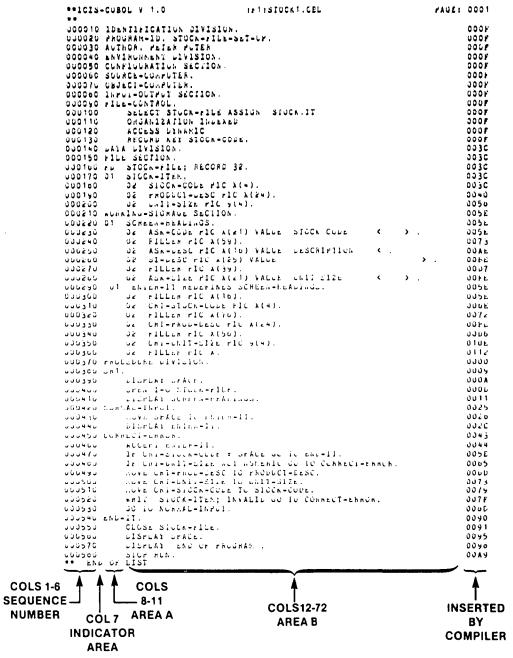


Figure 2. Sample Program Listing Showing Source Format

SPECIFICATIONS

Operating Environment

Required Hardware:

Intellec Microcomputer Development System
— Model 800
— Series II Model 220, Model 230

48KB of Memory

Dual Diskette Drives
— Single or Double Density

System Console
— Intel or non-Intel CRT

Recommended Hardware:

64KB of Memory

Double Density Dual Diskette Drives

Optional Hardware:

Line Printer

Required Software:

ISIS-II Diskette Operating System
— Single or Double Density

Optional Software:

ISIS-II CREDIT (CRT-Based Text Editor)

Documentation Package

iCIS-COBOL Language Reference Manual (9800927-01)

iCIS-COBOL Compiler Operating Instructions for ISIS-II Users (9800928-01)

iCIS-COBOL Pocket Reference (9800929-01)

Shipping Media

Flexible Diskettes
— Single and Double Density

ORDERING INFORMATION:

Product Code	Description
MDS-380*	iCIS-COBOL Software Package

Requires software license

*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Science Corporation.

PL/M 80
HIGH LEVEL PROGRAMMING LANGUAGE
INTELLEC® RESIDENT COMPILER

- **Provides Resident Operation on Intellec® Microcomputer Development System and Intellec® Series II Microcomputer Development Systems**
- **Produces Relocatable and Linkable Object Code**
- **Sophisticated Code Optimization Reduces Application Memory Requirements**
- **Speeds Project Completion with Increased Programmer Productivity**
- **Cuts Software Development and Maintenance Costs**
- **Improves Product Reliability with Simplified Language and Consequent Error Reduction**
- **Eases Enhancement as System Capabilities Expand**

The PL/M 80 High Level Programming Language Intellec Resident Compiler is an advanced, high level programming language for Intel 8080 and 8085 microprocessors, iSBC-80 OEM computer systems, and Intellec microcomputer development systems. PL/M has been substantially enhanced since its introduction in 1973 and has become one of the most effective and powerful microprocessor systems implementation tools available. It is easy to learn, facilitates rapid program development and debugging, and significantly reduces maintenance costs. PL/M is an algorithmic language in which program statements naturally express the algorithm to be programmed, thus freeing programmers to concentrate on system development rather than assembly language details (such as register allocation, meanings of assembler mnemonics, etc.). The PL/M compiler efficiently converts free-form PL/M programs into equivalent 8080/8085 instructions. Substantially fewer PL/M statements are necessary for a given application than would be using assembly language or machine code. Since PL/M programs are problem oriented and thus more compact, programming in PL/M results in a high degree of productivity during development efforts, resulting in significant cost reduction in software development and maintenance for the user.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Intel, INTEL, INTELLEC, MCS, im, ICS, ICE, UPI, BXP, iSBC, iSBX, iNSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

FUNCTIONAL DESCRIPTION

The PL/M compiler is an efficient multiphase compiler that accepts source programs, translates them into object code, and produces requested listings. After compilation, the object program may be first linked to other modules, then located to a specific area of memory, and finally executed. The diagram shown in Figure 1 illustrates a program development cycle where the program consists of three modules: PL/M, FORTRAN, and assembly language. A typical PL/M compiler procedure is shown in Table 1.

Features

Major features of the Intel PL/M 80 compiler and programming language include:

Resident Operation — on Intellec microcomputer development systems eliminates the need for a large in-house computer or costly timesharing system.

Object Code Generation — of relocatable and linkable object codes permits PL/M program development and debugging in small modules, which may be easily linked with other modules and/or library routines to form a complete application.

Extensive Code Optimization — including compile time arithmetic, constant subscript resolution, and common subexpression elimination, results in generation of short, efficient CPU instruction sequences.

Symbolic Debugging — fully supported in the PL/M compiler and ICE-85 in-circuit emulators.

Compile Time Options — includes general listing format commands, symbol table listing, cross reference listing, and "innerlist" of generated assembly language instructions.

Block Structure — aids in utilization of structured programming techniques.

Access — provided by high level PL/M statements to hardware resources (interrupt systems, absolute addresses, CPU input/output ports).

Data Definition — enables complex data structures to be defined at a high level.

Re-entrant Procedures — may be specified as a user option.

Benefits

PL/M is designed to be an efficient, cost-effective solution to the special requirements of microcomputer software development as illustrated by the following benefits of PL/M use:

Low Learning Effort — even for the novice programmer, because PL/M is easy to learn.

Earlier Project Completion — on critical projects, because PL/M substantially increases programmmer productivity while reducing program development time.

Lower Development Cost — because increased programmer productivity requiring less programming resources for a given function translates into lower software development costs.

Increased Reliability — because of PL/M's use of simple statements in the program algorithm, which are easier to correct and thus substantially reduce the risk of costly errors in systems that have already reached full production status.

Easier Enhancement and Maintenance — because programs written in PL/M are easier to read and easier to understand than assembly language, and thus are easier to enhance and maintain as system capabilities expand and future products are developed.

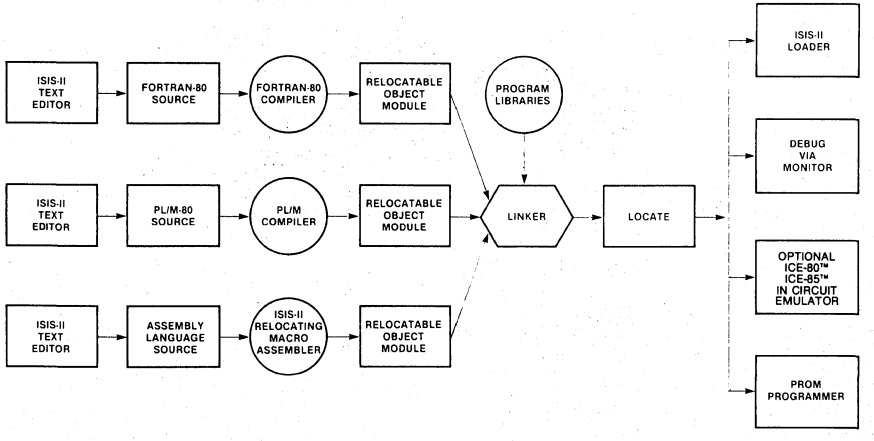


Figure 1. Program Development Cycle Block Diagram

Simpler Project Development — because the Intellec microcomputer development system with resident PL/M 80 is all that is needed for developing and debugging software for 8080 and 8085 microcomputers, and the use of expensive (and remote) timesharing or large computers is consequently not required.

```
                                    $OBJECT(:F1:FACT.OB2)
                                    $DEBUG
                                    $XREF
                                    $TITLE('FACTORIAL GENERATOR — PROCEDURE')
                                    $PAGEWIDTH(80)

 1                                  FACT:
                                    DO;

 2    1                             DECLARE NUMCH BYTE PUBLIC;

 3    1                             FACTORIAL: PROCEDURE (NUM,PTR) PUBLIC;
 4    2                                 DECLARE NUM BYTE, PTR ADDRESS;
 5    2                                 DECLARE DIGITS BASED PTR (161) BYTE;
 6    2                                 DECLARE (I,C,M) BYTE;

 7    2                                 NUMCH = 1; DIGITS(1) = 1;
 9    2                                 DO M = 1 TO NUM;
10    3                                     C = 0;
11    3                                     DO I = 1 TO NUMCH;
12    4                                         DIGITS(I) = DIGITS(I)*M + C;
13    4                                         C = DIGITS(I)/10;
14    4                                         DIGITS(I) = DIGITS(I) — 10*C;
15    4                                     END;

16    3                                     IF C<>0 THEN
17    3                                     DO;
18    4                                         NUMCH = NUMCH + 1; DIGITS(NUMCH) = C;
20    4                                         C = DIGITS(NUMCH)/10;
21    4                                         DIGITS(NUMCH) = DIGITS(NUMCH) — 10*C;
22    4                                     END
                                        END;

24    2                             END FACTORIAL;

25    1                             END;
```

Table 1. PL/M-80 Compiler Sample Factorial Generator Procedure

SPECIFICATIONS

Operating Environment

Required Hardware

Intellec microcomputer development system
65K bytes of memory
Dual diskette drives
System console — teletype

Optional Hardware

CRT as system console
Line printer

Required Software — ISIS-II diskette operating system

Shipping Media

Diskette

Reference Manuals

980026 — PL/M 80 Programming Manual (SUPPLIED)

9800300 — ISIS-II PL/M 80 Compiler Operator's Manual (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Requires software license

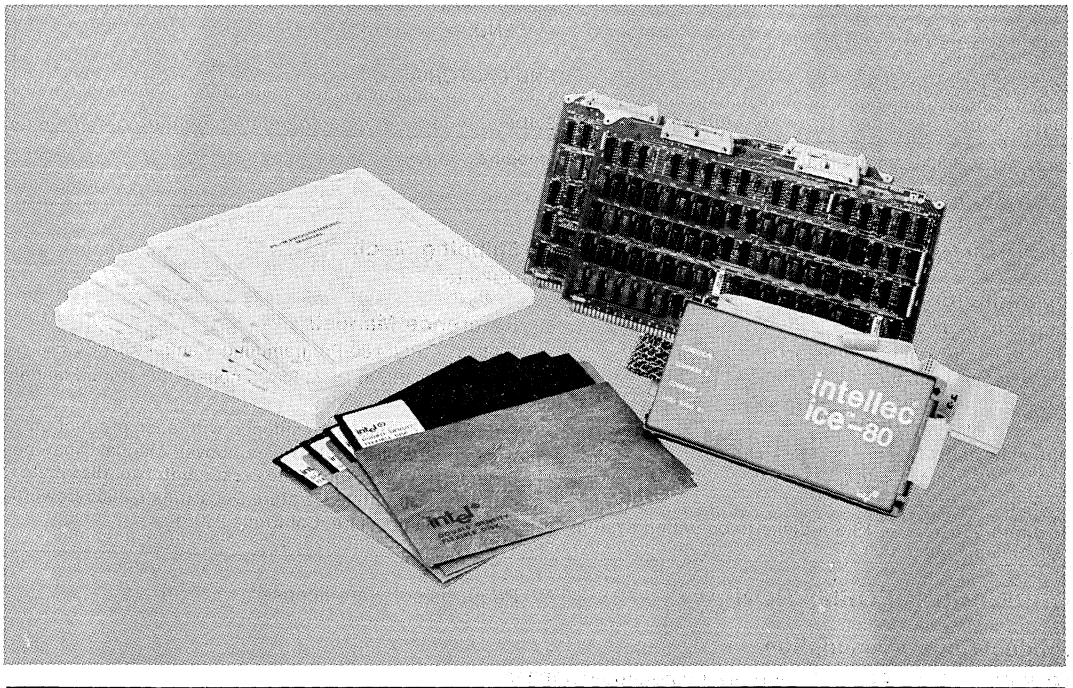
Product Code	Description
MDS-PLM*	PL/M 80 High level language compiler

*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

ICE-80™
8080 IN-CIRCUIT EMULATOR

- **Connects Intellec® System to User Configured System Via an External Cable and 40-pin Plug, Replacing the User System 8080**
- **Allows Real-Time (2 MHz) Emulation of User System 8080**
- **Shares Intellec® RAM, ROM, and PROM Memory and Intellec® I/O Facilities with User System**
- **Checks for Up to Three Hardware and Four Software Break Conditions**
- **Offers Full Symbolic Debugging Capabilities**
- **Eliminates Need for Extraneous Debugging Tools Residing in User System**
- **Provides Address, Data, and 8080 Status Information on Last 44 Machine Cycles Emulated**
- **Provides Capability to Examine and Alter CPU Registers, Main Memory, Pin, and Flag Values**
- **Integrates Hardware and Software Development Efforts**
- **Available in Diskette or Paper Tape Versions**

The Intellec ICE-80 8080 In-Circuit Emulator is an Intellec resident module designed to interface with any user configured 8080 system. With ICE-80 as a replacement for a prototype system 8080, the designer may emulate the system's 8080 in real time, single step the system's program, and substitute Intellec memory and I/O for user system equivalents. Powerful Intellec debug functions are extended into the user system. For the first time the designer may examine and modify his system with symbolic references instead of absolute values.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{i}m, ICS, ICE, UPI, BXP, iSBC, iSBX, iNSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

FUNCTIONAL DESCRIPTION

Integrated Hardware/Software Development

Use of the ICE-80 module enables the system integration phase, which can be so costly and frustrating when attempting to mesh completed hardware and software products, to become a convenient two-way debug tool when begun early in the design cycle. The user prototype need consist of no more than an 8080 CPU socket and a user bus to begin integration of software and hardware development efforts. With the ICE-80 mapping capabilities, system resources may be accessed for missing prototype hardware. Hardware designs may be tested using system software to drive the final product. A functional block diagram of the ICE-80 module is shown in Figure 1.

Symbolic Debugging Capability

ICE-80 provides for user-defined symbolic references to program memory addresses and data. Symbols may be substituted for numeric values in any of the ICE-80 commands. The user is thus relieved from looking up addresses of variables or program subroutines.

Symbol Table — The user symbol table generated along with the object file during a PL/M-80 compilation or a MAC80 or resident assembly, is loaded to memory along with the user program to be emulated. The user may add to this symbol table any additional symbolic values for memory addresses, constants, or variables found useful during system debugging. By referring to symbolic memory addresses, the user may be assured of examining, changing, or breaking at the intended location.

Symbolic Reference — ICE-80 provides symbolic definition of all 8080 registers, flags, and selected pins. The following symbolic references are also provided for user convenience: TIMER, a 16-bit register containing the number of ϕ_2 clock pulses elapsed during emulation; ADDRESS, the address of the last instruction emulated; INTERRUPTENABLED, the user 8080 interrupt mechanism status; and UPPERLIMIT, the highest RAM address occupied by user memory.

Debug Capability Inside User System

ICE-80 provides for user debugging of full prototype or production systems without introducing extraneous hardware or software test tools. ICE-80 connects to the user system through the socket provided for the user 8080 in the user system (See Figure 2). Intellec memory is used for the execution of the ICE-80 software, while I/O provides the user with the ability to communicate with ICE-80 and receive information on the generation of the user system. A sample ICE-80 debug session is shown in Figure 3.

I/O Mapping and Memory

Memory and I/O for the user system may be resident in the user system or "borrowed" from the Intellec system through ICE-80's mapping capability.

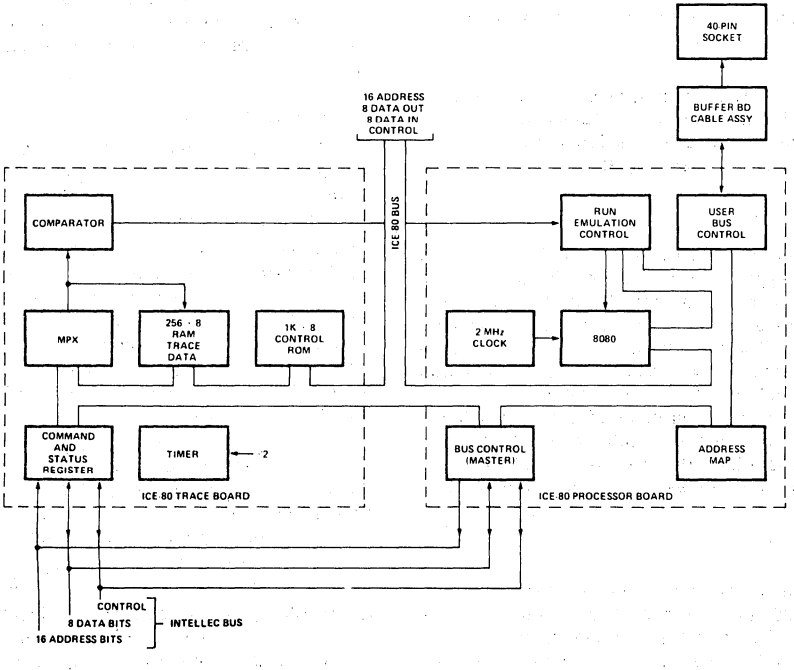


Figure 1. Functional Block Diagram of ICE-80 Module

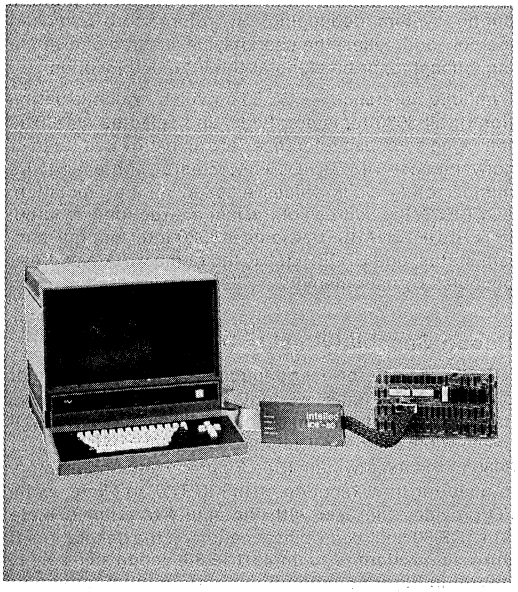


Figure 2. ICE-80 Module Installed in User System

Memory Blocking — ICE-80 separates user memory into 16 4K blocks. User I/O is divided into 16 16-port blocks. Each block of memory or I/O may be defined independently. The user may assign system equivalents to take the place of devices not yet designed for the user system during prototyping. In addition, memory or I/O may be accessed in place of user system devices during prototype or production checkout.

Error Messages — The user may also designate a block of memory or I/O as nonexistent. ICE-80 issues error messages when memory or I/O designated as nonexisting is accessed by the user program.

Real-Time Trace

ICE-80 captures valuable trace information while the user is executing programs in real time. The 8080 status, the user memory or port addressed, and the data read or written (snap data), is stored for the last 44 machine cycles executed. This provides ample data for determining how the user system was reacting prior to emulation break. It is available whether the break was user initiated or the result of an error condition. For detailed information on the actions of CPU registers, flags, or other system operations, the user may operate in single or multiple step sequences tailored to system debug needs.

Hardware

The heart of the ICE-80 is a microcomputer system utilizing Intel's 8080 microprocessor as its nucleus. This system communicates with the Intellec host processor via I/O commands. Host processor commands and ICE-80 status are interchanged through registers on the ICE-80 trace board. ICE-80 and the system also communicate through a control block resident in the Intellec main memory, which contains detailed configuration and status information transmitted at an emulation break. ICE-80 hardware consists of two PC boards — the processor and trace boards residing in the Intellec chassis — and a 6-foot cable interfacing to the user system. The trace and processor boards communicate with the system on the bus, and also with each other on a separate ICE-80 bus. ICE-80 connects to the user system through a cable that plugs directly into the socket provided for the user's 8080.

Trace Board

The trace board talks to the system as a peripheral device. It receives commands to ICE-80 and returns ICE-80 responses. While ICE-80 is executing the user program, the trace board collects data for each machine cycle emulated (snap data). The information is continuously stored in high-speed bipolar memory.

Breakpoint — The trace board also contains two 24-bit hardware breakpoint registers which can be loaded by the user. While in emulation mode, a hardware comparitor is constantly monitoring address and status lines for a match to terminate an emulation. A user probe is also available for attachment to any user signal. When this signal goes true a break condition is recognized.

Interrogation — The trace board signals the processor board when a command to ICE-80 or break condition has been detected. The ICE-80 CPU then sends data stored on the trace board to the control block in memory. Snap data, along with information on 8080 registers and pin status and the reason for the emulation break, are then available for access during interrogation mode. Error conditions, if present, are transmitted and automatically displayed for the user.

Processor Board

An 8080 CPU resides on the processor board. During emulation it executes instructions from the user's program. At all other times it executes instructions from the control program in the trace module's ROM.

Timing — The processor board contains an internal clock generator to provide clocks to the user emulation CPU at 2 MHz. The CPU can alternately be driven by a clock derived from user system signal lines. The clock source is selected by a jumper option on the board. A timer on the trace board counts the ϕ_2 clock pulses during emulation and can provide the user with the exact timing of the emulation.

On/Off Control — The processor board turns on an emulation when ICE-80 has received a run command from the system. It terminates emulation when a break condition is detected on the trace board, or the user's program attempts to access memory or I/O ports designated as nonexistent in the user system, or the user 8080 is inactive for a quarter of a second.

Status Storage — The address map located on the processor board stores the assigned location of each user memory or I/O block. During emulation the processor board determines whether to send/receive information

```
ISIS 8080 MACROASSEMBLER, V1.0                PAGE 1

                ;USER PROGRAM TO OUTPUT A SERIES OF
                ;CHARACTERS TO SDK-80 CONSOLE DEVICE
                ;
1320                    ORG     1320H
01E3            C0      EQU     1E3H    ;SDK-80 CONSOLE OUT DRIVER
                ;
1320 0601       START:  MVI     B,1     ;SET UP B VALUE
1322 3A3613             LDA     DAT1    ;LOAD A WITH DAT1 VALUE
135 4F          LOOP:   MDV     C,A
1326 CDE301             CALL    C0      ;SEND C VALUE TO CONSOLE
1329 79                 MOV     A,C     ;RESTORE A
132A 93                 SBB     B       ;SUBTRACT B FROM A
132B 323713             STA     RSLT    ;STORE RESULT IN RSLT
132E FE40               CPI     40H     ;LAST VALUE TO PRINT
1330 C22513             JNZ     LOOP    ;LOOP AGAIN IF A>40H
1333 C32013             JMP     START   ;ELSE RESTART WHOLE PROCEDURE
                ;
1336 5A         DAT1:   DB      5AH
1337            RSLT:   DS      1
0000                    END

ISIS, V1.0      INITIAL ICE-80 SESSION
-ICE80          (Note: The SDK-80 Monitor has already been used to initialize the SDK-80 Board)
ISIS ICE-80, V1.0
(1) **XFORM MEMORY 0 TO 1 U
    *XFORM IO 0FH U
(2) *LOAD PROG. HEX
    ERR = 067
    STAT = 11H TYPE = 06H CMND = 07H ADDR = 1320H GOOD = 06H BAD = 04H
    *CHANGE MEMORY 1321H = FFH
    ERR = 067
    STAT = 11H TYPE = 06H CMND = 07H ADDR = 1321H GOOD = FFH BAD = FDH
    *LOAD PROG. HEX
(3) *GO FROM START UNTIL RSLT WRITTEN
    EMULATION BEGUN
(4) ERR = 067
    STAT = 11H TYPE = 07H CMND = 02H
(5) *DISPLAY CYCLES 5

    STAT = A2H ADDR = 1326H DATA = CDH
    STAT = 82H ADDR = 1327H DATA = E3H
    STAT = 82H ADDR = 1328H DATA = 01H
    STAT = 04H ADDR = FFFFH DATA = 13H
    STAT = 04H ADDR = FFFEH DATA = 29H
(6) *CHANGE DOUBLE REGISTER SP = 13FFH
    *BASE HEX
    *EQUATE STOP = 1333H
(7) *GO FROM START UNTIL STOP EXECUTED THEN DUMP
    EMULATION BEGUN
    B = 01H C = 41H D = 00H E = 00H H = 00H L = 00H F = 56H A = 40H P = 1320H * = 1333H S = 13FFH
    EMULATION TERMINATED AT 1333H
(8) *EXIT
    *FFFF
```

Notes

1. Set up user memory and I/O. The program is set up to execute in block 1 (1000H–1FFFH) of user memory, and requires access to the SDK-80 monitor (block 0) and I/O ports in block 0FH. Both ports and memory are defined as available to the user system. All other memory and I/O is initialized by ICE-80 as nonexistent (guarded).

2. A load command generates an error. The type and command numbers indicate that a data mismatch occurred on a write to memory command. The data to be written to address 1320H should have been 06H. When ICE-80 read the data after writing it, a 04H was detected. A change command to a different memory address hints that bit 1 does not go to 1 anywhere in this memory block. Examination indicates that a pin was shorted on the RAM located at 1300H–13FFH in the prototype system. The problem is fixed and a subsequent load succeeds.

3. A real-time emulation is begun. The program is executed from 'START' (1320H) and continues until 'RSLT' is written [in location 1328H, the contents of the accumulator is stored in (written into) 'RSLT'].

4. An error condition results: TYPE 07, CMND 02 indicate the program accessed is a guarded area.

5. The last 5 machine cycles executed are displayed. The last instruction executed was a call (CDH). The fourth and fifth cycles are a push operation (designated by status 04H) to store the program counter before executing the call. The stack pointer was not initialized in the program and is accessing memory location FFFFH.

6. After making a note to initialize the stack pointer in the next assembly, a temporary fix is effected by setting the stack pointer to the top of user available memory.

7. After setting the base for displays to hex and adding the symbol 'STOP' to the symbol table, emulation is started which will terminate when the instruction at 1333H ('STOP') is executed. When emulation terminates, a dump of the contents of user 8080 registers is requested. One can see that the value of the accumulator is set at 40H, the stack pointer is set at 13FFH, the last address executed (*) is 1333H, and the program counter has been set to 1320H.

8. Exit returns control to the MDS monitor.

Figure 3. Sample ICE-80 Debug Session

on the Intellec or user bus by consulting the address map. The processor board allows the ICE-80 CPU to gain access to the bus as a master to "borrow" Intellec facilities. At an emulation break, the processor board stores the status of specified 8080 input and output signals, disables all interaction with the user bus, and commands the trace board to send stored information to a control block in Intellec memory for access during interrogation mode.

Cable Card

The cable card is included for cable driving. It transmits address and data bus information to the user system through a 40-pin connector that plugs into the user system in the socket designed for the 8080 when enabled by the processor module's user bus control logic.

Software

The ICE-80 software driver is a RAM-based program providing easy to use English language commands for defining breakpoints, initiating emulation, and interrogating and altering the user system status recorded during emulation. ICE-80 commands are configured with a broad range of modifiers to provide the user with maximum flexibility in describing the operation to be performed. Listings of emulation commands, interrogation commands, and utility commands are provided in Table 1, Table 2, and Table 3, respectively.

Command	Operation
Go	Initiates real-time emulation and allows user to specify breakpoints, data retrieval, and conditions under which emulation should be reinitiated.
Step	Initiates emulation in single or multiple instruction increments. User may specify register dump or tailor diagnostic activity to his needs following each step, and define conditions under which stepping should continue.
Range	Delimits blocks of instructions for which register dump or tailored diagnostics are to occur.
Continue	Resumes real-time emulation.
Call	Emulates user system interrupt.

Table 1. ICE-80 Emulation Commands

Command	Operation
Base	Establishes mode of display for output data.
Display	Prints contents of memory, 8080 registers, input ports, 8080 flags, 8080 pins, snap data, symbol table, or other diagnostic data on list device. May also be used for base-to-base conversion, or for addition or subtraction in any base.
Change	Alters contents of memory, register, output port, or 8080 flag.
XFORM	Defines memory and I/O status.
Search	Looks through memory range for specified value.

Table 2. ICE-80 Interrogation Commands

Command	Operation
Load	Fetches user symbol table and object code from input device.
Save	Sends user symbol table and object code to output device.
Equate	Enters symbol name and value to user symbol table.
Fill	Fills memory range with specified value.
Move	Moves block of memory data to another area of memory.
Timeout	Enables/disables user CPU ¼ second wait state timeout.
List	Defines list device (diskette-based version only).
Exit	Returns program control to monitor.

Table 3. ICE-80 Utility Commands

SPECIFICATIONS

Paper Tape-Based Operating Environment

Required Hardware
Intellec system
System console
Reader device
Punch device
ICE-80 module

Required Software
System monitor

Diskette-Based Operating Environment

Required Hardware
Intellec system
32K bytes RAM memory
System console
Intellec diskette operating system
ICE-80 module

Required Software
System monitor
ISIS-II

System Clock

Crystal controlled 2.185 MHz ± 0.01%. May be replaced by user clock through jumper selection.

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 8.00 lb (3.64 kg)

Electrical Characteristics

DC Power Requirements

V_{CC} = +5V, ±5%
I_{CC} = 9.81A max; 6.90A typ
V_{DD} = +12V, ±5%
I_{DD} = 79 mA max; 45 mA typ
V_{BB} = −9V, ±5%
I_{BB} = 1 mA max; 1μA typ

Environmental Characteristics

Operating Temperature — 0°C to 40°C

Operating Humidity — Up to 95% relative humidity without condensation

Equipment Supplied

Printed circuit modules (2)
Interface cables and buffer board
ICE-80 software driver, paper tape version
(ICE-80 software driver, diskette-based version is supplied with diskette operating systems)
Operator's Manual

ORDERING INFORMATION

Part Number	Description
MDS-80-ICE*	8080 CPU in-circuit emulator, cable assembly and interactive software included

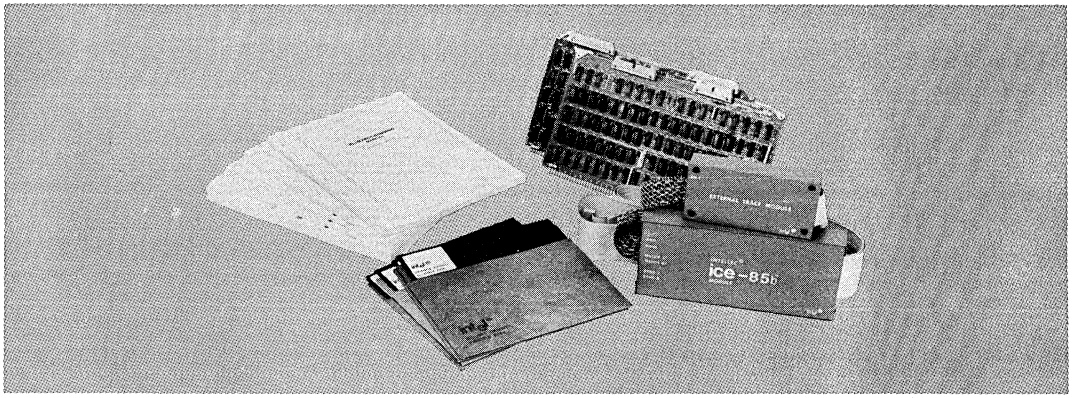
*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

ICE-85B™
MCS-85™ IN-CIRCUIT EMULATOR
WITH MULTI-ICE™ SOFTWARE

- **Connects the Intellec® system resources to the user-configured system via a 40-pin adaptor plug**
- **Executes user system software in real-time (5 MHz clock)**
- **Allows user-configured system to share Intellec® memory and I/O facilities**
- **Provides 1023 states of 8085 trace data**
- **Displays trace data from the user's 8085 in assembler mnemonics and allows personality groupings of data sampled by the external 18-channel trace module**
- **Offers full symbolic debugging capability for both assembly language and Intel's high-level compiler languages PL/M-80 and FORTRAN-80**
- **The Multi-ICE™ software provides:**
  - **—for two In-Circuit Emulators to operate simultaneously in a single Intellec Microcomputer Development System.**
  - **—support for ICE 85/85™, 85/49™, and 85/41A™ Emulator combinations**
  - **—enhanced software features: symbolic display of addresses, macro commands, compound commands, software synchronization of processes, and INCLUDE file capability.**

The ICE-85B™ module resides in the Intellec® Microcomputer Development System and interfaces to the user system's 8085. It provides the ability to examine and alter MCS-85™ registers, memory, flag values, interrupt bits and I/O ports. Using the ICE-85B module, the designer can execute prototype software in real-time or single-step mode and can substitute Intellec® system memory and I/O for user system equivalent. ICE capability can be extended to the rest of the user system peripheral circuitry by allowing the user to create and execute a library of user-defined peripheral chip analyzer routines.

Multi-ICE In-Circuit Emulator is a software product which allows two Intel In-Circuit Emulators to run simultaneously in a single Intellec Microcomputer Development System. Multi-ICE software used in lieu of the standard ICE software gives users full control of the two ICE modules for debugging of multi-processor systems.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{i}m, iCS, ICE, UPI, BXP, iSBC, INSITE, CREDIT RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix: e.g., iSBC-80.

©INTEL CORPORATION, 1980.

SYMBOLIC DEBUGGING CAPABILITY

ICE-85B allows the user to make symbolic references to I/O ports, memory addresses and data in his program. Symbols and PL/M-80 statement number may be substituted for numeric values in any of the ICE-85 commands. The user is relieved from looking up addresses of variables or program subroutines.

The user symbol table generated along with the object file during a PL/M-80 or FORTRAN-80 compilation or by the ISIS-II 8080/8085 Macro Assembler is loaded into the Intellec® System memory along with the user program which is to be emulated. The user may add to this symbol table any additional symbolic values for memory addresses, constants, or variables that are found useful during system debugging. By referring to symbolic memory addresses, the user can examine, change or break at the intended location.

ICE-85B provides symbolic definition of all 8085 registers, interrupt bits and flags. The following symbolic references are also provided for user convenience: TIMER, the low-order 16 bits of a register containing the number of 2 MHz clock pulses elapsed during emulation; HTIMER, the high-order 16 bits of the timer counter; PPC, the address of the last instruction emulated; BUFFERSIZE, the number of frames of valid trace data (between 0 and 1022).

PERSONALITY GROUPED DISPLAYS

Trace data in the 1023 by 42-channel real-time trace memory buffer is displayed in easy to read format. The user has the option to specify trace data displays in actual 8085 assembler instruction mnemonics. The data collected from the External Trace Module can be grouped and symbolically named according to user specifications and displayed in the appropriate number base designation. Simple ICE-85B commands allow the user to select any portion of the 42-bit trace buffer for immediate display.

MEMORY AND I/O MAPPING

Memory and I/O for the user system can be resident in the user system or "borrowed" from the Intellec® System through ICE-85B's mapping capability.

ICE-85B separates user memory into 32 2K blocks. Each block of memory can be defined independently. The user may assign Intellec® System equivalents to take the place of devices not yet designed for the user system during prototyping. In addition, Intellec® System memory or I/O can be accessed in place of suspect user system devices during prototyping or production checkout.

User ready synchronization—resource borrowing from the Intellec System is (at user option) independent of the user system; the user does not need to provide ready acknowledge when accessing resources mapped to the Intellec.

The user can also designate a block of memory or I/O as nonexistent. ICE-85B issues error messages when memory or I/O designated as nonexistent is accessed by the user program.

INTEGRATED HARDWARE/SOFTWARE DEVELOPMENT

The user prototype need consist of no more than an 8085 CPU socket and a user bus to begin integration of software and hardware development efforts. Through ICE-85B mapping capabilities, Intellec® System equivalents can be accessed for missing prototype hardware. Hardware designs can be tested using the system software which will drive the final product.

The system integration phase, which can be so costly when attempting to mesh completed hardware and software products, becomes a convenient two-way debug tool when begun early in the design cycle.

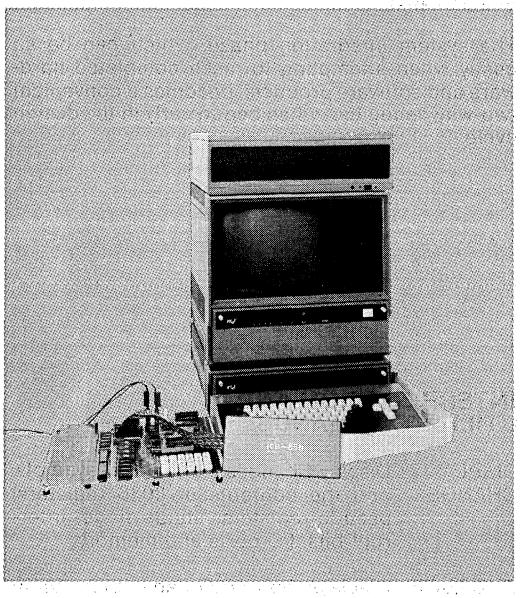
INTERROGATION AND UTILITY COMMANDS

DISPLAY/ CHANGE	Display/Changes the values of symbols and the contents of 8085 registers, pseudo-registers, status flags, interrupt bits, I/O ports and memory.
EVALUATE	Displays the value of an expression in the binary, octal, decimal or hexadecimal.
SEARCH	Searches user memory between locations in a user program for specified contents.
CALL	Emulates a procedure starting at a specified memory address in user memory.
ICALL	Executes a user-supplied procedure starting at a specified memory address in the Intellec® System memory.
EXECUTE	Saves emulated program registers and emulates a user-supplied subroutine to access peripheral chips in the user's system.

REAL TIME TRACE

ICE-85B captures valuable trace information from the emulating CPU and the External Trace Module while the user is executing programs in real time. The 8085 status, the user memory or port addressed, the data read or written, the serial data lines and data from 18 external signals, is stored for the last 1023 machine states executed (511 machine cycles). This provides ample data for determining how the user system was reacting prior to emulation break. It is available whether the break was user-initiated or the result of an error condition.

For detailed information on the actions of CPU registers, flags, or other system operations, the user may operate in single or multi-step sequences tailored to system debug needs.



EMULATION CONTROLS AND COMMANDS

GROUP — Defines into a symbolically named group, a channel or combination of channels from the 8085 Microprocessor and/or the External Trace Module.

GO — Initiates real-time emulation and controls emulation break conditions.

STEP — Initiates emulation in single instruction steps. User may specify the type and amount of information displayed following each step, and define conditions under which stepping should continue.

PRINT — Prints the user-specified portion of the trace memory to the selected list device.

EXTERNAL TRACE MODULE

TTL level signals from 18 points in the user system may be synchronously sampled by the External Trace Module and collected in ICE-85B's trace buffer. The signals can be collected from a single peripheral chip via the supplied 40-pin DIP clip or may be placed by the user on up to 18 separate signal nodes using the supplied 18 individual probe clips. These signals are included in the 42-channel breakpoint comparisons and clock qualifiers. Also, data from these 18 channels may be displayed in meaningful, user-defined groupings.

SYNCHRONOUS OPERATION WITH OTHER DESIGN AIDS

ICE-85B can be synchronized with other Intellec® design aids by means of two external synchronization lines. These lines are used to enable and disable ICE-85B trace data collection and to cause break conditions based on an external signal which may not be included in the ICE-85B breakpoint registers. In addition, ICE-85B can generate signals on these lines which may be used to control other design aids.

BREAK REGISTERS/ TRACE MEMORY

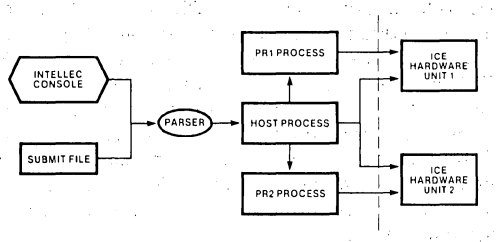
ICE-85B has two breakpoint registers which are used to break emulation, and two trace qualifier registers which are used to control the collection of trace data during emulation. Each register is 42 entries wide, one entry for each channel and each entry can take any one of the three values 0, 1 or "don't care."

The trace buffer, also 42 entries wide, collects data sampled from 24 8085 processor channels and 18 external channels sampled by the External Trace Module. The signals collected from the 8085 include address lines, data lines, status lines and serial input and output lines. The 18 channels extending from the External Trace Module synchronously sample and collect into the trace buffer any user-specified TTL compatible signal from the rest of the prototype system. "Break" and "trace qualification" may therefore occur as a result of a match of any combination of up to 42 channels of CPU and external circuitry signals.

MULTI-ICE™ OPERATION

Multi-ICE software is a debug tool which allows two ICE emulators to begin and stop in sequence. Once started, two ICE emulators emulate simultaneously and independently. Thus, Multi-ICE software permits the debugging of asynchronous or synchronous multi-processor systems.

A conceptual model for the Multi-ICE software can be illustrated with the following block diagram.



Block Diagram of Multi-ICE™ Operation

There are three processes in the Multi-ICE environment: the Host process and the two ICE processes to control the two ICE hardware modules. The processor for these three processes is the microcomputer in the Intellec Microcomputer Development System. Only the Host process is active when Multi-ICE software is invoked. The Parser interfaces with the console, receives commands from the console or from a file, translates them into intermediate code, and loads the code into the Host command code buffer or ICE command code buffers.

The Host process executes commands from its command code buffer using the execution software and hardware of the Host's current environment, either environment 1 or environment 2 (EN1 or EN2), as required. EN1 and EN2 are the operating environments of the two In-Circuit Emulators.

The user can change the execution environment (from EN1 to EN2 or vice versa) with the SWITCH command. Once the environment is selected, ICE operation is the same as with standard ICE software. In addition, the enhanced software capabilities are available to the user.

The two ICE processes (PR1 and PR2) execute commands from their command code buffers in their own environments (PR1 in EN1 and PR2 in EN2). The main functions of the two ICE execution processes are to control the operations of the two ICE hardware sets. The ACTIVATE command controls the execution of the ICE processes. Commands are passed on to each ICE unit to initiate the desired ICE functions.

The two ICE hardware units accept commands from the Host process or ICE processes. Once emulations start, the two ICE hardware sets will operate until a break condition is met or processing is interrupted by commands from the ICE execution processes.

Symbolic Display of Addresses

The user has the option of displaying a 16-bit address in the form of a symbol name or line number plus a hex number offset.

Macro Command

A macro is a set of commands which is given a name. Thus, a group of commands which is executed frequently may be defined as a macro. Each time the user wants to execute that group of commands, he may just invoke the macro by typing a colon followed by the macro name. Up to ten parameters may be passed to the macro.

Macro commands may be defined at the beginning of a debug session and then can be used throughout the whole session. If the user wants to save the macros for later use, he may use the PUT command to save the macro on diskette, or the user may edit the macro file off-line using the Intellec text editor. Later, the user may use the INCLUDE command to bring in the macro definition file that he created.

Example:

```
*DEFINE MACRO INITMEM     ;This macro clears the memory and then loads the programs.
.*SWITCH = EN1            ;Select environment 1 (ICE Module 1)
.*BYTE 0 TO 100=0         ;Initialize memory to 0.
.*LOAD:F1:DRIVER          ;Load user program into memory for ICE Module 1.
.*SWITCH = EN2            ;Select environment 2 (ICE Module 2)
.*LOAD:F1:DR2             ;Load user program into memory for ICE Module 2
.*EM                      ;End of Macro
*                         ;To execute this Macro, user types :INITMEM
```

Compound Command

Compound commands provide conditional execution of commands (IF Command) and execution of commands repeatedly until certain conditions are met (COUNT, REPEAT Commands).

Compound commands and Macro commands may be nested any number of times.

Example:

```
*DEFINE .I = 0             ;Define symbol .I to 0
*COUNT 100H                ;Repeat the following commands 100H times
.*IF .I AND 1 THEN         ;Check if .I is odd
..*BYT .I = .I             ;Fill the memory at location .I to value .I
..*END
.*.I = .I + 1              ;Increment .I by 1
.*END                      ;Command executes upon carriage-return after END
```

INCLUDE File Capability

The INCLUDE command causes input to be taken from the file specified until the end of the file is encountered, at which point, input continues to be

taken from the previous source. Nesting of INCLUDES is permitted. Since the command code file can be complex, the ability to edit offline becomes desirable. The INCLUDE command allows the user to pull in command code files and Macro commands created offline which can then be used for the particular debugging session.

Example:

```
*INCLUDE :F1:PROG1    ;Cause input to be taken
                       from file PROG1
*MAP 0 LENGTH 64K     ;Contents of the file
=USER                  PROG1 are listed on
                       screen as they are
                       executed.

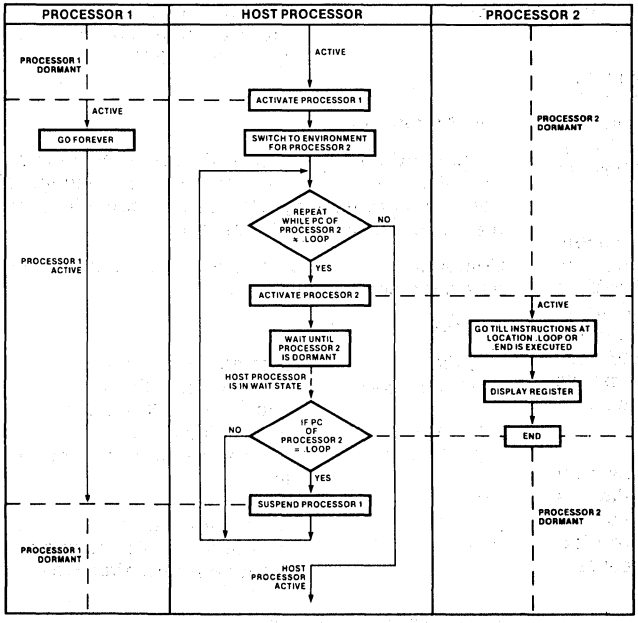
*MAP IO 0 TO FF
=USER
*SWITCH = EN2
*LOAD :F2:LED.HEX
*SWITCH = EN1         ;End of the file PROG1
*                     ;After the end of file is
                       reached, control is
                       returned to console.
```

Software Synchronization of Processes

Up to three processes (Host, PR1 and PR2) can be active simultaneously in the system. An ICE process can be activated (ACTIVATE), suspended (SUSPEND), killed (KILL), or continued (CONTINUE). The Host process can wait for other processes to become dormant before it becomes active again. Through these synchronization commands, the user can create a system test file off-line yet be able to synchronize the three processes when the actual system test is executed.

Example:

The capability of the software synchronization commands is demonstrated by the following example. The flowchart shows the synchronization requirements. The program steps show the actual implementation.




```
*ACTIVATE PR1          ;Activate PR1
.*GO FROM 800          ;Start ICE Module 1
.*END                  ;End of Activate block
PR1 EMULATION BEGUN
*SWI=EN2               ;Switch execution Environment to EN2
*REPEAT                ;Repeat the following block of
                        commands while PC is not equal to .Loop
.*WHILE PC < > .LOOP
.*ACT PR2              ;Activate PR2
..*GO TILL .LOOP OR .END ;Go till instruction at location .Loop
                          or at location .END is executed
..*REGISTER            ;Display the registers
..*END                 ;End of Activate block
.*WAIT PR2             ;Wait until PR2 is dormant
.*IF PC=.LOOP THEN
..*SUSPEND PR1
..*END                 ;End of IF block
.*END                  ;End of REPEAT block
*
```

Flowchart of the Example for Demonstrating Multi-ICE™ Synchronization Capability

SPECIFICATIONS

ICE-85B™ Operating Environment

Required Hardware:
- Intellec® Microcomputer Development System
  - (64K bytes RAM for Multi-ICE software)
  - (32K bytes RAM single ICE software)
  - System Console
  - Intellec® Diskette Operating System
  - ICE-85B Module

Required Software:
- System Monitor
- ISIS-II
- ICE-85B or Multi-ICE Software

Equipment Supplied

18-Channel External Trace Module
Printed Circuit Boards (2)
Interface Cable and Emulation Buffer Module
Operator's Manuals
ICE-85B Software
Multi-ICE Software
- Contains software that supports 85/85 Emulators, 85/49 Emulators and 85/41A Emulators

Emulation Clock

User's system clock or ICE-85B adaptor socket (10.0 MHz Crystal)

Physical Characteristics

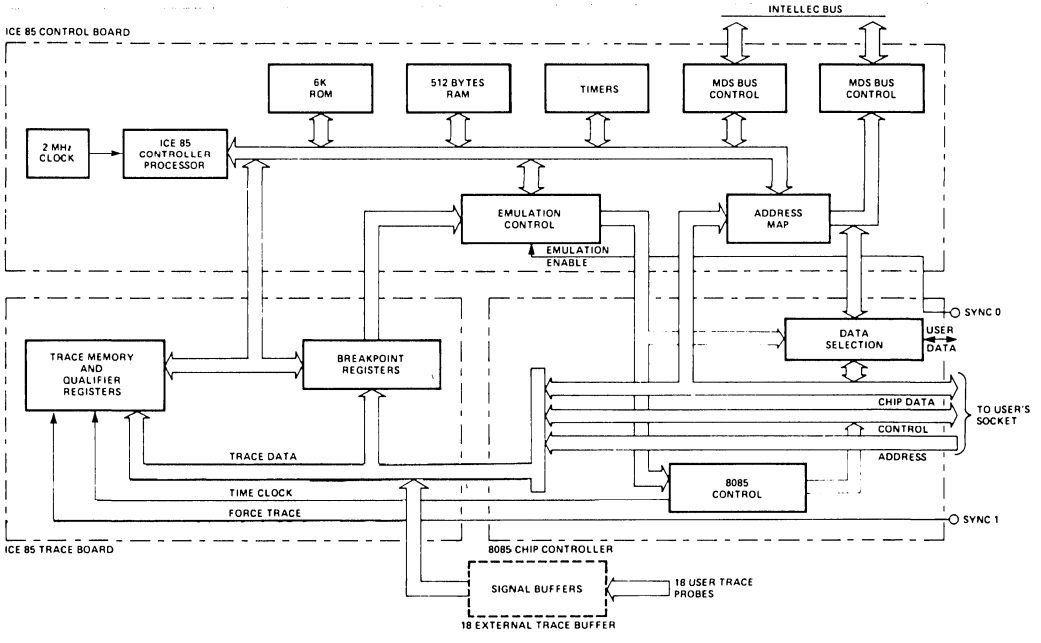
Printed Circuit Boards:
- Width: 12.00 in. (30.48 cm)
- Height: 6.75 in. (1715 cm)
- Depth: 0.50 in. (1.27 cm)
- Packaged Weight: 6.00 lb (2.73 kg)

Electrical Characteristics

DC Power:
- V_{CC} = + 5V ± 5%
- I_{CC} = 12A maximum; 10A typical
- V_{DD} = + 12V ± 5%
- I_{DD} = 80 mA maximum; 60 mA typical
- V_{BB} = − 10V ± 5%
- I_{BB} = 1 mA maximum; 10 μA typical

Environmental Characteristics

Operating Temperature: 0° to 40°C
Operating Humidity: Up to 95% relative humidity without condensation.



ICE-85B™ BLOCK DIAGRAM

Ordering Information

Part Number	Description
MDS*-85B-ICE	8085 CPU In-Circuit Emulator, 18-Channel External Trace Module and Multi-ICE software
MDS*-85U-ICE	Upgrade kit to convert ICE-85 or ICE-85A to ICE-85B functionality. Consists of Multi-ICE software and 5MHz Hardware

*"***MDS***" is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.

iAPX 86/88 Support Options

16

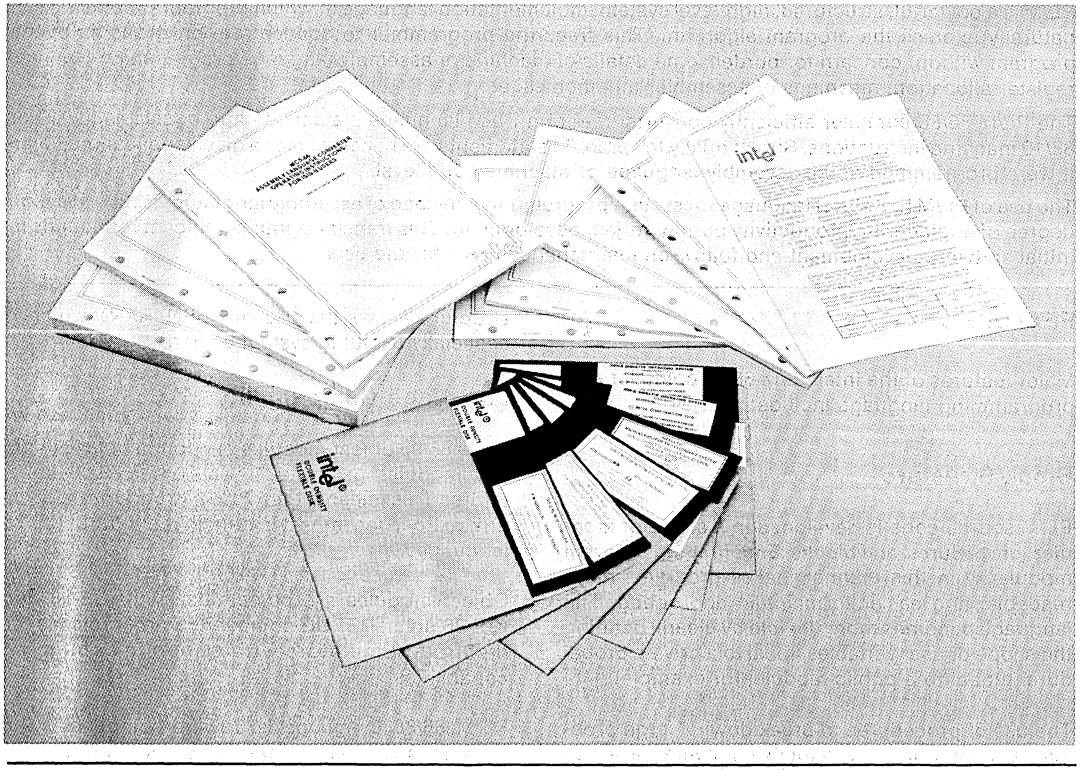
iAPX 86,88
SOFTWARE DEVELOPMENT PACKAGES FOR SERIES II

- **PL/M 86/88 High Level Programming Language**
- **ASM 86/88 Macro Assembler for iAPX 86,88 Assembly Language Programming**
- **LINK 86/88 and LOC 86/88 Linkage and Relocation Utilities**
- **CONV 86/88 Converter for Conversion of 8080/8085 Assembly Language Source Code to iAPX 86, 88 Assembly Language Source Code**
- **OH 86/88 Object-to-Hexadecimal Converter**
- **LIB 86/88 Library Manager**

The iAPX 86,88 Software Development Packages for Series II provide a set of software development tools for the iAPX 86/10 and iAPX 88/10 CPUs and the iSBC 86/12A single board computer. The packages operate under the ISIS-II operating system on Intellec Microcomputer Development Systems—Model 800 or Series II—thus minimizing requirements for additional hardware or training for Intel Microcomputer Development System users.

These packages permit 8080/8085 users to efficiently upgrade existing programs into iAPX 86/10 and 88/10 code from either 8080/8085 assembly language source code or PL/M 80 source code.

For the new Intel Microcomputer Development System user, the packages operating on an Intellec Series II, such as a Model 235, provide total iAPX 86,88 software development capability.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{l}m, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© INTEL CORPORATION, 1981.

APRIL 1981
9800757-04

PL/M 86/88 COMPILER FOR SERIES II

- **Language is Upward Compatible from PL/M 80, Assuring MCS-80/85™ Design Portability**
- **Supports 16-bit Signed Integer and 32-bit Floating Point Arithmetic in Accordance with IEEE Proposed Standard**
- **Easy-to-Learn, Block-Structured Language Encourages Program Modularity**
- **Produces Relocatable Object Code Which is Linkable to All Other 8086 Object Modules**
- **Supports Full Extended Addressing Features of the iAPX 86/10 and 88/10 Microprocessors (Up to 1 Mbyte)**
- **Code Optimization Assures Efficient Code Generation and Minimum Application Memory Utilization**

Like its counterpart for MCS-80/85 program development, PL/M 86/88 is an advanced, structured high-level programming language. The PL/M 86/88 compiler was created specifically for performing software development for the Intel iAPX 86,88 Microprocessors.

PL/M 86/88 has significant new capabilities over PL/M 80 that take advantage of the new facilities provided by the iAPX 86,88 microsystem, yet the PL/M 86/88 language remains compatible with PL/M 80.

With the exception of hardware-dependent modules, such as interrupt handlers, PL/M 80 applications may be recompiled with PL/M 86/88 with little need for modification. PL/M 86/88, like PL/M 80, is easy to learn, facilitates rapid program development, and reduces program maintenance costs.

PL/M is a powerful, structured, high-level system implementation language in which program statements can naturally express the program algorithm. This frees the programmer to concentrate on the logic of the program without concern for burdensome details of machine or assembly language programming (such as register allocation, meanings of assembler mnemonics, etc.).

The PL/M 86/88 compiler efficiently converts free-form PL/M language statements into equivalent 86/10 or 88/10 machine instructions. Substantially fewer PL/M statements are necessary for a given application than if it were programmed at the assembly language or machine code level.

The use of PL/M high-level language for system programming, instead of assembly language, results in a high degree of engineering productivity during project development. This translates into significant reductions in initial software development and follow-on maintenance costs for the user.

FEATURES

Major features of the Intel PL/M 86/88 compiler and programming language include:

Block Structure

PL/M source code is developed in a series of modules, procedures, and blocks. Encouraging program modularity in this manner makes programs more readable, and easier to maintain and debug. The language becomes more flexible by clearly defining the scope of user variables (local to a private procedure, global to a public module, for example).

The use of procedures to break down a large problem is paramount to productive software development. The PL/M 86/88 implementation of a block structure allows the use of REENTRANT which is especially useful in system design.

Language Compatibility

PL/M 86/88 object modules are compatible with object modules generated by all other 86/88 translators. This means that PL/M programs may be linked to programs written in any other 86/88 languages.

Object modules are compatible with ICE-88 and ICE-86 units; DEBUG compiler control provides the In-Circuit Emulators with symbolic debugging capabilities.

PL/M 86/88 Language is upward-compatible with PL/M 80, so that application programs may be easily ported to run on the iAPX 86 or 88.

Supports Five Data Types

PL/M makes use of five data types for various applications. These data types range from one to four bytes, and facilitate various arithmetic, logic, and addressing functions:

—Byte: 8-bit unsigned number
—Word: 16-bit unsigned number
—Integer: 16-bit signed number
—Real: 32-bit floating point number
—Pointer: 16-bit or 32-bit memory address indicator

Another powerful facility allows the use of BASED variables that map more than one variable to the same memory location. This is especially useful for passing parameters, relative and absolute addressing, and memory allocation.

Two Data Structuring Facilities

In addition to the five data types and based variables, PL/M supports two data structuring facilities. These add flexibility to the referencing of data stored in large groups.

—Array: Indexed list of same type data elements
—Structure: Named collection of same or different type data elements
—Combinations of Each: Arrays of structures or structures of arrays

8087 Numerics Support

PL/M programs that use 32-bit REAL data may be executed using the Numeric Data Processor for improved performance. All floating-point operations supported by PL/M may be executed on the 8087 NDP, or the 8087 Emulator (a software module) provided with the package. Determination of use of the chip or emulator takes place at link-time, allowing compilations to be run-time independent.

Built-In String Handling Facilities

The PL/M 86/88 language contains built-in functions for string manipulaiton. These byte and word functions perform the following operations on character strings: MOVE, COMPARE, TRANSLATE, SEARCH, SKIP, and SET.

Interrupt Handling

PL/M has the facility for generating interrupts to the iAPX 86 or 88 via software. A procedure may be defined with the INTERRUPT attribute, and the compiler will automatically initialize an interrupt vector at the appropriate memory location. The compiler will also generate code to same and restore the processor status, for execution of the user-defined interrupt handler routine. The procedure SET$INTERRUPT, the function retuning an INTERRUPT$PTR, and the PL/M statement CAUSE$INTERRUPT all add flexibility to user programs involving interrupt handling.

Segmentation Control

The PL/M 86/88 compiler takes full advantage of program addressing with the SMALL, COMPACT, MEDIUM, and LARGE segmentation controls. Programs with less than 64KB total code space can exploit the most efficient memory addressing schemes, which lowers total memory requirements. Larger programs can exploit the flexibility of extended one-megabyte addressing.

Code Optimization

The PL/M 86/88 compiler offers four levels of optimization for significantly reducing overall program size.

—Combination or "folding" of constant expressions; and short-circuit evaluation of Boolean expressions.
—"Strength reductions" (such as a shift left rather than multiply by 2); and elimination of common sub-expressions within the same block.
—Machine code optimizations; elimination of superfluous branches; re-use of duplicate code; removal of unreadable code.
—Byte comparisons (rather than 20-bit address calculations) for pointer variables; optimization of based-variable operations.

Compiler Controls

The PL/M 86/88 compiler offers more than 25 controls that facilitate such features as:

—Conditional compilation
—Intra- and Inter-module cross reference
—Corresponding assembly language code in the listing file
—Setting overflow conditions for run-time handling

BENEFITS

PL/M 86/88 is designed to be an efficient, cost-effective solution to the special requirements of iAPX 86 or 88 Microsystem Software Development, as illustrated by the following benefits of PL/M use:

Low Learning Effort

PL/M 86/88 is easy to learn and to use, even for the novice programmer.

Earlier Project Completion

Critical projects are completed much earlier than otherwise possible because PL/M 86/88, a structured high-level language, increases programmer productivity.

Lower Development Cost

Increases in programmer productivity translate immediately into lower software development costs because less programming resources are required for a given programmed function.

Increased Reliability

PL/M 86/88 is designed to aid in the development of reliable software (PL/M 86/88 programs are simple statements of the program algorithm). This substantially reduces the risk of costly correction of errors in systems that have already reached full production status, as the more simply stated the program is, the more likely it is to perform its intended function.

Easier Enhancements and Maintenance

Programs written in PL/M tend to be self-documenting, thus easier to read and understand. This means it is easier to enhance and maintain PL/M programs as the system capabilities expand and future products are developed.

iAPX 86,88 MACRO ASSEMBLER FOR SERIES II

- **Powerful and Flexible Text Macro Facility with Three Macro Listing Options to Aid Debugging**
- **Highly Mnemonic and Compact Language, Most Mnemonics Represent Several Distinct Machine Instructions**
- **"Strongly Typed" Assembler Helps Detect Errors at Assembly Time**
- **High-Level Data Structuring Facilities Such as "STRUCTUREs" and "RECORDs"**
- **Over 120 Detailed and Fully Documented Error Messages**
- **Produces Relocatable and Linkable Object Code**

ASM 86/88 is the "high-level" macro assembler for the iAPX 86,88 assembly language. ASM 86/88 translates symbolic 86/10, 88/10 assembly language mnemonics into 86/10, 88/10 relocatable object code.

ASM 86/88 should be used where maximum code efficiency and hardware control is needed. The iAPX 86,88 assembly language includes approximately 100 instruction mnemonics. From these few mnemonics the assembler can generate over 3,800 distinct machine instructions. Therefore, the software development task is simplified, as the programmer need know only 100 mnemonics to generate all possible 86/10, 88/10 machine instructions. ASM 86/88 will generate the shortest machine instruction possible given no forward referencing or given explicit information as to the characteristics of forward referenced symbols.

ASM 86/88 offers many features normally found only in high-level languages. The iAPX 86,88 assembly language is strongly typed. The assembler performs extensive checks on the usage of variables and labels. The assembler uses the attributes which are derived explicitly when a variable or label is first defined, then makes sure that each use of the symbol in later instructions conforms to the usage defined for that symbol. This means that many programming errors will be detected when the program is assembled, long before it is being debugged on hardware.

FEATURES

Major features of the Intel iAPX 86,88 assembler and assembly language include:

Powerful and Flexible Text Macro Facility

- Macro calls may appear anywhere
- Allows user to define the syntax of each macro
- Built-in functions
  conditional assembly (IF-THEN-ELSE, WHILE)
  repetition (REPEAT)
  string processing functions (MATCH)
  support of assembly time I/O to console (IN, OUT)
- Three Macro Listing Options include a GEN mode which provides a complete trace of all macro calls and expansions

High-Level Data Structuring Capability

- STRUCTURES: Defined to be a template and then used to allocate storage. The familiar dot notation may be used to form instruction addresses with structure fields.
- ARRAYS: Indexed list of same type data elements.
- RECORDS: Allows bit-templates to be defined and used as instruction operands and/or to allocate storage.

Fully Supports iAPX 86,88 Addressing Modes

- Provides for complex address expressions involving base and indexing registers and (structure) field offsets.
- Powerful EQU facility allows complicated expressions to be named and the name can be used as a synonym for the expression throughout the module.

Powerful STRING MANIPULATION INSTRUCTIONS

- Permit direct transfers to or from memory or the accumulator.
- Can be prefixed with a repeat operator for repetitive execution with a count-down and a condition test.

Over 120 Detailed Error Messages

- Appear both in regular list file and error print file.
- User documentation fully explains the occurrence of each error and suggests a method to correct it.

Support for ICE-86™ Emulation and Symbolic Debugging

- Debug options for inclusion of symbol table in object modules for In-Circuit Emulation with symbolic debugging.

Generates Relocatable and Linkable Object Code—Fully Compatible with LINK 86/88, LOC 86/88 and LIB 86/88

- Permits ASM 86/88 programs to be developed and debugged in small modules. These modules can be easily linked with other ASM 86/88 or PL/M 86/88 object modules and/or library routines to form a complete application system.

BENEFITS

The iAPX 86,88 macro assembler allows the extensive capabilities of the 86/10 and 88/10 CPU's to be fully exploited. In any application, time and space critical routines can be effectively written in ASM 86/88. The 86,88 assembler outputs relocatable and linkable object modules. These object modules may be easily combined with object modules written in PL/M 86/88—Intel's structured, high-level programming language. ASM 86/88 compliments PL/M 86/88 as the programmer may choose to write each module in the language most appropriate to the task and then combine the modules into the complete applications program using the iAPX 86,88 relocation and linkage utilities.

CONV 86/88
MCS-80/85™ to iAPX 86,88 ASSEMBLY LANGUAGE CONVERTER UTILITY PROGRAM

- **Translates 8080/8085 Assembly Language Source Code to iAPX 86,88 Assembly Language Source Code**
- **Provides a Fast and Accurate Means to Convert 8080/8085 Programs to the iAPX 86/10 and the 88/10, Facilitating Program Portability**
- **Automatically Generates Proper ASM 86/88 Directives to Set Up a "Virtual 8080" Environment that is Compatible with PL/M 86/88**

In support of Intel's commitment to software portability, CONV 86/88 is offered as a tool to move 8080/8085 programs to the iAPX 86/10 and the 88/10. A comprehensive manual, "MCS-86 Assembly Language Converter Operating Instructions for ISIS-II Users" (9800642), covers the entire conversion process. Detailed methodology of the conversion process is fully described therein.

— CONV 86/88 will accept as input an error-free 8080/8085 assembly-language source file and optional controls, and produce as output, optional PRINT and OUTPUT files.

— The PRINT file is a formatted copy of the 8080/8085 source and the 86/10 and 88/10 source file with embedded caution messages.

— The OUTPUT file is an 86/10 and 88/10 source file.

— CONV 86/88 issues a caution message when it detects a potential problem in the converted 86/10, 88/10 code.

— A transliteration of the 8080/8085 programs occurs, with each 8080/8085 construct mapped to its exact 86/10, 88/10 counterpart:

Registers
Condition flags
Instruction
Operands
Assembler directives
Assembler control lines
Macros

Because CONV 86/88 is a transliteration process, there is the possibility of as much as a 15%–20% code expansion over the 8080/8085 code. For compactness and efficiency it is recommended that critical portions of programs be re-coded in iAPX 86,88 assembly language.

Also, as a consequence of the transliteration, some manual editing may be required for converting instruction sequences dependent on:

—instruction length, timing, or encoding
—interrupt processing*
—PL/M parameter passing conventions*

*Mechanical editing procedures for these are suggested in the converter manual.

The accompanying figure illustrates the flow of the conversion process. Initially, the abstract program may be represented in 8080/8085 or iAPX 86,88 assembly language to execute on that respective target machine. The conversion process is porting a source destined for the 8080/8085 to the 86/10 or the 88/10 via CONV 86/88.

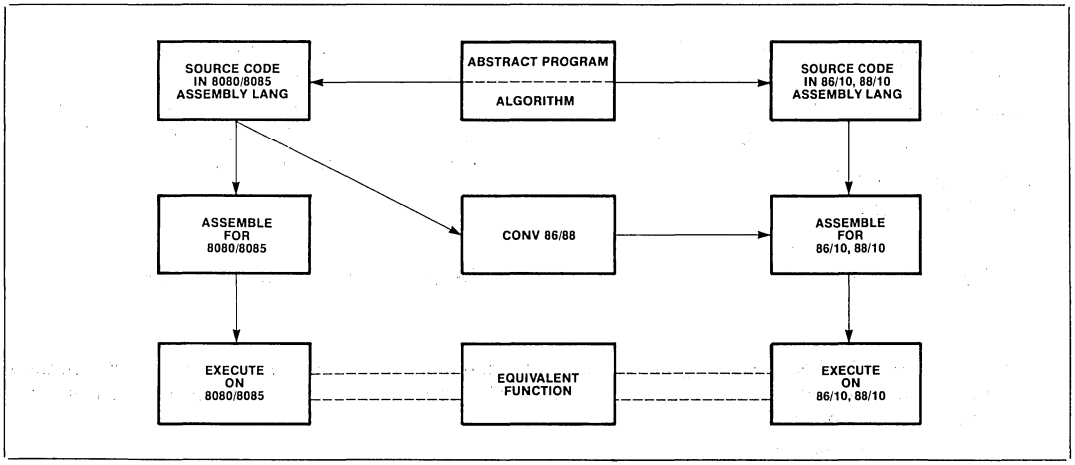


Figure 1. Porting 8080/8085 Source Code to the iAPX 86/10 and 88/10

LINK 86/88

- **Automatic Combination of Separately Compiled or Assembled iAPX 86, 88 Programs Into a Relocatable Module**
- **Automatic Selection of Required Modules from Specified Libraries to Satisfy Symbolic References**
- **Extensive Debug Symbol Manipulation, Allowing Line Numbers, Local Symbols, and Public Symbols to be Purged and Listed Selectively**
- **Automatic Generation of a Summary Map Giving Results of the LINK 86/88 Process**
- **Abbreviated Control Syntax**
- **Relocatable Modules may be Merged into a Single Module Suitable for Inclusion in a Library**
- **Supports "Incremental" Linking**
- **Supports Type Checking of Public and External Symbols**

LINK 86/88 combines object modules specified in the LINK 86/88 input list into a single output module. LINK 86/88 combines segments from the input modules according to the order in which the modules are listed.

LINK 86/88 will accept libraries and object modules built from PL/M 86/88, ASM 86/88, or any other translator generating Intel's iAPX 86/10 Relocatable Object Modules.

Support for incremental linking is provided since an output module produced by LINK 86/88 can be an input to another link. At each stage in the incremental linking process, unneeded public symbols may be purged.

LINK 86/88 supports type checking of PUBLIC and EXTERNAL symbols reporting an error if their types are not consistent.

LINK 86/88 will link any valid set of input modules without any controls. However, controls are available to control the output of diagnostic information in the LINK 86/88 process and to control the content of the output module.

LINK 86/88 allows the user to create a large program as the combination of several smaller, separately compiled modules. After development and debugging of these component modules the user can link them together, locate them using LOC 86/88 and enter final testing with much of the work accomplished.

LIB 86/88

- **LIB 86/88 is a Library Manager Program which Allows You to:**
  - **Create Specially Formatted Files to Contain Libraries of Object Modules**
  - **Maintain These Libraries by Adding or Deleting Modules**
  - **Print a Listing of the Modules and Public Symbols in a Library File**
- **Libraries Can be Used as Input to LINK 86/88 Which Will Automatically Link Modules from the Library that Satisfy External References in the Modules Being Linked**
- **Abbreviated Control Syntax**

Libraries aid in the job of building programs. The library manager program LIB 86/88 creates and maintains files containing object modules. The operation of LIB 86/88 is controlled by commands to indicate which operation LIB 86/88 is to perform. The commands are:

CREATE: creates an empty library file
ADD: adds object modules to a library file
DELETE: deletes modules from a library file
LIST: lists the module directory of library files
EXIT: terminates the LIB 86 program and returns control to ISIS-II

When using object libraries, the linker will call only those object modules that are required to satisfy external references, thus saving memory space.

LOC 86/88

- **Automatic Generation of a Summary Map Giving Starting Address, Segment Addresses and Lengths, and Debug Symbols and their Addresses**
- **Extensive Capability to Manipulate the Order and Placement of Segments in iAPX 86/10, 88/10 Memory**
- **Abbreviated Control Syntax**
- **Automatic and Independent Relocation of Segments. Segments May Be Relocated to Best Match Users Memory Configuration**
- **Extensive Debug Symbol Manipulation, Allowing Line Numbers, Local Symbols, and Public Symbols to be Purged and Listed Selectively**

Relocatability allows the programmer to code programs or sections of programs without having to know the final arrangement of the object code in memory.

LOC 86/88 converts relative addresses in an input module to absolute addresses. LOC 86/88 orders the segments in the input module and assigns absolute addresses to the segments. The sequence in which the segments in the input module are assigned absolute addresses is determined by their order in the input module and the controls supplied with the command.

LOC 86/88 will relocate any valid input module without any controls. However, controls are available to control the output of diagnostic information in the LOC 86/88 process, to control the content of the output module, or both.

The program you are developing will almost certainly use some mix of random access memory (RAM), read-only memory (ROM), and/or programmable read-only memory (PROM). Therefore, the location of your program affects both cost and performance in your application. The relocation feature allows you to develop your program on the Intellec development system and then simply relocate the object code to suit your application.

OH 86/88

- **Converts an iAPX 86/10, 88/10 Absolute Object Module to Symbolic Hexadecimal Format**
- **Facilitates Preparing a File for Later Loading by a Symbolic Hexadecimal Loader, such as the iSBC™ Monitor SDK-86 Loader, or Universal PROM Mapper**
- **Converts an Absolute Module to a More Readable Format that can be Displayed on a CRT or Printed for Debugging**

The OH 86/88 command converts an 86/10, 88/10 absolute object module to the hexadecimal format. This conversion may be necessary to format a module for later loading by a hexadecimal loader such as the iSBC 86/12 monitor or Universal Prom Mapper. The conversion may also be made to put the module in a more readable format that can be displayed or printed.

The module to be converted must be in absolute format; the output from LOC 86/88 is in absolute format.

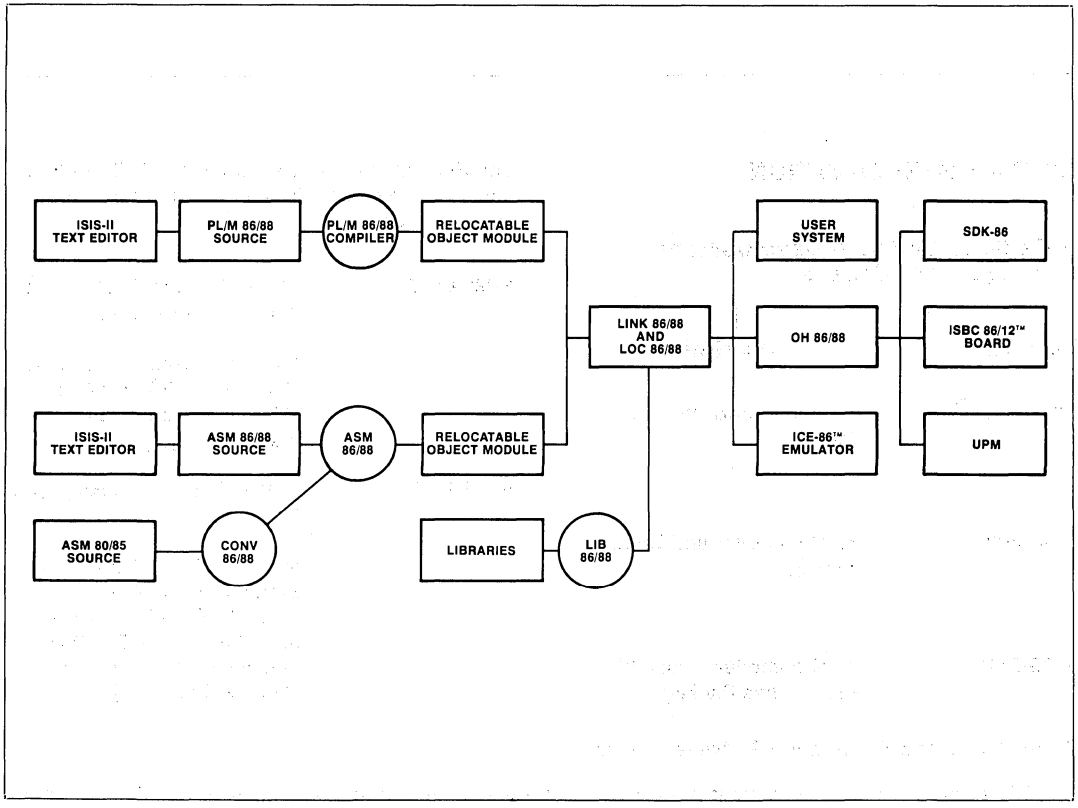


Figure 2. iAPX 86,88 Software Development Cycle

SPECIFICATIONS

Operating Environment

REQUIRED HARDWARE

Intellec® Microcomputer Development System
— Model 800
— Series II

64K Bytes of RAM Memory

Dual Diskette Drives
— Single or Double-Density

System Console
— CRT or Hardcopy Interactive Device

OPTIONAL HARDWARE

Universal PROM Programmer
ICE-86™ Emulator

REQUIRED SOFTWARE

ISIS-II Diskette Operating System

Documentation

PL/M-86 Programming Manual (9800466)

ISIS-II PL/M-86 Compiler Operator's Manual (9800478)

MCS-86 User's Manual (9800722)

MCS-86 Software Development Utilities Operating Instructions for ISIS-II Users (9800639)

MCS-86 Macro Assembly Language Reference Manual (9800640)

MCS-86 Macro Assembler Operating Instructions for ISIS-II Users (9800641)

MCS-86 Assembly Language Converter Operating Instructions for ISIS-II Users (9800642)

Universal PROM Programmer User's Manual (9800819A)

Shipping Media

— Single- and Double-Density Diskettes

ORDERING INFORMATION

iAPX 86,88 Software Development Packages for Series II:

Part No.	Description
MDS-308*	Assembler and Utilities Package
MDS-309*	PL/M compiler and Utilities Package
MDS-311*	PL/M compiler, Assembler, and Utilities Package

The PL/M Compiler, Assembler, and Utilities Package is also available in the following development support packages:

SP86A-KIT	SP86A Support Package (for Intellec® Model 800) Includes ICE-86™ In-Circuit Emulator (MDS-86 ICE) and iAPX 86,88 Software Development Package (MDS-311)
SP86B-KIT	SP86B Support Package (for Series II) Includes ICE-86™ In-Circuit Emulator (MDS-86-ICE), iAPX 86,88 Software Development Package (MDS/311), and Series II Expansion Chassis (MDS-201)

All Packages and Kits Require Software Licenses

*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

iAPX 286
EVALUATION PACKAGE

- **Provides elementary program development and debug capability for the iAPX 286 microprocessor:**
  - **Assembly**
  - **System/task build**
  - **Symbolic debug**
- **Easy evaluation of all microprocessor dependencies: architecture, execution speed, program benchmarks**
- **Provides a programmatic understanding of the iAPX 286 architecture:**
  - **Instruction set**
  - **Memory protection**
  - **Segmentation**
  - **Program timing**
- **Includes an iAPX 286 demonstration program that exploits and illustrates architectural features of the 286**

The Intel iAPX 286 Evaluation Package is an integrated set of software tools that aids the programmer in understanding how to use the iAPX 286 microprocessor. The package runs on a Series III Microcomputer Development System.

The Evaluation Package will allow a programmer to create, build, execute, and debug an assembly-level iAPX 286 task. It will also show how a programmer can take advantage of iAPX 286 architecture features.

The software tools contained in the package are a macro assembler, a task builder, run-time support procedures, an iAPX 286 simulator, and a demonstration program. The simulator has a built-in timer for benchmarking code sequences and programs. It also provides symbolic debugging capability, in addition to iAPX 286 program execution.

The benefits of using the iAPX 286 Evaluation Package are two-fold. System designers may now learn the iAPX 286 architecture (and determine its applicability) in the quickest manner possible. In addition, software may be developed now for a future 286 application, thereby getting a head start on a very time-consuming phase of design.

```
-RUN :F1:SM286E
SERIES III iAPX 286 Evaluation Package Simulator, V1.0

? LOAD DM286E  *the demo program
Program Size = 10432        Total Memory = 60128
? GO

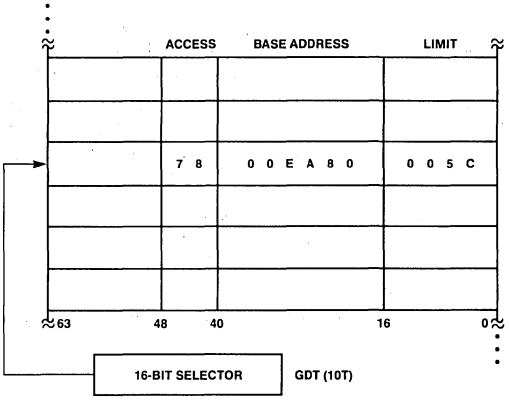
Interrupt  3 at 0118:0D72
?
? TR  *contents of the task register
0250  .RPL=0   .TI=0   .INDEX=004A
?
? TSS   *current task state segment
LINK=0000 SP0=0000 SS0=0260 SP1=0000 SS1=0139 SP2=0000 SS2=0142 IP=0D73
FL=0204 AX=0004 CX=0021 DX=0000 BX=0684 SP=FFA0 BP=FFA0 SI=003F DI=FFB5
ES=0260 CS=0118 SS=0260 DS=0118 RLDT=0018
?
? GO

Interrupt 13 at 0118:0D73  General Protection  Ecode = 0000
?
? LDT
LDT (1T) DSEG BASE=000430 LIMIT=0107 P=1 DPL=0 ED=0 W=1 A=0 SR=0000
LDT (2T) DSEG BASE=00E100 LIMIT=0005 P=1 DPL=3 ED=0 W=1 A=1 SR=0000

? GDT(10T) *the 10th global descriptor table entry
GDT (10T) TSS BASE=00EA80 LIMIT=005C B=0 P=1 DPL=0 SR=0058

? EXIT
-
```

Sample Simulator Session



Segment Descriptor Table

The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, CREDIT, i, ICE, iCS, i_m, Insite, Intel, int$_e$l, Intelevision, Intellec, iRMX, iSBC, iSBX, Library Manager, MCS, Megachassis, Micromap, Multibus, Multimodule, PROMPT, Promware, RMX/80, System 2000, UPI, μScope, and the combination of ICE, iCS, iRMX, iSBC, iSBX, MCS, or RMX and a numerical suffix.

© Intel Corporation, 1981.

SEPTEMBER 1981
AFN-02052A

FUNCTIONAL DESCRIPTION

iAPX 286 Evaluation Macro Assembler

The Evaluation Assembler (AS286E) accepts a source module written in the 286 Macro Assembly Language, and generates an object module and a listing file. The assembler is based upon ASM-86, and therefore performs type-checking on operands, supports complex data structures, and utilizes the same macro processor.

iAPX 286 Evaluation Builder

The Evaluation Builder (BD286E) accepts a single assembler object module, and generates a single-tasking executable load module. The Evaluation Builder performs the following functions:

—Assigns attributes to 286 segments: Privilege level, Access Rights, Base Address, Segment Length.
—Creates descriptor table entries (GDT & LDT) from segments.
—Initializes Segment Registers.
—Allows call gates, interrupt gates, and trap gates to be explicitly created, via the Interrupt Descriptor Table.
—Automatically creates call gates for X286E run-time procedures.
—Creates the Task State Segment for a one-task program.
—Produces a map showing all segments, gates, and public symbols.
—Binds segments to absolute addresses.

iAPX 286 Evaluation Simulator

The Evaluation Simulator (SM286E) loads and executes a 286 object module created by the Builder. Program execution functionally duplicates iAPX 286 processor operation; data protection, gates, processor traps and interrupts, and segmentation access are all supported in the same way as the iAPX 286. Compatibility mode and numerics are not included.

The symbolic debugger portion of the simulator supports two simultaneous code break-points and single-step execution, as well as modification of variables, registers, descriptor tables, and the task state segment. Code disassembly is also provided.

The Simulator has a built-in instruction timer to aid in benchmarking iAPX 286 programs. Another timer, which also counts clock cycles, can be used to generate interrupts at specific time intervals.

Run-Time Support Procedures

The Evaluation Package contains a set of run-time procedures (X286E) that may be "linked" to a user program at build-time to perform several software functions. These functions include creating and modifying segments, descriptors, and tables, creating new tasks, and dynamically allocating free memory for segments.

The Demonstration Program

The Demo Program (DM286E) is an application package that uses the Evaluation Tools (AS286E, BD286E, SM286E, X286E) to teach users how to program the iAPX 286.

It not only guides a programmer through the use of these tools, but the demo program itself illustrates how software can exploit the architecture of the 286. The following features are illustrated:

—Memory Protection using object descriptors.
—Gate creation and manipulation.
—Task switching, procedure entry and exit.
—Interrupt handling.
—Dynamic task creation.
—Inter-task communication.

The demonstration consists of a nucleus, a real-time clock interface, a time-of-day clock, a CPU-utilization spy, and a command interpreter. The user may execute these simultaneously on the simulator, and gain an understanding of how the 286 handles the functions listed above.

ORDERING INFORMATION

Product Code	Description
MDS-322	iAPX 286 Evaluation Package

(Requires Software License)

PL/M 86/88 SOFTWARE PACKAGE

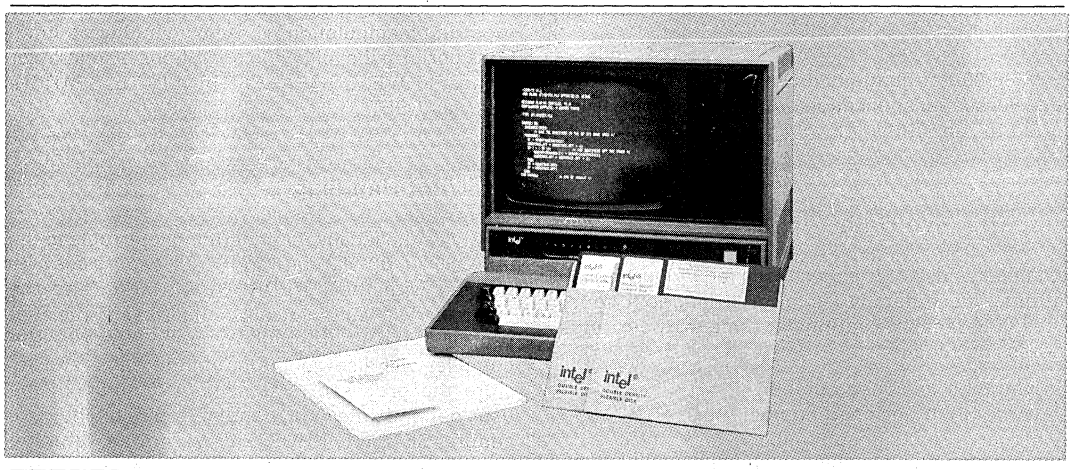
- **Executes on Series III iAPX 86 Processor for Fastest Compilations**
- **Language Is Upward Compatible from PL/M 80, Assuring MCS-80/85 Design Portability**
- **Supports 16-Bit Signed Integer and 32-Bit Floating Point Arithmetic in Accordance with IEEE Proposed Standard**
- **Easy-To-Learn Block-Structured Language Encourages Program Modularity**
- **Improved Compiler Performance Now Supports More User Symbols and Faster Compilation Speeds**
- **Produces Relocatable Object Code Which Is Linkable to All Other 8086 Object Modules**
- **Code Optimization Assures Efficient Code Generation and Minimum Application Memory Utilization**
- **Built-In Syntax Checker Doubles Performance for Compiling Programs Containing Errors**

Like its counterpart for MCS-80/85 program development, PL/M 86/88 is an advanced, structured high-level programming language. The PL/M 86/88 compiler was created specifically for performing software development for the Intel 8086 and 8088 Microprocessors.

PL/M is a powerful, structured, high-level system implementation language in which program statements can naturally express the program algorithm. This frees the programmer to concentrate on the logic of the program without concern for burdensome details of machine or assembly language programming (such as register allocation, meanings of assembler mnemonics, etc.).

The PL/M 86/88 compiler efficiently converts free-form PL/M language statements into equivalent 8088/8086 machine instructions. Substantially fewer PL/M statements are necessary for a given application than if it were programmed at the assembly language or machine code level.

The use of PL/M high-level language for system programming, instead of assembly language, results in a high degree of engineering productivity during project development. This translates into significant reductions in initial software development and follow-on maintenance costs for the user.



NOTE: The Intellec® Microcomputer Development System pictured here is not included with the PL/M 86/88 Software Package but merely depicts a language in its operating environment.

The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, I^{2}m, ICS, ICE, UPI, BXP, iSBC, iSBX, iNSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or ICS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

FEATURES

Major features of the Intel PL/M 86/88 compiler and programming language include:

Block Structure

PL/M source code is developed in a series of modules, procedures, and blocks. Encouraging program modularity in this manner makes programs more readable, and easier to maintain and debug. The language becomes more flexible, by clearly defining the scope of user variables (local to a private procedure, global to a public procedure, for example).

The use of procedures to break down a large problem is paramount to productive software development. The PL/M 86/88 implementation of a block structure allows the use of REENTRANT (recursive) procedures, which are especially useful in system design.

Language Compatibility

PL/M 86/88 object modules are compatible with object modules generated by all other 86/88 translators. This means that PL/M programs may be linked to programs written in any other 86/88 language.

Object modules are compatible with ICE-88 and ICE-86 units; DEBUG compiler control provides the In-Circuit Emulators with symbolic debugging capabilities.

PL/M 86/88 Language is upward-compatible with PL/M 80, so that application programs may be easily ported to run on the iAPX 86 or 88.

Supports Five Data Types

PL/M makes use of five data types for various applications. These data types range from one to four bytes, and facilitate various arithmetic, logic, and addressing functions:

— Byte: 8-bit unsigned number
— Word: 16-bit unsigned number
— Integer: 16-bit signed number
— Real: 32-bit floating point number
— Pointer: 16-bit or 32-bit memory address indicator

Another powerful facility allows the use of BASED variables that map more than one variable to the same memory location. This is especially useful for passing parameters, relative and absolute addressing, and memory allocation.

Two Data Structuring Facilities

In addition to the five data types and based variables, PL/M supports two data structuring facilities. These add flexibility to the referencing of data stored in large groups.

— Array: Indexed list of same type data elements
— Structure: Named collection of same or different type data elements
— Combinations of Each: Arrays of structures or structures of arrays

8087 Numerics Support

PL/M programs that use 32-bit REAL data may be executed using the Numeric Data Processor for improved performance. All floating-point operations supported by PL/M may be executed on the iAPX 86/20 or 88/20 NDP, or the 8087 Emulator (a software module) provided with the package. Determination of use of the chip or Emulator takes place at link-time, allowing compilations to be run-time independent.

Built-In String Handling Facilities

The PL/M 86/88 language contains built-in functions for string manipulation. These byte and word functions perform the following operations on character strings: MOVE, COMPARE, TRANSLATE, SEARCH, SKIP, and SET.

Interrupt Handling

PL/M has the facility for generating interrupts to the iAPX 86 or 88 via software. A procedure may be defined with the INTERRUPT attribute, and the compiler will automatically initialize an interrupt vector at the appropriate memory location. The compiler will also generate code to same and restore the processor status, for execution of the user-defined interrupt handler routine. The procedure SET$INTERRUPT, the function retuning an INTERRUPT$PTR, and the PL/M statement CAUSE$INTERRUPT all add flexibility to user programs involving interrupt and handling.

Compiler Controls

Including several that have been mentioned, the PL/M 86/88 compiler offers more than 25 controls that facilitate such features as:

— Conditional compilation
— Including additional PL/M source files from disk
— Intra- and Inter-module cross reference
— Corresponding assembly language code in the listing file
— Setting overflow conditions for run-time handling

Segmentation Control

The PL/M 86/88 compiler takes full advantage of program addressing with the SMALL, COMPACT, MEDIUM, and LARGE segmentation controls. Programs with less than 64KB total code space can exploit the most efficient memory addressing schemes, which lowers total memory requirements. Larger programs can exploit the flexibility of extended one-megabyte addressing.

Code Optimization

The PL/M 86/88 compiler offers four levels of optimization for significantly reducing overall program size.

— Combination or "folding" of constant expressions; and short-circuit evaluation of Boolean expressions.
— "Strength reductions" (such as a shift left rather than multiply by 2); and elimination of common sub-expressions within the same block.
— Machine code optimizations; elimination of superfluous branches; re-use of duplicate code; removal of unreadable code.
— Byte comparisons (rather than 20-bit address calculations) for pointer variables; optimization of based-variable operations.

Error Checking

The PL/M 86/88 compiler has a very powerful feature to speed up compilations. If a syntax or program error is detected, the compiler will skip the code generation and optimization passes. This usually yields a 2X performance increase for compilation of programs with errors.

A fully detailed set of programming and compilation errors is provided by the compiler.

Compiler Performance

Performance benchmarks may provide valuable information in estimating compile times for various programs. It is extremely important to understand, however, the effect of varying conditions on compiler performance. Storage media, coding style, program length, and the use of INCLUDE files significantly change the compiler's overall performance. We tested typical PL/M programs of varying lengths. The results are listed in Table 1.

Table 1. PL/M Program Compile Times

Program Size	Compile Time(Sec)	Lines/Minute
SMALL (71)	20	213
MEDIUM (610)	54	678
LARGE (1710)	128	802
LARGE (1403) (with very dense code, plus include file)	129	653

NOTE: These programs were run on a Series III with ISIS 4.1 and a hard disk. The lines per minute figures reflect fifteen percent blank lines and comments.

The compiler allows approximately 1000 ten-character user symbols.

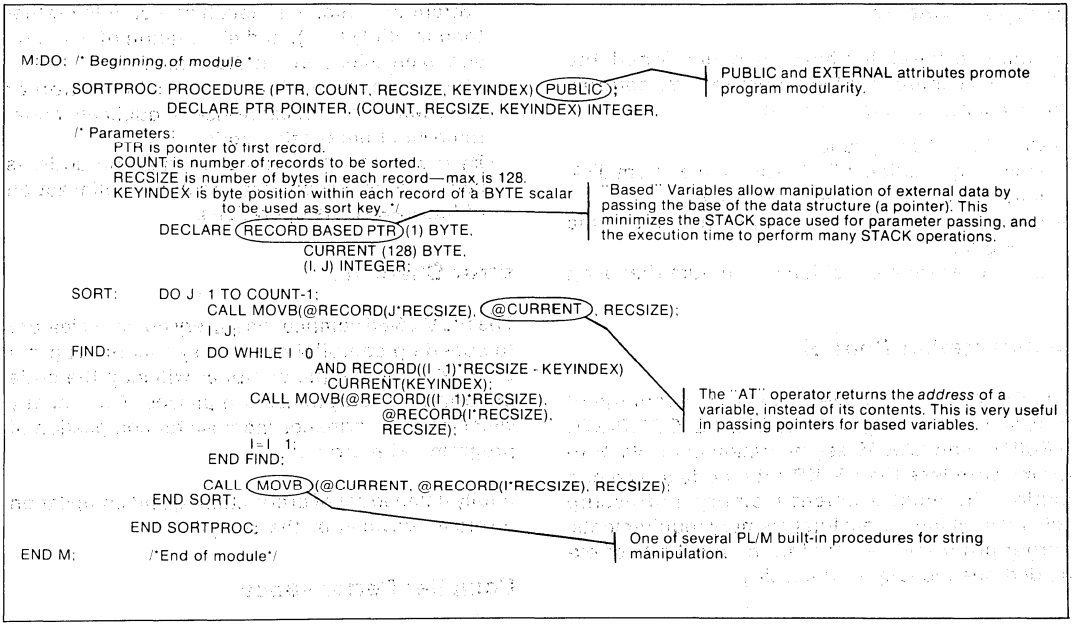


Figure 1. Sample PL/M 86/88 Program

BENEFITS

PL/M 86/88 is designed to be an efficient, cost-effective solution to the special requirements of iAPX 86 or 88 Microsystem Software Development, as illustrated by the following benefits of PL/M use:

Low Learning Effort

PL/M 86/88 is easy to learn and to use, even for the novice programmer.

Earlier Project Completion

Critical projects are completed much earlier than otherwise possible because PL/M 86/88, a structured high-level language, increases programmer productivity.

Lower Development Cost

Increases in programmer productivity translate immediately into lower software development costs because less programming resources are required for a given programmed function.

Increased Reliability

PL/M 86/88 is designed to aid in the development of reliable software (PL/M 86/88 programs are simple statements of the program algorithm). This substantially reduces the risk of costly correction of errors in systems that have already reached full production status, as the more simply stated the program is, the more likely it is to perform its intended function.

Easier Enhancements and Maintenance

Programs written in PL/M tend to be self-documenting, thus easier to read and understand. This means it is easier to enhance and maintain PL/M programs as the system capabilities expand and future products are developed.

SPECIFICATIONS

Operating Environment

REQUIRED HARDWARE:
Intellec® Microcomputer Development System
— Series III or equivalent
Dual Diskette Drives
— Single- or Double-Density
System Console
— CRT or Hardcopy Interactive Device

OPTIONAL HARDWARE:
Universal PROM Programmer
Line Printer
ICE-86™

REQUIRED SOFTWARE:
ISIS-II Diskette Operating System, V4.1 or later
Series III Operating System

Documentation Package

PL/M-86 User's Guide for 8086-based Development Systems (121636)

ORDERING INFORMATION

Part Number	Description
MDS-313*	PL/M 86/88 Software Package

Requires Software License

*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

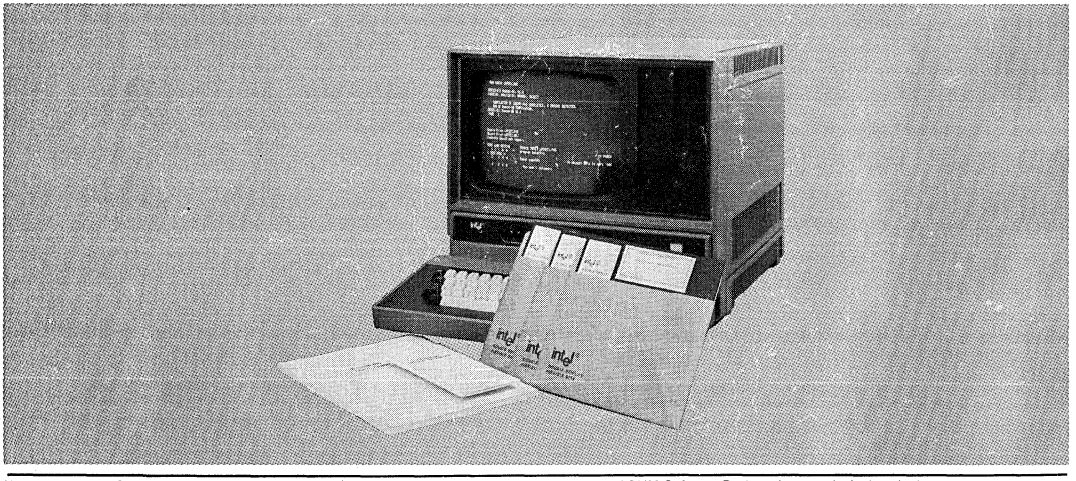
PASCAL 86/88
SOFTWARE PACKAGE

- **Resident on iAPX 86 Based Intellec® Series III Microcomputer Development System for Optimal Performance**
- **Object Compatible and Linkable with PL/M 86/88, ASM 86/88 and FORTRAN 86/88**
- **ICE™ Symbolic Debugging Fully Supported**
- **Implements REALMATH for Consistent and Reliable Results**
- **Supports iAPX86/20, 88/20 Numeric Data Processors**
- **Strict Implementation of ISO Standard Pascal**
- **Useful Extensions Essential for Microcomputer Applications**
- **Separate Compilation with Type-Checking Enforced Between Pascal Modules**
- **Compiler Option to Support Full Run-Time Range-Checking**

PASCAL 86/88 conforms to and implements the ISO Draft Proposed Pascal standard. The language is enhanced to support microcomputer applications with special features, such as separate compilation, interrupt handling and direct port I/O. To assist the development of portable software, the compiler can be directed to flag all non-standard features.

The PASCAL 86/88 compiler runs on the iAPX 86 Resident Intellec® Series III Microcomputer Development System. A well-defined I/O interface is provided for run-time support. This allows a user-written operating system to support application programs as an alternate to the development system environment. Program modules compiled under PASCAL 86/88 are compatible and linkable with modules written in PL/M 86/88, ASM 86/88 or FORTRAN 86/88. With a complete family of compatible programming languages for the iAPX 86, 88 one can implement each module in the language most appropriate to the task at hand.

PASCAL 86/88 object modules contain symbol and type information for program debugging using ICE-86™ emulator. For final production version, the compiler can remove this extra information and code.



Note: The Intellec® microcomputer development system pictured here is not included with the Pascal 86/88 Software Package but merely depicts the language in its operating environment.

The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, CREDIT, Intellec, Multibus, i, iSBC, Multimodule, ICE, iSBX, PROMPT, iRMX, iCS, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Micromap, μScope and the combination of iCE, iCS, iSBC, iSBX, MCS, or RMX and a numerical suffix.

© Intel Corporation 1980

121680-001 Rev. A

FEATURES

Includes all the language features of Jensen & Wirth Pascal as defined in the ISO Draft Proposed Pascal Standard.

Supports required extensions for microcomputer applications.

—Interrupt handling

—Direct port I/O

Separate compilation extensions allow:

—Modular decomposition of large programs

—Linkage with other Pascal modules as well as PL/M 86/88, ASM 86/88 and FORTRAN 86/88.

—Enforcement of type-checking at LINK-time

Supports numerous compiler options to control the compilation process, to INCLUDE files, flag non-standard Pascal statements and others to control program listings and object modules.

Utilizes the IEEE standard for Floating-Point Arithmetic (the Intel REALMATH standard) for arithmetic operations.

Well-defined and documented run-time operating system interfaces allow the user to execute the applications under user-designed operating systems.

BENEFITS

Provides a standard Pascal for iAPX 86, 88 based applications.

—Pascal has gained wide acceptance as the portable application language for microcomputer applications

—It is being taught in many colleges and universities around the world

—It is easy to learn, originally intended as a vehicle for teaching computer programming

—Improves maintainability: Type mechanism is both strictly enforced and user extendable

—Few machine specific language constructs

Strict implementation of the proposed ISO standard for Pascal aids portability of application programs. A compile time option checks conformance to the standard making it easy to write conforming programs.

PASCAL 86/88 extensions via predefined procedures for interrupt handling and direct port I/O make it possible to code an entire application in Pascal without compromising portability.

Standard Intel REALMATH is easy to use and provides reliable results, consistent with other Intel languages and other implementations of the IEEE proposed Floating-Point standard.

Provides run-time support for co-processors. All real-type arithmetic is performed on the 86/20 numeric data processor unit or software emulator. Run-time library routines, common between Pascal and other Intel languages (such as FORTRAN), permit efficient and consistently accurate results.

Extended relocation and linkage support allows the user to link Pascal program modules with routines written in other languages for certain parts of the program. For example, real-time or hardware dependent routines written in ASM 86/88 or PL/M 86/88 can be linked to Pascal routines, further extending the user's ability to write structured and modular programs.

PASCAL 86/88 programs "talk" to the resident operating system using Intel's standard interface for translated programs. This allows users to replace the development operating system by their own operating systems in the final application.

PASCAL 86/88 takes full advantage of iAPX 86, 88 high level language architecture to generate efficient machine code without using time-consuming optimization algorithms.

Compiler options can be used to control the program listings and object modules. While debugging, the user may generate additional information such as the symbol record information required and useful for debugging using ICE emulation. After debugging, the production version may be streamlined by removing this additional information.

SPECIFICATIONS

Operating Environment

REQUIRED HARDWARE
Intellec® Series III Microcomputer Development System
—System Console
—Double Density Dual Diskette Drive OR Hard Disk

REQUIRED SOFTWARE
ISIS-II Diskette Operating System V4.1 or later

Documentation Package

PASCAL 86 User's Guide (121539-001)

Shipping Media

Flexible Diskettes
—Single and Double Density

ORDERING INFORMATION

Part Number	Description
MDS*-314	PASCAL 86/88 Software Package

Requires software license.

* MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Science.

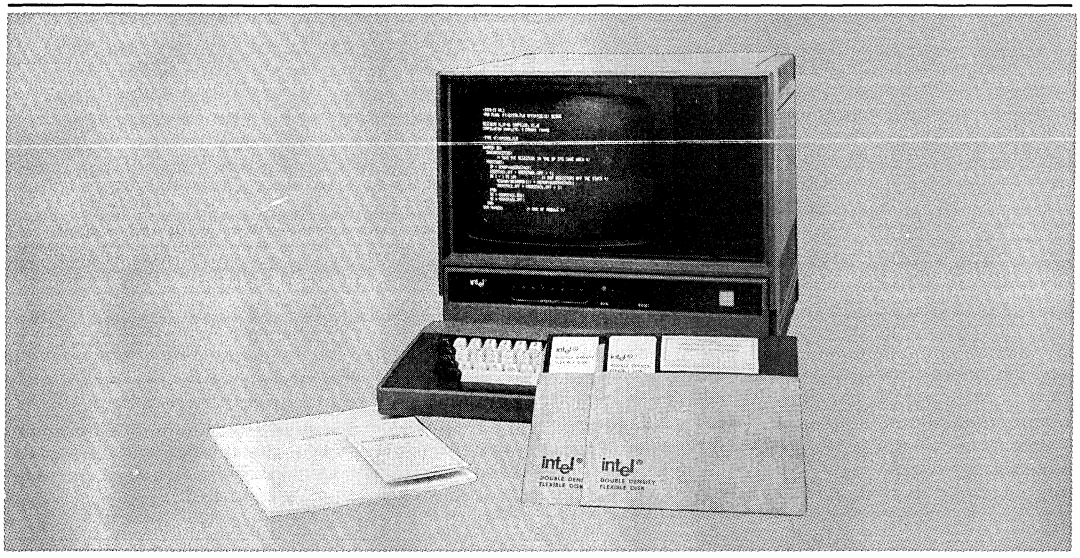
FORTRAN 86/88 SOFTWARE PACKAGE

- **Features high-level language support for floating-point calculations, transcendentals, interrupt procedures, and run-time exception handling**
- **Meets ANS FORTRAN 77 Subset Language Specifications**
- **Supports iAPX 86/20, 88/20 Numeric Data Processor for fast and efficient execution of numeric instructions**
- **Uses REALMATH Floating-Point Standard for consistent and reliable results**
- **Offers powerful extensions tailored to microprocessor applications**
- **Offers upward compatibility with FORTRAN 80**
- **Provides FORTRAN run-time support for iAPX 86,88-based design**
- **Provides users ability to do formatted and unformatted I/O with sequential or direct access methods**

FORTRAN 86/88 meets the ANS FORTRAN 77 Language Subset Specification and includes many features of the full standard. Therefore, the user is assured of portability of most existing ANS FORTRAN programs and of full portability from other computer systems with an ANS FORTRAN 77 Compiler.

FORTRAN 86/88 programs developed and debugged on the iAPX 86 Resident Intellec Series III Microcomputer Development System may be: tested with the prototype using ICE symbolic debugging, and executed on an RMX-86 operating system, or on a user's iAPX 86,88-based operating system.

FORTRAN 86/88 is one of a complete family of compatible programming languages for iAPX 86,88 development: PL/M, Pascal, FORTRAN, and Assembler. Therefore, users may choose the language best suited for a specific problem solution.



NOTE: The Intellec® Microcomputer Development System shown here merely shows the operating environment and is not included with the software package.

The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{i}m, iCS, ICE, UPI, BXP, iSBC, iSBX, iNSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTIMODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation, 1981.

FEATURES

Extensive High-Level Language Numeric Processing Support

Single (32-bit), double (64-bit), and double extended precision (80-bit) floating-point data types

REALMATH Proposed IEEE Floating-Point Standard) for consistent and reliable results

Full support for all other data types: integer, logical, character

Ability to use hardware (iAPX 86/20, 88/20 Numeric Data Processor) or software (simulator) floating-point support chosen at link time

ANS FORTRAN 77 Standard

Intel® Microprocessor Support

FORTRAN 86/88 language features support of iAPX 86/20, 88/20 Numeric Data Processor

Compiler generates in-line iAPX 86/20, 88/20 Numeric Data Processor object code for floating-point arithmetic (See Figure 1)

Intrinsics allow user to control iAPX 86/20, 88/20 Numeric Data Processor

iAPX 86,88 architectural advantages used for indexing and character-string handling

Symbolic debugging of application using ICE-86 and ICE-88 emulators

FLOATING-POINT-STATMENT

```
 TEMPER = (PRESS - VOLUM / QUEK) - 3.45 / (PRESS - VOLUM / QUEK)
&    - (PRESS - VOLUM / QUEK) * (PRESS - VOLUM / QUEK)
```

OBJECT CODE GENERATED

```
Intel FORTRAN-86 Compiler

                                                              ; STATEMENT # 2
0013   9BD9060C00          FLD       VOLUM
0018   9BD8360000          FDIV      QUEK
001D   9BD82E0800          FSUBR     PRESS
0022   9BDDD1              FST       TOS+1H
0025   9B2ED83E0000        FDIVR     CS:@CONST
002B   9BD9C9              FXCHG     TOS+1H
002E   9BDDD2              FST       TOS+2H
0031   9BDEE9              FSUBRP
0034   9BD9C1              FLD       TOS+1H
0037   9BD8C8              FMUL      TOS
003A   9BDDC2              FFREE     TOS+2H
003D   9BDEE1              FSUBP
0040   9BD91E0400          FSTP      TEMPER
0045   9B                  WAIT
```

IAPX 86/20, 88/20 MACHINE CODE

ASSEMBLER MNEMONICS

Figure 1. Object Code Generated by FORTRAN 86/88 for a Floating-Point Calculation Using iAPX 86/20, 88/20 Numeric Processor

Microprocessor Application Support

—Direct byte- or word-oriented port I/O

—Reentrant procedures

—Interrupt procedures

Flexible Run-Time Support

Application object code may be executed in iAPX 86, 88-based environment of user's choice:

—a Series III Intellec Development System with Series III Operating System

—an iAPX 86,88-based system with iRMX-86 Operating System

—an iAPX 86,88-based system with user-designed Operating System

Run-time exception handling for fixed-point numerics, floating-point numerics, and I/O errors

Relocatable object libraries for complete run-time support of I/O and arithmetic functions. In-line code execution is generated for iAPX 86/20, 88/20 Numeric Data Processor

BENEFITS

FORTRAN 86/88 provides a means of developing application software for the Intel iAPX 86,88 products lines in a familiar, widely accepted, and industry-standard programming language. FORTRAN 86/88 will greatly enhance the user's ability to provide cost-effective software development for Intel microprocessors as illustrated by the following:

Early Project Completion

FORTRAN is an industry-standard, high-level numerics processing language. FORTRAN programmers can use FORTRAN 86/88 on microprocessor projects with little retraining. Existing FORTRAN software can be compiled with FORTRAN 86/88 and programs developed in FORTRAN 86/88 can run on other computers with ANS FORTRAN 77 with little or no change. Libraries of mathematical programs using ANS 77 standards may be compiled with FORTRAN 86/88.

Application Object Code Portability for a Processor Family

FORTRAN 86/88 modules "talk" to the resident Intellec development operating system using Intel's standard interface for all development-system software. This allows an application developed on the Series III operating system to execute on iRMX/86, or a user-supplied operating system by linking in the iRMX/86 or other appropriate interface library. A standard logical-record interface enables communication with non-standard I/O devices.

Comprehensive, Reliable and Efficient Numeric Processing

The unique combination of FORTRAN 86/88, iAPX 86/20, 88/20 Numeric Data Processor, and REALMATH (Proposed IEEE Floating-Point Standard) provide universal consistency in results of numeric computations and efficient object code generation.

SPECIFICATIONS

Operating Environment

REQUIRED HARDWARE
Intellec® Series III Microcomputer Development System

—System Console

—Double-Density Dual-Diskette Drive. A Hard Disk is recommended

—Hard Disk*

*Recommended.

REQUIRED SOFTWARE
ISIS-II Diskette Operating System V4.1 or later

Documentation Package

FORTRAN 86/88 User's Guide (121539-001)

Shipping Media

Flexible Diskettes

—Single- and Double-Density

ORDERING INFORMATION

Part Number	Description
MDS**-315	FORTRAN 86/88 Software Package

Requires Software License

** MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Science.

Printed in USA/H-20/0481/CBM/BL

8087
SOFTWARE SUPPORT PACKAGE

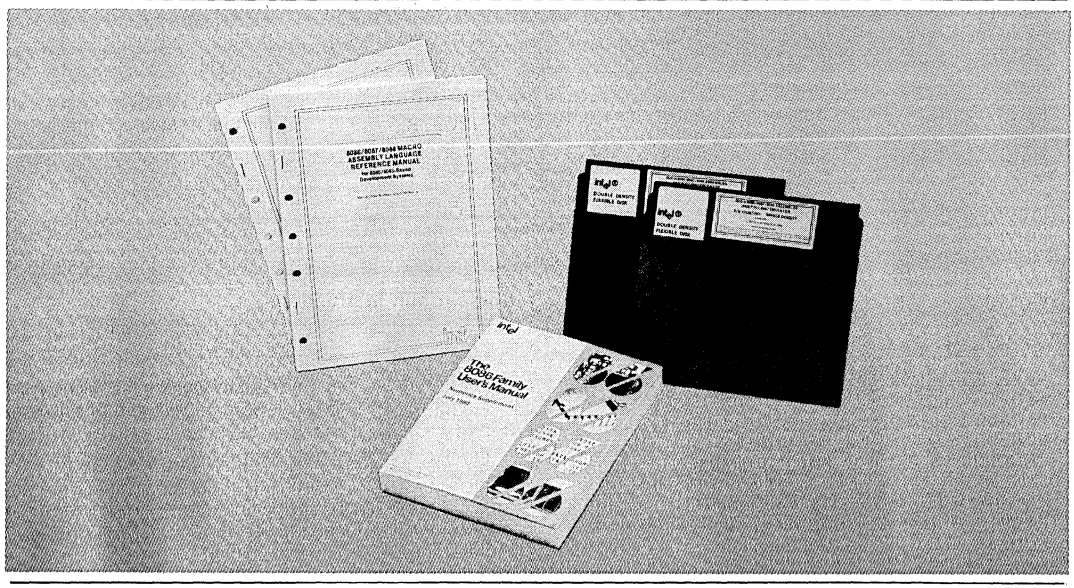
- **Program Generation for the 8087 Numeric Data Processor on the Intellec® Microcomputer Development System**
- **Consists of: 8086/8087/8088 Macro Assembler, 8087 Software Emulator**
- **Macro Assembler Generates Code for 8087 Processor or Emulator, While Also Supporting the 8086/8088 Instruction Set**
- **8087 Emulator Duplicates Each 8087 Floating-Point Instruction in Software, for Evaluation of Prototyping, or for Use in an End Product**
- **Macro Assembler and 8087 Emulator are Fully Compatible with Other 8086/8088 Development Software**
- **Implementation of the IEEE Proposed Floating-Point Standard (the Intel® Realmath Standard)**

The 8087 Software Support Package is an optional extention of Intel's 8086/8088 Software Development Package that runs under ISIS-II on an Intellec or Series II Microcomputer Development System.

The 8087 Software Support Package consists of the 8086/8087/8088 Macro Assembler, and the Full 8087 Emulator. The assembler is a functional superset of the 8086/8088 Macro Assembler, and includes instructions for over sixty new floating-point operations, plus new data types supported by the 8087.

The 8087 Emulator is an 8086/8088 object module that simulates the environment of the 8087, and executes each floating-point operation using software algorithms. This emulator functionally duplicates the operation of the 8087 Numeric Data Processor.

Also included in this package are interface libraries to link with 8086/8087/8088 object modules, which are used for specifying whether the 8087 Processor or the 8087 Emulator is to be used. This enables the run-time environment to be invisible to the programmer at assembly time.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, CREDIT, Intellec, Multibus, i, iSBC, Multimodule, ICE, iSBX, PROMPT, iRMX, iCS, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Micromap, μScope and the combination of iCE, iCS, iSBC, iSBX, MCS, or RMX and a numerical suffix.
© Intel Corporation 1980

121653-001 Rev. A

FUNCTIONAL DESCRIPTION

8086/8087/8088 Macro Assembler

The 8086/8087/8088 Macro Assembler translates symbolic macro assembly language instructions into appropriate machine instructions. It is an extended version of the 8086/8088 Macro Assembler, and therefore supports all of the same features and functions, such as limited type checking, conditional assembly, data structures, macros, etc. The extensions are the new instructions and data types to support floating-point operations. Realmath floating-point instructions (see Table 1) generate code capable of being converted to either 8087 instructions or interrupts for the 8087 Emulator. The Processor/Emulator selection is made via interface libraries at LINK-time. In addition to the new floating-point instructions, the macro assembler also introduces two new 8087 data types: QWORD (8 bytes) and TBYTE (ten bytes). These support the highest precision of data processed by the 8087.

Full 8087 Emulator

The Full 8087 Emulator is a 16-kilobyte object module that is linked to the application program for floating-point operations. Its functionality is identical to the 8087 chip, and is ideal for prototyping and debugging floating-point applications. The Emulator is an alternative to the use of the 8087 chip, although the latter executes floating-point applications up to 100 times faster than an 8086 with the 8087 Emulator. Furthermore, since the 8087 is a "co-processor," use of the chip will allow many operations to be performed in parallel with the 8086.

Table 1. 8087 Instructions

Arithmetic Instructions

Addition	
FADD	Add real
FADDP	Add real and pop
FIADD	Integer add
Subtraction	
FSUB	Subtract real
FSUBP	Subtract real and pop
FISUB	Integer subtract
FSUBR	Subtract real reversed
FSUBRP	Subtract real reversed and pop
FISUBR	Integer subtract reversed
Multiplication	
FMUL	Multiply real
FMULP	Multiply real and pop
FIMUL	Integer multiply
Division	
FDIV	Divide real
FDIVP	Divide real and pop
FIDIV	Integer divide
FDIVR	Divide real reversed
FDIVRP	Divide real reversed and pop
FIDIVR	Integer divide reversed
Other Operations	
FSQRT	Square root
FSCALE	Scale
FPREM	Partial remainder
FRNDINT	Round to integer
FXTRACT	Extract exponent and significand
FABS	Absolute value
FCHS	Change sign

Processor Control Instructions

FINIT/FNINIT	Initialize processor
FDISI/FNDISI	Disable interrupts
FENI/FNENI	Enable interrupts
FLDCW	Load control word
FSTCW/FNSTCW	Store control word
FSTSW/FNSTSW	Store status word
FCLEX/FNCLEX	Clear exceptions
FSTENV/FNSTENV	Store environment
FLDENV	Load environment
FSAVE/FNSAVE	Save state
FRSTOR	Restore state
FINCSTP	Increment stack pointer
FDECSTP	Decrement stack pointer
FFREE	Free register
FNOP	No operation
FWAIT	CPU wait

Comparison Instructions

FCOM	Compare real
FCOMP	Compare real and pop
FCOMPP	Compare real and pop twice
FICOM	Integer compare
FICOMP	Integer compare and pop
FTST	Test
FXAM	Examine

Table 1. 8087 Instructions (cont'd)

Transcendental Instructions

FPTAN	Partial tangent
FPATAN	Partial arctangent
F2XM1	2^x-1
FYL2X	$Y \bullet \log_2 X$
FYL2XP1	$Y \bullet \log_2(X+1)$

Constant Instructions

FLDZ	Load +0.0
FLD1	Load +1.0
FLDPI	Load π
FLDL2T	Load $\log_2 10$
FLDL2E	Load $\log_2 e$
FLDLG2	Load $\log_{10} 2$
FLDLN2	Load $\log_e 2$

Data Transfer Instructions

Real Transfers	
FLD	Load real
FST	Store real
FSTP	Store real and pop
FXCH	Exchange registers
Integer Transfers	
FILD	Integer load
FIST	Integer store
FISTP	Integer store and pop
Packed Decimal Transfers	
FBLD	Packed decimal (BCD) load
FBSTP	Packed decimal (BCD) store and pop

SPECIFICATIONS

Operating Environment

REQUIRED HARDWARE
Intellec® Microcomputer Development System
—Model 800
—Series II (Models 220, 225 or equivalent)

64K Bytes of RAM Memory

Minimum One Diskette Drive
—Single or Double* Density

System Console
—CRT or Hardcopy Interactive Device

OPTIONAL HARDWARE
Universal PROM Programmer*
Line Printer*

*Recommended

REQUIRED SOFTWARE
ISIS-II Diskette Operating System
—Single or Double Density

8086/8088 Software Development Package

Documentation Package

8086/8087/8088 Macro Assembly Language Reference Manual for 8080/8085-Based Development Systems (121623-001)

8086/8087/8088 Macro Assembler Operating Instructions for 8080/8085-Based Development Systems (121624-001)

The 8086 Family Users Manual Supplement for the 8087 Numeric Data Processor (121586-001)

Shipping Media

1 Single and 1 Double Density Diskette

ORDERING INFORMATION

Part Number	Description
MDS*-387	8087 Software Support Package

Requires Software License

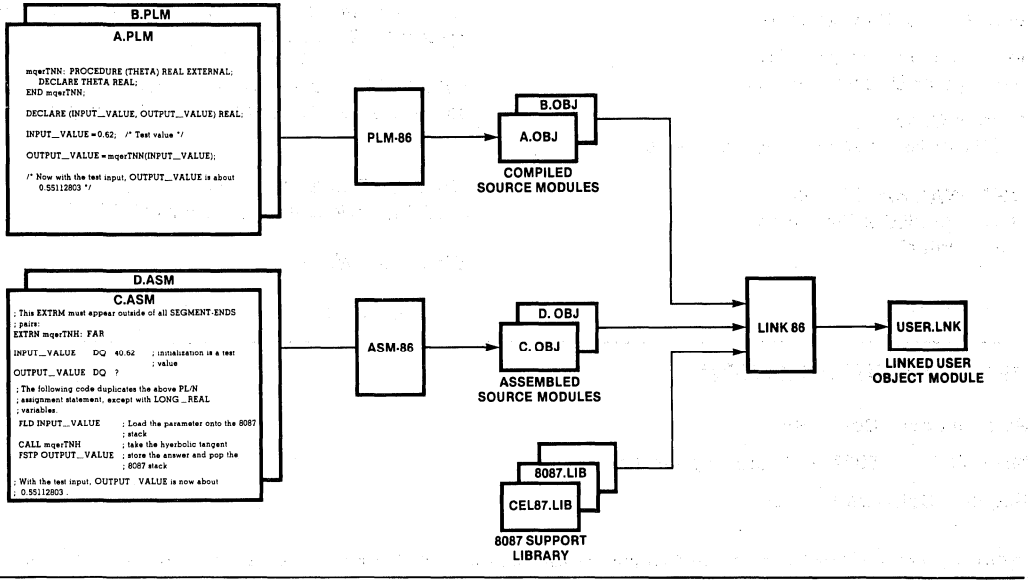
***MDS** is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

8087 SUPPORT LIBRARY

- **Library to support floating point arithmetic in PL/M-86 and ASM-86**
- **Common elementary function library provides trigonometric, logarithmic and other useful functions**
- **Decimal conversion module supports binary-decimal conversions**
- **Error-handler module simplifies floating point error recovery**
- **Full 8087 Software Emulator for software debugging without the 8087 component**
- **Accurate, verified and efficient implementation of algorithms for functions**
- **Supports proposed IEEE Floating Point Standard for high accuracy and software portability**

The 8087 Support Library provides PL/M-86 and ASM-86 users with the equivalent numeric data processing capability of Fortran-86. With the Library, it is easy for PL/M-86 and ASM-86 programs to do floating point arithmetic. Programs can link in modules to do trigonometric, logarithmic and other numeric functions, and the user is guaranteed accurate, reliable results for all appropriate inputs. The 8087 Support Library implements Intel's REALMATH standard and also supports the proposed IEEE Floating Point Standard. Consequently, by using this Library, the PL/M-86 user not only saves software development time, but is guaranteed that the numeric software meets industry standards and is portable—his software investment is maintained.

The 8087 Support Library consists of the common elementary function library, the decimal conversion module, the error handler module, the full 8087 Software emulator and interface libraries to the 8087 and to the 8087 emulator.



The following are trademarks of Intel Corporation and may only be used to identify Intel products: BXP, CREDIT, Intellec, Multibus, iSBC, Multimodule, ICE, iSBX, PROMPT, iCS, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Micromap, μScope and the combination of ICE, iCS, ISBCICS, i_m, Insite, Intel, Intelevision, Intellec, iSBC, iSBX, MCS, or RMX and a numerical suffix.

© INTEL CORPORATION, 1981

October 1981
Lit 2-210209-001

CEL87.LIB
THE COMMON ELEMENTARY FUNCTION LIBRARY

CEL87.LIB contains commonly used floating point functions. It is used along with the 8087 numeric coprocessor or the 8087 emulator and it provides a complete package of elementary functions, giving valid results for all appropriate inputs. This library provides PL/M-86 and ASM-86 users all the math functions supported intrinsically by the Fortran-86. Following is a summary of CEL87 functions, grouped by functionality.

Rounding and Truncation Functions:

mqerIEX, mqerIE2, and mqerIE4 round a real number to the nearest integer; to the even integer if there is a tie. The answer returned is real, a 16-bit integer or a 32-bit integer respectively.
mqerIAX, mqerIA2, mqerIA4 round a real number to the nearest integer, to the integer away from zero if there is a tie; the answer returned is real, a 16-bit integer or a 32-bit integer, respectively.
mqerICX, mqerIC2, mqerIC4 truncate the fractional part of a real input; the answer is real, a 16-bit integer or a 32-bit integer, respectively.

Logarithmic and Exponential Functions:

mqerLGD computes decimal (base 10) logarithms.
mqerLGE computes natural (base e) logarithms.
mqerEXP computes exponentials to the base e.
mqerY2X computes exponentials to any base.
mqerYI2 raises an input real to a 16-bit integer power.
mqerYI4 is as mqerYI2, except to a 32-bit integer power.
mqerYIS is as mqerYI2, but it accommodates PL/M-86 users.

Trigonometric and Hyperbolic Functions:

mqerSIN, mqerCOS, mqerTAN compute sine, cosine, and tangent.
mqerASN, mqerACS, mqerATN compute the corresponding inverse functions.
mqerSNH, mqerCSH, mqerTNH compute the corresponding hyperbolic functions.
mqerAT2 is a special version of the arc tangent function that accepts rectangular coordinate inputs.

Other Functions:

mqerDIM is FORTRAN's positive difference function.
mqerMAX returns the maximum of two real inputs.
mqerMIN returns the minimum of two real inputs.
mqerSGH combines the sign of one input with the magnitude of the other input.
mqerMOD computes a modulus, retaining the sign of the dividend.
mqerRMD computes a modulus, giving the value closest to zero.

DCON87.LIB
THE DECIMAL CONVERSION LIBRARY

DCON87.LIB is a library of procedures which convert binary representations of floating point numbers and ASCII-encoded string of digits.

The binary-to-decimal procedure mqcBIN DECLOW accepts a binary number in any of the formats used for the representation of floating point numbers in the 8087. Because there are so many output formats for floating point numbers, mqcBIN_DECLOW does not attempt to provide a finished, formatted text string. Instead, it provides the "building blocks" for you to use to construct the output string which meets your exact format specification.

The decimal-to-binary procedure mqcDEC_BIN accepts a text string which consists of a decimal number with optional sign, decimal point, and/or power-of-ten exponent. It translates the string into the caller's choice of binary formats.

Decimal-to-binary procedure mqcDECLOW_BIN is provided for callers who have already broken the decimal number into its constituent parts.

The procedures mqcLONG_TEMP, mqcSHORT_TEMP, mqcTEMP_LONG, and mqcTEMP_SHORT convert floating point numbers between the longest binary format, TEMP_REAL, and the shorter formats.

EH87.LIB
THE ERROR HANDLER MODULE

EH87.LIB is a library of five utility procedures which a user can utilize for writing trap handlers. Trap handlers are called when an unmasked 8087 error occurs.

The 8087 error reporting mechanism can be used not only to report error conditions, but also to let software implement IEEE standard options not directly supported by the chip. The three such extensions to the 8087 are: normalizing mode, non-trapping not-a-number (NaN), and non-ordered comparison. The utility procedures support these extra features.

DECODE is called near the beginning of the trap handler. It preserves the complete state of the 8087, and also identifies what function called the trap handler, and returns available arguments and/or results. DECODE eliminates much of the effort needed to determine what error caused the trap handler to be called.

NORMAL provides the "normalizing mode" capability for handling the "D" exception. By calling NORMAL in your trap handler, you eliminate the need to write code in your application program which tests for non-normal inputs.

SIEVE provides two capabilities for handling the "I" exception. It implements non-trapping NaN's and non-ordered comparisons. These two IEEE standard features are useful for diagnostic work.

ENCODE is called near the end of the trap handler. It restores the state of the 8087 saved by DECODE, and performs a choice of concluding actions, by either retrying the offending function or returning a specified result.

FILTER calls each of the above four procedures. If your error handler does nothing more than detect fatal errors and implement the features supported by SIEVE and NORMAL, then your interface to EH87.LIB can be accomplished with a single call to FILTER.

E8087
THE FULL 8087 EMULATOR

E8087 is an object module that functionally emulates the 8087 coprocessor chip. It is ideal for use during prototyping and debugging floating point programs. However, the target system should use the 8087 component because it executes 1000 times faster and uses significantly less memory.

E8087.LIB, 8087.LIB, 87NULL.LIB INTERFACE LIBRARIES

E8087. LIB, 8087.LIB and 87NULL. LIB libraries configure a user's application program for his run-time environment: running with the emulator, with the 8087 component or without floating point arithmetic, respectively.

SPECIFICATIONS

OPERATING ENVIRONMENT

8086/8088 Based Microcomputer System

DEVELOPMENT ENVIRONMENT

Required Hardware

Intellec Microcomputer Development System
— Model 800
— Series II (Models 220, 225 or equivalent)
— Series III
64K Bytes of RAM Memory

Minimum One Diskette Drive
— Single or Double* Density

System Console
— CRT or Hardcopy Interactive Device

Optional Hardware

Universal PROM Programmer*
Line Printer*

Required Software

For Series II:
8086/8088 Software Development Package—MDS-308, MDS-309 or MDS-311

Documentation Package

Numeric Support Library Manual

Shipping Media

1 Single and 1 Double Density Diskette

*Recommended

ORDERING INFORMATION

Part Number	Description
MDS* 319	8087 Support Library

Requires Software License

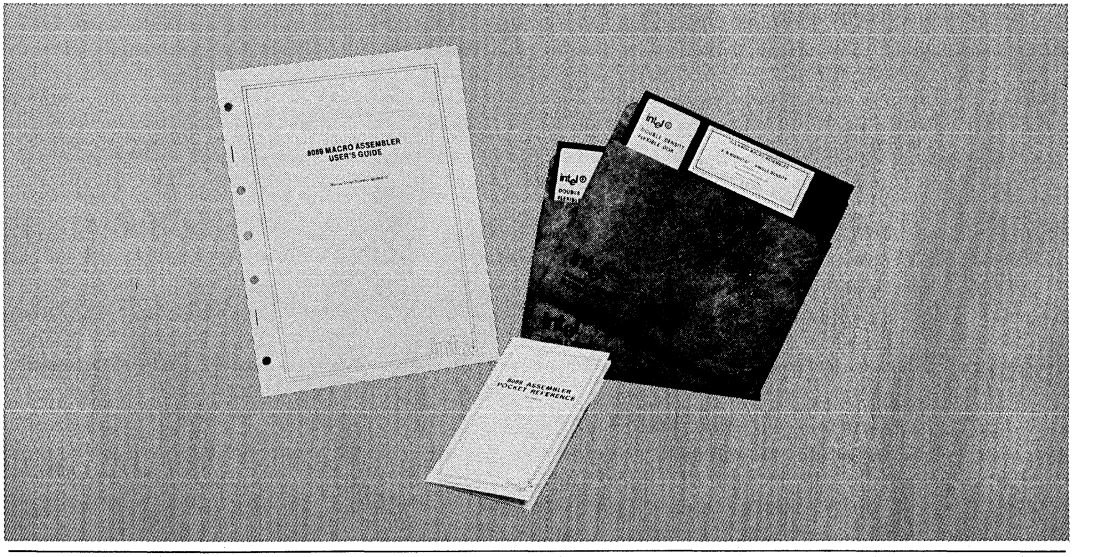
*MDS is an ordering code only and is not used as a product name or trademark. MDS is a registered trademark of Mohawk Data Sciences Corporation.

8089 IOP
SOFTWARE SUPPORT PACKAGE

- **Program Generation for the 8089 I/O Processor on the Intellec® Microcomputer Development System**
- **Contains 8089 Macro Assembler, plus Relocation and Linkage Utilities**
- **Relocatable Object Module Compatible with All iAPX 86 and iAPX 88 Object Modules**
- **Fully Supports Symbolic Debugging with the RBF-89 Software Debugger**
- **Supports 8089-Based Addressing Modes with a Structure Facility that Enables Easy Access to Based Data**
- **Powerful Macro Capabilities**
- **Provides Timing Information in Assembly Listing**
- **Fully Detailed Set of Error Messages**

The IOP Software Support Package extends Intellec Microcomputer Development System support to the 8089 I/O Processor. The macro assembler translates symbolic 8089 macro assembly language instructions into relocatable machine code. The relocation and linkage utilities provide compatibility with iAPX 86, iAPX 88, and 8089 modules, and make structured, modular programming easier.

The macro assembler also provides symbolic debugging capability when used with the RBF-89 software debugger. 8089 program modularity is supported with inter-segment jumps and calls. The macro assembler also provides instruction cycle counts in the listing file, for giving the programmer execution timing information. The programs in the 8089 Software Support Package run on any Intellec Series II or Model 800 with 64K bytes of memory.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, CREDIT, Intellec, Multibus, i, iSBC, Multimodule, ICE, iSBX, PROMPT, iCS, iRMX, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Micromap, μScope and the combination of iCE, iSBC, iSBX, MCS, or RMX and a numerical suffix.
® Intel Corporation 1979, 1980

9800999-02 Rev. B

FUNCTIONAL DESCRIPTION

The IOP Software Support Package contains:

ASM89 —The 8089 Macro Assembler.

LINK86 —Resolves control transfer references between 8089 object modules, and data references in 8086, 8088, and 8089 modules.

LOC86 —Assigns absolute memory addresses to 8089 object modules.

OH86 —Converts absolute object modules to hexadecimal format.

UPM —The Universal PROM Mapper, which supports PROM programming in all iAPX 86/11 and iAPX 88/11 applications.

ASM89 translates symbolic 8089 macro assembly language instructions into the appropriate machine codes. The ability to refer to both program and data addresses with symbolic names makes it easier to develop and modify programs, and avoids the errors of hand translation.

The powerful macro facility allows frequently used code sequences to be referred to by a single name, so that any changes to that sequence need to be made in only one place in the program. Common code sequences that differ only slightly can also be referred to with a macro call, and the differences can be substituted with macro parameters.

ASM89 provides symbolic debugging information in the object file. The RBF-89 debugger makes use of this information, so the programmer can symbolically debug 8089 programs. ASM89 also provides cycle counts for each instruction in the assembly listing file (see Table 1). These cycle counts help the programmer determine how long a particular routine or code sequence will take to execute on the 8089.

ASM89 provides relocatable object module compatibility with the 8086 and 8088 microprocessors. This object module compatibility, along with the 8086/8088 relocation and linkage utilities, facilitates the designing of iAPX 86/11 and iAPX 88/11 systems.

ASM89 fully supports the based addressing modes of the 8089. A structure facility allows the user to define a template that enables accessing of based data symbolically.

SPECIFICATIONS

Operating Environment

REQUIRED HARDWARE
Intellec® Microcomputer Development System
—Model 800
—Series II (Models 220, 225 or equivalent)

64K Bytes of RAM Memory

Minimum One Diskette Drive
—Single or Double* Density

System Console
—CRT or Hardcopy Interactive Device

OPTIONAL HARDWARE
Universal PROM Programmer*
Line Printer*

REQUIRED SOFTWARE
ISIS-II Diskette Operating System
—Single or Double Density

*Recommended

Documentation Package

8089 Macro Assembler User's Guide (9800938)

8089 Macro Assembler Pocket Reference (9800936)

MCS-86 Software Development Utilities Operating Instructions for ISIS-II Users (9800639)

Universal PROM Programmer User's Manual (9800819)

Shipping Media

—Single and Double Density Diskettes

ORDERING INFORMATION

Part Number	Description
MDS*-312	8089 IOP Software Support Package

Requires Software License

***MDS** is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

Table 1. Sample Program Listing

```
8089 MACRO ASSEMBLER


ISIS-II 8089 MACRO ASSEMBLER X105 ASSEMBLY OF MODULE TASK
OBJECT MODULE PLACED IN :F1:TASK.OBJ
ASSEMBLER INVOKED BY:  asm89 :f1:task.a89 gen macro debug pagewidth(132) print(:f1:taskx.lst)


LOC    OBJECT CODE              TIMING    INC MAC  LINE SOURCE

                                                    1  ;*************************************************
                                                    2  ;*                                               *
                                                    3  ;*              8089 TASK PROGRAM                *
                                                    4  ;*                                               *
                                                    5  ;*************************************************
                                                    6
                                                    7  name TASK
0000                                                8  TASK segment
                                                    9  ;
                                                   10  ;       In the first part of this sample program data is moved from
                                                   11  ;       8086 system RAM to memory local to the 8089 IOP.  In the second
                                                   12  ;       part, the data is moved from the local memory to a data port
                                                   13  ;       also in the 8089 I/O space.
                                                   14  ;
  F800                                             15  data@port@8251  equ     0c800h          ;8251 DP on 8089 local bus
  C801                                             16  command@port@8251 equ   0c801h          ;8251 CP on 8089 local bus
  8200                                             17  buffer@8089     equ     8200h           ;RAM buffer in 8089 I/O space
                                                   18  ;
                                                   19  extrn   buffer@8086                     ;RAM buffer in 8086 system memory
                                                   20  extrn   y                               ;location of the buffer count
                                                   21  ;
                                                   22  %*define (macro_1)
                                                   23  (       movi    gb,buffer@8089          ; Move buffer address into GB
                                                   24          lpdi    gc,y                    ; Load pointer to count into GC
                                                   25          movb    bc,[gc]                 ; Move byte count into BC
                                                   26  )
                                                   27  %*define (macro_2(param_1, param_2)) local loop
                                                   28  (       inc     %param_1                ; Increment pointer into source
                                                   29          dec     %param_2                ; Decrement byte count
                                                   30          jnz     %param_2,%loop          ; Loop back if byte count > 0
                                                   31  )
0000   1100 00000000            30    46          32  ONE:    lpdi    ga,buffer@8086          ; Load register GA with address
                                                   33                                          ; of 8086 buffer
                                                   34  %macro_1
0006   3130 8002                47    71    +1    35          movi    gb,buffer@8089          ; Move buffer address into GB
0008   5100 00000000            77   117    +1    36          lpdi    gc,y                    ; Load pointer to count into GC
0010   6002                    92   139    +1    37          movb    bc,[gc]                 ; Move byte count into BC
                                             +1    38
0012   8098 00CD               124   185          39  loop80: movb    [gb],[ga]               ; Move byte from 8086 to 8089 buffer
0016   0030                    134   202          40          inc     ga                      ; Increment pointer into 8086 buffer
                                                   41  %macro_2(gb,gc)
0018   2030                    144   219    +1    42          inc     %PARAM_1
                                             +2    43      gb                  ; Increment pointer into source
001A   403C                    154   236    +1    44          dec     %PARAM_2
                                             +2    45      gc                  ; Decrement byte count
001C   4048 F3                 173   259    +1    46          jnz     %PARAM_2
                                             +2    47      gc,%LOOP
                                             +2    48          LOOP80              ; Loop back if byte count > 0
                                             +1    49
001F   1130 00C8               191*  294          50  TWO:    movi    ga,data@port@8251       ; load GA with address of 8251 DP
0023   1130 01C8               209   309          51          movi    ga,command@port@8251    ; load GC with address of 8251 CP
                                                   52  %macro_1
0027   3130 8002               227   334    +1    53          movi    gb,buffer@8089          ; Move buffer address into GB
0028   5100 00000000           242   380    +1    54          lpdi    gc,y                    ; Load pointer to count into GC
0031   6002                    261   402    +1    55          movb    bc,[gc]                 ; Move byte count into BC
                                             +1    56
0033   880A FD                 286*  434          57  loop81: jnbt    [gc],0,loop81           ; loop until 8251 transmit ready
0036   8091 00CC               338   488          58          movb    [ga],[gb]               ; move message into buffer
                                                   59  %macro_2(gb,gc)
003A   2030                    348   497    +1    60          inc     %PARAM_1
                                             +2    61      gb                  ; Increment pointer into source
003C   403C                    358   514    +1    62          dec     %PARAM_2
                                             +2    63      gc                  ; Decrement byte count
003E   4048 F2                 377   537    +1    64          jnz     %PARAM_2
                                             +2    65      gc,%LOOP
                                             +2    66          LOOP81              ; Loop back if byte count > 0
                                             +1    67
0041   2048                    399   562          68          hlt
0043                                               69  TASK    ends
                                                   70  END


ASSEMBLY COMPLETE, NO ERRORS FOUND
```

ICE-86A™
iAPX 86 IN-CIRCUIT EMULATOR

- **Real-Time In-Circuit Emulation of iAPX 86 Microsystems**
- **Emulate Both Minimum and Maximum Modes of 8086 CPU**
- **Full Symbolic Debugging**
- **Breakpoints to Halt Emulation on a Wide Variety of Conditions**
- **Comprehensive Trace of Program Execution**
- **Disassembly of Trace or Program Memory from Object Code into Assembler Mnemonics**
- **Software Debugging With or Without User System**
- **Handles Full 1 Megabyte Addressability of iAPX 86**
- **Enhance Existing ICE-86™ Emulators to ICE-86A™ Capabilities with ICE-86U™ Upgrade Package**

The Intel® ICE-86A In-Circuit Emulator provides sophisticated hardware and software debugging capabilities for iAPX 86 microsystems and iAPX 86 Single-Board Computers. These capabilities include In-Circuit Emulation for the 8086 Central Processing Unit plus extensions to debug systems including the 8089 I/O Processor and 8087 Numeric Processor Extension. The emulator includes three circuit boards which reside in any Intellec® Microcomputer Development System. A cable and buffer box connect the Intellec system to the user system by replacing the user's 8086, thus extending powerful Intellec system debugging functions into the user system. Using the ICE-86A module, the designer can execute prototype 8086 or 8089 software in continuous or single-step modes and can substitute blocks of Intellec system memory for user equivalents. Breakpoints allow the user to stop emulation on user-specified conditions of the iAPX 86 system, and the trace capability gives a detailed history of the program execution prior to the break. All user access to the prototype system software may be done symbolically by referring to the source program variables and labels.

The ICE-86U In-Circuit Emulator upgrade package converts any existing ICE-86 module (non-A version) to the capabilities of an ICE-86A module.



The following are trademarks of Intel Corporation and its affiliates and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{l}m, iCS, ICE, UPI, BXP, iSBC, iSBX, Insite, iRMX, System 2000, CREDIT, iRMX/80, MULTIBUS, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTIMODULE, and the combination of MCS, ICE, iSBC, iRMX or iCS and a numerical suffix. © INTEL CORPORATION, 1981.

9800931-01

INTEGRATED HARDWARE/SOFTWARE DEVELOPMENT

The ICE-86A emulator allows hardware and software development to proceed interactively. This is more effective than the traditional method of independent hardware and software development followed by system integration. With the ICE-86A module, prototype hardware can be added to the system as it is designed. Software and hardware testing occurs while the product is being developed.

Conceptually, the ICE-86A emulator assists three stages of development:

1. It can be operated without being connected to the user's system, so the ICE-86A module's debugging capabilities can be used to facilitate program development before any of the user's hardware is available.

2. Integration of software and hardware can begin when any functional element of the user system hardware is connected to the 8086 socket. Through ICE-86A emulator mapping capabilities, Intellec memory, ICE module memory, or diskette memory can be substituted for missing prototype memory. Time-critical program modules are debugged before hardware implementation by using the 2K-bytes of high-speed ICE-resident memory. As each section of the user's hardware is completed, it is added to the prototype. Thus each section of the hardware and software is "system" tested as it becomes available.

3. When the user's prototype is complete, it is tested with the final version of the user system software. The ICE-86A module is then used for real-time emulation of the 8086 to debug the system as a completed unit.

Thus the ICE-86A module provides the user with the ability to debug a prototype or production system at any stage in its development without introducing extraneous hardware or software test tools.

SYMBOLIC DEBUGGING

Symbols and PL/M statement numbers may be substituted for numeric values in any of the ICE-86A emulator commands. This allows the user to make symbolic references to I/O ports, memory addresses, and data in the user program. Thus the user need not remember the addresses of variables or program subroutines.

Symbols can be used to reference variables, procedures, program labels, and source statements. A variable can be displayed or changed by referring to it by name rather than by its absolute location in memory. Using symbols for statement labels, program labels, and procedure names simplifies both tracing and breakpoint setting. Disassembly of a section of code from either trace or program memory into its assembly mnemonics is readily accomplished.

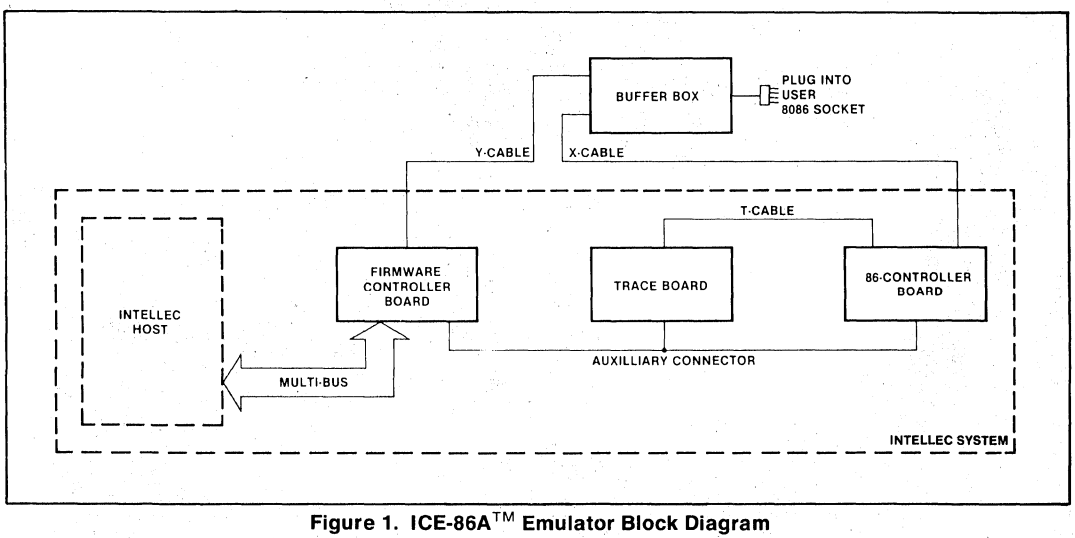
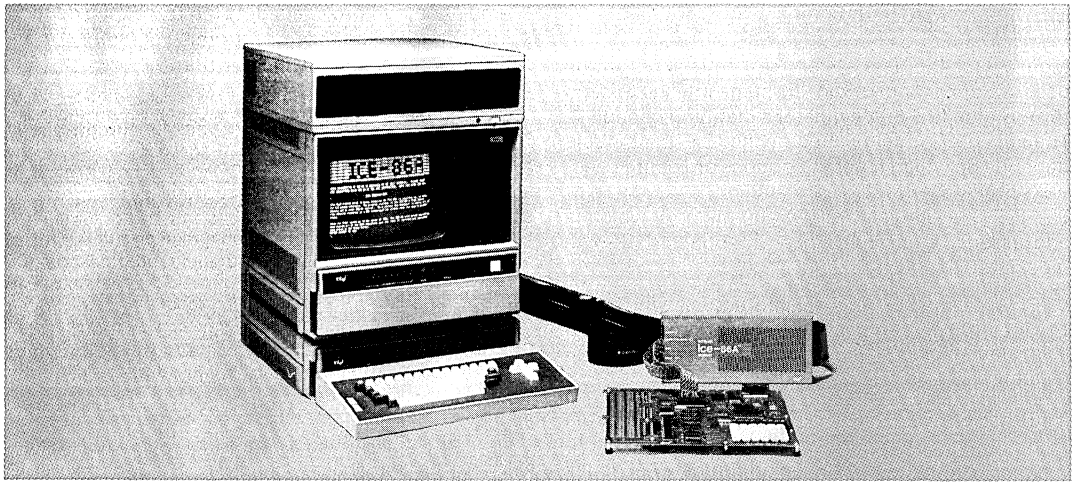


Figure 1. ICE-86A™ Emulator Block Diagram



A typical iAPX 86 development configuration. It is based on Intellec® Series III Development System, which hosts the ICE-86A™ emulator. The ICE-86A™ module is shown connected to a user prototype system, in this case, an SDK-86.

Furthermore, each symbol may have associated with it one of the data types BYTE, WORD, INTEGER, SINTEGER (for short, 8-bit integer), POINTER, REAL, DREAL, or TREAL. Thus the user need not remember the type of a source program variable when examining or modifying it. For example, the command "!VAR" displays the value in memory of variable VAR in a format appropriate to its type, while the command "!VAR = !VAR + 1" increments the value of the variable.

The user symbol table generated along with the object file during a PL/M-86, PASCAL-86 or FORTRAN-86 compilation or an ASM-86 assembly is loaded into memory along with the user program which is to be emulated. The user can utilize the available symbol table space more efficiently by using the SELECT option to choose which program modules will have symbols loaded in the symbol table. The user may also add to this symbol table any additional symbolic values for memory addresses, constants, or variables that are found useful during system debugging.

The ICE-86A module provides access through symbolic definition to all of the 8086 registers and flags. The READY, NMI, $\overline{\text{TEST}}$, HOLD, RESET, INTR, MN/$\overline{\text{MX}}$, and $\overline{\text{RQ}}$/$\overline{\text{GT}}$ pins of the 8086 can also be read. Symbolic references to key ICE-86A emulation information are also provided.

MACROS AND COMPOUND COMMANDS

The ICE-86A module provides a programmable diagnostic facility which allows the user to tailor its operation using macro commands and compound commands.

A macro is a set of ICE-86A commands which is given a single name. Thus, a sequence of commands which is executed frequently may be invoked simply by typing in a single command. The user first defines the macro by entering the entire sequence of commands which he wants to execute. He then names the macro and stores it for future use. He executes the macro by typing its name and passing up to ten parameters to the commands in the macro. Macros may be saved on a disk file for use in subsequent debugging sessions.

Compound commands provide conditional execution of commands (IF), and execution of commands until a condition is met or until they have been executed a specified number of times (COUNT, REPEAT).

Compound commands and macros may be nested any number of times.

MEMORY MAPPING

Memory for the user system can be resident in the user system or "borrowed" from the Intellec System through the ICE-86A emulator's mapping capability. The speed of run emulation by the ICE-86A module depends on which mapping options are being used.

The ICE-86A emulator allows the memory which is addressed by the 8086 to be mapped in 1K-byte blocks to:

1. Physical memory in the user's system, which provides 100 percent real-time emulation at the user-system clock rate (up to 5 MHz) with no wait states.
2. Either of two 1K-byte blocks of ICE-86A module high-speed memory, which allow nearly full-speed emulation (with two additional wait states per 8086-controlled bus cycle).
3. Intellec System memory, which provides emulation at approximately 0.02 percent of real-time with a 5 MHz clock.
4. A random-access diskette file, with emulation speed comparable to Intellec System memory, except the emulation must wait when a new page is accessed on the diskette.

The user can also designate a block of memory as non-existent. The ICE-86A module issues an error message when any such "guarded" memory is addressed by the user program.

As the user prototype progresses to include memory, emulation becomes real time.

OPERATION MODES

The ICE-86A software is a RAM-based program that provides the user with easy-to-use commands for initiating emulation, defining breakpoints, controlling trace data collection, and displaying and controlling system parameters. ICE-86A commands are configured with a broad range of modifiers which provide the user with maximum flexibility in describing the operation to be performed.

Emulation

Emulation commands to the ICE-86A emulator control the process of setting up, running and halting an emulation of the user's iAPX 86 System. Breakpoints and tracepoints enable the ICE-86A module to halt emulation and provide a detailed trace of execution in any part of the user's program. A summary of the emulation commands is shown in Table 1.

Table 1. Summary of ICE-86A™ Emulation Commands

Command	Description
GO	Initializes emulation and allows the user to specify the starting point and breakpoints. Example: GO FROM .START TILL .DELAY EXECUTED where START and DELAY are statement labels.
STEP	Allows the user to single-step through the program.

Breakpoints: The ICE-86A module has two breakpoint registers that allow the user to halt emulation when a specified condition is met. The breakpoint registers may be set up for execution or non-execution breaking. An execution breakpoint consists of a single address which causes a break whenever the 8086 executes from its queue an instruction byte which was obtained from the address. A non-execution breakpoint causes an emulation break when a specified condition other than an instruction execution occurs. A non-execution breakpoint condition, using one or both breakpoint registers, may be specified by any one of or a combination of:

1. *A set of address values.* Break on a set of address values has three valuable features:
   a. Break on a single address.
   b. The ability to set any number of breakpoints within a limited range (1024 bytes maximum) of memory.
   c. The ability to break in an unlimited range. Execution is halted on any memory access to an address greater than (or less than) any 20-bit breakpoint address.
2. *A particular status of the 8086 bus* (one or more of: memory or I/O read or write, instruction fetch, halt, or interrupt acknowledge).
3. *A set of data values* (features comparable to break on a set of address values, explained in point one).
4. *A segment register* (break occurs when the register is used in an effective address calculation).

Emulation break can also be set to occur on an external signal condition. An external breakpoint match output and emulation status lines are provided on the buffer box. These allow synchronization of other test equipment when a break occurs or when emulation is begun.

Tracepoints: The ICE-86A module has two tracepoint registers which establish match conditions to conditionally start and stop trace collection. The trace information is gathered at least twice per bus cycle, first when the address signals are valid and second when the data signals are valid. If the 8086 execution queue is otherwise active, additional frames of trace are collected.

Each trace frame contains the 20 address/data lines and detailed information on the status of the 8086. The trace memory can store 1,023 frames, or an average of about 300 bus cycles, providing ample data for detemining how the 8086 was reacting prior to emulation break. The trace memory contains the last 1,023 frames of trace data collected, even if this spans several separate emulations. The user has the option of displaying each frame of the trace data or displaying by instruction in actual ASM-86 Assembler mnemonics. Unless the user chooses to disable trace, the trace information is always available after an emulation.

Interrogation and Utility

Interrogation and utility commands give the user convenient access to detailed information about the user program and the state of the 8086 that is useful in debugging hardware and software. Changes can be made in both memory and the 8086 registers, flags, input pins, and I/O ports. Commands are also provided for various utility operations such as loading and saving program files, defining symbols and macros, displaying trace data, setting up the memory map, and returning control to ISIS-II. A summary of the basic interrogation and utility commands is shown in Table 2.

Table 2. Selected ICE-86A™ Module Interrogation and Utility Commands

Memory/Register Commands
Display or change the contents of:
- Memory
- 8086 Registers
- 8086 Status flags
- 8086 Input pins
- 8086 I/O ports
- ICE-86A Pseudo-Registers (e.g. emulation timer)

Memory Mapping Commands
Display, declare, set, or reset the ICE-86A memory mapping.

Symbol Manipulation Commands
Display any or all symbols, program modules, and program line numbers and their associated values (locations in memory).

Set the domain (choose the particular program module) for the line numbers.

Define new symbols as they are needed in debugging.

Remove any or all symbols, modules, and program statements.

Change the value of any symbol.

Select program modules whose symbols will be used in debugging.

TYPE
Assign or change the type of any symbol in the symbol table.

DASM
Disassemble user program memory into ASM-86 Assembler mnemonics.

RQ/GT
Set or display the status of the Request/Grant facility which enables the ICE-86A module to share the system bus with coprocessors.

BUS
Display which device in the user's iAPX 86 system is currently master of the system bus.

CAUSE
Display the cause of the most recent emulation break.

PRINT
Display the specified portion of the trace memory.

LOAD
Fetch user symbol table and object code from the input file.

EVALUATE
Display the value of an expression in binary, octal, decimal, hexadecimal, and ASCII.

CLOCK
Select the internal (ICE-86A module provided, for stand-alone mode only) or an external (user-provided) system clock.

RWTIMEOUT
Allows the user to time out READ/WRITE command signals based on the time taken by the 8086 to access Intellec memory or diskette memory.

ENABLE/DISABLE RDY
Enable or disable logical AND of ICE-86A emulator Ready with the user Ready signal for accessing Intellec memory, ICE memory, or diskette memory.

iAPX 86/20 DEBUGGING

The ICE-86A module has the extended capabilities to debug iAPX 86/20 microsystems which contain both the 8086 microprocessor and the 8087 Numeric Processor Extension (NPX). An iAPX 86/20 system is configured in the 8086's "maximum" mode and communication between the processors is accomplished through the $\overline{RQ}/\overline{GT}$ signals. Debugging can be done either using the 8087 chip itself (in which case the 8086 ESCAPE instruction is interpreted as a floating point instruction) or using the 8087 software emulator E8087 (where the 8086 INTERRUPT instruction is interpreted as a floating point instruction). Three new data types are defined to use the NPX:

REAL (4 byte Short Real)
DREAL (8 byte Long Real)
TREAL (10 byte Temporary Real)

While the 8087 NPX is not a programmable part, it does interact closely with the 8086 and can execute instructions in parallel with it. The ICE-86A module provides information about the relative timing of instruction execution in each processor so that the complete system can be debugged. Other debugging capabilities available through the ICE-86A module are: symbolically disassemble NPX call instructions from memory or trace history; display or change the control, status and flag values of the NPX; display the NPX stack either in hexadecimal or disassembled form; and display the last instruction address, last operand, and last operand address.

iAPX 86/11 DEBUGGING

The 8089 Real-Time Breakpoint Facility (RBF-89) is an extension of the ICE-86A emulator that aids in testing and trouble-shooting iAPX 86/11 systems designed around a combination of the 8086 CPU and the 8089 Input/Output Processor (IOP). RBF-89 interrogates 8089 registers, sets breakpoints in 8089 programs, and performs its other functions by preparing special control blocks in application system memory. It then issues input/output channel-attention commands to the 8089 in the user's system to perform these functions. While using the RBF-89 extension, the user can also enter and execute the other standard ICE-86A emulator commands.

RBF-89 allows the user to load his application (channel) program from diskette into 8089 IOP memory and execute it in real time. The program can reside in either local (system) RAM (accessible by both the 8086 and 8089 microprocessors), or remote RAM (accessible by the 8089 IOP only). The user may request execution to begin at any location and continue until normal termination, a specified breakpoint is reached, or the program is otherwise aborted. If a program is modified during a debugging session, RBF-89 can save the latest version by copying it from application system memory to a diskette file.

Breakpoints

RBF-89 supports setting up to twelve breakpoints (six per 8089 channel) in the user program. RBF-89 implements each breakpoint by inserting a HALT instruction at the breakpoint location, while saving the overwritten instruction in temporary storage. When a breakpoint is reached during program execution the program halts. At this point the user can examine 8089 registers, flags, and memory, and optionally resume program execution. The invoked breakpoint address is recorded in one of two breakpoint registers—one register for each 8089 channel. Through simple RBF-89 commands the user can display or change the contents of these registers.

Symbolic Debugging

As in the ICE-86A emulator, the RBF-89 extension accepts symbolic references for variables and labels, including symbols in the symbol table generated by the ASM-89 assembler.

Through RBF-89, the user can display and change the contents of :

— memory, which can be displayed as either numeric data or disassembled (8089 assembly-language mnemonic) code.

— all 8089 registers except the channel control pointer (CCP) and status flags.

Multiprocessor Operation

The ICE-86A emulator and RBF-89 support 8089 configurations in both local and remote modes. The ICE-86A emulator may be operating either in minimum or maximum mode. In maximum mode, the 8086 $\overline{RQ}/\overline{GT}$ lines are employed. This is required for the 8089 local mode configuration to provide local bus arbitration between the two processors. Using RBF-89, the user can:

Set $\overline{RQ}/\overline{GT}$ to operate for a local or remote configuration.

Display status to determine which processor controls the system bus.

Start and halt 8089 channel programs.

RBF-89 permits the 8089 and emulated 8086 to run simultaneously as well as sequentially. The user can specify breakpoints and begin program execution in three operating sequences:

Set breakpoints, start the 8089, and return control to the console until a breakpoint is reached or the program runs to completion or is aborted. Use this sequence when the 8086 and 8089 programs do not need to be executed simultaneously.

Set breakpoints, start the 8089, return control to the console, and start the 8086. This sequence lets both microprocessors run simultaneously.

Set breakpoints, start the 8086, and allow that program to drive the 8089 program in a master/slave relationship. This sequence would be used, for instance, to verify the 8086 communication driver program.

RBF-89 System Components

RBF-89 is furnished as a superset of the ICE-86A emulator software. Its main components are:

A HOST PROGRAM that resides in Intellec development system RAM, where it serves as an extension of the ICE-86A emulator's software driver. This program, executed by the development system, translates the user's keyboard input into low-level directives that can be processed by the RBF-89 control program (described below), and converts information supplied by the control program into easily understood display output.

A CONTROL PROGRAM that resides in ICE-86A emulator memory. Running on the emulator's 8086 microprocessor, the control program monitors such operations as preparing program control blocks for communication with the 8089 microprocessor; issuing commands to the 8089 to start, terminate, and continue the 8089 task program; and directing the 8089 to start execution of the RBF-89 utility program (described below).

A UTILITY PROGRAM that resides in the 8089 RAM in the user's prototype application system. This program, running on the 8089, reads and writes data to and from 8089 memory and registers, and sets and removes breakpoints in the user's task program.

The 200 bytes of RAM required by the utility program must be accessible to both the ICE-86A emulator and the 8089.

DC CHARACTERISTICS OF THE ICE-86A™ MODULE USER CABLE

1. Output Low Voltages [V_{OL}(Max)=0.4V]

	I_{OL} (Min)
AD0-AD15	12 mA (24 mA @ 0.5V)
A16/S3-A19/S7, $\overline{BHE}$/S7, $\overline{RD}$, $\overline{LOCK}$, QS0, QS1, $\overline{S0}$, $\overline{S1}$, $\overline{S2}$, $\overline{WR}$, M/$\overline{IO}$, DT/$\overline{R}$, $\overline{DEN}$, ALE, $\overline{INTA}$	8 mA (16 mA @ 0.5V)
HLDA	7 mA
$\overline{RQ}/\overline{GT}$	16 mA

2. Output High Voltages [V_{OH} (Min)=2.4V]

	I_{OH} (Min)
AD0-AD15	−3 mA
A16/S3-A19/S7, $\overline{BHE}$/S7, $\overline{RD}$, $\overline{LOCK}$, QS0, QS1, S0, $\overline{S1}$, $\overline{S2}$, $\overline{WR}$, M/IO, DT/R, $\overline{DEN}$, ALE, $\overline{INTA}$, HLDA	−2.6 mA
$\overline{RQ}/\overline{GT}$	250 mA

3. Input Low Voltages [V_{IL} (Max)=0.8V]

	I_{IL} (Max)
AD0-AD15	−0.2 mA
NMI, CLK	−0.4 mA
READY	−0.8 mA
INTR, HOLD, $\overline{TEST}$, RESET	−1.4 mA
MN/$\overline{MX}$ (0.1 μf to GND)	−3.3 mA

4. Input High Voltages [V_{IH} (Min)=2.0V]

	I_{IH} (Max)
AD0-AD15	80 μA
NMI, CLK	20 μA
READY	40 μA
INTR, HOLD, $\overline{TEST}$, RESET	−0.4 mA
MN/$\overline{MX}$ (0.1 μF to GND)	−1.1 mA

5. No current is taken from the user circuit at V_{CC} pin.

SPECIFICATIONS

ICE-86A Operating Environment

REQUIRED HARDWARE
Intellec microcomputer development system with:

1. Three adjacent slots for the ICE-86A module.
2. 64K bytes of Intellec memory. If user prototype program memory is desired, additional memory above the basic 64K is required.

System console
Intellec diskette operating system
ICE-86A module

REQUIRED SOFTWARE
System Monitor
ISIS-II, version 3.4 or subsequent
ICE-86A software

Equipment Supplied

Printed circuit boards (3)
Interface cable and emulation buffer module
Operator's manual
ICE-86A software, diskette-based

Emulation Clock

User system clock up to 5 MHz or 2 MHz ICE-86A internal clock in stand-alone mode

Physical Characteristics

PRINTED CIRCUIT BOARDS
Width: 12.00 in (30.48 cm)
Height: 6.75 in (17.15 cm)
Depth: 0.50 in (1.27 cm)
Packaged Weight: 9.00 lb (4.10 kg)

Electrical Characteristics

DC POWER
V_{CC} = +5V +5%−1%
I_{CC} = 17A maximum; 11A typical
V_{DD} = +12V ±5%
I_{DD} = 120 mA maximum; 80 mA typical
V_{BB} = −10V ±5% or −12V ±5% (optional)
I_{BB} = 25 mA maximum; 12 mA typical

Environmental Characteristics

OPERATING TEMPERATURE
0° to 40°C

OPERATING HUMIDITY
Up to 95% relative humidity without condensation.

ORDERING INFORMATION

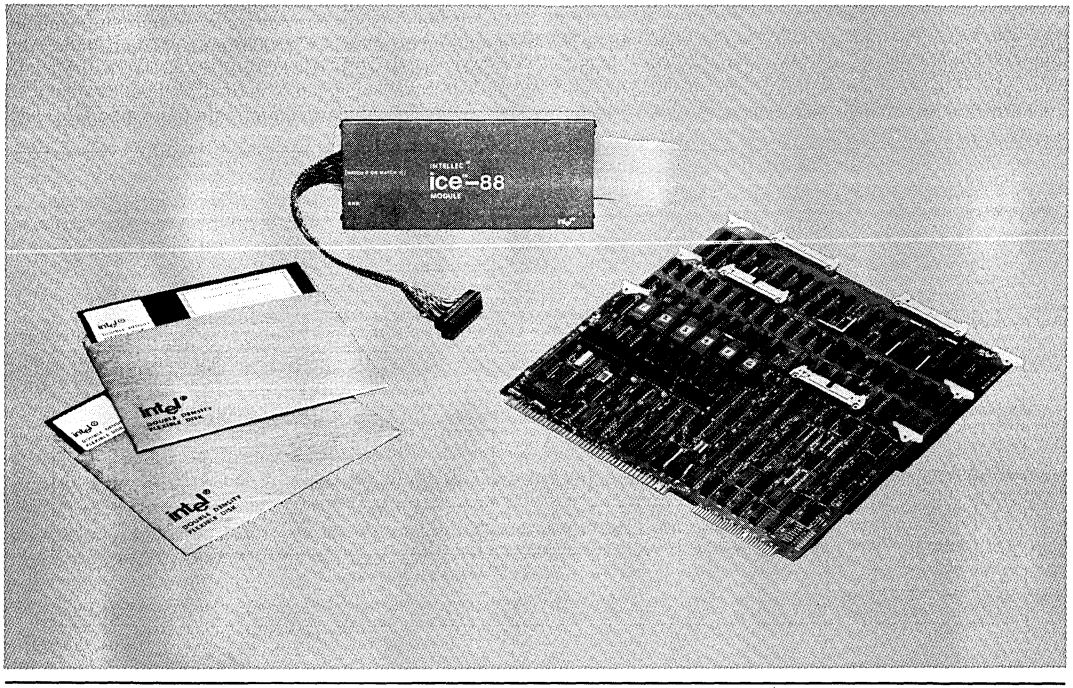
Part Number	Description
MDS*-86A-ICE	iAPX 86 microsystem in-circuit emulator, cable assembly, and interactive software
MDS*-86U-ICE	Upgrade kit to convert ICE-86 emulators to ICE-86A emulator capabilities.

*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.

ICE-88™
8088 IN-CIRCUIT EMULATOR

- **Hardware In-Circuit Emulation**
- **Full Symbolic Debugging**
- **Breakpoints to Halt Emulation on a Wide Variety of Conditions**
- **Comprehensive Trace of Program Execution, Both Conditional and Unconditional**
- **Disassembly of Trace or Memory from Object Code into Assembler Mnemonics**
- **2K Bytes of High Speed ICE-88™ Mapped Memory**
- **Software Debugging with or without User System**
- **Handles Full 1 Megabyte Addressability of 8088**
- **Compound Commands**
- **Command Macros**

The ICE-88 module provides In-Circuit Emulation for the 8088 microprocessor. It includes three circuit boards which reside in Intellec® Microcomputer Development Systems. A cable and buffer box connect the Intellec system to the user system by replacing the user's 8088. Powerful Intellec debug functions are thus extended into the user system. Using the ICE-88 module, the designer can execute prototype software in continuous or single-step mode and can substitute blocks of Intellec system memory for user equivalents. Breakpoints allow the user to stop emulation on user-specified conditions, and the trace capability gives a detailed history of the program execution prior to the break. All user access to the prototype system software may be done symbolically by referring to the source program variables and labels.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, int$_e$l, INTEL, INTELLEC, MCS, ^{i}m, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

INTEGRATED HARDWARE/SOFTWARE DEVELOPMENT

The ICE-88 emulator allows hardware and software development to proceed interactively. This is more effective than the traditional method of independent hardware and software development followed by system integration. With the ICE-88 module, prototype hardware can be added to the system as it is designed. Software and hardware testing occurs while the product is being developed.

Conceptually, the ICE-88 emulator assists three stages of development:

1. It can be operated without being connected to the user's system, so ICE-88 debugging capabilities can be used to facilitate program development before any of the user's hardware is available.
2. Integration of software and hardware can begin when any functional element of the user system hardware is connected to the 8088 socket. Through ICE-88 mapping capabilities, Intellec memory, ICE memory, or diskette memory can be substituted for missing prototype memory. Time-critical program modules are debugged before hardware implementation by using the 2K-bytes of high-speed ICE-resident memory. As each section of the user's hardware is completed, it is added to the prototype. Thus each section of the hardware and software is "system" tested as it becomes available.
3. When the user's prototype is complete, it is tested with the final version of the user system software. The ICE-88 module is then used for real time emulation of the 8088 to debug the system as a completed unit.

Thus the ICE-88 module provides the user with the ability to debug a prototype or production system at any stage in its development without introducing extraneous hardware or software test tools.

SYMBOLIC DEBUGGING

Symbols and PL/M statement numbers may be substituted for numeric values in any of the ICE-88 commands. This allows the user to make symbolic references to I/O ports, memory addresses, and data in the user program. Thus the user need not remember the addresses of variables or program subroutines.

Symbols can be used to reference variables, procedures, program labels, and source statements. A variable can be displayed or changed by referring to it by name rather than by its absolute location in memory. Using symbols for statement labels, program labels, and procedure names simplifies both tracing and breakpoint setting. Disassembly of a section of code from either trace or program memory into its assembly mnemonics is readily accomplished.

Furthermore, each symbol may have associated with it one of the data types BYTE, WORD, INTEGER, SINTEGER (for short, 8-bit integer) or POINTER. Thus the user need not remember the type of a source program variable when examining or modifying it. For example, the command "!VAR" displays the value in memory of variable VAR in a format appropriate to its type, while the command "!VAR = !VAR + 1" increments the value of the variable.

The user symbol table generated along with the object file during a PL/M-86 compilation or an ASM-86 assembly is loaded into memory along with the user program which is to be emulated. The user may add to this symbol table any additional symbolic values for memory addresses, constants, or variables that are found useful during system debugging.

The ICE-88 module provides access through symbolic definition to all of the 8088 registers and flags. The READY, NMI, $\overline{\text{TEST}}$, HOLD, RESET, INTR, and MN/$\overline{\text{MX}}$ pins of the 8088 can also be read. Symbolic references to key ICE-88 emulation information are also provided.

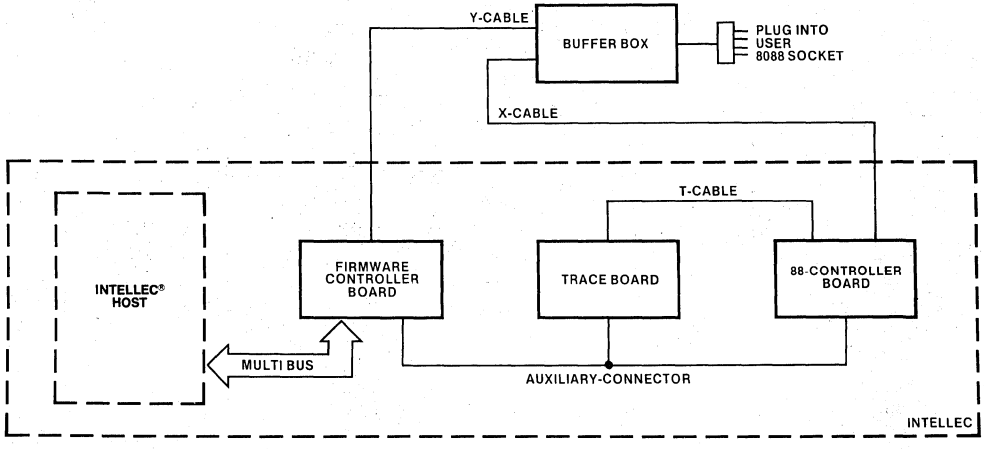
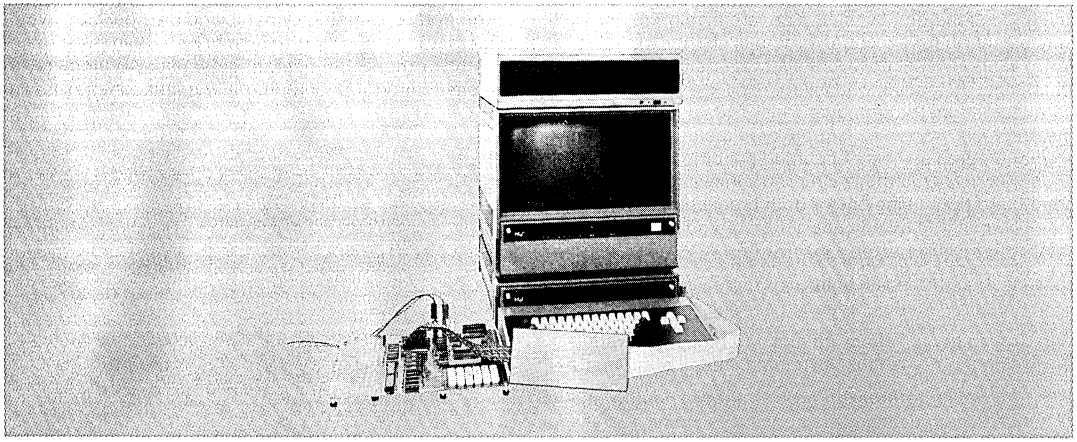


Figure 1. ICE-88™ Emulator Block Diagram

AFN-0145A



A typical ICE-88 development configuration. It is based on a Model 235 Development System, which also includes a Double Density Diskette Operating System and a Model 201 Expansion Chassis (which holds the ICE-88 emulator). The ICE-88 module is shown connected to a user prototype system.

MACROS AND COMPOUND COMMANDS

The ICE-88 module provides a programmable diagnostic facility which allows the user to tailor its operation using macro commands and compound commands.

A macro is a set of ICE-88 commands which is given a single name. Thus, a sequence of commands which is executed frequently may be invoked simply by typing in a single command. The user first defines the macro by entering the entire sequence of commands which he wants to execute. He then names the macro and stores it for future use. He executes the macro by typing its name and passing up to ten parameters to the commands in the macro. Macros may be saved on a disk file for use in subsequent debugging sessions.

Compound commands provide conditional execution of commands (IF), and execution of commands until a condition is met or until they have been executed a specified number of times (COUNT, REPEAT).

Compound commands and macros may be nested any number of times.

MEMORY MAPPING

Memory for the user system can be resident in the user system or "borrowed" from the Intellec System through ICE-88's mapping capability.

The ICE-88 emulator allows the memory which is addressed by the 8088 to be mapped in 1K-byte blocks to:

1. Physical memory in the user's system,
2. Either of two 1K-byte blocks of ICE-88 high-speed memory,
3. Intellec memory,
4. A random-access diskette file.

The user can also designate a block of memory as non-existent. The ICE-88 module issues an error message when any such "guarded" memory is addressed by the user program.

Command	Description
GO	Initializes emulation and allows the user to specify the starting point and breakpoints. Example: GO FROM .START TILL .DELAY EXECUTED where START and DELAY are statement labels.
STEP	Allows the user to single-step through the program.

Table 1. Summary of ICE-88 Emulation Commands.

OPERATION MODES

The ICE-88 software is a RAM-based program that provides the user with easy-to-use commands for initiating emulation, defining breakpoints, controlling trace data collection, and displaying and controlling system parameters. ICE-88 commands are configured with a broad range of modifiers which provide the user with maximum flexibility in describing the operation to be performed.

Emulation

Emulation commands to the ICE-88 emulator control the process of setting up, running and halting an emulation of the user's 8088. Breakpoints and tracepoints enable ICE-88 to halt emulation and provide a detailed trace of execution in any part of the user's program. A summary of the emulation commands is shown in Table 1.

Breakpoints — The ICE-88 module has two breakpoint registers that allow the user to halt emulation when a

specified condition is met. The breakpoint registers may be set up for execution or non-execution breaking. An execution breakpoint consists of a single address which causes a break whenever the 8088 executes from its queue an instruction byte which was obtained from the address. A non-execution breakpoint causes an emulation break when a specified condition other than an instruction execution occurs. A non-execution breakpoint condition, using one or both breakpoint registers, may be specified by any one of or a combination of:

1. *A set of address values.* Break on a set of address values has three valuable features:
   a. Break on a single address.
   b. The ability to set any number of breakpoints within a limited range (1024 bytes maximum) of memory.
   c. The ability to break in an unlimited range. Execution is halted on any memory access to an address greater than (or less than) any 20-bit breakpoint address.
2. *A particular status of the 8088 bus* (one or more of: memory or I/O read or write, instruction fetch, halt, or interrupt acknowledge).
3. *A set of data values* (features comparable to break on a set of address values, explained in point one).
4. *A segment register* (break occurs when the register is used in an effective address calculation).

An external breakpoint match output for user access is provided on the buffer box. This allows synchronization of other test equipment when a break occurs.

Tracepoints — The ICE-88 module has two tracepoint registers which establish match conditions to conditionally start and stop trace collection. The trace information is gathered at least twice per bus cycle, first when the address signals are valid and second when the data signals are valid. If the 8088 execution queue is otherwise active, additional frames of trace are collected.

Each trace frame contains the 20 address/data lines and detailed information on the status of the 8088. The trace memory can store 1,023 frames, or an average of about 300 bus cycles, providing ample data for determining how the 8088 was reacting prior to emulation break. The trace memory contains the last 1,023 frames of trace data collected, even if this spans several separate emulations. The user has the option of displaying each frame of the trace data or displaying by instruction in actual ASM-86 Assembler mnemonics. Unless the user chooses to disable trace, the trace information is always available after an emulation.

Interrogation and Utility

Interrogation and utility commands give the user convenient access to detailed information about the user program and the state of the 8088 that is useful in debugging hardware and software. Changes can be made in both memory and the 8088 registers, flags, input pins, and I/O ports. Commands are also provided for various utility operations such as loading and saving program files, defining symbols and macros, displaying trace data, setting up the memory map, and returning control to the ISIS-II operating system. A summary of the basic interrogation and utility commands is shown in Table 2.

Memory/Register Commands
Display or change the contents of: • Memory • 8088 Registers • 8088 Status flags • 8088 Input pins • 8088 I/O ports • ICE-88 Pseudo-Registers (e.g. emulation timer)
Memory Mapping Commands
Display, declare, set, or reset the ICE-88 memory mapping.
Symbol Manipulation Commands
Display any or all symbols, program modules, and program line numbers and their associated values (locations in memory).
Set the domain (choose the particular program module) for the line numbers.
Define new symbols as they are needed in debugging.
Remove any or all symbols, modules, and program statements.
Change the value of any symbol.
TYPE
Assign or change the type of any symbol in the symbol table.
ASM
Disassemble user program memory into ASM-86 Assembler mnemonics.
PRINT
Display the specified portion of the trace memory.
LOAD
Fetch user symbol table and object code from the input file.
SAVE
Send user symbol table and object code to the output file.
LIST
Send a copy of all output (including prompts, input line echos, and error messages) to the chosen output device (e.g. disk, printer) as well as the console.
EVALUATE
Display the value of an expression in binary, octal, decimal, hexadecimal, and ASCII.
SUFFIX/BASE
Establish the default base for numeric values in input text/output display (binary, octal, decimal, or hexadecimal).
CLOCK
Select the internal (ICE-88 provided, for stand-alone mode only) or an external (user-provided) system clock.
RWTIMEOUT
Allows the user to time out READ/WRITE command signals based on the time taken by the 8088 to access Intellec memory or diskette memory.
ENABLE/DISABLE RDY
Enable or disable logical AND of ICE-88 Ready with the user Ready signal for accessing Intellec memory, ICE memory, or diskette memory.

Table 2. Summary of Basic ICE-88 Interrogation and Utility Commands.

DIFFERENCES BETWEEN ICE-88™ EMULATION AND THE 8088 MICROPROCESSOR

The ICE-88 module emulates the actual operation of the 8088 microprocessor with the following exceptions:

- The ICE-88 module will not respond to a user system NMI or RESET signal when it is out of emulation.
- Trap is ignored in single step mode and on the first instruction step of an emulation.
- The MIN/MAX line, which chooses the "minimum" or "maximum" configuration of the 8088, must not change dynamically in the user system.
- In the "minimum" mode, the user HOLD signal must remain active until HLDA is output by the ICE-88 emulator.
- The $\overline{RQ}/\overline{GT}$ lines in the "maximum" configuration are not supported.

The speed of run emulation by the ICE-88 module depends on where the user has mapped his memory. As the user prototype progresses to include memory, emulation becomes real time.

Memory Mapped To	Estimated Speed
User System	100% of real time*, up to 5 MHz clock
ICE	2 wait states per 8088-controlled bus cycle
Intellec	Approximately 0.02% of real time at 5 MHz clock
Diskette	**

* 100% of real time is emulation at the user system clock rate with no wait states.

** The emulation speed from diskette is comparable to Intellec memory, but emulation must wait when a new page is accessed on the diskette.

DC CHARACTERISTICS OF ICE-88™ USER CABLE

1. Output Low Voltages [V_{OL}(Max) = 0.4V]

	I_{OL}(Min)
AD0-AD7, A8-A15, $\overline{SS0}$, A16/S3-A19/S7, $\overline{RD}$, $\overline{LOCK}$, QS0, QS1, $\overline{S0}$, $\overline{S1}$, $\overline{S2}$, $\overline{WR}$, M/$\overline{IO}$, DT/$\overline{R}$, $\overline{DEN}$, ALE, $\overline{INTA}$	8mA (16mA @ 0.5V)
HLDA	5mA
$\overline{MATCH0\ OR\ MATCH1}$ (on buffer box)	16mA

2. Output High Voltages [V_{OH}(Min) = 2.4V]

	I_{OH}(Min)
AD0-AD7, A8-A15 A16/S3-A19/S7, $\overline{SS0}$, $\overline{RD}$, $\overline{LOCK}$, $\overline{QS0}$, QS1, $\overline{S0}$, $\overline{S1}$, $\overline{S2}$, $\overline{WR}$, M/$\overline{IO}$, DT/$\overline{R}$, $\overline{DEN}$, ALE, $\overline{INTA}$	−2.0mA
HLDA	−3.0mA
$\overline{MATCH0\ OR\ MATCH1}$ (on buffer box)	−0.8mA

3. Input Low Voltages [V_{IL}(Max) = 0.8V]

	I_{IL}(Max)
AD0-AD7	−0.2mA
NMI, CLK	−0.4mA
READY	−0.8mA
INTR, HOLD, $\overline{TEST}$, RESET	−1.4mA
MN/$\overline{MX}$ (0.1μf to GND)	−3.3mA

4. Input High Voltages [V_{IH}(Min) = 2.0V]

	I_{IH}(Max)
AD0-AD7	80μA
NMI, CLK	20μA
READY	60μA
INTR, HOLD, $\overline{TEST}$, RESET	−0.4mA
MN/$\overline{MX}$ (0.1μF to GND)	−1.1mA

5. $\overline{RQ}/\overline{GT0}$, $\overline{RQ}/\overline{GT1}$ are pulled up to +5V through a 5.6K ohm resistor. No current is taken from user circuit at V_{CC} pin.

SPECIFICATIONS

Operating Environment

Required Hardware

Intellec microcomputer development system with:

1. Three adjacent slots for the ICE-88 module. (Series II requires Model 201 Expansion Chassis).
2. 64K bytes of Intellec memory. If user prototype program memory is desired, additional memory above the basic 64K is required.

System console
Intellec diskette operating system
ICE-88 module

Required Software

System monitor
ISIS-II, version 3.4 or subsequent
ICE-88 software

Equipment Supplied

Printed circuit boards (3)
Interface cable and emulation buffer module
Operator's manual
ICE-88 software, diskette-based

Emulation Clock

User system clock up to 5 MHz or 2 MHz ICE-88 internal clock in stand-alone mode

Physical Characteristics

Printed Circuit Boards

Width: 12.00 in (30.48 cm)
Height: 6.75 in (17.15 cm)
Depth: 0.50 in (1.27 cm)
Packaged Weight: 9.00 lb (4.10 kg)

Electrical Characteristics

DC Power

$V_{CC} = +5V\ +5\% - 1\%$
$I_{CC} = 15A$ maximum; 11 A typical
$V_{DD} = +12V \pm 5\%$
$I_{DD} = 120mA$ maximum; 80 mA typical
$V_{BB} = -10V \pm 5\%$ or $-12V \pm 5\%$ (optional)
$I_{BB} = 25mA$ maximum; 12 mA typical

Environmental Characteristics

Operating Temperature: 0° to 40°C

Operating Humidity: Up to 95% relative humidity without condensation.

ORDERING INFORMATION

Part Number	Description
MDS-88-ICE*	8088 CPU in-circuit emulator

*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Science Corporation.

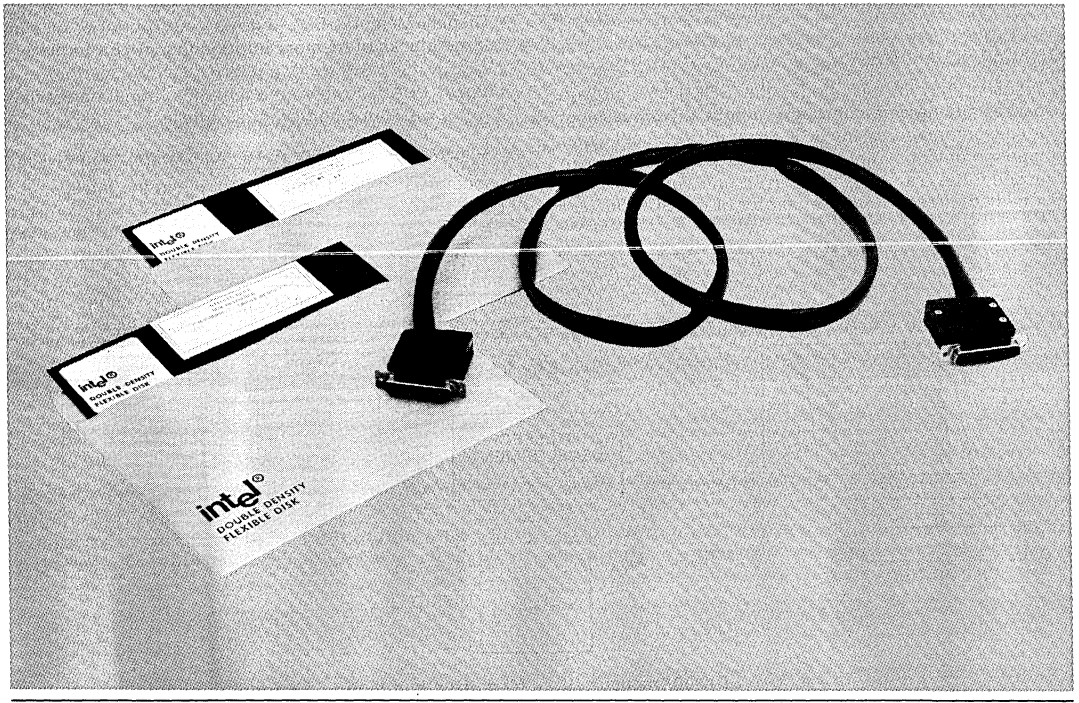
17

Prototype Microcomputer Kits

SDK-C86
MCS-86™ SYSTEM DESIGN KIT
SOFTWARE AND CABLE INTERFACE TO INTELLEC® DEVELOPMENT SYSTEM

- **Provides the Software and Hardware Communications Link Between an Intellec® Development System and the SDK-86**
- **Intellec® System Files can be Accessed and Down-Loaded to the SDK-86 Resident Memory**
- **Data in SDK-86 Memory can be Uploaded and Saved in Intellec® System Files**
- **Enhances and Extends the Power and Usefulness of the SDK-86**
- **Allows the SDK-86 to Become an Execution Vehicle for ISIS-II Developed 8086 Object Code Using the Series II 8086/8088 Software Development Packages**
- **All SDK-86 Serial Port Mode Commands Become Available at Console of the Intellec® System**

The SDK-C86 product provides the software and hardware link for using the SDK-86 monitor in conjunction with an Intellec® Development System while adding features of data transfer between SDK-86 memory and Intellec® System files. The user may enter programs and data into the SDK-86 and then save them on a diskette. Also, programs and data may be created on the Intellec® System using the Series II 8086/8088 Software Development Packages, then loaded into the SDK-86 for testing and checkout. This provides a real time execution environment of the SDK-86 as a peripheral to the Intellec® System.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{i}m, iCS, ICE, UPI, BXP, iSBC, iSBX, iNSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

HARDWARE

There are two serial ports on the Intellec® System back panel, TTY and CRT. Assuming that one of the ports is used for the Intellec® console, the SDK-C86 cable can plug into the unused port. The SDK-86 is jumper selectable to accept either the CRT (RS232) or TTY (20mA current loop) signals.

The edge connector on the SDK-86 has the MULTIBUS™ form factor. No signals are connected to the fingers except the power supply traces. Therefore, the SDK-86 can plug directly into the Intellec® motherboard to obtain power while using the SDK-C86 cable as the communication link.

SOFTWARE

Two programs must be invoked to operate in the SDK-86 slave mode. One program runs on the SDK-86, and another runs in any ISIS-II environment that includes a diskette drive.

The serial I/O monitor is installed on the SDK-86 and operates as though it was talking to a terminal. The software in the Intellec® allows the Intellec®, with a console device, to behave as if it were a terminal to the SDK-86.

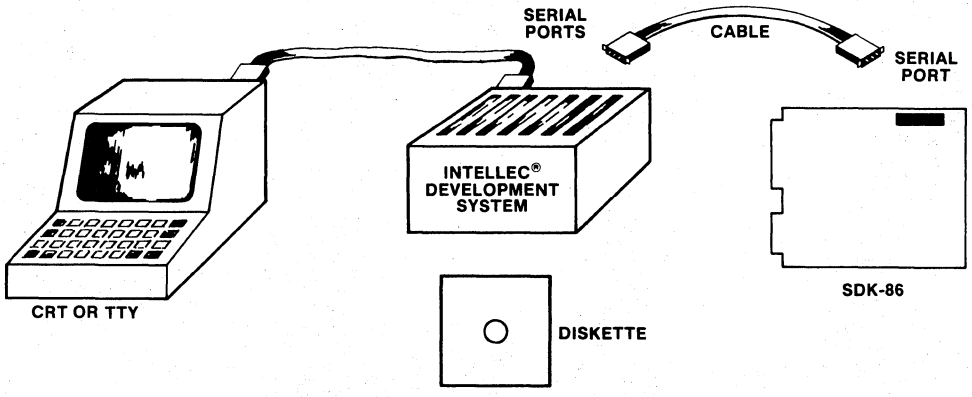
The SDK-C86 software program in the Intellec reads the console input device, then passes the character to the SDK-86 through the serial port. It also receives the characters from the SDK-86 and displays them at the console output device. Besides the basic transfer function, this program also recognizes and performs the Upload and Download functions.

COMMAND MODES

- Transparent: In this mode, the SDK-C86 software passes all characters through without any processing. All the commands of the SDK-86 monitor (except paper tape commands) are available and will function in exactly the same manner as if the terminal were attached directly to the serial port of the SDK-86.
- Upload/Download: In this mode the SDK-C86 software, in the Intellec®, recognizes the mnemonic for Upload or Download from the terminal. It "translates" the Download command to an R (Read hexadecimal tape) command and the Upload command to a W (Write hexadecimal tape). The R and W commands are then passed on to the SDK-86 monitor. Using these paper tape commands allows for a checksummed transfer of data between the Intellec® and the SDK-86 memory.

COMMAND SUMMARY

- Reset — starts the SDK-86 monitor.
- Execute with Breakpoint (G) — Allows you to execute a user program and cause it to halt at a predetermined program step — useful for debugging.
- Single Step (N) — allows you to execute a user program one instruction at a time — useful for debugging.
- Substitute Memory (S, SW) — allows you to examine and modify memory locations in byte or word mode.
- Examine Register (X) — allows you to examine and modify the 8086's register contents.
- Block Move (M) — allows you to relocate program and data portions in memory.
- Input or Output (I, IW, O, OW) — allows direct control of the SDK-86's I/O facilities in byte or word mode.
- Display Memory (D) — allows you to print or display large blocks of memory information in HEX format.
- Load (L) — allows you to load hex format object files into SDK-86 memory from an Intellec.
- Transfer (T) — allows you to save contents of SDK-86 memory in a hex format object file in the Intellec.

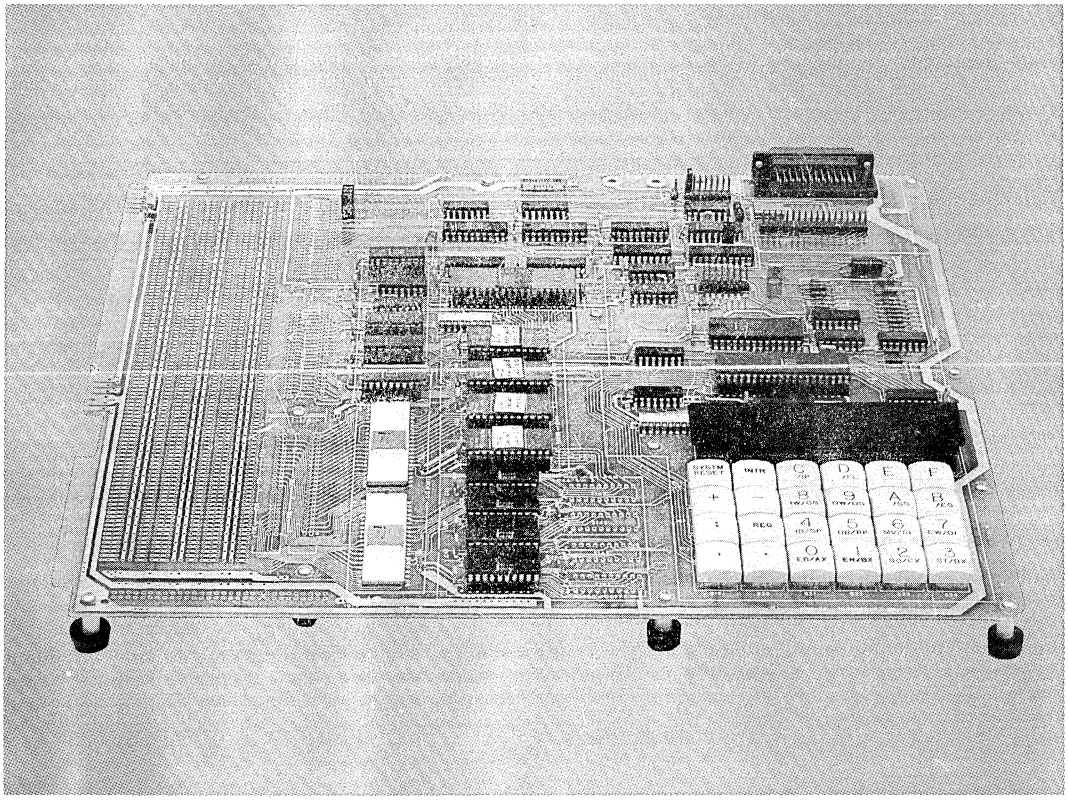


SDK-86/Intellec® Slave Mode Configuration

SDK-86
MCS-86™ SYSTEM DESIGN KIT

- **Complete Single Board Microcomputer System Including CPU, Memory, and I/O**
- **Easy to Assemble Kit Form**
- **High Performance 8086 16-Bit CPU**
- **Interfaces Directly with TTY or CRT**
- **Interactive LED Display and Keyboard**
- **Wire Wrap Area for Custom Interfaces**
- **Extensive System Monitor Software in ROM**
- **Comprehensive Design Library Included**

The SDK-86 MCS-86 System Design Kit is a complete single board 8086 microcomputer system in kit form. It contains all necessary components to complete construction of the kit, including LED display, keyboard, resistors, caps, crystal, and miscellaneous hardware. Included are preprogrammed ROMs containing a system monitor for general software utilities and system diagnostics. The complete kit includes an 8-digit LED display and a mnemonic 24-key keyboard for direct insertion, examination, and execution of a user's program. In addition, it can be directly interfaced with a teletype terminal, CRT terminal, or the serial port of an Intellec system. The SDK-86 is a high performance prototype system with designed-in flexibility for simple interface to the user's application.



FUNCTIONAL DESCRIPTION

The SDK-86 is a complete MCS-86 microcomputer system on a single board, in kit form. It contains all necessary components to build a useful, functional system. Such items as resistors, caps, and sockets are included. Assembly time varies from 4 to 10 hours, depending on the skill of the user. The SDK-86 functional block diagram is shown in Figure 1.

8086 Processor

The SDK-86 is designed around Intel's 8086 microprocessor. The Intel 8086 is a new generation, high performance microprocessor implemented in N-channel, depletion load, silicon gate technology (HMOS), and packaged in a 40-pin CerDIP package. The processor features attributes of both 8-bit and 16-bit microprocessors in that it addresses memory as a sequence of 8-bit bytes, but has a 16-bit wide physical path to memory for high performance. Additional features of the 8086 include the following:

- Direct addressing capability to one megabyte of memory
- Assembly language compatibility with 8080/8085
- 14 word x 16-bit register set with symmetrical operations
- 24 operand addressing modes
- Bit, byte, word, and block operations
- 8 and 16-byte signed and unsigned arithmetic in binary or decimal mode, including multiply and divide
- 4 or 5 or 8 MHz clock rate

A block diagram of the 8086 microprocessor is shown in Figure 2.

System Monitor

A compact but powerful system monitor is supplied with the SDK-86 to provide general software utilities and system diagnostics. It comes in preprogrammed read only memories (ROMs).

Communications Interface

The SDK-86 communicates with the outside world through either the on-board light emitting diode (LED) display/keyboard combination or the user's TTY or CRT terminal (jumper selectable), or by means of a special mode in which an Intellec development system transports finished programs to and from the SDK-86. Memory may be easily expanded by simply soldering in additional devices in locations provided for this purpose. A large area of the board (22 square inches) is laid out as general purpose wire-wrap for the user's custom interfaces.

Assembly

Only a few simple tools are required for assembly: soldering iron, cutters, screwdriver, etc. The SDK-86 assembly manual contains step-by-step instructions for easy assembly with a minimum of mistakes. Once construction is complete, the user connects his kit to a power supply and the SDK-86 is ready to go. The monitor starts immediately upon power-on or reset.

Commands — Keyboard mode commands, serial port commands, and Intellec slave mode commands are summarized in Table 1, Table 2, and Table 3, respectively. The SDK-86 keyboard is shown in Figure 3.

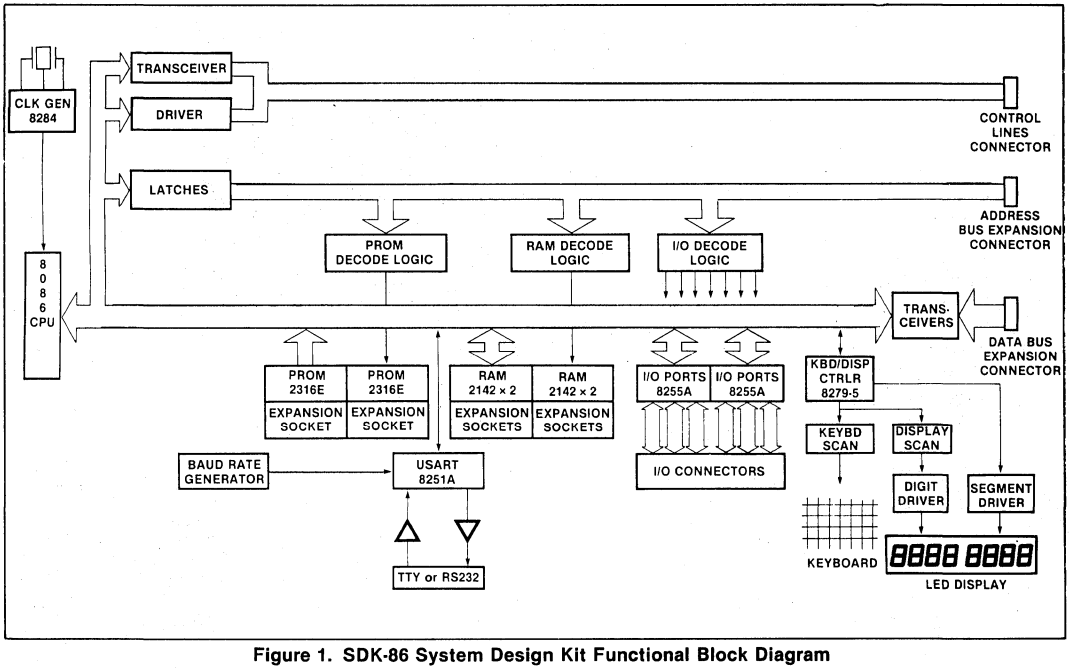


Figure 1. SDK-86 System Design Kit Functional Block Diagram

Documentation

In addition to detailed information on using the monitors, the SDK-86 user's manual provides circuit diagrams, a monitor listing, and a description of how the system works. The complete design library for the SDK-86 is shown in Figure 4 and listed in the specifications section under Reference Manuals.

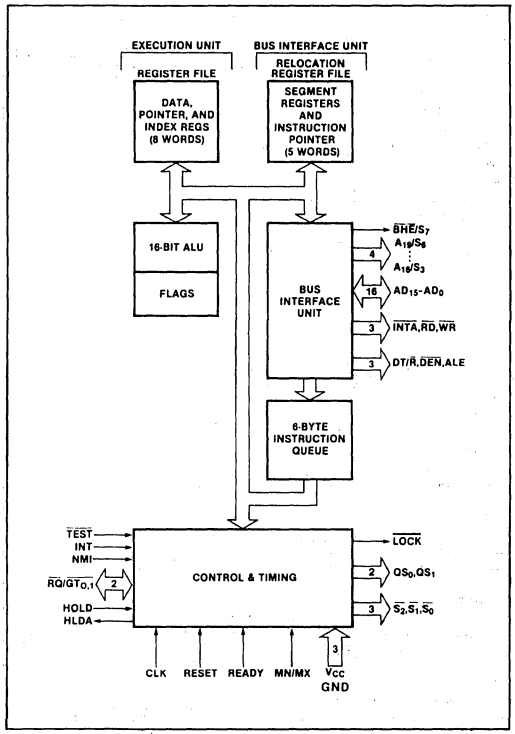


Figure 2. 8086 Microprocessor Block Diagram

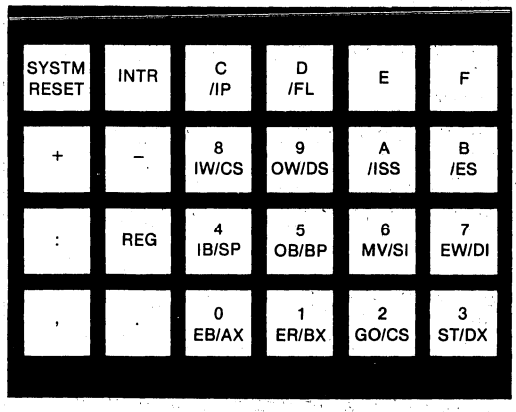


Figure 3. SDK-86 Keyboard



Figure 4. SDK-86 Design Library

Table 1. Keyboard Mode Commands

Command	Operation
Reset	Starts monitor.
Go	Allows user to execute user program, and causes it to halt at predetermined program stop. Useful for debugging.
Single step	Allows user to execute user program one instruction at a time. Useful for debugging.
Substitute memory	Allows user to examine and modify memory locations in byte or word mode.
Examine register	Allows user to examine and modify 8086 register contents.
Block move	Allows user to relocate program and data portions in memory.
Input or output	Allows direct control of SDK-86 I/O facilities in byte or mode.

Table 2. Serial Mode Commands

Command	Operation
Dump memory	Allows user to print or display large blocks of memory information in hex format than amount visible on terminal's CRT display.
Start/continue display	Allows user to display blocks of memory information larger than amount visible on terminal's CRT display.
Punch/read paper tape	Allows user to transmit finished programs into and out of SDK-86 via TTY paper tape punch.

8086 INSTRUCTION SET

Table 4 contains a summary of processor instructions used for the 8086 microprocessor.

Table 4. 8086 Instruction Set Summary

Mnemonic and Description	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
Data Transfer				
MOV = Move:				
Register/memory to/from register	1 0 0 0 1 0 d w	mod reg r/m		
Immediate to register/memory	1 1 0 0 0 1 1 w	mod 0 0 0 r/m	data	data if w=1
Immediate to register	1 0 1 1 w reg	data	data if w=1	
Memory to accumulator	1 0 1 0 0 0 0 w	addr-low	addr-high	
Accumulator to memory	1 0 1 0 0 0 1 w	addr-low	addr-high	
Register/memory to segment register	1 0 0 0 1 1 1 0	mod 0 reg r/m		
Segment register to register/memory	1 0 0 0 1 1 0 0	mod 0 reg r/m		
PUSH = Push:				
Register/memory	1 1 1 1 1 1 1 1	mod 1 1 0 r/m		
Register	0 1 0 1 0 reg			
Segment register	0 0 0 reg 1 1 0			
POP = Pop:				
Register/memory	1 0 0 0 1 1 1 1	mod 0 0 0 r/m		
Register	0 1 0 1 1 reg			
Segment register	0 0 0 reg 1 1 1			
XCHG = Exchange:				
Register/memory with register	1 0 0 0 0 1 1 w	mod reg r/m		
Register with accumulator	1 0 0 1 0 reg			
IN = Input				
Fixed port	1 1 1 0 0 1 0 w	port		
Variable port	1 1 1 0 1 1 0 w			
OUT = Output				
Fixed port	1 1 1 0 0 1 1 w	port		
Variable port	1 1 1 0 1 1 1 w			
XLAT=Translate byte to AL	1 1 0 1 0 1 1 1			
LEA=Load EA to register	1 0 0 0 1 1 0 1	mod reg r/m		
LDS=Load pointer to DS	1 1 0 0 0 1 0 1	mod reg r/m		
LES=Load pointer to ES	1 1 0 0 0 1 0 0	mod reg r/m		
LAHF=Load AH with flags	1 0 0 1 1 1 1 1			
SAHF=Store AH into flags	1 0 0 1 1 1 1 0			
PUSHF=Push flags	1 0 0 1 1 1 0 0			
POPF=Pop flags	1 0 0 1 1 1 0 1			
Arithmetic				
ADD = Add:				
Reg./memory with register to either	0 0 0 0 0 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 s w	mod 0 0 0 r/m	data	data if s:w=01
Immediate to accumulator	0 0 0 0 0 1 0 w	data	data if w=1	
ADC = Add with carry:				
Reg./memory with register to either	0 0 0 1 0 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 s w	mod 0 1 0 r/m	data	data if s:w=01
Immediate to accumulator	0 0 0 1 0 1 0 w	data	data if w=1	
INC = Increment:				
Register/memory	1 1 1 1 1 1 1 w	mod 0 0 0 r/m		
Register	0 1 0 0 0 reg			
AAA=ASCII adjust for add	0 0 1 1 0 1 1 1			
DAA=Decimal adjust for add	0 0 1 0 0 1 1 1			
SUB = Subtract:				
Reg./memory and register to either	0 0 1 0 1 0 d w	mod reg r/m		
Immediate from register/memory	1 0 0 0 0 0 s w	mod 1 0 1 r/m	data	data if s:w=01
Immediate from accumulator	0 0 1 0 1 1 0 w	data	data if w=1	
SBB = Subtract with borrow				
Reg./memory and register to either	0 0 0 1 1 0 d w	mod reg r/m		
Immediate from register/memory	1 0 0 0 0 0 s w	mod 0 1 1 r/m	data	data if s:w=01
Immediate from accumulator	0 0 0 1 1 1 0 w	data	data if w=1	
DEC = Decrement:				
Register/memory	1 1 1 1 1 1 1 w	mod 0 0 1 r/m		
Register	0 1 0 0 1 reg			
NEG=Change sign	1 1 1 1 0 1 1 w	mod 0 1 1 r/m		

Mnemonic and Description	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
CMP = Compare:				
Register/memory and register	0 0 1 1 1 0 d w	mod reg r/m		
Immediate with register/memory	1 0 0 0 0 0 s w	mod 1 1 1 r/m	data	data if s:w=01
Immediate with accumulator	0 0 1 1 1 1 0 w	data	data if w=1	
AAS=ASCII adjust for subtract	0 0 1 1 1 1 1 1			
DAS=Decimal adjust for subtract	0 0 1 0 1 1 1 1			
MUL=Multiply (unsigned)	1 1 1 1 0 1 1 w	mod 1 0 0 r/m		
IMUL=Integer multiply (signed)	1 1 1 1 0 1 1 w	mod 1 0 1 r/m		
AAM=ASCII adjust for multiply	1 1 0 1 0 1 0 0	0 0 0 0 1 0 1 0		
DIV=Divide (unsigned)	1 1 1 1 0 1 1 w	mod 1 1 0 r/m		
IDIV=Integer divide (signed)	1 1 1 1 0 1 1 w	mod 1 1 1 r/m		
AAD=ASCII adjust for divide	1 1 0 1 0 1 0 1	0 0 0 0 1 0 1 0		
CBW=Convert byte to word	1 0 0 1 1 0 0 0			
CWD=Convert word to double word	1 0 0 1 1 0 0 1			
Logic				
NOT=Invert	1 1 1 1 0 1 1 w	mod 0 1 0 r/m		
SHL/SAL=Shift logical/arithmetic left	1 1 0 1 0 0 v w	mod 1 0 0 r/m		
SHR=Shift logical right	1 1 0 1 0 0 v w	mod 1 0 1 r/m		
SAR=Shift arithmetic right	1 1 0 1 0 0 v w	mod 1 1 1 r/m		
ROL=Rotate left	1 1 0 1 0 0 v w	mod 0 0 0 r/m		
ROR=Rotate right	1 1 0 1 0 0 v w	mod 0 0 1 r/m		
RCL=Rotate through carry flag left	1 1 0 1 0 0 v w	mod 0 1 0 r/m		
RCR=Rotate through carry right	1 1 0 1 0 0 v w	mod 0 1 1 r/m		
AND = And:				
Reg./memory and register to either	0 0 1 0 0 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 0 w	mod 1 0 0 r/m	data	data if w=1
Immediate to accumulator	0 0 1 0 0 1 0 w	data	data if w=1	
TEST = And function to flags, no result:				
Register/memory and register	1 0 0 0 0 1 0 w	mod reg r/m		
Immediate data and register/memory	1 1 1 1 0 1 1 w	mod 0 0 0 r/m	data	data if w=1
Immediate data and accumulator	1 0 1 0 1 0 0 w	data	data if w=1	
OR = Or:				
Reg./memory and register to either	0 0 0 0 1 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 0 w	mod 0 0 1 r/m	data	data if w=1
Immediate to accumulator	0 0 0 0 1 1 0 w	data	data if w=1	
XOR = Exclusive or:				
Reg./memory and register to either	0 0 1 1 0 0 d w	mod reg r/m		
Immediate to register/memory	1 0 0 0 0 0 0 w	mod 1 1 0 r/m	data	data if w=1
Immediate to accumulator	0 0 1 1 0 1 0 w	data	data if w=1	
String Manipulation				
REP=Repeat	1 1 1 1 0 0 1 z			
MOVS = Move byte/word	1 0 1 0 0 1 0 w			
CMPS = Compare byte/word	1 0 1 0 0 1 1 w			
SCAS = Scan byte/word	1 0 1 0 1 1 1 w			
LODS = Load byte/wd to AL/AX	1 0 1 0 1 1 0 w			
STOS = Stor byte/wd frm AL/A	1 0 1 0 1 0 1 w			
Control Transfer				
CALL = Call:				
Direct within segment	1 1 1 0 1 0 0 0	disp-low	disp-high	
Indirect within segment	1 1 1 1 1 1 1 1	mod 0 1 0 r/m		
Direct intersegment	1 0 0 1 1 0 1 0	offset-low	offset-high	
		seg-low	seg-high	
Indirect intersegment	1 1 1 1 1 1 1 1	mod 0 1 1 r/m		

continued

Table 4. 8086 Instruction Set Summary (Continued)

Mnemonic and Description	76543210	76543210	76543210
JMP = Unconditional Jump:			
Direct within segment	11101001	disp-low	disp-high
Direct within segment-short	11101011	disp	
Indirect within segment	11111111	mod 100 r/m	
Direct intersegment	11101010	offset-low	offset-high
		seg-low	seg-high
Indirect intersegment	11111111	mod 101 r/m	
RET = Return from CALL:			
Within segment	11000011		
Within seg adding immed to SP	11000010	data-low	data-high
Intersegment	11001011		
Intersegment, adding immediate to SP	11001010	data-low	data-high
JE/JZ=Jump on equal/zero	0111 0100	disp	
JL/JNGE=Jump on less/not greater or equal	01111100	disp	
JLE/JNG=Jump on less or equal/not greater	01111110	disp	
JB/JNAE=Jump on below/not above or equal	0111 0010	disp	
JBE/JNA=Jump on below or equal/ not above	01110110	disp	
JP/JPE=Jump on parity/parity even	01111010	disp	
JO=Jump on overflow	01110000	disp	
JS=Jump on sign	01111000	disp	
JNE/JNZ=Jump on not equal/not zero	01110101	disp	
JNL/JGE=Jump on not less/greater or equal	01111101	disp	
JNLE/JG=Jump on not less or equal/ greater	01111111	disp	
JNB/JAE=Jump on not below/above or equal	0111 0011	disp	
JNBE/JA=Jump on not below or equal/above	01110111	disp	
JNP/JPO=Jump on not par/par odd	01111011	disp	
JNO=Jump on not overflow	01110001	disp	

Mnemonic and Description	76543210	76543210
JNS Jump on not sign	01111001	disp
LOOP Loop CX times	11100010	disp
LOOPZ/LOOPE=Loop while zero/equal	11100001	disp
LOOPNZ/LOOPNE=Loop while not zero/equal	11100000	disp
JCXZ=Jump on CX zero	11100011	disp
INT Interrupt		
Type specified	11001101	type
Type 3	11001100	
INTO=Interrupt on overflow	11001110	
IRET Interrupt return	11001111	
Processor Control		
CLC Clear carry	11111000	
CMC Complement carry	11110101	
STC Set carry	11111001	
CLD Clear direction	11111100	
STD=Set direction	11111101	
CLI Clear interrupt	11111010	
STI=Set interrupt	11111011	
HLT Halt	11110100	
WAIT Wait	10011011	
ESC Escape (to external device)	11011xxx	mod x x x r/m
LOCK Bus lock prefix	11110000	

Notes

AL = 8-bit accumulator
AX = 16-bit accumulator
CX = Count register
DS = Data segment
ES = Extra segment
Above/below refers to unsigned value.
Greater = more positive;
Less = less positive (more negative) signed values
if d = 1 then "to" reg; if d = 0 then "from" reg
if w = 1 then word instruction; if w = 0 then byte instruction

if mod = 11 then r/m is treated as a REG field
if mod = 00 then DISP = 0*, disp-low and disp-high are absent
if mod = 01 then DISP = disp-low sign-extended to 16-bits, disp-high is absent
if mod = 10 then DISP = disp-high: disp-low

if r/m = 000 then EA = (BX) + (SI) + DISP
if r/m = 001 then EA = (BX) + (DI) + DISP
if r/m = 010 then EA = (BP) + (SI) + DISP
if r/m = 011 then EA = (BP) + (DI) + DISP
if r/m = 100 then EA = (SI) + DISP
if r/m = 101 then EA = (DI) + DISP
if r/m = 110 then EA = (BP) + DISP*
if r/m = 111 then EA = (BX) + DISP
DISP follows 2nd byte of instruction (before data if required)

*except if mod = 00 and r/m = 110 then EA = disp-high: disp-low.

if s:w = 01 then 16 bits of immediate data form the operand.
if s:w = 11 then an immediate data byte is sign extended to form the 16-bit operand.
if v = 0 then "count" = 1; if v = 0 then "count" in (CL)
x = don't care.
if v = 0 then "count" = 1; if v = 1 then "count" in (CL) register.
z is used for string primitives for comparison with ZF FLAG.

SEGMENT OVERRIDE PREFIX

001 reg 110

REG is assigned according to the following table:

16-Bit (w = 1)	8-Bit (w = 0)	Segment
000 AX	000 AL	00 ES
001 CX	001 CL	01 CS
010 DX	010 DL	10 SS
011 BX	011 BL	11 DS
100 SP	100 AH	
101 BP	101 CH	
110 SI	110 DH	
111 DI	111 BH	

Instructions which reference the flag register file as a 16-bit object use the symbol FLAGS to represent the file:

FLAGS = X:X:X:X:(OF):(DF):(IF):(TF):(SF):(ZF):X:(AF):X:(PF):X:(CF)

Mnemonics (C) Intel, 1978

SPECIFICATIONS

Central Processor

CPU — 8086 (5 MHz clock rate)

Note
May be operated at 2.5 MHz or 5 MHz, jumper selectable, for use with 8086.

Memory

ROM — 8K bytes 2316/2716
RAM — 2K bytes (expandable to 4K bytes) 2142

Addressing

ROM — FE000-FFFFF
RAM — 0-7FF (800-FFF available with additional 2142's)

Note
The wire-wrap area of the SDK-86 PC board may be used for additional custom memory expansion.

Input/Output

Parallel — 48 lines (two 8255A's)
Serial — RS232 or current loop (8251A)
Baud Rate — selectable from 110 to 4800 baud

Interfaces

Bus — All signals TTL compatible

Parallel I/O — All signals TTL compatible

Serial I/O — 20 mA current loop TTY or RS232

Note
The user has access to all bus signals which enable him to design custom system expansions into the kit's wire-wrap area.

Interrupts (256 vectored)

Maskable
Non-maskable
TRAP

DMA

Hold Request — Jumper selectable. TTL compatible input.

Software

System Monitor — Preprogrammed 2716 or 2316 ROMs

Addresses — FE000-FFFFF

Monitor I/O — Keyboard/display or TTY or CRT (serial I/O)

Physical Characteristics

Width — 13.5 in. (34.3 cm)

Height — 12 in. (30.5 cm)

Depth — 1.75 in. (4.45 cm)

Weight — approx. 24 oz. (3.3 kg)

Electrical Characteristics

DC Power Requirement
(Power supply not included in kit)

Voltage	Current
V_{CC} 5V ± 5%	3.5A
V_{TTY} – 12V ± 10%	0.3A
	(V_{TTY} required only if teletype is connected)

Environmental Characteristics

Operating Temperature — 0-50°C

Reference Manuals

9800697A — SDK-86 MCS-86 System Design Kit Assembly Manual

9800722 — MCS-86 User's Manual

9800640A — 8086 Assembly Language Programming Manual

8086 Assembly Language Reference Card

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

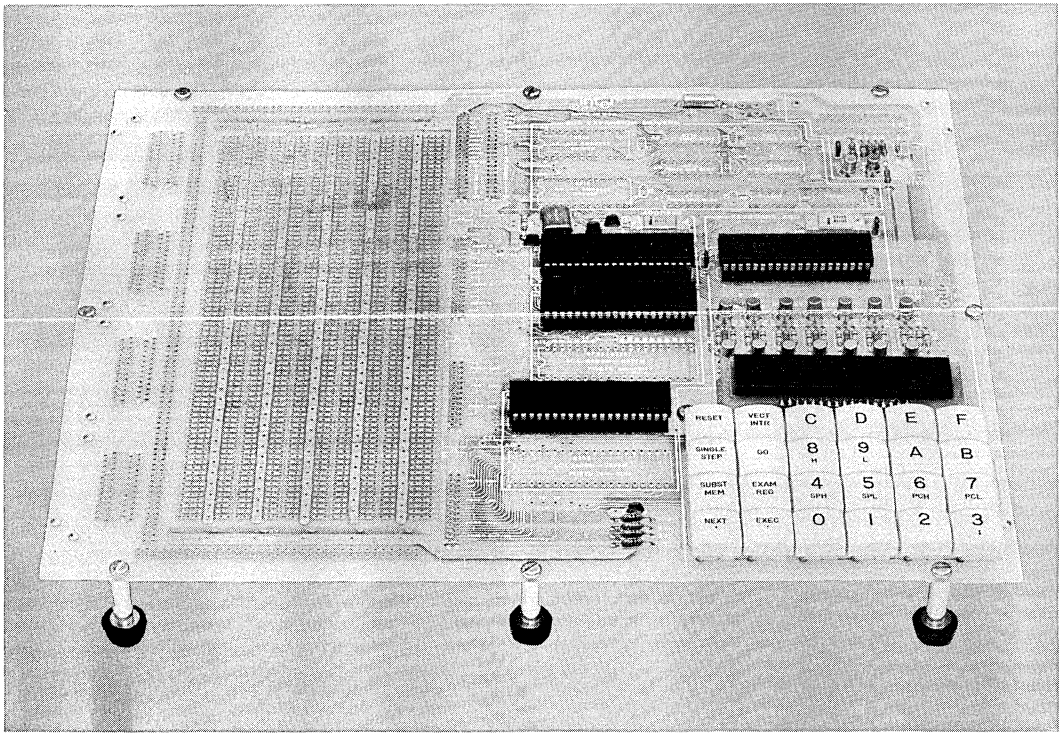
ORDERING INFORMATION

Part Number	Description
SDK-86	MCS-86 system design kit

SDK-85
MCS-85™ SYSTEM DESIGN KIT

- **Complete Single Board Microcomputer System Including CPU, Memory, and I/O**
- **Easy to Assemble, Low Cost, Kit Form**
- **Extensive System Monitor Software in ROM**
- **Interactive LED Display and Keyboard**
- **Large Wire-Wrap Area for Custom Interfaces**
- **Popular 8080A Instruction Set**
- **Interfaces Directly with TTY**
- **High Performance 3 MHz 8085A CPU (1.3 μs Instruction Cycle)**
- **Comprehensive Design Library Included**

The SDK-85 MCS-85 System Design Kit is a complete single board microcomputer system in kit form. It contains all components required to complete construction of the kit, including LED display, keyboard, resistors, caps, crystal, and miscellaneous hardware. Included is a preprogrammed ROM containing a system monitor for general software utilities and system diagnostics. The complete kit includes a 6-digit LED display and a 24-key keyboard for a direct insertion, examination, and execution of a user's program. In addition, it can be directly interfaced with a teletype terminal. The SDK-85 is an inexpensive, high performance prototype system that has designed-in flexibility for simple interface to the user's application.



FUNCTIONAL DESCRIPTION

The SDK-85 is a complete 8085A microcomputer system on a single board, in kit form. It contains all necessary components to build a useful, functional system. Such items as resistors, capacitors, and sockets are included. Assembly time varies from three to five hours, depending on the skill of the user. The SDK-85 functional block diagram is shown in Figure 1.

8085A Processor

The SDK-85 is designed around Intel's 8085A microprocessor. The Intel 8085A is a new generation, complete 8-bit parallel central processing unit (CPU). Its instruction set is 100% software upward compatible with the 8080A microprocessor, and it is designed to improve the present 8080A's performance by higher system speed. Its high level of system integration allows a minimum system of three IC's: 8085A (CPU), 8156 (RAM), and 8355/8755 (ROM/PROM). A block diagram of the 8085A microprocessor is shown in Figure 2.

System Integration — The 8085A incorporates all of the features that the 8224 (clock generator) and 8228 (system controller) provided for the 8080A, thereby offering a high level of system integration.

Addressing — The 8085A uses a multiplexed data bus. The 16-bit address is split between the 8-bit address bus and the 8-bit address/data bus. The on-chip address latches of 8155/8156/8355/8755 memory products allows a direct interface with the 8085A.

System Monitor

A compact but powerful system monitor is supplied with the SDK-85 to provide general software utilities and system diagnostics. It comes in a pre-programmed ROM.

Communications Interface

The SDK-85 communicates with the outside world through either the on-board LED display/keyboard combination, or the user's TTY terminal (jumper selectable

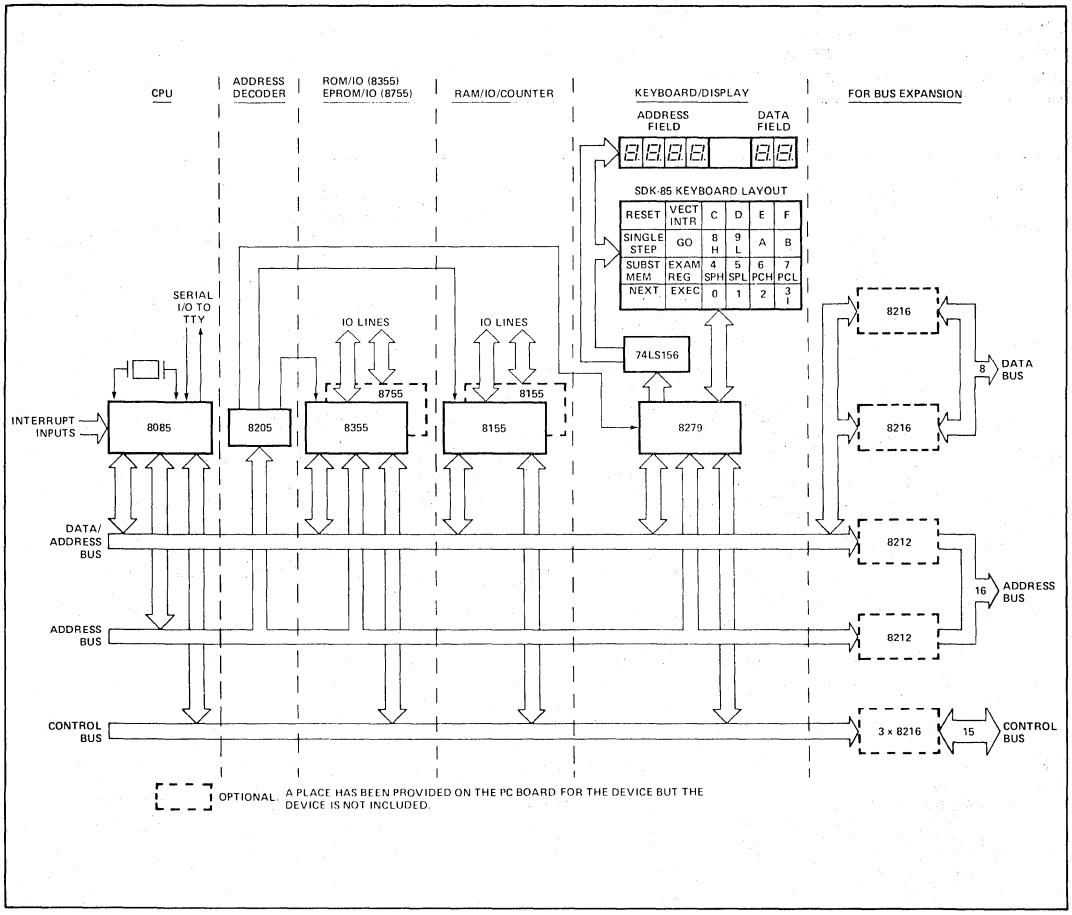
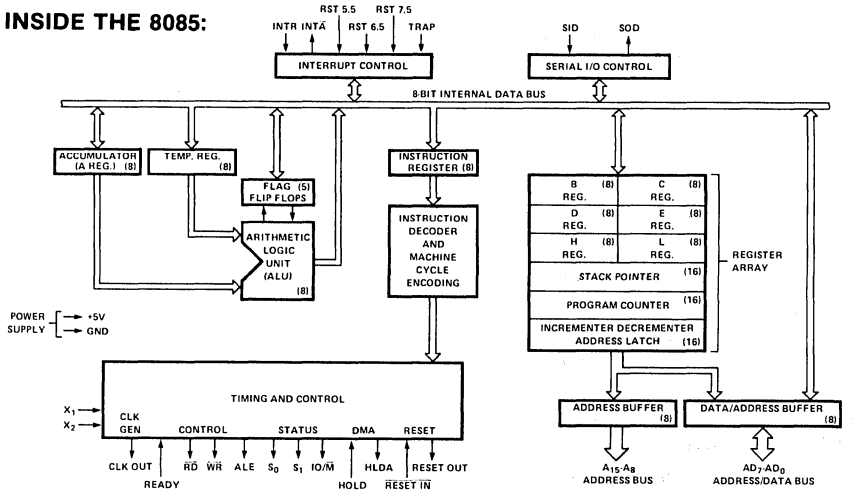


Figure 1. SDK-85 System Design Kit Functional Block Diagram



- SEVEN 8-BIT REGISTERS. SIX OF THEM CAN BE LINKED IN REGISTER PAIRS FOR CERTAIN OPERATIONS.
- 8-BIT ALU.
- 16-BIT STACK POINTER (STACK IS MAINTAINED OFFBOARD IN SYSTEM RAM MEMORY).
- 16-BIT PROGRAM COUNTER.

Figure 2. 8085A Microprocessor Block Diagram

Both memory and I/O can be easily expanded by simply soldering in additional devices in locations provided for this purpose. A large area of the board (45 sq. in.) is laid out as general purpose wire-wrap for the user's custom interfaces.

Assembly

Only a few simple tools are required for assembly; soldering iron, cutters, screwdriver, etc. The SDK-85 user's manual contains step-by-step instructions for easy assembly without mistakes. Once construction is complete, the user connects his kit to a power supply and the SDK-85 is ready to go. The monitor starts immediately upon power-on or reset.

Table 1. Keyboard Monitor Commands

Command	Operation
Reset	Starts monitor.
Go	Allows user to execute user program.
Single step	Allows user to execute user program one instruction at a time—useful for debugging.
Substitute memory	Allows user to examine and modify memory locations.
Examine register	Allows user to examine and modify 8085A's register contents.
Vector interrupt	Serves as user interrupt button.

Commands — Keyboard monitor commands and teletype monitor commands are provided in Table 1 and Table 2 respectively.

Table 2. Teletype Monitor Commands

Command	Operation
Display memory	Displays multiple memory locations.
Substitute memory	Allows user to examine and modify memory locations one at a time.
Insert instructions	Allows user to store multiple bytes in memory.
Move memory	Allows user to move blocks of data in memory.
Examine register	Allows user to examine and modify the 8085A's register contents.
Go	Allows user to execute user programs.

Documentation

In addition to detailed information on using the monitors, the SDK-85 user's manual provides circuit diagrams, a monitor listing, and a description of how the system works. The complete design library for the SDK-85 is shown in Figure 7-11 and listed in the Specifications section under Reference Manuals.

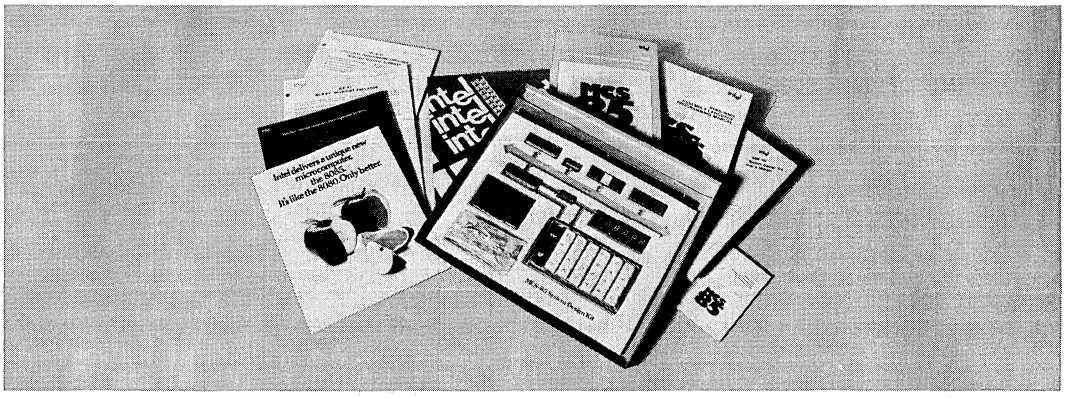


Figure 3. SDK-85 Design Library

8085A INSTRUCTION SET

Table 3 contains a summary of processor instructions used for the 8085A microprocessor.

Table 3. Summary of 8085A Processor Instructions

Mnemonic[1]	Description	Instruction Code[2] D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Clock[3] Cycles
MOVE, LOAD, AND STORE										
MOVr1r2	Move register to register	0	1	D	D	D	S	S	S	4
MOV M,r	Move register to memory	0	1	1	1	0	S	S	S	7
MOV r,M	Move memory to register	0	1	D	D	D	1	1	0	7
MVI r	Move immediate register	0	0	D	D	D	1	1	0	7
MVI M	Move immediate memory	0	0	1	1	0	1	1	0	10
LXI B	Load immediate register Pair B & C	0	0	0	0	0	0	0	1	10
LXI D	Load immediate register Pair D & E	0	0	0	1	0	0	0	1	10
LXI H	Load immediate register Pair H & L	0	0	1	0	0	0	0	1	10
STAX B	Store A indirect	0	0	0	0	0	0	1	0	7
STAX D	Store A indirect	0	0	0	1	0	0	1	0	7
LDAX B	Load A indirect	0	0	0	0	1	0	1	0	7
LDAX D	Load A indirect	0	0	0	1	1	0	1	0	7
STA	Store A direct	0	0	1	1	0	0	1	0	13
LDA	Load A direct	0	0	1	1	1	0	1	0	13
SHLD	Store H & L direct	0	0	1	0	0	0	1	0	16
LHLD	Load H & L direct	0	0	1	0	1	0	1	0	16
XCHG	Exchange D & E, H & L registers	1	1	1	0	1	0	1	1	4
STACK OPS										
PUSH B	Push register pair B & C on stack	1	1	0	0	0	1	0	1	12
PUSH D	Push register pair D & E on stack	1	1	0	1	0	1	0	1	12
PUSH H	Push register pair H & L on stack	1	1	1	0	0	1	0	1	12
PUSH PSW	Push A and flags on stack	1	1	1	1	0	1	0	1	12
POP B	Pop register pair B & C off stack	1	1	0	0	0	0	0	1	10
POP D	Pop register pair D & E off stack	1	1	0	1	0	0	0	1	10
POP H	Pop register pair H & L off stack	1	1	1	0	0	0	0	1	10
POP PSW	Pop A and flags off stack	1	1	1	1	0	0	0	1	10
XTHL	Exchange top of stack, H & L	1	1	1	0	0	0	1	1	16
SPHL	H & L to stack pointer	1	1	1	1	1	0	0	1	6
LXI SP	Load immediate stack pointer	0	0	1	1	0	0	0	1	10
INX SP	Increment stack pointer	0	0	1	1	0	0	1	1	6
DCX SP	Decrement stack pointer	0	0	1	1	1	0	1	1	6
JUMP										
JMP	Jump unconditional	1	1	0	0	0	0	1	1	10
JC	Jump on carry	1	1	0	1	1	0	1	0	7/10
JNC	Jump on no carry	1	1	0	1	0	0	1	0	7/10
JZ	Jump on zero	1	1	0	0	1	0	1	0	7/10
JNZ	Jump on no zero	1	1	0	0	0	0	1	0	7/10
JP	Jump on positive	1	1	1	1	0	0	1	0	7/10
JM	Jump on minus	1	1	1	1	1	0	1	0	7/10
JPE	Jump on parity even	1	1	1	0	1	0	1	0	7/10
JPO	Jump on parity odd	1	1	1	0	0	0	1	0	7/10
PCHL	H & L to program counter	1	1	1	0	1	0	0	1	6
CALL										
CALL	Call unconditional	1	1	0	0	1	1	0	1	18
CC	Call on carry	1	1	0	1	1	1	0	0	9/18
CNC	Call on no carry	1	1	0	1	0	1	0	0	9/18
CZ	Call on zero	1	1	0	0	1	1	0	0	9/18
CNZ	Call on no zero	1	1	0	0	0	1	0	0	9/18
CP	Call on positive	1	1	1	1	0	1	0	0	9/18
CM	Call on minus	1	1	1	1	1	1	0	0	9/18
CPE	Call on parity even	1	1	1	0	1	1	0	0	9/18
CPO	Call on parity odd	1	1	1	0	0	1	0	0	9/18
RETURN										
RET	Return	1	1	0	0	1	0	0	1	10
RC	Return on carry	1	1	0	1	1	0	0	0	6/12
RNC	Return on no carry	1	1	0	1	0	0	0	0	6/12
RZ	Return on zero	1	1	0	0	1	0	0	0	6/12
RNZ	Return on no zero	1	1	0	0	0	0	0	0	6/12
RP	Return on positive	1	1	1	1	0	0	0	0	6/12
RM	Return on minus	1	1	1	1	1	0	0	0	6/12

continued

Table 3. Summary of 8085A Processor Instructions (Continued)

Mnemonic[1]	Description	Instruction Code[2] D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Clock[3] Cycles
RPE	Return on parity even	1	1	1	0	1	0	0	0	6/12
RPO	Return on parity odd	1	1	1	0	0	0	0	0	6/12
RESTART										
RST	Restart	1	1	A	A	A	1	1	1	12
INCREMENT AND DECREMENT										
INR r	Increment register	0	0	D	D	D	1	0	0	4
DCR r	Decrement register	0	0	D	D	D	1	0	1	4
INR M	Increment memory	0	0	1	1	0	1	0	0	10
DCR M	Decrement memory	0	0	1	1	0	1	0	1	10
INX B	Increment B & C registers	0	0	0	0	0	0	1	1	6
INX D	Increment D & E registers	0	0	0	1	0	0	1	1	6
INX H	Increment H & L registers	0	0	1	0	0	0	1	1	6
DCX B	Decrement B & C	0	0	0	0	1	0	1	1	6
DCX D	Decrement D & E	0	0	0	1	1	0	1	1	6
DCX H	Decrement H & L	0	0	1	0	1	0	1	1	6
ADD										
ADD r	Add register to A	1	0	0	0	0	S	S	S	4
ADC r	Add register to A with carry	1	0	0	0	1	S	S	S	4
ADD M	Add memory to A	1	0	0	0	0	1	1	0	7
ADC M	Add memory to A with carry	1	0	0	0	1	1	1	0	7
ADI	Add immediate to A	1	1	0	0	0	1	1	0	7
ACI	Add immediate to A with carry	1	1	0	0	1	1	1	0	7
DAD B	Add B & C to H & L	0	0	0	0	1	0	0	1	10
DAD D	Add D & E to H & L	0	0	0	1	1	0	0	1	10
DAD H	Add H & L to H & L	0	0	1	0	1	0	0	1	10
DAD SP	Add stack pointer to H & L	0	0	1	1	1	0	0	1	10
SUBTRACT										
SUB r	Subtract register from A	1	0	0	1	0	S	S	S	4
SBB r	Subtract register from A with borrow	1	0	0	1	1	S	S	S	4
SUB M	Subtract memory from A	1	0	0	1	0	1	1	0	7
SBB M	Subtract memory from A with borrow	1	0	0	1	1	1	1	0	7
SUI	Subtract immediate from A	1	1	0	1	0	1	1	0	7
SBI	Subtract immediate from A with borrow	1	1	0	1	1	1	1	0	7

Mnemonic[1]	Description	Instruction Code[2] D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Clock[3] Cycles
LOGICAL										
ANA r	And register with A	1	0	1	0	0	S	S	S	4
XRA r	Exclusive Or register with A	1	0	1	0	1	S	S	S	4
ORA r	Or register with A	1	0	1	1	0	S	S	S	4
CMP r	Compare register with A	1	0	1	1	1	S	S	S	4
ANA M	And memory with A	1	0	1	0	0	1	1	0	7
XRA M	Exclusive Or memory with A	1	0	1	0	1	1	1	0	7
ORA M	Or memory with A	1	0	1	1	0	1	1	0	7
CMP M	Compare memory with A	1	0	1	1	1	1	1	0	7
ANI	And immediate with A	1	1	1	0	0	1	1	0	7
XRI	Exclusive Or immediate with A	1	1	1	0	1	1	1	0	7
ORI	Or immediate with A	1	1	1	1	0	1	1	0	7
CPI	Compare immediate with A	1	1	1	1	1	1	1	0	7
ROTATE										
RLC	Rotate A left	0	0	0	0	0	1	1	1	4
RRC	Rotate A right	0	0	0	0	1	1	1	1	4
RAL	Rotate A left through carry	0	0	0	1	0	1	1	1	4
RAR	Rotate A right through carry	0	0	0	1	1	1	1	1	4
SPECIALS										
CMA	Complement A	0	0	1	0	1	1	1	1	4
STC	Set carry	0	0	1	1	0	1	1	1	4
CMC	Complement carry	0	0	1	1	1	1	1	1	4
DAA	Decimal adjust A	0	0	1	0	0	1	1	1	4
INPUT/OUTPUT										
IN	Input	1	1	0	1	1	0	1	1	10
OUT	Output	1	1	0	1	0	0	1	1	10
CONTROL										
EI	Enable interrupts	1	1	1	1	1	0	1	1	4
DI	Disable interrupts	1	1	1	1	0	0	1	1	4
NOP	No-operation	0	0	0	0	0	0	0	0	4
HLT	Halt	0	1	1	1	0	1	1	0	5
NEW 8085 INSTRUCTIONS										
RIM	Read interrupt mask	0	0	1	0	0	0	0	0	4
SIM	Set interrupt mask	0	0	1	1	0	0	0	0	4

Notes

1. All mnemonics copyright © Intel Corporation 1977.
2. DDD or SSS: B = 000, C = 001, D = 010, E = 011, H = 100, L = 101, Memory = 110, A = 111.
3. Two possible cycle times. (6/12) indicates instruction cycles dependent on condition flags.

SPECIFICATIONS

Central Processor

CPU — 8085A

Instruction Cycle — 1.3 μs

Tcy — 330 ns

Memory

ROM — 2K bytes (expandable to 4K bytes) 8355/8755A

RAM — 256 bytes (expandable to 512 bytes) 8155

Addressing

ROM — 0000-07FF (expendable to 0FFF with an additional 8355/8755A)

RAM — 2000-20FF (2800-28FF available with an additional 8155)

Note

The wire-wrap area of the SDK-85 PC board may be used for additional custom memory expansion up to the 64K-byte addressing limit of the 8085A.

Input/Output

Parallel — 38 lines (expandable to 76 lines)

Serial — Through SID/SOD ports of 8085A. Software generated baud rate.

Baud Rate — 110

Interfaces

Bus — All signals TTL compatible
Parallel I/O — All signals TTL compatible
Serial I/O — 20 mA current loop TTY

Note
By populating the buffer area of the board, the user has access to all bus signals that enable him to design custom system expansions into the kit's wire-wrap area.

Interrupts

Three Levels
(RST 7.5) — Keyboard interrupt
(RST 6.5) — TTL input
(INTR) — TTL input

DMA

Hold Request — Jumper selectable. TTL compatible input.

Software

System Monitor — Pre-programmed 8755A or 8355 ROM
Addresses — 0000-07FF
Monitor I/O — Keyboard/display or TTY (serial I/O)

Physical Characteristics

Width — 12.0 in. (30.5 cm)
Height — 10 in. (25.4 cm)
Depth — 0.50 in. (1.27 cm)
Weight — approx. 12 oz

Electrical Characteristics

DC Power Requirement (power supply not included in kit)

Voltage	Current
V_{CC} 5V ± 5%	1.3A
V_{TTY} – 10V ± 10%	0.3A (V_{TTY} required only if teletype is connected)

Environmental Characteristics

Operating Temperature — 0-55°C

Reference Manuals

9800451 — SDK-85 User's Manual (SUPPLIED)

9800366 — MCS-85 User's Manual (SUPPLIED)

9800301 — 8080/8085 Assembly Language Programming Manual (SUPPLIED)

8085/8080 Assembly Language Reference Card (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

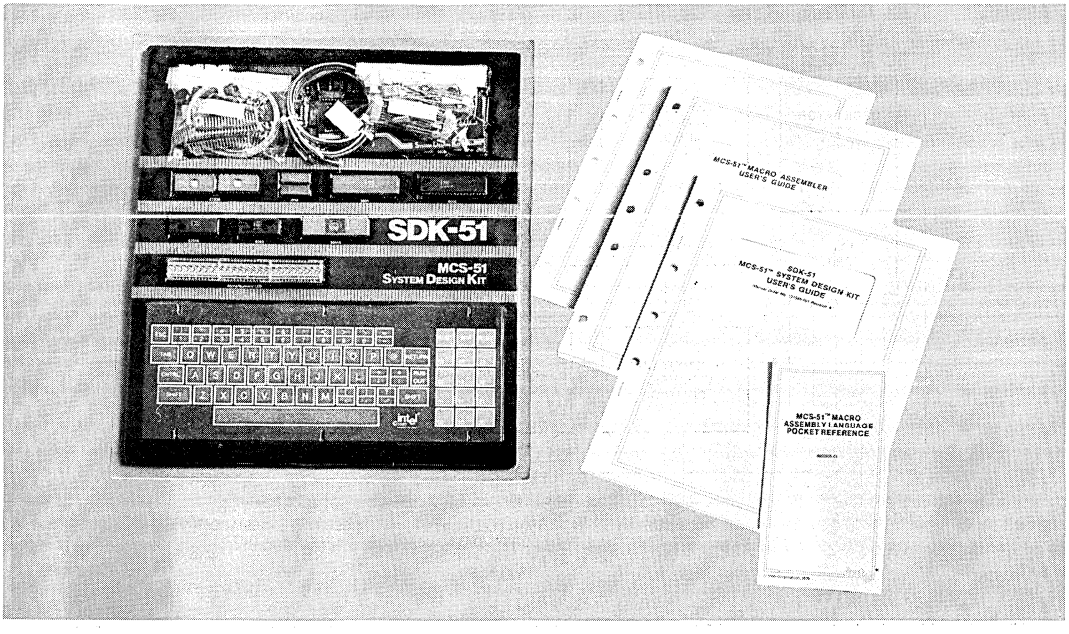
ORDERING INFORMATION

Part Number	Description
SDK-85	MCS-85 system design kit

SDK-51
MCS-51 SYSTEM DESIGN KIT

- **Complete single-board microcomputer kit:**
  - **— Intel 8031 CPU**
  - **— ASCII keyboard and 24-character alpha-numeric display**
  - **— Wire-wrap area for custom circuitry**
  - **—User-configurable RAM**
  - **— Serial and parallel interfaces**
- **Extensive system software in ROM:**
  - **— Single-line assembler and disassembler**
  - **— System debugging commands**
    - **Go**
    - **Step**
    - **Breakpoints**
- **Interface software:**
  - **— Serial port**
  - **— Audio cassette**
  - **— Intellec® system**
- **User's guide, assembly manual, and MCS-51 design manuals**

The SDK-51 MCS-51 System Design Kit contains all of the components required to assemble a complete single-board microcomputer based on Intel's high-performance 8051 single-chip microcomputer. SDK-51 uses the external ROM version of the 8051 (8031). Once you have assembled the kit and supplied +5V power, you can enter programs in MCS-51 assembly language mnemonics, translate them into MCS-51 object code, and run them under control of the system monitor. The kit supports optional memory and interface configurations, including a serial terminal link, audio cassette storage, EPROM program memory, and Intellec® development system upload and download capability.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, Intellec, MCS and ICE, and the combination of MCS or ICE and a numerical suffix. 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 CORPORATION, 1980

FUNCTIONAL DESCRIPTION

The SDK-51 is a kit which includes hardware and software components to assemble a complete MCS-51 family single-board microcomputer. Only common laboratory tools and test equipment are required to assemble the kit. Assembly generally requires 5 to 10 hours, depending on the experience of the user.

The MCS-51 Microcomputer Series

MCS-51 is a series of high-performance single-chip microcomputers for use in sophisticated real-time applications such as instrumentation, industrial control and intelligent computer peripherals. The 8031, 8051, and 8751 microcomputers belong to the 8051 family, which is the first family in the MCS-51 series.

In addition to their advanced features for control applications, MCS-51 family devices have a microprocessor bus and arithmetic capability such as hardware multipy and divide instructions, which make the SDK-51 a versatile stand-alone microcomputer board.

The 8031, 8051, and 8751 CPUs

The 8031, 8051, and 8751 CPUs each combine, on a single chip, a 128 x 8 data RAM; 32 input/output lines; two 16-bit timer/event counters; a five-source, two-level nested interrupt structure; a serial I/O port; and on-chip oscillator and clock circuits. An 8051 block diagram is shown in Figure 1.

The 8031, the SDK-51's CPU, is a CPU without on-chip program memory. The 8031 can address 64K bytes of external program memory in addition to 64K bytes of external data memory. For systems requiring extra capability, each member of the 8051 family can be expanded using standard memories and the byte-oriented MCS-80 and MCS-85 peripherals. The 8051 is an 8031 with the lower 4K bytes of program memory filled with on-chip mask-programmable ROM while the 8751 has 4K bytes of ultraviolet light-erasable, electrically programmable ROM (EPROM).

The 8031 CPU operates at a 12 MHz clock rate, resulting in 4 μs multiply and divide and other instructions of 1 μs and 2 μs.

For additional information on the 8051 family, see the 8051 User's Manual or MCS-51 Macroassembler User's Guide.

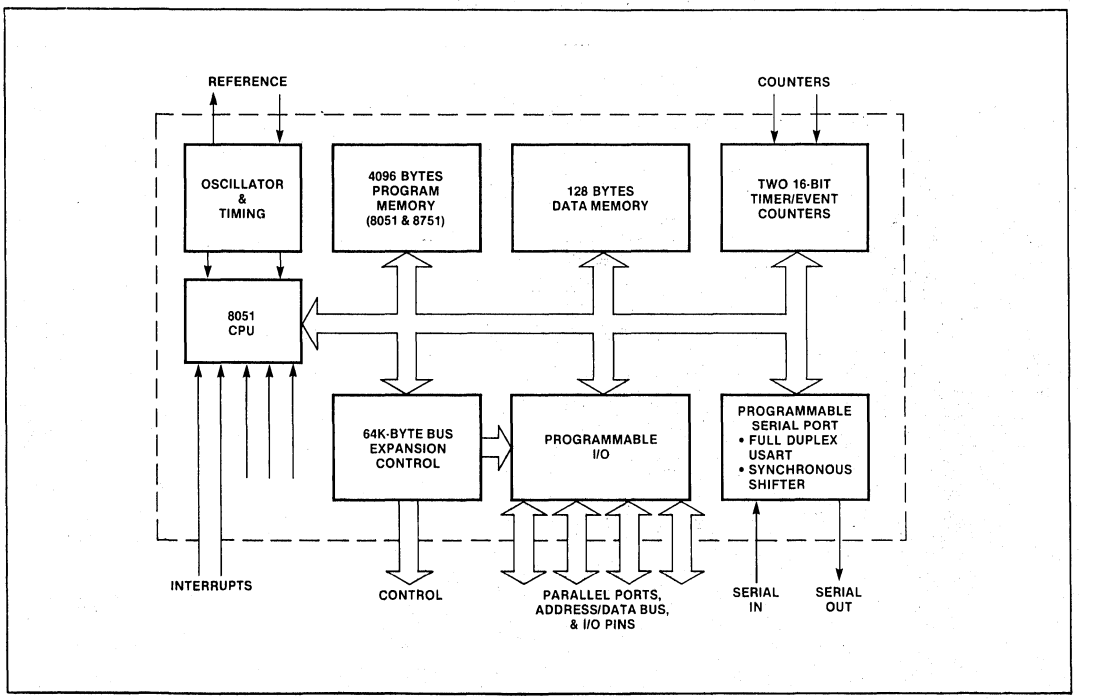


Figure 1. 8051 Block Diagram

System Software

A compact but powerful system monitor is contained in 8K bytes of pre-programmed ROM. The monitor includes system utilities such as command interpretation, user program debugging, and interface controls. Table 1 summarizes the SDK-51 monitor commands.

The ROM devices also include a single-line assembler and disassembler. The assembler lets you enter programs in MCS-51 assembly language mnemonics directly from the ASCII keyboard. The disassembler supports debugging by letting you look at MCS-51 instructions in mnemonic form during system interrogation.

Memory

The two 64K external memory spaces are combined into a single memory space which you can configure between program memory and data memory. The kit includes 1K-byte of static RAM. The board has space and printed circuitry for an additional 15K bytes of RAM and 8K bytes of ROM.

User Interface

The kit includes a typewriter-format, ASCII-subset keyboard and a 24-character, alpha-numeric LED

Table 1. SDK-51 Commands

Command	Operation
Set breakpoint	Define addresses for breaking execution.
Display cause	Ask the system why execution stopped.
Upload, download	Transfer files to and from Intellec® development system.
Save, load	Transfer files to and from optional cassette interface.
Set top of program memory	Define partition between program memory and data memory.
Set baud	Define baud rate value of serial port.
Display memory	Examine and change program memory or data locations.
Assemble	Translate an MCS-51 assembly mnemonic into object code.
Disassemble	Translate program memory into MCS-51 assembly language mnemonics.
Go	Start execution between a selected pair of addresses.
Step	Execute a specified number of instructions.

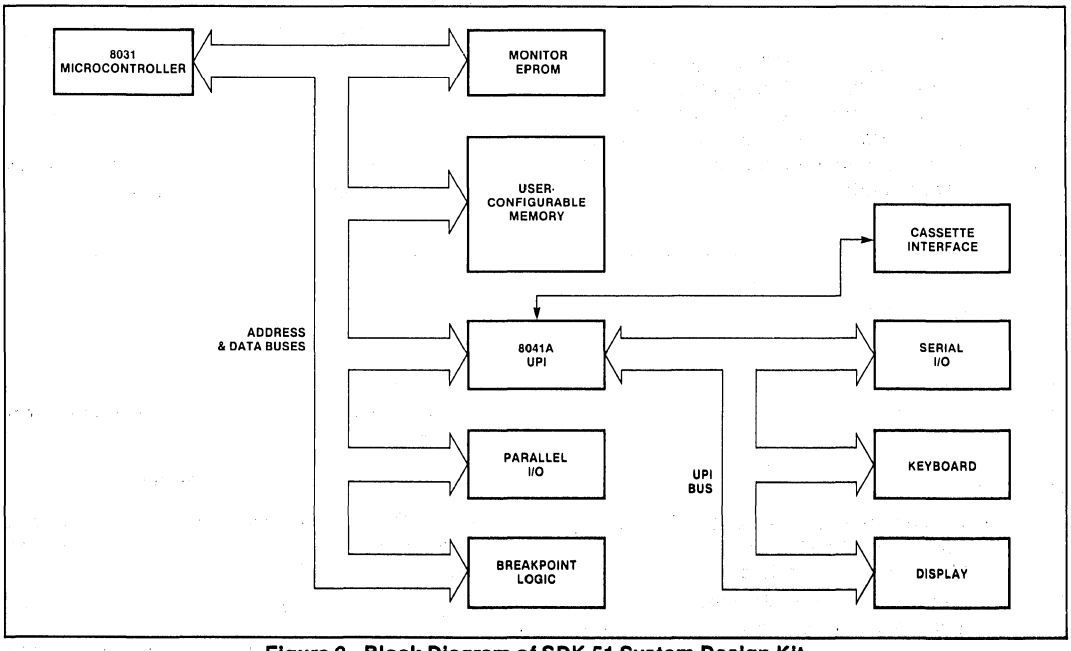


Figure 2. Block Diagram of SDK-51 System Design Kit

display. The standard keyboard and display provide full access to all of the SDK-51's capabilities. All of the SDK-51 interfaces are controlled by a pre-programmed Intel 8041 Universal Peripheral Interface.

A 3 x 4 matrix keyboard can be jumpered to port 1 of the 8031.

Optional Interfaces

TERMINAL

An RS-232-compatible CRT or printing terminal or a current-loop-interface terminal may be used as a listing device by connecting it to the board's serial interface connector and supplying + 12 and – 12 volts to the board.

AUDIO CASSETTE

The kit includes hardware, software, and user's guide instructions to connect and operate an audio cassette tape recorder for low-cost program and data storage.

INTELLEC SYTSTEM

An SDK-51 and an Intellec Model 800 or Series II development system with ISIS-II can upload and download files through the serial interface without adding any software to the Intellec system.

Parallel I/O

The kit includes an Intel 8155 parallel I/O device which expands the 8031 I/O capability by providing 22 dedicated parallel lines. Three 40-pin headers between the 8031 and 8155 devices and the wire-wrap area facilitate interconnections with the user's custom circuitry.

Debugging

Hardware breakpoint logic in the SDK-51 checks the address of a program or external data-memory access against values defined by the user and stops execution when it sees a "break" condition. After a breakpoint, you can examine and modify registers, memory locations, and other points in the system. A step command lets you execute instructions in a single-step mode.

Assembly and Test

The SDK-51 assembly manual describes hardware assembly in a step-by-step process that includes checking each hardware subsystem as it is installed. Building the system requires only a few common tools and standard laboratory instruments.

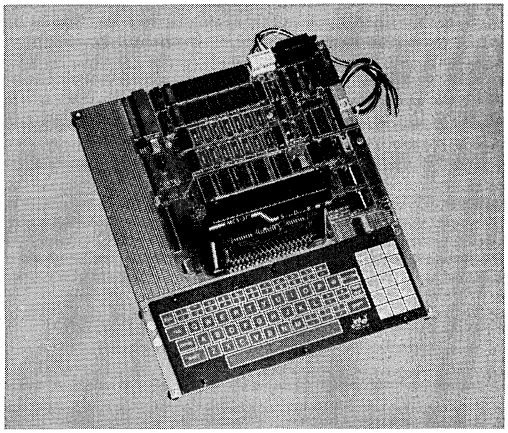


Figure 3. SDK-51 Assembled with Additional RAM and ROM Devices Installed

SPECIFICATIONS

Control Processor

Intel 8031 microcomputer
12 MHz clock rate

Memory

RAM — 1K-byte static, expandable in 1K segments to 16K-byte with 2114 RAM devices; user-configurable as program or data memory.

ROM — Printed circuitry for 8K bytes of program memory in 4K segments using 2732A EPROM devices.

Interfaces

Keyboard — 51-key, ASCII subset, typewriter format, 12-key (3 x 4) matrix

Display — 24-character, alpha-numeric

Serial — RS-232 with user-selectable baud rate. Printed circuitry for 110 baud 20 mA current loop interface. 8031 serial port.

Parallel — 22 lines, TTL compatible

Cassette — Audio cassette tape storage interface

Software

System monitor preprogrammed in on-board ROM

MCS-51 assembler and disassembler preprogrammed in on-board ROM

Interface control software preprogrammed in 8041's on-chip ROM

Assembly and Test Equipment Required

Needle-nose pliers

Small Phillips screwdriver

Small diagonal wire cutters

Soldering pencil, ≤30 watts, 1/16″ diameter tip

Rosin-core, 60-40 solder, 0.05″ diameter

Volt-Ohm-Milliammeter, 1 meg-ohm input impedance

Oscilloscope, 1 volt/division vertical sensitivity, 200 μs/division sweep rate, single trace, internal and external triggering

Physical Characteristics

Length — 13.5 in. (34.29 cm)
Width — 12 in. (30.48 cm)
Height — 4 in. (10.16 cm)
Weight — 3 lb (1.36 kg)

Electrical Characteristics

DC Power Requirement (supplied by user, cable included with kit)

Voltage	Current
+ 5V ±5%	3A
+12V ±5%*	100 mA
−12V ±5%*	100 mA

* ± 12 volts required only for operation with serial interface.

Environmental Characteristics

Operating Temperature — 0 to 40°C

Relative Humidity — 10% to 90%, non-condensing

Reference Manuals

SDK-51 User's Manual

SDK-51 Assembly Manual

SDK-51 Monitor Listing

MCS-51 Macro Assembler User's Guide

MCS-51 Macro Assembly Language Pocket Reference

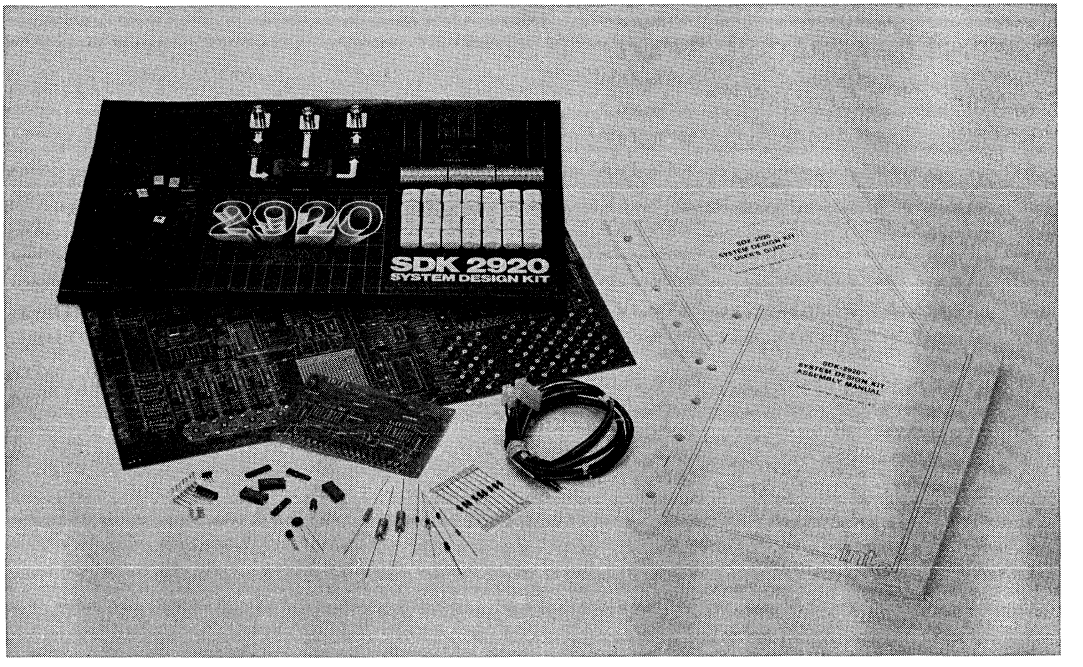
ORDERING INFORMATION

Part Number	Description
MCI-51-SDK	MCS-51 System Design Kit

SDK-2920
2920 SYSTEM DESIGN KIT

- **Complete 2920 program development:**
  - **2920 assembly language keyboard**
  - **Single-line assembler/disassembler**
  - **24-character, alphanumeric display**
  - **2920 memory display and modify**
  - **List program memory to line printer with symbol table**
- **Decimal-to-binary conversion program**
- **File handling capabilities:**
  - **Up/down load of object file to Intellec or audio cassette**
  - **Up load source file with symbol table to Intellec**
  - **2920 EPROM programming**
- **Real-time execution of a programmed 2920**
- **Breadboarding area**

The SDK-2920 contains all of the components required to assemble a complete single board microcomputer system for programming and evaluation of the 2920 Analog Signal Processor. The 8085/8041A microcomputer-based program development section allows you to immediately enter programs in 2920 assembly mnemonics, translate them to 2920 object code, and program the on-board 2920 EPROM. The kit supports basic filing options such as up/down loading to/from an Intellec, audio cassette, and line printer. The SDK-2920 also provides the user with a 2920 run mode section allowing real-time execution of a programmed 2920. This section comes complete with BNC connectors and Intel's 2912 PCM line filters required for one input and one output network. The kit supports optional input and output circuitry on the run mode section.



FUNCTIONAL DESCRIPTION

The SDK-2920 is a kit which includes all the necessary hardware and software components to assemble, using common laboratory tools and test equipment, a complete single board 2920 programming and evaluation aid. Assembly generally requires 4 to 8 hours, depending on the experience of the user.

The 2920 Signal Processor

The Intel® 2920 Signal Processor is a programmable, single chip analog and digital signal processor specifically designed to replace analog subsystems in real-time processing applications. Its instruction set plus the high precision (25 bits) digital arithmetic logic unit provides the capability to implement very complex subsystems. Typical functions performed by the 2920 include: lowpass and bandpass filters with up to 20 complex pole and/or zero pairs; threshold detectors; limiters; rectifiers; up to 25-bit multiplication and division; approximations to nonlinear functions such as square law and logarithm; logical operations; input and output multiplexing of signals; logical outputs for decision type processing; and analog outputs for multifrequency oscillators, waveform generators, etc. In addition, several 2920's may be cascaded for very complex processing applications with no loss in throughput rate.

Tables 1 and 2 show the 2920 instruction set and op codes.

Table 1. Shift Op Codes

Operation	Mnemonic	Op Code				Scale Factor
		3	2	1	0	
Shift Right 13 Bits	R13	1	1	0	0	2^{-13}
Shift Right 12 Bits	R12	1	0	1	1	2^{-12}
•	•	•	•	•	•	•
•	•	•	•	•	•	•
•	•	•	•	•	•	•
Shift Right 1 Bit	R01	0	0	0	0	2^{-1}
No Shift	R00	1	1	1	1	1
Shift Left 1 Bit	L01	1	1	1	0	2
Shift Left 2 Bits	L02	1	1	0	1	4

System Software

A compact but powerful system monitor is contained in 6K bytes of preprogrammed ROM. The monitor includes system utilities such as command interpretation, user program debugging, and interface controls.

The monitor ROM devices also include a single-line assembler and disassembler. The assembler lets you enter programs in 2920 assembly language mnemonics. The disassembler supports debugging by letting you look at or change either hexadecimal values or 2920 instructions during program interrogation.

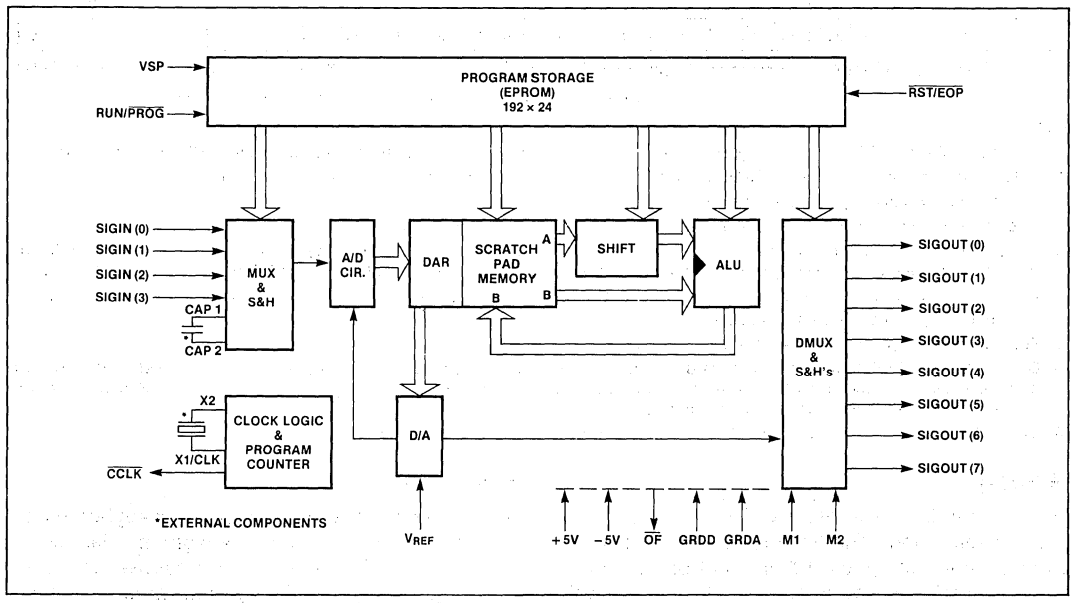
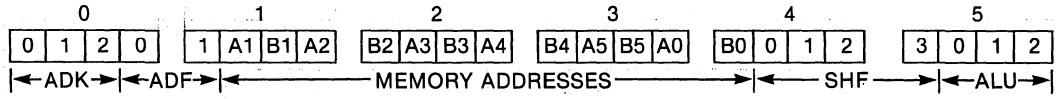


Figure 1. 2920 Function Block Diagram (Run Mode)

Table 2. Instruction Set and Op Codes

Mnemonics	Op Codes[1]			Operations	Notes
Code Condition	**ALU**	**ADF**	**ADK**		
Digital Instructions	**2,1,0**	**1,0**	**2,1,0**		
ADD	110	↑	↑	$(A \times 2^N) + B \rightarrow B$	
SUB	101			$B - (A \times 2^N) \rightarrow B$	
LDA	111			$(A \times 2^N) + 0 \rightarrow B$	
XOR	000			$(A \times 2^N) \oplus B \rightarrow B$	
AND	001			$(A \times 2^N) \cdot B \rightarrow B$	
ABS	011			$\lvert(A \times 2^N)\rvert \rightarrow B$	
ABA[11]	100	[3]	[3]	$\lvert(A \times 2^N)\rvert + B \rightarrow B$	
LIM	010			Sign(A) → ± F.S. → B[4]	
ADD CND( )[2]	110			$(A \times 2^N) + B \rightarrow B$ IFF DAR(K) = 1 $B \rightarrow B$ IFF DAR(K) = 0	
SUB CND( )[2,8]	101			$B - (A \times 2^N) \rightarrow B$ & CY → DAR(K) IFF $CY_P = 1$ $B + (A \times 2^N) \rightarrow B$ & CY → DAR(K) IFF $CY_P = 0$	[5]
LDA CND( )[2]	111			$(A \times 2^N) \rightarrow B$ IFF DAR(K) = 1 $B \rightarrow B$ IFF DAR(K) = 0	
ABA[11] CND( )[9]	100			$(A \times 2^N) + B \rightarrow B$	
XOR CND( )[9]	000			$(A \times 2^N) \oplus B \rightarrow B$	
Analog Instructions					
IN(K)	↑	00	0-3	Signal sample from input channel K	
OUT(K)		10	0-7	D/A to output channel K	
CVTS		00	6	Determine sign bit	
CVT(K)	[7]	01	0-7	Perform A/D on bit K	[6]
EOP		00	5	Program counter to zero	
NOP		00	4	No operation	
CND(K)		11	0-7	Select bit K for conditional instructions	
CNDS	↓	00	7	Select sign bit for conditional instructions	



Note: The input pins for each nibble bit from left to right are D0, D1, D2, D3.

NOTES:

1. Op codes ALU and ADF are in binary notation, ADK is in decimal notation and represents the value "K" when appropriate.
2. CND( ) can be either CND(K) or CNDS testing amplitude bits or the sign bit of the DAR respectively.
3. Determined by analog instructions below.
4. B is set to full scale (F.S.) amplitude with the same sign as the "A" port operand.
5. The previous carry bit (CY_P) is tested to determine the operation. The present carry bit (CY) is loaded into the Kth bit location of the DAR. "Present carry (CY) is generated independent of overflow. It will represent the carry (CY) of a calculated 28-bit result."
6. EOP will also enable overflow correction if it was disabled during a program pass. The EOP must occur in ROM location 188.
7. Determined by digital instructions above.
8. For SUB CNDS Operation $\overline{CY} \rightarrow$ DAR(S).
9. Does not affect DAR. In this case, CND Is used with XOR/ABA to enable/disable the ALU overflow saturation algorithm. Use of either instruction causes the ALU output to roll over rather than go to full scale with sign bit preserved. An EOP instruction will also enable the ALU overflow saturation algorithm.
10. Clarification of CY_{OUT} sense for certain operations. For LDA, XOR, AND, ABS; $CY_{OUT} \rightarrow 0$.

Memory

The kit contains 1.25K bytes of RAM for 2920 program development. The RAM is used as 2920 program memory for up to a 192-instruction 2920 program. The RAM space is also used for a symbol table up to 40 user defined symbols.

User Interface

The kit includes a function and hex keyboard and a formatted 24-character, 18-segment display for easy 2920 code entry. The interactive keyboard and display enables the system monitor to step the user through a command entry sequence with

the friendliness of a menu-driven operating system.

Optional Interfaces

An RS-232 or 20 mA current loop compatible CRT or printer may be used as a listing or file storage device by connecting it to the board's serial interface connector and supplying + 12 and – 12 volts to the board. In addition, the kit provides an audio cassette interface, allowing the use of an audio cassette as a mass storage device.

Debugging

Program development is made easy by use of interactive error messages that will inform the user of illegal entries at the time of program development. Syntax errors are checked for at time of EPROM programming, giving the user the option to continue the programming or not.

The run-mode section allows the user to execute a programmed 2920 in real time, with his own input stimulus and output circuit or instrumentation. The kit is supplied with the 2920-18 and a 5 MHz crystal (800 ns instructions). However, the kit will support the 2920-16 (600 ns instructions) with a 6.67 MHz crystal or clock.

Assembly and Test

The SDK-2920 assembly manual describes assembly in a step-by-step process that includes checking segments of hardware as they are installed. Building the system requires only a few common tools and standard instruments.

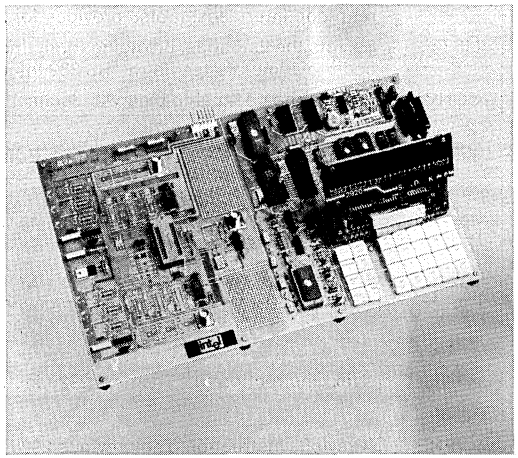


Figure 3. Assembled SDK-2920

CONTROL SECTION

RESET	LIST LOAD
HEX/ASM →	EDIT ←
INSRT NEXT	DEL PREV
SHIFT	CONV CR

2920 DATA KEYS

KP KN	ADD C	ABS D	ABA E	AND F
+/– R	SUB 8	LDA 9	LIM A	XOR B
L .	IN 4	OUT 5	CVTS 6	CVT 7
DAR Y	NOP 0	CNDS 1	CND 2	EOP 3

Figure 2. Keyboard Arrangement

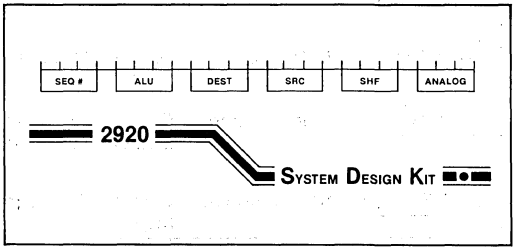


Figure 4. Display Layout

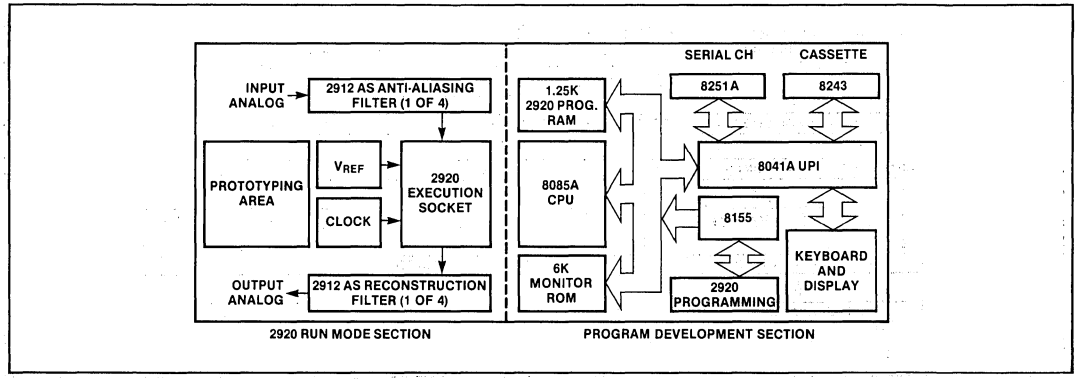


Figure 5. SDK-2920 Functional Block Diagram

Table 3. SDK-2920 Control Commands

RESET	—	Sets the monitor to its initialization program and responds to the selection of one of the four modes. The display will prompt the user with EDIT? LOAD? LIST? CONV?.
SHIFT	—	Selects the upper case characters or functions.
EDIT	—	Selects the edit mode, allowing for 2920 program entry and/or modification. The commands available in the edit mode are shown below in Table 4.
LOAD	—	Selects the load mode, providing for up/down loading to/from the RS-232, cassette, or the 20 mA current loop interfaces. It also provides for 2920 EPROM read, program and verify.
LIST	—	Selects the list mode, providing for listing the 2920 program source code, symbol table, and 2920 hex code to a line printer via the RS-232 interface.
CONV	—	Selects the decimal-to-binary-to-decimal conversion program.

Table 4. Edit Mode Commands

→	Cursor Right	The blinking cursor is moved right one position unless at the end of displayed field.
←	Cursor Left	Blinking cursor is moved left until the sequence number is encountered, then it skips to the left edge of the display.
NEXT	Next Instruction	The next 2920 instruction is displayed unless at end of memory.
PREV	Previous Instruction	The previous 2920 instruction is displayed unless at beginning of memory.
LIST	List Memory	Send disassembled 2920 instructions to serial port.
HEX/ASM	Mode Toggle	Toggle edit mode between symbolic assembly and hexadecimal.
INSRT	Insert Instruction	Expand the program in memory by one location and insert a NOP at current memory display address.
DEL	Delete Instruction	Contract the program in memory by one location and remove the instruction at the current memory display position.

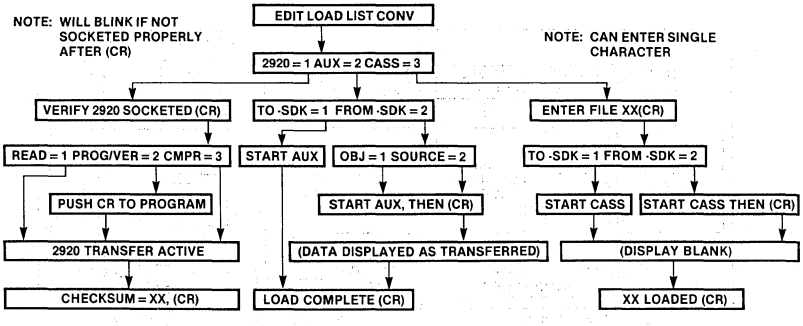


Figure 6. Load Command and Display Tree

SPECIFICATIONS

Control Processor

Intel 8085A microprocessor
6.144 MHz clock rate

Memory

RAM — 1.25K-byte static
ROM — 6K-byte

Interfaces

Keyboard — 28-key with shift, providing 54 functions and characters

Display — 24-character, 18-segment LED

Serial — RS-232 with user-selectable baud rate and 20 mA current loop

Cassette — Hardware and software for audio cassette tape storage interace

Software

System monitor preprogrammed in ROM

2920 assembler and disassembler preprogrammed in ROM

Interface control software preprogrammed in 8041 on-chip ROM

Assembly and Test Equipment Required

- Needle-nose pliers
- Small Phillips screwdriver
- Small flat-blade screwdriver
- Small diagonal wire cutters
- Soldering pencil, <30 watts, 1/16″ diameter tip
- Rosin-core, 60-40 solder, 0.05″ diameter
- Volt-ohm-milliameter, 1 meg ohm input impedance
- Oscilloscope, 1 volt/division vertical sensitivity, 200 μs/division sweep rate, single trace, internal and external triggering

Physical Characteristics

Length — 16 in. (40.64 cm)
Width — 10 in. (25.40 cm)
Height — 4 in. (10.16 cm)
Weight — 3 lb (1.36 kg)

Electrical Characteristics

DC Power Requirements (supplied by user, cables included with the kit)

Program Development Section:

Voltage	Current	Comments
+5V ±5%	1.0A	Required for program development
+12V ±5%	100 mA	Required for 2920 EPROM programming and RS-232 interface
−12V ±5%	100 mA	Required for RS-232 interface

Run Mode Section:

Voltage	Current	Comments
+5V ±5%	300 mA	Required for operation as supplied
	200 mA	Required for each additional 2912/74LS324 pair
−5V ±5%	250 mA	Required for operation as supplied
	200 mA	Required for each additional 2912/74LS324 pair

Environmental Characteristics

Operating Temperature — 0 to 40°C
Relative Humidity — 10% to 90% non-condensing

Reference Manuals

SDK-2920 System Design Kit User's Guide
SDK-2920 System Design Kit Assembly Manual
2920 Analog Signal Processor Design Handbook

ORDERING INFORMATION

Part Number	Description
MCI-20-SDK	2920 System Design Kit

MCS–48™ Development Systems

18

INTELLEC PROMPT 48
MCS-48 MICROCOMPUTER DESIGN AID

Complete low cost design aid and EPROM programmer for revolutionary MCS-48 single component computers

Simplifies microcomputing, allowing user to enter, run, debug, and save machine language programs with calculator-like ease

Utilizes two removable 8-bit MCS-48 CPUs
- **— 8748 CPU with erasable, reprogrammable on-chip program memory**
- **— 8035 CPU with off-chip program memory**

1K-byte erasable, reprogrammable on-chip (8748), expandable program memory, 1K-byte RAM in PROMPT system

64 bytes RAM on-chip, expandable register memory

256 bytes expandable RAM data memory in PROMPT system

27 on-chip TTL compatible expandable I/O lines

On-chip clock, internal timer/event counter, two vectored interrupts, eight level stack control

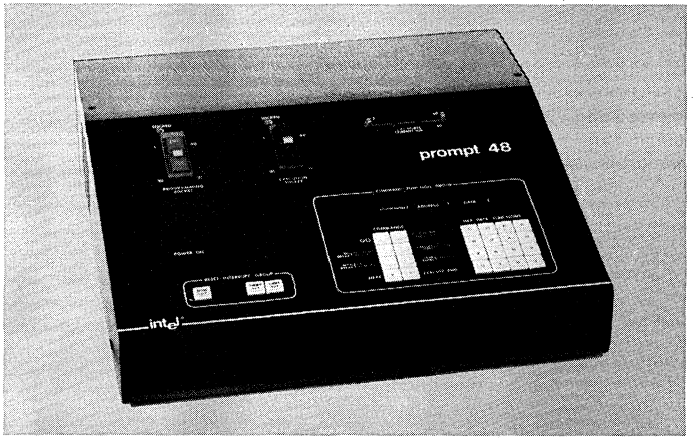
Single +5V DC system power requirement

Integral keyboard and displays (no teletypewriter or CRT terminal required)

Extensive PROMPT 48 monitor, allowing system I/O, bus, and memory expansion

Includes comprehensive design library

The Intellec Prompt 48 MCS-48 Microcomputer Design Aid is a low cost, fully-assembled design aid for the revolutionary 8748 single component microcomputer. PROMPT 48 simplifies the programming of MCS-48 systems — programs may be entered and debugged with calculator-like ease on the large, informative display and keyboard panel. The comprehensive design library with tutorial manual is ideal for newcomers to microcomputing. PROMPT 48's panel connector allows easy access to I/O ports and system bus. Thus users can expand program memory beyond the 1K bytes provided internally.



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 CORPORATION, 1979

AFN-00352A-01

FEATURES

Single Component Computer

The 8748 is the first microcomputer fully integrated on one component. All elements of a computing system are provided, including CPU, RAM, I/O, timer, interrupts, and erasable, reprogrammable nonvolatile program memory.

Programming Socket

PROMPT's programming socket programs this revolutionary "smart PROM"—the 8748—in a highly reliable, convenient manner. A fail-safe interlock ensures the device is properly inserted before applying programming pulses. Each location may be individually programmed, one byte at a time. A read-before-write programming algorithm prevents device damage by inadvertently programming unerased memory.

MCS-48 Processors

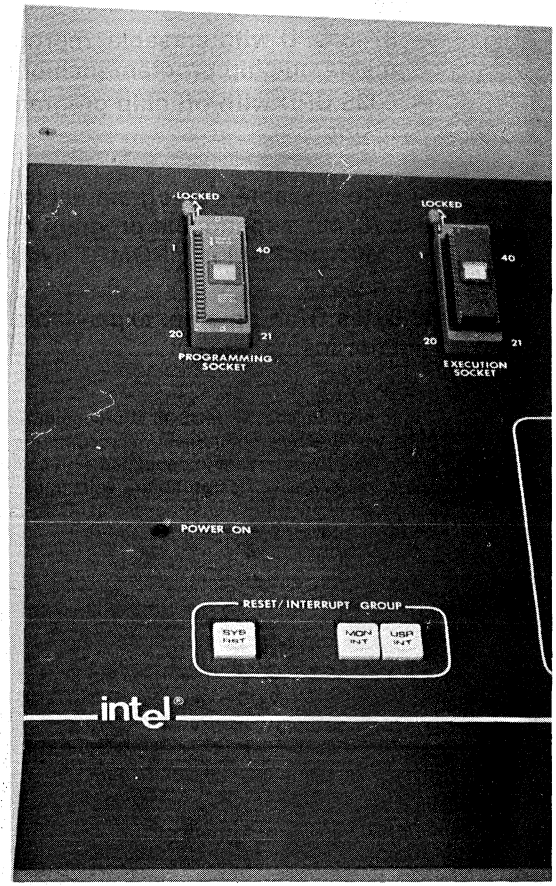
The execution socket accepts either an 8035 or an 8748 MCS-48 processor. Both are supplied with each PROMPT 48, and either can serve as heart of the PROMPT system. There are no processors within the PROMPT 48 mainframe, which instead contains monitor ROM and RAM, user RAM, peripherals, drivers, and sophisticated control circuitry. Once a processor is seated in the execution socket and power is applied, the PROMPT system comes to life. Various access modes may be selected such as program execution from PROMPT system RAM, or from on-chip PROM. Thus programs may first be executed from PROMPT RAM with the 8035 processor. When debugging is complete, the 8035 (execution socket) processor can program the 8748 (programming socket) processor. Finally, a programmed 8748 processor may be exercised by itself from the execution socket. The execution socket processor runs either monitor or user programs.

System Monitor

The system reset command initializes the PROMPT system and enters the monitor. The monitor interrupt command exits a user program gracefully, preserving system status and entering the monitor. The user interrupt command causes an interrupt only if the PROMPT system is running a user program. A comprehensive system monitor resides in four 1K-byte read only memories. It drives the PROMPT keyboard and displays and responds to commands and functions. The top 16 bytes of on-chip program memory must be used by the PROMPT system to switch between monitor and user programs. It requires one level of the MCS-48 eight-level stack.

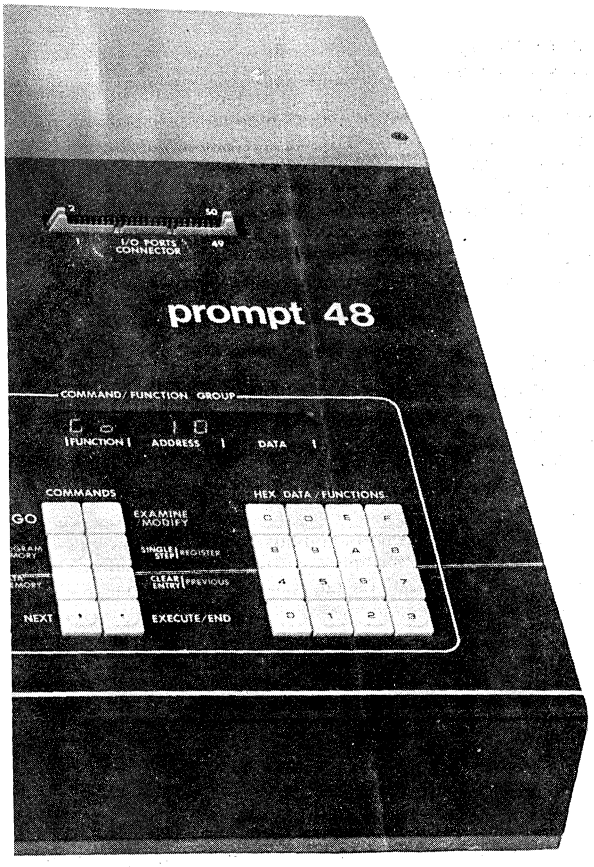
Commands

PROMPT 48's commands are grouped and color-coded to simplify access to the 8748's separate program and data memory. Registers, data memory, or program memory, may be examined and modified with the examine and modify commands. Then either the next or previous register and memory locations may be accessed with one keystroke. Programs may be exercised in three modes. The go no break (GO NO BREAK) runs in real time. The go with break (GO WITH BREAK) mode is not real time — after each instruction the MCS-48 program counter is compared against pending breakpoints. If no break is encountered, execution resumes. The go single step (GO SINGLE STEP) mode exercises one instruction at a time. Commands are like sentences, with parameters separated by ⊡ NEXT. Each command ends with ⊡ EXECUTE/END. In addition to the PROMPT basic commands, thirteen functions simplify programming. Each is started merely by pressing a hex data/function key and entering parameters as required, as shown in Table 1.



Cable Interface

An optional cable, PROMPT-SER, directly connects the PROMPT system to virtually any terminal via a rear access slot.



Key	Function	Operation
2	Port 2 map	Allows specification of direction of each pin on port 2. Port 2 is multiplexed to address external program memory and expand I/O. Thus it must be buffered; the P2 map command establishes the direction of buffering.
3	Program EPROM	Programs 8748 EPROMs.
4	Byte search (with optional mask)	Sweeps through register, data, or program memory searching for byte matches. Starting and ending memory addresses are specified.
5	Word search (with optional mask)	Sweeps through register, data, or program memory searching for word matches. Starting and ending memory addresses are specified.
6	Hex calculator	Computes hexadecimal sums and differences.
7	8748 program for debug	Similar to program EPROM, but ensures that the top of program memory contains monitor re-entry code for debugging.
8	Compare	Verifies any portions of EPROM program memory against PROMPT memory.
9	Move memory	Allows blocks of register, data, or program memory to be moved.
A	Access	Specifies one of six access modes for PROMPT 48. For example EPROM, PROMPT RAM, or external program memory, and a variety of input/output options may be selected.
B	Breakpoint	Allows any or all of the eight breakpoints to be set and cleared.
C	Clear	Clears portions of register, data, or program memory.
D	Dump	Dumps register, data, or program memory to PROMPT's serial channel: for example, a teletypewriter paper tape punch.
E	Enter	Enters (reads) register, data, or program memory from PROMPT's serial channel.
F	Fetch	Fetches programs from EPROM to PROMPT RAM.

Table 1. PROMPT 48 Commands and Functions

Access

Easy access to the pins of the executing processor is provided via the I/O ports and bus connector. Only the EA external access, SS single step, and X1, X2 clock inputs are reserved for the PROMPT system.

Expansion

Program or data memory may be expanded beyond that provided on-chip or in the PROMPT system. I/O ports may be expanded, as with the 8243, or peripheral controllers may be memory-mapped. The I/O ports and Bus connector allows the execution socket processor to be directly interfaced to prototype systems, yet be controlled from the PROMPT panel.

Control

The command/function group panel keyboard and displays completely control PROMPT 48—a teletypewriter or CRT terminal is not needed. A hyphen prompting character appears whenever a command or function can be entered. Addresses and data are shown whenever examining registers and memory. Parameters for commands and functions are also shown.

FUNCTIONAL DESCRIPTION

"PROMPT" stands for PROgraMming Tool. It is a programmer for 8748 EPROMs, and a versatile aid for debugging MCS-48 programs. Programs can be entered via its integral panel keyboard, programming socket, or serial channel. Almost any terminal can be interfaced to the serial channel, including a teletypewriter, CRT, or an Intellec microcomputer development system. Intellec PROMPT 48 simplifies the programming of MCS-48 systems. Like the 8748 it is radically new, highly integrated, and expandable. Like the MCS-48 family, it is low cost, and ideal for small applications and programs. It is a design aid, not a development system with sophisticated software and peripherals.

MCS-48 Processors

PROMPT 48 comes complete with two of Intel's revolutionary MCS-48 processors: an 8748-4 Single Component 8-Bit Microcomputer and and 8035-4 Single Component 8-Bit Microcomputer. Advances in n-channel MOS technology allow Intel, for the first time to integrate into one 40-pin component all computer functions:

8-bit CPU
1K x 8-bit EPROM/ROM program memory
64 x 8-bit RAM data memory
27 input/output lines
8-bit timer/event counter

Performance — More than 90 instructions — each one or two cycles — make the single chip MCS-48 equal in performance to most multi-chip microprocessors. The MCS-48 is an efficient controller and arithmetic processor, with extensive bit handling, binary, and BCD arithmetic instructions. These are encoded for minimum program length; 70% are single byte operation codes, and none is more than two bytes.

Flexibility — Three interchangeable, pin-compatible devices offer flexibility and low cost in development and production, as follows:

8748 — with user-programmable and erasable EPROM program memory for prototype and pre-productions systems.

8048 — with factory-programmed mask ROM memory for low-cost, high volume production.

8035 — without program memory, for use with external program memories.

Circuitry — Each MCS-48 processor operates on a single +5V supply, with internal oscillator and clock driver, and circuitry for interrupts and resets. Extra circuitry is in the 8048 ROM processor to allow low power standby operation. The 64 x 8 RAM data memory can be independently powered.

Compatibility — For systems requiring additional compatibility, the MCS-48 can be expanded with the new 8243 I/O expander, 8155 I/O and 256-byte RAM, 8755 I/O and 2K-byte EPROM, or 8355 I/O and 2K ROM devices. MCS-48 processors readily interface to MCS-80/85 peripherals and standard memories.

Memory Capacity

PROMPT 48 is a complete, fully assembled and powered microcomputer system including program memory, data memory, I/O, and system monitor beyond that available on MCS-48 single component computers. 1K bytes of PROMPT system RAM serve as "writable program memory" — a ROM simulator for the program memory on each MCS-48 computer. 256 bytes of PROMPT system RAM serve as "external data memory," beyond the 64 register bytes on each MCS-48 computer. Users may further expand program or data memory via the panel I/O ports and bus connector.

Programming

Programs written first in assembly language, are entered in machine language and debugged with calculator-like ease on the large, informative display and keyboard panel. Most MCS-48 operations can be specified with only two keystrokes. Once entered, routines can be exercised one instruction (single step) or many instructions at a time. The principal MCS-48 register — the accumulator — is displayed while single stepping. Programs can be executed in real time (GO NO BREAK) or with as many as eight different breakpoints (GO WITH BREAK).

Control

PROMPT 48 can be fully controlled either by the panel keyboard and displays, or remotely by a serial channel. Thus a teletypewriter or CRT can be used but neither is required.

Access

The PROMPT panel I/O ports and bus connector allow easy access to all MCS-48 pins except those reserved for control by the PROMPT system, namely EA external access, SS single step, and X1, X2 clock inputs.

Optional Expansion

PROMPT 48 may be expanded beyond the resources on both the MCS-48 single component computer and the PROMPT system. External program and data memory may be interfaced and input/out ports added with the 8243 I/O expander.

Documentation

The PROMPT 48 manual includes chapters for the reader with little or no programming experience. Topics treated range from number systems to microcomputer hardware design. A novel, unifying set of tutorial diagrams — MICROMAPS — simplify microcomputer concepts. PROMPT's handy, pocket-sized reference cardlet can be affixed to the mainframe. Programming pads aid in the organization and documentation of programs. These features, plus a comprehensive design library of manuals, articles, and application notes, make the Intellec PROMPT 48 ideal for the newcomer to microcomputing.

SPECIFICATIONS

Timing

Basic Instruction — 2.5 μs
Cycle Time — t_{CY} = 2.5 μs
Clock — 6 MHz ± 0.1%

Memory Bytes

The 8748 contains 64 bytes of register memory, no external data memory, and 1024 bytes of RAM program memory. The PROMPT system provides 256 bytes of external data memory, and 1024 bytes of RAM program memory. PROMPT RAM program memory can be used in place of the on-chip EPROM program memory; thus programs less than 1024 bytes may be designed. For larger programs additional memory can be directly interfaced to the MCS-48 bus via the PROMPT panel I/O ports and bus connector.

Memory Configuration

Memory	Maximum	On Chip	In PROMPT 48
Register	64	64	0
Data	3328	0	256
Program	4096	1024 EPROM	1024 RAM

I/O Ports

All MCS-48 I/O ports are accessible on the PROMPT panel connector.

Bus — A true bidirectional 8-bit port with associated strobes. If the bidirectional feature is not needed, bus can serve as either a statically latched output port or a non-latching input port. Input and output lines cannot be mixed.

Ports 1 and 2 — Data written to these 8-bit ports is latched and remains unchanged until written. As inputs these lines are not latching. The lines of ports 1 and 2 are called quasibidirectional. A special output structure allows each line of port 1 and half of port 2 to serve as an input, an output, or both. Any mix of input, output, and both lines is allowed.

T0, T1, and INT — Three pins that can serve as inputs. T0 can be designated as a clock output. Input/output can be expanded via the PROMPT panel connector with a special I/O expander (8243) or standard peripherals.

Reset and Interrupts

Reset — initializes the PROMPT system and enters the monitor.

Monitor Interrupt — exits a user program gracefully, preserving system status and entering the monitor.

User Interrupt — causes an interrupt only if the PROMPT system is running a user program.

The processor traps to location 3_{16}. The MCS-48 timer/event counter is not used by the PROMPT system and is available to the user. Either timer flag or interrupt will signal when overflow has occurred. The timer interrupt can be used only in the go-no-break (real time) mode.

EPROM Programming

PROMPT 48 provides a programming socket to directly program 8748s. Programs are loaded into the PROMPT RAM program memory via keyboard. EPROM, teletypewriter, or other serial interface. A fail-safe interlock ensures programming pulses are applied only if the device is properly inserted. Inadvertant reprogramming is prevented by a read-before-write programming algorithm. Each location may be individually programmed, one byte at a time.

Panel I/O Ports and Bus Connectors

All MCS-48 pins, except five, are accessible on the I/O ports and bus connector. The five reserved for PROMPT system control are EA external access, SS single step, X1, X2 crystal inputs, and 5V. Due to internal buffering of the MCS-48 bus, access times will be negligibly degraded by the PROMPT system. Since MCS-48 processors do not communicate internal address gate status, bus data must be driven out if neither PSEN nor RD is asserted.

System Devices

Both user programs and the PROMPT monitor enjoy access to system devices: serial I/O, panel displays, and keyboard. These are memory-mapped to program memory addresses beyond 2K.

Serial I/O — The serial I/O port (data 820_{16}, control 821_{16}) is defined by software and jumpers for 110 baud, 20 mA current loop, but can easily be jumpered for other baud rates and RS232C levels. Asynchronous or synchronous transmission, data format, control characters, and parity can be programmed.

Panel Displays — Eight display ports (data $810\text{-}817_{16}$) allow each of the panel displays to be written from user programs. Data written on a display device will time out after a fixed interval. Displays must be refreshed on a polled or interrupt-driven basis. User programs can call software drivers which provide this capability.

Keyboard — Software is used to debounce the panel keyboard (data 810_{16}). The monitor's input routines (see Software Drivers) provide this debouncing and can be called from user programs.

Commands

Single step / With break / No break } Go

Examine/modify { Register / Data / Program } Memory

Open previous/clear/entry ⊡ Next ☐ Execute/End

Functions

2 Port 2 map
3 Program EPROM (8748)
4 Search (R, D or P)* memory for 1 byte, optional mask
5 Search (R, D or P) memory for 2 bytes, optional mask
6 Hexadecimal calculator +, –
7 8748 program EPROM for debug
8 Compare EPROM with memory
9 Move memory (R, D or P)
A Access
B Breakpoint
C Clear memory (R, D or P)
D Dump memory (R, D or P)
E Enter (read) memory (R, D or P)
F Fetch EPROM program memory

Note
*R, D, or P is register, data, or program.

Software Drivers

Panel Keyboard In — KBIN, KDBIN
Panel Display Out — DGS6, DGOUT, HXOUT, BLK, REFS, ENREF
Serial Channel — CI, CO, RI, PO, CSTS

Connectors

Serial I/O — 3M 3462-0001 Flat Crimp/AMP 88106-1 Flat Crimp/ TI H312113 Solder/AMP 1-583485-5 Solder.

Panel I/O Ports and Bus Connector — 3M 3425 Flat Crimp. A complete cable set including wirewrap header for prototyping is included with each PROMPT.

Equipment Supplied

PROMPT 48 mainframe with two MCS-48 processors (8748, 8035), display/keyboard, EPROM programmer, power supply, cabinet, and ROM-based monitor.
110 V AC power cable
110 or 220 V AC
Fuse
Panel I/O ports
Bus connector cable set

Physical Characteristics

Height — 5.3 in. (13.5 cm) max
Width — 17 in. (43.2 cm)
Depth — 17 in. (43.2 cm) max
Weight — 21 lb. (9.6 kg)

Electrical Characteristics

Power Requirements — either 115 or 230V AC (± 10%) may be switch selected on the mainframe. 1.8 amps max current (at 125 V AC).
Frequency — 47-63 Hz

Environmental Characteristics

Operating Temperature — 0°C to +40°C
Non-Operating Temperature — 20°C to +65°C

Reference Manuals

9800402 — Intellec PROMPT 48 User's Manual (SUPPLIED)
9800270 — MCS-48 User's Manual (SUPPLIED)
9800255 — MCS-48 and UPI-41 Assembly Language Programming Manual (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

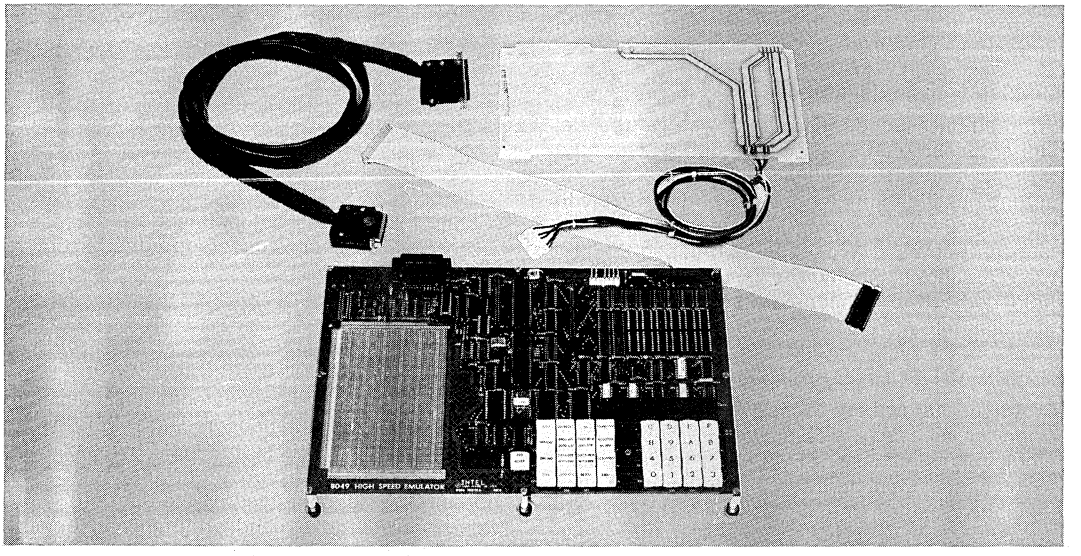
ORDERING INFORMATION

Part Number	Description
PROMPT-48 or PROMPT-48-220V	Intellec PROMPT 48 MCS-48 microcomputer design aid. Complete with two MCS-48 processors (8748 and 8305), EPROM programmer, integral keyboard, displays, and system monitor in ROM.
PROMPT-SER	Serial cable for connecting PROMPT to TTY, CRT

HSE-49
HIGH-SPEED EMULATOR

- **Single-board execution and debugging vehicle with integral keypad and display**
- **System monitor firmware in ROM**
- **Wire-wrap area for prototyping**
- **Intellec®-system compatible power pick-up card and serial-link cable**
- **Real-time 11 MHz emulation of user system**
- **Breakpoints on user program and external data addresses**
- **Examine and alter MCS® -48 registers, memory and status values**
- **No-break triggering of oscilloscope or logic analyzer**

The HSE-49™ emulator is a fully-assembled stand-alone development tool with on-board 33-key keypad, 8-character display, two 8039 microcontrollers, 2K bytes of user-program RAM, a serial port and cable, and a ROM-based monitor which supervises the emulator operation and user interface. The emulator provides a means for executing and debugging programs for the 8048/8049 family of microcontrollers at speeds up to 11 MHz. It interfaces to a user-designed system through an emulation cable and 40-pin plug, which replaces the MCS-48™device in the user's system. Using the HSE-49 keypad, a designer can run programs in real-time or single-step modes, set up to 8000 breakpoint flags, and display or change the contents of user program memory, internal and external data memory, and internal MCS-48 hardware registers. When linked to a host Intellec® development system, the HSE-49 emulator system-debugging capabilities, with the development system program assembly and storage facilities, provide the tools required for total product development.



FUNCTIONAL DESCRIPTION

The HSE-49 High-Speed Emulator is a stand-alone execution and debugging tool for 8048/8049 family microcontroller-based systems which are designed to run at speeds up to 11 MHz. It may be used alone or with other Intel microcomputer development system products to facilitate system integration early in the product development cycle, in parallel with hardware and software development. The convenient two-way debugging which early integration permits results in reduced total development time and cost.

System Development

SOFTWARE DEVELOPMENT

After an application program has been written in MCS-48 assembly language, the HSE-49 emulator may be used to debug software even if prototype hardware is not yet available.

Stand-alone Mode

The designer first hand assembles the source code from ASM-48 mnemonics into hex code, and then loads the program into the HSE-49 emulator user-program RAM through the on-board hex keypad, or through the serial port from a hex file stored on a user-supplied peripheral device. The emulator may then be used to execute the user program in a variety of debugging modes, and to alter the program as necessary. The altered program may then be uploaded to a user-supplied storage device.

Development System Mode

With an Intellec development system, the designer assembles the source code using the MCS-48 macroassembler and downloads the resulting hex file through the serial port and cable to the HSE-49 emulator user-program RAM. The emulator is then used to execute and debug the program as above. Finally, the resulting program is uploaded to the development system and stored in a disk file.

HARDWARE/SOFTWARE INTEGRATION

Prototype hardware may be developed off-board or on the wire-wrap area provided on the HSE-49 emulator board. The HSE-49 emulator interfaces to the user-system hardware through the supplied emulation cable, which plugs into the MCS-48 device socket in the user system. With the plug in place, the emulator executes code from the user-program memory and exercises the prototype hardware. Additional hardware is added as it becomes available, and the system is debugged using the emulator's capabilities to break emulation and to examine and change the user program and processor status values. The completed system may be final tested prior to ROM-code entry by replacing the HSE-49 emulation plug with a programmed 874X device in the user-system MCS-48 device socket, and running the system (with crystal input and power supplied) at full speed. Figure 1 shows a typical development configuration utilizing a host Intellec development system and the HSE-49 emulator, with the emulation cable interface to a user-designed system.

For enhanced system debugging capabilities the designer may elect to use Intel's ICE-49™ in-circuit emulator. The ICE-49 module permits real-time emulation of the user system up to 6 MHz, and offers the added benefits of symbolic debugging, 255 instruction-cycle real-time trace, and full emulation without the stack, interrupt or I/O limitations to which the HSE-49 emulator is subject (see discussion under "Limitations" heading). For further information on the ICE-49 emulator, refer to the ICE-49™ In-Circuit Emulator Data Sheet (order number 305200).

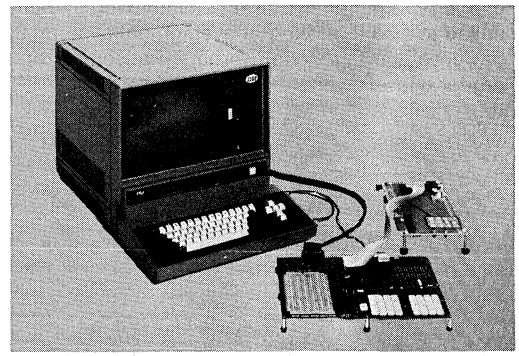


Figure 1. Intellec® System Based HSE-49 Emulator Connected to User Prototype System

Emulator Overview

EMULATION AND MASTER PROCESSORS

The user's program is emulated by an 8039 microprocessor, the emulation processor (EP), which executes code that is stored in 2K bytes of external RAM for ease of program development. Additional RAM may be added by the user in the provided sockets to expand program and external data memory to 4K bytes each.

A second microcontroller — an 8039 with off-chip ROM program memory — is used to scan the on-board keypad and display, interpret and imple-

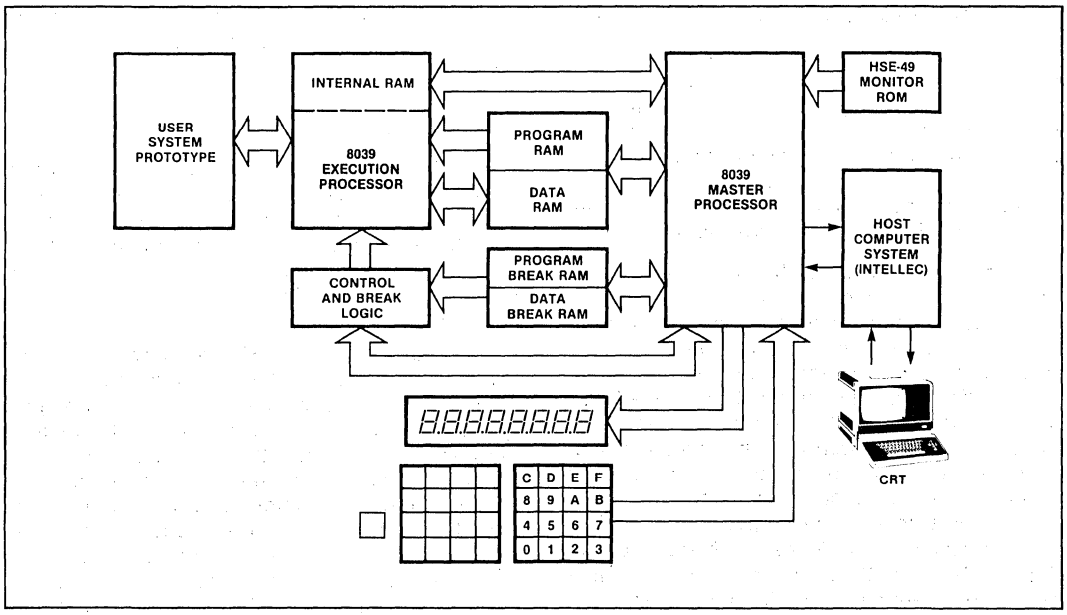


Figure 2. HSE-49 Emulator Signal Flow Diagram

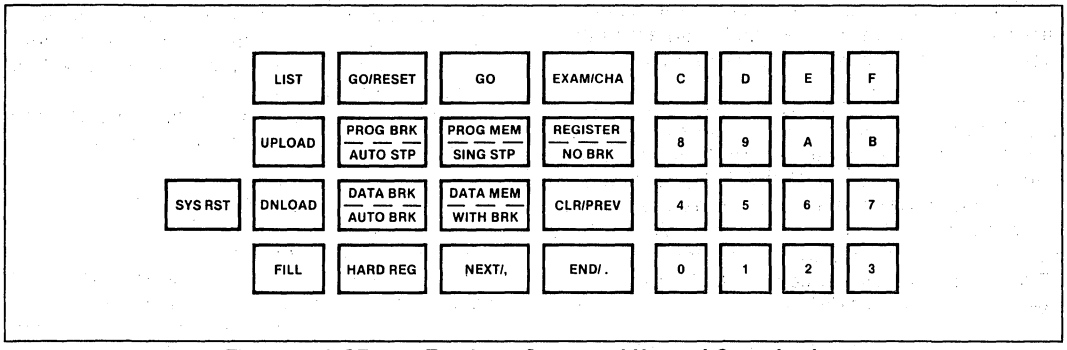


Figure 3. HSE-49 Emulator Command Keypad Organization

ment commands, drive serial interfaces, etc. In general, this master processor (MP) is used to interface the emulation processor's memory spaces with the outside world and control the operation of the EP. Figure 2 shows how the two processors interrelate with the rest of the system. Figure 3 shows the layout of the 33-key keypad through which the user interfaces to the emulator.

MP MONITOR

The monitor program executed by the MP includes commands for filling, reading, or writing the various memory spaces, including the execution processor's program RAM, external ("MOVX") data RAM, accumulator, PSW, PC, timer/counter, working registers, and internal RAM; executing the user's program from arbitrary addresses in various debugging modes; and uploading or downloading object or data files from diskettes using a host development system. No special software is needed for the Intellec system other than ISIS II Version 3.4 or later. The data format is compatible with the standard Intel hex file format produced by ASM-48; the baud rate may be altered from 110 baud (default state) up to 1200 baud from the onboard keypad. Blocks of data may be transmitted to a CRT or printer and displayed in a tabular format.

INTERPROCESSOR COMMUNICATION

An 8212 8-bit latch is used to communicate data and commands between the master and emulation processors. Under control of the MP, this register, call the "Link" register, may be logically mapped into either the program or data RAM address spaces. When this is done, the RAM in the respective memory space is disabled and the link responds to all accesses regardless of address.

When the MP detects that the EP has been halted by the breakpoint hardware, or when the operator presses a key while the program is executing, the program break sequence is initiated. The low-order 23 bytes of user-program memory is read into a buffer within the internal RAM of the MP. A short program for reading and transmitting internal EP status is written into the low-order user-program memory. (This is one of several "mini-monitors" overlayed on the user-program area.) The link register is mapped logically into the user-program memory, and loaded with the 8049 machine code for a "CALL" instruction to the mini-monitor program area. The EP is then allowed to fetch a single instruction from the link, forcing the "CALL" to the mini-monitor onto the EP data bus. The link register is then mapped to the external data RAM address space. A block diagram of the system at this point is shown in Figure 4.

From this point on, the EP executes code contained in the mini-monitor which makes the EP accumulator, timer/counter and PSW values available to the MP (through "MOVX" instructions to the link register) so that the EP internal status may be saved in the MP internal data RAM. The MP then loads a different mini-monitor into the same EP program RAM area which allows it to read and save the internal RAM of the EP.

At this point, the HSE-49 emulator may be interrogated or given instructions by the operator from the hex keypad. The emulator operation remains transparent to the user, who need not be concerned, for example, with the actual (altered) location of the EP low-order program RAM or internal data RAM.

In order to resume user-program execution, a status restoration mini-monitor is overlayed. This restores the EP internal status using a scheme analogous to the one in which the status was originally saved. The final step of the last mini-monitor is an "RETR" instruction, after which the EP is again halted. The low-order program memory saved earlier is rewritten into the appropriate area, the break logic is configured for the desired execution mode, and the EP is released to run at full speed until the next break situation is encountered.

Operation Modes

The HSE-49 firmware is a ROM-based program that provides the user with simple key-stroke commands for initiating emulation, defining breakpoints, and displaying and controlling system parameters. A summary of the HSE-49 emulator commands is given in Table 1.

SIX EMULATION MODES are provided by the HSE-49 emulator to aid in hardware and software debugging. The user may single-step through a program or have the emulator automatically step through the program with a user-defined idle time between steps. Three real-time emulation commands allow 1) real-time emulation with breakpoints not enabled (a user-accessible pulse is generated each time a breakpoint is encountered, however, facilitating user-defined logic analysis), 2) real-time emulation with breakpoints enabled, and 3) real-time emulation with automatic breakpointing, whereby the emulator executes in real time between breakpoints, and pauses at each breakpoint for a user-defined time before automatically resuming real-time emulation. A final command initiates real-time execution, beginning emulation at user-program location 000H, from the Emulator Processor hardware reset state.

BREAKPOINTS may be set at any combination of program memory and external data memory address locations from 000H to FFFH. This unlimited capability to specify breakpoints for all possible combinations of addresses complements the somewhat different breakpointing features available with the ICE-49 emulator.

The ICE-49 emulator permits the symbolic specification of breakpoints on program memory and external data memory addresses, or external data memory address reads or writes individually, and upon the input of an external synchronization signal.

INTERROGATION AND UTILITY commands are provided by the HSE-49 emulator which allow the user to examine, change or fill the various emulator memory spaces. Additional commands are provided to upload or download the contents of the memory spaces to or from a host Intellec development system, or other peripheral device, through the HSE-49 emulator serial port.

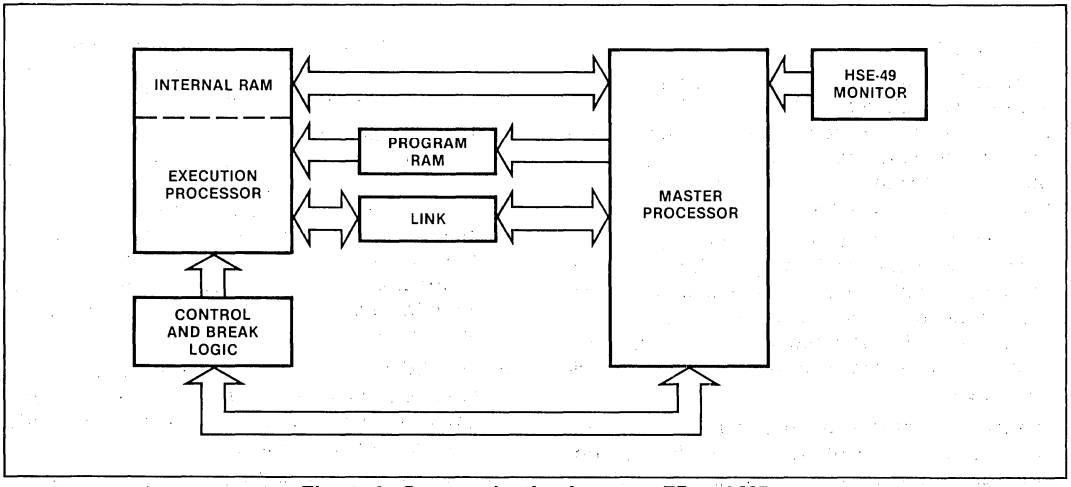


Figure 4. Communication between EP and MP

Table 1. HSE-49 Emulator Command Description

Command	Command Modifier	Description
GO		Begins emulation:
	NO BRK	Real-time breakpoints not enabled
	W/BRK	Real-time breakpoints enabled
	SING STP	Steps program one instruction
	AUTO STP	Automatically steps/pauses/steps/...
	AUTO BRK	Automatically emulates real-time/pauses at breakpoint/emulates real-time/...
GO/RESET	(NONE)	Begins real-time emulation from EP hardware reset state, beginning at program location 000H
B, C	(PROG MEM, DATA MEM)	Sets (B) or Clears (C) breakpoint at specified address within Program or External Data memory
SYS RST	(NONE)	Resets emulation and master processors and clears all breakpoints
EXAM/CHA	*	Examine/Change memory location
FILL	*	Fill range of memory addresses with a single data value
LIST	*	List memory to output device through HSE-49 serial port
DNLOAD	*	Download hex-file format memory to HSE-49 emulator through HSE-49 serial port
UPLOAD	*	Upload memory within a range of addresses through HSE-49 serial port to Intellec Development System or external peripheral device
	*:	* = Memory types allowed for above commands:
	PROG MEM	User-program memory
	DATA MEM	External data memory (if installed)
	PROG BRK	User-program breakpoint memory
	DATA BRK	External data breakpoint memory
	REGISTER	Register memory and internal data memory
	HARD REG	Hardware registers/system control parameters

Limitations

The HSE-49 emulator was not designed to have the same capabilities that Intel's ICE™in-circuit emulators have, and certain features have been deleted to keep the circuitry relatively simple. As a result, the following limitations exist and should be taken into account when using the system.

1. As explained previously, user-program execution is terminated by forcing the EP to execute a "CALL" instruction to the mini-monitor. Because this requires one level of the EP subroutine stack, the user program can be using a maximum of seven levels of stack when a break is initiated.
2. Because program execution is initiated by forcing the EP to execute an "RETR" instruction, the EP interrupt-in-progress flip-flop will be cleared. Therefore, if interrupts are enabled, emulation should not be resumed from an address within an interrupt servicing routine; if it is, the EP may incorrectly recognize an interrupt request which should be ignored.
3. The I/O status of ports P0 and $P2_2$-$P2_3$ with respect to user-supplied hardware is determined by the HSE-49 emulator hardware configuration rather than by software. Therefore the I/O modes of these ports may not be altered while a program is executing. These ports may be configured as is inputs, latched outputs or bidirectional ports by changing socketed HSE-49 emulator hardware.
4. The "ANL BUS, #nn" and "ORL BUS, #nn" instructions may not be used in the user program, as external hardware cannot properly restore these functions.

Several other minor limitations with the HSE-49 emulator operation are explained in the Operating Instructions.

SPECIFICATIONS

Equipment Supplied

Printed Circuit Board with Integral Keypad, Display, ROM Monitor, (2) 8039 microprocessors and 2K bytes user program RAM
Emulation Cable and Plug
Serial-Link Cable
Power Pick-up Card with Cable
Power Cable

Emulation Clock

11 MHz supplied, or user supplied crystal for 3.6 MHz to 11 MHz clock

Serial I/O

20 mA Current Loop or RS232 (jumper selectable)

Physical Characteristics

Width: 14.0 in (35.6 cm)
Length: 10.0 in. (25.4 cm)
Height: 0.5 in. (1.27 cm)
Packaged Weight: 4.0 lb (1.8 kg)

D.C. Electrical Characteristics

V_{CC} = +5V ±5%
I_{CC} = 2.0A max; 1.5A typical
V_{RS232} = +12V ±5%; −12V ±5%
I_{RS232} = 0.020A max; 0.015A typical

(V_{RS232} required only if using RS232 mode of serial port)

Environmental Characteristics

Operating Temperature: 0° to 55°C

Operating Humidity: Up to 90% relative humidity without condensation

ORDERING INFORMATION

Part Number	Description
MCI-49-HSE	8048/8049 family CPU high-speed (11 MHz) emulator, cable assembly and ROM firmware

ICE-49
MCS-48 IN-CIRCUIT EMULATOR

Emulates 8049, 8048, 8748, 8039, 8035, and 8021* Microcomputers

Extends Intellec microcomputer development system debug power to user configured system via external cable and 40-pin plug, replacing system MCS-48 device

Emulates user system MCS-48 device in real time

Shares static RAM memory with user system for program debug

Provides hardware comparators for user designated break conditions

Eliminates need for extraneous debugging tools residing in user system

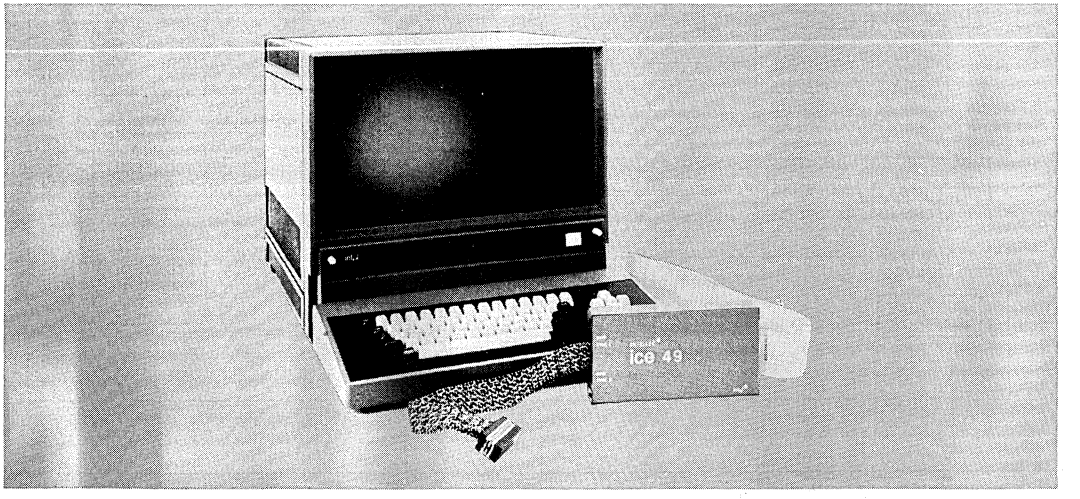
Collects bus, register, and MCS-48 status information on instructions emulated

Provides capability to examine and alter MCS-48 registers, memory, and flag values, and to examine pin and port values

Integrates hardware and software efforts early to save development time

The ICE-49 MCS-48 In-Circuit Emulator module is an Intellec-resident module that interfaces with any MCS-48 system. The MCS-48 family consists of the 8049, 8048, 8748, 8039, 8035, and 8021 microcomputers. The ICE-49 module interfaces with an MCS-48 system through a cable terminating in an MCS-48 pin-compatible plug which replaces the MCS-48 device in the sytem. With the ICE-49 plug in place, the designer has the capability to execute the system in real time while collecting up to 255 instruction cycles of real-time trace data. In addition, he can single step the system program to monitor more closely the program logic during execution. Static RAM memory is available through the ICE-49 module to emulate MCS-48 program and data memory. The designer can display and alter the contents of data and replacement RAM control memory, internal MCS-48 registers and flags and I/O ports. Powerful debug capability is extended into the MCS-48 system while ICE-49 debug hardware and software remain inside the Intellec system. Symbolic reference capability allows the designer to use meaningful symbols rather than absolute values when examining and modifying memory, registers, flags, and I/O ports in this system.

*EM1 emulator board is also required.



FUNCTIONAL DESCRIPTION

Debug Capability Inside User System

The ICE-49 module provides the user with the ability to debug a full prototype or production system without introducing extraneous hardware or software test tools. The module connects to the user system through the socket provided for the MCS-48 device in the user system. Intellec memory is used for the execution of the ICE-49 software. The Intellec console and file handling capabilities provide the designer with the ability to communicate with the ICE-49 module and display information on the operation of the prototype system. The ICE-49 module block diagram is shown in Figure 1.

Batch Testing

In conjunction with the ISIS-II diskette operating system, the ICE-49 module can run extensive system diagnostics without operator intervention. The designer or test engineer can define a complete diagnostic exercise, which is stored in a file on the diskette. When activated with an ISIS-II submit command, this file can instruct the ICE-49 module to execute the diagnostic routine and store the results in another file on the diskette. Results are available to the designer at his convenience. In this way, routine diagnostics and long term testing may be done without tying up valuable manpower.

Integrated Hardware/Software Development

The user prototype need consist of no more than an MCS-48 socket and timing logic to begin integration of software and hardware development efforts. Through the ICE-49 module mapping capabilities, Intellec system resources can be accessed to replace prototype memory. Hardware designs can be tested using the system software to drive the final product. Thus, the system integration phase, which can be costly when attempting to mesh completed hardware and software products, becomes a convenient two-way debug tool when begun early in the design cycle.

Real-Time Trace

The ICE-49 module captures trace information while the designer is executing programs in real time. The instructions executed, program counter, port values for bus 0, port 1 and port 2, and the values of selected MCS-48 status lines are stored for the last 255 instruction cycles executed. When retrieved for display, code is disassembled for user convenience. This provides data for determining how the user system was reacting prior to emulation break, and is available whether the break was user initiated or the result of an error condition. For more detailed information on the actions of internal registers, flags, or other system operations, the user may operate in single or multiple step sequences tailored to system debug needs.

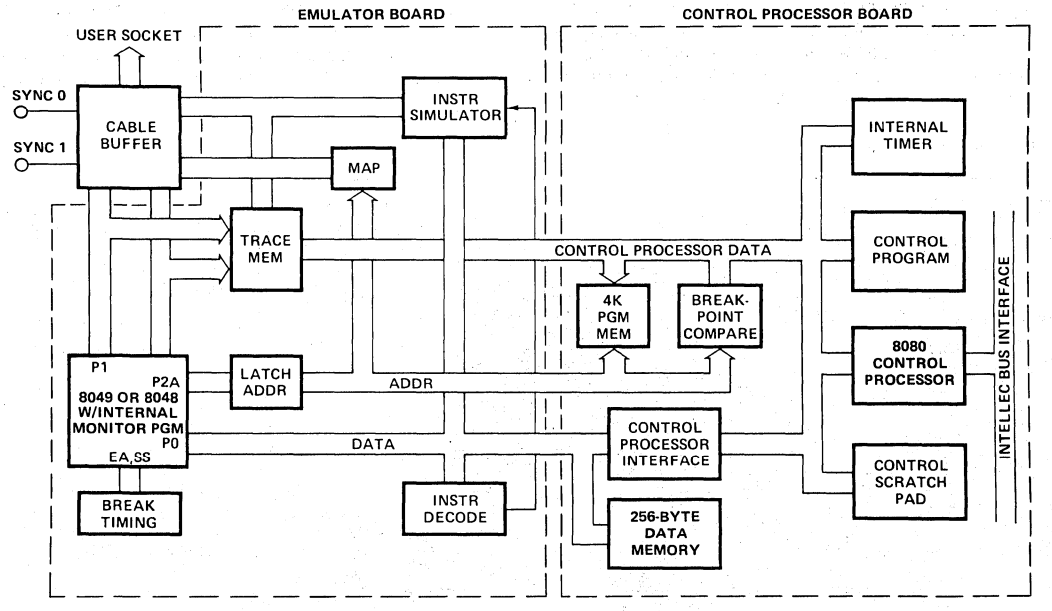


Figure 1. ICE-49 Module Block Diagram

Memory Mapping

The 8049, 8748 and 8048 contain internal program and data memory. Both program and data memory can be expanded using external memory devices.

Internal Memory — When the MCS-48 microcomputer is replaced by the ICE-49 socket in a system, the ICE-49 module supplies static RAM memory as a replacement for the internal microcomputer memory. The ICE-49 module has enough RAM memory available to emulate up to the total 4K control memory capability of the system. The ICE-49 module also provides for up to 384 bytes of data memory.

External Memory — The ICE-49 module separates replacement control memory into sixteen 256-byte blocks. Replacement external data memory consists of one 256-byte block. Each block of memory can be defined separately as supplied by the user system or supplied by the ICE-49 module. The user may assign ICE-49 equivalent memory to take the place of external memory not yet supplied in his system.

Symbolic Debugging

ICE-49 software provides symbolic definition of all MCS-48 registers, flags, and selected MCS-48 pins. Symbolically defined pseudo registers provide access to the sense of MCS-48 flip flops which enable time, counter, interrupt, and flag-0/flag-1 options. In addition, the user may reference locations in program and data memory, or their contents, symbolically. The user symbol table generated along with the object file during a program assembly may be loaded to Intellec memory for access during emulation. The user is encouraged to add to this symbol table any additional symbolic values for memory addresses, constants, or variables he may find useful during system debugging. Symbols may be substituted for numeric values in any of the ICE-49 commands. Symbolic reference is a great advantage to the system designer. He is no longer burdened with the need to recall or look up those addresses of key locations in his program that can change with each assembly. Meaningful symbols from his source program may be used instead. For example, the command:

```
GO FROM .START TILL XDATA. RSLT WRITTEN
```

begins execution of the program at the address referenced by the label START in the designers assembly program. A breakpoint is set to occur the first time the microprocessor writes to the external data memory location referenced by RSLT. The designer does not have to be concerned with the physical locations of START and RSLT. The ICE-49 software driver supplies them automatically from information stored in the symbol table.

Hardware

The ICE-49 module is a microcomputer system utilizing Intel's 8049 or 8048/8748 microcomputer as its nucleus. The 8049 provides the 8049, 8039 emulation characteristics. The 8048/8748 provides the 8748/8048/8035/8021 emulation characteristics. The ICE-49 module uses an Intel 8080 to communicate with the Intellec host processor via a common memory space. The 8080 also controls an internal ICE-49 bus for intramodule communication. ICE-49 hardware consists of two PC boards, the controller board, and the emulator board, all of which reside in the Intellec chassis. A cable interfaces the ICE-49 boards to the MCS-48 system. The cable terminates in a MCS-48 pin compatible plug which replaces any MCS-48 device in the user system. The ICE-49 module block diagram is shown in Figure 1.

Real-Time Trace

Trace Buffer — While the ICE-49 module is executing the user program, it is monitoring port, program counter, data, and status lines. Values for each instruction cycle executed are stored in a 255 x 44 real-time RAM trace buffer. A resetable timer resident on the controller board counts instruction cycles.

Controller Board

The ICE-49 module talks to the Intellec system as a peripheral device. The controller board receives commands from the Intellec system and responds through the parameter block. Three 15-bit hardware breakpoint registers are available for loading by the user. While in emulation mode, a hardware comparator is constantly monitoring address and status lines for a match to terminate an emulation. The breakpoint registers provide a signal when a match is detected. The user may disable the emulation break capability and use the signal to synchronize other debug tools. The controller board returns real-time trace data, MCS-48 register, flag, and pin values, and ICE-49 status information, to a control block in the Intellec system when emulation is terminated. This information is available to the user through the ICE-49 interrogation commands. Error conditions, when present, are automatically displayed on the Intellec system console. The controller board also contains static RAM memory, which can be used to emulate MCS-48 program and data memory in real time. 4K of memory is available in sixteen 256-byte pages to emulate MCS-48 PROM or PROM program memory. A 256-byte page of data memory is available to access in place of MCS-48 external data memory. The controller board address map directs the ICE-49 module to access either replacement ICE-49 memory or actual user system external memory in 256-byte segments based on information provided by the user.

Emulator Board

The emulator board contains the 8049* and peripheral logic required to emulate the MCS-48 device in the user system. A software selectable 6 MHz or 3 MHz clock drives the emulated MCS-48 device. This clock can be disabled and replaced with a user supplied TTL clock in the user system.

*Use 8048 with internal monitor program when emulating 8748/8048/8035/8021.

Cable Card

The cable card is included for cable driving. It transmits address and data bus information to the user system through a 40-pin connector which plugs into the user system in the socket designed for the MCS-48 device.

Software

The ICE-49 software driver is a RAM-based program which provides the user with an easy to use command language (see Table 1, Table 2, and Table 3) for defining breakpoints, initiating real-time emulation or single step operation, and interrogating and altering user system status recorded during emulation. The ICE-49 command language contains a broad range of modifiers to provide the user with maximum flexibility in defining the operation to be performed. The ICE-49 software driver is available on diskette and operates in 32K of Intellec RAM memory.

Command	Operation
Enable	Activates breakpoint and display registers for use with go and step commands.
Go	Initiates real-time emulation and allows user to specify breakpoints and data retrieval.
Step	Initiates emulation in single instruction increments. Each step is followed by register dump. User may optionally tailor other diagnostic activity to his needs.
Interrupt	Emulates user system interrupt.

Table 1. ICE-49 Emulation Commands

Command	Operation
Display	Prints contents of memory, MCS-48 device registers, I/O ports, flags, pins, real-time trace data, symbol table, or other diagnostic data on list device.
Change	Alters contents of memory, register, output port, or flag. Sets or alters breakpoints and display registers.
Map	Defines memory status.
Base	Establishes mode of display for output data.
Suffix	Establishes mode of display input data.

Table 2. ICE-49 Interrogation Commands

Command	Operation
Load	Fetches user symbol table and object code from input device.
Save	Sends user symbol table and object code to output device.
Define	Enters symbol name and value to user symbol table.
Move	Moves block of memory data to another area of memory.
List	Defines list device.
Exit	Returns program control to ISIS-II.
Evaluate	Converts expression to equivalent values in binary, octal, decimal, and hex.
Remove	Deletes symbols from symbol table.
Reset	Reinitializes ICE-49 hardware.

Table 3. ICE-49 Utility Commands

SPECIFICATIONS

ICE-49 Operating Environment

Required Hardware

Intellec microcomputer development system
System console
Intellec diskette operating system
ICE-49 Module

Required Software

System monitor
ISIS-II

Equipment Supplied

Printed circuit boards (control board, emulator board)
Interface cables and buffer module
ICE-49 software, diskette-based version (single density or double density)
8048 with internal monitor program

System Clock

Crystal controlled 6.0 MHz internal, 3.0 MHz internal or user supplied TTL external: software selectable.

Physical Characteristics

Width — 12.00 in. (30.48 cm)
Height — 6.75 in. (17.15 cm)
Depth — 0.50 in. (1.27 cm)
Weight — 8.00 lb. (3.64 kg)

Electrical Characteristics

DC Power Requirements

V_{CC} = +5V ±5%
I_{CC} = 10A max; 7.0A typ
V_{DD} = +12V ±5%
I_{DD} = 79 mA max; 45 mA typ
V_{BB} = −10V ±5%
I_{BB} = 20 mA max

Input Impedance — @ICE-49 user socket pins:

V_{IL} = 0.8V (max), I_{IL} = -1.6 mA,
V_{IH} = 2.0V (min), I_{IH} = 40 μA

For Bus:
V_{IL} = 0.8V (max), I_{IL} = -250 μA
V_{IH} = 2.0V (min), I_{IH} = 20 μA

Output Impedance — @ICE-49 user socket pins:

P1, P2:
V_{OL} = 0.5V (max), I_{OL} = 16 mA
V_{OH} = V_{CC} (10K pullup)

For Bus:
V_{OL} = 0.5V (max), I_{OL} = 25 mA
V_{OH} = 3.65V (min), I_{OH} = -1 mA

Others:
V_{OL} = 0.5V (max), I_{OL} = 16 mA
V_{OH} = 2.4V (max), I_{OH} = -400 μA

Environmental Characteristics

Operating Temperature — 0°C to 40°C (Room Temperature)

Operating Humidity — Up to 95% relative humidity without condensation

Reference Manuals

9800632 — ICE-49 Operator's Manual (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
MDS-49-ICE	8049, 8048, 8039, 8748, 8035, 8021 CPU in-circuit emulator. Cable assembly and interactive diskette software included

EM1
8021 EMULATION BOARD

EPROM functional equivalent of 8021 — single component 8-bit microcomputer

Based on 8748 — user programmable/ erasable EPROM 8-bit computer

Operates with ICE-49™ module to provide full in-circuit debugging of 8021 prototype system

Connects to prototype system through 8021 pin compatible plug

On-card 3.0 MHz or external TTL driven clock

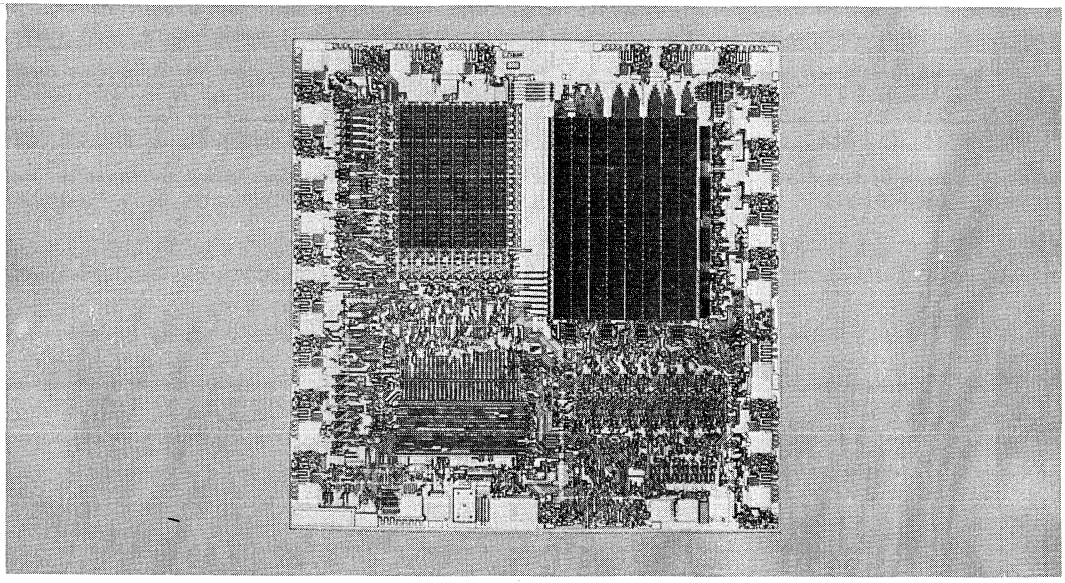
Portable 4" x 7" microcomputer circuit assembly

The EM1 emulator board is a ready-to-use 4" x 7" microcomputer circuit assembly that emulates the Intel 8021 microcomputer. A 12-inch flat-cable assembly connects the board to the 8021 socket in a prototype system. The board is designed so that it can be mounted either as a stand-alone unit, or within the prototype assembly.

The 8021 microcomputer has 1K x 8 mask-programmable ROM program memory and 64 by 8 RAM data memory. The EM1 is controlled by an Intel 8748, with 1K of EPROM program memory and a 64 byte data memory. The EPROM can be programmed and erased repeatedly during hardware and software development. The EM1 has several ancillary circuits that perform the following functions which are specific to the 8021:

Zero crossing detector
Crystal controlled clock/buffer
Port 0 simulator

For prototype debugging, the 8748 can be removed from its socket and replaced with a cable to an ICE-49 module. When used with the EM1, ICE-49 module emulates the 8021 in real-time, or single-steps the 8021 program at the user's command. A full range of capabilities for examining and modifying 8021 memory and status are supplied through ICE-49 module.



HARDWARE

The EM1 emulation board uses the 8748 to perform the emulation.

P0 Simulator

Port 0 of the 8021 is a quasi*-bidirectional port. The P0 simulator converts the data bus of the 8748 into a quasi-bidirectional port.

Crystal Control Clock Buffer

The EM1 allows user to select an on-board oscillator or a TTL clock driven from the 8021 user's prototype system via a Cambion Suitcase jumper.

Jumper	Position	State
W1	A — B	On-Board
	C — D	External TTL Clock

*A bidirectional port which serves as an input port, output port, or both even though outputs are statically latched.

Zero Cross Detection Simulator

The zero cross detection simulator enables the 8748's T1 input to detect zero-crossings. The circuitry provides a high level signal on a positive crossing and a low level signal on a negative crossing of zero to the T1 input of the 8748.

Reset Buffer

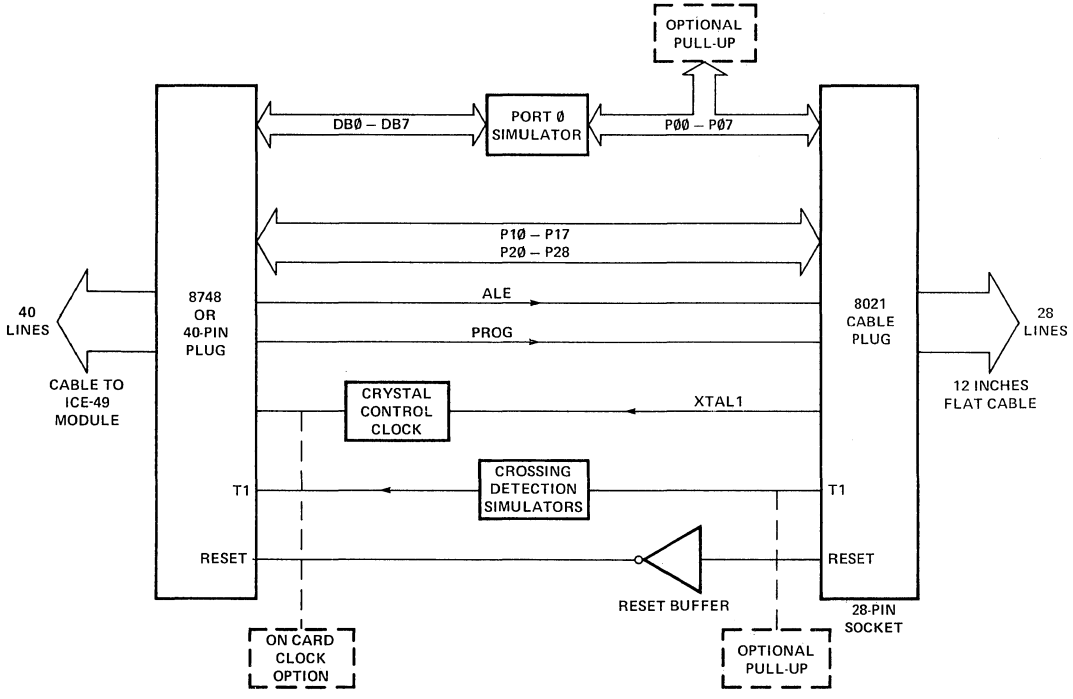
The 8021 resets on a logic HIGH level signal. However, the 8748 resets on a logic LOW level, thus an inverter is provided on the EM1 to make the two chips compatible.

Optional Pull-Ups

Resistors are provided to simulate the optional pull-up resistors on T1 input and Port 0 of the 8021. A removable resistor pack is used on Port 0. The T1 input pull up can be installed by soldering in a 50K resistor.

Software

When emulating the 8021 with EM1, the user must observe the 8021 instruction set.



SPECIFICATIONS

Operating Environment

Stand-Alone
  Required Hardware:
    EM1 emulation board

In-Circuit Emulation
  Required Hardware:
    EM1 emulation board
    Intellec Microcomputer Development System configurated with ICE-49 module

Equipment Supplied

EM1 printed circuit board

12" long flat cable terminating in 28-pin plug, pin compatible with 8021

EM1 Operator's Manual

System Clock

Crystal controlled 3.0 MHz on board or user supplied TTL external clock: hardware jumper selectable.

Physical Characteristics

Width: 7.0 in (17.78 cm)
Height: 4.0 in. (10.16 cm)
Depth: 0.75 in. (1.91 cm)
Weight: < 1.0 lbs. (0.45 kg)

Electrical Characteristics

DC Power:
  V_{CC} 5V ±5%
  I_{CC} 300 mA (max.)

Environmental Characteristics

Operating Temperature: 0 — 55°C

Operating Humidity: up to 95% relative humidity without condensation

ORDERING INFORMATION

PART NUMBER	Description
MDS-EM1	8021 Emulation Board

EM2
8022 EMULATION BOARD

Portable 4.25″ x 2.75″ microcomputer circuit assembly

Connects directly into prototype system through Intel® 8022* pin compatible socket

Provides Intel® 8755A — 2K x 8 EPROM

EPROM functional and electrical equivalent of Intel® 8022 — single component 8-bit computer

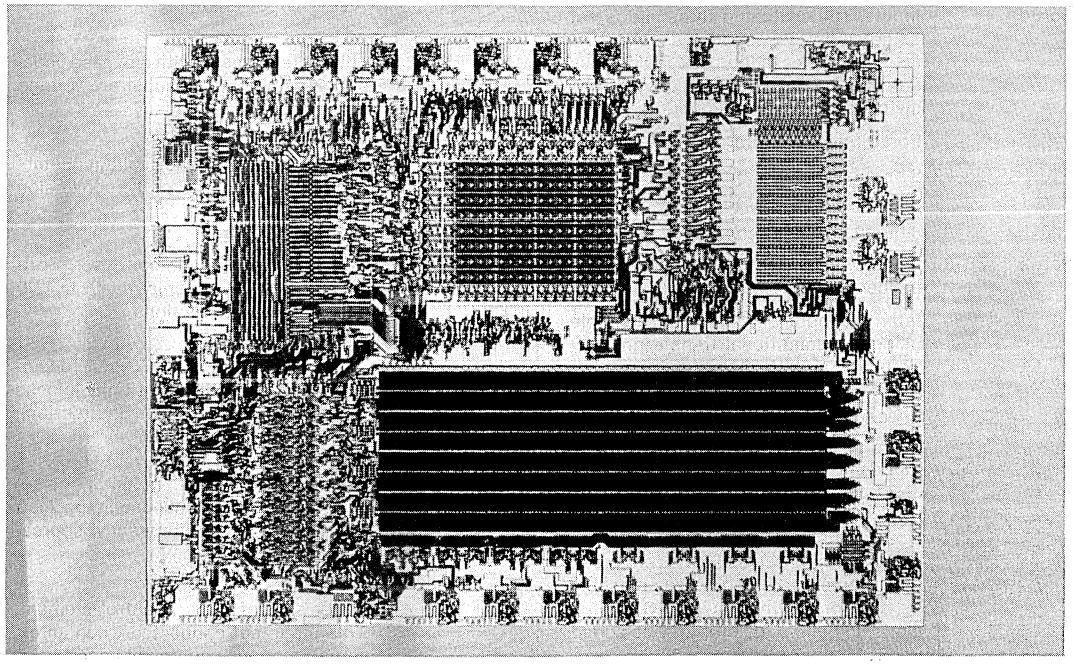
The EM2 emulator board is a ready-to-use 4.25″ x 2.75″ microcomputer circuit assembly that emulates the Intel® 8022 single chip microcomputer. The emulator board is designed to plug directly into the 8022 socket. No interfacing and interconnection cables are necessary. Power is obtained from the user's system.

The EM2 emulator board provides the user a full EPROM functional and electrical equivalent of the 8022 single component 8-bit microcomputer.

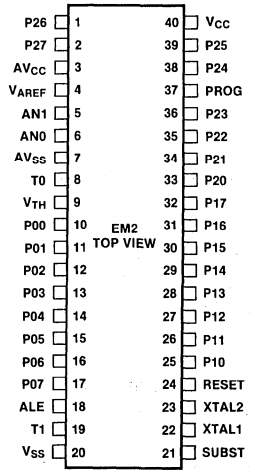
The EM2 emulator board consists of an Intel® 8022 emulator chip and an Intel® 8755A, providing the EM2 emulator board with a 2K x 8 EPROM program memory which can be programmed and erased repeatedly during hardware and software development.

The 8022E emulator chip is a modified version of the 8022 intended for use in design support systems. Instead of using resident ROM memory as the 8022, the 8022E uses an external 2K EPROM 8755A memory for program storage, allowing easy program modification.

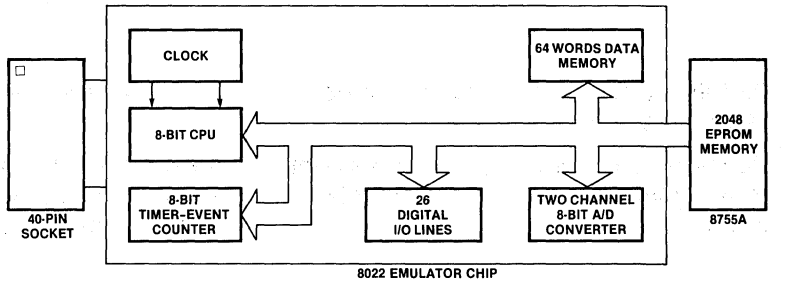
*See Intel® 8022 Data Sheet.



40-PIN SOCKET CONFIGURATION



EM2 BLOCK DIAGRAM



□ PIN 1
SQUARE SOLDER PAD

PIN DESCRIPTION

Designation	Pin #	Function
V_{SS}	20	Circuit GND potential.
V_{CC}	40	+5V circuit power supply.
PROG	37	Output strobe for Intel® 8243 I/O expander.
P00–P07 Port 0	10–17	8-bit open-drain port with comparator inputs. The switching threshold is set externally by V_{TH}. Optional pull-up resistors may be added via ROM mask selection. (The emulator board has switch selection of this option.)
V_{TH}	9	Port 0 threshold reference pin.
P10–P17 Port 1	25–32	8-bit quasi-bidirectional port.
P20–P27	33–36	8-bit quasi-bidirectional port.
Port 2	38–39 1–2	P20–P23 also serve as a 4-bit I/O expander for Intel® 8243.
T0	8	Interrupt input and input pin testable using the conditional transfer instructions JT0 and JNT0. Initiates an interrupt following a low level input if interrupt is enabled. Interrupt is disabled after a reset.
T1	19	Input pin testable using the JT1 and JNT1 conditional transfer instructions. Can be designated the timer/event counter input using the STRT CNT instruction. Also serves as the zero-cross detection input to allow zero-crossover sensing of slowly moving AC inputs. Optional pull-up resistor may be added via ROM mask selection.
RESET	24	Input used to initialize the processor by clearing status flip-flops and setting the program counter to zero.
AV_{SS}	7	A/D converter GND potential. Also establishes the lower limit of the conversion range.
AV_{CC}	3	A/D +5V power supply.
SUBST	21	Substrate pin used with a bypass capacitor to stabilize the substrate voltage and improve A/D accuracy.
V_{AREF}	4	A/D converter reference voltage. Establishes the upper limit of the conversion range.
AN0, AN1	6,5	Analog inputs to A/D converter. Software selectable on-chip via SEL AN0 and SEL AN1 instructions.
ALE	18	Address Latch Enable. Signal occurring once every 30 input input clocks (once every single cycle instruction), used as an output clock.
XTAL1	22	One side of crystal, inductor, or resistor input for internal oscillator. Also input for external frequency source. (Not TTL compatible.)
XTAL2	23	Other side of timing control element. This pin is not connected when an external frequency source is used.

On the EM2 Board:

The Intel® 8755A EPROM can be programmed using any of the modules listed in Table 1.

Module	Description
UPP-103	Universal PROM Programmer. Requires UPP-955, which includes 8755A Personality Card with 40-pin adapter socket.
PROMPT-48	Intellec® MCS-48 Microcomputer Design Aid. Requires PROMPT-475 Programming Adapter.
PROMPT-80/85	Intellec® 8080/8085 Microcomputer Design Aid. Requires PROMPT-975 Programming Adapter.

Table 1. 8755A Proramming Module

The 8755A EPROM is erased when exposed to light with wavelengths shorter than approximately 4000 Angstroms (Å). Sunlight and certain fluorescent lamps have wavelengths in the 3000Å to 4000Å range. If the 8755A is to be exposed to sunlight or room fluorescent lighting for extended periods, then opaque labels should be placed over the window to prevent unintentional erasure.

The recommended erasure procedure is exposure to ultraviolet light which has a wavelength of 2537Å. The integrated dose (UV intensity multiplied by exposure time) for erasure should be a minimum of 15W-sec/cm. The erasure time with this dosage is approximately 15 to 20 minutes using an ultraviolet lamp with a 12,000μ W/em^2 power rating. Place the 8755A within one inch of the lamp during erasure. Some lamps include a filter which should be removed before erasure.

SPECIFICATIONS

Operating Environment

Intel® 8755A EPROM Programming

UPP-103
PROMPT-48
PROMPT-80/85

Intellec Microcomputer Development System

Software

8048 Assembler
ISIS-II Diskette Operating System

Equipment Supplied

EM2 Printed Circuit Board
EM2 Reference Manual

Physical Characteristics

Width: 2.75 in. (6.98 cm)
Height: 4.25 in. (10.79 cm)
Depth: 1.5 in. (3.81 cm)
Weight: 0.5 lb (0.23 kg)

Electrical Characteristics

DC Power

$V_{CC} = 5V \pm 5\%$
$I_{CC} = 300$ mA (maximum)

Environmental Characteristics

Operating Temperature — 0 to 55°C

Operating Humidity — Up to 95% relative humidity without condensation

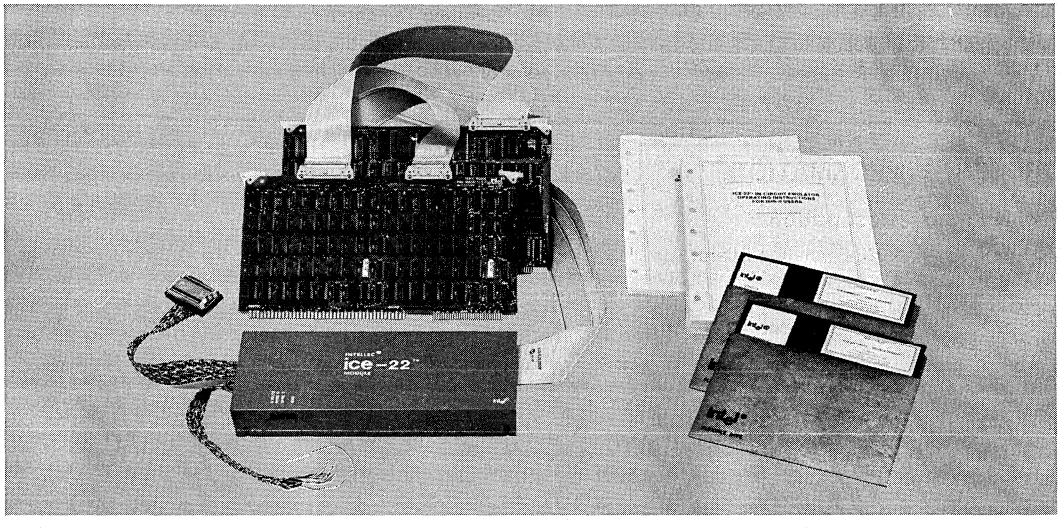
ORDERING INFORMATION

Part Number	Description
MDS-EM2	8022 Emulation Board

ICE-22™
8022 IN-CIRCUIT EMULATOR

- **Single-line assembler allows mnemonic program instruction changes**
- **Full symbolic debugging**
- **ICE™-resident user-program RAM for real-time execution**
- **Examine and alter 8022 registers, memory, and digital port values, and examine analog port data**
- **Two user-specified breakpoint registers**
- **500 instruction cycle trace**
  - **— conditionally triggered**
  - **— 16 user-definable trace probes**
  - **— symbolic groupings and display**
- **32-bit half-microsecond emulation timer**
- **HELP facility summarizes ICE-22™ command syntax at the console**
- **User confidence test of ICE-22™ hardware**

The ICE-22 module resides in the Intellec® Microcomputer Development System and interfaces to any user-designed 8022 system through a cable terminating in an 8022 emulator microprocessor and a pin-compatible plug. The emulator processor, together with 2K bytes of user-program RAM located in the ICE-22 buffer box, replaces the 8022 device in the user system while maintaining the 8022 electrical and timing characteristics. Powerful Intellec debugging functions are thus extended into the user system. Using the ICE-22 module, the designer can emulate the system's 8022, including full A/D converter function, in real-time or single-step mode. Breakpoints allow the user to stop emulation on user-specified conditions, and a trace qualifier feature allows the conditional collection of 500 instruction cycles of trace data. The ICE-22 trace includes 8022 status information and, through ICE-22 external logic probes, can provide data on up to 16 signal nodes in the user-system peripheral circuitry. For the first time in any ICE module, the designer may alter program memory using ASM-48 mnemonics and symbolic references without returning to ISIS II control. In addition, user-created peripheral chip analyzer routines may be applied to the logic probe data, thereby expanding the in-circuit emulation function to the entire system.



FUNCTIONAL DESCRIPTION

Integrated Hardware/Software Development

The ICE-22 emulator allows hardware and software development to proceed interactively. This is more effective than the traditional method of independent hardware and software development followed by system integration. With the ICE-22 module, prototype hardware can be added to the system as it is designed. Software and hardware testing occur while the product is being developed.

Conceptually, the ICE-22 emulator assists three stages of development:

It can be operated without being connected to the user's system, so ICE-22 debugging capabilities can be used in conjunction with the Intellec text editor and MCS-48™ macro-assembler to facilitate program development before any of the user's hardware is available.

Integration of software and hardware can begin when any functional element of the user system hardware is connected to the 8022 socket. As each section of the user's hardware is completed, it is added to the prototype. Thus, each section of the hardware and software is "system" tested as it becomes available.

When the user's prototype is complete, it is tested with the final version of the user system software. The ICE-22 module is then used for real-time emulation of the 8022 to debug the system as a completed unit, and verify system performance before any ROM codes are entered. A final product verification test may be performed prior to ROM code entry by using the separately available EM-2 8022 emulation board (8022 EPROM equivalent) within the eventual product package.

Thus, the ICE-22 module provides the user with the ability to debug a prototype or production system at any stage in its development without introducing extraneous hardware or software test tools.

Symbolic Debugging

The ICE-22 emulator permits the user to define and use symbolic rather than absolute references to program and data memory addresses; additional symbols are predefined by the ICE-22 software for referencing registers, flags, and input/output ports. Thus, the user need not become involved with machine code, or recall or look up the addresses of key locations in his program as they change with each assembly.

For each symbol that is used for memory reference in an ICE-22 emulator command, the emulator supplies the symbol value location as stored in the ICE-22 emulator symbol table. This table can be loaded with the symbol table produced by the assembler during application program assembly. Furthermore, the user can interactively modify the emulator symbol table by adding new symbols or changing or deleting old ones. This feature provides great flexibility in debugging and minimizes the need to work with hexadecimal values.

Through symbolic references in combination with other features of the emulator, the user can easily:

- Disassemble program memory to mnemonics
- Assemble mnemonic instructions into executable code.
- Examine or modify 8022 internal registers, data memory, or digital port contents.
- Examine analog port data
- Symbolically define groups of user probes, and use these groups to symbolically specify breakpoints and trace qualifiers, or to format external trace data output.

Operation Modes

The ICE-22 software is a RAM-based program that provides the user with easy-to-use commands for initiating emulation, defining breakpoints, controlling trace data collection, and displaying and controlling system parameters. ICE-22 commands are configured with a broad range of modifiers which provide the user with maximum flexibility in describing the operation to be performed.

EMULATION

The ICE-22 module can emulate the operation of a prototype 8022 system, including full emulation of the 8022 analog to digital converter, at real-time speed (0.6 to 3.6 MHz) or in single or multiple steps. Emulation commands to the ICE-22 module control the process of setting up, running, and halting an emulation of the user's 8022-based system. Breakpoints, comparison registers, and tracepoints enable the ICE-22 emulator to halt emulation and provide a detailed trace of execution in any part of the user's program. A summary of the emulation commands is shown in Table 1.

Breakpoints

The ICE-22 hardware includes two breakpoint registers that allow the user to halt emulation when specified conditions are met. The emulator continuously compares the values stored on the breakpoint registers with the status of specified

data, addresses, and/or external logic probes, and halts emulation when this comparison is satisfied. When an instruction initiates a break, that instruction is executed completely before the break takes place. The ICE-22 emulator then regains control of the console and enters the Interrogation Mode. With the breakpoint feature, the user can request an emulation break when his program:

- Executes an instruction at a specific address or within a range of addresses
- Executes a particular opcode
- Receives a specific signal on a logic probe, digital port pin, or group of probes or pins
- Fetches a particular data value from the user program memory

Breakpoints can be composed of conditions on 22 channels which reflect internal 8022 activities, plus the 16 external logic probe channels; all but one of the channels may be specified as "Don't Care" channels. Address ranges must be specified as a range of pages (r00H) to sFFH), a range of 16-byte paragraphs within a page (pr0H) to psFH), or a range of bytes within a paragraph (pqrH to pqsH) where, in each case, s is a digit greater than or equal to the digit r.

Table 1. Major Emulation Commands

Command	Description
GO	Begins real-time emulation and optionally specifies break conditions.
BR0, BR1, BR	Sets or displays either or both Breakpoint Registers used for stopping real-time emulation.
STEP	Begins single-step emulation and optionally specifies terminating conditions.
CR0, CR1, CR2, CR3, CR	Sets or displays comparison criteria in all or individual Comparison Registers used for stopping automatic single-step emulation.
TR	Specifies or displays trace-data collection conditions, and optionally sets Qualifier Register (QR).

Comparison Registers

Four comparison registers are provided that allow the user to halt single step emulation when the single condition specified in any one of these registers is satisfied. The comparison registers differ from the breakpoint registers in that, 1) the comparisons $<$, $\leq$, $>$, $\geq$, and $\neq$ are permitted in addition to the = condition, 2) more 8022 and ICE varibles may be compared, and 3) the comparators themselves may be variables.

Trace and Tracepoints

Tracing is used with both real-time and single-step emulation to record diagnostic information in the trace buffer as a program is executed. The information collected includes opcodes executed, program counter and Port 2 values, and 16 logic probe values for the last 500 instruction cycles. (There are one or two cycles per instruction, depending on the particular instruction.) This information can be displayed as assembler instruction mnemonics, if desired, for analysis during Interrogation or Single-Step Mode. The trace-collection facility may be set to run conditionally or unconditionally. One unique trace qualifier, specified in the same way as a breakpoint, governs conditional trace activity. It can be used to condition trace data collection to take place as follows:

- Under all conditions (constantly occurring)
- Only while the trace qualifier is satisfied
- For the 500 instruction cycles preceding the time when a trace qualifier is first satisfied (pre-triggered trace)
- For the next 500 instruction cycles after a trace qualifier is first satisfied (post triggered trace).

INTERROGATION AND UTILITY

Interrogation and utility commands give the user convenient access to detailed information about the user program and the state of the 8022 that is useful in debugging hardware and software. Changes can be made in both memory and the 8022 registers, flags, and digital port values. Commands are also provided for various utility operations such as loading and saving program files, defining symbols and logic probe groups, displaying trace data, controlling system synchronization and returning control to ISIS-II. A summary of the basic interrogation and utility commands is shown in Table 2. Two new emulator features are discussed below.

SINGLE-LINE ASSEMBLER — The single-line assembler (ASM command) is a new in-circuit emulation feature that permits the designer to examine and alter program memory using assembly language mnemonics, without leaving emulation mode or requiring time-consuming program reassembly. When assembling new mnemonic instructions into program memory, previously de-

fined symbolic references (from the original program assembly, or subsequently defined during the emulation session) may be used in the instruction operand field, and the emulator will supply the absolute address or data values as stored in the emulator symbol table. These features greatly reduce the designer's time spent translating to and from machine code and searching for absolute addresses, with a corresponding reduction in transcription errors.

HELP — The HELP file is a new ICE feature that allows the designer to display ICE-22 command syntax information at the Intellec console. By typing "HELP", a listing of all items for which help messages are available is displayed; typing "HELP <Item>" then displays relevant information about the item requested, including typical usage examples. The "HELP" listing and a "HELP ASM" message for the ASM command are shown in Table 3.

Table 2. Major Interrogation and Utility Commands

Command	Description
LOAD	Loads user object program (8022 code) into user-program memory, and user symbols into ICE-22 emulator symbol table.
DEFINE/REMOVE	Defines/removes symbols in ICE-22 emulator symbol table.
SAVE	Saves ICE-22 emulator symbol table and/or user object program in ISIS-II hexadecimal file.
LIST	Copies all emulator console input and output to ISIS-II file.
Change/Display Commands	Change or display value of symbolic reference in ICE-22 emulator symbol table, or contents of key-word references (including registers, I/O ports, and status flags), or memory references.
Group Commands	Define, change, remove, or display user-defined logic probe channel groups.
Trace Commands	Position trace buffer pointer; select and format trace output; enable or disable automatic display of trace data and register contents during single-step emulation.
PRINT	Displays trace data pointed to by trace buffer pointer.
Synchronization Line Commands	Set and display enabled/disabled status of SYNC0 and SYNC1 synchronization line outputs or latched inputs (used to allow real-time emulation or tracing to start and stop synchronously with external events).
ASM	Assembles mnemonic instructions into user-program memory, or disassembles and displays user-program memory contents.
INTERRUPT	Simulates external or timer interrupt sequence.
EVALUATE	Evaluates expression and displays resulting value.
SECONDS	Displays contents of emulation timer, in microseconds.
HELP	Displays help messages for ICE-22 emulator command-entry assistance.
EXIT	Terminates ICE-22 emulator operation.

Table 3. HELP Command

```
*HELP
Help is available for the following items.  Type HELP followed by the
item name.  (For more information about HELP, type HELP HELP.)

Real-Time Emulation:     Trace Collection:     Change/Display/Define/Remove:
GO  GR  SYO              TR  QR  SYI           ASM          REGISTER     DEFINE
BR  BRO  BRI             <MATCHSCOND>          CBYTE        STACK        REMOVE
<BREAKSREG>                                    DBYTE        SECONDS      GROUP
<MATCHSCOND>             Trace Display:        <CHANGE>     <CPUSREF>    SYMBOL
                         TRACE                 <DISPLAY>    <ICESREF>
Step Emulation:          OLDEST
STEP    SR               NEWEST                State/Mode:
CR CRO CRI CR2 CR3       MOVE                  BASE         ENABLE       SYMBOLIC
<COMPARISONSREG>         PRINT                 SUFFIX       DISABLE      SYNC
<COMPARISONSCOND>        DUMP                  LIST         RESET

EVALUATE      <ADDRESS>                 <INSTRUCTION>          <PRIMARY>
EXIT          <ADDRESSSPARTITION>       <MASKEDSCONSTANT>      <PRIMARYS10>
HELP          <CHANNEL>                 <MATCHSGROUP>          <STRING>
INTERRUPT     <CHANNELSPARTITION>       <MATCHSVALUE>          <SYMBOLICSREF>
LOAD          <EXPR>                    <MEMORYSREF>           <SYSTEMSGROUP>
SAVE          <EXPRS10>                 <NUMERICSCONSTANT>     <USERSGROUP>
              <IDENTIFIER>              <PATHNAME>



*HELP ASM
ASM - Command to display or change 8022 code memory using assembler
instructions.
   (1)  ASM <ADDRESS> [TO/LENGTH <ADDRESS>]
        (display 8022 code memory as assembler instructions)
   (2)  ASM <ADDRESS> =
                    <INSTRUCTIONSLIST>
                    END
        (change memory starting at <ADDRESS>)
           Ex:   ASM 100 =
                   MOV A,@R0
                   JNZ 100
                   END
   (3)  ASM <ADDRESS> TO/LENGTH <ADDRESS> =
                                          <INSTRUCTIONSLIST>
                                          END
        (Change several locations and perform range checking or repetition.
        If the instructions require more memory than the size of the range,
        an error occurs.  If the instructions require less memory, then
        the data is repeated until the range is filled.)
<INSTRUCTIONSLIST> - Standard 8022 instructions typed one per line.  The
   operand "#<EXPR>" can be used where "#data" is required, and the operand
   "<ADDRESS>" can be used where "addr" is required.  A continuation
   prompt ".*" is issued after each carriage return is typed.
```

SPECIFICATIONS

ICE-22 Operating Requirements

Intellec® Microcomputer Development System (32K RAM required)

System console

Intellec® Diskette Operating System (single or double density) ISIS-II v. 3.4 or later

Equipment Supplied

- Printed circuit boards (2)
- Emulation buffer box, Intellec interface cables, and user-interface cable with 8022 emulation processor
- 16 external trace probes
- Synchronization cables
- Crystal power accessory
- Operating instructions manual
- Diskette-based ICE-22 software (single and double density)

Emulation Clock

User's system clock (0.6 to 3.6 MHz) or ICE-22 crystal power accessory (3.0 MHz)

Environmental Characteristics

Operating Temperature: 0° to 40°C

Operating Humidity: Up to 95% relative humidity without condensation.

Physical Characteristics

Printed Circuit Boards

Width: 12.00 in. (30.48 cm)
Height: 6.75 in. (17.15 cm)
Depth: 0.50 in. (1.27 cm)

Buffer Box

Width: 4.5 in. (11.43 cm)
Length: 10.0 in. (25.40 cm)
Depth: 1.25 in. (3.18 cm)

Packaged Weight: 8.0 lb (3.63 kg)

Electrical Characteristics

DC Power Requirements

$V_{CC} = +5V, +5\%, -1\%$
$I_{CC} = 13.2A$ max; 11.0A typical
$V_{DD} = +12V, \pm 5\%$
$I_{DD} = 0.1A$ max; 0.05A typical
$V_{BB} = -10V, \pm 5\%$
$I_{BB} = 0.05A$ max; 0.01A typical

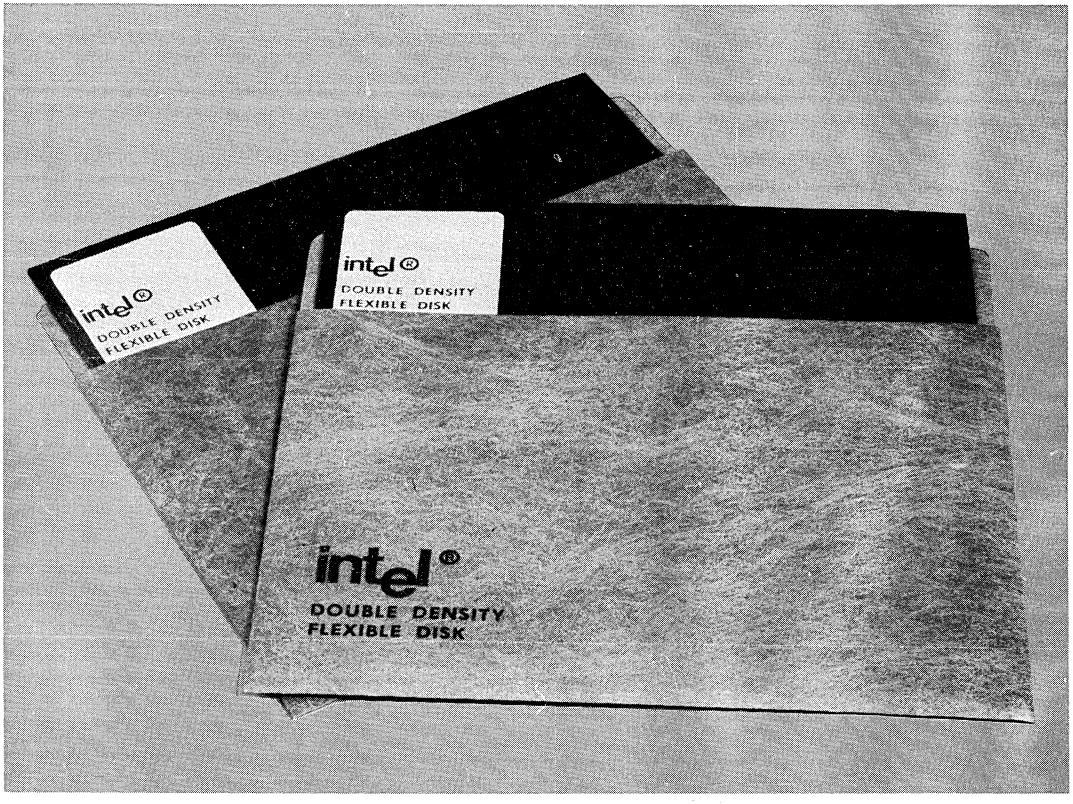
ORDERING INFORMATION

Part Number	Description
MCI-22-ICE	8022 Microcontroller In-Circuit Emulator, cable assembly and interactive diskette software

MCS™-48 DISKETTE-BASED SOFTWARE SUPPORT PACKAGE

- **Extends Intellec microcomputer development system to support MCS-48 development**
- **MCS-48 assembler provides conditional assembly and macro capability**
- **Takes advantage of powerful ISIS-II file handling and storage capabilities**
- **Provides assembler output in standard Intel hex format**

The MCS-48 assembler translates symbolic 8048 assembly language instructions into the appropriate machine operation codes, and provides both conditional and macroassembler programming. Output may be loaded either to an ICE-49 module for debugging or into a Universal PROM Programmer for 8748 PROM programming. The MCS-48 assembler operates under the ISIS-II operating system on Intellec Microcomputer Development systems.



FUNCTIONAL DESCRIPTION

The MCS-48 assembler translates symbolic 8048 assembly language instructions into the appropriate machine operation codes. The ability to refer to program addresses with symbolic names eliminates the errors of hand translation and makes it easier to modify programs when adding or deleting instructions. Conditional assembly permits the programmer to specify which portions of the master source document should be included or deleted in variations on a basic system design, such as the code required to handle optional external devices. Macro capability allows the programmer use of a single label to define a routine. The MCS-48 assembler will assemble the code required by the reserved routine whenever the macro label is inserted in the text. Output from the assembler is in standard Intel hex format. It may be either loaded directly to an in-circuit emulator (ICE-49) module for integrated hardware/software debugging, or loaded into a Universal PROM Programmer for 8748 PROM programming. A sample assembly listing is shown in Table 1.

Table 1. Sample MCS-48 Diskette-Based Assembly Listing

```
ISIS-II 8048 MACROASSEMBLER, V1.0                                    PAGE 1

LOC  OBJ                    SEQ          SOURCE STATEMENT

                             1   ;DECIMAL ADDITION ROUTINE. ADD BCD NUMBER
                             2   ;AT LOCATION 'BETA' TO BCD NUMBER AT 'ALPHA' WITH
                             3   ;RESULT IN 'ALPHA'. LENGTH OF NUMBER IS 'COUNT' DIGIT
                             4   ;PAIRS. (ASSUME BOTH BETA AND ALPHA ARE SAME LENGTH
                             5   ;AND HAVE EVEN NUMBER OF DIGITS OR MSD IS 0 IF
                             6   ;ODD)
                             7   INIT    MACRO   AUGND,ADDND,CNT
                             8           MOV     R0, #AUGND
                             9   L1:     MOV     R1, #ADDND
                            10           MOV     R2, #CNT
                            11           ENDM
                            12   ;
0001E                       13   ALPHA   EQU     30
0028                        14   BETA    EQU     40
0032                        15   COUNT   EQU     5
0100                        16           ORG     100H
                            17           INIT    ALPHA, BETA, COUNT
0100 B81E                   18+          MOV     R0, #ALPHA
0102 B928                   19+L1:       MOV     R1, #BETA
0104 BA32                   20+          MOV     R2, #COUNT
0106 97                     21           CLR     C
0107 F0                     22   LP:     MOV     A, @R0
0108 71                     23           ADDC    A, @R1
0109 57                     24           DA      A
010A A1                     25           MOV     @R0, A
010B 18                     26           INC     R0
010C 19                     27           INC     R1
010D EA07                   28           DJNZ    R2, LP
                                         END

 USER SYMBOLS
 ALPHA  0001E    BETA 0028    COUNT 0005    LP 0107
 L1     0102

 ASSEMBLY COMPLETE, NO ERRORS


ISIS-II ASSEMBLER SYMBOL CROSS REFERENCE, V1.0                       PAGE 1

SYMBOL CROSS REFERENCE

 ALPHA  13#   17
 BETA   14#   17
 COUNT  15#   17
 INIT    7#   17
 L1     19#
 LP     22#   28
```

SPECIFICATIONS

Operating Environment

Required Hardware

Intellec Microcomputer Development System
32K RAM (non-macro use)
48K RAM (use of macro facility)
One or two Floppy disk drives
  — Single or Double density
System Console
  — CRT or interactive hardcopy device

Required Software

ISIS-II Diskette Operating System

Optional Hardware

ICE-49 In-Circuit Emulator
Line Printer
Universal PROM Programmer with 8748 personality card

Shipping Media

Diskette

Reference Manuals

9800255 — MCS-48 and UPI-41 Assembly Language Programming Manual (SUPPLIED)

9800236 — Universal PROM Mapper Operator's Manual

9800306 — ISIS-II User's Guide

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Product Code	Description
MDS-D48	Diskette-based assembler for MCS-48 family of microprocessors.

MCS–51™ Development Systems

19

8051 SOFTWARE DEVELOPMENT PACKAGE

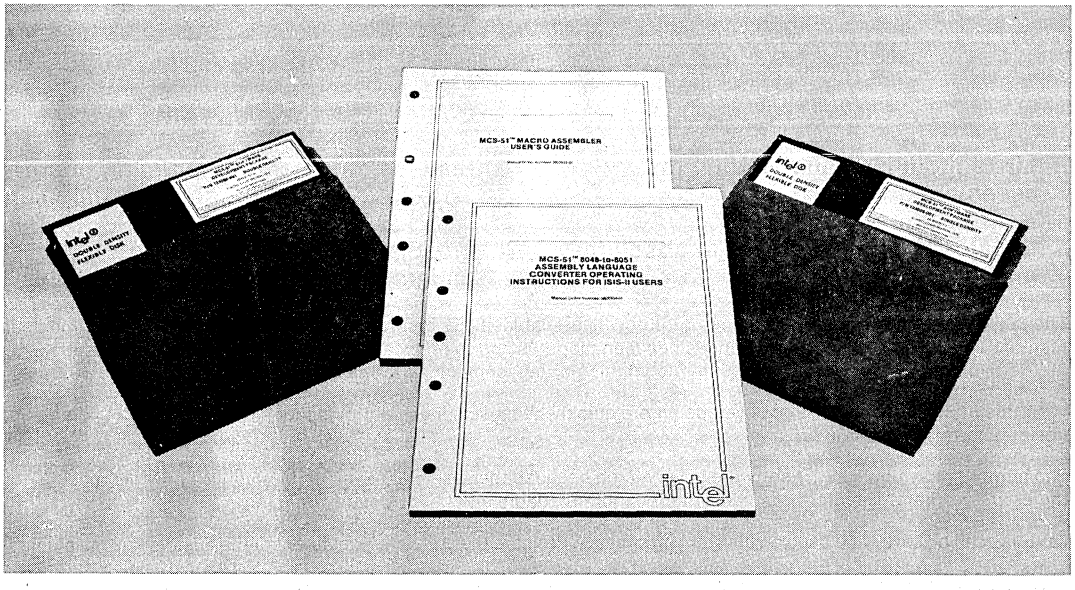
- **Symbolic relocatable assembly language programming for 8051 microcontrollers**
- **Extends Intellec® Microcomputer Development System to support 8051 program development**
- **Produces Relocatable Object Code which is linkable to other 8051 Object Modules**
- **Encourage modular program design for maintainability and reliability**
- **Macro Assembler features conditional assembly and macro capabilities**
- **CONV51 Converter for translation of 8048 assembly language source code to 8051 assembly language source code**
- **Provides upward compatibility from the MCS-48™ family of single-chip microcontrollers**

The 8051 software development package provides development system support for the powerful 8051 family of single chip microcomputers. The package contains a symbolic macro assembler and MCS-48 source code converter.

The assembler produces relocatable object modules from 8051 macro assembly language instructions. The object code modules can be linked and located to absolute memory locations. This absolute object code may be used to program the 8751 EPROM version of the chip. The assembler output may also be debugged using the ICE-51™ in-circuit emulator.

The converter translates 8048 assembly language instructions into 8051 source instructions to provide software compatibility between the two families of microcontrollers.

This diskette-based software package runs under ISIS-II on an Intellec Microcomputer Development System with 64K bytes of memory.



The following are trademarkes of Intel Corporation and may be used only to identify Intel products: BXP, CREDIT, Intellec, Multibus, i, iSBC, Multimodule, ICE, iSBX, PROMPT, iCS, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Micromap, μScope and the combination of ICE, iCS, iSBC, iSBX, MCS, or RMX and a numerical suffix.
© Intel Corporation 1981

October, 1981
162771
AFN 01944B

8051 MACRO ASSEMBLER

- **Supports 8051 family program development on Intellec® Microcomputer Development Systems**
- **Gives symbolic access to powerful 8051 hardware features**
- **Produces object file, listing file and error diagnostics**
- **Object files are linkable and locatable**
- **Provides software support for many addressing and data allocation capabilities**
- **Symbolic Assembler supports symbol table, cross-reference, macro capabilities, and conditional assembly**

The 8051 Macro Assembler (ASM51) translates symbolic 8051 macro assembly language modules into linkable and locatable object code modules. Assembly language mnemonics are easier to program and are more readable than binary or hexadecimal machine instructions. By allowing the programmer to give symbolic names to memory locations rather than absolute addresses, software design and debug are performed more quickly and reliably. Furthermore, since modules are linkable and relocatable, the programmer can do his software in modular fashion. This makes programs easy to understand, maintainable and reliable.

The assembler supports macro definitions and calls. This is a convenient way to program a frequently used code sequence only once. The assembler also provides conditional assembly capabilities.

Cross referencing is provided in the symbol table listing, showing the user the lines in which each symbol was defined and referenced.

ASM51 provides symbolic access to the many useful addressing features of the 8051 architecture. These features include referencing for bit and byte locations, and for providing 4-bit operations for BCD arithmetic. The assembler also provides symbolic access to hardware registers, I/O ports, control bits, and RAM addresses.

Math routines are enhanced by the MULtiply and DIVide instructions.

If an 8051 program contains errors, the assembler provides a comprehensive set of error diagnostics, which are included in the assembly listing or on another file. Program testing may be performed by using the Universal PROM Programmer and 8751 personality card to program the 8751 EPROM version of the chip, or by using the ICE-51 in-circuit emulator.

RL51 LINKER AND RELOCATOR PROGRAM

- **Links modules generated by the assembler**
- **Locates the linked object to absolute memory locations**
- **Enables modular programming of software for efficient program development**
- **Modular programs are easy to understand, maintainable and reliable**

The 8051 linker and relocator (RL51) is a utility which enables 8051 programmers to develop software in a modular fashion. The linker resolves all references between modules and the relocator assigns absolute memory locations to all the relocatable segments, combining relocatable partial segments with the same name.

With this utility, software can be developed more quickly because small functional modules are easier to understand, design and test than large programs.

The number of symbols in the software is very large because the assembler symbol limit applies only per module not the entire program. Therefore programs can be more readable and better documented.

Modules can be saved and used on different programs. Therefore the software investment of the customer is maintained.

RL51 produces two files. The absolute object module file can be directly executed by the 8051 family. The listing file shows the results of the link/locate process.

CONV51
8048 TO 8051 ASSEMBLY LANGUAGE CONVERTER UTILITY PROGRAM

- **Enables software written for the MCS-48™ family to be upgraded to run on the 8051**
- **Maps each 8048 instruction to a corresponding 8051 instruction**
- **Preserves comments; translates 8048 macro definitions and calls**
- **Provides diagnostic information and warning messages embedded in the output listing**

The 8048 to 8051 Assembly Language Converter is a utility to help users of the MCS-48 family of microcomputers upgrade their deisgns with the high performance 8051 architecture. By converting 8048 source code to 8051 source code, the software investment developed for the 8048 is maintained when the system is upgraded.

The goal of the converter (CONV51) is to attain functional equivalence with the 8048 code by mapping each 8048 instruction to a corresponding 8051 instruction. In some cases a different instruction is produced because of the enhanced instruction set (e.g., bit CLR instead of ANL).

Although CONV51 tries to attain functional equivalence with each instruction, certain 8048 code sequences cannot be automatically converted. For example, a delay routine which depends on 8048 execution speed would require manual adjustment. A few instructions, in fact, have no 8051 equivalent (such as those involving P4-P7). Finally, there are a few areas of possible intervention such as PSW manipulation and interrupt processing, which at least require the user to confirm proper translation. The converter always warns the user when it cannot guarantee complete conversion.

CONV51 produces two files. The output file contains the ASM51 source program produced from the 8048 instructions. The listing file produces correlated listings of the input and output files, with warning messages in the output file to point out areas that may require users' intervention in the conversion.

SPECIFICATIONS

OPERATING ENVIRONMENT

Required Hardware:

Intellec Microcomputer Development System with

64K Bytes of RAM

Flexible Disk Drive(s)

System Console

—CRT or hard copy device

Optional Hardware:

Universal PROM Programmer
Line Printer
ICE-51 In-Circuit Emulator

Required Software:

ISIS-II Diskette Operating System (V3.4 or later)

Documentation Package:

MCS-51 Macro Assembler User's Guide
MCS-51 Utilities User's Guide for 8080/8085 Based Development System
MCS-51 Macro Assembly Language Pocket Reference
MCS-51 Assembler and Utilities Pocket Reference
MCS-51 8048-to-8051 Assembly Language Converter Operating Instructions for ISIS-II Users

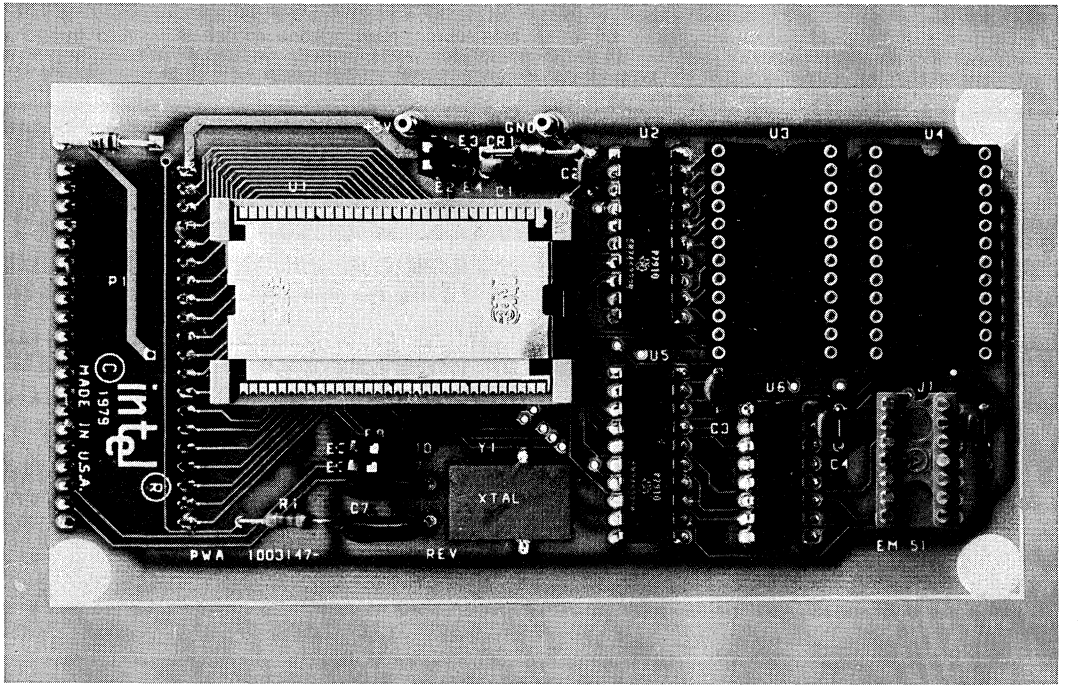
ORDERING INFORMATION

Part Number	Description
MCI-51-ASM	8051 Software Development Package

EM-51
8051 EMULATION BOARD

- **Emulates 8051/8751/8031 functions on a 2.75″ × 5.25″ board assembly**
- **Replaces the 8051/8751/8031 in prototype systems**
- **Plugs directly into 8051/8751/8031 sockets**
- **Includes a 2732A EPROM device for program memory**

The EM-51 emulation board is a small, ready-to-use microcomputer assembly that replaces an Intel 8051 family single-chip microcomputer in a prototype system. EM-51 includes sockets for 2716 or 2732 EPROMs, which substitute for the 8051/8751 on-chip program memory during prototype development. An Intel 2732A 4K x 8 EPROM is included with the board. With the memory in place, an EM-51 board becomes a full functional and electrical equivalent of the 8051 or 8751 microcomputer.



FUNCTIONAL DESCRIPTION

The EM-51 emulator board uses an Intel 8051 family single-chip microcomputer. The microcomputer is configured with additional input and output lines to let its on-chip ROM/EPROM program memory be replaced by EM-51 memory. The 8051's internal address, data, and control lines are connected through buffers to two sockets which accommodate the memory device(s).

Jumpers on the EM-51 board are used to select among memory, clock, and power options.

Memory Options

The EM-51 board uses an Intel 2732A EPROM, which provides 4K bytes of memory to replace the on-chip program memory. The board has two EPROM sockets, permitting you to use 2716 2K × 8 EPROM as an option. Table 1 lists the memory device options by speed and power.

Clock Options

EM-51 operates with either a prototype system crystal or an on-board crystal supplied by the user. For example, the on-board option is helpful if you need to reduce the crystal-to-chip spacing. Intel Application Note AP-35 describes crystal selection criteria.

Power Options

An EM-51 can be powered from the prototype 8051 socket, an external 5-volt power supply, or a combination of the two. The combination option lets your prototype power supply support the 8051 while an external supply powers the additional EM-51 circuitry.

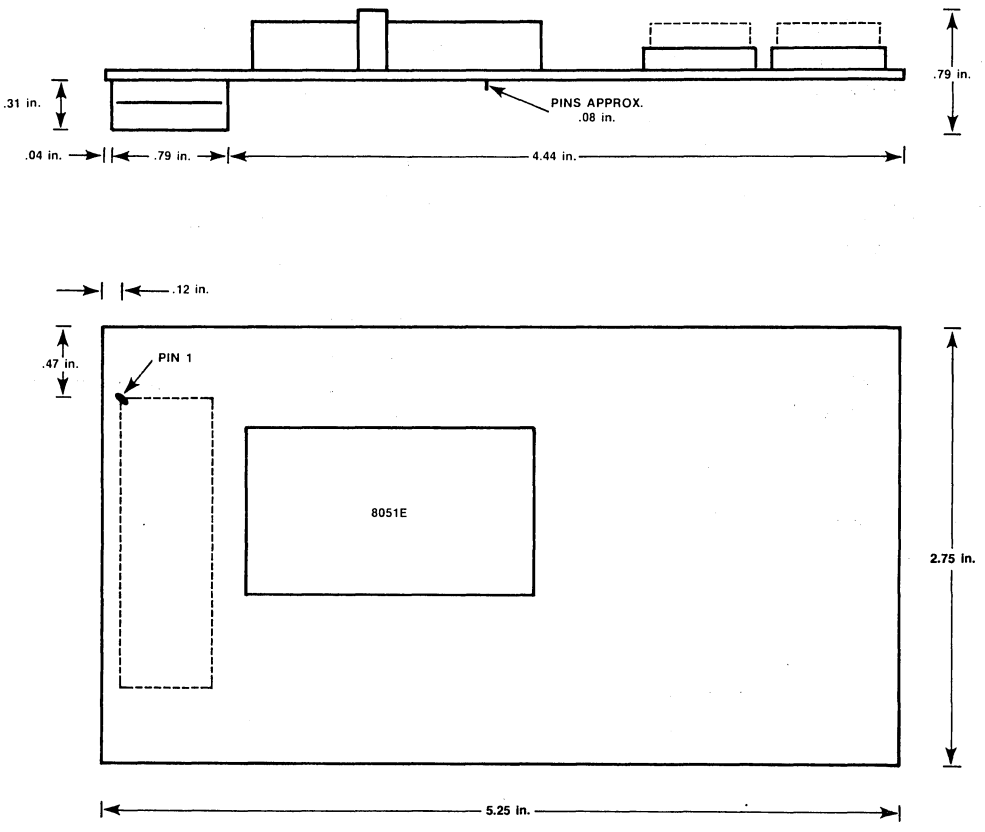


Figure 1. Dimensions

Table 1.

	EM-51 Memory Options					
Memory Device	**2732A-2**	**2732A**	**2732**	**2716-1**	**2716-2**	**2716**
Size per device	4K × 8			2K × 8		
Access time (ns)						
(from address)	200	250	450	350	390	450
(from output enable)	70	100	120	120	120	120
Maximum Operating Frequency (MHz)	12.0	12.0	8.0	9.5	9.0	8.0
Maximum Power Consumption (mA)						
(8051)*	225	225	225	225	225	225
(other circuitry)	500	500	500	550	550	550

* The 8051 configuration used on EM-51 may require more power than standard 8051, 8751, or 8031 devices.

SPECIFICATIONS

Equipment Supplied

EM-51 board assembly
EM-51 Operator's Manual
2732A EPROM

Equipment Required

EPROM programmer
Power supply

Optional Equipment

Crystal for on-board clock

Physical Characteristics

Length: 5.25 in. (13.34 cm)
Width: 2.75 in. (6.99 cm)
Depth: 0.75 in. (1.91 cm)
Weight: 4 oz. (113 gm)

Electrical Characteristics

DC Power: V_{CC} = 5V ±5%
I_{CC} = 775 mA (maximum)

Timing: Same as Intel 8051 microcomputer

Environmental Characteristics

Operating Temperature: 0°C to 55°C
Operating Humidity: 95% (Maximum) relative humidity, non-condensing

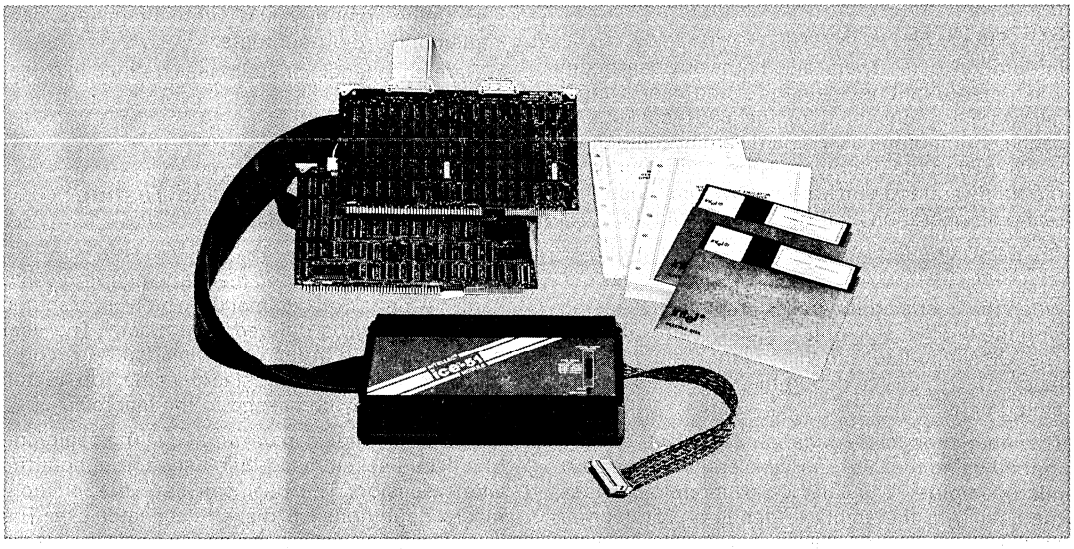
ORDERING INFORMATION

Part Number: MCI-51-EM
Description: 8051 Emulation Board

ICE-51™
8051 IN-CIRCUIT EMULATOR

- **Precise, full-speed, real-time emulation**
  - **— Load, drive, timing characteristics**
  - **— Full-speed program RAM**
  - **— Serial and parallel ports**
- **User-specified breakpoints**
- **Execution trace**
  - **— User-specified qualifier registers**
  - **— Conditional trigger**
  - **— Symbolic groupings and display**
  - **— Instruction and frame modes**
- **Emulation timer**
- **Full symbolic debugging**
- **Single-line assembly and disassembly for program instruction changes**
- **Macro commands and conditional block constructs for automated debugging sessions**
- **HELP facility: ICE-51 command reference at the console**
- **User confidence test of ICE-51 hardware**

The ICE-51 module resides in the Intellec® Microcomputer Development System and interfaces to any user-designed 8051 system through a cable terminating in an 8051 emulator microprocessor and a pin-compatible plug. The emulator processor, together with 8K bytes of user program RAM located in the ICE-51 buffer box, replaces the 8051 device in the user system while maintaining the 8051 electrical and timing characteristics. Powerful Intellec debugging functions are thus extended into the user system. Using the ICE-51 module, the designer can emulate the system's 8051 in real-time or single-step mode. Breakpoints allow the user to stop emulation on user-specified conditions, and a trace qualifier feature allows the conditional collection of 1000 frames of trace data. Using the single-line 8051 assembler the user may alter program memory using ASM51 mnemonics and symbolic references, without leaving the emulator environment. Frequently used command sequences can be combined into compound commands and identified as macros with user-defined names.



The following are trademarks of Intel Corporation and may be used only to describe Intel products: Intel, Intellec, MCS and ICE, and the combination of MCS or ICE and a numerical suffix. 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 CORPORATION, 1981

August 1981
162886

FUNCTIONAL DESCRIPTION

Integrated Hardware and Software Development

The ICE-51 emulator allows hardware and software development to proceed interactively. This approach is more effective than the traditional method of independent hardware and software development followed by system integration. With the ICE-51 module, prototype hardware can be added to the system as it is designed. Software and hardware integration occurs while the product is being developed.

The ICE-51 emulator assists four stages of development:

SOFTWARE DEBUGGING

It can be operated without being connected to the user's system before any of the user's hardware is available. In this stage ICE-51 debugging capabilities can be used in conjunction with the Intellec text editor and 8051 macroassembler to facilitate program development.

HARDWARE DEVELOPMENT

The ICE-51 module's precise emulation characteristics and full-speed program RAM make it a valuable tool for debugging hardware, including time-critical serial port, parallel port, and timer interfaces.

SYSTEM INTEGRATION

Integration of software and hardware can begin when any functional element of the user system hardware is connected to the 8051 socket. As each section of the user's hardware is completed, it is added to the prototype. Thus, each section of the hardware and software is "system" tested in real-time operation as it becomes available.

SYSTEM TEST

When the user's prototype is complete, it is tested with the final version of the user system software. The ICE-51 module is then used for real-time emulation of the 8051 to debug the system as a completed unit.

The final product verification test may be performed using the 8751 EPROM version of the 8051 microcomputer. Thus, the ICE-51 module provides the user with the ability to debug a prototype or production system at any stage in its development without introducing extraneous hardware or software test tools.

Symbolic Debugging

The ICE-51 emulator permits the user to define and use symbolic, rather than absolute, references to program and data memory addresses; additional symbols are predefined by the ICE-51 software for referencing registers, flags, and input/output ports. Thus, the user need not recall or look up the addresses of key locations in his program as they change with each assembly, or become involved with machine code.

When a symbol is used for memory reference in an ICE-51 emulator command, the emulator supplies the corresponding location as stored in the ICE-51 emulator symbol table. This table can be loaded with the symbol table produced by the assembler during application program assembly. The user can obtain the symbol table during software preparation simply by using the "DEBUG" switch in the ASM51 macroassembler. Furthermore, the user can interactively modify the emulator symbol table by adding new symbols or changing or deleting old ones. This feature provides great flexibility in debugging and minimizes the need to work with absolute memory values.

Through symbolic references in combination with other features of the emulator, the user can easily:

- Interpret the results of emulation activity collected during trace.
- Disassemble program memory to mnemonics, or assemble mnemonic instructions to executable code.
- Examine or modify 8051 internal registers, data memory, or port contents.
- Reference labels or addresses defined in a user program.

Automated Debugging and Testing

MACRO COMMAND

A macro is a set of commands which is given a name. A group of commands which is executed frequently can be defined as a macro. The user can execute the group of commands by typing a colon followed by the macro name. Up to ten parameters may be passed to the macro.

Macro commands can be defined at the beginning of a debug session and then used throughout the whole session. The user can save one or more macro definitions on diskette for later use. The Intellec text editor may be used to edit the macro file. The macro definitions are easy to include in any later emulation session.

The power of the development system can be applied to manufacturing testing as well as development by writing test sequences as macros. The macros are stored on diskettes for use during system test.

COMPOUND COMMAND

Compound commands provide conditional execution of commands (IF command) and execution of commands repeatedly until certain conditions are met (COUNT, REPEAT commands).

Compound commands may be nested any number of times, and may be used in macro commands.

Example:

```
*DEFINE .I = 0          ; Define symbol .I to 0
*COUNT 100H            ; Repeat the following
                         commands 100H times.
.*IF .I AND 1 THEN     ; Check if .I is odd
..*CBYTE .I = .I       ; Fill the memory at location .I
                         to value .I
..*END                 ; End if structure
.*.I = .I + 1          ; Increment .I by 1.
.*END                  ; Command executes upon
                         carriage-return after END
```

(The characters *, .*, and ..* shown in this example are system prompts which include an indication of the nesting level of compound commands.)

Operating Modes

The ICE-51 software is an Intellec RAM-based program that provides the user with easy-to-use commands for initiating emulation, defining breakpoints, controlling trace data collection, and displaying and controlling system parameters. ICE-51 commands are configured with a broad range of modifiers which provide the user with maximum flexibility in describing the operation to be performed.

EMULATION

The ICE-51 module can emulate the operation of a prototype 8051 system, at real-time speed (1.2 to 12 MHz) or in single steps. Emulation commands to the ICE-51 module control the process of setting up, running, and halting an emulation of the user's 8051-based system. Breakpoints and tracepoints enable the ICE-51 emulator to halt emulation and provide a detailed trace of execution in any part of the user's program. A summary of the emulation commands is shown in Table 1.

Breakpoints

The ICE-51 hardware includes two breakpoint registers that allow the user to halt emulation when specified conditions are met. The emulator continuously compares the values stored in the breakpoint registers with the status of specified address, opcode, operand, or port values, and halts emulation when this comparison is satisfied. When an instruction initiates a break, that instruction is executed completely before the break takes place. The ICE-51 emulator then regains control of the console and enters the Interrogation Mode. With the breakpoint feature, the user can request an emulation break when his program:

- Executes an instruction at a specific address or within a range of addresses.
- Executes a particular opcode.
- Receives a specific signal on a port pin.
- Fetches a particular operand from the user program memory.
- Fetches an operand from a specific address in program memory.

Table 1. Major Emulation Commands

Command	Description
GO	Begins real-time emulation and optionally specifies break conditions.
BR0, BR1, BR	Sets or displays either or both Breakpoint Registers used for stopping real-time emulation.
STEP	Performs single-step emulation.
QR0, QR1, QR	Sets or displays match conditions for qualified trace.
TR	Specifies or displays trace-data collection conditions and optionally sets Qualifier Register (QR0, QR1).
SY, SY1, SY0	Set and display status of synchronization line outputs or latched inputs. Used to allow real-time emulation or trace to start and stop synchronously with external events.

Trace and Tracepoints

Tracing is used with both real-time and single-step emulation to record diagnostic information in the trace buffer as a program is executed. The information collected includes opcodes executed, port values, and memory addresses. The ICE-51 emulator collects up to 1000 frames of trace data.

This information can be displayed as assembler instruction mnemonics, if desired, for analysis during interrogation or single-step mode. The trace-collection facility may be set to run conditionally or unconditionally. Two unique trace qualifier registers, specified in the same way as break-

point registers, govern conditional trace activity. The qualifiers can be used to condition trace data collection to take place as follows:

- Under all conditions (forever).
- Only while the trace qualifier is satisfied.
- For the frames or instructions preceding the time when a trace qualifier is first satisfied (pre-trigger trace).
- For the frames or instructions after a trace qualifier is first satisfied (post-triggered trace).

Table 2 shows an example of a trace display.

Table 2. Trace Display (Instruction Mode)

```
*T ALL
FRAME  LOC    OBJ    INSTRUCTION                  P1   P2   P3   TOVF
0000:  0000H  E0     MOVX   @DPTR,A               00H  00H  FFH   0
0001:  0001H  FF     MOV    R7,A                   00H  00H  FFH   0
0011:  0002H  F4     CPL    A                      00H  00H  FFH   0
0015:  0003H  22     RET    A                      00H  00H  FFH   0
0019:  0004H  75C0AA MOV    .P1,#AAH               00H  00H  FFH   0
0027:  0007H  C3     CLR    C                      00H  00H  FFH   0
0031:  0008H  98     SUBB   A,R0                   AAH  00H  FFH   0
0035:  0009H  0000   NOP    0000                   AAH  00H  FFH   0
0043:  000AH  75A0FF MOV    0A0H,#.P2              AAH  00H  FFH   0
0051:  000DH  0156   AJMP   0000H                 AAH  00H  FFH   0
```

INTERROGATION AND UTILITY

Interrogation and utility commands give the user convenient access to detailed information about the user program and the state of the 8051 that is useful in debugging hardware and software. Changes can be made in memory and in the 8051 registers, flags, and port values. Commands are also provided for various utility operations such as loading and saving program files, defining symbols, displaying trace data, controlling system synchronization and returning control to ISIS-II. A summary of the basic interrogation and utility commands is shown in Table 3. Two time-saving emulator features are discussed below.

SINGLE-LINE ASSEMBLER/DISASSEMBLER — The single-line assembler/disassembler (ASM and DASM commands) permits the designer to examine and alter program memory using assembly language mnemonics, without leaving the emulator environment or requiring time-consuming program reassembly. When assembling new mnemonic instructions into program memory, previously defined symbolic references (from the original program assembly, or subsequently defined during the emulation session) may be used in the instruction operand field. The emulator will supply the absolute address or data values as stored in the emulator symbol table. These features eliminate user time spent translating to and from machine code and searching for absolute addresses, with a corresponding reduction in transcription errors.

Table 3. Major Interrogation and Utility Commands

Command	Description
HELP	Displays help messages for ICE-51 emulator command-entry assistance.
LOAD	Loads user object program (8051 code) into user program memory, and user symbols into ICE-51 emulator symbol table.
SAVE	Saves ICE-51 emulator symbol table and/or user object program in ISIS-II hexadecimal file.
LIST	Copies all emulator console input and output to ISIS-II file.
EXIT	Terminates ICE-51 emulator operation.
DEFINE	Defines ICE-51 emulator symbol or macro.
REMOVE	Removes ICE-51 emulator symbol or macro.
ASM	Assembles mnemonic instructions into user program memory.
DASM	Disassembles and displays user program memory contents.
Change/Display Commands	Change or display value of symbolic reference in ICE-51 emulator symbol table, contents of key-word references (including registers, I/O ports, and status flags), or memory references.
EVALUATE	Evaluates expression and displays resulting value.
MACRO	Displays ICE-51 macro or macros.
INTERRUPT	Displays serial, external, or timer interrupt register settings.
SECONDS	Displays contents of emulation timer, in microseconds.
Trace Commands	Position trace buffer pointer and select format for trace display.
PRINT	Displays trace data pointed to by trace buffer pointer.

HELP — The HELP file allows the user to display ICE-51 command syntax information at the Intellec console. By typing "HELP", a listing of all items for which help messages are available is displayed; typing "HELP <Item>" then displays relevant information about the item requested, including typical usage examples. Table 4 shows some sample HELP messages.

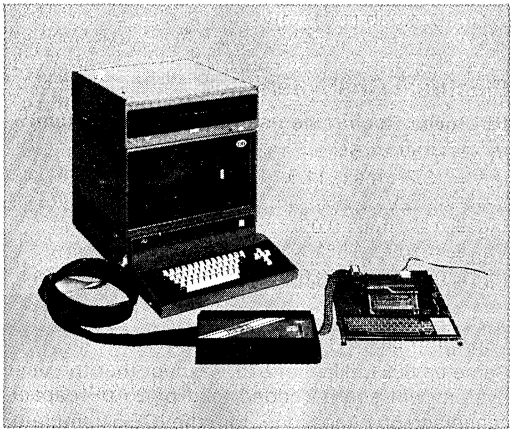


Figure 1. A Typical 8051 Development Configuration. The host system is an Intellec Model 225, plus 1 megabyte dual double-density flexible disk storage. The ICE-51 module is connected to an SDK-51 system design kit.

Emulation Accuracy

The speed and interface demands of a high-performance single-chip microcomputer require extremely accurate emulation, including full-speed, real-time operation with the full function of the microcomputer. The ICE-51 emulator achieves accurate emulation with an 8051 bond-out chip, a special configuration of the 8051 microcomputer family, as its emulation processor.

Each of the 40 pins on the user plug is connected directly to the corresponding 8051 pin on the bond-out chip. Thus the user system sees the emulator as an 8051 microcomputer at the 8051 socket. The resulting characteristics provide extremely accurate emulation of the 8051, including speed, timing characteristics, load and drive values, and crystal operation. The emulator may draw more power from the user system than a standard 8051 family device.

Additional bond-out pins provide signals such as internal address, data, clock, and control lines to the emulator buffer box. These signals let static RAM in the buffer box substitute for on-chip program ROM or EPROM or external program memory. The 8K bytes of full-speed RAM in the buffer box can be mapped in 4K blocks to anywhere within the 64K program memory space of the 8051. The bond-out chip also gives the emulator "back-door" access to internal chip operation, so that the emulator can break and trace execution without interfering with the values on the user-system pins.

Table 4. HELP Command

```
*HELP
Help is available for the following items. Type HELP followed by
the item name. The help items cannot be abbreviated. (For more
information, type HELP HELP or HELP INFO.)
Emulation:    Trace Collection:  Misc:       <address>
GO GR  SYP    TR GP GRP GPI SYI  BASE        <CPU$keyword>
BR BRP BRI                       DISABLE     <expr>
STEP          Trace Display:     ENABLE      <ICE51$keyword>
              TRACE MOVE PRINT   ERROR       <identifier>
              OLDEST NEWEST      EVALUATE    <instruction>
                                 HELP        <masked$constant>
Change/Display/Define/Remove:    INFO        <match$cond>
<CHANGE>  REMOVE CBYTE WBIT      <LIGHTS>    <numeric$constant>
<DISPLAY> SYMBOL IBYTE IASM      LIST        <partition>
REGISTER  RESET  PBYTE ASM       LOAD        <string>
SECONDS   WRITE  RBYTE MAP       SAVE        <string$constant>
DEFINE    STACK  XBYTE SY        SUFFIX      <symbolic$ref>
                                 SYMBOLIC    <system$symbols>
Macro:                Compound               <trace$reference>
DEFINE  DIR           Commands:              <unlimited$match$cond>
DISABLE ENABLE        COUNT                  <users$symbols>
INCLUDE PUT           IF
<MACRO$DISPLAY>       REPEAT
<MACRO$INVOCATION>
*
```

```
*HELP IF
IF - The conditional command allows conditional execution of one
or more commands based on the values of boolean conditions.
     IF <expr> [THEN] <cr>          <trueflist>::=[<command> <cr>]F
      <trueflist>                   <falseflist>::=[<command> <cr>]F
      [OBIF <expr> <cr>             <command>::=An ICE-51 command.
      <trueflist>]F
      [ELSE <cr>
      <falseflist>]
      END
The <expr>s are evaluated in order as 16-bit unsigned integers.
If one is reached whose value has low-order bit 1 (TRUE), all
commands in the <trueflist> following that <expr> are then
executed and all commands in the other <trueflist>s and in the
<falseflist> are skipped.  If all <expr>s have value with low-
order bit 0 (FALSE), then all commands in all <trueflist>s are
skipped and, if ELSE is present, all commands in the <falseflist>
are executed.
     (EX:  IF .LOOP=5 THEN
               STEP
               ELSE
               GO
               END)
```

SPECIFICATIONS

ICE-51 Operating Requirements

Intellec® Microcomputer Development System (64K RAM required)

System console

Intellec® Diskette Operating System (single or double density) ISIS-II v. 3.4 or later

Equipment Supplied

- Printed circuit boards (2)
- Emulation buffer box, Intellec interface cables, and user-interface cable with 8051 emulation processor
- Crystal power accessory
- Operating instructions manual
- Diskette-based ICE-51 software (single and double density)

Emulation Clock

User's system clock (1.2 to 12 MHz) or ICE-51 crystal power accessory (12 MHz)

Environmental Characteristics

Operating Temperature: 0° to 40°C
Operating Humidity: Up to 95% relative humidity without condensation.

Physical Characteristics

Printed Circuit Boards
Width: 12.00 in. (30.48 cm)
Height: 6.75 in. (17.15 cm)
Depth: 0.50 in. (1.27 cm)

Buffer Box
Width: 8.00 in. (20.32 cm)
Length: 12.00 in. (30.48 cm)
Depth: 1.75 in. (4.44 cm)
Weight: 4.0 lb (1.81 kg)

Electrical Characteristics

DC Power Requirements (from Intellec system)

$V_{CC} = +5V, +5\%, -1\%$
$I_{CC} = 13.2A$ max; 11.0A typical
$V_{DD} = +12V, \pm 5\%$
$I_{DD} = 0.1A$ max; 0.05A typical
$V_{BB} = -10V, \pm 5\%$
$I_{BB} = 0.05A$ max; 0.01A typical

User plug characteristics at 8051 socket

Same as 8031, 8051, or 8751, except that the user system will see an added load of 25 pF capacitance and 50 μA leakage from the ICE-51 emulator user plug at ports 0, 1, and 2.

ORDERING INFORMATION

Part Number	Description
MCI-51-ICE	8051 Microcontroller In-Circuit Emulator, cable assembly and interactive diskette software

UPI–41A™ Development Systems

20



ICE-41A™
UPI-41A IN-CIRCUIT EMULATOR

Extends Intellec microcomputer development system debug power to user configured system via external cable and 40-pin plug, replacing user UPI-41A™ devices

Emulates user system UPI-41A™ devices in real time

Allows user configured system to use static RAM memory for program debug

Provides hardware comparators for user designated break conditions

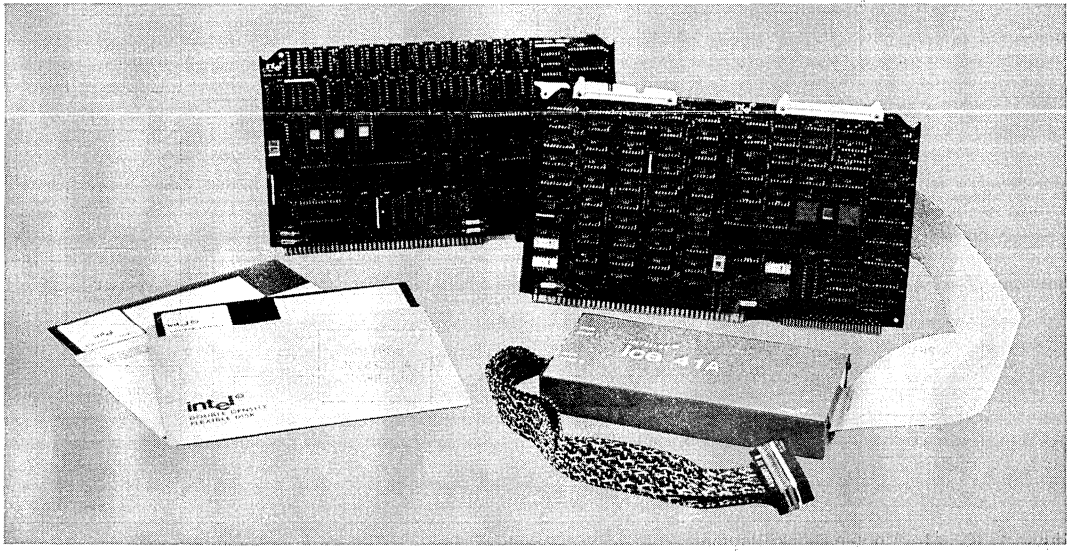
Eliminates need for extraneous debugging tools residing in user system

Collects address, data, and UPI-41A™ status information on machine cycles emulated

Provides capability to examine and alter UPI-41A™ registers, memory, and flag values, and to examine pin and port values

Integrates hardware and software efforts early in engineering cycle to save development time

The ICE-41A UPI-41A In-Circuit Emulator module is an Intellec system resident module that interfaces to any user configured UPI-41A system. The ICE-41A module interfaces with a UPI-41A pin-compatible plug which replaces the UPI-41A device in the system. With the ICE-41A plug in place, the designer has the capability to execute the system in real time while collecting up to 255 instruction cycles of real time trace data. In addition, he can single step the system program during execution. Static RAM memory is available through the ICE-41A module to store UPI-41A programs. The designer may display and alter the contents of program memory, internal UPI-41A registers and flags, and I/O ports. Powerful debug capability is extended into the UPI-41A system while ICE-41A debug hardware and software remain inside the Intellec system. Symbolic reference capability allows the designer to use symbols rather than absolute values when examining and modifying memory, registers, flags, and I/O ports in the system.



FUNCTIONAL DESCRIPTION

Debug Capability Inside User System

Intellec memory is used for the execution of the ICE-41A software. The Intellec CRT console and the file handling capabilities provide the designer with the ability to communicate with the ICE-41A module and display information on the operation of the prototype system. The ICE-41A module block diagram is shown in Figure 1.

Symbolic Debugging

Symbol Table — ICE-41A software allows the user to make symbolic references to I/O ports, memory addresses, and data in his program. The user symbol table which is generated along with the object file during a program assembly can be loaded to Intellec memory for access during emulation. The user may add to this symbol table any additional symbolic values for memory addresses, constants, or variables that he may find useful during system debugging. By referring to symbol memory addresses, the user can examine, change or break at the intended location. In addition, ICE-41A provides symbolic definition of all UPI-41A registers and flags.

Symbolic Reference — Symbolic reference is a great advantage to the system designer. He is no longer burdened with the need to recall or look up addresses of key locations in his program which can change with each assembly. Meaningful symbols from his source program can be used instead. For example, the command:

```
GO FROM .START TILL CODE. RSLT
```

begins execution of the program at the address referenced by the label START in the designer's assembly program. A breakpoint is set to occur the first time the microprocessor executes the program memory location referenced by RSLT. The designer does not have to be concerned with the physical locations of START and RSLT. The ICE-41A software driver supplies them automatically from information stored in the symbol table.

Memory Replacement

The 8741/8741A and 8041/8041A contain internal program and data memory. When the UPI-41A microcomputer is replaced by the ICE-41A socket in a system, the ICE-41A module supplies static RAM memory as a replacement for the internal microcomputer memory. The ICE-41A module has enough RAM memory available to emulate up to the total 1K control memory capability of the system.

Real-Time Trace

The ICE-41A module captures trace information while the designer is executing programs in real time. The instructions executed, program counter, port values for port 1 and port 2, and the values of selected UPI-41A status lines are stored for the last 255 instruction cycles executed. When retrieved for display, code is disassembled for user convenience. This provides data for determining how the user system was reacting prior to emulating break.

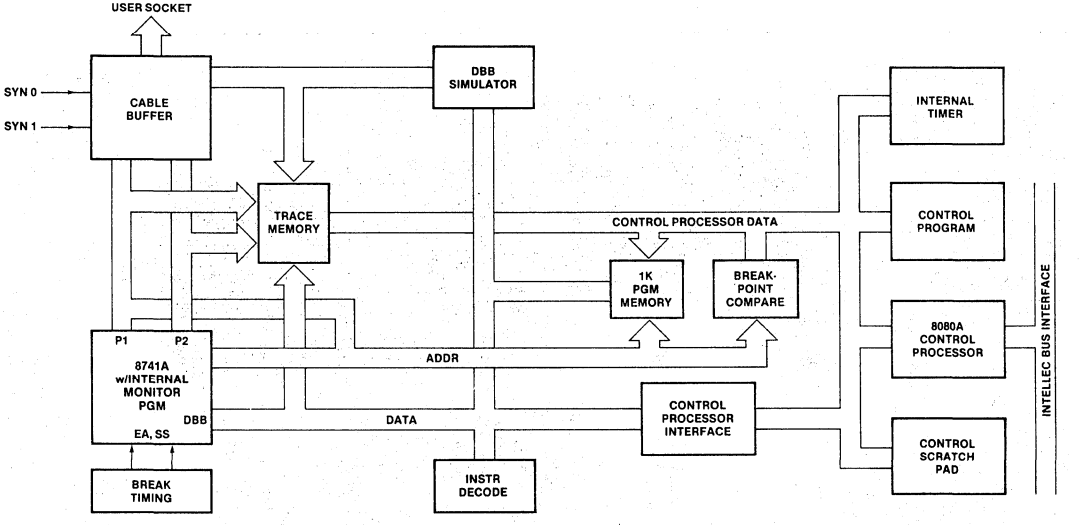


Figure 1. ICE-41A Module Block Diagram

Integrated Hardware/Software Development

The user prototype systems need no more than a UPI-41A socket and timing logic to begin integration of software and hardware development efforts. Through the ICE-41A module, Intellec system resources can be accessed to replace the prototype system. UPI-41A software development can proceed without the prototype hardware. Hardware designs can be tested using previously tested system software.

Hardware

The ICE-41A module is a microcomputer system utilizing Intel's UPI-41A microprocessor as its nucleus. This system communicates with the Intellec system 8080A processor via direct memory access. Host processor commands and ICE-41A status are interchanged through a DMA channel. ICE-41A hardware consists of two printed circuit boards, the controller board and the emulator board, which reside in the Intellec system chassis. A cable assembly interfaces the ICE-41A module to the user's UPI-41A system. The cable terminates in a UPI-41A pin-compatible plug which replaces any UPI-41A device in the user system.

Controller Board

The ICE-41A module interfaces to the Intellec systems as a peripheral device. The controller board receives commands from the Intellec system and responds through a DMA port. Three 10-bit hardware breakpoint registers are available which can be loaded by the user. While in emulation mode, a hardware comparator is constantly monitoring address lines for a match which will terminate an emulation. The controller board returns real-time trace data, UPI-41A registers, flag and port values, and status information to a control block in the Intellec system when emulation is terminated. This information is available to the user through the ICE-41A interrogation commands. Error conditions, when detected, are automatically displayed on the Intellec system console.

Emulator Board

The emulator board contains the 8741A and peripheral logic required to emulate the UPI-41A device in the user system. A 6 MHz clock drives the emulated UPI-41A device. This clock can be replaced with a user supplied TTL clock in the user system or can be strapped internally for 3 MHz operation.

Cable Card

The cable card is included for cable driving. It transmits address and data bus information to the user system through a 40-pin connector which plugs into the user system in the socket designed for the UPI-41A device.

Software

The ICE-41A software driver is a RAM-based program which provides the user with command language (see Table 1, Table 2, and Table 3) for defining breakpoints, initiating real-time emulation or single step operation, and interrogation and altering user system status recorded during emulation. The ICE-41A command language contains a broad range of modifiers which provide the user with maximum flexibility in defining the operation to be performed. The ICE-41A software driver is available on diskette and operates in 32K of Intellec RAM memory.

Command	Operation
Enable	Activates breakpoint and display registers for use with go and step commands.
Go	Initiates real-time emulation and allows user to specify breakpoints and data retrieval.
Step	Initiates emulation in single instruction increments. Each step is followed by register dump. User may optionally tailor other diagnostic activity to his needs.
Interrupt	Emulates user system interrupt

Table 1. ICE-41A Emulation Commands

Command	Operation
Display	Prints contents of memory, UPI-41A device registers, I/O ports, flags, pins, real-time trace data, symbol table, or other diagnostic data on list device.
Change	Alters contents of memory, register, output port, or flag. Sets or alters breakpoints and display registers.
Base	Establishes mode of display for output data.
Suffix	Establishes mode of display for input data.

Table 2. ICE-41A Interrogation Commands

Command	Operation
Load	Fetches user symbol table and object code from input device.
Save	Sends user symbol table and object code to output device.
Define	Enters symbol name and value to user symbol table.
Move	Moves block of memory data to another area of memory.
Print	Prints user specified portion of trace memory to selected list device.
List	Defines list device.
Exit	Returns program control to ISIS-II.
Evaluate	Converts expression to equivalent values in binary, octal, decimal, and hex.
Remove	Deletes symbols from symbol table.
Reset	Reinitializes ICE-41A hardware.

Table 3. ICE-41A Utility Commands

SPECIFICATIONS

ICE-41A Operating Environment

Required Hardware
Intellec microcomputer development system
System console
Intellec diskette operating system
ICE-41A module

Required Software
System monitor
ISIS-II
ICE-41A diskette-based software

System Clock

Crystal controlled 6.0 MHz or 3.0 MHz internal or user supplied TTL external

Physical Characteristics

Printed Circuit Boards

Width: 12.00 in. (30.48 cm)
Height: 6.75 in. (17.15 cm)
Depth: 0.50 in. (1.27 cm)
Weight: 8.00 lb (3.64 kg)

Cable Buffer Box

Width: 8.00 in. (20.32 cm)
Height: 4.00 in. (10.16 cm)
Depth: 1.25 in. (3.17 cm)
Flat Cable: 4.00 ft (121.92 cm)
User Cable: 15.00 in. (38.10 cm)

Electrical Characteristics

DC Power Requirements

$V_{CC} = +5V, \pm 5\%$
$I_{CC} = 10A$ max; 8A typ
$V_{DD} = +12V, \pm 5\%$
$I_{DD} = 100$ mA max; 60 mA typ
$V_{BB} = -10V$
$I_{BB} = 30$ mA

Input Impedance

@ ICE-41A user socket pins:
$V_{IL} = 0.8V$ max; $I_{IL} = 1.6$ mA
$V_{IH} = 2.0V$ min; $I_{IH} = 40\ \mu A$
@ Bus:
$V_{IL} = 0.8V$ max; $I_{IL} = 250\ \mu A$
$V_{IH} = 2.0V$ min; $V_{IH} = 20\ \mu A$

Output Impedance

@ P1, P2:
$V_{OL} = 0.5V$ max; $I_{OL} = 16$ mA
$V_{OH} = V_{CC}$ (10K pullup)
@ Bus:
$V_{OL} = 0.5V$ max; $I_{OL} = 25$ mA
$V_{OH} = 3.65V$ min; $I_{OH} = 1$ mA

Others

$V_{OL} = 0.5V$ max; $I_{OL} = 16$ mA
$V_{OH} = 2.4V$ max; $I_{OH} = 400\ \mu A$

Equipment Supplied

Controller board
Emulator board
Interface cables and buffer module
Operator's manual
ICE-41A diskette based software

Reference Manuals

9800465 — ICE-41A Operator's Manual (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.

ORDERING INFORMATION

Part Number	Description
MDS-41A-ICE	UPI-41A (8741, 8041, 8741A, 8041A) CPU In-circuit emulator, cable assembly and interactive diskette software included

2920 Signal Processor Development Systems

21

2920 SIGNAL PROCESSING APPLICATIONS COMPILER

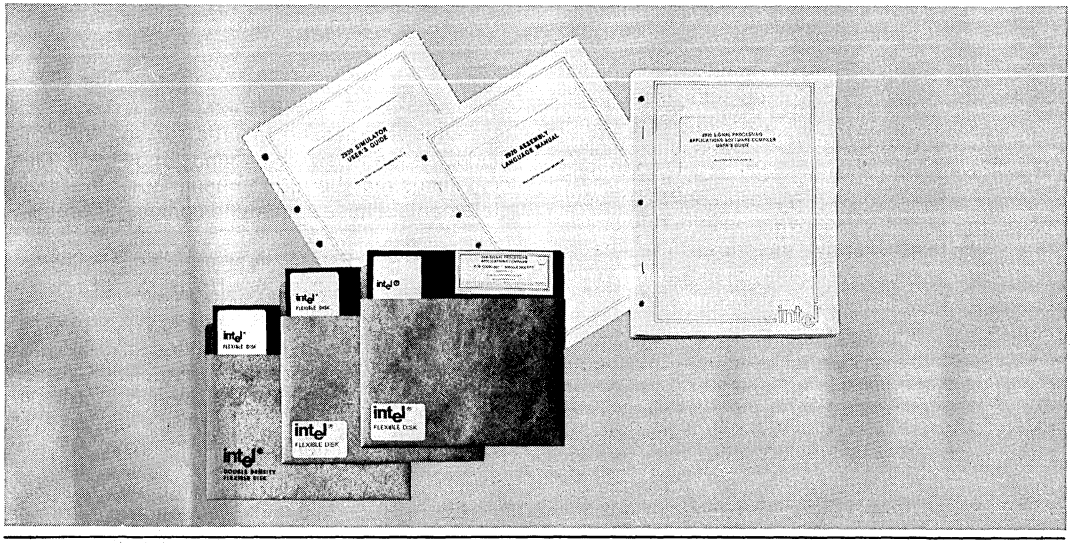
- **Compiler generates 2920 Assembly Language Code**
- **Extensive command set for designing electrical filters**
- **Graphics capability enhances analysis of filter response**
- **Powerful MACRO capability for executing frequently used routines**
- **Interactive software support tool for 2920 Signal Processor**
- **Extends Intellec® Microcomputer Development System support of the 2920**
- **Contains MACRO library for several standard filters and signal processing functions**

The 2920 Signal Processing Applications Compiler (SPAS20) is an interactive tool for designing software to execute on the 2920 Signal Processor.

The SPAS20 package can be visualized as being comprised of three inter-related sections: A compiler section, a filter design section, and a MACRO section.

Among the capabilities of SPAS20 are: ability to generate 2920 assembly language code directly from specifications of signal processing building blocks such as filters and waveform genrators; ability to generate 2920 assembly language code for several classes of algebraic equations such as Y=C * X, Y=C * Y, and Y=C * X + Y where X,Y are variables and C is a constant; ability to examine time and frequency responses of filter sections specified by continuous or sampled poles and zeroes; ability for users to implement more complex commands by grouping sets of commonly used commands into a MACRO.

The SPAS20 package runs under ISIS-II on any Intellec Microcomputer Development System with 64K RAM. The output of SPAS20 can be assembled with the 2920 assembler, tested with the 2920 Simulator, and programmed into the 2920 chip with the Universal PROM Programmer for prototyping.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, Intellec, Multibus, i, iSBC, Multimodule, ICE, iSBX, PROMPT, iCS, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Micromap, μScope and the combination of ICE, iCS, iSBC, iSBX, MCS, or RMX and a numerical suffix.
© Intel Corporation 1980

121563-001 Rev. A

FUNCTIONAL DESCRIPTION

The 2920 Signal Processing Applications Compiler gives the analog designer a "high level language" for his 2920 applications—it decreases the need to code 2920 assembly language. Furthermore, the compiler is interactive. This feature enables the designer to define a filter, graph its response, and change its parameters many times, without having to program and test the filter in an actual 2920 implementation. The command language is very similar to that of Intel's In-Circuit Emulators.

Once a filter is realized by moving poles and zeros in the continuous and sampled planes, the filter may be coded and written onto an ISIS file. Several other file commands are available to store and retrieve command sequences for SPAS20 sessions.

SPAS20 Command Language

DEFINE	This command defines a pole or zero by associating it with a number (i.e., POLE 3), and with real and imaginary coordinates in the continuous or sampled plane. This command also defines a symbol by associating a name with a numeric value, or a MACRO by providing a pointer to a specified command sequence.
GRAPH/ OGRAPH	This command graphically displays the values of object(s) specified. For example, GRAPH GAIN and GRAPH PHASE are used to display filter response. The OGRAPH command will "overgraph" the new response over the old response, after any changes have been made. (You may also graph Group Delay, Step, and Impulse.)
MOVE	Allows the definition of a pole or zero to be changed—its coordinates, its plane, or both.
REMOVE	Deletes the definition of a pole, zero, symbol, or macro.
HELP	Types an explanatory message on the console, pertaining to a command or its attributes.
HOLD	Command to correct attenuation due to sample-and-hold distortion: if ON, it corrects absolute gain by sin(x)/x and phase by adding x, where x=TS*FREQ*π. It corrects group delay by subtracting π*TS.
EVALUATE	Gives the decimal numeric value of any expression.
CODE	Creates 2920 assembly language code for given poles, zeros, and equations.

The SPAS20 compiler also recognizes the following commands for file handling:

PUT/ APPEND	Writes out objects (commands) to a specified file, either creating a new one or appending an existing one. This enables the user to store all or part of a SPAC20 session on a diskette to be brought back later with the INCLUDE command.
DISPLAY	Copies the contents of a file to the console.
INCLUDE	Executes a sequence of instructions from a diskette file as if they were typed in from the console.
LIST	Creates a file containing all console interactions.

In addition to naming macros for specific command sequences, compound and conditional commands may be formed using all of the above statements. These compound commands are:

IF	Establishes conditional flow of control within a block of commands.
REPEAT	Used for repetition of a block of commands; executes indefinitely or until a condition is met (using WHILE, UNTIL, and END statements).
COUNT	Establishes the number of times a command sequence is to be executed, in a looping fashion.

SPAS20 MACRO Facility

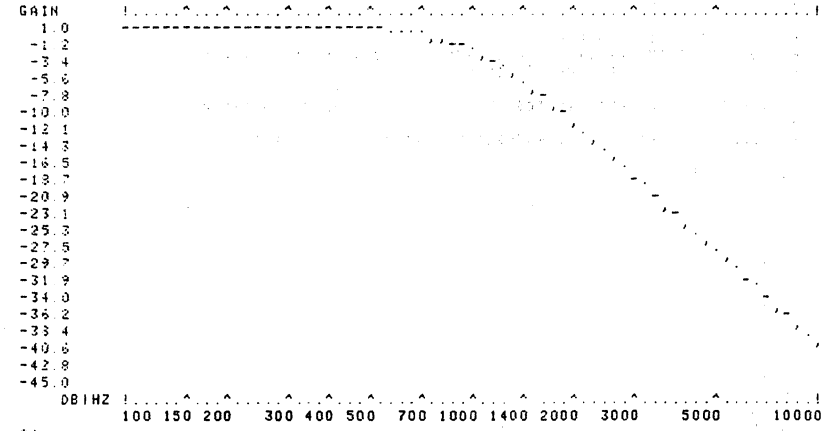
A macro is a sequence of commands that is stored on a temporary diskette file. The command sequence is executed when the macro name is entered as a command. This saves repetitive entry of the sequence, and permits alogorithms to be saved on diskette for future use. This SPAS20 facility allows you to do the following:

- Display the text of any macro.
- Define a macro, specifying its name and any parameters that are to be used by the block. This definition is followed by the contents of the macro (commands) and the EM statement to end its definition.
- Invoke a macro by entering its name and appropriate values for any parameters.
- List the names of all defined macros.
- Remove any or all macros.

SAMPLE SPAS20 SESSION

```
-:F1:SPAC20.SFT

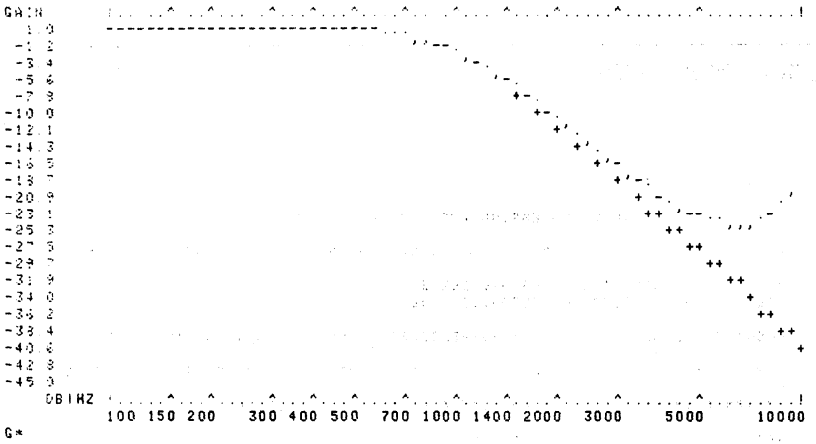
ISIS-II 2920 SIGNAL PROCESSING APPLICATIONS COMPILER, V1.0
*
*DEFINE POLE 1 = -707,707        ; CREATE A POLE IN CONTINUOUS S-PLANE
*
*PZ        ; LIST ALL POLES AND ZEROS
POLE 1 = -707.00000,707.00000,CONTINUOUS
*
*FSCALE = 100,10000        ; ESTABLISHES FREQUENCY RANGE OF INTEREST
*
*YSCALE = -45,1        ; ESTABLISHES MAGNITUDE RESPONSE RANGE OF INTEREST
*
*GRAPH GAIN        ; PLOT MAGNITUDE RESPONSE OF POLE PAIR
```



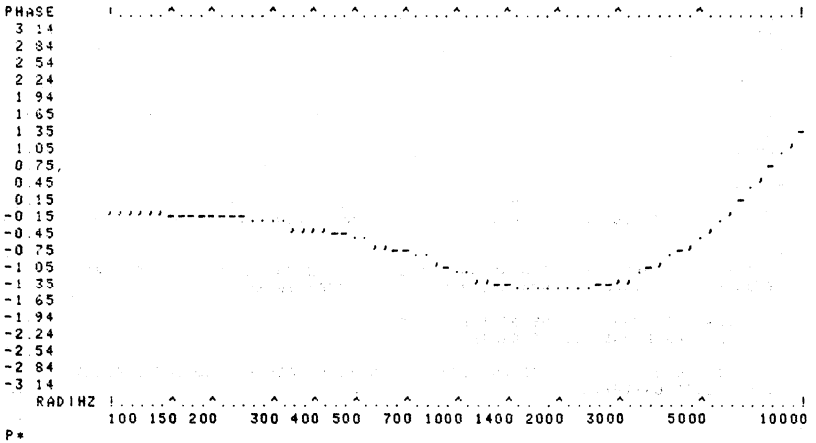

```
**
*
*; THE UNITS USED IN GRAPHING GAIN ARE SHOWN IN THE LOWER LEFT CORNER.
*; GAIN IN DECIBELS IS GRAPHED VERSES FREQUENCY IN HERTZ.
*
*; PREPARE TO MOVE TO THE DIGITAL DOMAIN.
*; SAMPLE RATE MUST BE SPECIFIED.
*
*TS = 1/13020        ; RATE FOR 192 INSTRUCTION PROGRAM AND 10MHZ CLOCK
TS = 7.6805004/10**5
*
```

SAMPLE SPAS20 SESSION (Cont'd.)

```
*MOVE POLE TO Z        ; CONVERT FILTER TO DIGITAL VIA MATCHED-Z TRANSFORMATION
1 POLES/ZEROES MOVED
*
*P         ; LIST TRANSFORMED POLE
POLE 1 = 0.71092836,0.34118369,Z
*
*; COMPARE RESPONSES OF THE ANALOG AND DIGITAL FILTERS BY GRAPHING THE
*; NEW RESPONSE OVER THE OLD
*
*OGRAPH GAIN
```




```
G*
*
*; PLUS SIGNS INDICATE OLD CURVE.
*; NOTE THAT THE DIGITAL FILTER RESPONSE BEGINS TO INCREASE AGAIN
*; AT HALF THE SAMPLE RATE ( 6510 HZ ).
*
*; THE PHASE CHARACTERISTICS OF THIS FILTER CAN BE EXAMINED.
*
*YSCALE = -PI,PI        ; ESTABLISHES RANGE OF INTEREST
*
*GRAPH PHASE
```




```
P*
*
*PUT :F1:POLE PZ        ; SAVE THE POLE LOCATION IN A DISK FILE BACKUP
*
*CODE POLE 1 INST(11        ; GENERATE 2920 ASSEMBLY CODE FOR THIS FILTER
B1=1.33989990  B2=-0.50541914
```

SAMPLE SPAS20 SESSION (Cont'd.)

```
        OPTIMIZED 2920 CODE IS NOW GENERATED. TO SAVE SPACE, SOME
      OF THE SCREEN OUTPUT HAS BEEN DELETED. NORMALLY ALL ATTEMPTS
      BY THE COMPILER TO GENERATE CODE ARE ECHOED ON THE SCREEN  )

INST=10
 POLE 1 = 0.71089458,0.34116779,2
  BEST: PERROR = 3.3795874/10**5,1.58846567/10**5

; NOTE: MAKE SURE SIGNAL IS <0.74635571
LDA OUT2_P1,OUT1_P1,R00
     ; OUT2_P1=1.00000000*OUT1_P1
LDA OUT1_P1,OUT0_P1,R00
     ; OUT1_P1=1.00000000*OUT0_P1
SUB OUT0_P1,OUT1_P1,R05
     ; OUT0_P1=1.00000000*OUT0_P1-0.031250000*OUT1_P1
ADD OUT0_P1,OUT0_P1,R03
     ; OUT0_P1=1.12500000*OUT0_P1-0.035156250*OUT1_P1
ADD OUT0_P1,OUT1_P1,R02
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1
SUB OUT0_P1,OUT2_P1,R01
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50000000*OUT2_P1
SUB OUT0_P1,OUT2_P1,R08
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50390625*OUT2_P1
ADD OUT0_P1,OUT2_P1,R11
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50341796*OUT2_P1
SUB OUT0_P1,OUT2_P1,R09
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50537109*OUT2_P1
ADD OUT0_P1,IN0_P1,R00
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50537109*OUT2_P1+1.00000000*IN0_P1
*
*; THE CODE COMMAND SPECIFIED THAT THE POLE PAIR BE CODED IN LESS THAN 11
*; INSTRUCTIONS, SO 10 INSTRUCTIONS WERE GENERATED, WITH COMMENTS.
*; THE FINAL ERROR IN RADIUS AND ANGLE FOR THE POLE PAIR WAS OF THE
*; ORDER OF 1/10**5 AS INDICATED ABOVE IN PERROR.
*; THIS OPTIMIZED 2920 ASSEMBLY CODE CAN NOW BE APPENDED TO A FILE
*; WHICH MAY CONTAIN OTHER CODED FUNCTIONAL BLOCKS OF A 2920 PROGRAM
*
*EXIT
```

SPECIFICATIONS

Operating Environment

REQUIRED HARDWARE:

Intellec® Microcomputer Development System
—Model 800
—Series II (Model 220, 230, or 240)
64K bytes of memory
One or two floppy diskette drives
—Single or Double Density
System Console
—CRT or TTY

REQUIRED SOFTWARE:

ISIS-II diskette operating system, V3.4 or later

OPTIONAL HARDWARE:

iSBC-310 High Speed Mathematics unit
(Speeds calculations by factor of 4X or more).

Universal PROM Programmer

OPTIONAL SOFTWARE:

MCI-20-SPS (Contains 2920 Signal Processor Applications Compiler and the 2920 Assembler Simulator.)

Documentation Package

2920 Signal Processing Applications Compiler User's Guide (121529)

Shipping Media

Single and Double Density Flexible Diskettes

ORDERING INFORMATION

Product Code	Description
MCI-20-SPAS	2920 Signal Processing Applications Compiler

2920 SOFTWARE SUPPORT PACKAGE

- **Complete software design and development support for the 2920**
- **Extends Intellec® Microcomputer Development System to support 2920 software development**

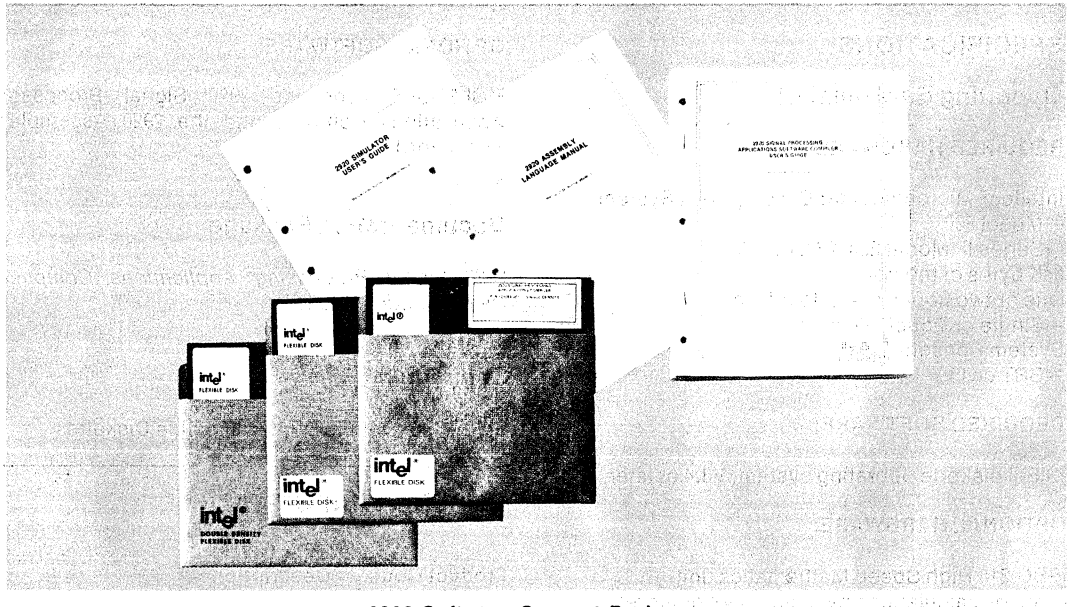
The 2920 Software Support Package furnishes a 2920 Signal Processing Applications Software/Compiler, 2920 Assembler, and 2920 Software Simulator. These three software design and development tools run on the Intellec® Microcomputer Development System.

The 2920 Signal Processing Application Software/Compiler is an interactive tool for designing software to be executed on the 2920 Signal Processor. The compiler accepts English-like statements from the user and generates 2920 assembly language code.

The assembler translates symbolic 2920 assembly language programs into the machine operation code. The user can load the code into the simulator for 2920 simulation or to the Universal PROM Programmer for 2920 EPROM programming.

The simulator, operating entirely in software, allows the user to test and symbolically debug 2920 programs. The user can specify input signals, simulate program execution, set up breakpoints, display input and output, and display and alter the contents of the 2920 registers and memory locations. The simulator can also stop or trace the program and constructively give the user access to the key elements inside a 2920 for analyzing his program.

The compiler, assembler, and simulator enable the designer to develop and test an entire program without a complete prototype design. The 2920 designer works on the Intellec® Microcomputer Development System rather than on a breadboard. The development system can program, store and recall programs or routines and aid in 2920 program design.



2920 Software Support Package

The following are trademarks of Intel Corporation and may be used only to identify Intel products: BXP, Intellec, Multibus, i, iSBC, Multimodule, ICE, iSBX, PROMPT, iCS, Library Manager, Promware, Insite, MCS, RMX, Intel, Megachassis, UPI, Intelevision, Microamp, μScope and the combination of ICE, iCS, iSBC, iSBX, MCS, or RMX and a numerical suffix.

Intel Corporation 1980

Sept 1980
1662208

2920 SIGNAL PROCESSING APPLICATIONS SOFTWARE/COMPILER

- **Compiler generates 2920 Assembly Language Code**
- **Extensive command set for designing electrical filters**
- **Graphics capability enhances analysis of filter response or piecewise linear function approximations**
- **Powerful MACRO capability for executing frequently used routines**
- **Interactive software support tool for 2920 Signal Processor**
- **Extends Intellec® Microcomputer Development System support of the 2920**
- **Contains MACRO library for several standard filters and signal processing functions**

The 2920 Signal Processing Applications Software/Compiler (SPAS20) is an interactive tool for designing software to execute on the 2920 Signal Processor.

The SPAS20 package can be visualized as being comprised of four inter-related sections: A compiler section, a filter design section, a curve fitting section, and a MACRO section.

Among the abilities of SPAS20 are: ability to generate 2920 assembly language code directly from specifications of signal processing building blocks such as filters and waveform generators; ability to generate 2920 assembly language code for several classes of algebraic equations such as $Y = C * X$, $Y = C * Y$, and $Y = C * X + Y$ where X, Y are variables and C is a constant; ability to generate 2920 assembly language code for one variable function $Y(X) = F(X)$; ability to examine time and frequency responses of filter sections specified by continuous or sampled poles and zeroes; ability to examine piecewise linear approximation of specific function; ability for users to implement more complex commands by grouping sets of commonly used commands into a MACRO.

The SPAS20 package runs under ISIS-II on any Intellec® Microcomputer Development System with 64K RAM. The output of SPAS20 can be assembled with the 2920 assembler, tested with the 2920 Simulator, and programmed into the 2920 chip with the Universal PROM Programmer for prototyping.

FUNCTIONAL DESCRIPTION

The 2920 Signal Processing Applications Software/ Compiler gives the analog designer a "high level language" for his 2920 applications—it decreases the need to code 2920 assembly language. Furthermore, the compiler is interactive. This feature enables the designer to define a filter, or transfer function, graph their response, and change their parameters many times, without having to program and test in an actual 2920 implementation.

Once a filter is realized by moving poles and zeros in the continuous and sampled planes, the filter may be coded and written onto an ISIS file. Similarly, after a function Y = F(X) has been defined, the code for a piecewise linear approximation can be stored onto an ISIS file. Several other file commands are available to store and retrieve command sequences for SPAS20 sessions.

SPAS20 Command Language

DEFINE	This command defines a pole or zero by associating it with a number (i.e., POLE 3), and with real and imaginary coordinates in the continuous or sampled plane. This command also defines a symbol by associating a name with a numeric value, or a MACRO by providing a pointer to a specified command sequence.
GRAPH/ OGRAPH	This command graphically displays the values of object(s) specified. For example, GRAPH GAIN and GRAPH PHASE are used to display filter response. The OGRAPH command will "overgraph" the new response over the old response, after any changes have been made. (You may also graph Group Delay, Step, and Impulse.)
MOVE	Allows the definition of a pole or zero to be changed—its coordinates, its plane, or both.
REMOVE	Deletes the definition of a pole, zero, symbol, or macro.
HELP	Types an explanatory message on the console, pertaining to a command or its attributes.
FIT	This command performs curve fitting, i.e. it approximates an arbitrary user supplied function with a piecewise linear function.
DATA	This command allows for specification of a set of vertices (i.e. X – Y coordinate pairs) which determine a piecewise linear approximation of some defined function, filter response characteristics, etc.
HOLD	Command to correct attenuation due to sample-and-hold distortion: if ON, it corrects absolute gain by sin(x)/x and phase by adding x, where x=TS*FREQ*π. It corrects group delay by subtracting π*TS.
EVALUATE	Gives the decimal numeric value of any expression.
CODE	Creates 2920 assembly language code for given poles, and zeros, equations, and user defined functions.

The SPAS20 compiler also recognizes the following commands for file handling:

PUT/ APPEND	Writes out objects (commands) to a specified file, either creating a new one or appending an existing one. This enables the user to store all or part of a SPAS20 session on a diskette to be brought back later with the INCLUDE command.
DISPLAY	Copies the contents of a file to the console.
INCLUDE	Executes a sequence of instructions from a diskette file as if they were typed in from the console.
LIST	Creates a file containing all console interactions.

In addition to naming macros for specific command sequences, compound and conditional commands may be formed using all of the above statements. These compound commands are:

IF	Establishes conditional flow of control within a block of commands.
REPEAT	Used for repetition of a block of commands; executes indefinitely or until a condition is met (using WHILE, UNTIL, and END statements).
COUNT	Establishes the number of times a command sequence is to be executed, in a looping fashion.

SPAS20 MACRO Facility

A macro is a sequence of commands that is stored on a temporary diskette file. The command sequence is executed when the macro name is entered as a command. This saves repetitive entry of the sequence, and permits alogorithms to be saved on diskette for future use. This SPAS20 facility allows you to do the following:

- Display the text of any macro.
- Define a macro, specifying its name and any parameters that are to be used by the block. This definition is followed by the contents of the macro (commands) and the EM statement to end its definition.
- Invoke a macro by entering its name and appropriate values for any parameters.
- List the names of all defined macros.
- Remove any or all macros.

Intel also supplies several MACRO library files containing the following commonly needed MACROs:

- Filter design MACROS
  - — Butterworth filter
  - — Chebyshev filter
  - — Bilinear transform
  - — Evaluate gain or phase of digital filter in parallel form
  - — Time response simulation
- Function design MACROs
  - — Code and error optimization
  - — Calculate instertitial error
- MACROs for generation of 2920 code
  - — Code for all-POLE filter
  - — Input and A/D conversion
  - — Multiplication
  - — Division
  - — Logarithm functions
  - — Square-root functions
  - — Sinewave oscillator

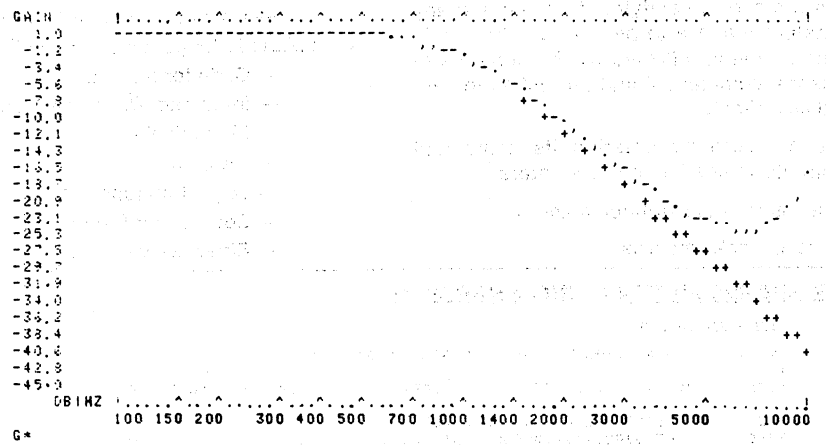
SAMPLE SPAS20 FILTER DESIGN SESSION

```
-:FI : SPAS20 . SFT
ISIS-II 2920 SIGNAL PROCESSING APPLICATIONS COMPILER. V2.0
*
*DEFINE POLE 1 = -707,707        ; CREATE A POLE IN CONTINUOUS S-PLANE
*
*PZ        ; LIST ALL POLES AND ZEROS
POLE 1 = -707.00000,707.00000,CONTINUOUS
*
*FSCALE = 100,10000        ; ESTABLISHES FREQUENCY RANGE OF INTEREST
*
*YSCALE = -45,1        ; ESTABLISHES MAGNITUDE RESPONSE RANGE OF INTEREST
*
*GRAPH GAIN        ; PLOT MAGNITUDE RESPONSE OF POLE PAIR

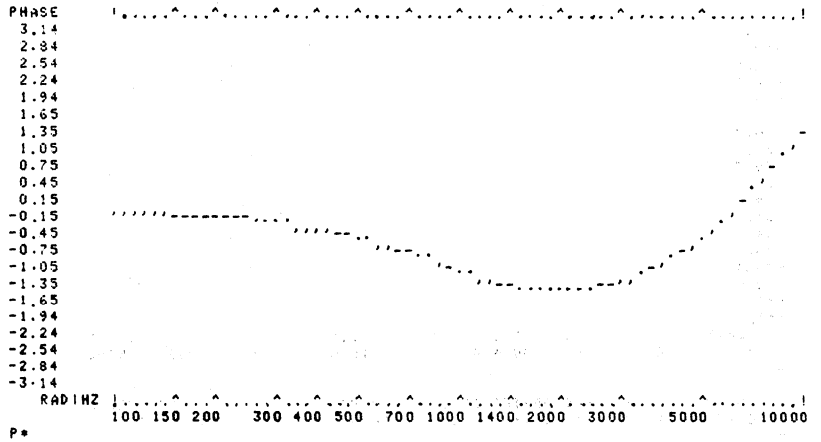
GAIN      !.....^...^.....^...^...^....^....^....^....^....^....^.....^.....^.....!
  1.0     ---------------------------....
 -1.2                                   ''--.
 -3.4                                        '-.
 -5.6                                          '.
 -7.8                                           '-
-10.0                                             '-
-12.1                                              '.
-14.3                                               '.
-16.5                                                '.
-18.7                                                  '-.
-20.9                                                    '-
-23.1                                                     '-
-25.3                                                      '.
-27.5                                                       '.
-29.7                                                       '.
-31.9                                                        '-.
-34.0                                                          '-
-36.2                                                          '-
-38.4                                                           '.
-40.6                                                           '.
-42.8                                                            '
-45.0
    DBIHZ !.....^...^.....^...^...^....^....^... ^ ...^.....^.....^.....^.....!
          100 150 200    300 400 500   700 1000 1400 2000   3000      5000      10000
**
*
*; THE UNITS USED IN GRAPHING GAIN ARE SHOWN IN THE LOWER LEFT CORNER.
*; GAIN IN DECIBELS IS GRAPHED VERSES FREQUENCY IN HERTZ.
*
*; PREPARE TO MOVE TO THE DIGITAL DOMAIN.
*; SAMPLE RATE MUST BE SPECIFIED.
*
*TS = 1/13020        ; RATE FOR 192 INSTRUCTION PROGRAM AND 10MHZ CLOCK
TS = 7.6805004/10**5
*
```

SAMPLE SPAS20 FILTER DESIGN SESSION (Cont'd.)

```
*MOVE POLE TO Z        ; CONVERT FILTER TO DIGITAL VIA MATCHED-Z TRANSFORMATION
1 POLES/ZEROES MOVED
*
*P          ; LIST TRANSFORMED POLE
POLE 1 = 0.71092836,0.34118369,Z
*
*; COMPARE RESPONSES OF THE ANALOG AND DIGITAL FILTERS BY GRAPHING THE
*; NEW RESPONSE OVER THE OLD
*
*GRAPH GAIN
```




```
G*
*
*; PLUS SIGNS INDICATE OLD CURVE
*; NOTE THAT THE DIGITAL FILTER RESPONSE BEGINS TO INCREASE AGAIN
*; AT HALF THE SAMPLE RATE ( 6510 HZ ).
*
*; THE PHASE CHARACTERISTICS OF THIS FILTER CAN BE EXAMINED
*
*YSCALE = -PI,PI        ; ESTABLISHES RANGE OF INTEREST
*
*GRAPH PHASE
```




```
P*
*
*PUT :F1:POLE PZ        ; SAVE THE POLE LOCATION IN A DISK FILE BACKUP
*
*CODE POLE 1 INST<11        ; GENERATE 2920 ASSEMBLY CODE FOR THIS FILTER
B1=1 33989990  B2=-0.50541914
```

SAMPLE SPAS20 FILTER DESIGN SESSION (Cont'd.)

```
        OPTIMIZED 2920 CODE IS NOW GENERATED  TO SAVE SPACE, SOME
     OF THE SCREEN OUTPUT HAS BEEN DELETED  NORMALLY ALL ATTEMPTS
     BY THE COMPILER TO GENERATE CODE ARE ECHOED ON THE SCREEN. )

INST=10
 POLE 1 = 0.71089458,0.34116779,Z
  BEST: PERROR = 3.3795874/10**5,1.58846567/10**5

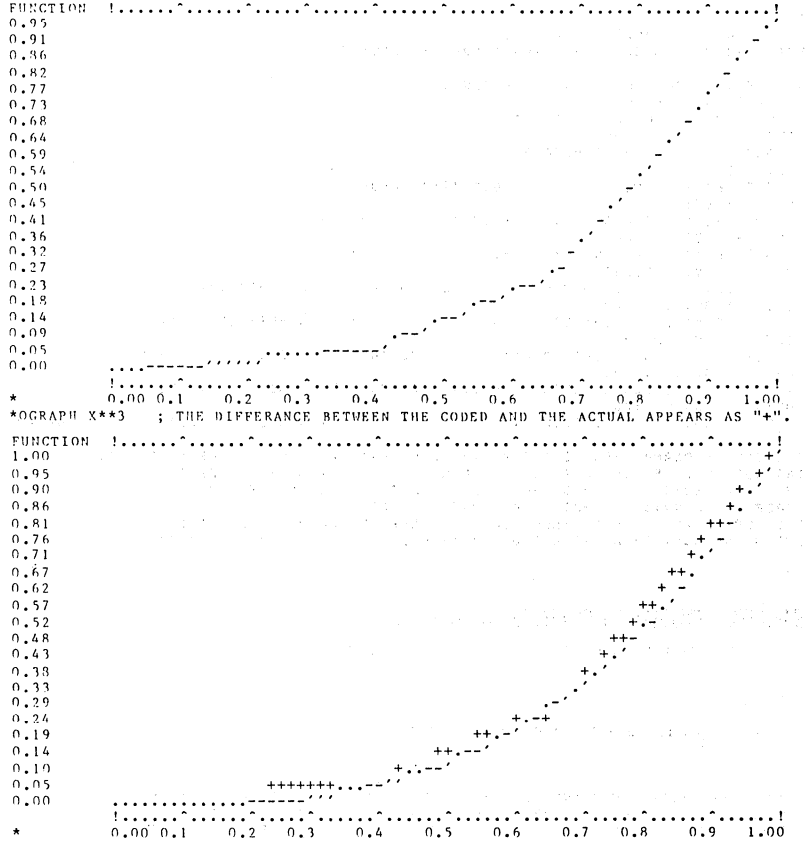
; NOTE: MAKE SURE SIGNAL IS <0.74635571
LDA OUT2_P1,OUT1_P1,R00
     ; OUT2_P1=1.00000000*OUT1_P1
LDA OUT1_P1,OUT0_P1,R00
     ; OUT1_P1=1.00000000*OUT0_P1
SUB OUT0_P1,OUT1_P1,R05
     ; OUT0_P1=1.00000000*OUT0_P1-0.031250000*OUT1_P1
ADD OUT0_P1,OUT0_P1,R03
     ; OUT0_P1=1.12500000*OUT0_P1-0.035156250*OUT1_P1
ADD OUT0_P1,OUT1_P1,R02
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1
SUB OUT0_P1,OUT2_P1,R01
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50000000*OUT2_P1
SUB OUT0_P1,OUT2_P1,R08
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50390625*OUT2_P1
ADD OUT0_P1,OUT2_P1,R11
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50341796*OUT2_P1
SUB OUT0_P1,OUT2_P1,R09
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50537109*OUT2_P1
ADD OUT0_P1,IN0_P1,R00
     ; OUT0_P1=1.12500000*OUT0_P1+0.21484375*OUT1_P1-0.50537109*OUT2_P1+1.00000000*IN0_P1
*
*; THE CODE COMMAND SPECIFIED THAT THE POLE PAIR BE CODED IN LESS THAN 11
*; INSTRUCTIONS, SO 10 INSTRUCTIONS WERE GENERATED, WITH COMMENTS.
*; THE FINAL ERROR IN RADIUS AND ANGLE FOR THE POLE PAIR WAS OF THE
*; ORDER OF 1/10**5 AS INDICATED ABOVE IN PERROR.
*; THIS OPTIMIZED 2920 ASSEMBLY CODE CAN NOW BE APPENDED TO A FILE
*; WHICH MAY CONTAIN OTHER CODED FUNCTIONAL BLOCKS OF A 2920 PROGRAM
*
*EXIT
```

SAMPLE SPAS20 CURVE FITTING SESSION

```
-;        DEMONSTRATION OF THE SPAS20 CURVE-FITTING PACKAGE
-
-
-SPAS20.SFT

ISIS-II 2920 SIGNAL PROCESSING APPLICATIONS SOFTWARE/COMPILER, V2.0
*LIST XCUBED.829
*
*
*;  THE CURVE FITTING COMMANDS IN SPAS20 WILL GENERATE 2920 CODE TO CALCULATE
*;  SOME FUNCTION SUCH AS  X**3.   X**3 COULD BE COMPUTED ON THE 2920 CHIP
*;  WITH TWO MULTIPLIES USING ABOUT 18 INSTRUCTIONS AND THE DAR.  HOWEVER IT
*;  WOULD TIE UP THE DAR TOO LONG.  THE CODE GENERATED BY THE CURVE FITTING
*;  COMMANDS DOES NOT USE THE DAR.
*
*CODE FIT X$CUBED(X) = X**3 ERROR<.05   ;ERROR BOUND OF .05
*
*CODE     ; HERE IS THE CODE GENERATED.
LDA TEMP,X,R00
     ; TEMP=1.00000000*X
LDA XCUBED,X,R01
     ; XCUBED=0.50000000*X
ADD XCUBED,X,R06
     ; XCUBED=0.51562500*X
ADD TEMP,X,R01
     ; TEMP=0.50000000*X+1.00000000*TEMP
ADD XCUBED,TEMP,R05
     ; XCUBED=1.00000000*XCUBED+0.031250000*TEMP
SUB XCUBED,TEMP,R02
     ; XCUBED=1.00000000*XCUBED-0.21875000*TEMP
ADD TEMP,X,R00
     ; TEMP=1.00000000*X+1.00000000*TEMP
ADD XCUBED,TEMP,R08
     ; XCUBED=1.00000000*XCUBED+0.0039062500*TEMP
SUB XCUBED,TEMP,R04
     ; XCUBED=1.00000000*XCUBED-0.058593750*TEMP
LDA XCUBED,XCUBED,L02
     ; XCUBED=4.00000000*XCUBED-0.23437500*TEMP
  *
  *INST     ; THE FUNCTION WAS CODED IN THIS MANY INSTRUCTIONS;
  INST = 10.0000000
  *
```

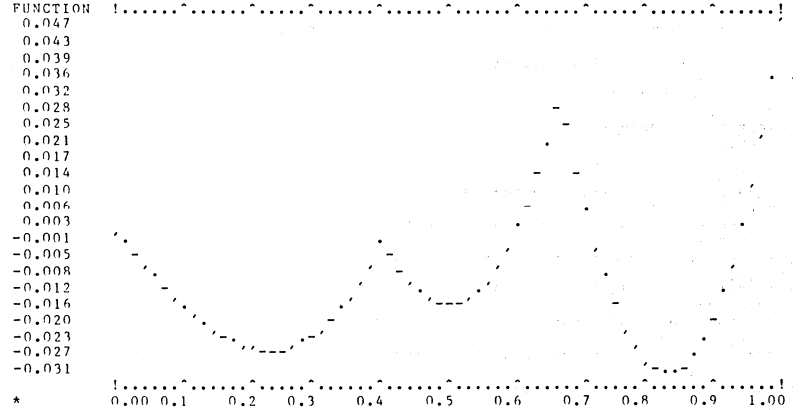
```
*ERROR     ; THE CODE APPROXIMATES X**3 WITHIN THIS ERROR;
ERROR = 0.046875000
*
*DATA 0 THRU 1    ;EXAMINE THE PIECEWISE LINEAR FUNCTIONS VERTECES.
DATA 0.00000000 THRU 1.00000000 = 0.00000000 AT 0.00000000,&
0.065625012 AT 0.40000000,&
0.26562500 AT 0.66666669,&
0.95312500 AT 1.00000000
*
*GRAPH DATA(X)    ;THE DATA ARRAY APPROXIMATES THE FUNCTION AND CAN BE GRAPHED.
```




```
*OGRAPH X**3    ; THE DIFFERANCE BETWEEN THE CODED AND THE ACTUAL APPEARS AS "+".
```



```
*GRA (X**3)-DATA(X)    ; THE ERROR WILL BE GRAPHED.
```




```
*EXIT    ; THAT'S ALL FOLKS
-
```

2920 ASSEMBLER

2920 program development on Intellec® Microcomputer Development Systems

Translates symbolic assembly language instructions into 2920 machine code

Produces Assembly Listing, Object Code File, and Error Diagnostics

Output used for 2920 programming with the Intellec PROM Programmer or the 2920 Simulator for program debug

The 2920 Assembler translates symbolic 2920 Assembly Language instructions into the appropriate machine operation codes. Through this facility, the programmer is able to symbolically program 2920 hardware operations. Compared to machine code, these symbolic references provide faster programming, easier debugging, and greater reliability.

The Assembler produces an object code file (executable machine code), a complete assembly listing, and error diagnostics. The object code output from the Assembler may be loaded directly into the Intel Universal PROM Programmer for programming the 2920 EPROM. The object code may also be loaded to the 2920 Simulator for 2920 system design and debug.

The 2920 Assembler runs under the ISIS-II Operating System on the Intellec Microcomputer Development Systems.

Sample 2920 Assembly Listing

```
ISIS-II 2920 ASSEMBLER X102                                    PAGE   1

ASSEMBLER INVOKED BY: AS2920 SAW ASM DEBUG

SAWTOOTH WAVE GENERATOR

  LINE   LOC OBJECT SOURCE STATEMENT

     1             $TITLE('SAWTOOTH WAVE GENERATOR')
     2             ;
     3             ;
     4     0 0000EF      IN0                ; SAMPLE INPUT CHANNEL 0
     5     1 0000EF      IN0
     6     2 0000EF      IN0
     7     3 008AEB      SUB Y,KP1,IN0      ; SIMULTANEOUSLY CALCULATE SAWTOOTH
     8     4 008A0A      SUB Y,KP1,R1,IN0   ; BY SUBTRACTING 3/16 FROM Y
     9     5 0044EF      LDA DAR,Y,IN0      ; ALSO CHECK SIGN BIT OF Y
    10     6 7A8AED      ADD Y,KP7,CNDS     ; IF Y NEGATIVE START NEXT TOOTH
    11     7 6000EF      CVTS               ; CONVERT SAMPLED INPUT TO DIGITAL (SIGN BIT)
    12     8 7082EF      LDA Y,KP0,CNDS     ; SUPPRESS SAWTOOTH IF INPUT WAS < 0
    13     9 4044EF      LDA DAR,Y          ; PREPARE TO OUTPUT SAWTOOTH
    14    10 4000EF      NOP                ; ANALOG LEVEL MUST SETTLE
    15    11 4000EF      NOP
    16    12 4000EF      NOP
    17    13 8000EF      OUT0               ; OUTPUT SAWTOOTH
    18    14 8000EF      OUT0
    19    15 8000EF      OUT0
    20    16 5000EF      EOP                ; PROGRAM WILL END IN THREE MORE INSTRUCTIONS
    21    17 8000EF      OUT0
    22    18 8000EF      OUT0
    23    19 8000EF      OUT0
    24
    25                   END


SYMBOL:                        VALUE:

Y                                    0


ASSEMBLY COMPLETE
ERRORS     =     0
WARNINGS   =     0
RAMSIZE    =     1
ROMSIZE    =    20
```

2920 SIMULATOR

Speeds test and debug of 2920 programs

Simulates 2920 internal operation

Operates on Intellec® Microcomputer Development Systems

Allows users to specify 2920 input signals, and display or alter ROM, RAM, and system variables

Output and internal data can be saved on disk for further analysis.

Provides ability to set breakpoints and to collect trace information

Easy-to-learn commands

The 2920 Simulator is a software facility that provides testing and symbolic debugging of 2920 programs in an Intellec Microcomputer Development Systems environment. The 2920 designers have the capability to specify the 2920 input signals, to set breakpoints, to collect and display 2920 input, output, system variables, and ROM and RAM data values during simulation. The 2920 Simulator accepts the hex format object files produced by the 2920 assembler. Output values and internal trace data may be saved on ISIS-II disk files for further analysis.

Functional Description

2920 Input Signal Specification

The four analog signal inputs to the 2920 processor can be specified as algebraic combinations of basic functions of time. The basic functions are SIN, COS, EXP, LOG, SQR, SAW, SQW, ABS.

2920 Simulation

The simulation of 2920 machine instructions is performed in software. All 2920 internal registers, memory, input values, output values, and other system variables can be examined and modified. The internal processing of the 2920 is simulated. Time constants for the sample and hold capacitators are assumed to be zero. Calculation of input signals is performed in single precision floating point. The speed of simulation varies with the complexity of the input signal, breakpoint setting, and trace condition. Exclusive of I/O time requirements, 2920 instructions will be simulated at a rate of approximately several hundred instructions per second.

Breakpoint Capabilities

After each instruction is simulated, the breakpoint is evaluated to determine whether to stop or continue simulation. Conditional breakpoints are also provided for debugging purposes. Simulation can be manually stopped at any time by pressing the ESC key on the Intellec console.

Trace Capabilities

Based on the qualifier's condition, trace data records can be collected during simulation. The trace data records are stored in Intellec resident memory and are optionally written to the console for display or to a disk file for record.

Symbolic Debugging Capabilities

The 2920 Simulator allows the user to refer to program addresses symbolically. The user can load or save the symbols generated from the hex format object files or created during the debugging session. 2920 program memory in ROM can be disassembled, or filled with assembled instructions.

The 2920 Simulator is designed to provide users with powerful, easy-to-use commands. The user interfaces to the Simulator by entering commands to the Intellec console. The commands consist of one command line, terminated by one of the two line terminators — carriage return or line feed.

The 2920 Simulator offers two types of commands:

Simulation and Control Commands

Command	Operation
Simulate	Starts simulation of the input signals and the 2920 program in simulated ROM memory. Initial setting is "FOREVER."
Trace	Controls the trace selection. Initial setting is "TIME."
Qualifier	Sets qualifier condition during trace. Initial setting is "ALWAYS."
Breakpoint	Sets breakpoint condition during simulation. Initial setting is "NEVER."

Interrogation and Utility Commands

Command	Operation
Display	Displays the values of symbols, RAM, ROM, input, output, registers and system variables.
Change	Alters the values of symbols, RAM, ROM, input, register and system variables.
Base	Establishes the mode of display for output data.
Suffix	Establishes the mode of display for input data.
Load	Fetches user symbol table and object code from input device.
Save	Sends user symbol table and object code to output device.
Define	Enters symbol name and value to user symbol table.
Console	Controls the console on/off display.
List	Defines list device.
Exit	Returns program control to ISIS-II.
Evaluate	Converts expression to equivalent values in binary, decimal, and hex.
Remove	Deletes symbols from symbol table.
Help	Provides a brief summary of the syntax for the command languages.
Graphics On/Off	Switches output mode between list and graphics.
X Size	Enters horizontal display size.

Keyword References

The 2920 Simulator provides users with keyword references to gain access to all of the numeric valued system variables including simulated 2920's memory, register, status flags and input/output. These keyword references can function as the evaluation command, display command, and change command.

- **2920 Processor Keyword References**

IN0	Analog input 0 in volts
IN1	Analog input 1 in volts
IN2	Analog input 2 in volts
IN3	Analog input 3 in volts
OUT0	Analog output 0 in volts (read only)
OUT1	Analog output 1 in volts (read only)
OUT2	Analog output 2 in volts (read only)
OUT3	Analog output 3 in volts (read only)
OUT4	Analog output 4 in volts (read only)
OUT5	Analog output 5 in volts (read only)
OUT6	Analog output 6 in volts (read only)
OUT7	Analog output 7 in volts (read only)
IN	Sampled and held analog input signal in volts
DAR	Digital to analog register (RAM location 40)
PC	Program counter (integer 1 to 192)
CY	Carry (integer 0 or 1)
OVF	Overflow (integer 0 or 1, read only)
OVE	Overflow enable (integer 0 or 1)

- **Software Simulator Keyword References**

TIME	Elapsed simulated time in seconds (read only)
TQUAL	Time when the qualifier last matched in seconds (read only)
COUNT	Number of instructions simulated since last SIMULATE command (integer, read only)
BUFFERSIZE	Number of trace data records (integer, read only)
TINST	Time between successive instructions in seconds (read only)
SIZE	Number of instructions in program disregarding actual EOP placement
TPROG	Time between successive program passes in seconds
VREF	Reference analog level voltage in volts

The above keyword references are designed to aid 2920 program debugging.

ISIS Compatibilities

The 2920 software simulator runs under the ISIS "submit" facility. The 2920 software simulator uses the ISIS-II line editing capabilities to correct errors in an input line on the Intellec console.

Sample 2920 Simulation Session

```
-
-SM2920.SFT

ISIS-II 2920 SIMULATOR, V1.1
*
*;    THIS IS THE SIMULATION OF THE 'SAWTOOTH GENERATOR'
*
*LIST SRC.LOG        ; LISTS THE SIMULATION SESSION TO AN ISIS FILE
*LOAD SRC.HEX        ; LOAD THE OBJECT CODE INTO THE 2920 SIMUALTOR
*ROM 0 TO 5          ; DISPLAY SRC PROGRAN
ROM 000 = LDA .K,KP5,R00,NOP
ROM 001 = ADD .K,KP1,R05,NOP
ROM 002 = LDA .K,.K,R02,NOP
ROM 003 = SUB .OSC,.K,R00,NOP
ROM 004 = LDA DAR,.OSC,R00,NOP
ROM 005 = ADD .OSC,KP4,L01,CNDS
*TPROG=1/10000        ; SET THE SAMPLE RATE
*TRA=PC,RAM .K        ; SET THE ITEMS TO BE TRACED
*BASE=B               ; DISPLAY THE RESULTS IN BINARY
*SIMULATE FROM 0 TILL COUNT=3 ; SIMULATE THREE INSTRUCTIONS
                                TO VERIFY CONSTANT

   PC                                RAM 0
SIMULATION BEGUN
    1.000000000000000000000000     0.101000000000000000000000
    2.0000000E+0                   0.101000010000000000000000
    3.0000000E+0                   0.001010000100000000000000
SIMULATION TERMINATED
*QUALIFIER=PC=0      ; TRACE EVERY PROGRAM PASS
*TRACE=T,DAR,RAM .OSC ; SET THE ITEMS TO BE TRACED
*RAM .OSC=ONE        ; INITIALIZE THE RAM LOCATION
*BREAKPOINT=T>.00132 ; SIMULATE FOR TWO CYCLES
*BASE=D              ; SET THE BASE TO DECIMAL
*SIMULATE FROM 0     ; BEGIN SIMULATION
   T                  DAR                RAM 1
```

```
SIMULATION BEGUN
     0.00010000        0.83984375        0.84277334
     0.00020000        0.68359375        0.68554683
     0.00030000        0.52734375        0.52832026
     0.00040000        0.36718750        0.37109370
     0.00050000        0.21093750        0.21386714
     0.00060000        0.05468750        0.05664056
     0.00070000       -0.10156250        0.89941396
     0.00080000        0.73828125        0.74218745
     0.00090000        0.58203125        0.58496089
     0.00100000        0.42578125        0.42773433
     0.00110000        0.26953125        0.27050776
     0.00120000        0.10937500        0.11328119
     0.00130000       -0.04687500        0.95605459
SIMULATION TERMINATED
*GRAPH ON            ; SWITCHES THE DISPLAY MODE TO GRAPHICS
*TRACE=T,0,DAR,RAM .OSC,-1,-1,1,1  ; SETS ITEMS TO BE TRACED
*RAM .OSC=ONE        ; INITIALIZE THE RAM LOCATION
*SIMULATE FROM 0
   T                 0                 DAR               RAM 1             -1
        -1                  1                 1
SIMULATION BEGUN
->*                                      1                                  *     *
0 *                                      1                            *           *
. *                                      1                        *           *
0 *                                      1                    *               *
0 *                                      1           *                        *
0 *                                      1 *                                  *
1 *                                  2   1                                3   *
0 *                                      1                                *   *
0 *                                      1                        *           *
0 *                                      1                    *               *
0 *                                      1               *                    *
  *                                      1    *                               *
  *                                    2 1                                3 *
SIMULATION TERMINATED
*EXIT
-
```

SPECIFICATIONS

Operating Equipment

Required Hardware

Intellec® Microcomputer Development System
—Model 800 or 888
—Series II Model 220, 225, 230
64K Bytes of RAM Memory
One or two Floppy Disk Drives
—Single or Double Density
System Console
—CRT or interactive hard copy device

Required Software

ISIS-II Diskette Operating System

Optional Hardware

Line Printer
iSBC-310 High-Speed Mathematics Unit
Universal PROM Programmer

Optional Software

FORTRAN-80 (Product Code MDS-301)

Documentation Package

2920 Assembly User's Guide (9800987)
2920 Simulator User's Guide (9800988)
2920 Signal Processing Application Compiler User's Guide (121529)

Shipping Media

Flexible Diskettes
—Single and Double Density

ORDERING INFORMATION

Product Code	Description
MCI-20-SPS	2920 Software Support Package Includes 2920 Signal Processing Application Software/Compiler and 2920 Assembler/Similator Software

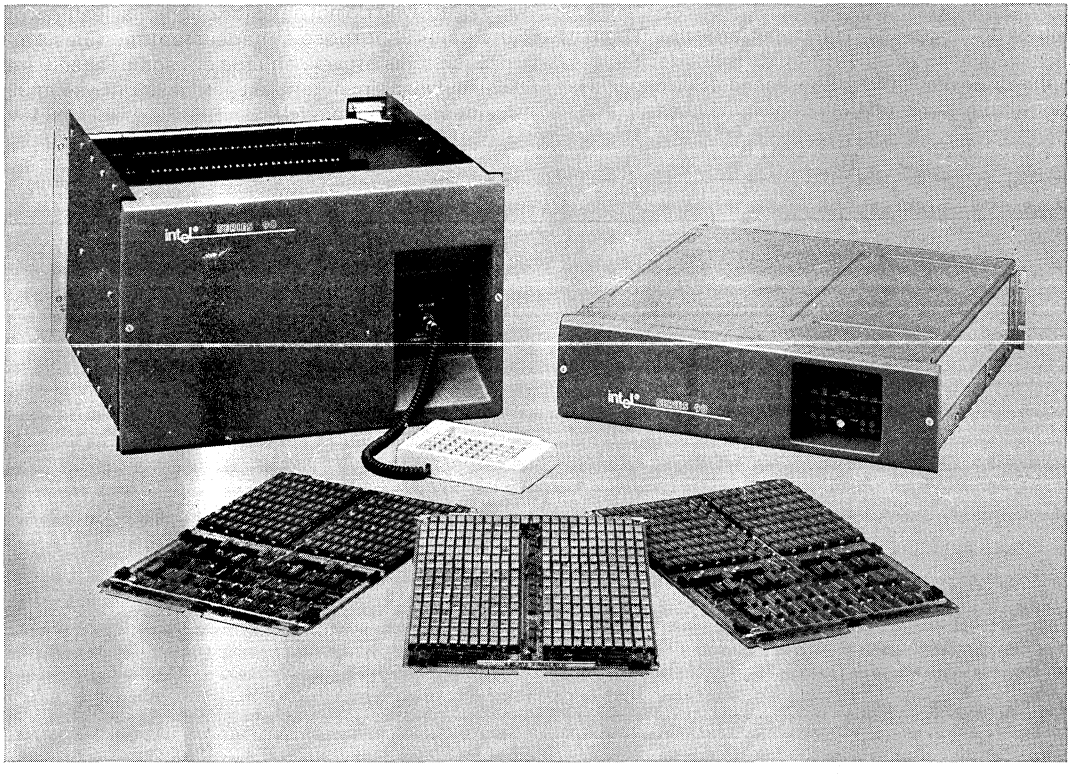
Memory Systems 22

SERIES 90
GENERAL PURPOSE MEMORY SYSTEM

- **Memory Modules only or Fully Integrated and Tested Packaged Systems**
- **Standard BXP™ Memory Bus**
- **10 MHz Word Transfer Rates (Maximum)**
- **Word Widths from 16 Bits to 88 Bits**
- **Multiple System Packages Available**
- **Variety of Storage Technologies and Performances**
- **Optional Enhancement Modules Available Including ECC**
- **User Configurable for Each Application**

The Intel® Series 90 is a family of general purpose memory products which are available as standard systems and are user configurable to provide an optimum solution for individual applications. The system can incorporate new and future generations of technologies as they become available, without changing the user's system design, providing an easy growth path for the future. All systems are fully tested to Intel's quality standards in the user's configuration.

For full information, refer to the Series 90 Data Catalog.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_{el}$, INTEL, INTELLEC, MCS, ^{i}m, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation, 1981.

JUNE 1981
AFN-01720B

FUNCTIONAL DESCRIPTION

The heart of the Series 90 is the standard BXP™ (Byte Exchange Path) memory bus, which provides a common data exchange and control path for a wide variety of memory storage media. Basic system architecture is shown in Figure 1. The user has the option of interfacing to the system either to the BXP bus (System 90 and System 92 only) or to the general purpose enhancement modules described below.

There are three basic systems presently in the Series 90 family. The System 90 is a high-performance dynamic memory system with performances of 275 and 350 nanosecond cycle times and capacities ranging from 256 kilobytes up to 16 megabytes within a single chassis. The System 91 is a lower-cost product with slower cycle times of 500 and 650 nanoseconds and capacities up to 32 megabytes in a single chassis. Greater capacities can be easily accommodated by daisy-chaining systems together. The System 91 becomes cost effective at capacities greater than a megabyte.

The System 92 is the highest-performance member of the Series 90 family and uses static technology with a cycle time of 100 nanoseconds. See capacities range from 64 Kb to 4 Mb in a single chassis.

INTERFACE TYPES

As stated above, the Series 90 may be used by interfacing either to the BXP bus or to a general purpose enhancement module. This section gives the user an overview of the product features and enables a preliminary selection of the optimum configuration.

BXP™ Systems

The BXP memory bus provides an extremely flexible, yet easy to use, set of protocols. The bus may be operated in either a synchronous or asynchronous mode of operation, for read, write, read-modify-write, or swap cycles. Interleaving is an important feature of these systems, improving thruput by overlapping memory module cycles. Interleaving enables word data rates up to 10 MHz bus limitation, independent of the cycle time of the memory modules and can be achieved with either synchronous (sequential addresses) cycles or asynchronous cycles.

For users who do not need the additional features of the general purpose enhancement modules, the BXP systems provide an efficient solution, and are available within either the System 90 or System 92 members of the family.

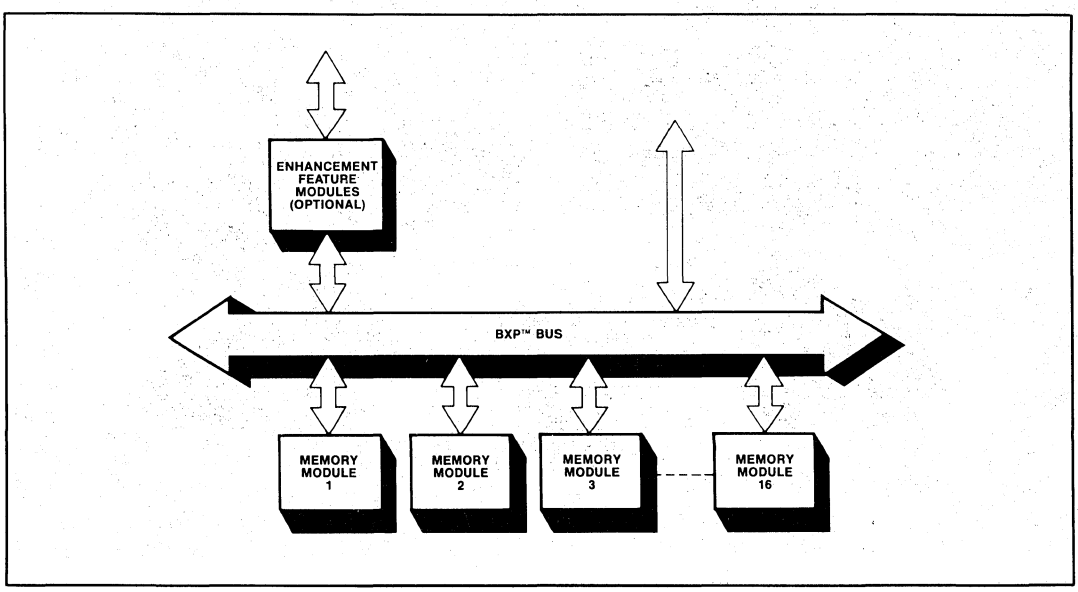


Figure 1. System Architecture

Enhancement Features

These enhancements provide additional degrees of functionality over and above that provided by the ECX (error correcting) and iQX (intelligent controller). See Table 2 for a feature comparison, including memory module speeds and capacities.

ECX

This feature essentially performs all the functions of the BXP systems, and provides additional functions, primarily Error Checking and Correction (ECC). This corrects all single bit errors and detects double bit errors. These systems are also available with an optional error logger and display to facilitate maintenance and diagnostics.

iQX

For asynchronous applications that do not require interleaving, this intelligent enhancement offers advanced features such as ECC; fault-tolerant (uninterruptible) operation; versatile maintenance tools; self-test and automatic memory diagnostics; remote error reporting and fault-isolation; offloading of memory management overhead; and simple message-driven interface protocol.

SYSTEM PACKAGES

The Series 90 is available as a completely tested and packaged system. There are two Series 90 packaged systems available for 19" rack mount.

The 5¼" high HMS chassis has self-contained power supplies and cooling, and will house up to four memory modules.

The 12¼" high VMS has self-contained power supplies and requires externally supplied vertical air flow. An optional 5¼" rack-mountable blower assembly is available. The VMS will house up to 16 memory modules.

The maximum system capacities as a function of the module capacities shown in Table 2 are shown in Table 3.

Table 2. Feature Comparison

	BXP™ Systems	Enhancement Features	
		ECX	iQX
System 90	275 ns cycle, 256 Kb; 350 ns cycle, 256 or 1024 Kb; synchronous or asynchronous; interleaved	275 ns cycle, 256 Kb; 350 ns cycle, 256 or 1024 Kb; synchronous or asynchronous; ECC interleaved;	500 ns cycle, 256 or 1024 Kb; asynchronous; ECC plus fault tolerance; advanced diagnostics
System 91	Not Available	500 or 650 ns cycle; 2048 Kb; synchronous or asynchronous; ECC	800 ns cycle; 2048 Kb; asynchronous; ECC plus fault tolerance; advanced diagnostics
System 92	100 ns cycle, 64 Kb or 256 Kb; synchronous or asynchronous	110 ns cycle, 64 or 256 Kb; synchronous or asynchronous; ECC	Not Available

Table 3. System Capacity Versus Module Capacity. Kb (Kilobytes), Mb (Megabytes)

	Kb per Memory Module	HMS Maximum Capacity	VMS Maximum Capacity
System 90	256 Kb 1024 Kb	1 Mb 4 Mb	4 Mb 16 Mb
System 91	2048 Kb	8 Mb	32 Mb
System 92	64 Kb 256 Kb	256 Kb 1024 Kb	1 Mb 4 Mb

ORDERING INFORMATION

Consult Series 90 Configuration Guide.

PRELIMINARY

SERIES 90/iQX
INTELLIGENT MEMORY SYSTEM

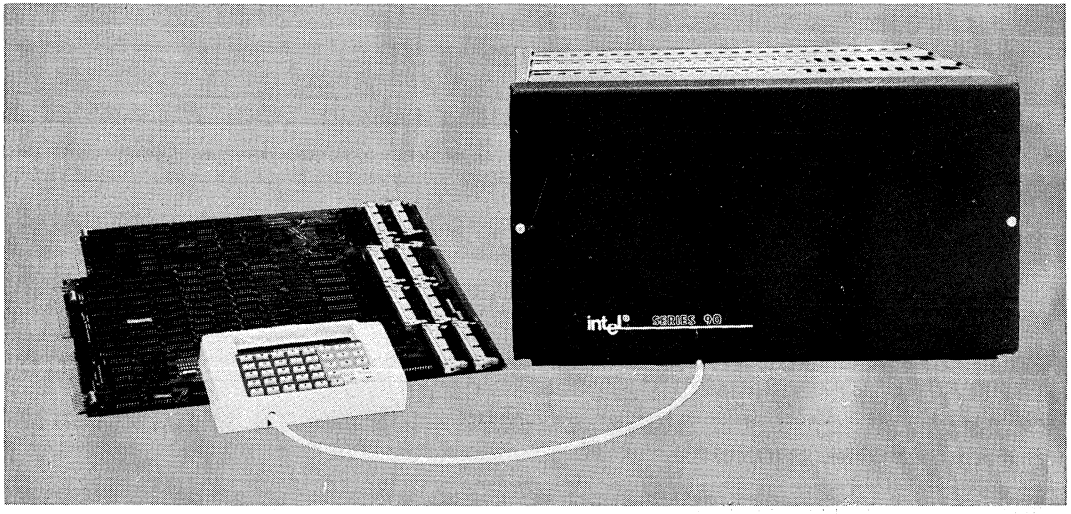
- **Uninterrupted System Availability**
- **Simple Message-Driven Interface**
- **Memory Task Processing**
- **ECC Plus Soft-Error Scrubbing**
- **Built-In Diagnostics**
- **Remote Maintenance and Error Reporting**
- **Word Widths Up to 80 Bits Plus ECC**
- **Up to 32 Megabytes Capacity**
- **Cycle Time: 500 ns and 800 ns**
- **Asynchronous, Bidirectional Operation**
- **Power Supply Monitoring and Control**
- **Optional Hand-Held Maintenance Terminal**

The Intel Series 90/iQX is an optional intelligent enhancement to the Series 90 family of random-access memory systems. It is available as System 90/iQX, a high-performance dynamic RAM system; and System 91/iQX, a high-density dynamic RAM system. The enhancement feature, which is based on an Intel 8086 microprocessor, provides uninterrupted system availability, memory tasking, ECC plus soft-error scrubbing, extensive built-in diagnostics, remote maintenance, and error-reporting capabilities. An integral power-supply monitor and control unit monitors, reports, and margins up to four power supplies.

Optionally available for the Series 90/iQX is a hand-held maintenance terminal (the Service Communicator) which allows the operator to communicate with the Series 90/iQX using a versatile set of simple instructions.

The Series 90/iQX can provide up to 32 megabytes of memory in a self-contained system. Two system configurations are available, a horizontal mount system and a vertical mount system.

For full information, refer to Series 90 Data Catalog.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{l}m, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation, 1981.

NOVEMBER 1981
AFN-01742B

FUNCTIONAL DESCRIPTION

The Series 90/iQX system (Figure 1) consists of a Series 90 vertical or horizontal mount memory system. The vertical system provides up to 16 memory modules and up to 32 megabytes of memory; the horizontal system provides up to 4 memory modules and up to 8 megabytes of memory. The iQX module is located between the user and the BXP™ bus; it provides the memory system's intelligence features, such as fault tolerance, diagnostics, and memory tasking. In addition, a power-supply monitor and control unit is included to monitor, report on, and margin up to four power supplies. It will automatically issue a warning if an out-of-tolerance voltage is detected.

The Service Communicator (option) is a hand-held unit by which the operator, through a 40-key alphanumeric keyboard and a 16-character LED display, can communicate with the Series 90/iQX. Its primary function is as a maintenance tool for running diagnostic programs. Approximately 40 commands are available to the operator.

The memory system is operated in an asynchronous, non-interleaved mode on bidirectional lines and performs the five basic operating cycles (read, write, byte write, read-modify-write, and swap). Refresh operations are user-programmed to be either burst, distributed, or distributed bursts. Word widths of up to 80 bits plus 8 ECC bits can be accommodated.

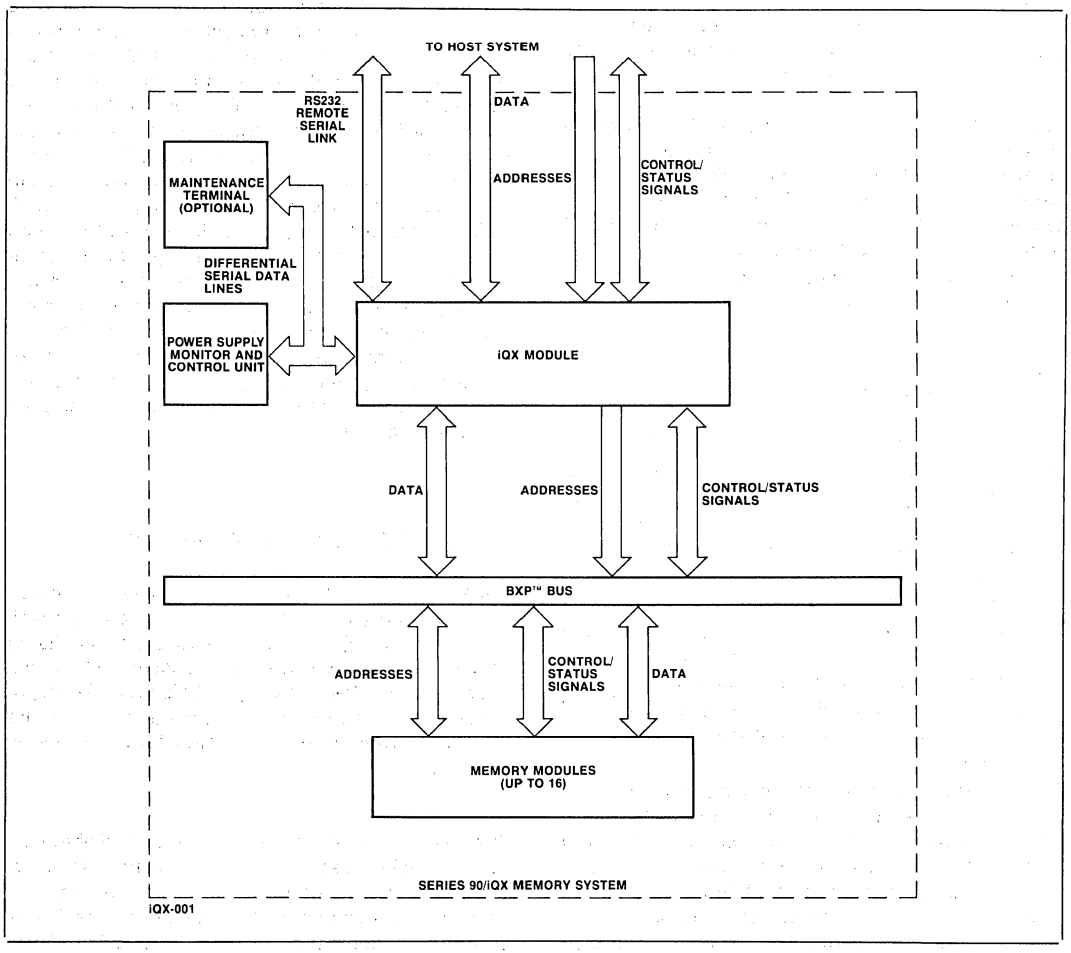


Figure 1. Series 90/iQX Memory System, Block Diagram

RELIABILITY

The Series 90/iQX provides uninterrupted system availability through a comprehensive fault-tolerant mechanism that includes ECC, soft-error scrubbing, and dynamic memory re-allocation.

The error correction circuitry corrects single-bit errors and detects double-bit errors. Soft errors are generally transient single-bit errors. However, cumulative soft errors may lead to uncorrectable double-bit errors. The soft-error scrubber monitors the error detection circuits for single-bit errors. If the errors are soft, they are corrected and restored; if hard, they are handled by the memory re-allocation mechanism, which automatically redirects memory selection away from failed locations to spare memory. The re-allocation activities are totally transparent to the host system. An alert message is sent to the host when the spare memory reserve is low.

With ECC, soft-error scrubbing, and memory re-allocation, data integrity and system reliability are greatly improved. The mean time between failures of the system is extended by effective checks against system degradation over repeated hard-error and soft-error occurrence. By proper spare-memory design, full system availability can be attained between longer service intervals.

MAINTAINABILITY

The Series 90/iQX is designed to reduce maintenance and repair costs over the life of the equipment through a complete package of versatile maintenance tools from error logging and on-board diagnostics to a friendly interactive procedure with the iQX controller. Highlights of the maintenance features are described below:

1. The Service Communicator can be connected to the memory system to read out error counts and locations logged during system operation. The error locations are interpreted by the 8086 and displayed to service personnel by rows and columns.
2. The service communicator can perform diagnostics on the memory system in either the "off-line" or "on-line" modes of the host. The on-board diagnostic capability allows system trouble-shooting without using host time or requiring special test routines on the host. Voltage margin operation for preventive maintenance can also be controlled from the terminal.
3. The 8086-based module also serves as a communications controller that links the system, through a modem, to a remote diagnostic site. The aforementioned on-board diagnostics and fault reporting can all be requested and directed from the remote site, resulting in fewer site visits by service personnel and shortening the visits that are required. This remote capability helps OEM and network services implement centralized maintenance and system control.

MESSAGE-DRIVEN INTERFACE (MAILBOX) AND TASKING

Tasking allows the iQX intelligent memory controller to perform operations on itself or on the data contained in the memory system. To accomplish tasking, a "mailbox" protocol is used. The "mailbox" is a portion of the memory set aside for the iQX/host communication. The host places a command word and parameters in the mailbox and activates an alert signal. The iQX reads the mailbox, executes the command, stores any resulting data in the mailbox, and returns a "task complete" signal. This message-driven (or "mailbox") protocol can support a more extensive exchange of control information with a simple interface. Furthermore, tasking can relieve the host of such chores as moving blocks of memory, filling parts of the memory with constants, setting/resetting memory protect, or executing diagnostic programs and reporting results. All tasking software resides in the intelligent controller module.

CAPACITY AND PERFORMANCE

The total capacity of the Series 90/iQX is determined by the memory module and system package (horizontal or vertical). The capacities available are listed under Specifications, Storage Capacity, in this data sheet. With the VMS the capacity is up to 32 megabytes of dynamic RAM. System capacity can be further increased, up to a total of four Series 90 chassis.

The Series 90/iQX memory system is available with system cycle times of 500 and 800 nanoseconds, depending on memory module type. Refer to the Specifications section of this document, the Series 90 Configuration Guide, or the Series 90 Data Catalog for further information.

CHASSIS

Two chassis configurations are available for the Series 90/iQX: a horizontal mount chassis and a vertical mount chassis. Each system is available with either 115 or 220 VAC input power.

The horizontal mount chassis (HMS) includes up to four memory modules, its own DC power supplies, and a blower assembly to provide cooling air. The chassis measures 5.21 in. (13.2 cm) high by 19 in. (48.2 cm) wide by 19.5 (49.5 cm) deep, and weighs approximately 30 lbs (13.5 Kg).

The vertical mount chassis (VMS) includes up to 16 memory modules and its own DC power supplies. The vertical mount system does not include an integrated blower. A separate blower assembly is available. The vertical chassis measures 12.25 in. (31.1 cm) high by 19.0 in. (48.2 cm) wide by 17.5 in. (44.4 cm) deep, and weighs approximately 100 lbs (45 Kg).

SPECIFICATIONS

Cycle and Access Times

	Cycle Time	Access Time
System 90/iQX	500	500
System 91/iQX	800	800

Word Lengths

16, 32, 37, 64, and 80 bits plus ECC

Storage Capacity

System 90/iQX
- HMS: Up to 1 Mb in 256 Kb increments or up to 4 Mb in 1 Mb increments
- VMS: Up to 4 Mb in 256 Kb increments or up to 16 Mb in 1 Mb increments

System 91/iQX
- HMS: Up to 8 Mb in 2 Mb increments
- VMS: Up to 32 Mb in 2 Mb increments

Operating Cycles

Read Cycle
Write Cycle
Byte Write Cycle
Read-Modify-Write Cycle
Swap Cycle

Environmental Requirements

Ambient Operating Temperature—0°C to +50°C
Ambient Storage Temperature—40°C to +125°C
Relative Humidity—10%to 90% without condensation
Operating Altitude—0 to 10,000 feet (0 to 3,048 meters)
Non-Operating Altitude—0 to 50,000 feet (0 to 15,240 meters)

ORDERING INFORMATION

Consult Series 90 Configuration Guide

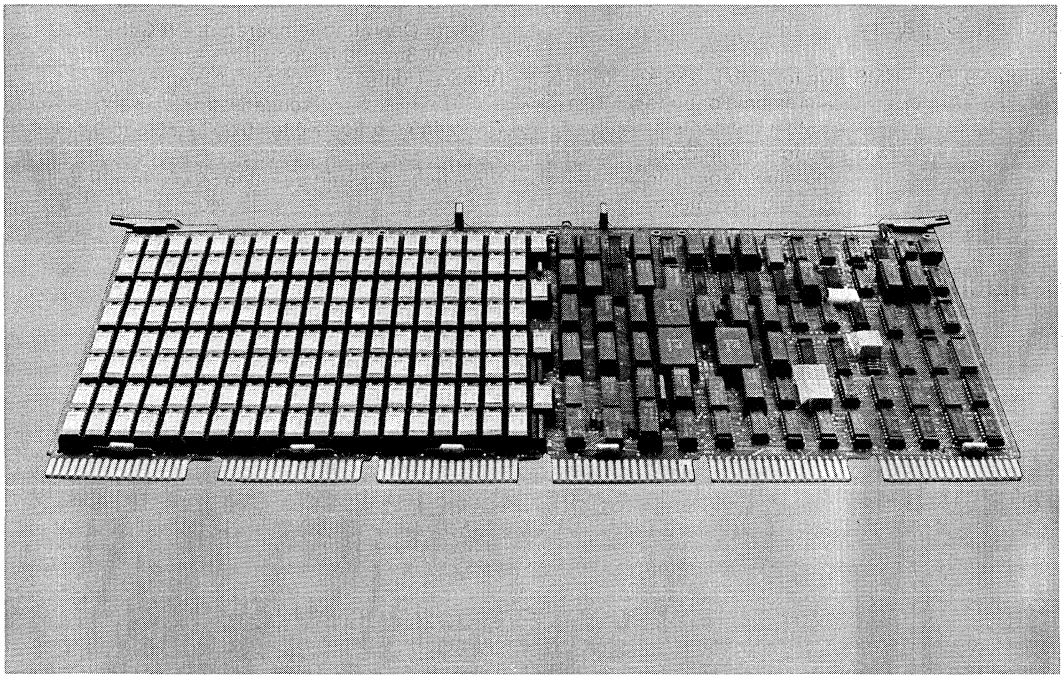
CM-5044E
UNIBUS* ADD-IN MEMORY

- PDP*11/44, PDP11/24 and other UNIBUS computers
- Capacities to 1 megabyte
- ECC version — single bit correct, double bit correct
- Hardware/software compatible
- Switches set addressing on 128K byte boundaries
- Control status register (CSR)
- Sockets spare RAMs
- On-line/off-line switch
- Supports interleaving and battery back-up
- Service available

The Intel® CM-5044E memory card is designed for use in any DEC* UNIBUS, modified UNIBUS, or extended UNIBUS computer. It is an ECC version similar to and compatible with the DEC MS11M memory used on the PDP 11/44. Compatibility extends to hardware, operating system, and diagnostic software. The CM-5044E can also be used as an ECC replacement for the DEC MS11L parity memory used on the PDP 11/24, PDP 11/34 and other UNIBUS computers.

The CM-5044E is available in capacities of 256K, 512K or 1M byte. This gives the user space and power savings while performance is equal to or better than the MS11 models. The card occupies a single slot and addressing is switch selectable to 128K byte boundaries so that the CM-5044E can be used with any number of MS11 memory cards. Features supported include interleaving (switch selectable), battery back-up, and a Control Status Register which allows the processor to read and manipulate status on the memory card. The Intel CM-5044E also contains power, select and error indicators, socketted memory chips and spare memories, and an on-line/off-line switch. All Intel memory cards are fully tested and include a one year factory warranty. In addition, the CM5044E is fully supported by Intel's field service plans which include installation and on-site contracts for maintenance.

*PDP, DEC, UNIBUS are Trademarks of Digital Equipment Corporation



SPECIFICATIONS

Word Size

32 bits + 7 ECC bits

Memory Capacity

256K, 512K, 1024K Bytes

Performance

500 ns cycle (maximum)
500 ns read access (maximum)
250 ns write access (maximum)

Power

Configuration	Operating, Max	Standby, Max
256 KB	7.3 AMPS	5.6 AMPS
512 KB	7.5 AMPS	5.8 AMPS
1 MB	7.9 AMPS	6.2 AMPS

Address Selection

DIP switches select the starting address or any 64K word (128K Byte) boundary.

Modes of Operation

Read	Refresh (Transparent)
Write (full word)	Memory Initialization (Transparent)
Byte Write	Interleaving

Compatibility

PDP 11/04, 11/24, 11/34, 11/44, 11/45, 11/55, 11/60 computers

Interface protocol is compatible with unibus, modified unibus, and extended unibus protocols.

5044 can be used in any system as a replacement for MS11L (parity) or MS11M (ECC) memory modules.

Completely software compatible with all operating systems and CZMSD Diagnostics.

Dimensions

15.7 in (39.88 cm) Width
8.5 in (21.69 cm) Height
0.375 in (0.953 cm) Depth
2.2 lbs (1.0 kg) Weight

Environment

0°C to 50°C Operating
−40°C to 125°C non-operating
0% to 90% relative humidity (no condensation)

Reference Manual

112586 CM-5044E Technical Manual (supplied)

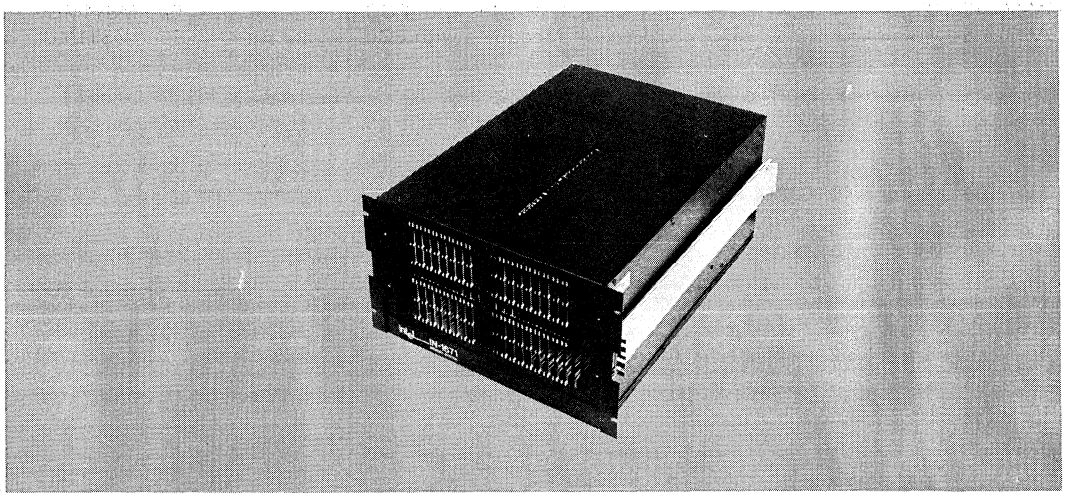
ORDERING INFORMATION

Model Number	Capacity	Equivalent
CM-5044E-256	256 Bytes with ECC	One MS11MB
CM-5044E-512	512K Bytes with ECC	Two MS11MB
CM-5044E-1MB	1 Megabyte with ECC	Four MS11MB

in-1671
*PDP-11/70 ADD-ON MEMORY SYSTEM

- **Total hardware and software compatibility with *PDP-11/70**
- **in-1671 256K-byte basic system expandable to two megabytes in 256K-byte increments**
- **User access to full *PDP-11/70 main memory space at nearly four megabytes**
- **MOS RAMs provide high-density memory with low system cost**
- **Two-way memory system interleaving**
- **ECC and error coding for single-bit error correction and double-bit error detection.**
- **High-speed CPU data transfer rate**
  - **—write cycle time 550 ns**
  - **—byte write cycle time 1000 ns**
  - **—read cycle time 770 ns**
  - **—refresh cycle time 550 ns**
  - **—access time 550 ns (no error)**
  - **—access time 600 ns (with single-bit error)**
- **Installation and maintenance available from Intel**
- **UL recognition**

The Intel® in-1671 *PDP-11/70 Add-On Memory System is a monolithic memory system offering up to two megabytes of add-on memory for *PDP-11/70 users, thus improving *PDP-11/70 system performance with increased capacity, increased data transfer speed, improved functional and component reliability, and reduced cost. The complete in-1671 utilizes five types of printed circuit cards: power control card, memory unit cards, control unit card, data card, and error logging card. All components are fully hardware- and software-compatible with the DEC* central processing unit (CPU) and are engineered to meet or exceed the specifications of similar DEC components. The Intel in-1671 is used in *PDP-11/70 computer systems as a direct replacement for the memory module (MJ11) supplied by DEC, and uses cables and interface signals identical to those used in DEC memory modules. The unit may be installed without change or modification to the DEC software, CPU, memory bus, or I/O structure. High-density memory is provided by dynamic MOS RAM devices, and is expandable in 256K-byte increments to 2048K bytes. High-speed 770-nanosecond read and 550-nanosecond write cycle times allow maximum utilization of Unibus* data throughput. The in-1671 system includes error check and correction (ECC), error monitoring, and error logging.



FUNCTIONAL DESCRIPTION

The in-1671 semiconductor memory system is specifically designed as an add-on memory for the DEC *PDP-11/70 processor. The basic memory device is a monolithic integrated circuit using N-channel MOS transistors. The complete system uses five types of printed circuit cards: 1) control card containing address and control circuitry; 2) data card to provide interfacing and parity checking for read and write data; 3) error logger card to monitor and log data error conditions; 4) memory cards (MU-167As) for storage; and 5) power control card to provide AC and DC low status to the processor. Each memory module contains its own power supply and cooling. All on-line operating sequences are controlled by processor-supplied controller signals and by internally-generated response signals sent to the processor from the memory, and all system components are engineered to meet or exceed the specifications of similar DEC components. A simplified block diagram of the in-1671 memory system is shown in Figure 1.

Compatibility

The in-1671 is specifically designed for use in *PDP-11/70 computer systems as a direct replacement for the DEC-supplied memory module (MJ-11), and uses cables and address, data, and interface control signals identical to those in DEC memory modules. The unit may be installed in the DEC central processing unit (CPU) memory cabinet without change or modification to the DEC software, CPU, memory bus, or I/O structure. All system voltages, currents, timing, and I/O signal requirements are compatible with the *PDP-11/70.

Capacity

Card Capacity—Each memory card in the in-1671 memory system has a capacity of 128K bytes. Two memory cards make up the basic 256K-byte storage area of the in-1671 memory system. A fully-expanded memory system contains two megabytes. Interleaving is possible between two memory systems with the same capacity.

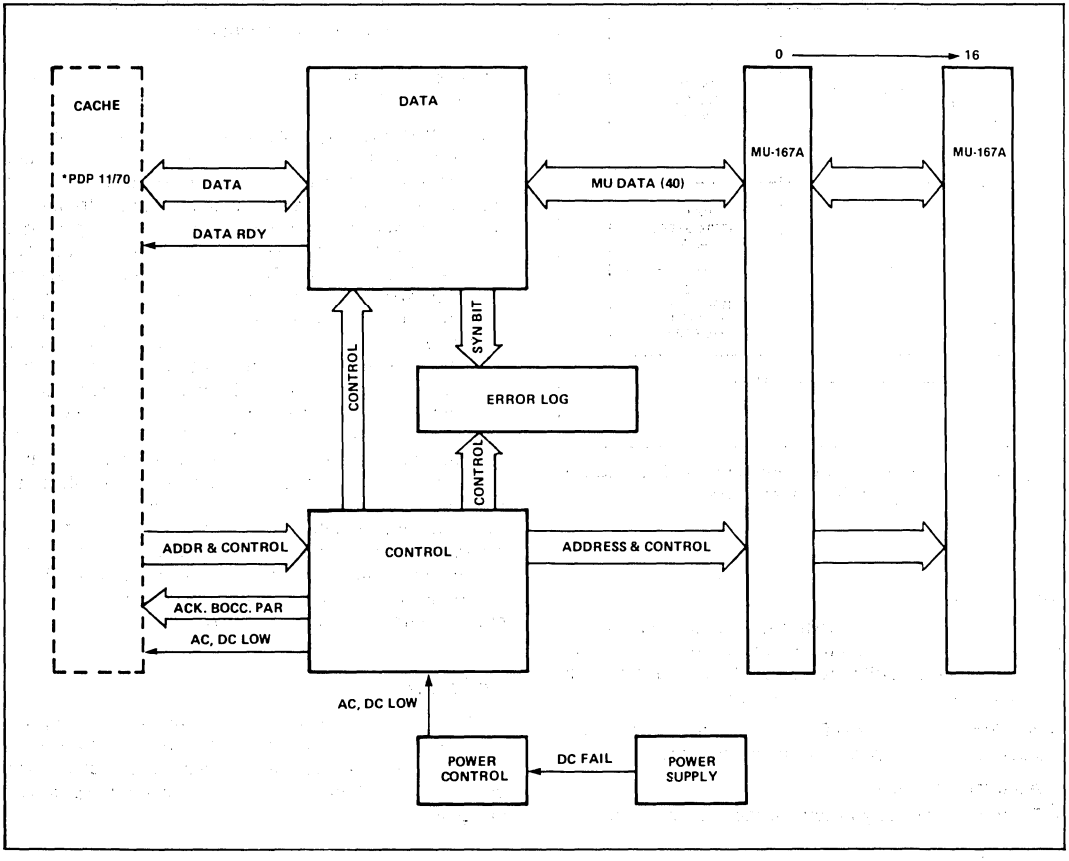


Figure 1. in-1671 Add-On Memory System Block Diagram

System Capacity—The storage area of the in-1671 system consists of 16 Intel MU-167A memory system units. Each MU-167A contains 80 Intel 2117 16K dynamic MOS RAMs to provide a capacity of 64K words by 20 bits per card. The 16 MU-167As in the system are configured to provide 2 megabytes of storage with a double-word 40-bit interface. Thirty-two bits are assigned as interface data and eight bits to error check and correction (ECC). These systems are housed in a chassis with power supplies and fan assemblies. Two such chassis may be mounted in one equipment rack and interfaced in daisy-chain fashion with the processor.

Expandability—Each unit is easily field-upgradable to larger capacities with the addition of memory cards or a memory rack containing cards. A diagram showing CPU and memory cabinet dimensions is shown in Figure 2.

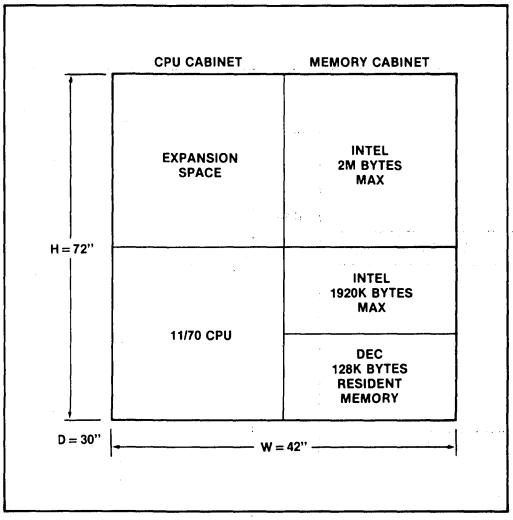


Figure 2. PDP-11/70 CPU Cabinet and in-1671 Add-On Memory Cabinet Dimensions

Maintainability

The in-1671 memory system provides convenient controls, reconfiguration switches, and status indicators as standard maintainability features, all located to provide easy maintenance access to the user.

Rear Panel Controls—The following controls are located on the rear panel of the memory module:

- AC circuit breaker
- Three power supply output voltage adjustments

Card Controls—The following controls are located either on the control card or on the error logger card:

- Address select switches to set the starting address for the memory bank(1)
- Memory on-line/off-line switch to disconnect the in-1671 from the PDP-11/70 memory bus(1)
- Reset logic switch(1)
- Error logger on-line/off-line switch(2)
- Lamp test(2)
- ECC on/off switch(2)
- Error logger scan switch(2)

Card Indicators—The following indicators are provided either on the control card or on the error logger card:

- Address parity error indicator(1)
- Mismatch error indicator(1)
- Write data parity error indicator(1)
- Address display(2)
- Syndrome bits display(2)
- Single-bit mode or double-bit mode indicator(2)
- ECC on-line/off-line mode indicator(2)

(1) on control card
(2) on error logger card

The in-1671 memory system includes error check and correction (ECC) as a standard feature. During read and byte-write operations, multiple bit errors are detected and single-bit errors are corrected prior to the transfer of data. The standard *PDP-11/70 parity bits (one per eight-bit byte) are also generated and are included in the data transfer. The in-1671 ECC improves memory system reliability 10 to 25 times over systems with parity checking only. During write operations, the four standard parity bits are used to check the 32-bit word for errors and then ECC logic removes the parity bits from the data word and generates eight ECC bits to be stored with the data word. During a read operation, the eight ECC bits are read from memory with the 32-bit data word. All 40 bits are processed to generate eight syndrome bits which indicate whether or not a single- or multiple-bit error has occurred. Single-bit errors are automatically corrected before the data is sent to the processor. Multiple-bit errors cause all four standard parity bits to be forced to the error state and to be sent to the processor. The error logger records the location of the row of memory chips in error and which MU-167A board failed.

Reliability

Each unit is fully tested in a temperature-cycling environment. Because of the 100% burn-in performed on each card, the user is assured of receiving Intel's proven quality and reliability. All Intel DSC-compatible products are covered by a one-year warranty.

SPECIFICATIONS

Storage Capacity

Up to 2048 bytes in 256K-byte increments

Word Length

16 bits per memory word, plus 4 parity bits per double word

Performance

Cycle Time — 1000 ns max

Access Time — 600 ns max

Operational Modes

Read
Write
Byte write (1 or 2 bytes)

Interface Characteristics

TTL-compatible
36 bidirectional data input/output lines
23 binary address input lines (single-ended)

Installation Requirements

All cables and connectors supplied by Intel
1 DEC memory cabinet (21"W x 30"D x 72"H)
1 in-1671 memory system

Notes

1. Installation can be done by the customer or purchased from Intel.
2. Maintenance contracts are available from Intel.

Physical Characteristics

Width — 17.12 in. (43.48 cm)
Height — 10.5 in. (26.67 cm)
Depth — 25.0 in. (63.50 cm)
Weight — Less than 70 lb (31.5 kg)

Electrical Characteristics

AC POWER REQUIREMENTS

Requirement	115V AC (max)	220V AC (max)
Voltage	98V AC to 132V AC	187V AC to 252V AC
Frequency	47 to 63 Hz single-phase	47 to 63 Hz single-phase
Input current	8A max	4A max

Environmental Characteristics

Temperature — 0°C to 50°C operating ambient, −40°C to +85°C non-operating

Humidity — 10% to 90% non-condensing

Altitude — 10,000 ft max, operating; 40,000 ft max, non-operating

Equipment Supplied

Ribbon cable supplied for installation

Reference Manuals

TM-1670-000 — in-1670/in-1671 Technical Manual (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative.

ORDERING INFORMATION

SY-1671-256D/I* — 256K bytes (128K words)
Add-On Memory System for PDP-11/70

SY-1671-512D/I — 512K bytes (256K words)
Same as above except for capacity

SY-1671-768D/I — 768K bytes (384K words)
Same as above except for capacity

SY-1671-1024D/I — 1.024M bytes (512K words)
Same as above except for capacity

SY-1671-1280D/I — 1.280M bytes (640K words)
Same as above except for capacity

SY-1671-1536D/I — 1.536M bytes (768K words)
Same as above except for capacity

SY-1671-1792D/I — 1.792M bytes (896K words)
Same as above except for capacity

SY-1671-2048D/I — 2.048M bytes (1.024M words)
Same as above except for capacity

*Specify SY-1671-256D for domestic (115VAC)
SY-1671-256I for international (220VAC)

XF-1671-256	256K bytes expansion for in-1671 PDP-11/70
EX-1671	Card Extender
XX-1671-001	Memory Card
XX-1671-002	Data Interface Card
XX-1671-003	Control Interface Card
XX-1671-004	Error Logger Card
XX-1671-005	Terminator Card
XX-1671-006	Power Control Card
XX-1671-007	I/O Cable Assembly (Short)
XX-1671-008	Power Supply
XX-1671-009	Diagnostic Memory Tape
XX-1671-010	Branch Spare Kit (contains EX-1671 and XXX-1671-001 thru -009)
XX-1671-011	I/O Cable Assembly (Long)

in-5034
PDP*-11/04, 11/34 ADD-IN MEMORY CARD

Low Cost

High Speed

Total Hardware and Software Compatibility

Low Power

High Reliability

Hex-Height Card Size

On-Board Address Select DIP Switches

Spare Memory Chips On-Board

On-Board Parity Checking and Generation

On-Board Control Status Register (CSR)

Battery Backup Provision

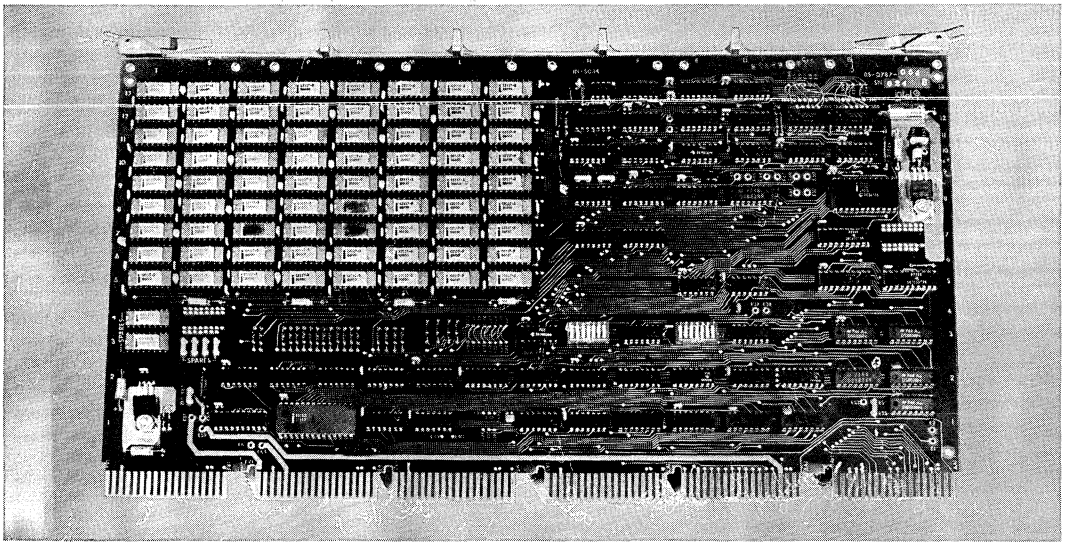
One Year Warranty

The Intel® in-5034 is a 64K x 18 bit plug in memory card designed for use in any PDP-11/04, 11/34 computer memory slot. It is totally hardware and software compatible with the PDP-11 system. No modifications or jumpers to the computer or memory module are required for installation.

The in-5034 memory, with all address and control circuitry, is contained on a single Hex-Height PC card. This memory card requires only one modified Unibus slot.

The in-5034 is designed for low power dissipation. When operating at full speed in a PDP-11/34, or PDP-11/04, the in-5034 dissipates 25.5 watts. In standby, the power dissipation is 12.7 watts. The power dissipation is calculated for worst case maximum. The in-5034, upon detection of power-fail, switches to refresh cycles only. The contents of the memory will be retained provided that there are battery backups on +5VBB, +15VBB and −15VBB voltages.

The in-5034 provides on-board parity checking and generation. Control Status Register (CSR) is also included on-board at no extra cost. Parity check and CSR perform the equivalence of DEC's parity controller modules function for the in-5034 memory.



Two pre-tested memory devices plugged into sockets are provided on board for spare. These spare memory devices can be used to replace any failing memory devices in the field. The failing memory devices can be pinpointed with the aid of the address decode chart and DEC memory diagnostic messages.

The in-5034 features maximum read and write cycle times of 600 nanoseconds alowing maximum utilization of modified Unibus throughput.

With DEC's memory management, up to 128K words of in-5034 memory can be added to the PDP-11 system. The address range for each card is set by on-board DIP switches.

Each memory card is fully tested in a temperature cycling environment. Because of the 100% burn-in performanced on each card, the customer is assured of receiving Intel's provent quality and reliability. In addition, all Intel PDP-11 compatible products are covered by a one year Intel warranty.

PRODUCT SPECIFICATIONS

Storage Capacity:	64K words per board
Word Length:	16-bits plus 2 parity bits
Cycle Time:	600ns
Read Access Time:	410ns
Write Access Time:	320ns
Modes of Operation:	Read Write Read byte Write byte
Address Input:	18 Lines, binary, (Single-ended)
Data Input/Output:	18 Lines, bi-directional, (Single-ended)
Input Control:	3 lines, (Single Ended)
Output Control:	1 line, (Single Ended)
Interface:	Modified UNIBUS compatible, 1 bus load

DC POWER REQUIREMENTS

POWER	OPERATING	STANDBY
System Power		
+5V ± 5%	2 Amps	2 Amps
+15V ± 5%*	0.7 Amps	0.1 Amps
+20V ± 5%**	0.7 Amps	0.1 Amps
−15V ± 5%	30mA	20mA
Battery Backup Power		
+5V ± 5%		500mA
+15V ± 5%*		100mA
+20V ± 5%**		100mA
−15V ± 5%	200mA	

*Used if semiconductor backplane.
**Used if core memory backplane.

Temperature:	0°C to +50°C operating ambient, −40°C to +125°C non-operating
Relative Humidity:	Up to 90% with no condensation
Altitude:	0 to 10,000 feet operating. Up to 50,000 feet non-operating
Dimensions:	15.4" W × 8.5" D × 0.375" H
Weight:	Less than 3 lbs.

OTHER INTEL PDP-11 PRODUCTS

- The in-1671 provides up to 2 megabytes of add-on memory with Error Correction Codes for PDP-11/70 users.

ORDERING INFORMATION

Model	Capacity	Description
CM-5034-864	64K x 18	PDP-11/04 and PDP-11/34 Add-In memory

in-5160
NOVA* 3 ADD-IN MEMORY

Software compatibility with NOVA 3 family

On-board parity control

Optional error correction with on-board error logger and error display

Optional on-board memory management and protection unit

64K, or 128K word capacity and compatible with resident core memory

High circuit density, reducing system cost

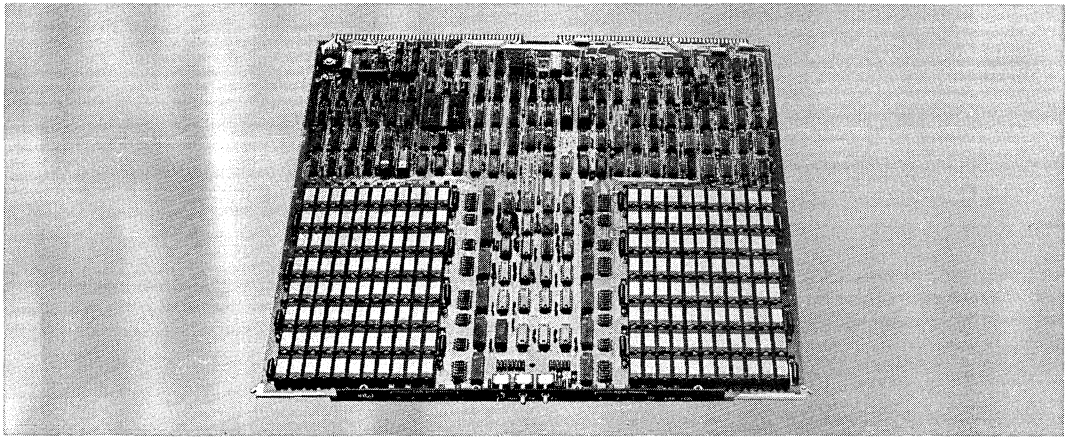
Low power dissipation

Battery backup operation supported

Full commercial operating temperature/ humidity range

One year warranty

The Intel in-5160 is a 64K, or 128K word plug compatible memory card designed for use in the Data General NOVA 3 computer. It is totally hardware and software compatible with the NOVA 3, and configures easily with resident Data General memory boards. All in-5160 control memory, address, parity, bus driving, and voltage regulator circuitry, as well as all options, are contained on one 15-inch x 15-inch printed circuit card, which occupies only one card slot. This high circuit density allows a NOVA 3/4 to be configured with the maximum addressable memory, a memory management unit, the memory protect option, and the parity generate/check option, with two slots left for a disk controller and a serial controller or any other two cards. The in-5160 is designed for low power dissipation and uses a maximum of 50 watts (40 watts typical). In case of power failure the contents of the memory will be retained if a battery backup is provided. With this low power dissipation, no additional cooling fans are required. The parity feature and the optional memory management and protection unit (MMPU) are functionally identical to the corresponding units available from Data General and both features are fully compatible with the NOVA 3 software. The parity feature generates and checks parity on a 16-bit data word and will work not only for the in-5160, but for any other 17-bit memory in the same system. The optional MMPU allows memory addressing and allocation up to 128K words and provides several system protect functions under software control. The optional ECC error check and correction provides error correction facilities on the board along with an error logger. All single bit errors will be corrected; all double bit errors and a significant portion of multiple bit errors are detected.



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 CORPORATION, 1979

*NOVA is a registered trademark of Data General Corporation.

FUNCTIONAL DESCRIPTION

The in-5160 NOVA 3 Add-In Memory is a semiconductor memory system specifically designed as a single card add-in containing the maximum amount of memory addressable by existing software. Also on the board are all memory associated system functions previously available only on a separate card, or not available at all: parity generation and checking, memory management, memory protection, error detection/correction, and error logging with error location display. All normal memory functions are also contained on the card: address buffers and decoding logic, data transceivers, control logic, timing, mode enable logic, and refresh control logic. On-line operating sequences are controlled by processor supplied control signals, signals received from data channel devices and by internal control signals. Response signals sent to the central processing unit (CPU) are generated by the parity logic, MMPU, and other control logic circuitry. All system components are engineered to meet or exceed the specifications of similar Data General components.

Compatibility

The in-5160 can be inserted into any NOVA 3 series computer and will work with any 17-bit memory already in the system. The 96kW version must be positioned in lower memory. Other capacities may start on any 32kW boundary. If the previous memory capacity is not a multiple of 32K words, the in-5160 is set to occupy the address range from zero up to the in-5160 card capacity. The previously installed memory is then set to follow the ending address of the in-5160. For example, a 64K word in-5160 could be used as shown in Table 1.

Condition	Address Range for in-5160	Address Range for Resident NOVA Memory
No previous memory	0-64KW	—
16K resident memory	0-64KW	64KW-80KW
32K resident memory	32KW-96KW	0-32KW
64K resident memory	64KW-128KW	0-64KW

Table 1. in-5160 Memory Address Range

Capacity

The in-5160 is manufactured with two types of memory devices: a 16K x 1-bit dynamic RAM and an 8K x 1-bit dynamic RAM. Both chips are manufactured with NMOS technology and are tested to the same specifications.

Addressability

Card select for the in-5160 is determined by the decode of the three extended address inputs and the settings of the address select switches on the card. These switches enable the card's starting and ending addresses to be set on any 32K word boundaries within the 128K word address field available with memory management. For cards with memory capacity greater than 32K words, the address range must be selected so that the memory occupies contiguous 32K address blocks.

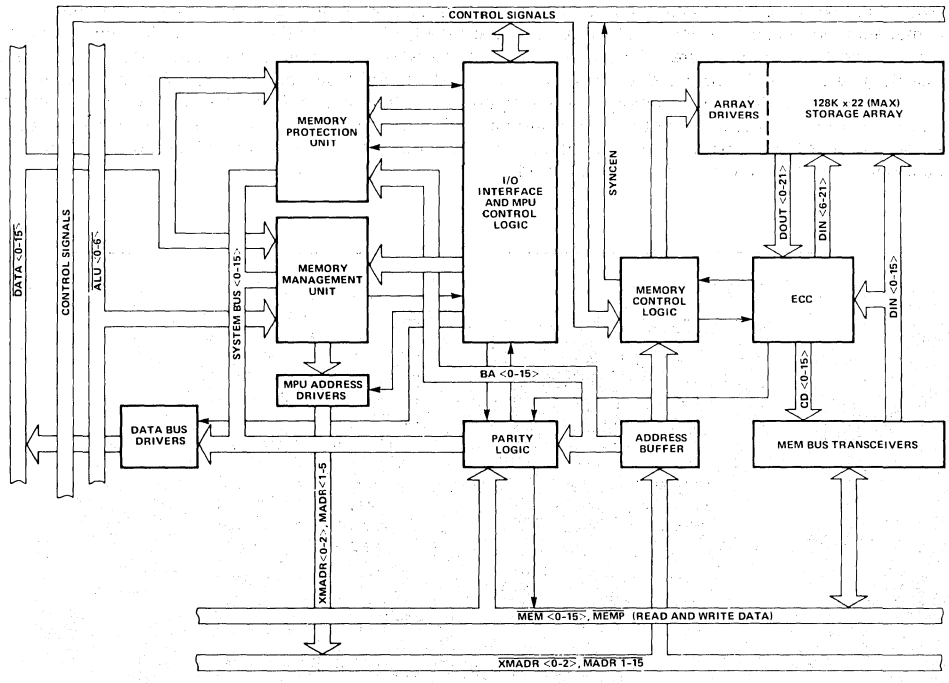


Figure 1. in-5160 NOVA-3 Add-In Memory Simplified Block Diagram

Operating Modes

The in-5160 emulates all of the operating modes supported by Data General on the NOVA 3 series computers. In normal operation, the in-5160 performs read, read-hold, write, and read-modify-write processor controlled cycles. Refresh cycles are timed out internally and use chip row addresses generated by a counter on the board. The in-5160 offers the largest storage capacity that can be handled by existing software and has a higher effective speed than the NOVA 3 semiconductor memories. Read or write operations required 500 nanoseconds and the read-modify-write operation is completed in 1100 nanoseconds, whether initiated when the memory is idle or immediately following a preceding cycle. On consecutive memory cycles, the Data General NOVA 3 memory requires 500 nanoseconds for read cycles, 1200 nanoseconds for write cycles and 2200 nanoseconds for read-modify-write. Since consecutive memory cycles occur frequently, the Intel memory can provide significantly faster throughput. The data valid time is extended during the read-hold cycle by the processor pulling the hold input low. The longer data valid time is required during defer cycles, console initiated read cycles (the examine switch), or during data channel read cycles. For data channel memory cycles, the memory operates in either write or read-hold modes. All read, write, or read-modify-write cycles can be performed with or without single bit error correction, map, memory protection, or parity error reset or interrupt.

Memory Management and Protection Unit (MMPU)

Memory expansion — The MMPU provides the capability to use a memory of up to 128K words and supplies the hardware necessary to make use of the NOVA 3 software protection features. The expansion of memory capacity is accomplished by replacing the five highest order address bits (the logical address) with seven bits from a location in the translation memory (the physical address). Each logical address is 'mapped' (or translated) to one, and only one, physical address, which may be anywhere within the 128K physical address field. The contents of the translation memory (the map) may be set up using ordinary I/O instructions.

Memory protection — The memory protection function of the MMPU provides detection capability for five types of software violations:

- **Write protect** — prevents writing into a write protected 1K page
- **Validity** — prevents accessing, in any manner, a page declared invalid
- **Defer** — protects against runaway defer cycles
- **I/O** — enables I/O violation detection in the CPU
- **Auto Index** — Prevents auto-indexing operations.

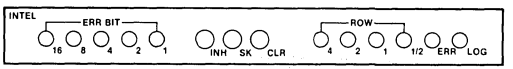
When a violation is detected, the MMPU initiates a map interrupt routine in the processor.

Parity

The parity feature is standard on all in-5160 memory boards. During write operations, the parity logic generates a parity bit for the 16-bit data word. Data and parity bit are then stored in the memory. During read operations, 16 bits of data and one parity bit are read out from the memory and checked for a parity error. If a parity error is detected, the address and parity bit are latched in the parity error address register, and either a reset or interrupt signal (jumper selectable) is returned to the processor. The CPU reads the contents of the parity error address register by using standard I/O instructions. The parity feature services not only the in-5160, but all 17-bit memories in the same system. On boards with the ECC option, the parity logic still functions for other 17-bit memories in the system and reports multiple bit errors detected by the ECC logic to the CPU as parity errors.

Error Correction and Error Logger

The in-5160 error correction circuitry (ECC) provides onboard capability to correct any single bit error and detect any double bit error in the read data. Many errors of more than 2 bits will also be detected. ECC greatly increases memory reliability and minimizes non-recoverable system failures. The multiple bit errors are flagged to the processor as parity errors. All errors are logged as they occur and are simultaneously displayed on the in-5160's control panel. The error correction function can be disabled by a switch on the control panel; however, the error logger will continue to monitor and store error information. The chip location of single bit errors and row location of double bit and multiple bit errors are stored in the error logger memory and displayed on LEDs on the control panel as they occur. The error logger controls and indicators are shown in Figure 2.



ERR BIT — THESE INDICATORS SHOW FAULTY DATA BIT (BINARY CODED).
16,8,4,2,1 *EXAMPLE:* INDICATORS 1 AND 4 ON, INDICATES BIT 5 IN ERROR.

SK (SEEK) — PUSHBUTTON SWITCH. WHEN PUSHED AND HELD, PERFORMS LED TEST. WHEN RELEASED, ERROR LOGGING MEMORY IS SEARCHED FOR ERROR INFORMATION.

CLR (CLEAR) — PUSHBUTTON SWITCH. CLEARS ERROR LOGGING MEMORY.

INH (INHIBIT) — TOGGLE SWITCH. IN ON POSITION INHIBITS ERROR CORRECTION FUNCTION. DOES NOT INHIBIT ERROR LOGGER.

ROW — THESE INDICATORS SHOW ROW LOCATION OF FAULTY RAM
4,2,1,½ (BINARY CODED).

LOG — THIS INDICATOR COMES ON TO INDICATE ERROR LOGGING IN PROGRESS.

ERR — INDICATES ERROR IS DETECTED; SINGLE BIT OR DOUBLE BIT/MULTIPLE BIT

Figure 2. Error Logger Control Panel and Indicators

INSTALLATION

The installation of the in-5160 on a NOVA 3 system is covered in detail by the technical manual which accompanies the product. The main points of consideration during installation are address selection and parity error action (RESET or interrupt). The location of these switches and jumpers are illustrated in the manual for easy location.

RELIABILITY

The in-5160 is designed for low power dissipation. When operating with all options, and at full speed in a NOVA 3, the 128K in-5160 dissipates 170 Btu/hr; in battery backup (standby) mode, it dissipates 55 Btu/hr. This results in a cooler running system, an increased margin for power supply operation, and improved system reliability. Every card is fully tested in a temperature cycling environment for eight hours and in a NOVA 3 computer to test all options with the NOVA software. Because of the 100% burn-in performed on each card, the user is assured of receiving Intel's proven quality and reliability. In addition, all Data General compatible products manufactued by Intel are covered by a one year warranty.

SPECIFICATIONS

Storage Capacity

64K, or 128K words

Word Length

Standard — 16 data bits and one parity bit
With ECC — 16 data bits and six ECC check bits

Cycle Time

Read or Write Cycle — 500 ns
RMW — 1100 ns

Access Time

300 ns (access and cycle times are extended 100 ns if error correction takes place)

Operating Modes for Memory

Processor — Read, read-hold, write, read-modify-write
Data Channel — Read-hold, write
MMPU — Translation enabled/disabled
Parity — Disabled/generate even parity, generate odd parity

Interface Characteristics

Interface — NOVA-3 backplane compatible, one card slot required

Note: If the in-5160 is equipped with MMPU option, backplane slot 2 is required.

Address Input — MADR 1-15: 15 Lines; ALU 0-6: 7 lines; DS 0-5: 6 lines

Address Input/Output —MADR 1-15
(Generated in MMPU) X MADR 0-2

Memory Data Input/Output — 16 bidirectional lines
I/O Data Input/Output — 16 bidirectional lines
Control Inputs — 36 lines
Control Outputs — 14 lines

Physical Characteristics

Width — 15 in. (38.1 cm)
Length — 15 in. (38.1 cm)
Height — 0.4 in. (1.02 cm)
Weight — Less than 3 lb (1.36 kg)

Electrical Characteristics

DC Power Requirements (128K, all options)

Voltage (±5%)	Active	Current Battery Backup
+5V	6.8A	2.2A
+15V	890mA	350mA
−5V	5mA	

Environmental Characteristics

Temperature — 0 °C to 50 °C operating ambient; −40 °C to 120 °C non-operating ambient

Relative Humidity — Up to 90% with no condensation

Altitude — 10,000 ft max operating; 50,000 ft max non-operating

Reference Manuals

TM-5160-000 — in-5160 Technical Manual, Number 111768 (SUPPLIED)

Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative.

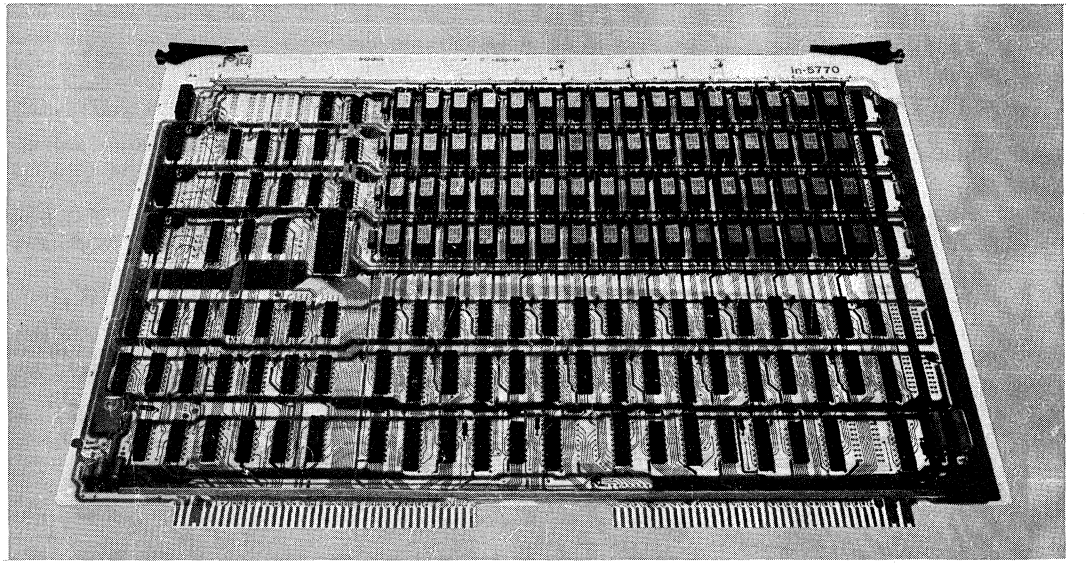
ORDERING INFORMATION

Model	Capacity	Description
CM-5160A-064	64K x 17-bit	64K memory with parity
CM-5160A-128	128K x 17-bit	128K memory with parity
CM-5160B-064	64K x 22-bit	64K memory with ECC
CM-5160B-128	128K x 22-bit	128K memory with parity and ECC
CM-5160C-064	64K x 22-bit	64K memory with parity and MMPU
CM-5160C-128	128K x 22-bit	128K memory with parity and MMPU
CM-5160D-0364	64K x 22-bit	64K memory with parity, ECC and MMPU
CM-5160D-128	128K x 22-bit	128K memory with parity, ECC and MMPU

in-5770
VIDEO-REFRESH MEMORY SYSTEM

- **Single-card memory system complete with addressing and timing logic**
- **256K-word x 4-bit capacity in four image planes, each 256K x 1 bit wide**
- **Four-bit resolution, allowing 16 shades of grey for image enhancement**
- **Expandable in multiple card systems for greater resolution in both grey scale and multicolor displays**
- **Utilizes MOS dynamic RAMs for maximum bit density at low system cost**
- **Operates in both parallel- and single-bit mode control**
- **Provides automatic refresh or externally initiated refresh**
- **High-speed 14.3 MHz bit rate in serial read mode**

The Intel in-5770 Video-Refresh Memory System is a special-purpose, single-card, random-access memory system designed to store and retrieve digital video image data for sophisticated computer-driven CRT displays. The storage capacity of each card is 256K (K = 1024) 4-bit words arranged in four image planes to provide the 16 grey shades necessary to ensure the high picture quality required for medical and scientific laboratory computer modeling displays; a typical image enhancement application is the detailed analysis and interpretation of black-and-white x-ray images. Multiple in-5770 cards may be used in parallel for systems applications requiring more than four bits per picture element for special graphic displays such as color enhancement, overlays, or enlargements of portions of the display. The in-5770 refresh memory is specifically designed for image enhancement applications; refresh may be accomplished either by normal read and write cycles or by user-generated refresh cycles within the specified retention time. Each memory card contains all the logic and timing circuitry required to generate memory addresses and clock pulses, and utilizes 16K x 1 MOS dynamic RAM technology for maximum bit density at low system cost. An optional chassis is available for multiple card housing.



FUNCTIONAL DESCRIPTION

The in-5770 is a video-refresh, random-access memory card system designed specifically for storing and retrieving digital video image data with high resolution, but also used for conventional storage applications. The complete system consists of one 16 x 11.25-inch edge connector–type printed circuit card containing the semiconductor storage area and all the required address and data latches and control logic needed to operate the memory. The two standard configurations are 512 x 512 using Intel's 16K x 1 dynamic MOS RAM and 512 x 256 using the 8K x 1 dynamic MOS RAM. The memory is arranged in four planes either 16K x 16 or 8K x 16. A block diagram of the in-5770 memory system is shown in Figure 1.

Capacity

Each CM-5770 memory card has the capacity to store a 512 x 512 display with four bits per picture element. The two standard configurations are 512 x 512 x 4 (CM-5770-512) and 512 x 256 x 4 (CM-5770-256). The memory storage area in the 512 x 512 version consists of 64 Intel 16K x 1 dynamic N-channel MOS RAM silicon-gate memory chips. The 16,384 storage cells on each chip are arranged in a 128 x 128 array requiring seven row and seven column addresses to select one cell. Each cell holds one bit of information. The memory chips are arranged in four rows of 16 chips each, with each 4 x 4 block of chips representing one 16K x 16 plane of data. The memory storage area in the 512 x 256 version uses 64 Intel 8K

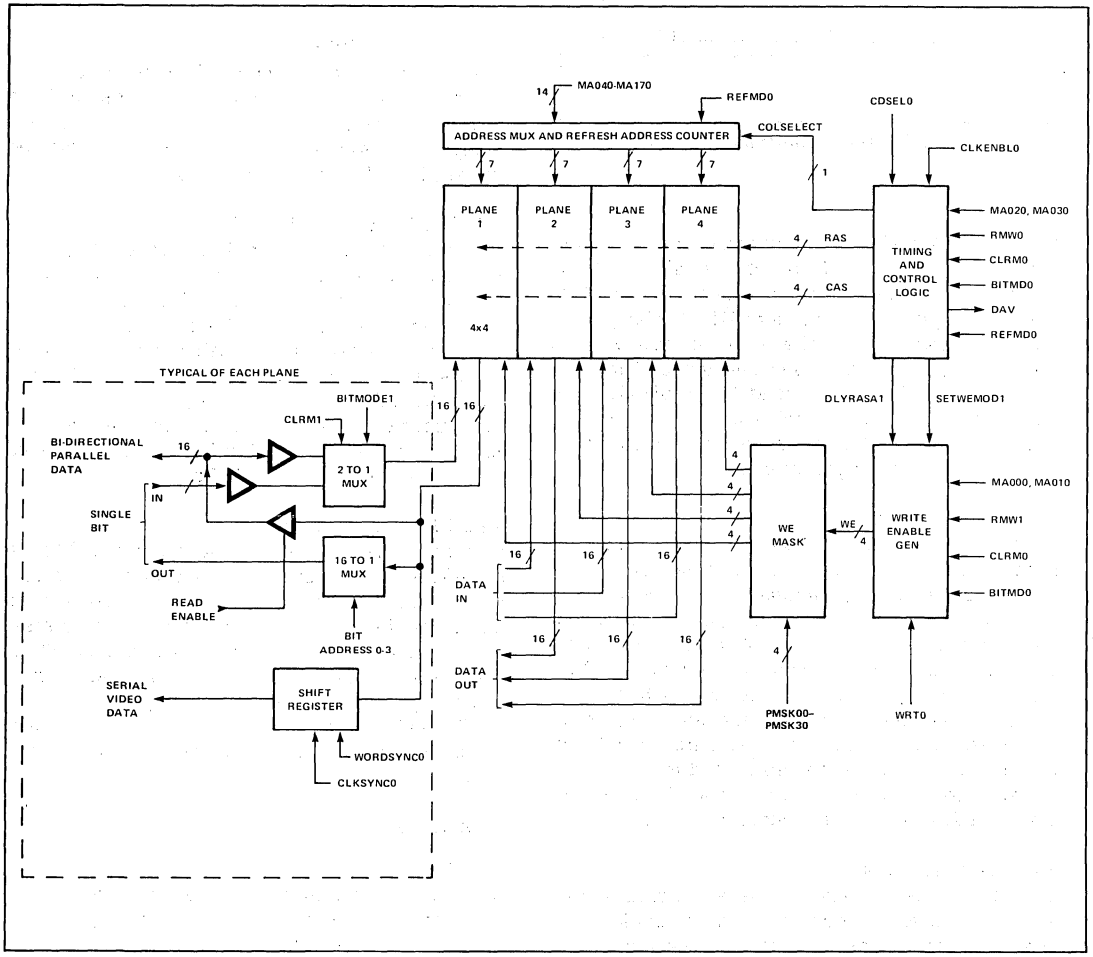


Figure 1. in-5770 Video Refresh Memory System Block Diagram

x 1 dynamic N-channel MOS RAM memory chips with 8192 storage cells arranged as a 64 x 128 array. The 8K RAMs are partial 16K devices with only one half of the array operational. The in-5770 memory capacity may be expanded using up to a maximum of 24 CM-5770 cards per system.

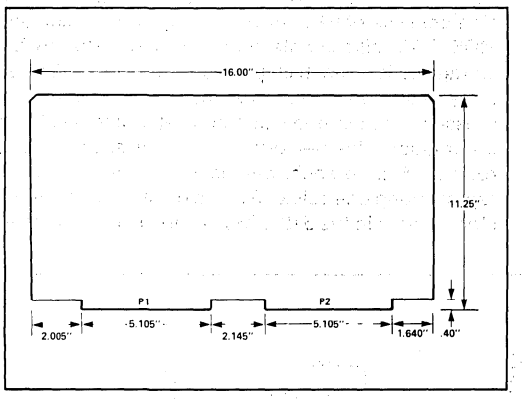


Figure 2. in-5770 Card Dimensions

Operation

The in-5770 provides both parallel- and single-bit mode control to perform four basic memory operations plus the refresh operation. The card is read in a sequential manner for CRT refreshing, with random-access read, write, or read-modify-write operations performed in between the CRT refresh reads. During a CRT refresh read, 16 bits of data from each plane are read on the memory card and are internally converted to serial bit lines, clocked at 70-nanosecond or longer intervals.

READ OPERATION

In all read modes, a full 16-bit data word is read out from the addressed location in each memory plane. During parallel read operations, a 16-bit data word is read from each plane of the memory array and placed on the output lines, 64 lines in all, resulting in 64 bits being simultaneously available at the interface. During serial read operations, 16 bits from each memory plane are loaded into shift registers and transferred out serially on four output lines. During single-bit read operations, a 16-bit word is read from each of the four memory planes, but only one bit from each plane is transferred to the single-bit output lines.

WRITE OPERATION

In write modes, 64 bits of data are stored in the location specified by the address inputs. During parallel write operations, 64 bits are received at the interface, and stored as four 16-bit words, one in each memory plane. The parallel write mode writes a 16-bit data word selectively in any or all of the planes, under control of a plane select mask. During single-bit write operations, four bits (one pixel) are received at the interface and stored with one bit in each memory plane.

READ-MODIFY-WRITE OPERATION

In a read-modify-write operation, internal memory timing is arranged so that read data is output from the card during the first portion of the cycle, followed by a 90 ns modify period during which a new data word is generated by the user and stored in the memory. During a parallel read-modify-write operation, a full 16-bit word is read and rewritten in each memory plane. During a single bit read-modify-write operation, a single-bit is read and rewritten into each memory plane.

CLEAR OPERATION

A clear mode provides a convenient way for clearing the screen and for generating test patterns. During a clear mode operation, the data applied on the single-bit input lines is written into all 16 bits of the selected address in all four memory planes.

VIDEO REFRESH OPERATION

In image enhancement applications, photographic or video images are converted to a matrix of picture elements with binary elements designating the intensity assigned to each. This information is sent from the computer to the refresh memory and stored for later CRT display. Since data sent directly to a screen from the computer quickly fades, the in-5770 continuously recreates stored graphic images as required by projecting a 512 x 512 matrix of shaded picture elements onto the CRT screen.

MEMORY REFRESH OPTIONS

All rows of memory devices within the in-5770 must be refreshed once every 2 milliseconds to prevent loss of data. The refresh operation takes place automatically during serial read for video refresh. However, if normal read operations do not use all 128 chip rows within the memory, refresh cycles must be externally initiated. Data refreshing in N-Channel MOS RAMs is normally achieved by sequentially scanning the memory at the rate of 128 times each 2 milliseconds.

Interface

INPUT/OUTPUT

All in-5770 input/output signals are compatible with TTL integrated circuits. The input and output signal operations for the in-5770 are summarized in Tables 2 and 3, respectively.

Table 2. in-5770 Input Signal Operations

Signal	Operation
Addresses (0–17)	Controls address selection of both single bit data and parallel data on 18 bidirectional lines.
Card select	Enables memory activity on card when low.
Memory cycle start	Initiates memory cycle upon going from high to low.
Bit mode	Indicates to memory, when low, that single bit write cycle or read-modify-write cycle is to be performed. Enables parallel operation when high.
Write	Initiates write operation when held low at beginning of memory cycle.
Read-modify-write	Initiates read operation when held low at beginning of memory cycle. Initiates write operation after 90 ns delay. Writes user supplied data into location initially read out.
Plane select write mask	Inhibits writing operation, when driven low during write or read-modify-write operation on data within given planes, on all data within that plane.
Clear mode	Initiates write operation, when held low at beginning of cycle, of data on single bit write data lines into all 16 bits of addressed word.
Parallel read enable	Enables, when low, readout of data onto 64 bidirectional lines.
Video data load sync	Enables, when low, parallel loading of shift registers for serial transfer out of video data during serial read operations.
Video data serial clock sync	Clocks shift registers, upon going from low to high, for serial transfer out of video data during serial read operations.
Refresh mode	Initiates refresh cycle, when held low at beginning of memory cycle, on address location specified by internal refresh counter.
Single bit write data (0–3)	Transfers single bit write data to memory on four bidirectional lines, 16 for each plane.
Parallel memory data (0–15)	Transfers parallel data to and from memory on 64 bidirectional lines, 16 for each plane.

Reliability

Intel manufactures all the integrated circuit devices from which all Intel memory systems are produced. Because of this vertical integration, Intel controls the quality of its equipment from the beginning process of making the device through the final test and delivery of the system, thus assuring the users of products with proven quality and reliability. In addition all Intel memory systems products are covered by a one year Intel warranty.

Table 3. in-5770 Output Signal Operations

Signal	Operation
Data available	Indicates to processor, upon going high at access time, that valid read data is available on data bus.
Single bit data (0–3)	Transfer single bit read data out of memory on four unidirectional lines.
Serial video data (0–3)	Transfers serial video data out of memory on four unidirectional lines.

SPECIFICATIONS

Storage Capacity

256K × 4 bits, addressable as 64-bit words (16 bits per image plane), or as 4-bit words (1 bit per image plane)

Word Length

16 bits on the parallel bus per image plane
1 bit on the single bit bus per image plane
1 bit on the serial data bus per image plane

Performance

CYCLE TIME
Read, Write, or Refresh: 450 ns max
Read-Modify-Write: 780 ns max

ACCESS TIME
Parallel Data: 380 ns max
Single Bit Data: 390 ns max

RETENTION TIME
2 ms max

SERIAL DATA RATE
14.3 MHz max

Operational Modes

Parallel write (16 bits)
Parallel read-modify-write (16 bits)
Single bit write
Single bit read-modify-write
Parallel read (16 bits)
Single bit read
Serial read
Clear mode write
Memory refresh

Interface Characteristics

CONNECTOR
Two 80-pin double-sided PC edge, 0.125 in. centers, type SAE 8100 (Stanford Applied Engineering) or equivalent

INPUT/OUTPUT
TTL compatible (single ended)

ADDRESS INPUT
18 binary lines

DATA INPUT
Parallel Data: 64 bidirectional lines
Single Bit Data: 4 unidirectional lines

DATA OUTPUT
Parallel Data: 64 bidirectional lines
Single Bit Data: 4 unidirectional lines
Serial Mode Data: 4 unidirectional lines

CONTROL INPUT
17 lines: memory cycle start, write, single bit mode, read-modify-write, card select, refresh, plane select write masks 0 through 3, clear mode, parallel read enable 0 through 3, serial data clock, serial data load enable, video data load sync

CONTROL OUTPUT
Data available

Interface Signals

INPUT
Low ..... −1.0V to +0.8V @ 2 mA
High ..... +2.2V to +5.5V @ 100 μA

OUTPUT
Low ..... −0.5V to +0.5V @ 15 mA
High ..... +2.4V to +5.2V @ 200 μA

Physical Characteristics

CARD
Width ..... 11.25 in. (28.58 cm)
Length ..... 16 in. (40.64 cm)
Weight ..... 2 lb (1 kg)
Mounting Centers ..... 0.625 in. (1.59 cm)

Electrical Characteristics

DC POWER REQUIREMENTS

Voltage (±5%)	Current (max)
+12V	2.5A
+5V	2.75A
−5V	100 mA

Environmental Requirements

TEMPERATURE
0°C to 55°C operating ambient
−40°C to 125°C non-operating

RELATIVE HUMIDITY
Up to 90% non-condensing

ALTITUDE
10,000 ft max, operating
40,000 ft max, non-operating

COOLING
200 linear ft per minute

ORDERING INFORMATION

Model	Capacity	Description
CM-5770-512	256K × 4	512 × 512 × 4 video refresh memory system card

PRELIMINARY

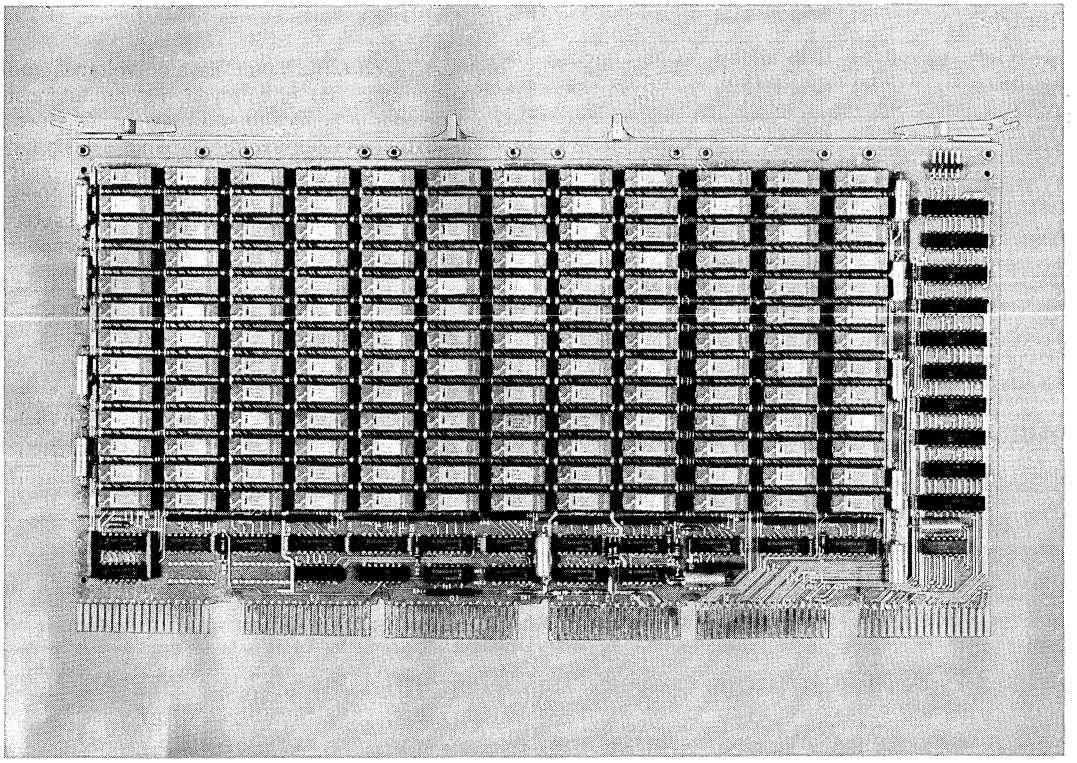
MU-5750
VAX* 11/750, PDP* 11/70 ADD-IN MEMORY

- **Replaces DEC* MS750AA or MS750AB**
- **Hardware/software compatible**
- **256K or 512K byte capacity**
- **Supports battery backup**
- **On-line/off-line switch**

The Intel® MU-5750 is an add-in memory board for use in the DEC* VAX* 11-750 or PDP* 11/70 computers and is functionally equivalent to DEC's own memory board. The VAX or PDP 11/70 memory subsystem provides all of the necessary control and timing for this memory board. No modification to the computer is necessary to use the MU-5750. An on-line/off-line switch is provided on-board to allow memory boards to be isolated for diagnostic purposes without being removed.

The 256KB MU-5750 is a direct replacement for one DEC M8728 memory board (MS750AA) and the 512KB MU-5750 replaces two M8728 boards (MS750AB). Up to eight 256KB boards or up to four 512KB boards can be used to expand the VAX 11/750 memory capacity up to a full 2M bytes. Similarly, the MU-5750 can be used to expand the PDP 11/70 to its full capacity of 4M bytes.

*DEC, PDP, and VAX are trademarks of Digital Equipment Corporation, Maynard, Massachusetts.



The following are trademarks of Intel Corporation and its affiliates and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{l}m, iCS, ICE, UPI, BXP, iSBC, iSBX, Insite, iRMX, System 2000, CREDIT, iRMX/80, MULTIBUS, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTIMODULE, and the combination of MCS, ICE, iSBC, iRMX or iCS and a numerical suffix. © INTEL CORPORATION, 1981.

JULY 1981
AFN-01988A

Storage Capacity

256K or 512K Eight-Bit Bytes

Word Length

32 Data Bits Plus 7 Error Check and Correction (ECC) Bits

Speed

Access Time ..... 285 Nanoseconds
Cycle Time ..... 350 Nanoseconds

Operational Cycles

Read, Write, Byte Write, Initialize, Refresh

Interface Characteristics

—TTL Compatible
—7 Chip Row and Column Address Lines
—13 Control Lines

Electrical

Model (KB)	Voltage (Volts)	Standby Current (mA)	Operating Current (mA)
256	+12	234.00	1540.50
	+5	2.60	1.59
	−5	7.80	13.70
512	+12	468.00	1774.50
	+12	3.10	3.12
	−5	15.60	25.35

Dimensions

Length ..... 8.0 in. (20.3 cm)
Height ..... 15.69 in. (39.0 cm)
Width ..... (256KB) One Card Slot
(512KB) Two Card Slots

Environment

TEMPERATURE

0°C to 50°C ..... Operating
−40°C to 125°C ..... Non-Operating

HUMIDITY

Up to 90% Without Condensation

ALTITUDE

Up to 10,000 ft. (3,030m) ..... Operating
Up to 50,000 ft. (15,150m) ..... Non-Operating

RELIABILITY

Intel tests each MU-5750 at elevated voltages and temperatures for a period proven to bring out most latent defects in semiconductor devices. This process, together with temperature cycling, is designed to remove all potential failures before the memory card reaches the customer. Because of the 100% burn-in performed on each card, the user is assured of Intel's proven quality and reliability. This product is covered by a one-year warranty.

ORDERING INFORMATION

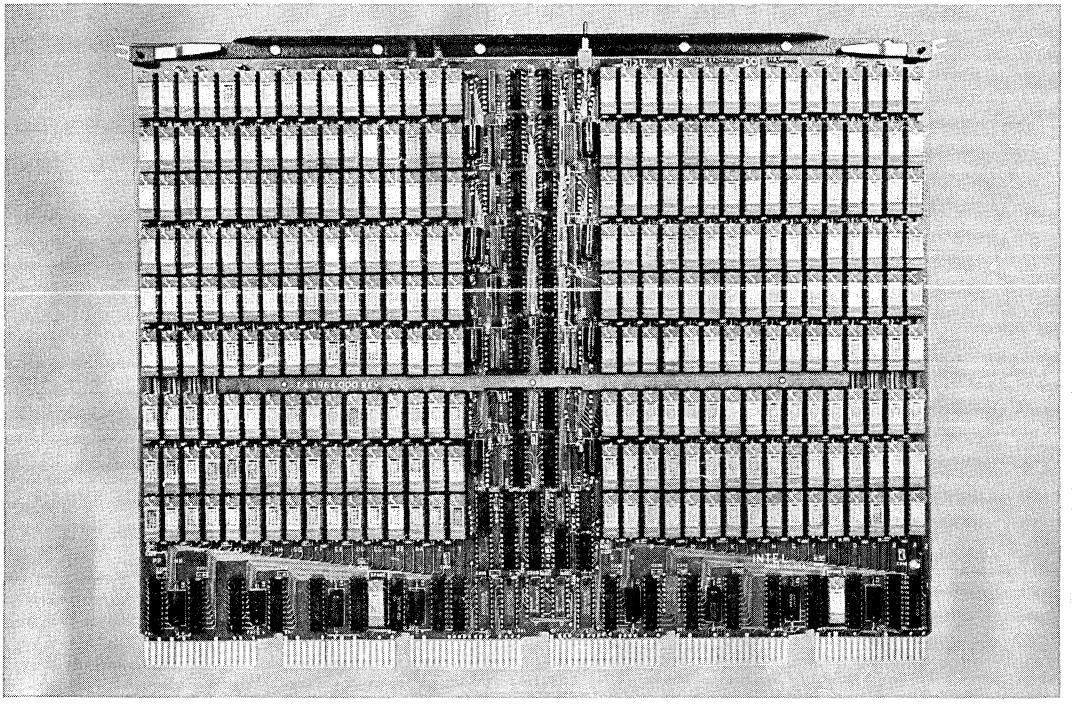
Model	Capacity	Equivalent VAX 11/750	Equivalent PDP 11/70
MU-5750-256	256K eight-bit bytes	MS750-AA (one M8728)	— (one M8728)
MU-5750-512	512K eight-bit bytes	MS750-AB (two M8728)	MK11-CE (two M8728)

MU-5780
VAX*-11/780 ADD-IN MEMORY CARD

- **Replaces DEC* MS780-DA, DB or DC**
- **256KB, 512KB or 1024KB Capacity**
- **On-Line/Off-Line Switch**
- **Equivalent to 1, 2 or 4 DEC M8210 Memory Modules**
- **Complete Hardware and Software Compatibility**
- **Supports Battery Back Up**

The Intel® MU-5780 Add-In Memory Card is a 256KB, 512KB or 1024KB plug-compatible memory card designed for use in any DEC VAX-11/780 computer, to expand the MS780-C or MS780-CC basic VAX-11/780 memories up to 4096K bytes (four megabytes). The MU-5780 operates at speeds controlled by the VAX-11/780 memory controller. Typical read cycle and access times are 530 and 250 nanoseconds respectively. All card components are hardware- and software-compatible with the VAX-11/780 computer. The MU-5780 can be installed without change or modification to either the computer or the memory module by plugging it into the proper memory backplane slot. The MU-5780 measures 39.8 cm (15.7 inches) by 31.4 cm(12.4 inches) and requires the same +5VDC, −5VDC, +12VDC, and +5VDC battery backup as the memory card it replaces.

*VAX and DEC are registered trademarks of Digital Equipment Corporation.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{i}m, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

FUNCTIONAL DESCRIPTION

The MU-5780 Add-In Memory Card consists of a single printed-circuit card designed to fit into the memory backplane of a DEC VAX-11/780. It contains all circuitry required for memory operation, including address-decoding circuits, memory-control circuits, and a memory storage area. All system components are engineered to meet or exceed the specifications of similar DEC components. A block diagram of the MU-5780 memory system is shown in Figure 1.

Compatibility

The MU-5780 can be operated with or in place of the DEC Model M8210 Semiconductor Array. It must be used in conjunction with the DEC M8213 Memory Controller, the M8212 Memory Data Path, and the M8214 Memory SBI Interface Boards.

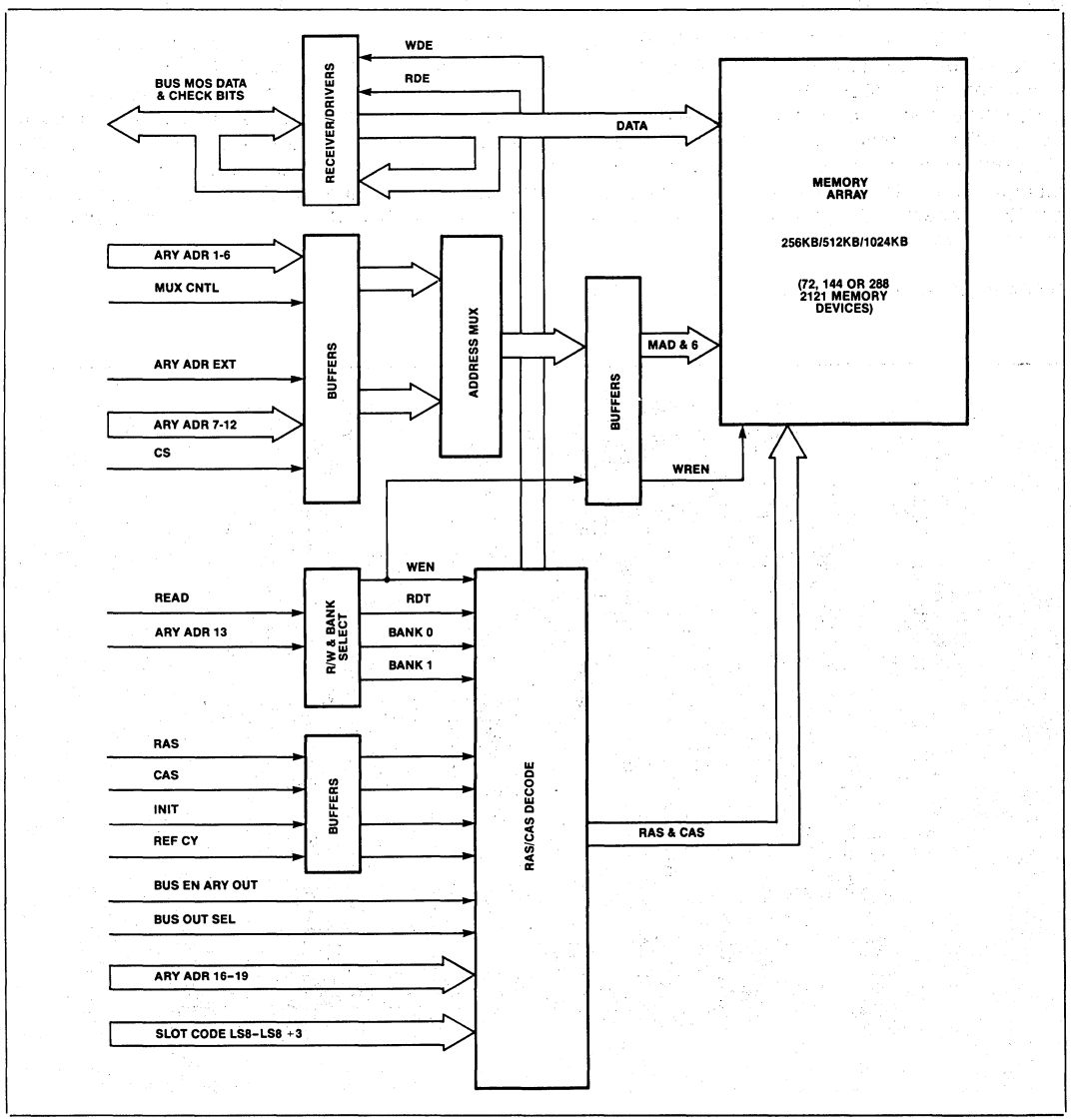


Figure 1. MU-5780 Block Diagram

Capacity

The MU-5780 is available in three configurations, 256K bytes (MU-5780-256), 512 bytes (MU-5780-512) or 1024K bytes (MU-5780-1MB). The configurations are organized as 32K x 72 bits, 64K x 72 bits or 128K x 72 bits. Each word contains 64 data bits and 8 check bits. The memory storage area consists of either 72, 144, or 288, Intel 2121, 32K x 1 dynamic memory IC's.

Addressability

The MU-5780 memory address circuitry consists of 19 address inputs plus four slot-code bits supplied to the memory by the VAX computer. The four slot-code bits are subtracted from the last four address bits (ARY ADR 16, 17, 18, and 19). Each subtraction selects a 256KB memory segment of the MU-5780 installed in the addressed slot.

An on-board multiplexer, under control of the MUX CNTRL command, applies seven address bits as row addresses, which select one of 128 rows on the memory modules, and seven column addresses, which select one of 128 columns on the memory modules. One address bit, 13, selects the memory bank. Figure 2 shows the address configuration.

INSTALLATION

The Intel® MU-5780 memory cards are installed in Slots 3 through 17 of the DEC MS-780 Memory System Backplane. Since some MU-5780s have a higher capacity than the DEC boards they replace, all memory slots may not be used in the MU-5780 application. With the MU-5780 512 version, the first card is installed in the first vacant memory slot (17 or lower), and in every alternate slot. With the MU-5780 1MB version, the first card is installed in the first vacant slot (17 or lower) and then in every fourth slot. Intel-supplied continuity boards must be installed in the empty slots between these MU-5780s. Continuity boards simulate the memory cards which would normally be installed in the slots and also direct air flow over the memory cards.

Reliability

Each MU-5780 is fully tested in a temperature-cycling environment. Because of the 100% burn-in performed on each card, the user is assured of receiving Intel's proven quality and reliability. In addition, the MU-5780 is covered by a one-year Intel warranty.

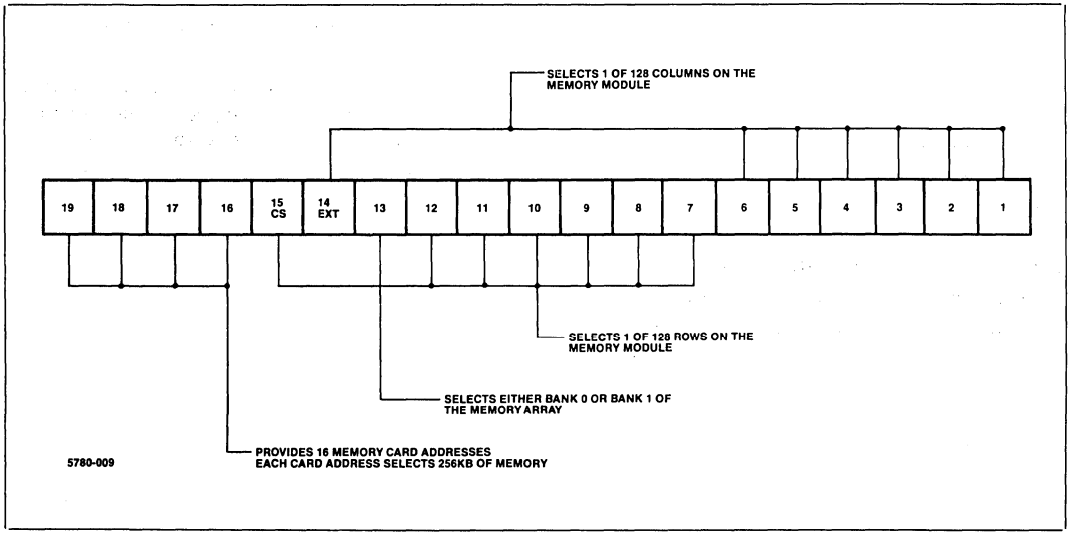


Figure 2. MU-5780 Address Configuration

SPECIFICATIONS

Storage Capacity

256KB MU-5780-256
512KB MU-5780-512
1024KB MU-5780-1MB

Word Length

64 data bits plus 8 check bits

Performance (Cycle Times—Maximum)

Read ..... 600 nanoseconds
Write ..... 600 nanoseconds
Initialize ..... 600 nanoseconds
Refresh ..... 600 nanoseconds

(Times shown are for the VAX memory controller. MU-5780 performance is 530 nanoseconds cycle time and 250 nanoseconds access time.)

Operational Modes

Read	Initialize
Write	Refresh

Refresh Rate

14 microseconds

Interface Characteristics

Address Lines:	23 lines, including four slot code lines
Data Inputs/Outputs:	72 bidirectional lines (64 data lines, 8 check bit lines)
Control Inputs:	Seven lines low-power Schottky
Power Requirement:	+5VDC, −5VDC, +12VDC, +5VDC battery backup

Physical Characteristics

Width ..... 39.8 cm (15.7 inches)
Height ..... 31.4 cm (12.4 inches)
Thickness ..... One card slot
Weight ..... 2.05 kilos (4.5 lbs.)

Environmental Characteristics

Temperature:	0°C to 50°, operating ambient −40°C to +125°C, non-operating
Relative Humidity:	Up to 90% without condensation
Altitude:	3,030 meters (10,000 ft) maximum, operating 15,151 meters (50,000 ft) maximum, non-operating

Reference Manuals

MU-5780 Add-In Memory Card (supplied), 112388

Reference manuals are shipped with each product only if designated (see above). Manuals may be ordered from any Intel sales representative.

ORDERING INFORMATION

Part Number	Capacity	Description
MU-5780-256	256K Bytes	256KB, 32K x 72
MU-5780-512	512K Bytes	512KB, 64K x 72
MU-5780-1MB	1024K Bytes	1024KB, 128K x 72

intel

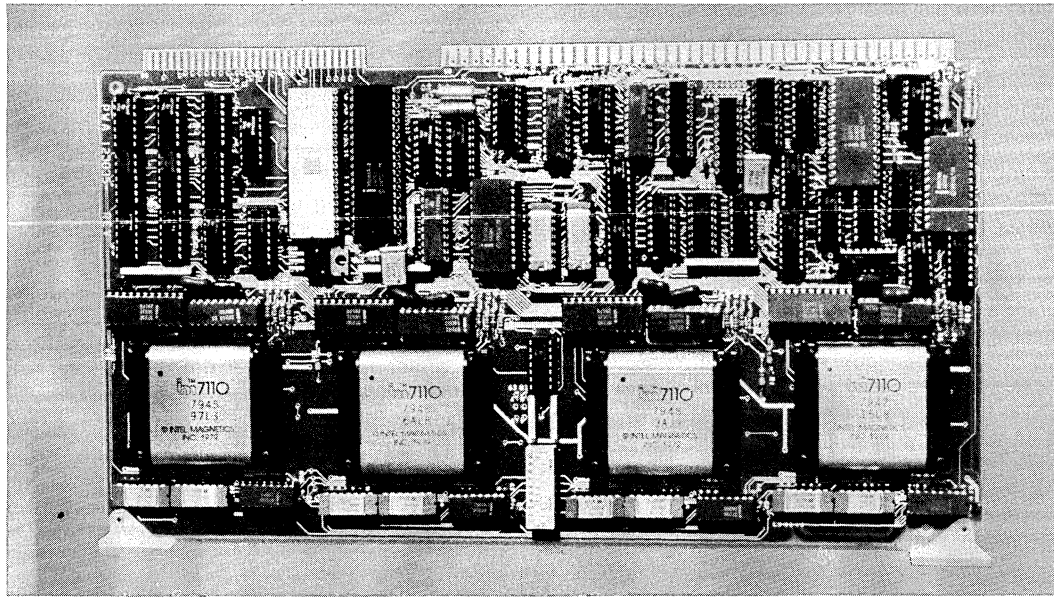
iSBC 254™
BUBBLE MEMORY BOARD

- **128K Byte to 512K Byte Capacity on Single iSBC™ Board**
- **48MS Average Access Time**
- **Low Power Consumption 35 Watts for 512 KB Version**
- **Non-Volatile Storage**
- **Multiple I/O Modes, Including DMA**
- **Driver Software for Operation with RMX 80/86**

The iSBC 254™ is a non-volatile memory utilizing the Intel 7110 one-megabit bubble memory element. The board is offered in three capacities: 128K, 256K, or 512K bytes.

The iSBC 254 can be operated in three I/O modes: polled status, interrupt-driven, or DMA. The 128K byte version can operate at a maximum transfer rate of 12.5K bytes per second. The multiple bubble elements of the 256K byte and 512K byte versions can be accessed in parallel to achieve maximum transfer rates of 25K and 50K bytes per second, respectively.

The physical outline of the iSBC 254 is the standard 12 inch by 6.75 inch iSBC card format. The depth of the board, however, is 0.62 inches, requiring two normally spaced card slots for adequate mechanical clearance. Power requirements for the iSBC 254 are 4 Amps at +5 volts and 1.2 Amps at +12 volts, maximum.



The following are trademarks of Intel Corporation and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{i}m, iCS, ICE, UPI, BXP, iSBC, iSBX, INSITE, iRMX, CREDIT, RMX/80, μScope, Multibus, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTI MODULE, and the combination of MCS, ICE, SBC, RMX or iCS and a numerical suffix; e.g., iSBC-80.

© Intel Corporation 1980

OPERATIONAL DESCRIPTION

Three distinct modes of operation relate to the transfer of data. Each is briefly described below.

Data Transfer

In DMA (Direct Memory Access) Mode, the iSBC 254 board utilizes the Intel 8257 DMA Controller, in conjunction with the 7220 Bubble Memory Controller, to perform three distinct types of DMA operation: DMA Read, DMA Write, and DMA Verify. In a DMA Read operation, data is transferred from memory to the FIFO (first-in/first-out RAM) of the 7220 BMC (Bubble Memory Controller). In a DMA Write operation, data is transferred from the 7220 BMC FIFO to memory. In a DMA Verify operation, the iSBC 254 board gains control of the bus, but no actual transfer of data takes place. DMA Read and DMA Write operations are used for high-speed data transfers involving bus-accessible memory; DMA Verify operations are typically used to maintain control of the system bus while verification tasks, such as checking newly acquired data, are being performed.

In Polled Mode, the CPU periodically checks the Status Register of the 7220 BMC. The Status Register can indicate a variety of conditions, one of which is that the BMC FIFO is ready to receive data or that the FIFO contains data to be read.

In DRQ (Data Request) Mode, when the FIFO of the 7220 BMC is half empty (during a write operation) or half full (during a read operation), the DRQ pin becomes active and an interrupt is issued, signalling that data may be written (bus to iSBC 254 board) or read (iSBC 254 board to bus).

Two distinct modes of operation relate to monitoring the bubble memory board status (via the 7220 Bubble Memory Controller). Each is briefly described below.

Status Monitoring Modes

In Interrupt Mode, a change in the 7220 BMC Status Register will cause an interrupt to occur, and the host processor will then look at that register to see what change has occurred. Any of the following may be indicated: the BMC sequencer is busy; an operation has been completed; an operation has failed; a timing error has occurred; and/or a correctable, uncorrectable, or parity error has occurred.

In Polled Mode, as described above under Data Transfer, the CPU periodically checks the Status Register of the 7220 BMC. The Status Register can indicate a variety of conditions: that the BMC Sequencer is busy; that an operation is complete; that an operation has failed; that a timing error has occurred; that a correctable, uncorrectable, or parity error has occurred; or, in terms of data transfer, that the BMC FIFO is ready to receive data or that the FIFO contains data to be read.

SOFTWARE DESCRIPTION

The iSBC 254 board can run under either the iRMX/80 or the iRMX/86 operating system.

Under the iRMX/80 operating system, the Bubble Manager (BMGR), a software task that runs with the iRMX/80 operating system, keeps track of free or available space on the Magnetic Bubble Memory. Another task, Bubble I/O (BUBIO), controls all iSBC 254 board operations. Bubble I/O can run with or without the Bubble Manager.

Under the iRMX/86 operating system, the iSBC 254 board is supported as an integral part of the I/O system software, which is part of the iRMX/86 operating system. (The iRMX/86 operating system device driver for the iSBC 254 board is linked with the I/O system software, which is in turn linked with the iRMX/86 operating system.) Because the iRMX/86 operating system provides convenient codes for performing operations, because all devices "look" the same when running under this operating system, and because of a variety of built-in features, the the iRMX/86 operating system provides great flexibility.

Software programs on both single- and double-density diskettes are provided with the iSBC 254 board. EX-254 is a set of programs that demonstrates how to use the various iSBC 254 board software commands.

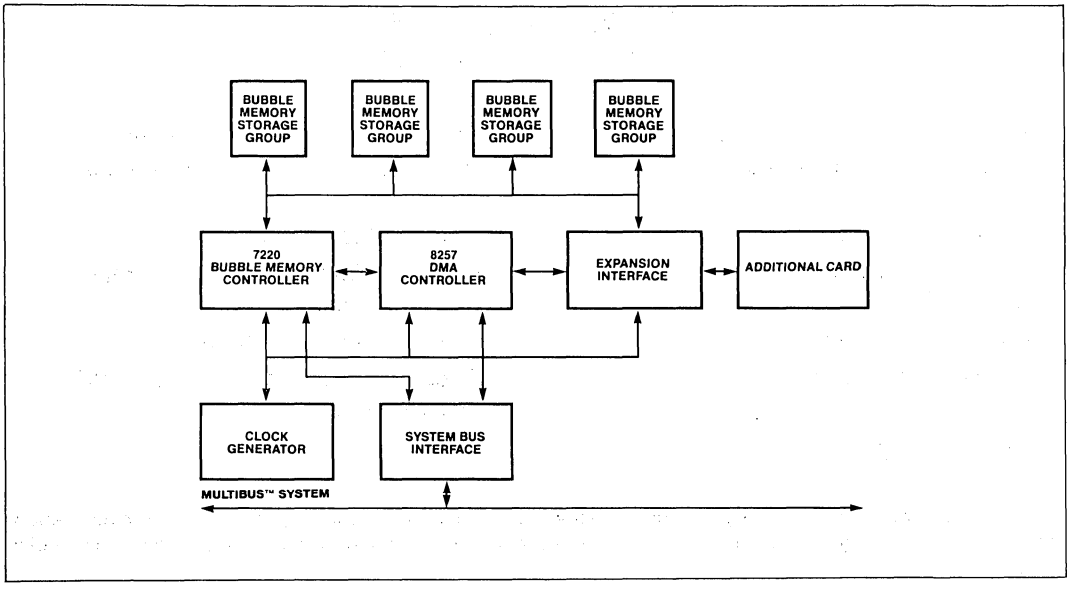


Figure 1. iSBC 254™ Board, Block Diagram

SPECIFICATIONS

Memory Size

128K, 256K, or 512K bytes

Interface

All address, data, and control signals are TTL-compatible and Intel MULTIBUS system compatible.

Electrical Characteristics

D.C. Power
+5 volts D.C. ±5%, 3.0A (max.)
+12 volts D.C. ±5%, 1.4A (max.)

Performance

Rotating Field Rate: 50KHz
Maximum Data Rate: 50K bytes/second
Average Access Time: 48ms

Connector

86-pin double-sided PC edge connector with 0.40 cm (0.156 in.) contact centers.
Mating Connector: Control Data VFB01E43D0A1 or Viking 2VH43/1ANE5.

Physical Characteristics

Length: 30.48 cm (12 in.)
Height: 17.15 cm (6.75 in.)
Depth: 1.57 cm (0.62 in.)

Note: Because of its depth, the iSBC 254 board requires two card slots.

Environment

Board Operating Temperature: 0–55°C

Equipment Supplied

iSBC 254 Bubble Memory Board
iSBC 254 Operation Manual
iSBC 254 Software (single- and double-density diskettes)

iPAB PLUG-A-BUBBLE™ MEMORY SYSTEM

- **Removable bubble memory cassette**
- **128K bytes storage capacity per cassette**
- **Ruggedized cassettes**
- **Optional interface to iSBX™ multimodule bus**
- **Automatic error correction**
- **Non-volatile storage**
- **Powerfail data protection**
- **Write protection**
- **Average access time of 48 ms**
- **Burst data rate of 12.5K bytes per second**
- **Designed to support DMA**
- **Operates from standard +5V and +12V power supplies**
- **Low power consumption**
- **Accommodates up to sixteen holders and cassettes**

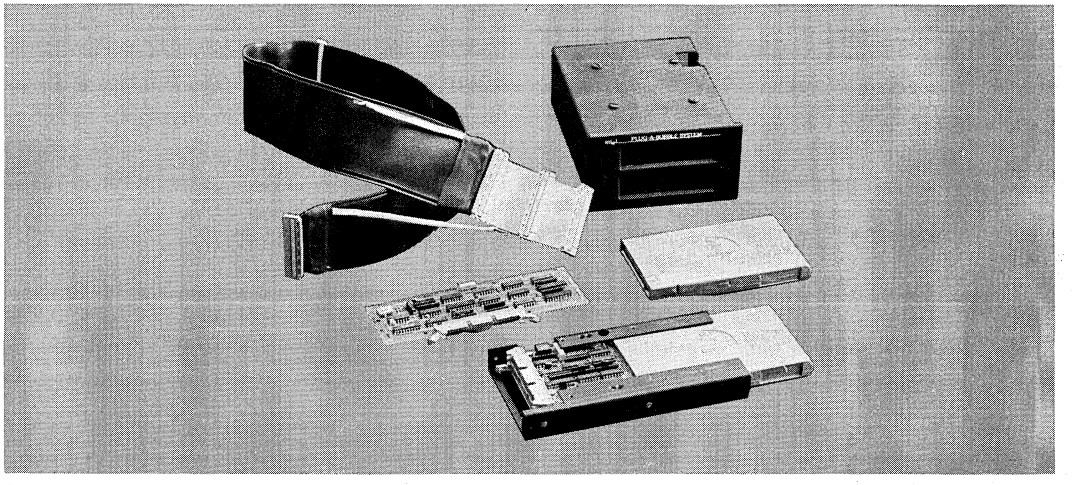
The Plug-A-Bubble™ system (iPAB) is a bubble memory system designed for easy removal and replacement of media (bubble memory cassettes), similar to the removal and replacement of media (magnetic discs) in disc systems.

The "media" of the iPAB system is a sturdy protective casing that contains a one-megabit bubble memory, support components, three LEDs, and discrete capacitors and resistors. The protective casing is designed to protect these devices from dust, shock, and other elements of an adverse environment.

For users of single-board computers with the iSBX bus, an iSBX Multimodule interface card (MMI/O card) is available for generating the required signals for the cassette holder.

iPAB is a bubble memory system and is therefore non-volatile memory. Like disc systems, the iPAB system has write protection (a switch on the cartridge holder provides this); it also has powerfail protection.

Below is a photo of the components that can make up an iPAB system. The photo shows: *top left,* standard cable (iPAB 0362); *top center,* chassis (iPAB 525); *top right* (slightly lower), holder and cassette (iPAB 381); *bottom left,* Multimodule interface (MMI/O) card (iPAB 258); *bottom right,* cassette (iPAB 128). (For various configurations of an iPAB system, see Figure 1.)



GENERAL DESCRIPTION

A Plug-A-Bubble™ iPAB system may be configured in various ways, depending on a user's needs, but common to every iPAB system is at least one iPAB cassette and one iPAB holder (iPAB 381).

An iPAB cassette consists of a PC board with a one-megabit bubble memory device, support components including a controller, three LEDs, a number of discrete resistors and capacitors, and a protective casing that encloses the PC board and protects it from dust, shock, and other elements of an adverse environment.

An iPAB cassette holder "holds" a cassette (a holder and a cassette are mated via a plug on the holder and a jack on the cassette); it also contains a jack for the required external signals to/from the holder. The holder provides logic for selection of the cassette, bidirectional circuitry for gating of data, and write protection logic.

Variable components of an iPAB system are the standard data cable, (iPAB 0362), an MMI/O card (iPAB 258), a chassis (iPAB 525). (See Figure 1 for various user and iPAB system configurations.)

The standard cable is shielded and 36 inches long. It comes with four connectors: Two are for connecting holders; one is for connecting the cable to an MMI/O card, if an MMI/O card is used, or to a user-supplied device that generates the required holder signals, if an MMI/O card is not used; and one is for connecting the end of the cable to a terminator card in the last holder.

The MMI/O card is an iSBX Multimodule interface card available for users with single-board computers with the iSBX bus. The function of this card is to generate, under control of the user's system, the required signals for the iPAB cassette holder(s). Without external power, a single MMI/O card can support two holders and cassettes; with external power, up to eight holders and cassettes can be supported. The MMI/O card contains: latches for the address lines that are used for iPAB cassette selection; data gating circuitry; parity generating circuitry; MMI/O status logic; circuitry for generating a wait signal for a DMA controller (not contained in the iPAB system) or the CPU; circuitry for generating the iPAB clock; and jumpers for selecting a variety of options. The MMI/O card comes with the standard cable described above.

If the MMI/O card is not used, the user must generate the required signals for the iPAB cassette holder(s). Note that a holder has manual switch setting and address lines that allow the user to select any one of sixteen different holders, so that with sufficient power a single iPAB system without an MMI/O card may support up to sixteen holders and cassettes.

The iPAB chassis houses two iPAB holders. Its dimensions are 5.75 × 3.25 × 8.0 inches, so that it occupies the same space as a 5.25-inch floppy disc drive.

COMMUNICATION

Communication with individual iPAB cassettes is accomplished via reading/writing two I/O ports in the 7220-1 Bubble Memory Controller (BMC). (The 7220-1 BMC is one of the bubble memory support components contained in the iPAB cassette.) These two ports allow access to a variety of registers in the 7220-1 BMC, so that commands may be issued, status may be read, and a variety of parameters relating to bubble memory operation may be set.

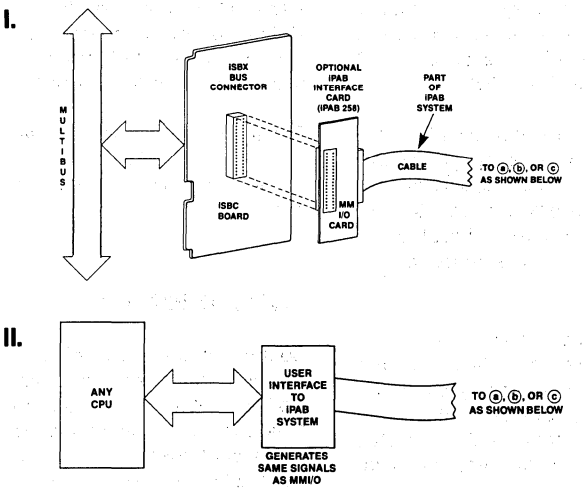
MODES OF OPERATION

There are three possible modes of operating: Interrupt Mode, Polled Mode, and DMA (Direct Memory Access) Mode. Interrupt Mode and Polled Mode are used both for data transfers and for monitoring status; DMA Mode, which requires some additional circuitry to implement, is used for data transfers only.

In terms of *data transfers,* Interrupt Mode is implemented by connecting the Data Request line to a host interrupt line, the Data Request line indicating, when active, that the FIFO of the 7220-1 Bubble Memory Controller is half empty (during a write operation) or half full (during a read operation), and hence that service is required by the host processor. (The 7220-1 BMC FIFO is a First-In/First-Out register used as a data buffer between host and bubble memory.) In Polled Mode, the host processor periodically checks the status register of the 7220-1 BMC to determine whether service is required. With some additional circuitry, a third mode, DMA Mode, is possible. (The 7220-1 BMC has DMA handshaking capability for DMA transfers, but to implement this mode a user-supplied DMA controller is necessary.)

In terms of monitoring cassette status, Interrupt Mode is implemented by connecting the 7220-1 BMC Interrupt (INT) line to a host interrupt line and setting

USER CONFIGURATIONS



iPAB SYSTEM

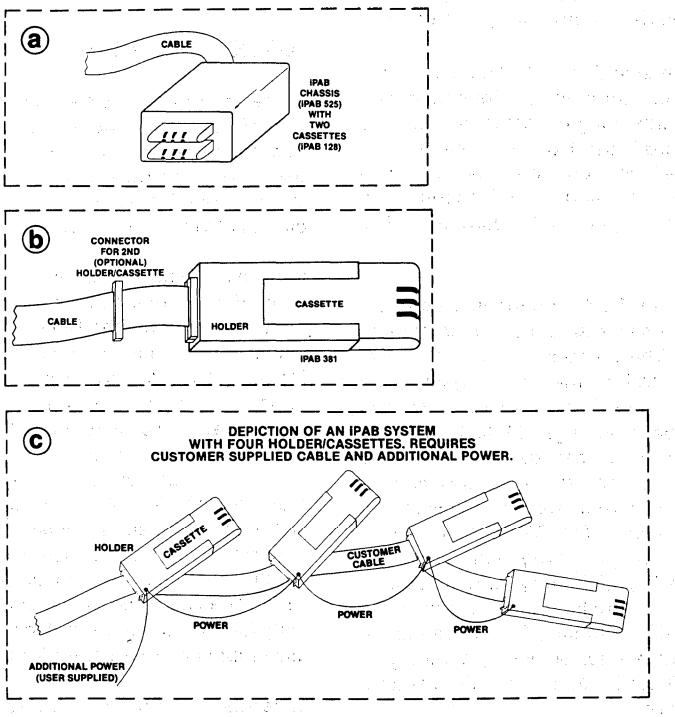
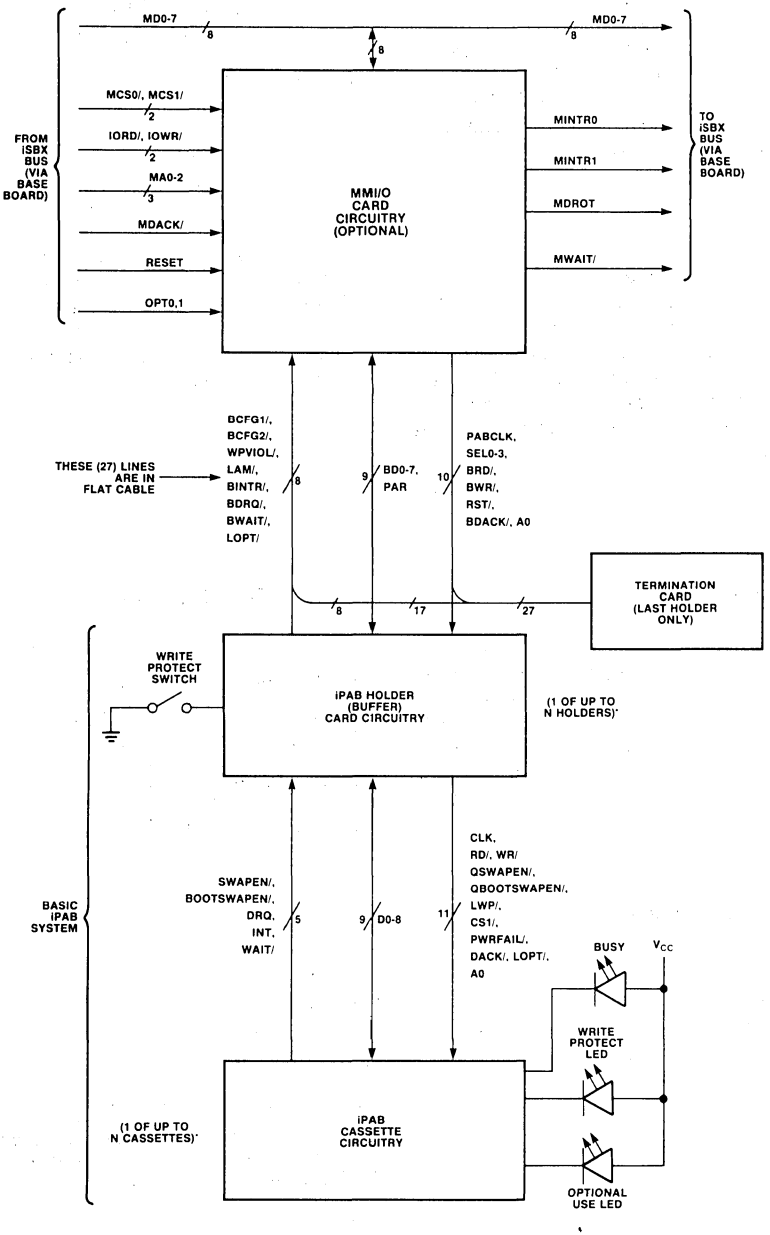


Figure 1. User iPAB Configurations/Options



*Currently N 2. With future design changes, the value of N may be increased (up to N 8).
For ease of presentation, only one iPAB basic system is shown.

Figure 2. iPAB System Overall Block Diagram (with optional MMI/O card)

various bits in the enable register (Enable ICD, Enable RCD, and/or Interrupt Enable (Error) for error interrupts; Interrupt Enable (Normal) for an interrupt to occur at the completion of a task). Then either (or both) the completion of a task or the occurrence of an error will result in an interrupt. In Polled Mode, the host processor periodically reads the 7220-1 BMC status register to determine if an operation has been completed or if an error has occurred.

SOFTWARE

Actual control of bubble memory operation is accomplished via reading or writing various registers in the 7220-1 Bubble Memory Controller as described above. But for most users it is convenient to talk to bubble memory in a higher-level language —or via a software "driver" that translates the higher-level language into the language of the bubble memory.

For users with the iRMX/86 or iRMX/88 operating systems, who wish to perform data transfers in DMA Mode, iPAB software drivers are available as part of those operating systems.

For users who want to write their own drivers, detailed instructions are provided in the iPAB technical manual (Part Number 112583) on how to program the registers for desired bubble memory operation, and technical support is available from the Intel Application group.

For users with Intel Microcomputer Development Systems, a Bubble Demo package is available on floppy disc, allowing users to read data from, and write data to, the bubble system; test its operation; read the bubble's bootloop; and test the iPAB's write protection switch.

SPECIFICATIONS

Capacity per Cassette ..... 128K bytes
Page Length ..... 68 bytes—ECC disabled;
64 bytes—ECC enabled
Number of Pages 2,048
Rotating Field Rate ..... 50 kHz
Date Rate ..... 12.5K bytes/sec. (burst rate)
Average Access Time ..... 48 ms

Power Requirements

	Current	
Voltage	**Holder**	**Cassette**
VCC +5V ± 5%	280 ma	290 ma
VDD +12V ± 5%	N/A	400 ma
Total Power Consumption PD_M = 7.65W (Maximum) PD_T = 5.1W (Typical)		

Environmental Specifications

Environmental Parameter	Specification
Operating Temperature	0 to 55°C
Non-Operating Temperature (with Data Retention)	−40 to +100°C
Humidity (Without Condensation)	0 to 95% FRH
Shock	30g over 11 ms with ½ sinewave shape
Vibration	5 Hz to 15 Hz @ .50 inches double-amplitude displacement (.65g to 6g); 15 Hz to 2000 Hz @ 5g (.42 to .000025 inches double-amplitude displacement)
Insertion Cycles	2000

ORDER INFORMATION

Part Number	Description
iPAB 381	Cassette holder (7.75″ × 3.8″ × 1″) and one 128K byte cassette (6.1″ × 3.6″ × .81″). (Note: When fully inserted in the holder, the cassette extends 2.5″ from the end of the holder into which it is inserted.)
iPAB 128	128K-byte bubble memory cassette.
iPAB 258	iSBX Multimodule interface card (MMI/O card). This comes with shielded 36″ cable with connectors for two cassette holders.
iPAB 525	This is a 5.75″ × 3.25″ × 8.0″ chassis housing two iPAB cassette holders. It occupies the same space as does a 5.25″ floppy disc drive.
iPAB 0362	Shielded 36″ cable with connectors for two cassette holders.

Software

iRMX™/86 iPAB Driver	Comes with iRMX/86 software.
iRMX/88 iPAB Driver	Comes with iRMX/88 software.
Bubble Demo	This is also available from MSO Intel upon request (Order No. 113073-001).

PRELIMINARY

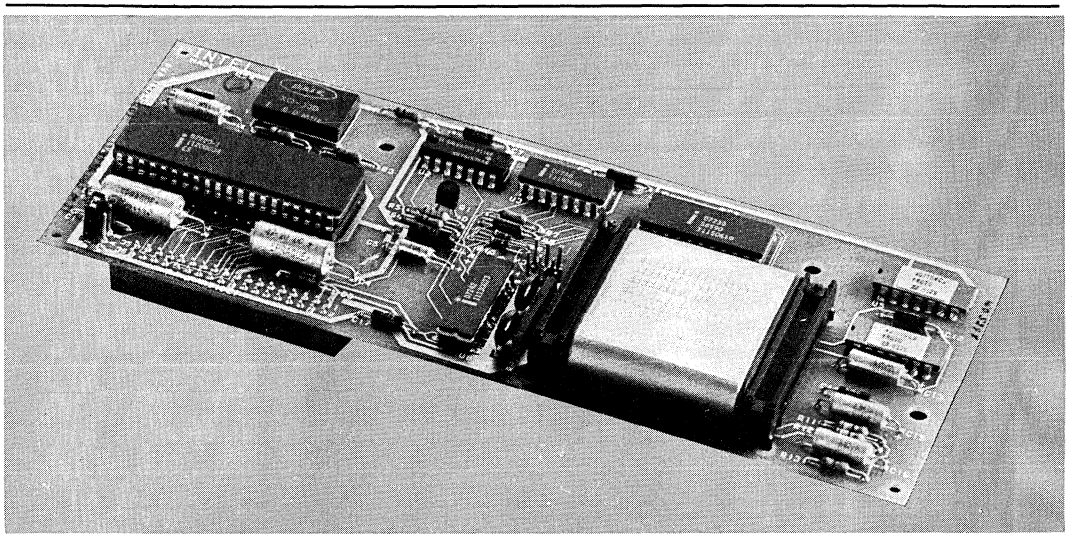
iSBX 251™ MAGNETIC BUBBLE MULTIMODULE™ BOARD

- **iSBX™ bus compatible**
- **Capacity: 128K bytes**
- **Performance:**
  —average access time: 48ms
  —burst data rate: 12.5K bytes/sec
- **DMA compatibility**
- **Non-volatile storage**
- **Automatic error correction**
- **Operates from standard +5V and +12V power supplies**
- **Software compatibility with iRMX™ Operating System**
- **Power fail data protection**
- **Low power consumption**

The Intel iSBX 251 magnetic bubble MULTIMODULE board is a completely assembled and tested non-volatile memory based on the Intel 7110 one-megabit bubble memory. This board plugs into any Intel iSBC Single Board Computer equipped with an iSBX connector, providing immediate, high density storage. This arrangement frees the MULTIBUS for other traffic while the host iSBC board accesses the bubble memory.

Support circuitry provides the user with a simple interface to the bubble memory. For instance, sixteen commands are available to transfer data and to view the operational status of the iSBX 251 board. Additionally, memory reliability is increased if the automatic Error Check and Correction (ECC) feature of the support circuitry is selected.

The iSBX 251 board can be wired to transfer data in one of three modes: polled access, interrupt driven, and Direct Memory Access (DMA). The user may also select polled or interrupt driven access to the 7220-1 Status Register. Thus, the user can tailor the type of access to the individual application.



The following are trademarks of Intel Corporation and its affiliates and may be used only to identify Intel products: i, Int$_e$l, INTEL, INTELLEC, MCS, ^{l}m, iCS, ICE, UPI, BXP, iSBC, iSBX, Insite, iRMX, System 2000, CREDIT, iRMX/80, MULTIBUS, PROMPT, Promware, Megachassis, Library Manager, MAIN MULTIMODULE, and the combination of MCS, ICE, iSBC, iRMX or iCS and a numerical suffix. © INTEL CORPORATION, 1981.

AUGUST 1981
AFN-02016A

SPECIFICATIONS

Storage Capacity

—128K Eight-Bit Bytes
—2048 Pages
—Page Length:
64 bytes with ECC
68 bytes without ECC

Operational Modes

Polled, Interrupt Driven, or DMA (with Host DMA Controller)

Electrical Requirements

D.C. power, supplied through iSBX connector:
+5V ±5%, 365 mA (max.)
+12V ±5%, 400 mA (max.)

Performance

Rotating Field Rate	50 KHZ
Maximum Data Rate	12.5K byte/sec
Average Access Time	48 ms

Interface Requirements

—TTL compatible
—iSBX 251 male connector plugs into 36-pin or 44-pin host female connector
—Connector located per Intel iSBX Bus Specification (order number 142686) double wide form factor

Physical Characteristics

Width	7.24 cm (2.85 in.)
Length	19.05 cm (7.50 in.)
Height	2.53 cm (0.996 in.)
Weight	362.9 gm (12.8 oz.)

Environment

Temperature:
0°C to 60°C (32°F to 140°F) ..... Operating
−40°C to 100°C (−40°F to 212°F) .. Non-Operating with Data Retention

Relative Humidity:
0% to 95% without condensation

RELIABILITY

Intel tests each iSBX 251 board at elevated voltages and temperatures for a period proven to bring out most latent defects. This process, together with temperature cycling, is designed to remove all potential failures before the memory board reaches the customer. Because of the 100% burn-in performed on each board, the user is assured of Intel's proven quality and reliability.

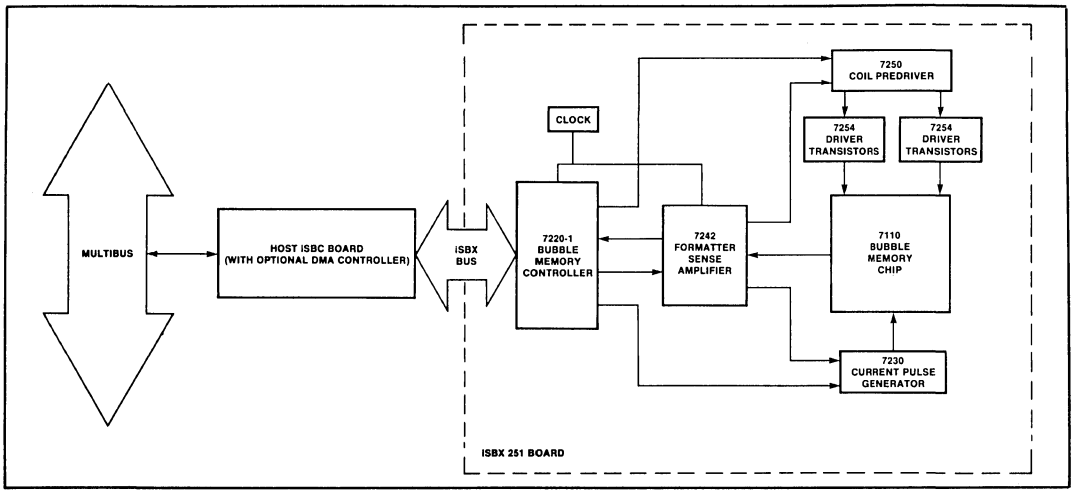


Figure 1. Block Diagram

Insite™ User's Program Library

23

intel

INSITE™ PROGRAM LIBRARY

- **Programs for 8008, 8048, 8080, 8088 and 8086 Processors**
- **Accepted Program Submittals Entitle You to a Free Membership or Free Program Package**
- **Diskettes, Paper Tapes, and Listings Available for Library Programs**
- **Program Library Catalog Offering Hundreds of Programs**
- **Updates of New Programs Sent During Subscription Period**

Insite, Intel's Software Index and Technology Exchange Library, is a collection of programs, subroutines, procedures, and macros written by users of Intel's microcomputers, single-board computers, and Intellec development systems. Thanks to customer contributions to Insite, Intel is able to make these programs available to all users of Intel microcomputers who are Library members. By taking advantage of the availability of these general-purpose routines, the microcomputer design engineer and programmer can save many hours of programming and debugging time. The library of programs also serves as a good learning tool for those unfamiliar with Intel assembly language or the high-level languages for Intel's family of microcomputers.

Membership. Membership in Insite is available on an annual basis. Intel customers may become members through an accepted program contribution or membership fee. New members should use the membership form below.

Program Submittals. The Insite Library is built on user submittals. Customers are encouraged to submit their programs. (Details and forms are available through the Insite Library.) For each accepted program, submittors will receive a choice of three free regular priced programs on diskette, three paper tapes, or a free membership with Insite.

Program Library Service. PAPER TAPES, DISKETTES OR SOURCE LISTINGS are available for every program in Insite. Diskettes are available on single or double density. Membership is required to purchase programs.

Insite™ Program Library Catalog. Each member will be sent the Program Library Catalog consisting of an abstract for each program indicating the function of the routine, required hardware and software, and memory requirements.

Insite members will be updated with abstracts of new programs submitted to the Library during the subscription period. For catalog and yearly subscription fee please refer to the Intel OEM Price List or contact the nearest Intel Sales Office.

The following are trademarks of Intel Corporation and may be used only to identify Intel products: Insite, Intel, intel, Intellec. ©Intel Corporation 1981

insite™ PROGRAM LIBRARY MEMBERSHIP FORM

I am interested in becoming a member of Insite™

Enclosed is ☐ Check or Money Order ☐ Program Submittal ☐ Purchase Order (Inside U.S. only)

NAME: _____

COMPANY: _____ TELEPHONE: _____

SHIP TO: _____

_____

_____

SEND TO NEAREST LOCATION:

North America: Intel Corporation, Insite Program Library MS 6-5000, Microcomputer Systems, 3065 Bowers, Santa Clara, California 95051, Ph. 408-987-8080

Europe: Intel International Corp. S.A., Insite Program Library, Rue du Moulin a Papier 51, Boite 1, B-1160 Brussels, Belgium, Ph. 02-660-3010

Orient: Intel Japan K.K., Insite Program Library, Flowerhill-Shinmachi, East Bldg., 1-23-9 Shinmachi, Setagaya-ku, Tokyo 154, Japan, Ph. 813-426-9261 (PME & FSE), 813-426-9267 (CS & Fin.)

AFN-02045A

INSITE PROGRAM SUBMITTAL REQUIREMENTS

Submittal Requirements

Programs submitted for Insite review must follow the guidelines below:

A. Programs must support Intel products

B. Program submittals must be packaged to include:

—Complete submittal form

—Source listing (the output of a compilation/assembly)

—Well-documented source code and executable code furnished on ISIS-formatted diskette or ASCII-coded paper tape

—Detailed instructions (step-by-step from source code compilation/assembly through program execution)

—Test program demonstrating accurate operation and output

C. Programs must be written in a language capable of compilation/assembly by the currently supported version of an Intel standard compiler/assembler. Acceptable languages are documented in the following manuals (available through Intel's Literature Department):

—*MCS-48/UPI-41 Assembly Language Manual*, Stock Number 305000

—*8080/8085 Assembly Language Programming Manual*, Stock Number 401100

—*8086 Assembly Language Reference Manual*, Stock Number 402105

—*BASIC-80 Reference Manual*, Stock Number 400710

—*ICIS-COBOL Language Reference Manual*, Stock Number 409115

—*Fortran-80 Programming Manual*, Stock Number 400625

—*PL/M-80 Programming Manual*, Stock Number 401700

—*PL/M-86 Programming Manual*, Stock Number 402325

—*Intellec Series III Microcomputer Development System Programmer's Reference Manual*, Stock Number 404440

—*PASCAL-80 Language Reference Manual*, Stock Number 400660

Submittal Acknowledgement

Program acceptance/rejection will be acknowledged through correspondence. Programs accepted by Insite become the property of the Insite Library. Programs rejected by Insite will be returned to the submittor with an explanation for the rejection.

insite™ INTEL® USER'S LIBRARY SUBMITTAL FORM

☐ 8008 ☐ 8048 ☐ 8080/8085 ☐ Other (use additional sheets if necessary)

Program Title	
Function	
Required Hardware	
Required Software	
Input Parameters	
Output Results	

SAMPLE

Registers Modified:	Programmer:
RAM Required:	Company:
ROM Required:	Address:
Maximum Subroutine Nesting Level:	City:
Assembler/Compiler Used:	State:

©Intel Corporation, 1978.

Software Distribution Operation

24

DIGITAL RESEARCH INC.
CP/M* 2.2 OPERATING SYSTEM

- **High-performance, single-console operating system**
- **Simple, reliable file system matched to microcomputer resources**
- **Table-driven architecture allows field reconfiguration to match a wide variety of disk capacities and needs**
- **Extensive documentation covers all facts of CP/M applications**
- **General-purpose subroutines and table-driven data-access algorithms provide a truly universal data management system**
- **Upward compatibility from all previous versions**
- **More than 500 commercially available compatible software products**

CP/M 2.2 is a monitor control program for microcomputer system and application development using the Intel 8080/8085-based microcomputer with IBM-compatible flexible disks for backup storage (see the *CP/M-86* Operating System* data sheet for information on CP/M for Intel 8086/8088-based systems). CP/M provides a general environment for program construction, storage, and editing, along with the program assembly and check-out facilities.

The CP/M monitor provides rapid access to programs through a comprehensive file management package. The file subsystem supports a named file structure, allowing dynamic allocation of file space as well as sequential and random file access. Using this file system, a large number of distinct programs can be stored in both source- and machine-executable form.

CP/M also supports a powerful context editor, Intel-compatible assembler, and debugger subsystems. Optional software includes a powerful Intel-compatible macro assembler, and a symbolic debugger, along with various high-level languages.

FEATURES

CP/M is logically divided into four distinct modules:

BIOS—Basic I/O System

—Provides primitive operations for access to disk drives and interface to standard peripherals (teletype, CRT, paper tape reader/punch, and user-defined peripherals)

—Allows user modification for tailoring to a particular hardware environment

BDOS—Basic Disk Operating System

—Provides disk management for one to sixteen disk drives containing independent file directories

—Implements disk allocation strategies for fully dynamic file construction and minimization of head movement across the disk

—Uses less than 4K of memory

—Makes programs transportable from system to system

—Entry points include the following primitive operations which can be programmatically accessed:

SEARCH	Look for a particular disk file by name
OPEN	Open a file for further operations
CLOSE	Close a file after processing
RENAME	Change the name of a particular file
READ	Read a record from a particular file
WRITE	Write a record to a particular file
SELECT	Select a particular disk drive for further operations

CCP—Console Command Processor

—Provides primary user interface by reading and interpreting commands entered through the console

—Loads and transfers control to transient programs, such as assemblers, editors, and debuggers

—Processes built-in standard commands including:

ERA	Erase specified files
DIR	List file names in the directory
REN	Rename the specified file
SAVE	Save memory contents in a file
TYPE	Display the contents of a file on the console

TPA—Transient Program Area

—Holds programs which are loaded from the disk under command of the CCP

—Programs created under CP/M can be checked out by loading and executing these programs in the TPA

—User programs, loaded into the TPA, may use the CCP area for the program's data area

—Transient commands are specified in the same manner as built-in commands

—Additional commands can be easily defined by the user

—Defined transient commands include:

PIP	Peripheral Interchange Program —implements the basic media transfer operations necessary to load, print, punch, copy, and combine disk files; PIP also performs various reformatting and concatenation functions. Formatting options include parity-bit removal, case conversion, Intel hex file validation, subfile extraction, tab expansion, line number generation, and pagination
ED	Text Editor—allows creation and modification of ASCII files using extensive context editing commands: string substitution, string search, insert, delete and block move; ED allows text to be located by context, line number, or relative position with a macro command for making extensive text changes with a single command line
ASM	Fast 8080 Assembler—uses standard Intel mnemonics and pseudo operations with free-format input, and conditional assembly features
DDT	Dynamic Debugging Tool—contains an integral assembler/disassembler module that lets the user patch and display memory in either assembler mnemonic or hexadecimal form and trace program execution with full register and status display; instructions can be executed between breakpoints in real-time, or run fully monitored, one instruction at a time
SUBMIT	Allows a group of CP/M commands to be batched together and submitted to the operating system by a single command
STAT	Lists the number of bytes of storage remaining on the currently logged disks, provides statistical information about particular files, and displays or alters device assignments
LOAD	Converts Intel hex format to absolute binary, ready for direct load and execution in the CP/M environment
SYSGEN	Creates new CP/M system disks for back-up purposes
MOVCPM	Provides regeneration of CP/M systems for various memory configurations and works in conjunction with SYSGEN to provide additional copies of CP/M

BENEFITS

—Easy implementation on any computer configuration which uses an Intel 8080/8085 Central Processing Unit (see the CP/M-86 data sheet for CP/M applications on the iAPX86 CPU)

—Extensive selection of CP/M-compatible programs allows production and support of a comprehensive software package at low cost

—Easy migration path to timesharing systems with multiprogramming and multiterminal features (see the MP/M*, CP/NET*, and MP/NET* data sheets)

—Field programmability for special-purpose operating system requirements

—Upward compatibility from previous versions of CP/M release 1

—Provides field specification of one to sixteen logical drives, each containing up to eight megabytes

—Files may contain up to 65,536 records of 128 bytes each but may not exceed the size of any single disk

—Each disk is designed for 64 distinct files—more directory entries may be allocated if necessary

—Individual users are physically separated by user numbers, with facilities for file copy operations from one user area to another

—Relative-record random-access functions provide direct access to any of the 65,536 records of an eight-megabyte file

SPECIFICATIONS

Hardware Required

Intellec Microcomputer Development System
—Series II or
—Model 800

Optional: RAM up to 64K

—additional floppy disk drives
—Winchester disk drive

Documentation Package

Part Number	Title
121865 and optional binder 121864	CP/M 2.2 documentation consisting of 7 manuals: An Introduction to CP/M Features and Facilities CP/M 2.2 User's Guide CP/M Assembler (ASM) User's Guide CP/M Dynamic Debugging Tool (DDT) User's Guide ED: A Context Editor for the CP/M Disk System User's Manual CP/M 2 Interface Guide CP/M 2 Alteration Guide

Shipping Media

(Specify by Alpha Character when ordering.)

A—single density (Intellec compatible, IBM 3740/1 format)

Order Code **Product Description**

Availability: January 1982

An Intel Master Software License and Amendment required.
*CP/M is a registered trademark of Digital Research, Inc.
*CP/M-86, MP/M, CP/NET, and MP/NET are trademarks of Digital Research, Inc.

DIGITAL RESEARCH INC.
CP/M-86* OPERATING SYSTEM

- **High-performance, single-user operating system for 16-bit computers based on Intel's iAPX-86 (8086) and compatible iAPX-88 (8088) microprocessors**
- **CP/M-86 files are completely compatible with versions of CP/M* used with Intel 8-bit 8080- and 8085-based microcomputer systems**
- **Supports up to 16 logical drives, each containing up to eight megabytes for a total of 128 megabytes of on-line storage**
- **CP/M-86 manages up to 1 Mbyte of on-line RAM storage for full support of programs as large and as powerful as those found on minicomputers**
- **The standard CP/M-86 package includes an assembler, an interactive debugger, and additional utilities for complete program development**

CP/M-86 is a single-user operating system designed especially for 16-bit computers based on the Intel iAPX-86 (8086) and compatible iAPX-88 (8088) microprocessors. The system fully utilizes the one megabyte (1,048,576 bytes) of main memory available for application programs.

CP/M-86 occupies only 11K of memory with a floppy disk-based I/O System occupying about 1K, depending on the number of peripherals supported. The remainder of the 8086/8088 address space may be defined by the user. Flexibility is provided by the system's ability to reside anywhere in memory, and it can be relocated by the user without changing the operating system.

Because CP/M-86 files are completely compatible with CP/M for Intel 8-bit microprocessors, there is an easy upgrade of existing CP/M applications software to 16-bit iAPX-86 (8086)-based systems.

FEATURES

Major features of the CP/M-86 operating system include:

Modular Design

Based on a proven, modular design, the system includes the:

—CCP: Console Command Processor
The CCP is the human interface to the operating system and performs decoding and execution of user commands

—BDOS: Basic Disk Operating System
The BDOS is the logical, invariant portion of the operating system; it supports a named file system with a maximum of 16 logical drives, containing up to 8 megabytes each for a potential of 128 megabytes of on-line storage

—BIOS: Basic Input/Output System
The physical variant portion of the operating system, BIOS contains the system-dependent input/output device handlers

CP/M Compatibility

CP/M-86 files are completely compatible with CP/M for 8080- and 8085-based microcomputer systems. This simplifies the conversions of software developed under CP/M and allows full advantage of 16-bit 8086-based systems.

The user will notice no significant difference between CP/M and CP/M-86. Commands such as DIR, TYPE, REN, ERA, PIP, ED, and STAT respond the same way in both systems.

CP/M-86 uses the 8086 registers corresponding to 8080 registers for system call and return parameters to further simplify software transport. The execution environment in CP/M-86 allows code and data segments to overlap, making the mixture of code and data that often appears in 8-bit applications acceptable to the 8086.

File Management

CP/M-86 can support up to 16 logical drives, each containing up to eight megabytes, for a maximum of 128 megabytes of on-line storage. Any one file can reach the full drive size, with space dynamically allocated and released. Each device has a directory of file control blocks that map the physical location of each file on the disk. Disk definition tables in the BIOS translate this logical drive, directory, and file structure to the physical characteristics of the disk. This file system is identical to the file system of CP/M.

Memory Management

CP/M-86 can reside anywhere in memory and is easily located to minimize dependence on absolute addresses. Multiple programs may reside in memory simultaneously. Also, a transient program may load additional programs for execution under its own control. Up to a total of 8 programs may use non-contiguous memory areas which are managed through a user-defined memory configuration table.

Basic I/O System (BIOS) Organization

The distribution version of CP/M-86 is set up for operation with the Intel iSBC 86/12A and an Intel 204 diskette controller. All hardware dependencies are, however, concentrated in subroutines which are collectively referred to as the Basic I/O System, or BIOS. A CP/M-86 system implementor can modify these subroutines to tailor CP/M-86 to fit nearly any 8086 or 8088 operating environment.

To simplify the preparation of a custom BIOS, a source listing of a working BIOS, a skeleton for a custom module, and cross-development utilities are supplied. The cross-development programs allow development of custom BIOS and CP/M-86 applications software on an 8-bit CP/M system.

PROM Loader

For users who have the iSBC 86/12 hardware configuration, there is an optional PROM Loader. The firmware brings the CP/M-86 loader into the system and sets up the hardware to initialize CP/M-86.

Utilities

CP/M-86 is supplied with eight utilities:

PIP
The Peripheral Interchange Program provides file transfer between devices and disk files and performs various reformatting and concatenation functions. Formatting options include parity-bit removal, case conversion, Intel "hex" file validation, subfile extraction, tab expansion, line number generation, and pagination.

ED
The CP/M-86 Text Editor allows creation and modification of ASCII files using extensive commands: string substitution, string search, insert and delete. ED allows text to be located by context, line number, or relative position with a macro command for making extensive text changes with a single command line.

ASM-86
ASM-86, the CP/M-86 Assembler is an 8086 assembler using standard Intel mnemonics. It also allows users to define unique instructions with the code-macro facility. ASM-86 is supplied in two forms: an 8086 cross assembler designed to run under CP/M (an 8-bit system), and an 8086 assembler designed to run under CP/M-86.

DDT-86
The CP/M-86 Dynamic Debugging Tool allows the user to test and debug programs interactively in a CP/M-86 environment. The command set allows users to trace program execution with full register and status display. DDT-86 contains an integral assembler/disassembler module that lets users patch and display memory in assembler mnemonic form.

SUBMIT
Allows a group of CP/M-86 commands to be batched together and submitted to the operating system by a single command.

STAT
Lists the number of bytes of storage remaining on the currently logged disks, provides statistical information about particular files, and displays or alters device assignments.

GENCMD and LMCMD
GENCMD and LMCMD are used to produce CMD

memory image files suitable for execution under CP/M-86. The GENCMD utility processes Intel files, which may be produced either by Digital Research's ASM-86 or by Intel's OH86 utility. LMCMD processes Intel L-module files resulting from the standard Intel LOC86 Object Code Relocator Utility. These commands allow generation of new utilities.

BENEFITS

—Designed especially for full advantage of Intel iAPX-86/12A (8086)-based computer systems

—Complete compatibility with versions of CP/M for Intel 8080/8085-based microcomputer systems makes it easy to upgrade existing CP/M applications software to preserve software investment (see the *CP/M Operating System* data sheet for additional CP/M benefits)

—Cross-development tools, including the ASM-86 assembler and the GENCMD utility, can reside on a 8080/8085-based CP/M system; with these tools, the programmer can assemble a custom BIOS program and generate a loadable object file that runs on the target system

—Proven modular design occupies a small amount of memory to give maximum user-defined space for application programs in the available one-megabyte main memory.

—Allows up to 128 megabytes of on-line magnetic storage

—CP/M-86 manages up to 1 Mbyte of on-line RAM storage for full support of programs as large and as powerful as those found on minicomputers

—Allows multiple programs in memory for transient program execution; multiple programs may use non-contiguous memory areas for application programs

—ASM-86 and DDT-86 provide the basic tools for assembly language development and program debugging using the iAPX-86/12A (8086) microprocessor

—CP/M-86 is a complete system including the operating system and development tools; a comprehensive set of manuals describes system operation, customization, interfacing of applications programs, listings of sample programs, and operation of the assembler, debugger, and editor

—The CP/M-86 package includes full documentation for the product; documentation is also available separately; a CP/M-86 PROM loader is available for the Intel Single-Board Computer

SPECIFICATIONS

Hardware Required

—iSBC 86/12A
—iSBC 204
—CRT

Optional:

—up to 1 megabyte RAM
—up to 16 disk drives (8 megabytes each)

Documentation Package

Part Number	Description
121866	CP/M-86 system, program and user's guides

Shipping Media

(Specify by Alpha Character when ordering.)

A— Single Density (Intellec compatible, IBM 3740/1 format)

Order Code	Description

Availability: January 1982

An Intel Master Software License and Amendment required.
*CP/M is a registered trademark of Digital Research, Inc.
*CP/M-86 is a trademark of Digital Research, Inc.

MICROSOFT* MACRO-80 UTILITY SOFTWARE PACKAGE

- **Includes the MACRO-80 macro assembler, LINK-80 linking loader, and CREF-80 cross-reference facility**
- **Supports a complete, Intel-standard MACRO facility, including IRP, IRPC, REPEAT, local variables, and EXITM**
- **Supports conditional assembly, including testing of assembly pass, symbol definition, and parameters to MACROs**
- **Code is assembled in relocatable modules for easy manipulation by the LINK-80 linking loader**
- **Assembly rate of over 1000 lines per minute**
- **Provides "big computer" assembler features without sacrificing speed or memory space**
- **Provides a complete set of listing controls**
- **LINK-80 loads relocatable modules at user-specified locations**
- **CREF-80 cross-reference facility alphabetizes program variables and shows where each is defined and referenced**

The Microsoft Utility Software Package is a complete system for developing assembly language programs, routines, and subroutines. The Utility Software Package includes the MACRO-80 macro assembler, the LINK-80 linking loader, and the CREF-80 cross-reference facility. The CP/M* version also includes the LIB-80 Library Manager.

The Utility Software Package is supplied with all Microsoft compilers to provide assembly language subroutine support to main programs in the high-level programming languages. The LINK-80 linking loader is used by all Microsoft compilers for linking and loading compiled relocatable modules. Thus, LINK-80 allows the programmer to link together relocatable modules from different Microsoft languages.

FEATURES

MACRO-80 Macro Assembler

MACRO-80 incorporates almost all "big computer" assembler features without sacrificing speed or memory space. The assembler supports a complete, Intel-standard macro facility, including IRP, IRPC, REPEAT, local variables, and EXITM. Macro names take precedence over instruction mnemonics and pseudo operations. Nesting of macros is limited only by memory. Code is assembled in relocatable modules that are easily manipulated with the flexible linking loader. Conditional assembly capability is greatly enhanced by an expanded set of conditional pseudo operations that include testing of assembly pass, symbol definition, and parameters to macros. Conditionals may be nested up to 255 levels.

More MACRO-80 features:

—Comment blocks

—Variable input radix from base 2 to base 16

—Octal or hex listings

—INCLUDE statement assembles an alternate source file into the current program

—PRINTX statement for printing assembly or diagnostic messages

—PHASE/DEPHASE statements allow code to reside in one area of memory but execute in another

—Complete set of listing controls

LINK-80 Linking Loader

With LINK-80, any number of programs may be loaded with one command, relocatable modules may be loaded in user-specified locations, and external references between modules are resolved automatically by the loader. The loader also performs library searches for system subroutines and generates a memory load map showing the locations of the main program and subroutines.

CREF-80 Cross-Reference Facility

The Cross-Reference Facility that is included with the Utility Software Package supplies a convenient alphabetic list of all program variable names, along with line numbers where they are referenced and defined.

SPECIFICATIONS

Operating Environment

MACRO-80 resides in approximately 19K bytes of memory. LINK-80 resides in approximately 14K bytes of memory. CREF-80 requires about 6K bytes. The MACRO-80 Utility Software Package is compatible with the CP/M* operating system.

Required Hardware

Intellec® Microcomputer Development System
—Model 800 or
—Series II
—minimum of 1 diskette drive

Required Software

CP/M Operating System or MP/M-II* Operating System.

Documentation Package

One copy of each manual is supplied with the software package. Additional copies are available.

Part Number	Description
121797	Microsoft Utility Software Manual

Shipping Media

(Specify by Alpha Character when ordering.)
A—Single Density (Intellec® compatible, IBM 3740/1 format)
B—Double Density (Intellec® compatible, M^2FM format)

ORDERING INFORMATION

Order Code	Description
SD106CPM80A	Microsoft MACRO-80 Utility Software Package, CP/M version (Single-Density Diskette)
SC106CPM80B	Microsoft MACRO-80 Utility Software Package, CP/M version (Double-Density Diskette)

An Intel Master Software License and Amendment required.
*Microsoft is a trademark of Microsoft, Inc.
*CP/M is a registered trademark of Digital Research, Inc.
*MP/M-II is a trademark of Digital Research, Inc.

MICROSOFT* BASIC-80 INTERPRETER SOFTWARE PACKAGE

- **Compatible with other Microsoft BASIC compilers and interpreters**
- **Sophisticated string handling and structured programming features for applications development**
- **Direct transfer of BASIC programs to the 8085, 8086 and 8088**
- **Random and sequential file manipulation where random file record length is user-definable**
- **Read or write memory location capabilities**
- **Meets the requirements for the ANSI subset standard for BASIC, and supports many enhancements**
- **Extensive text editing features built-in**
- **Automatic line number generation and renumbering**
- **Supports assembly language subroutine calls**
- **Trace facilities for easier debugging**

BASIC Release 5.0 from Microsoft is an extensive implementation of BASIC. Microsoft BASIC gives users what they want from a BASIC—ease of use plus the features that are comparable to a minicomputer or large mainframe.

BASIC-80 meets the requirements for the ANSI subset standard for BASIC, as set forth in document BSRX3.60-1978. It supports many unique features rarely found in other BASICs.

FEATURES

—Four variable types: Integer (−32768, +32767), String (up to 255 characters), Single-Precision Floating Point (7 digits), Double-Precision Floating Point (16 digits).

—Trace facilities (TRON/TROFF) for easier debugging.

—Error trapping using the ON ERROR GOTO statement.

—PEEK and POKE statements to read or write any memory location.

—Automatic line number generation and renumbering, including reference line numbers.

—Matrices with up to 255 dimensions.

—Boolean operators OR, AND, NOT, XOR, EQV, IMP.

—Formatted output using the PRINT USING facility, including asterisk fill, floating dollar sign, scientific notation, trailing sign, and comma insertion.

—Direct access to I/O ports with the INP and OUT functions.

—Extensive program editing facilities via EDIT command and EDIT mode subcommands.

—Assembly language subroutine calls (up to 10 per program) are supported.

—IF/THEN/ELSE and nested IF/THEN/ELSE constructs.

—Supports variable-length random and sequential disk files with a complete set of file manipulation statements: OPEN, CLOSE, GET, PUT, KILL, NAME, MERGE.

BASIC-80 Commands, Statements, Functions

AUTO	RENUM	NAME
LIST	WIDTH	SAVE
NULL	CONT	EDIT
TROFF	MERGE	NEW
CLEAR	RUN	TRON
LOAD	DELETE	

Program Statements

CALL	RANDOMIZE	RETURN
GOSUB	COMMON	WAIT
END	DEF FN	ON GOSUB
GOTO	ERROR	DIM
STOP	POKE	FOR/NEXT/
WHILE/	RESUME	STEP
WEND	SWAP	IF/THEN/
CHAIN	DEFDBL	ELSE
DEF USR	DEFSTR	ON ERROR
LET	DEFSNG	GOTO
REM	DEFINT	OPTION BASE

Input/Output Statements and Functions

CLOSE	GET	NAME
KILL	POS	PUT
OUT	FIELD	EOF
RESTORE	LSET/RSET	SPC
READ	PRINT	INKEY$
TAB	USING	INPUT
DATA	LOC	OPEN
LINE	MKI$	CVD
INPUT	MKS$	CVI
PRINT	MKD$	CVS
WRITE	LLIST	
LPRINT	LPOS	

Arithmetic Functions

ABS	SIN	LOG
INT	CDBL	FIX
SGN	CSNG	COS
ATN	CINT	RND
EXP	SQR	TAN

String Functions

ASC	STR$	INSTR
LEN	HEX$	RIGHT$
STRING$	OCT$	MID$
CHR$	VAL	SPACE$
LEFT$		

Operators

=	*	XOR
∧	<=	NOT
<	+	EQV
>	<>	MOD
-	\	IMP
/	>=	OR
		AND

Special Functions

ERL	ERR	VARPTR
USR	FRE	PEEK

SPECIFICATIONS

Operating Environment

The standard disk version of Microsoft BASIC-80 occupies 24K bytes of memory. Microsoft BASIC-80 Interpreter is compatible with the CP/M* operating system.

Required Hardware

Intellec Microcomputer Development System
—Model 800 or
—Series II
—minimum of 1 diskette drive

Required Software

CP/M Operating System or MP/M-II* Operating System.

Documentation Package

One copy of each manual is supplied with the software package. Additional copies are available.

Part Number	Description
121806	BASIC-80 Reference Manual
121857	BASIC Reference Book

intel

MICROSOFT* COBOL-80 SOFTWARE PACKAGE

- **GSA validation at low-intermediate level implementation of ANSI 74 requirements**
- **Combines the most useful Level 1 and Level 2 ANSI 74 features, making it the most extensive implementation of COBOL for microcomputers**
- **Interactive accept/display features allow direct interaction between data, program and operator**
- **Highly efficient pseudo-code compiler produces compact, fast code**
- **Complete utility software package, including macro assembler, linking loader, and MICROSOFT M/SORT**
- **Tailorable screen-formatting facilities allow configuration to any intelligent terminal for menu and forms generation**
- **COBOL offers four kinds of files for simplicity and speed in data retrieval**

COBOL has been the language of choice for commercial data processing for over two decades, and because of its established superiority, more useful business software has been written in COBOL than in any other language. That means quality, full-function packages, many written by the field's foremost experts, are available now. Existing systems for payroll, accounts receivable, general ledger, inventory, order entry, and forecasting can run under Microsoft COBOL-80.

COBOL offers the flexibility that makes products adaptable, versatile, and available to thousands of microcomputer users. COBOL-80 not only retains the high-level features of standard COBOL, but also introduces superior interactive capabilities and user-oriented features which represent a major advance in the commercial use of COBOL. With COBOL-80, direct interaction between data, program, and operator becomes possible. The immediate acknowledgement or rejection of user-entered data facilitates equally immediate corrections and modifications.

FEATURES

COBOL-80 ANSI Standard and GSA Validated

Because COBOL is so widely used, assurances of compatibility and portability are important to all users. The American National Standards Institute has established guidelines which provide a basis for making informed comparisons of different COBOL compilers. Elements of the COBOL language are allocated to twelve different functional processing modules, at two different levels. Microsoft COBOL combines the most useful Level 1 and Level 2 features.

The United States government, through the General Services Administration (GSA), tests COBOL compilers to validate their compatibility and their implementation of the ANSI 74 standard. Microsoft COBOL for the CP/M operating system has been validated by GSA as a low-intermediate implementation of the ANSI 74 standard. Microsoft COBOL is approved for federal government installations, plus you have assurance that Microsoft COBOL will perform to specifications.

COBOL-80 supports such special features as:

Advanced verbs:
  STRING, UNSTRING, COMPUTE, SEARCH,
  PERFORM (VARYING/UNTIL)
Abbreviated and compound conditions
Sequential, Relative, and Indexed files
ASCII, packed and binary data formats
Runtime assignment of file names
Full COPY facility
Line Sequential files
Trace style debugging
COMP-3 data format
Program CHAIN
Program Segmentation
Formatted screen ACCEPT/DISPLAY
  with a single command

Program Structuring

COBOL's unique suitability to the business environment is due in large part to the structuring capabilities built into the language. COBOL-80 programs have a natural, logical organization which reflects the nature of commercial data. This efficient, clean, top-down design makes COBOL-80 programs faster to write and easier to maintain than those written in other computer languages.

```
0000-MAIN-LINE.
  PERFORM 1000-INITIALIZE.
  PERFORM 2000-ENTER ORDER
    UNTIL END-SESSION.
  IF REPORT-REQUIRED
    PERFORM 3000-PRINT SUMMARY.
  PERFORM 4000-TERMINATE.
  CHAIN "MAINMENU".

1000-INITIALIZE.
  DISPLAY SIGN-ON-SCREEN.
  ACCEPT SIGN-ON-SCREEN.
  PERFORM 1100-CHECK-SECURITY-CODE.
  MOVE ZERO TO ORDER-COUNT
    ERROR-CODE.
    SESSION-TOTAL.
```

Data Structuring

The data in COBOL programs is arranged hierarchically. Information is stored in a logical structure with direct interconnection between related pieces of data. (Note here how COBOL's PICTURE specification allows exact decimal arithmetic for precise representation of any dollar amount.)

```
05 TRANSACTION
  10 TRAN-REF.          PIC X(12).
  10 TRAN-DATE.
    15 TRAN-MM          PIC XX.
    15 TRAN-DD          PIC XX.
    15 TRAN-YY          PIC XX.
  10 POST-DATE.
    15 POST-MM          PIC XX.
    15 POST-DD          PIC XX.
    15 POST-YY          PIC XX.
  10 TRAN-AMOUNT        PIC S9(8)V99.
  10 TRAN-AMOUNT        PIC X(4).
```

Once this structure has been coded it can be stored in a file and used in any program. Only one statement is needed to retrieve it: COPY TRANSDEF.COB.

COBOL-80 obviates the need to recode every program in a system when a single definition is added or changed. Any chance of inconsistencies between programs is eliminated. At execution, each item in a COBOL-80 data structure may be referenced individually, or grouped items may be manipulated as easily. Processing code accesses the structure at any appropriate level:

```
MOVE JOURNAL-TRAN TO TRANSACTION.
MOVE CURRENT-DATE TO POST-DATE.
IF TRANS-AMOUNT<0
  MOVE TRANSACTION TO CR-TRAN.
ELSE
  MOVE TRANSACTION TO DB-TRAN.
```

Screen Handling

COBOL's move from the batch environment to online applications has brought a new emphasisis to the interaction between the application and the terminal operator. COBOL-80 provides a SCREEN SECTION for the definition of formatted screens. Special syntax is available for cursor positioning, protected and unprotected fields, highlighting, full and partial screen erase, and for defining connections between fields defined on the screen and data source/destination fields in WORKING-STORAGE. Data entry forms, menus, reports and other screens are ACCEPTed or DISPLAYed with a single command, so coding is simple. At execution time, related data items can be keyed, viewed together, and corrected before being entered into the program.

COBOL-80's screen handling facilities are Data General compatible, and learning to use the features is easy. Data descriptions are generally of the same form as in other sections of the Data Division. The screen section allows full screen descriptions without forcing allocation of WORKING-STORAGE space for unused portions of the screen.

On output, COBOL's extensive formatting capabilities provide neat, precise reports with minimal effort.

Program Chaining

COBOL-80's CHAIN verb facilitates interactive systems. Any number of separately coded applications or application segments can be reached through a main menu. These menu-driven applications which make ideal use of COBOL-80's interactive capabilities, allow smooth transfer of control from one program to another. With CHAIN, control is transferred from the menu program to any executable module as specified at runtime. COBOL-80 programs can also CALL COBOL-80 subprograms or Microsoft FORTRAN-80 or assembly language subroutines.

File Handling

COBOL-80 aims for simplicity and speed in data retrieval. COBOL-80 offers four kinds of files, so data storage can aptly correspond to the application for which it will be used. Each file type has unique characteristics:

Sequential	The fastest possible access to test, packed, and binary data in any combination.
Line Sequential	Text data in line format. Compatible with the files generated by many text editors.
Relative	Random access by record number. Direct disk access makes relative files extremely fast, yet data can be accessed in any sequence, deleted or updated interactively, and cross-referenced by key across files.
Indexed Sequential	The most powerful data access method available from any data processing language in any environment. A field within each record is chosen as a key, and the value of this key identifies a record to be read, written, updated, or deleted.

Segmentation

COBOL-80's program segmentation capability makes maximum use of memory when large programs are executed. Segmentation brings individual program sections into memory as they are needed. This allows execution of programs whose size may exceed machine memory by several times. Program segmentation is easily implemented by assigning a single number to each program section which indicates whether the section is to be resident in memory or overlaid during execution. Any section to be overlaid is automatically read into a preallocated section of memory during execution.

Microsoft COBOL-80 and the ANSI Standard

Module	Features of Microsoft COBOL
Nucleus	All of Level 1, plus these features of Level 2: CONDITIONS: Level 88 conditions with value series or range Use of logical AND/OR/NOT in conditions Use of algebraic relational symbols (<,>,=) Implied subject, or both subject and relation, in relational conditions Sign Test Nested IF statements; parentheses in conditions VERBS: Extensions to the functions of ACCEPT and DISPLAY for formatted screen handling ACCEPTance of data from DATE/DAY/TIME STRING and UNSTRING statements COMPUTE with multiple receiving fields PERFORM VARYING IDENTIFIERS: Mnemonic-names for ACCEPT or DISPLAY devices Procedure-names consisting of digits only Qualification of names (in Procedure Division statements only)

Sequential, Relative, and Indexed I/O	All of Level 1 plus these features of Level 2:
	RESERVE clause
	Multiple operands in OPEN & CLOSE, with individual options per file
	Value of FILE-ID is data-name
Sequential I/O	EXTEND mode for OPEN
	WRITE ADVANCING data name lines
	LINAGE phrase and AT END-OF-PAGE clause
Relative and Indexed I/O	DYNAMIC access mode (with READ NEXT)
	START (with key relationals EQUAL, GREATER, or NOT LESS)
Library	Level 1
Inter-Program Communication	Level 1
Table Handling	All of Level 1
	Full Level 2 formats for SEARCH statement
Debugging	Special extensions to ANSI-74 Standard providing convenient trace style debugging.
	Conditional Compilation: lines with "D in column 7" are bypassed unless "WITH DEBUGGING MODE" is given in SOURCE-COMPUTER paragraph.
Segmentation	Level 1

Utility Software Package

COBOL-80 includes the Microsoft Utility Software Package. The Utility Software Package includes the MACRO-80 macro assembler, the LINK-80 linking loader, and the CREF-80 Cross-Reference Facility. Refer to the description of the Utility Software Package for full details.

M/SORT

The COBOL-hosted version of M/SORT (M/SORT-C) is a record-sorting facility available to the programmer through the 1974 ANSI COBOL SORT/MERGE statements. For M/SORT-C, the source of the input records may be one or a set of disk files; or, records may be constructed in memory by a user-written COBOL procedure and RELEASED to M/SORT one at a time.

Similarly, the sorted output records may be automatically written to a disk file; or, records may be left in memory for processing by a user-written OUTPUT PROCEDURE within the COBOL program. M/SORT-C can load an indexed sequential file.

M/SORT-C provides for a special SORT STATUS register which can collect and return any errors encountered during a sort.

SPECIFICATIONS

Operating Environment

COBOL-80 requires about 40K bytes of user memory in addition to the operating system's space. COBOL object programs will run on systems as small as 32K bytes.

Required Hardware

Intellec Microcomputer Development System
—Model 800 or
—Series II
—minimum of 1 diskette drive

Required Software

CP/M* Operating System or MP/M-II* Operating System.

Documentation Package

One copy of each manual is provided with the software package. Additional copies may be ordered.

Part Number	Description
121801	COBOL-80 Reference Manual
121802	COBOL-80 User's Guide
121797	Microsoft Utility Software Manual
121862	M/SORT Reference Manual

intel

MICROSOFT* FORTRAN-80 SOFTWARE PACKAGE

- **Full ANSI standard FORTRAN X3.9-1966 (except the complex data type)**
- **Provides numerous enhancements to the 1966 ANSI standard**
- **Compiles several hundred statements per minute in a single pass**
- **Optimizes the generated object code by common subexpression, elimination, peephold optimization, constant folding, and branch optimizations**
- **Provides diagnostic output: descriptive error messages, error summaries, and fully symbolic listings of generated machine language**
- **Supplies an extensive library of efficient subroutines**
- **Supports user-written non-standard I/O drivers for each logical unit number to interface non-standard devices to FORTRAN programs**

Microsoft FORTRAN-80 brings the most popular science and engineering programming language to 8080/8085 microcomputers. FORTRAN-80 is comparable to FORTRAN compilers on large mainframes and minicomputers. The FORTRAN-80 package includes full ANSI Standard FORTRAN X3.9-1966 except the COMPLEX data type. With this compiler, users may take advantage of many applications programs already written in FORTRAN.

FEATURES

FORTRAN-80 enhances the 1966 ANSI Standard in several ways:

—Single-byte LOGICAL variables which can be used as integer quantities in the range +127 to −127.

—DO loops which use LOGICAL variables for tighter, faster execution of small loops.

—Mixed-mode arithmetic expressions.

—Hexadecimal constants.

—Hollerith (character) literals accepted.

—Logical operations on integer data. .AND., .OR., .NOT., .XOR., can be used for 8-bit, 16-bit, or 32-bit Boolean operations.

—READ/WRITE End-of-File or Error-Condition transfer. END=n and ERR=n (where n is the statement number) can be included in READ or WRITE statements to transfer control to the statement specified by n when an error or end-of-file is detected.

—ENCODE/DECODE for FORMAT operations to memory.

—IMPLICIT statement for redefining default variable types by specifying a type and a range of initial letters.

—INCLUDE statement for including commonly used subroutines, code, or declarations from another file.

—INTEGER*4 variables and constants using 32 bits in the range of +2,147,483,647 to −2,147,483,648.

—Support for CP/M* version 2.x providing access to a maximum of 65,535 random records in a file as large as 8 megabytes.

COMPILER DESCRIPTION

FORTRAN-80 compiles several hundred statements per minute in a single pass. It requires no more than 27K bytes of memory to compile most programs. Additional memory, when available, is used for symbol table storage and optimizations. Compiled programs are relocatable modules that are linked and loaded at runtime.

The FORTRAN-80 compiler optimizes generated object code in several ways:

Common Subexpression Elimination
: Common subexpressions are evaluated once; the value is automatically substituted in subsequent occurrences of the subexpression.

Peephole Optimization
: In special cases, small sections of code can be replaced by more compact code. Example: I=I+1 uses an INX H instruction instead of a DAD.

Constant Folding
: Integer constant expressions are evaluated at compile time.

Branch Optimizations
: The number of conditional jumps in arithmetic and logical IFs is minimized.

The compiler also provides diagnostic output. Descriptive error messages include the preceding twenty characters. At program's end, the compiler generates an error summary. A fully symbolic listing of the generated machine language is also produced. This is supplemented by tables of addresses assigned to labels, variables, and constants.

Subroutine Library

FORTRAN-80 supplies an extensive library of efficient subroutines. Only the necessary subroutines are loaded by the linker at load time. The FORTRAN library includes the following Intrinsic Functions:

ABS	DATAN	DSIN	MAX1
AINT	DATAN2	DSORT	MIN0
ALOG	DBLE	EXP	MIN1
ALOG10	DCOS	FIX	MOD
AMAX0	DEXP	FLOAT	OUT
AMAX1	DIM	IABS	PEEK
AMIN0	DLOG	IDIM	POKE
AMIN1	DLOG10	IDINT	SIGN
AMOD	DMAX1	INP	SIN
ATAN	DMIN1	INT	SNGL
ATAN2	DMOD	ISIGN	SQRT
COS	DSIGN	MAX0	TANH
DABS			

The library also contains efficient routines for 16-bit and 32-bit integer arithmetic and 32-bit and 64-bit floating point arithmetic.

Utility Software Package

The FORTRAN-80 package includes the Microsoft Utility Software Package. The Utility Software Package includes the MACRO-80 macro assembler, the LINK-80 linking loader, and the CREF-80 Cross-Reference Facility. Refer to the description of the Microsoft Utility Software Package for full details.

Custom I/O Drivers

Users may write non-standard I/O drivers for each Logical Unit Number, so interfacing non-standard devices to FORTRAN programs is straightforward.

SPECIFICATIONS

Operating Environment

The FORTRAN-80 compiler occupies approximately 24K bytes of memory, using the CP/M operating system. Most programs can be compiled within 27K bytes of additional memory. At link time, the LINK-80 linking loader occupies about 14K bytes. The Run-time Library requires 2K–14K bytes, depending on the type of program being run.

Required Hardware

Intellec Microcomputer Development System

—Model 800 or
—Series II
—minimum of 1 diskette drive

Required Software

CP/M* Operating System or MP/M-II* Operating System.

intel

MICROSOFT*BASIC-86 INTERPRETER SOFTWARE PACKAGE

- **Compatible with other Microsoft BASIC compilers and interpreters**
- **Sophisticated string handling and structured programming features for applications development**
- **Direct transfer of BASIC programs to the 8080, 8085 and 8088**
- **Random and sequential file manipulation where random file record length is user-definable**
- **Read or write memory location capabilities**
- **Meets the requirements for the ANSI subset standard for BASIC, and supports many enhancements**
- **Extensive text editing features built-in**
- **Automatic line number generation and renumbering**
- **Supports assembly language subroutine calls**
- **Trace facilities for easier debugging**

BASIC Release 5.0 from Microsoft is an extensive implementation of BASIC. Microsoft BASIC gives users what they want from a BASIC—ease of use plus the features that are comparable to a minicomputer or large mainframe.

BASIC-86 meets the requirements for the ANSI subset standard for BASIC, as set forth in document BSRX3.60-1978. It supports many unique features rarely found in other BASICs.

FEATURES

—Four variable types: Integer (−32768, +32767), String (up to 255 characters), Single-Precision Floating Point (7 digits), Double-Precision Floating Point (16 digits).

—Trace facilities (TRON/TROFF) for easier debugging.

—Error trapping using the ON ERROR GOTO statement.

—PEEK and POKE statements to read or write any memory location.

—Automatic line number generation and renumbering, including referenced line numbers.

—Matrices with up to 255 dimensions.

—Boolean operators OR, AND, NOT, XOR, EQV, IMP.

—Formatted output using the PRINT USING facility, including asterisk fill, floating dollar sign, scientific notation, trailing sign, and comma insertion.

—Direct access to I/O ports with the INP and OUT functions.

—Extensive program editing facilities via EDIT command and EDIT mode subcommands.

—Assembly language subroutine calls (up to 10 per program) are supported.

—IF/THEN/ELSE and nested IF/THEN/ELSE constructs.

—Supports variable-length random and sequential disk files with a complete set of file manipulation statements: OPEN, CLOSE, GET, PUT, KILL, NAME, MERGE.

BASIC-86 Commands, Statements, Functions

AUTO	RENUM	NAME
LIST	WIDTH	SAVE
NULL	CONT	EDIT
TROFF	MERGE	NEW
CLEAR	RUN	TRON
LOAD	DELETE	

Program Statements

CALL	RANDOMIZE	RETURN
GOSUB	COMMON	WAIT
END	DEF FN	ON GOSUB
GOTO	ERROR	DIM
STOP	POKE	FOR/NEXT/
WHILE/	RESUME	STEP
WEND	SWAP	IF/THEN/
CHAIN	DEFDBL	ELSE
DEF USR	DEFSTR	ON ERROR
LET	DEFSNG	GOTO
REM	DEFINT	OPTION BASE

Input/Output Statements and Functions

CLOSE	GET	NAME
KILL	POS	PUT
OUT	FIELD	EOF
RESTORE	LSET/RSET	SPC
READ	PRINT	INKEY$
TAB	USING	INPUT
DATA	LOC	OPEN
LINE	MKI$	CVD
INPUT	MKS$	CVI
PRINT	MKD$	CVS
WRITE	LLIST	
LPRINT	LPOS	

Arithmetic Functions

ABS	SIN	LOG
INT	CDBL	FIX
SGN	CSNG	COS
ATN	CINT	RND
EXP	SQR	TAN

String Functions

ASC	STR$	INSTR
LEN	HEX$	RIGHT$
STRING$	OCT$	MID$
CHR$	VAL	SPACE$
LEFT$		

Operators

=	*	XOR
^	<=	NOT
<	=	EQV
>	<>	MOD
-	\	IMP
/	=	OR
		AND

Special Functions

ERL	ERR	VARPTR
USR	FRE	PEEK

SPECIFICATIONS

Operating Environment

The standard disk version of Microsoft BASIC-86 occupies 32K bytes of memory. Microsoft BASIC-86 Interpreter is compatible with the CP/M-86 operating system.

Required Hardware

Microsoft* BASIC-86 Interpreter will operate with any currently supported configuration of CP/M-86 or MP/M-86.

Required Software

CP/M-86*Operating System or MP/M-86*Operating System.

Documentation Package

One copy of each manual is provided with the software package. Additional copies are available.

Part Number	Description
121810	BASIC-86 Reference Manual
121857	BASIC Reference Book

Shipping Media

(Please specify by Alpha Character when ordering.)

A—Single Density (Intellec compatible, IBM 3740/1 format)

ORDERING INFORMATION

Order Code	Description
SD203CPM86A	Microsoft BASIC-86 Interpreter Software Package, CP/M-86 version (Single-Density Diskette)

An Intel Master Software License and Amendment required.
*Microsoft is a trademark of Microsoft, Inc.
*CP/M-86 is a registered trademark of Digital Research, Inc.
*MP/M-86 is a trademark of Digital Research, Inc.

PRELIMINARY

JOVIAL-86 CROSS-COMPILER

- **IBM 370 or DEC*-10 host**
- **Full MIL-STD-1589B language implementation**
- **Extensively optimized object code**
- **Supports IEEE floating-point math with 8087 coprocessor**
- **Functional simulation capability on the host**
- **RAM/ROM separation of constants and code supported**
- **Object compatible and linkable with Intel's languages for 8086/8088**
- **Generates source symbols, lines and type information for debugging with ICE-86A**
- **Type-checking of parameters between modules optionally enforced by the timer**
- **Choice of silicon or software implementation for floating point**
- **Supports 1 MB of code space**

JOVIAL was developed in the late 1950s by Jules Schwartz and colleagues to aid in programming large complex realtime systems. The language was adopted by the U.S. Air Force in 1967 and its use is now required for programming of all embedded avionics systems.

Intel's JOVIAL Compiler implements the latest published standard of this language MIL-STD-1589B for the iAPX 86/20 microprocessor. Intel's implementation of J73 offers a complete solution for programming applications in JOVIAL from compiling modules on the mainframe to downloading software to Intel's Series III development system where a host of JOVIAL-86–compatible languages, relocation and linkage software, symbolic debugger and an in-circuit emulator (ICE-86A) are available.

FEATURES

JOVIAL language was designed for programming large complex realtime systems. The language is a derivative of Algol and has borrowed compound statement and control structures from it. JOVIAL supports fixed and floating-point, signed and unsigned integers, strings, tables and status items (enumerated type) and permits items to be any length from 1 bit to 256 bytes. The language provides access to almost any configuration of logical or arithmetic values including packed structures of flexible description. The communication pool (COMPOOL) permits sharing of system data among many subprograms by providing a centralized data and procedure description which enforces uniform usage and simplifies centralized configuration management of a JOVIAL system data base.

JOVIAL COMPILER DESCRIPTION

The JOVIAL compiler operates in multiple passes. The front-end phase analyzes the input program and translates it into an intermediate language. The global optimizer phase analyzes the intermediate code and applies extensive optimizations such as constant and value folding, multiply distribution and operator reassociation for efficient address calculation, compile time short-cut booleans, unreachable code deletion, and unnecessary store suppression. The code generator phase generates relocatable object code from this intermediate code. Other functions in the final phase are responsible for generating deficient listings and COMPOOL output.

The compiler supports development of complex and large applications by providing a comprehensive cross-reference listing that identifies the statement number where the variable is set. The compiler can also generate a statistical listing showing the distribution of various symbol table classes and intermediate language operators. The assembly language output listing displays the object program in assembly mnemonics. And optionally, the compiler can produce a reformatted source program.

PROGRAM DEVELOPMENT PROCESS

The JOVIAL source programs may be developed and maintained on the mainframe IBM 370 or DEC-10, or alternately, the source programs may be created an maintained on the Intellec Series III Microcomputer Development System, sent to the mainframe for compilation and brought back to the Series III for further processing. The ONLINE utility allows the Series III to act as a terminal to the mainframe to edit and examine files, etc. The transfer of source and object files between the mainframe and the Series III is accomplished with either the 2780/3780 emulator or the upload/download software on Series III.

The object code produced by the compiler is in Intel's standard object module format and is, therefore, compatible with the standard Series III utilities like LINK86, LOC86, LIB86 and OH86. The compatibility with subprograms written in PASCAL86, FORTRAN86, ASM86 and PL/M-86 is assured by following Intel's standard parameter passing and register usage conventions. The compiler generates source program symbols, lines and type information in the object code to allow symbolic debugging with the Series III debugger and in-circuit emulator.

The compiler generates 8087 instructions for floating-point operations. In the absence of the 8087 processor, the 8087 emulator software may be linked with the JOVIAL program to execute the floating-point instructions.

SPECIFICATIONS

Operating Environment

Intel's JOVIAL compiler runs under MVS on IBM 370 and requires a 500 Kb partition. The DEC-10 version runs under TOPS-10 operating system and requires 75K words.

Required Hardware

— Intellec Series III Microcomputer Development System
— Necessary Modems and Cables for Serial Link

Required Software

— Series III Operating System
— iAPX86 Family Utilities

Optional Software

— Mainframe Link (2780/3780 Emulator)
— 8086/8087/8088 Relocatable Assembler

Documentation Package

One copy of each manual is provided with the software package. Additional copies may be ordered.

Part No.	Description
121886	JOVIAL Language Specification
121887	JOVIAL-86 User's Guide

Shipping Media

One (1) Phase encoded, 1600 B.P.I. Magnetic Tape
One (1) Single-density Diskette (Intellec Compatible)
One (1) Double-density Diskette (Intellec Compatible)

ORDERING INFORMATION

Order Code	Description

Availability: Second Quarter 1982

*DEC and DECSYSTEM-10 are registered trademarks of Digital Equipment Corporation.



Intel Corporation
3065 Bowers Avenue
Santa Clara, CA 95051

Intel International
Rue du Moulin à Papier 51, Boite 1,
B-1160 Brussels, Belgium

Intel Japan K.K.
Flower Hill-Shinmachi East Bldg.
1-23-9, Shinmachi, Setagayu-ku
Tokyo 154, Japan

Printed in USA/C717/90K/182/RRD SZ