





# **SYSTEMS DATA CATALOG**

**JANUARY 1981**

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.

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

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

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

BXP	Intelelevision	MULTIBUS*
CREDIT	Intellec	MULTIMODULE
i	iSBC	PROMPT
ICE	iSBX	Promware
ICS	Library Manager	RMX
i <sub>m</sub>	MCS	UPI
Insite	Megachassis	μScope
Intel	Micromap	

and the combinations of ICE, ICS, iSBC, MCS or RMX and a numerical suffix.

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:

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

# Table of Contents

## Quality Assurance Flow Chart

### CHAPTER 1

#### Single Board Computers

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

### CHAPTER 2

#### iSBX™ MULTIMODULE™ Boards

iSBX 331 Fixed Floating Point Math MULTIMODULE Board .....	2-1
iSBX 332 Floating Point Math MULTIMODULE Board .....	2-7
iSBX 350 Parallel I/O MULTIMODULE Board .....	2-12
iSBX 351 Serial I/O MULTIMODULE Board .....	2-16

### CHAPTER 3

#### iCS™ Industrial Control Series

ICS 80 Industrial Chassis .....	3-1
ICS 910/920/930 Signal Conditioning/Termination Panels .....	3-6
iSBC 941 Industrial Digital Processor .....	3-14

### CHAPTER 4

#### Run-Times Systems Software

iRMX /80 Real-Time Multi-Tasking Executive .....	4-1
iSBC 801 FORTRAN Run-Time Package .....	4-6
iSBC 802 BASIC-80 Configurable RMX/80 Disk-Based Interpreter .....	4-9
iRMX 86 Operating System .....	4-13
iRMX 88 Real-Time Multi-Tasking Executive .....	4-25
iSBC 957A Intellec-iSBC 86/12A Interface and Execution Package .....	4-31

### CHAPTER 5

#### Peripheral Controllers

iSBC 202 Double Density Diskette Controller .....	5-1
iSBC 204 Single Density Flexible Diskette Controller .....	5-4
iSBC 208 Flexible Disk Controller .....	5-8
iSBC 215A/iSBC 215B Winchester Disk Controller .....	5-12
iSBC 218 Flexible Disk Controller .....	5-17
iSBC 220 SMD Disk Controller .....	5-21

### CHAPTER 6

#### Memory Expansion Boards

iSBC 016 16K-Byte RAM Memory Board .....	6-1
iSBC 032/048/064 RAM Memory Boards .....	6-3
iSBC 090 Memory System .....	6-5
iSBC 094 4K-Byte CMOS RAM Memory Battery Backup Board .....	6-9
iSBC 250 1 Megabit Bubble Memory Board .....	6-12
iSBC 254 Bubble Memory Board .....	6-15
iSBC 416 16K EPROM Expansion Board .....	6-16
iSBC 464 or (pSBC 464*) 64K-Byte EPROM Expansion Board .....	6-18
iSBC 108A/116A Combination Memory and I/O Expansion Boards .....	6-21
iSBC 300 or (pSBC 300*) 32K-Byte RAM Expansion Module iSBC 340 or (pSBC 340*) 16K-Byte EPROM Expansion Module .....	6-29
iSBC 301 4K-Byte RAM MULTIMODULE Board .....	6-30

## CHAPTER 7

### Digital I/O Expansion and Signal Conditioning Boards

ISBC 337 MULTIMODULE Numeric Data Processor .....	7-1
ISBC 501 Direct Memory Access Controller .....	7-9
ISBC 508 I/O Expansion Board .....	7-13
ISBC 517 Combination I/O Expansion Board .....	7-15
ISBC 519 or (pSBC 519*) Programmable I/O Expansion Board .....	7-19
ISBC 530 Teletypewriter Adapter .....	7-23
ISBC 556 Optically Isolated I/O Board .....	7-25
ISBC 569 Intelligent Digital Controller .....	7-27

## CHAPTER 8

### Communication Controllers

ISBC 534 or (pSBC 534*) Four Channel Communications Expansion Board .....	8-1
ISBC 544 Intelligent Communications Controller .....	8-5

## CHAPTER 9

### Analog I/O Expansion & Signal Conditioning Boards

ISBX 311 Analog Input MULTIMODULE Board .....	9-1
ISBX 328 Analog Output MULTIMODULE Expansion Board .....	9-5
ISBX 711 Analog Input Board .....	9-9
ISBX 724 Analog Output Board .....	9-13
ISBX 732 Analog Combination I/O Board .....	9-16

## CHAPTER 10

### System Packaging & Power Supplies

ISBC 604/614 or (pSBC 604/614*) Modular Cardcage/Backplane .....	10-1
ISBC 655 System Chassis .....	10-3
ISBC 660 System Chassis .....	10-5
ISBC 635 Power Supply .....	10-8
ISBC 640 Power Supply .....	10-11

## CHAPTER 11

### Microcomputer Development Systems

Introduction - New Dimensions in Development Solutions .....	11-1
Model 120 Inteltec Series II Microcomputer Development System .....	11-3
Model 225 Inteltec Series II/85 Microcomputer Development System .....	11-7
Model 286 Inteltec Series III Microcomputer Development System .....	11-12
Model 290 Network Manager Inteltec Network Development System-1 (NDS-1) .....	11-18

## CHAPTER 12

### Microcomputer Development Systems Options

Expansion Chassis Inteltec Series II Microcomputer Development System .....	12-1
Model 503 Double Density Upgrade Kit for Inteltec Microcomputer Development System .....	12-3
Model 505 Intergrated Processor Card .....	12-5
Model 556 iAPX 86 Resident Processor Board Package .....	12-8
Model 590 Network Manager Upgrade Package Inteltec Network Development System-1 (NDS-1) .....	12-11
Model 810 Software Development Module .....	12-14
Mainframe Link for Distributed Development .....	12-18
Credit CRT-Based Text Editor Microcomputer Development Systems .....	12-21

## CHAPTER 13

### Flexible and Hard Disk Systems

Inteltec Single/Double Density Flexible Disk System .....	13-1
Model 740 Inteltec Hard Disk Subsystem .....	13-5

## CHAPTER 14

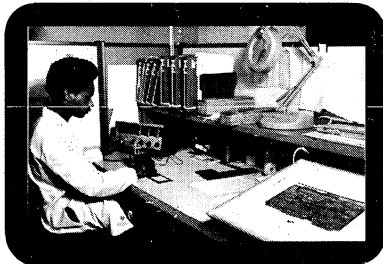
### MCS 80/85™ Development Systems and Options

FORTTRAN 80 8080/8085 ANS FORTTRAN 77 Inteltec Resident Compiler .....	14-1
Basic-80 Extended ANS 1978 Basic Inteltec Resident Interpreter .....	14-5

8080/8085 Fundamental Support Package (FSP) .....	14-8
iCIS COBOL Software Package .....	14-12
PL/M 80 High Level Programming Language Intellec Resident Compiler .....	14-16
PASCAL 80 Software Package .....	14-19
SP80 Support Package .....	14-24
SP85 Support Package .....	14-26
ICE-80, 8080 In-Circuit Emulator .....	14-28
ICE-85B, MCS-85 In-Circuit Emulator with Multi-ICE Software .....	14-34
<b>CHAPTER 15</b>	
<b>iAPX 86/88 Support Options</b>	
Series II 8086/8088 Software Development Packages .....	15-1
PL/M 86/88 Software Package .....	15-11
PASCAL 86/88 Software Package .....	15-16
8087 Software Support Package .....	15-19
8089 IOP Software Support Package .....	15-22
SP86A/SP86B Support Package .....	15-25
SP88 Support Package .....	15-27
ICE-86, 8086 In-Circuit Emulator .....	15-30
ICE-88, 8088 In-Circuit Emulator .....	15-36
<b>CHAPTER 16</b>	
<b>Prototype Microcomputer Kits</b>	
SDK-85, MCS-85 System Design Kit .....	16-1
SDK-86, MCS-86 System Design Kit .....	16-7
SDK-C86, MCS-86 System Design Kit Software and Cable Interface to Intellec Development System .....	16-13
<b>CHAPTER 17</b>	
<b>MCS-48™ Development Systems</b>	
MCS-48 Diskette-Based Software Support Package .....	17-1
ICE-49, MCS-48 In-Circuit Emulator .....	17-3
HSE-49 High Speed Emulator .....	17-7
EM1 8021 Emulation Board .....	17-13
EM2 8022 Emulation Board .....	17-16
ICE-22, 8022 In-Circuit Emulator .....	17-19
<b>CHAPTER 18</b>	
<b>MCS-51™ Development Systems</b>	
8051 Software Development Package .....	18-1
<b>CHAPTER 19</b>	
<b>UPI-41A™ Development Systems</b>	
ICE-41A, UPI-41A In-Circuit Emulator .....	19-1
<b>CHAPTER 20</b>	
<b>2920 Signal Processor Development Systems</b>	
2920 Software Support Package .....	20-1
<b>CHAPTER 21</b>	
<b>Memory Systems</b>	
Introduction .....	21-1
Series 90 General Purpose Memory System .....	21-2
in-5770 Video Refresh Memory System .....	21-5
in-1670, PDP*-11/70 Add-On Memory System .....	21-11
in-5150 Eclipse Add-In Memory .....	21-15
in-5160 Nova 3 Add-In Memory .....	21-19
MU-5780 VAX-11/780 Add-In Memory Card .....	21-24
<b>CHAPTER 22</b>	
Insite™ User's Program Library .....	22-1
<b>CHAPTER 23</b>	
Intel Microcomputer Workshops .....	23-1
<b>CHAPTER 24</b>	
Product Service .....	24-1

# QUALITY ASSURANCE

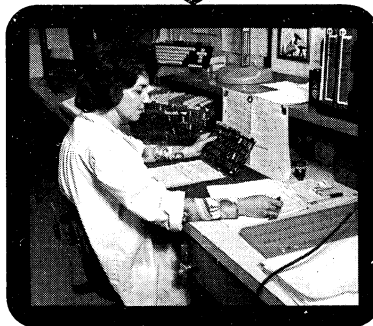
A typical product flow



*Gold Thickness Measurement on Raw Boards*

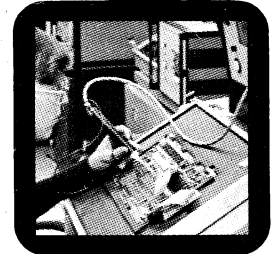


*Kit Audit*

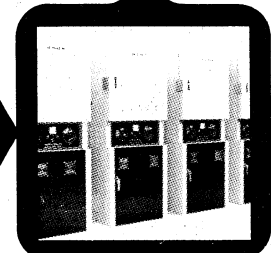


*Visual Inspection of Printed Wire Assemblies*

*Automatic Test Systems Utilized in Board Testing*



*Single Board Computer Test Fixture*



*Ovens Utilized for Board Pre-Bake*

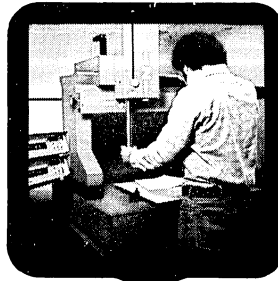
*Cable Testing Using an Automatic Test System*



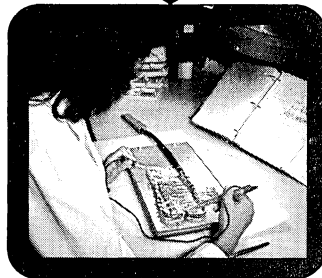
*Checking Alignment of Floppy Disk Drives*



*XYZ Measurement of Sheet Metal*



**ASSEMBLY**



*In-Process QC of System in Assembly*

**SYSTEM TEST**



*Aging in System Test*



*Boot-Up Test in System Test*



*Shipping Inspection*







*Single Board  
Computers*

---

**1**

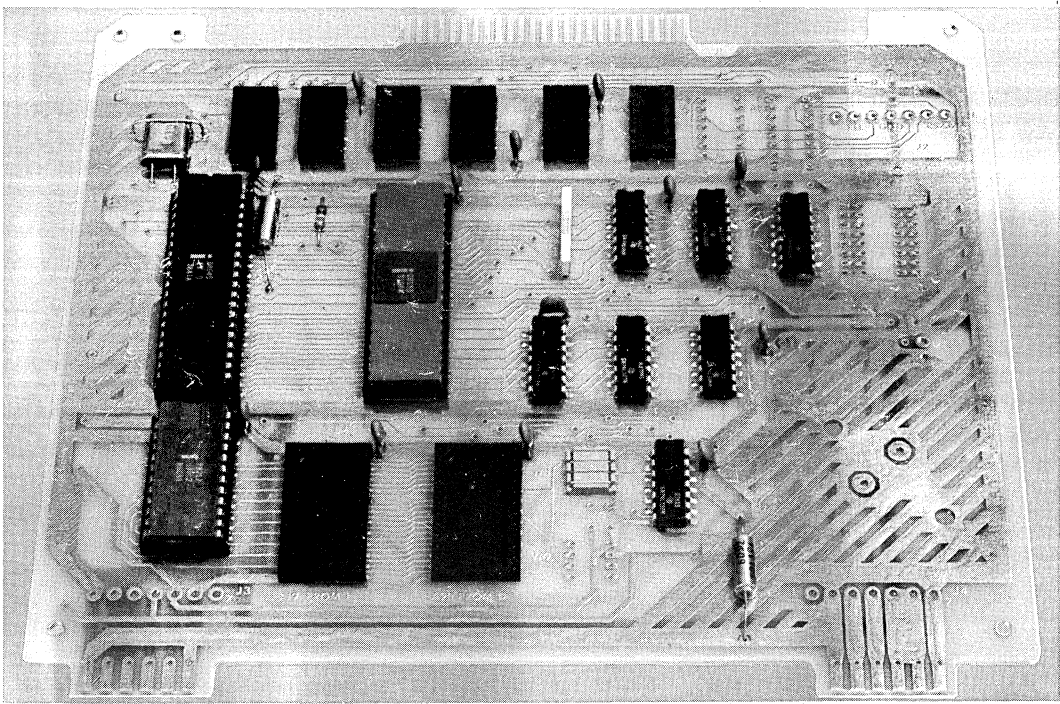
THE UNIVERSITY OF CHICAGO  
LIBRARY



## 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) or Intel 2316E masked ROMs. Optionally, if only 2K bytes are required, read only memory may be added in 1K-byte increments using Intel 2708 EPROMs or Intel 2608 masked ROMs.

**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 back-plane.

**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

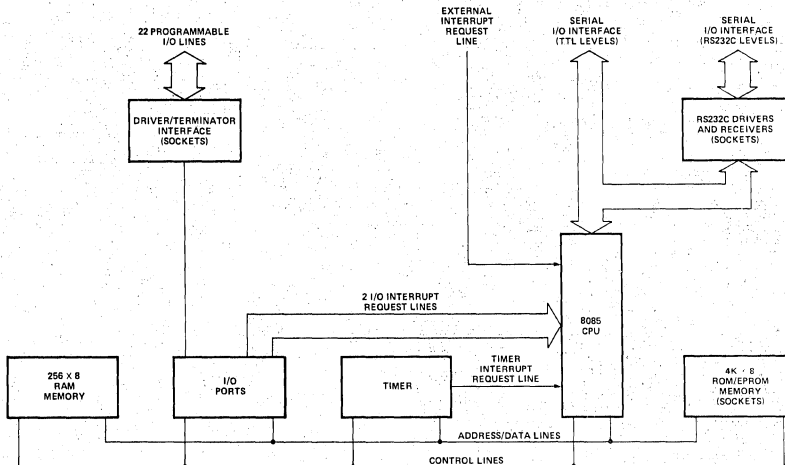


Figure 1. iSBC Block Diagram Showing Functional Components

# iSBC 80/04

**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 <sup>1</sup>
4	3	X		X		X <sup>2</sup>

**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.

identical 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/I/O 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 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 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<sub>16</sub>, RST 7.5: 3C<sub>16</sub>, RST 6.5: 34<sub>16</sub>, RST 5.5: 2C<sub>16</sub>). 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.

**FORTAN-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  $\mu$ s,  $\pm$  0.1%

#### Note

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

### Memory Addressing

**ROM/EPROM** — 0-0FFF<sub>H</sub>  
**RAM** — 3F00<sub>H</sub>

### 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 <sub>16</sub>	Highest	Non-maskable
Timer	RST 7.5	3C <sub>16</sub>	↑ ↓ Lowest	Maskable
I/O Port 2	RST 6.5	34 <sub>16</sub>		Maskable
I/O Port 1	RST 5.5	2C <sub>16</sub>		Maskable

### Timer

**Input Frequency Reference** — 122.88 kHz  $\pm$  0.1% (8.14  $\mu$ s period nominal)

### Output Frequencies/Timing Intervals

Function	Timer/Counter	
	Min	Max
Programmable pulse	8.14 $\mu$ s	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 $\mu$ s	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  $\pm$  0.1%

**Connectors**

Interface		Pins (no.)	Center (in.)	Mating Connectors <sup>1</sup>
Voltages	+5V, +12V, -5V <sup>2</sup>	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 <sup>3</sup>	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**

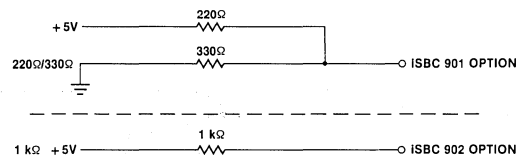
**I/O Drivers** — The following line drivers are all compatible with the I/O driver sockets on the iSBC 80/04:

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)



# iSBC 80/04

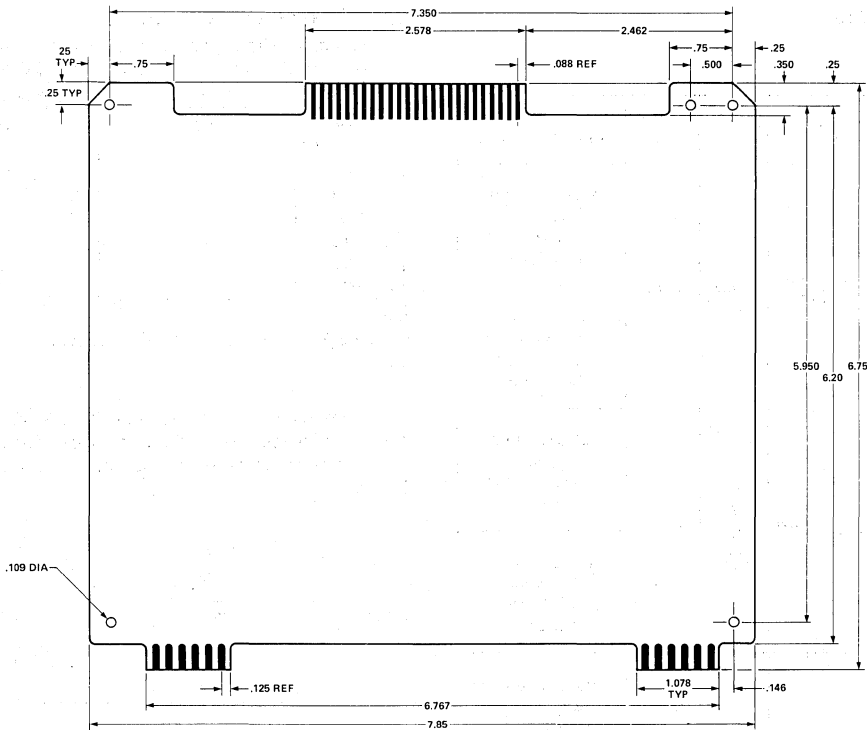


Figure 2. iSBC 80/04 Dimensions

## Electrical Characteristics

### DC Power Requirements

Voltage (± 5%)	Without PROM <sup>1</sup> (max)	With 2716 EPROM <sup>2</sup> (max)	With 2708 EPROM <sup>3</sup> (max)
V <sub>CC</sub> = +5V	I <sub>CC</sub> = 600 mA	1.45A	1.25A
V <sub>DD</sub> = +12V <sup>4</sup>	I <sub>DD</sub> = 0	7 mA <sup>5</sup>	137 mA
V <sub>BB</sub> = -5V <sup>4</sup>	I <sub>BB</sub> = 0	0	90 mA
V <sub>AA</sub> = -12V <sup>5</sup>	I <sub>AA</sub> = 0	23 mA <sup>5</sup>	23 mA <sup>5</sup>

### Notes

- Does not include power required for optional EPROM/ROM, I/O drivers, and I/O terminators.
- With two Intel 2716 EPROMs and 220Ω/330Ω terminators installed for 22 input ports; all terminator inputs low.
- With two Intel 2708 EPROMs and 220Ω/330Ω terminators installed for 22 input ports; all terminator inputs low.
- Required for 2708 EPROMs.
- 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

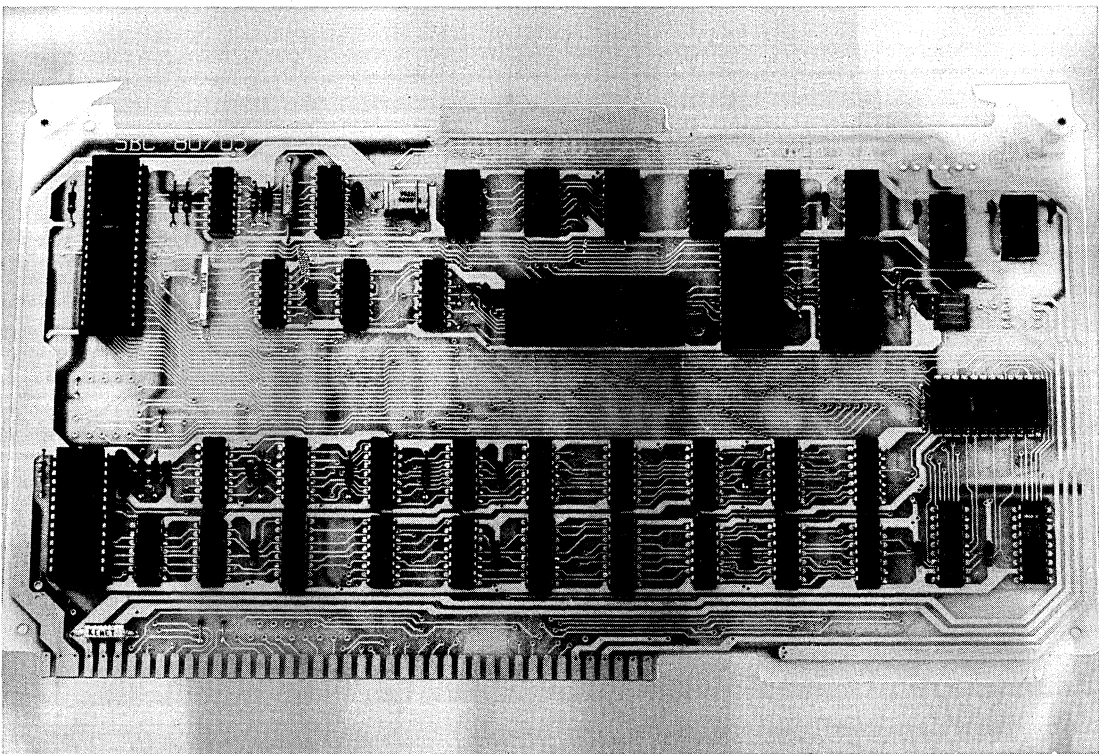
Part Number	Description
SBC 80/04	Single Board Computer



## 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 x 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.



## 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) or Intel 2316E masked ROMs. Optionally, if only 2K bytes are required, read only memory may be added in 1K-byte increments using Intel 2708 EPROMs or Intel 2608 masked ROMs.

### 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/I/O/Timer. The system software is used to con-

figure 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

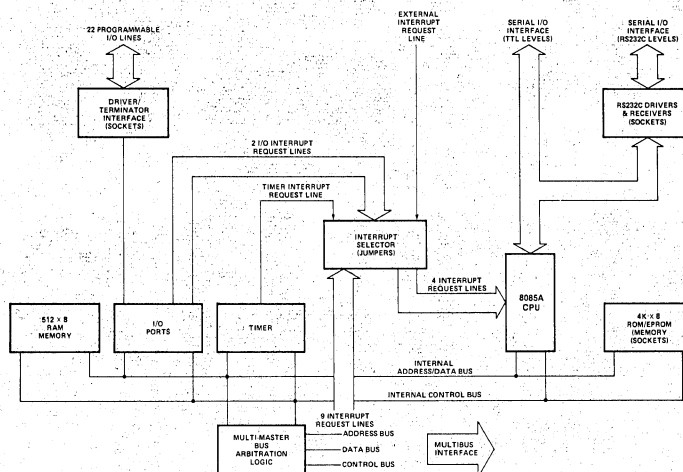


Figure 1. iSBC 80/05 Block Diagram Showing Functional Components

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 <sup>1</sup>
4	3	X		X		X <sup>2</sup>

**Notes**

- 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.
- 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 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 80/05 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<sub>16</sub>, RST 7.5: 3C<sub>16</sub>, RST 6.5: 34<sub>16</sub>, RST 5.5: 2C<sub>16</sub>). A single 8085A jump

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.

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.

**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 310 High Speed Mathematics Unit. Memory may be expanded to 65,536 bytes by adding user specified combinations of RAM boards, EPROM boards, or combinations 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 sin-

## iSBC 80/05

gle 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 editor, 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  $\mu$ s,  $\pm$  0.1%

#### Note

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

### Memory Addressing

**ROM/EPROM** — 0-0FFF<sub>H</sub>

**RAM** — 3E00<sub>H</sub>

### 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 <sub>16</sub>	Highest	Non-maskable
RST 7.5	3C <sub>16</sub>	$\updownarrow$	Maskable
RST 6.5	34 <sub>16</sub>		Maskable
RST 5.5	2C <sub>16</sub>		Maskable
		Lowest	

### Timer

**Input Frequency Reference** — 122.88 kHz  $\pm$  0.1% (8.14  $\mu$ s period nominal)

### Output Frequencies/Timing Intervals

Function	Timer/Counter	
	Min	Max
Programmable pulse	8.14 $\mu$ s	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 $\mu$ s	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  $\pm$  0.1%

## iSBC 80/05

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 <sup>1</sup>	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

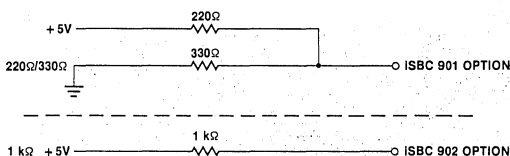
**I/O Drivers** — The following line drivers are all compatible with the I/O driver sockets on the iSBC 80/05:

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 <sup>1</sup> (max)	With 2716 EPROM <sup>2</sup> (max)	With 8708 EPROM <sup>3</sup> (max)
V <sub>CC</sub> = +5V	I <sub>CC</sub> = 1.80 mA	2.65A	2.45A
V <sub>DD</sub> = +12V <sup>4</sup>	I <sub>DD</sub> = 0	7 mA <sup>5</sup>	137 mA
V <sub>BB</sub> = -5V <sup>4</sup>	I <sub>BB</sub> = 0	0	90 mA
V <sub>AA</sub> = -12V <sup>5</sup>	I <sub>AA</sub> = 0	23 mA <sup>5</sup>	23 mA <sup>5</sup>

### Notes

- Does not include power required for optional EPROM/ROM, I/O drivers, and I/O terminators.
- With two Intel 2716 EPROMs and 220Ω/330Ω terminators installed for 22 input ports; all terminator inputs low.
- With two Intel 2708 EPROMs and 220Ω/330Ω terminators installed for 22 input ports; all terminator inputs low.
- Required for 2708 EPROMs.
- 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

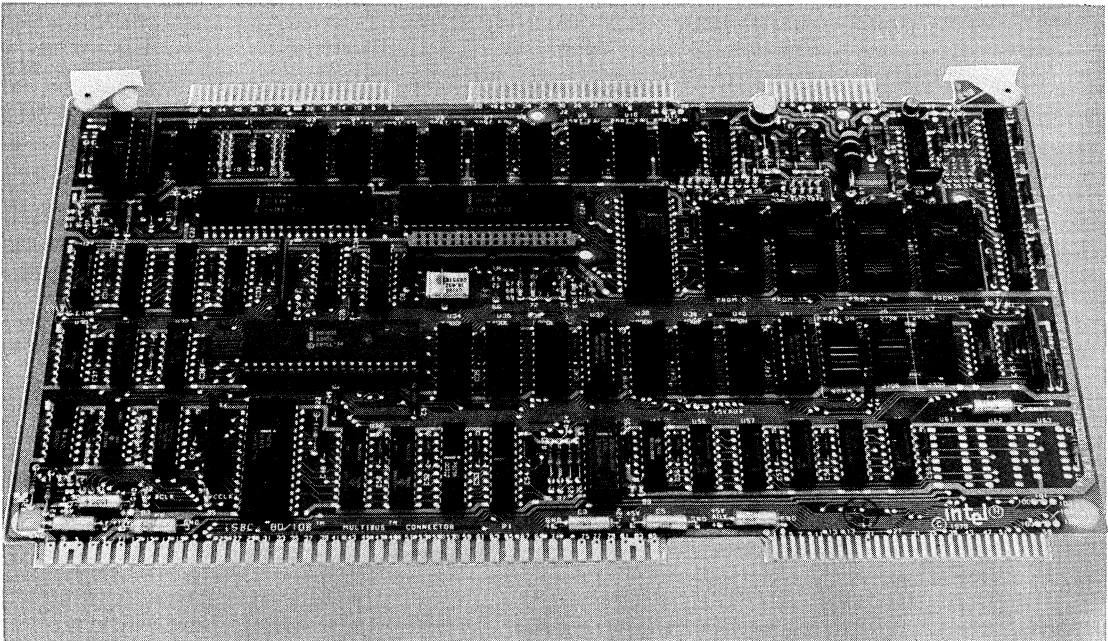
Part Number	Description
SBC 80/05	Single Board Computer



## 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 socket 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.

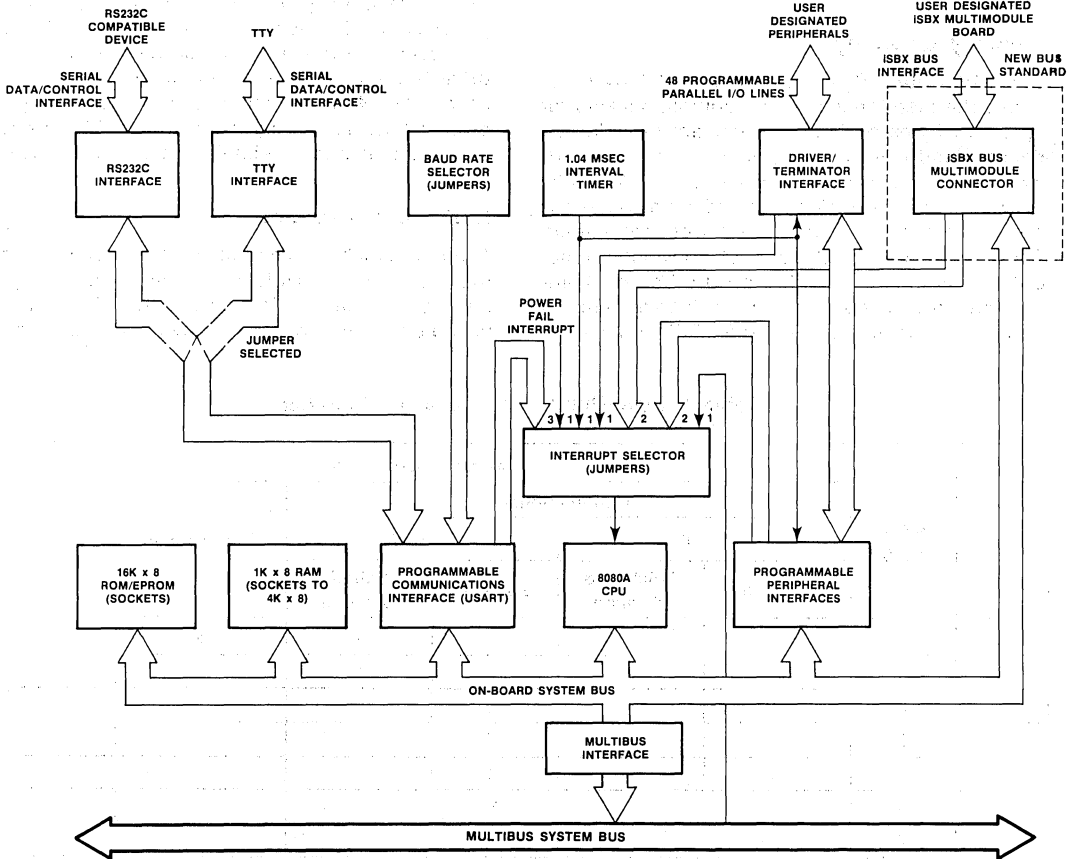


Figure 1. iSBC 80/10B Single Board Computer Block Diagram



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 1Kx4 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, 2758, or 2608); in

2K byte increments up to 8K bytes (using Intel 2716 or 2316A); 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 <sup>1</sup>
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 originate 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<sub>16</sub>.

### 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 RMX/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 RMX/80 package. User configurability is aided on the Intellec microcomputer development system by the Interactive Configuration Utility program provided with the RMX/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 RMX/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 RMX/80, and to the iSBC 310 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 RMX/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 RMX/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 RMX/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  $\mu$ sec

**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, 2316E

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 Port B Port C Control	E4 E5 E6 E7
8255A No. 2 Port A Port B Port C Control	E8 E9 EA EB
8251A Data Control	EC ED
iSBX Multimodule MCS0 MCS1	F0-F7 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)
307.2	—	÷ 16    ÷ 64 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).

### Electrical Characteristics

#### DC Power Requirements

Voltage	Without EPROM <sup>1</sup>	With 2708 EPROM <sup>2</sup>	With 2758, 2716, or 2732 EPROM <sup>3</sup>	Power Down Requirements (RAM and Support Circuit)
V <sub>CC</sub> = +5V ±5%	I <sub>CC</sub> = 2.0A	3.1 A	3.46 A	84 mA + 140 mA/K (2114A-5)
V <sub>DD</sub> = +12V ±5%	I <sub>DD</sub> = 150 mA	400 mA	150 mA	Not Required
V <sub>BB</sub> = -5V ±5%	I <sub>BB</sub> = 2 mA	200 mA	2 mA	Not Required
V <sub>AA</sub> = -12V ±5%	I <sub>AA</sub> = 175 mA	175 mA	175 mA	Not Required

#### NOTES:

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

### 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)

**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:

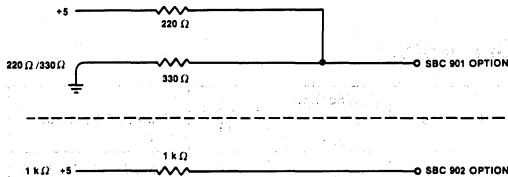
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

**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**

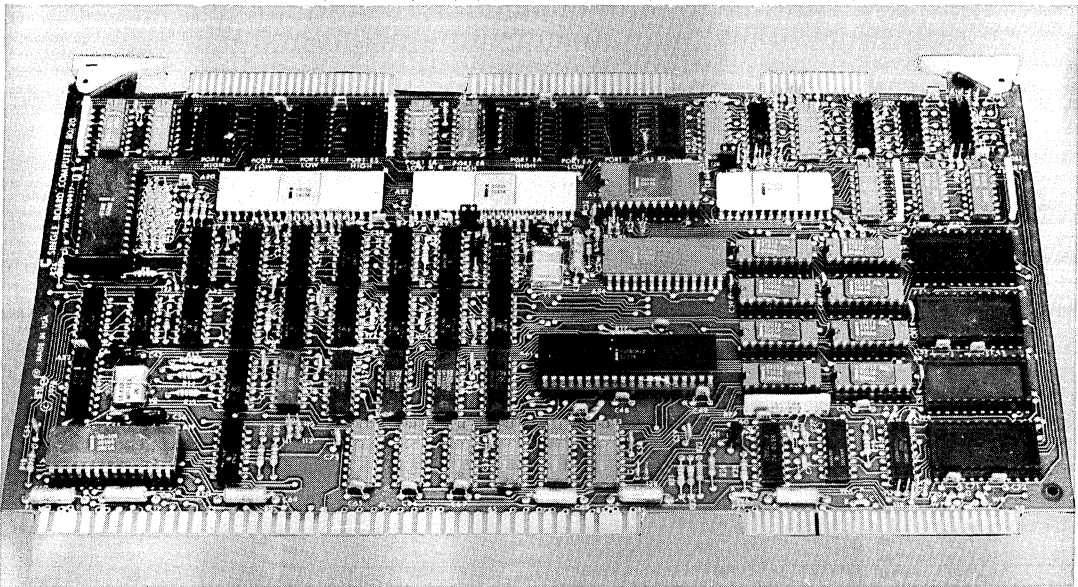
Part Number	Description
SBC 80/10B	Single Board Computer



## iSBC 80/20-4 or (pSBC80/20-4\*) 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 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, two programmable timers, MULTIBUS control logic, and bus expansion drivers all reside on each board.



## 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 Intel 2608 ROMs, or read only memory may be added in 2K-byte increments using Intel 2716 EPROMs or Intel 2316E masked ROMs. 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

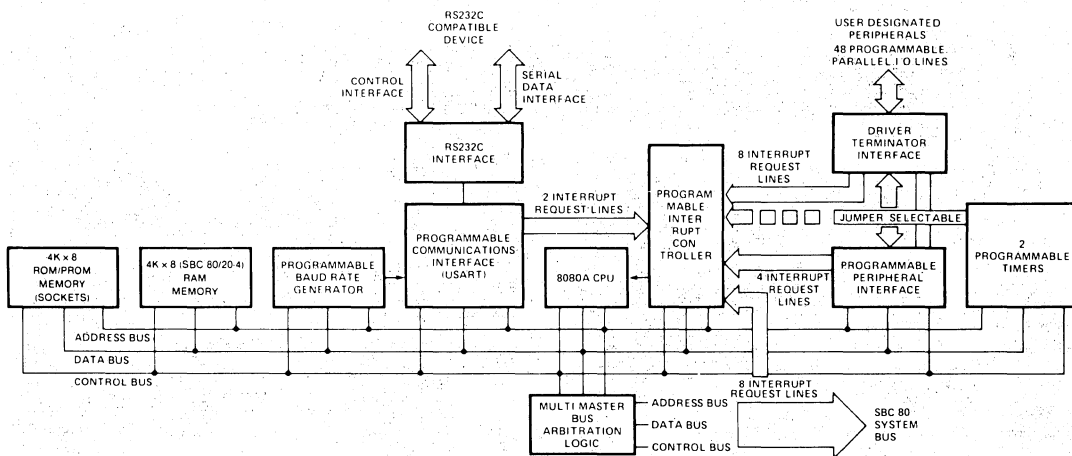


Figure 1. iSBC 80/20 and iSBC 80/20-4 Block Diagram Showing Functional Components

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. The iSBC 530 Teletypewriter Adapter provides an optically isolated interface for those systems requiring a 20 mA current loop. The iSBC 530 may be used to interface the iSBC 80/20-4 to teletypewriters or other 20 mA current loop equipment.

**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				Bidirectional	Control
		Unidirectional					
		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 <sup>1</sup>
	4	X		X			X <sup>1</sup>
4	8	X	X	X	X	X	
5	8	X	X	X	X		
6	4	X		X			X <sup>2</sup>
	4	X		X			X <sup>2</sup>

**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 addresses 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 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 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 310 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 RMX/80 Real-Time Multi-Tasking Executive. iSBC 80/20-4 based user programs (tasks) can take advantage of the RMX/80 executive to do all necessary scheduling, inter-task communication, and memory space allocation. RMX/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 Intel micro-computer development system. The resident macro-assembler, 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 Intel 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.

**FORTTRAN-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 Intel 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  $\mu$ s

**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<sub>H</sub>, 7FFF<sub>H</sub>, BFFF<sub>H</sub>, ... FFFF<sub>H</sub>)

### 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 (DEH).

### 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.

## iSBC 80/20-4

### 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.

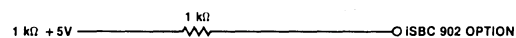
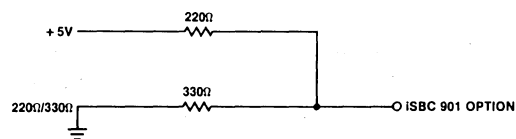
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

**Note**

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

Ports 1 and 4 have 20 mA totem-pole bidirectional drivers and 1 k $\Omega$  terminators.

**I/O Terminators** — 220 $\Omega$ /330 $\Omega$  divider or 1 k $\Omega$  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 ( $\pm 5\%$ )	Without PROM <sup>1</sup> (max)	With 4K PROM <sup>2</sup> (max)	With ISBC 530 <sup>3</sup> (max)	RAM Only <sup>4</sup> (max)	With 8K PROM <sup>5</sup> (max)
V <sub>CC</sub> = +5V	I <sub>CC</sub> = 4.0A	4.9A	4.9A	1.1A	5.2A
V <sub>DD</sub> = +12V	I <sub>DD</sub> = 90 mA	350 mA	450 mA	—	90 mA
V <sub>BB</sub> = -5V	I <sub>BB</sub> = 2 mA	180 mA	180 mA	—	2 mA
V <sub>AA</sub> = -12V	I <sub>AA</sub> = 20 mA	20 mA	120 mA	—	20 mA

**Notes**

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

### Environmental Characteristics

**Operating Temperature** — 0°C to 55°C

### 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

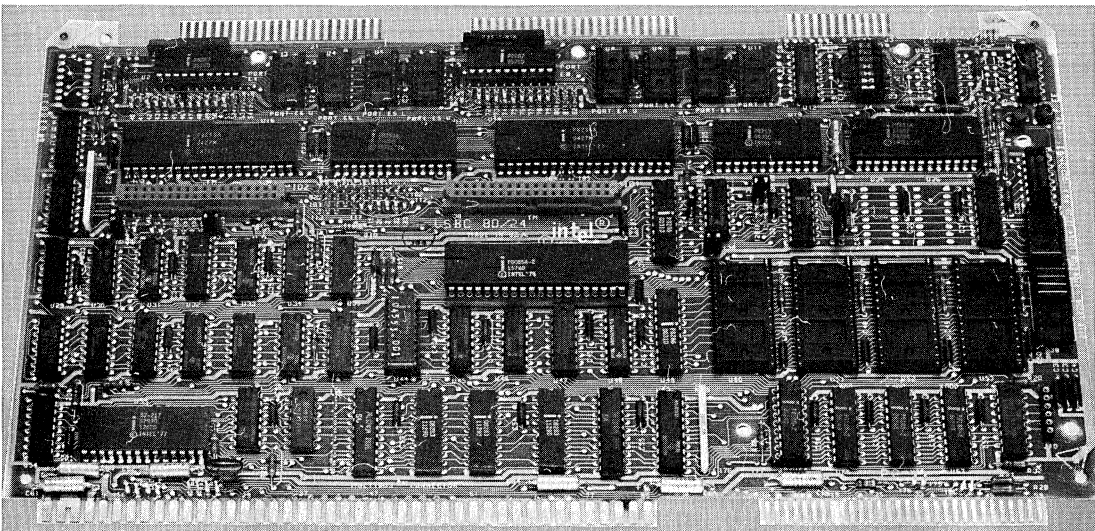
SBC 80/20-4 Single Board Computer with 4K bytes RAM



## 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 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, 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.

**FUNCTIONAL DESCRIPTION**

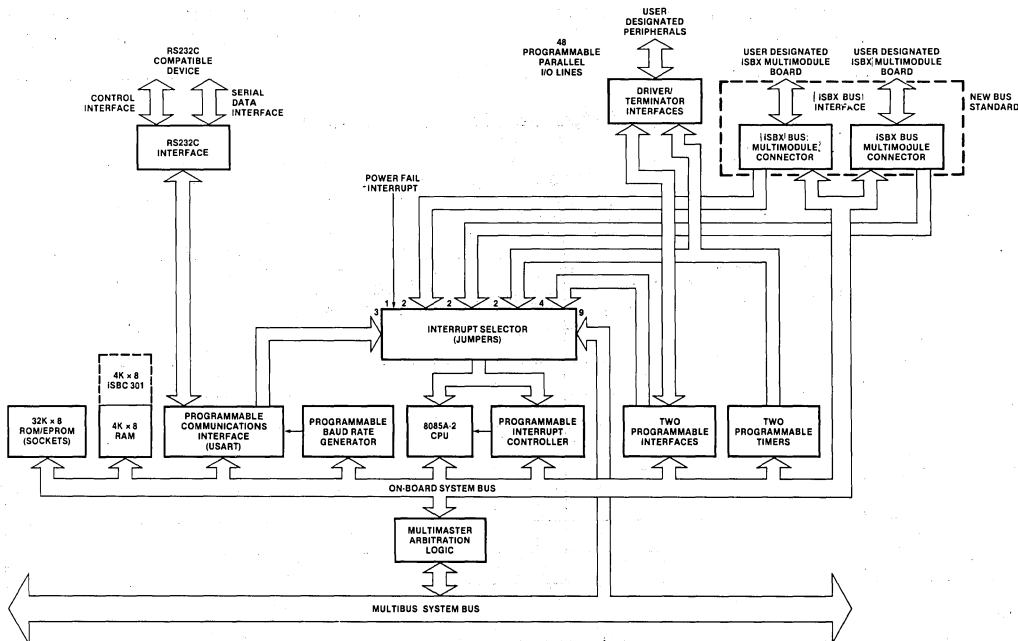
**Central Processing Unit**

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**

## 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				Bidirectional	Control
		Unidirectional					
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X		
2	8	X	X	X	X		
3	4	X		X			X <sup>1</sup>
	4	X		X			X <sup>1</sup>
4	8	X	X	X	X	X	
5	8	X	X	X	X		
6	4	X		X			X <sup>2</sup>
	4	X		X			X <sup>2</sup>

**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. The iSBC 530 Teletypewriter Adapter provides an optically isolated interface for those systems requiring a 20 mA current loop. The iSBC 530 adapter may be used to interface the iSBC 80/24 board to teletypewriters or other 20 mA current loop equipment.

**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 RMX/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 RMX/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 RMX/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 RMX/80, and to the iSBC 310 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 RMX/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 RMX/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 RMX/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  $\mu$ sec (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 Port B Port C Control	E4 E5 E6 E7
8255A No. 2 Port A Port B Port C Control	E8 E9 EA EB
8251A Data Control	EC, EE ED, EF
iSBX MULTIMODULE J5 MCS0 MCS1	C0-C7 C8-CF
iSBX MULTIMODULE J6 MCS0 MCS1	F0-F7 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<sub>H</sub>).

### 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	↓ Lowest	Maskable
RST 6.5	34		Maskable
RST 5.5	2C		Maskable

# iSBC 80/24

## 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  $\pm$  0.1% (0.930  $\mu$ sec 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 $\mu$ sec	60.948 msec	3.72 $\mu$ sec	1.109 hrs
Programmable One-Shot	1.86 $\mu$ sec	60.948 msec	3.72 $\mu$ sec	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 $\mu$ sec	60.948 msec	3.72 $\mu$ sec	1.109 hrs
Hardware Triggered Strobe	1.86 $\mu$ sec	60.948 msec	3.72 $\mu$ sec	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  $\pm$  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

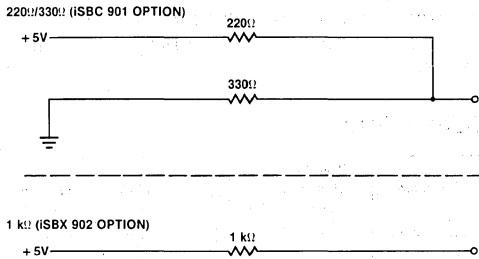
**NOTE:**

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

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

**I/O Terminators**—220 $\Omega$ /330 $\Omega$  divider or 1 k $\Omega$  pullup

# iSBC 80/24



- 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.
- 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.

## 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 <sub>CC</sub> = +5V ±5% (max)	V <sub>DD</sub> = +12V ±5% (max)	V <sub>BB</sub> = -5V ±5% (max)	V <sub>AA</sub> = -12V ±5% (max)
Without EPROM <sup>1</sup>	3.34A	40 mA	—	20 mA
RAM Only <sup>2</sup>	0.14A	—	—	—
With iSBC 530 <sup>3</sup>	3.34A	140 mA	—	120 mA
With 4K EPROM <sup>4</sup> (using 2708)	3.74A	300 mA	180 mA	20 mA
With 4K EPROM <sup>4</sup> (using 2758)	4.43A	40 mA	—	20 mA
With 8K EPROM <sup>4</sup> (using 2716)	4.43A	40 mA	—	20 mA
With 16K EPROM <sup>4</sup> (using 2732)	4.71A	40 mA	—	20 mA
With 32K EPROM <sup>4</sup> (using 2764)	4.71A	40 mA	—	20 mA

### NOTES:

- Does not include power for optional EPROM, I/O drivers, and I/O terminators.
- RAM chips powered via auxiliary power bus.

## ORDERING INFORMATION

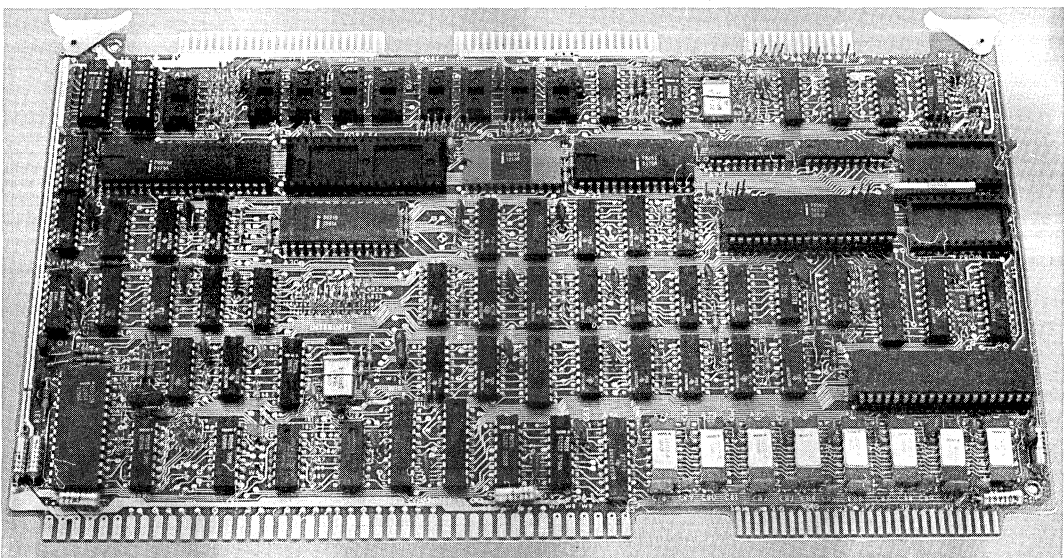
Part Number	Description
SBC 80/24	Single Board Computer



## 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 x 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.



**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

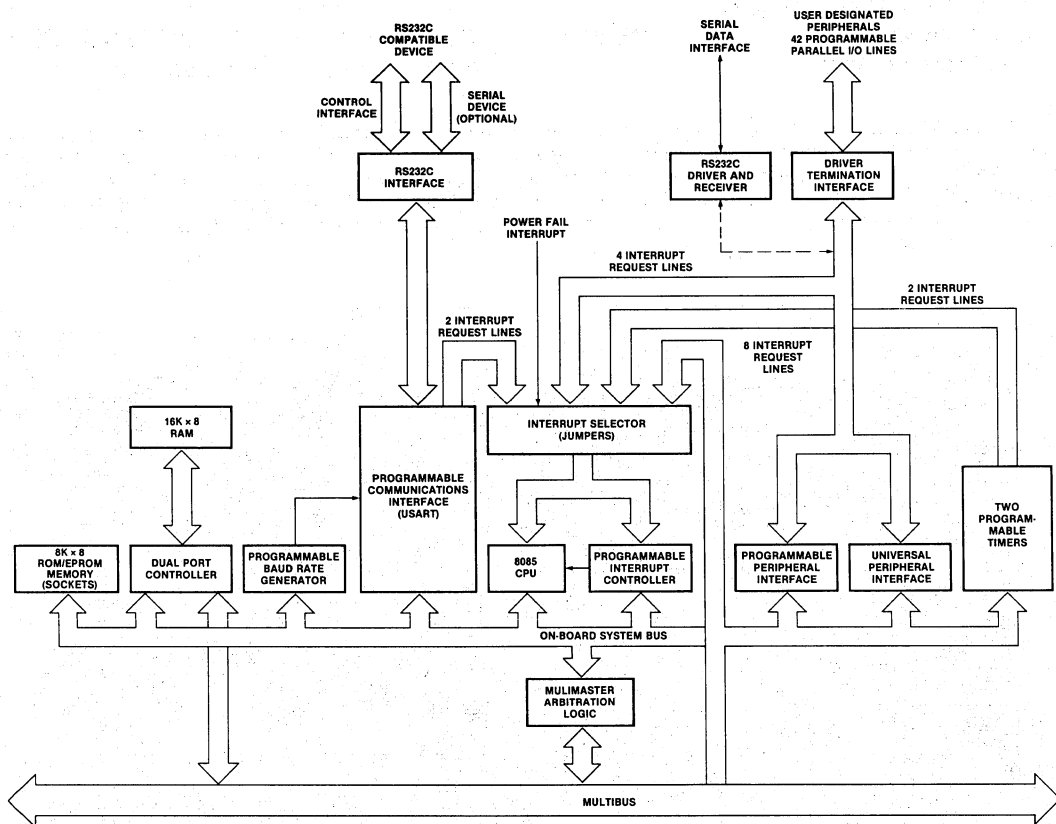


Figure 1. iSBC 80/30 Single Board Computer Block Diagram

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) or Intel 2608 ROMs; in 2K-byte increments up to a maximum of 4K bytes using Intel 2716 EPROMs or Intel 2316E 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.

**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 flexibil-

ity 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				Bidirectional	Control
		Unidirectional					
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X		
2	8	X	X	X	X		
3	4	X		X		X <sup>1</sup>	
	4	X		X		X <sup>1</sup>	

**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. The IIBC 530 Teletypewriter Adapter provides an optically isolated interface for those systems requiring a 20 mA current loop. The IIBC 530 may be used to interface the IIBC 80/30 to teletypewriters or other 20 mA current loop equipment.

**Multimaster Capability**

The IIBC 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 IIBC 80/30 provides full MULTIBUS arbitration control logic. This control logic allows up to three IIBC 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 IIBC 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 IIBC 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 IIBC 80/30 RS232C

USART serial port. In utilizing the IIBC 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 IIBC 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 IIBC 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 periph-

eral 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 using Intel MULTIBUS compatible expansion boards. High speed integer and floating point arithmetic capabilities may be added by using the ISBC 310 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 sub-cages are available to support multiboard systems.

**Real-Time Software**

Intel's RMX/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 RMX/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 Inteltec 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 Inteltec 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 RMX/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 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  $\mu$ s

#### 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 or 2608 ROMs); 0-0FFF (using 2716 EPROMs or 2316E ROMs); 0-1FFF (using 2732 ROMs)

**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	
153.6	—	+ 16	+ 64
76.8	—	9600	2400
38.4	38400	4800	1200
19.2	19200	2400	600
9.6	9600	1200	300
4.8	4800	600	150
2.4	2400	300	75
1.76	1760	150	—
		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		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  $\pm$  0.1% (0.041  $\mu$ s period, nominal); 1.23 MHz  $\pm$  0.1% (0.81  $\mu$ s period, nominal); or 153.60 kHz  $\pm$  0.1% (6.51  $\mu$ s 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 $\mu$ s	427.1 ms	3.26 $\mu$ s	466.50 min
Programmable one-shot	1.63 $\mu$ s	427.1 ms	3.26 $\mu$ 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 $\mu$ s	427.1 ms	3.26 $\mu$ s	466.50 min
Hardware triggered strobe	1.63 $\mu$ s	427.1 ms	3.26 $\mu$ s	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  $\pm$  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.

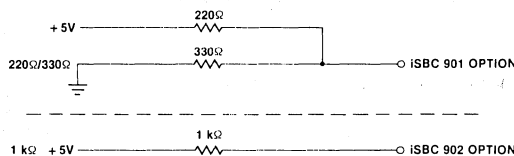
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

**Note**

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

Port 1 of the 8255A has 20 mA totem-pole bidirectional drivers and 1 k $\Omega$  terminators.

**I/O Terminators** — 220 $\Omega$ /330 $\Omega$  divider or 1 k $\Omega$  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 <sub>CC</sub> = +5V ± 5%(max)	V <sub>DD</sub> = +12V ± 5%(max)	V <sub>BB</sub> = -5V ± 5%(max)	V <sub>AA</sub> = -12V ± 5%(max)
Without EPROM <sup>1</sup>	I <sub>CC</sub> = 3.5A	I <sub>DD</sub> = 220 mA	I <sub>BB</sub> = —	I <sub>AA</sub> = 50 mA
With 8041/8741 <sup>2</sup>	3.6A	220 mA	—	50 mA
RAM only <sup>3</sup>	350 mA	20 mA	2.5 mA	—
With iSBC 530 <sup>4</sup>	3.5A	320 mA	—	150 mA
With 2K EPROM <sup>5</sup> (using 8708)	4.4A	350 mA	95 mA	40 mA
With 2K EPROM <sup>5</sup> (using 2758)	4.6A	220 mA	—	50 mA
With 4K EPROM <sup>5</sup> (using 2716)	4.6A	220 mA	—	50 mA
With 8K EPROM <sup>5</sup> (using 2332)	4.6A	220 mA	—	50 mA

**Notes**

- Does not include power required for optional EPROM/ROM, 8041A/8741A I/O drivers, and I/O terminators.
- Does not include power required for optional EPROM/ROM, I/O drivers and I/O terminators.
- RAM chips powered via auxiliary power bus
- 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.
- Includes power required for two EPROM/ROM chips, 8041A/8741A and 220 $\Omega$ /330 $\Omega$  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**

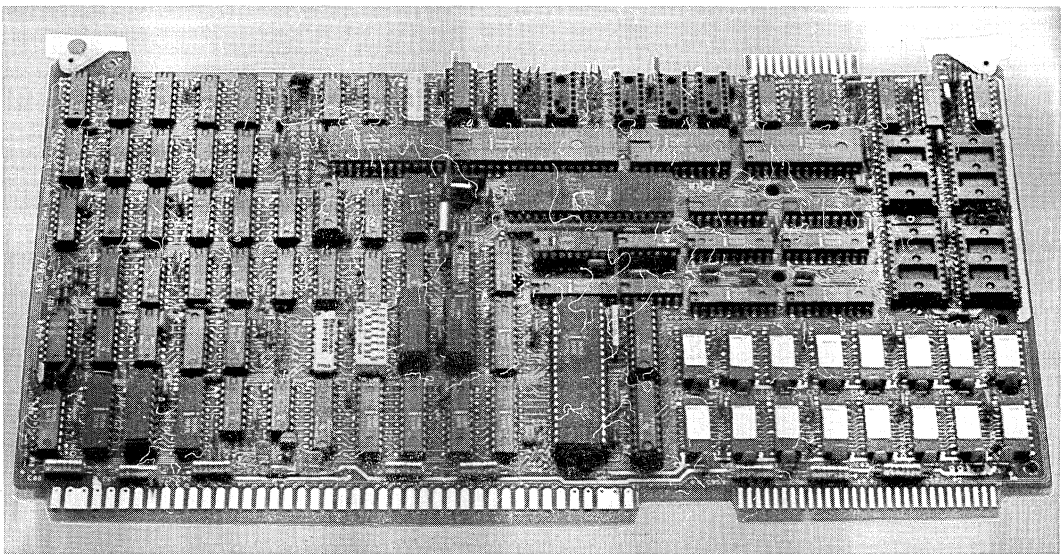
Part Number	Description
SBC 80/30	Single Board Computer with 16K bytes RAM



## iSBC 86/12A 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 mega-byte
- 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 MULTI-MODULE 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.



## 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.

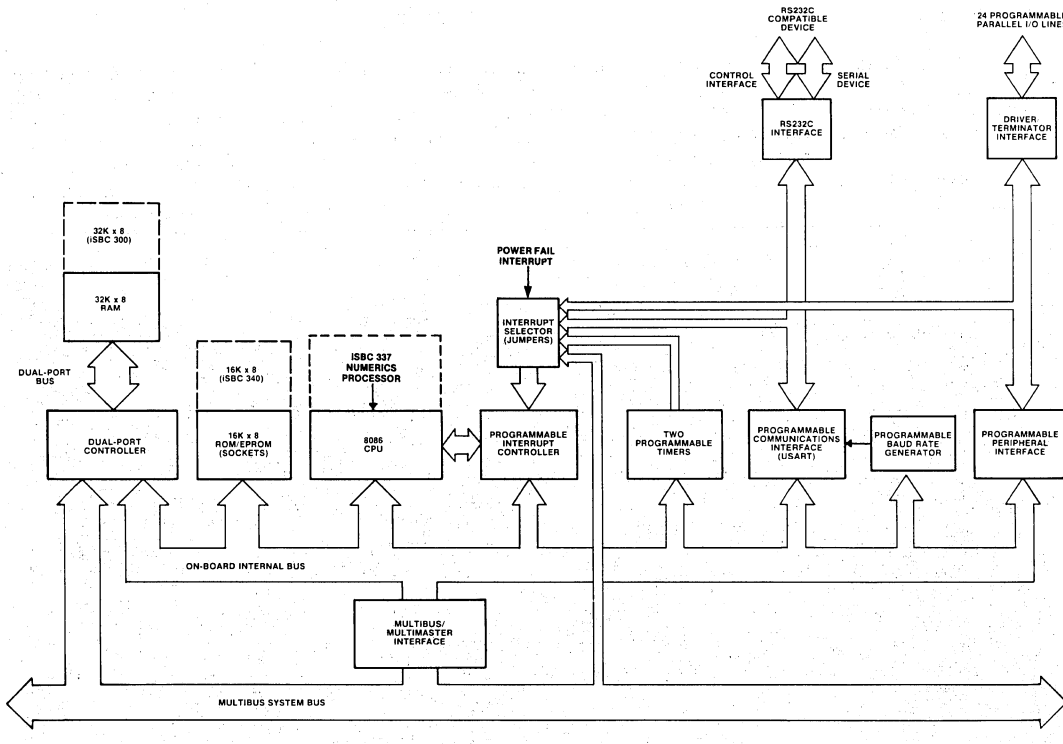


Figure 1. iSBC 86/12A Single Board Computer Block Diagram

### 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<sub>H</sub>. 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 non-volatile 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 Intel 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 MULTIMODULE 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 MULTIBUS 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 <sup>1</sup>
	4	X		X			X <sup>1</sup>

**Note**

1. Port of part 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 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 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 Capability

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 00008H. 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

**\*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 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.

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.

### 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 Expansion** — 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-FFFFFH (using 2758 EPROMs); FE000-FFFFFH (using 2716 EPROMs); FC000-FFFFFH (using 2732 EPROMs); F8000-FFFFFH (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: 00000-07FFFH; 64K bytes: 00000-0FFFFH.

**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	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  $\pm$  0.1% (0.041  $\mu$ s period, nominal); 1.23 MHz  $\pm$  0.1% (0.81  $\mu$ s period, nominal); or 153.60 kHz  $\pm$  0.1% (6.51  $\mu$ s 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 $\mu$ s	427.1 ms	3.26 s	466.50 min
Programmable one-shot	1.63 $\mu$ s	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 $\mu$ s	427.1 ms	3.26 s	466.50 min
Hardware triggered strobe	1.63 $\mu$ s	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  $\pm$  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

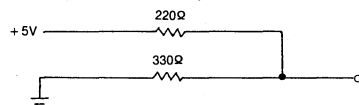
**Note:**

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

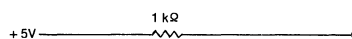
Port 1 of the 8255A has 20 mA totem-pole bidirectional drivers and 1 k $\Omega$  terminators.

**I/O Terminators** — 220 $\Omega$ /330 $\Omega$  divider or 1 k $\Omega$  pullup

**220 $\Omega$ /330 $\Omega$  (ISBC 901 OPTION)**



**1 K  $\Omega$  (ISBC 902 OPTION)**



## iSBC 86/12A

### 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 <sub>CC</sub> = +5V ± 5% (max)	V <sub>DD</sub> = +12V ± 5% (max)	V <sub>BB</sub> = -5V ± 5% (max)	V <sub>AA</sub> = -12V ± 5% (max)
Without EPROM <sup>1</sup>	5.2A	350 mA	—	40 mA
RAM Only <sup>3</sup>	390 mA	40 mA	1.0 mA	—
With iSBC 530 <sup>4</sup>	5.2A	450 mA	—	140 mA
With 4K EPROM <sup>5</sup> (using 2758)	5.5A	350 mA	—	40 mA
With 8K EPROM <sup>5</sup> (using 2716)	5.5A	350 mA	—	40 mA
With 16K EPROM <sup>5</sup> (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

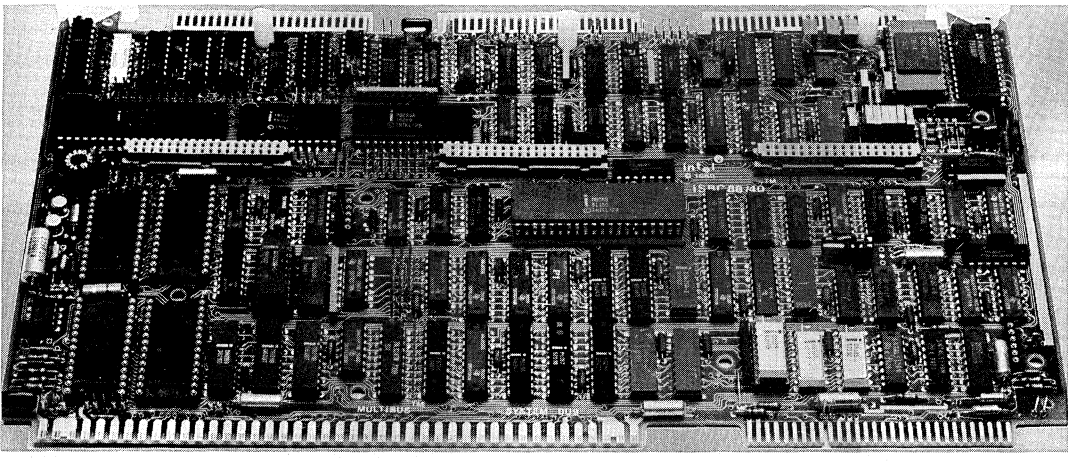
Part Number	Description
SBC 86/12A	Single Board Computer with 32K bytes RAM



## iSBC 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/E<sup>2</sup>PROM sockets for up to 32K bytes, expandable to 64K bytes with iSBC 341 expansion MULTIMODULE
- On-board 21-volt power supply for E<sup>2</sup>PROM 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.



# iSBC 88/40

## 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 computer 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 5 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.

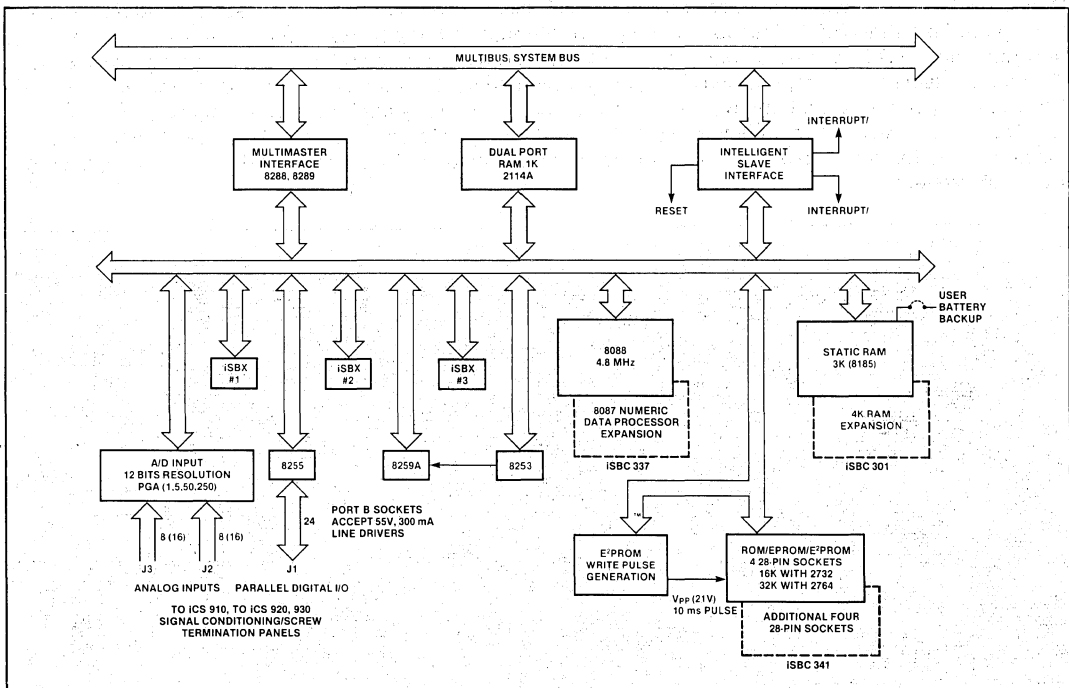


Figure 1. iSBC 88/40 Measurement and Control Computer Block Diagram

**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 1.2  $\mu$ sec 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 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 address 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 system 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.

## 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 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. 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, 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.

**E<sup>2</sup>PROM Capabilities**

The four 28-pin sockets can also accommodate Intel 2816 E<sup>2</sup>PROMs, 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 E<sup>2</sup>PROMs.

**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<sup>2</sup>PROM write operations.

**Interrupt Capability**

The iSBC 88/40 board provides 9 vectored interrupt levels. The highest level is the NMI (Non-maskable 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 00008H. The Intel 8259A Programmable Interrupt Controller (PIC) provides vectoring for the next eight interrupt levels. As shown in Table 2, a selection of four priority processing modes is available to the designer to match system requirements. Operating mode and priority assignments may be re-configured dynamically via software at any time

**Table 1. Input/Output Port Modes of Operation**

Port	Lines (qty)	Mode of Operation				Bidirectional	Control
		Unidirectional					
		Input		Output			
		Latched	Latched & Strobed	Latched	Latched & Strobed		
1	8	X	X	X	X		
2	8	X	X	X	X		
3	4	X		X		X <sup>1</sup>	
	4	X		X		X <sup>1</sup>	

**NOTE:**

- 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.

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 achieved 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 noisier 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 volts) it provides 11 bits plus sign.

**SPEED** — The A-to-D converter conversion speed is 50  $\mu$ s (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 gains of 1

and 5, 250  $\mu$ s at a gain of 50, and 2.5 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  $\pm 1/2$  LSB. Offset is adjustable under program control to insure  $\pm 0.024\%$  FSR  $+ 1/2$  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 $\times$  (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.

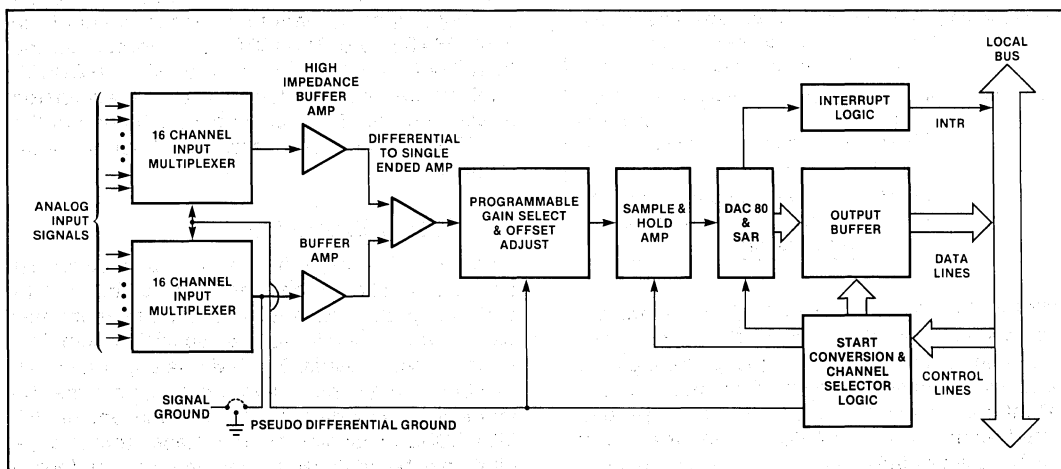
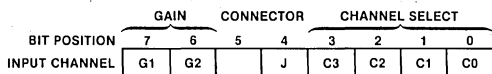
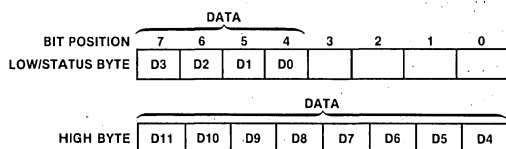


Figure 2. iSBC 88/40 Analog Input Section

**Output Command** — Select input channel and start conversion.



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



**Offset Correction** — At higher gains ( $\times 50$ ,  $\times 250$ ) 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 deter-

mine 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 Inteltec 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/E<sup>2</sup>PROM

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 E<sup>2</sup>PROM 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-FFFF (using 2716 EPROMs)
- FC000-FFFF (using 2732 EPROMs)
- F8000-FFFF (using 2764 EPROMs)

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

- FC000-FFFF (using 2716 EPROMs)
- F8000-FFFF (using 2732 EPROMs)
- F0000-FFFF (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; 550 nanoseconds

### Interval Timer

#### Output Frequencies —

Function	Single Timer		Dual Timers (Two Timers Cascaded)
	Min.	Max.	
Real-Time Interrupt Interval	0.977 $\mu$ s	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  $\pm$  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 volts).

**Accuracy** — Including noise and dynamic errors

Gain	25°C
1	±0.05% FSR*
5	±0.075% FSR*
50	±0.075% FSR*
250	±0.1% 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)** — 54 PPM per degree centigrade, 74 PPM at other gains.

Offset TC (in % of FSR/°C) — Gain	Offset TC
1	0.0016%
5	0.0022%
50	0.016%
250	0.08%

**Input Overvoltage Protection** — 30 volts

**Input Impedance** — 20 megohms (minimum)

**Conversion Speed** — 50 μs (maximum) at gain = 1 or 5

**Common Mode Rejection Ratio** — 60 dB (minimum)

**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)** —

Config-uration	+5V		+5V Aux		+12V		-12V	
	Typ	Max	Typ	Max	Typ	Max	Typ	Max
iSBC 88/40 <sup>1,2</sup>	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 60°C (32°C to 131°F)

**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 80/40 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**

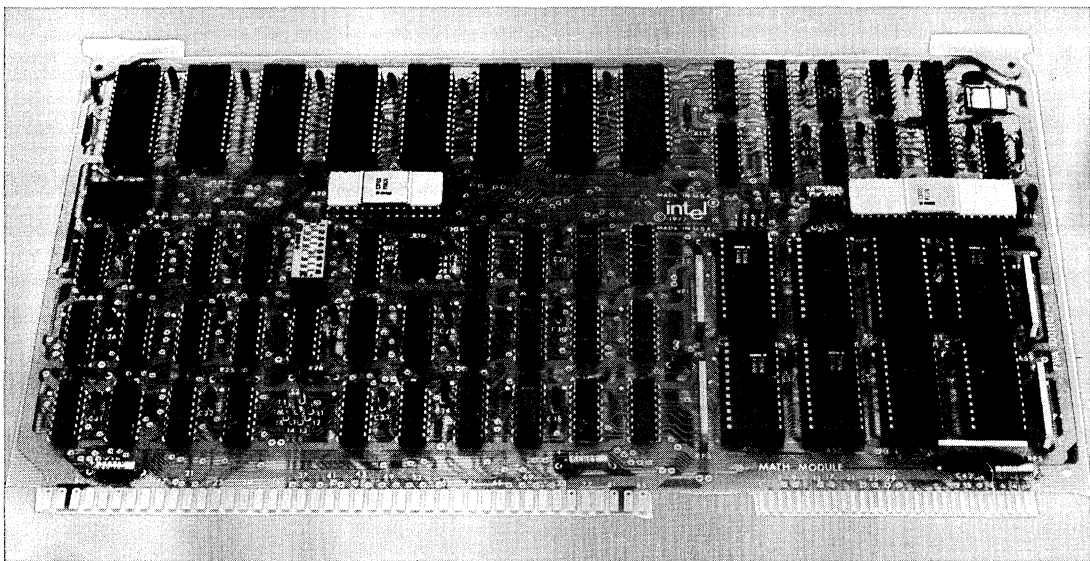
Part Number	Description
SBC 88/40	Measurement and Control Computer



## iSBC 310 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 310 via system bus
- Single +5V power requirement

The iSBC 310 High Speed Mathematics Unit is a member of Intel's complete family of OEM computers and expansion modules. The iSBC 310 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 310 utilizes Intel's high speed Series 3000 Bipolar Microprocessor. The iSBC 310 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 310 using I/O and memory read/write commands. To pass arguments from the iSBC 80 to the iSBC 310, a memory write command is used for each byte to be loaded into the iSBC 310's working registers. An operation command is then given to the iSBC 310 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 310 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 310 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 310 may be operated in either an interrupt driven or polled mode. Three status indications are available:

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

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

**Error** — The iSBC 310 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.

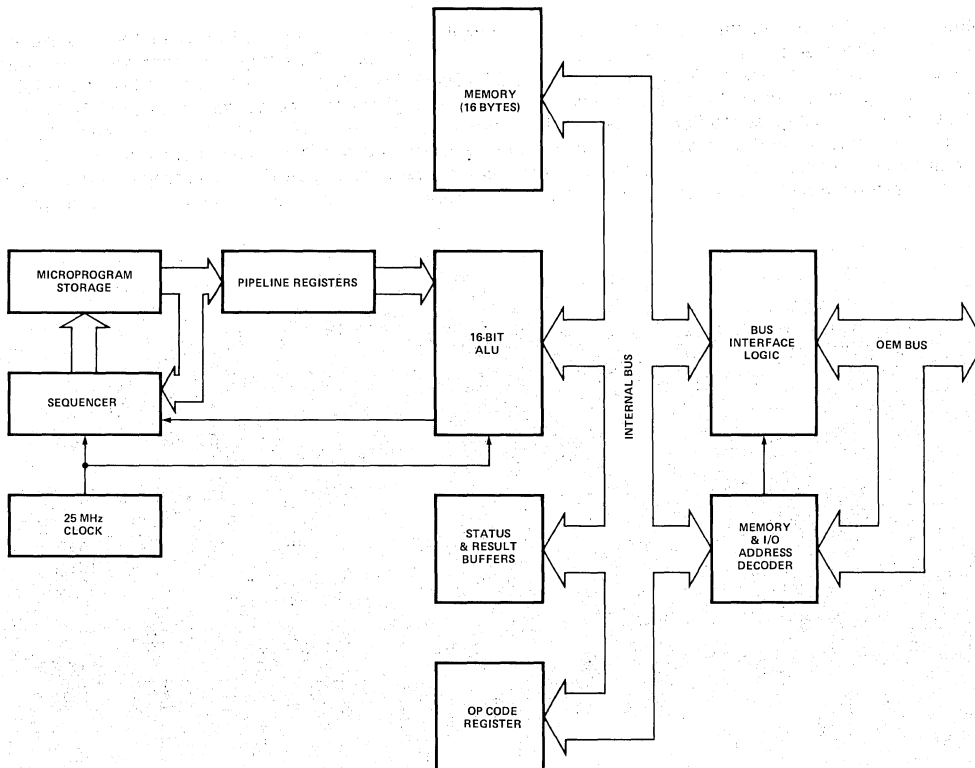


Figure 1. iSBC 310 Block Diagram Showing Functional Components

Table 1. iSBC 310 Arithmetic Functions

Operation	Op Code	Max Time <sup>1</sup> ( $\mu$ s)	Notes <sup>2</sup>
Fixed point multiply (MUL)	0	19	$M_1 = m_1 \cdot m_1$
Fixed point divide (DIV)	1	28	$m_1 = m_1/M_2$ , $m_2 = \text{remainder}$
Extended fixed point divide (EDIV)	E	94	$M_1 = M_1/m_2$ , $M_2 = \text{remainder}$
Floating point multiply (FMUL)	2	91	$X_1 = X_1 \cdot X_2$
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**

- Does not include register setup time
- $m$  — 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



# ISBC 310

## 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 310.

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 310 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 <sub>CC</sub>	I <sub>CC</sub>
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 310 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 310	High Speed Mathematics Unit

---

*iSBX™ Multimodule™  
Boards*

---

**2**

5

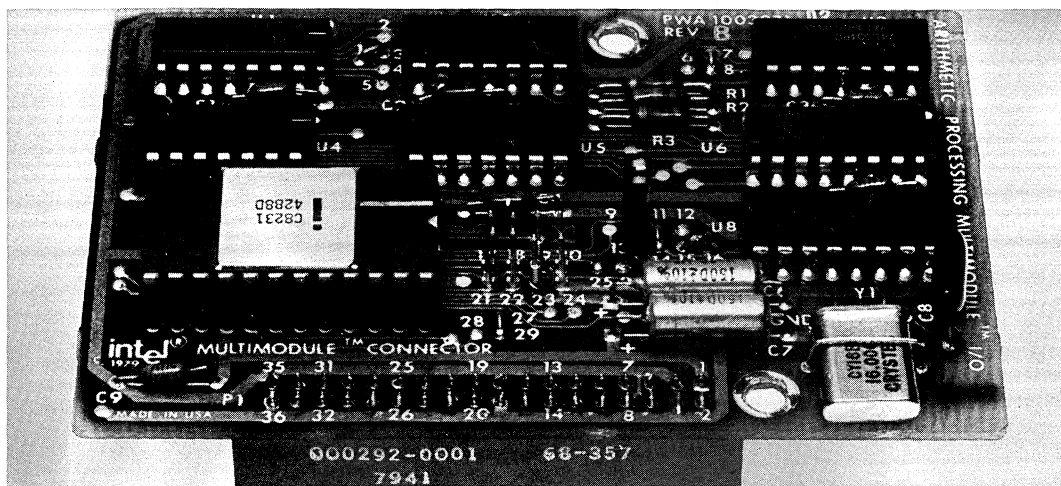
Handwritten text, possibly a signature or name, appearing as a series of loops and curves.



## 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
- Trigonometric 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 40x 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.



# ISBX 331

## 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).

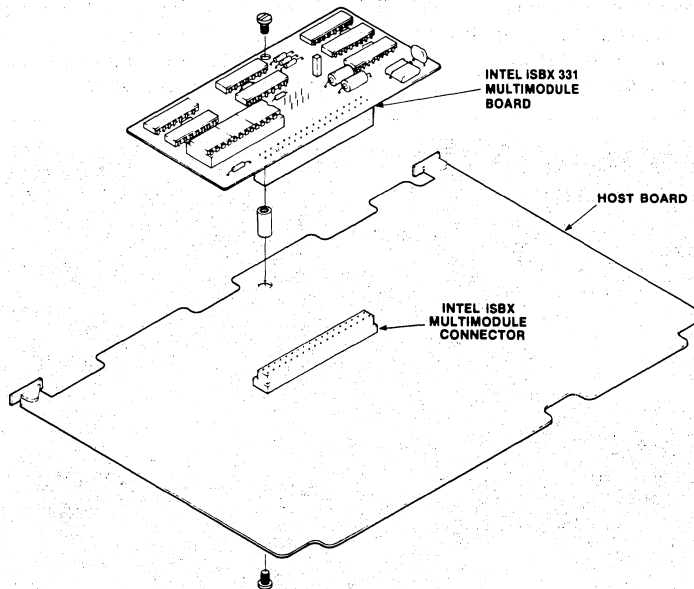


Figure 1. Installation of ISBX 331 Module on a Host Board

# iSBX 331

Table 1. Command Summary

## Double Precision Floating Point Instructions (32-Bit)

Instruction	Description	Hex Code	Stack Contents After Execution <sup>(1)</sup>				Status Flags Affected <sup>(3)</sup>
			A	B	C	D	
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 <sup>A</sup> 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 $\pi$ onto Stack	1 A	R	A	B	C	S, Z
PWR	B <sup>A</sup> 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 <sup>(1)</sup>				Status Flags Affected <sup>(3)</sup>
			A	B	C	D	
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

# ISBX 331

**Table 1. Command Summary (continued)**

**Single Precision Fixed Point Instructions (16-Bit)**

Instruction	Description	Hex Code	Stack Contents After Execution <sup>(2)</sup>								Status Flags Affected <sup>(3)</sup>
			A <sub>U</sub>	A <sub>L</sub>	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	
CHSS	Change Sign of A <sub>U</sub>	7 4	R	A <sub>L</sub>	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	S, Z, O
FIXS	Floating to Fixed Point Conversion	1 F	R	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	U	U	U	S, Z, O
POPS	Stack Pop	7 8	A <sub>L</sub>	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	A <sub>U</sub>	S, Z
PTOS	Stack Push	7 7	A <sub>U</sub>	A <sub>U</sub>	A <sub>L</sub>	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	S, Z
SADD	Add A <sub>U</sub> and A <sub>L</sub>	6 C	R	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	A <sub>U</sub>	S, Z, C, E
SDIV	Divide A <sub>L</sub> by A <sub>U</sub>	6 F	R	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	U	S, Z, E
SMUL	Multiply A <sub>L</sub> by A <sub>U</sub> (R = lower 16 bits)	6 E	R	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	U	S, Z, E
SMUU	Multiply A <sub>L</sub> by A <sub>U</sub> (R = upper 16 bits)	7 6	R	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	U	S, Z, E
SSUB	Subtract A <sub>U</sub> from A <sub>L</sub>	6 D	R	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	A <sub>U</sub>	S, Z, C, E
XCHS	Exchange A <sub>U</sub> and A <sub>L</sub>	7 9	A <sub>L</sub>	A <sub>U</sub>	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	S, Z
NOP	No Operation	0 0	A <sub>U</sub>	A <sub>L</sub>	B <sub>U</sub>	B <sub>L</sub>	C <sub>U</sub>	C <sub>L</sub>	D <sub>U</sub>	D <sub>L</sub>	

**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<sub>U</sub>, A<sub>L</sub>, B<sub>U</sub>, B<sub>L</sub>, C<sub>U</sub>, C<sub>L</sub>, D<sub>U</sub>, D<sub>L</sub>). A<sub>U</sub> is the TOS and A<sub>L</sub> is NOS. Upon completion of a command the stack is composed of: the result (R); undefined (U); or the initial contents (A<sub>U</sub>, A<sub>L</sub>, B<sub>U</sub>, B<sub>L</sub>,...).
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.

# ISBX 331

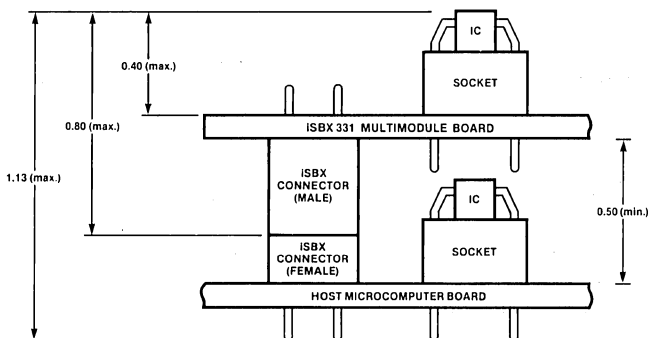


Figure 2. ISBX 331 MULTIMODULE Board Mounting Clearances (inches)

## SPECIFICATIONS

### Word Size

Data—8 bits.

### On-Board Clock Rate

4.0 MHz  $\pm$  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:

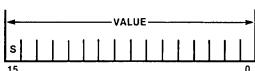
The port addresses are determined on the host iSBC microcomputer. Refer to the Hardware Reference Manual for your host iSBC microcomputer to determine the first digit (X) of the connector port addresses.

### Arithmetic Functions

See Table 1.

### Data Formats

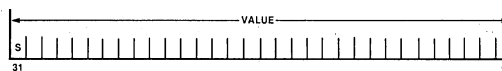
#### Single Precision Fixed Point (16 bits)



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.

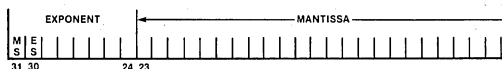
#### Double Precision Fixed Point (32 bits)



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:

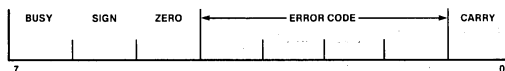
$$\text{Value} = \text{mantissa} \times 2^{\text{exponent}}$$



# iSBX 331

## 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

$V_{CC} = +5V \pm 5\%$      $I_{CC} = 365 \text{ mA max.}$

$V_{DD} = +12V \pm 5\%$      $I_{DD} = 75 \text{ mA max.}$

## 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

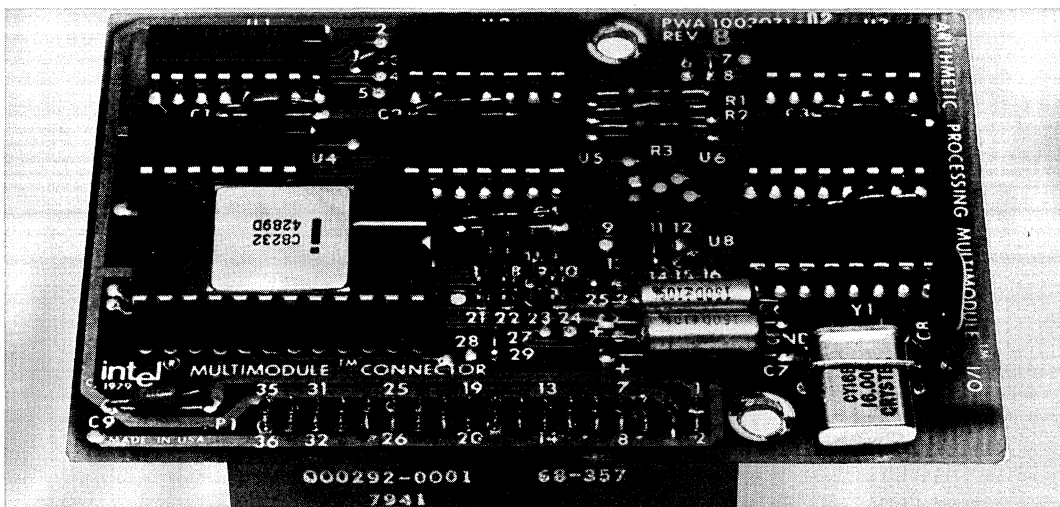
Part Number	Description
SBX 331	Fixed/Floating Point Math MULTIMODULE Board



## 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



## 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).

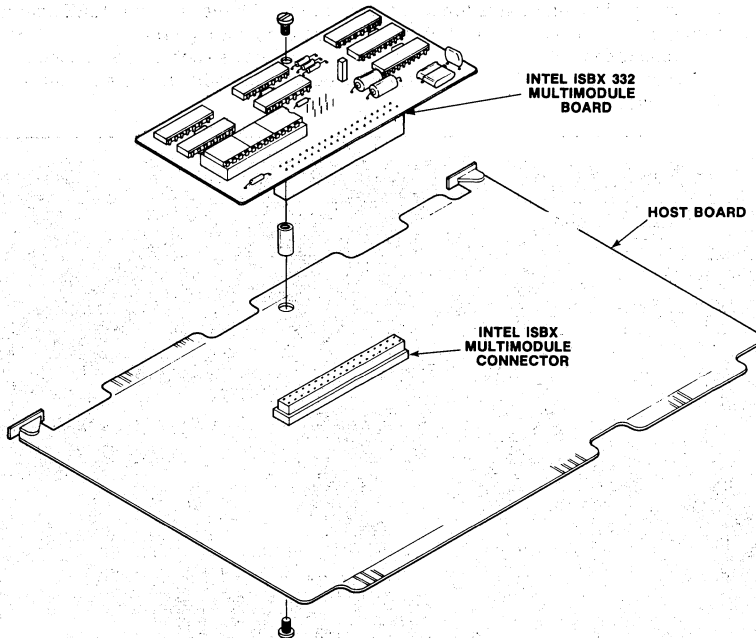


Figure 1. Installation of ISBX 332 Module on a Host Board

Table 1. Command Summary

Command Bits 7 6 5 4 3 2 1 0	Mnemonic	Description
X 0 0 0 0 0 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.

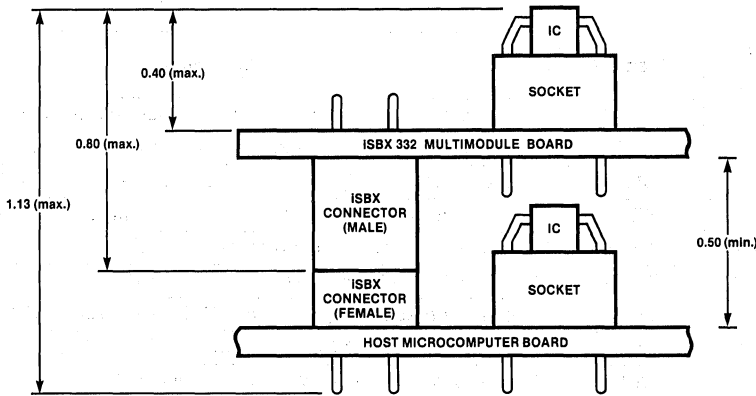


Figure 2. Mounting Clearances (Inches)

**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:**

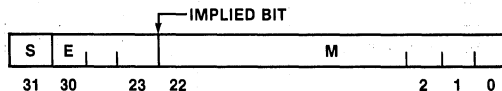
The port addresses are determined on the host ISBC microcomputer. Refer to the Hardware Reference Manual for your host ISBC microcomputer to determine the first digit (X) of the connector port address.

**Arithmetic Functions**

See Table 1

**Floating Point Format**

**Single Precision Floating Point (32 Bits)**



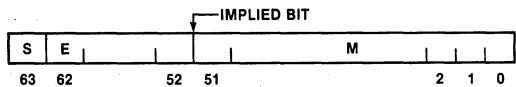
Bit 31: S = Sign of the mantissa. 1 represents negative and 0 represents positive.

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 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**

$V_{CC} = +5V \pm 5\%$                        $I_{CC} = 365 \text{ mA max.}$   
 $V_{DD} = +12V \pm 5\%$                       $I_{DD} = 75 \text{ mA max.}$

**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**

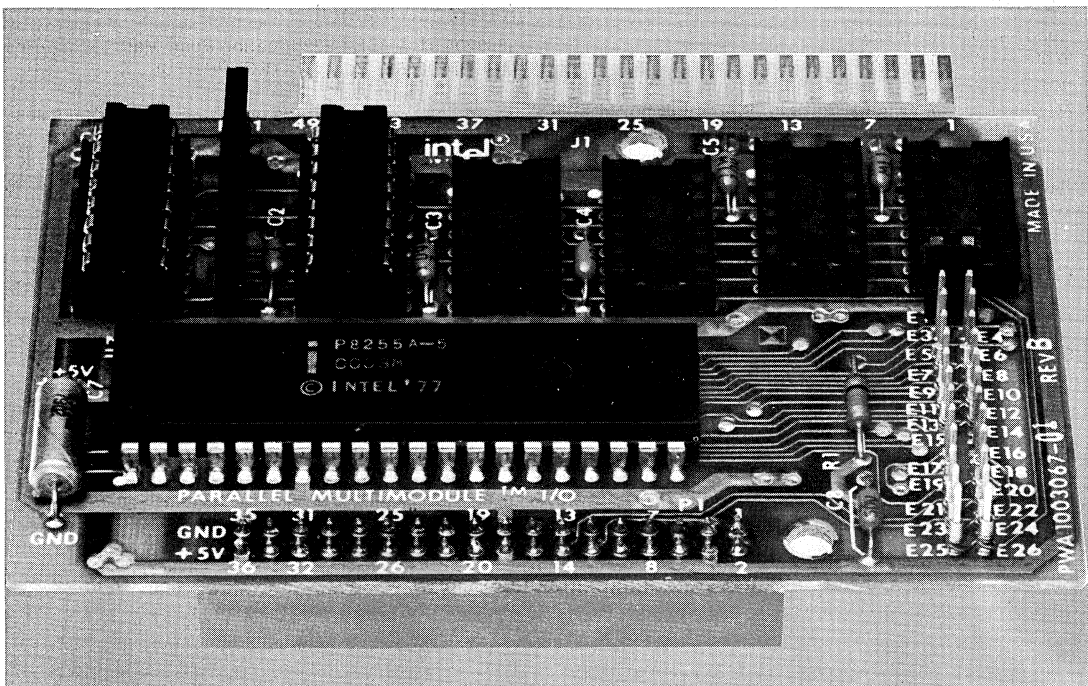
Part Number	Description
SBX 332	Floating Point Math MULTIMODULE Board



## 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).

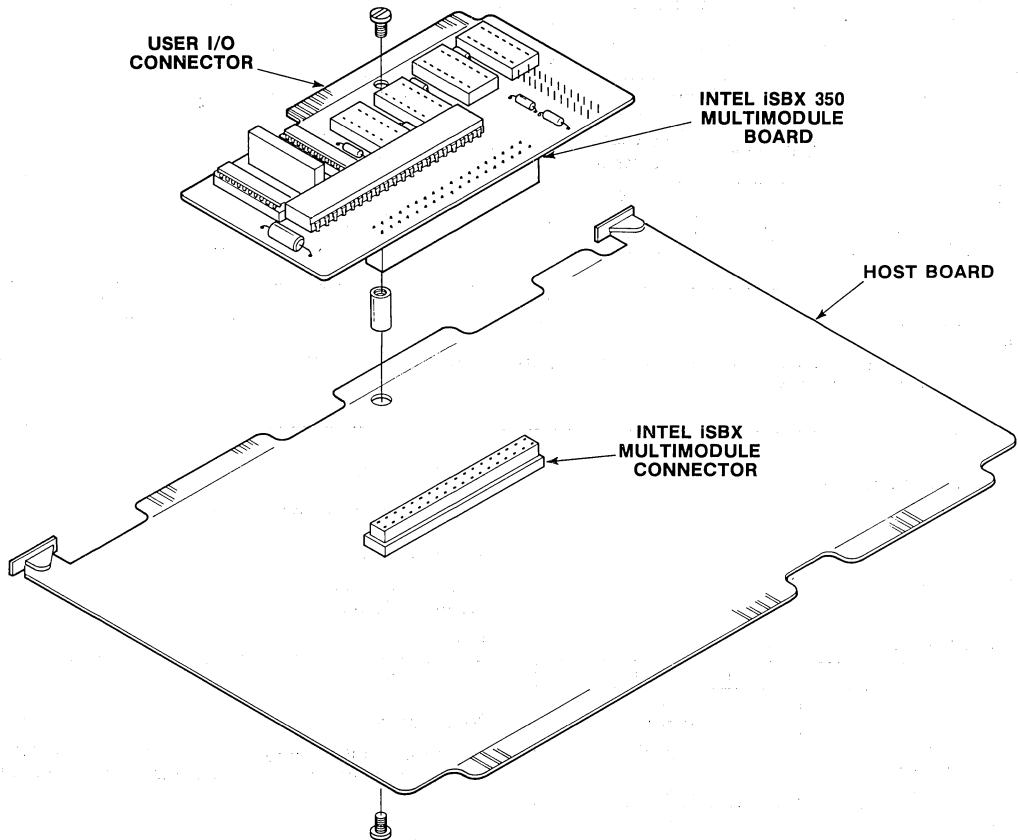


Figure 1. Installation of iSBX 350 Module on a Host Board



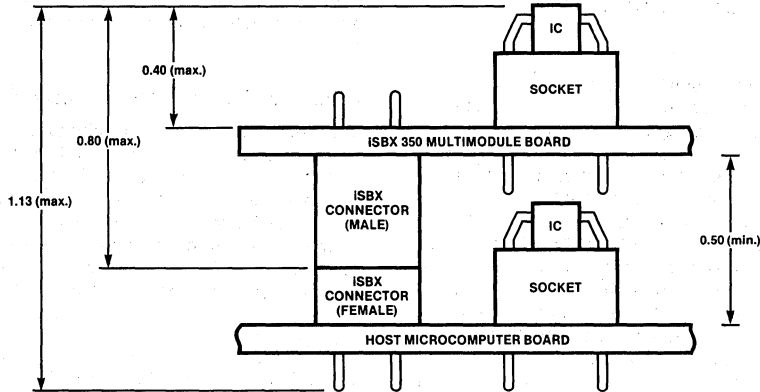


Figure 2. Mounting Clearances (inches)

Table 1. Input/Output Port Modes of Operation

Port	Lines (qty)	Mode of Operation				Bidirectional	Control
		Unidirectional					
		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 <sup>1</sup>
	4	X		X			X <sup>1</sup>

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

## iSBX 350

### 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.

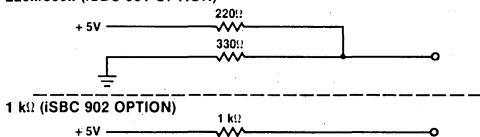
Driver	Characteristic	Sink Current (mA)
7438	I, OC	48
7437	I	48
7432	NI	16
7426	I, OC	16
7409	NI, OC	16
7408	NI	16
7403	I, OC	16
7400	I	16

**Note:**

I = Inverting, NI = Non-Inverting, OC = Open Collector

Port 1 has 25 mA totem pole drivers and 1 k $\Omega$  terminators.

**I/O Terminators** — 220 $\Omega$ /330 $\Omega$  divider or 1 k $\Omega$  pull up.  
220 $\Omega$ /330 $\Omega$  (ISBC 901 OPTION)



### 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

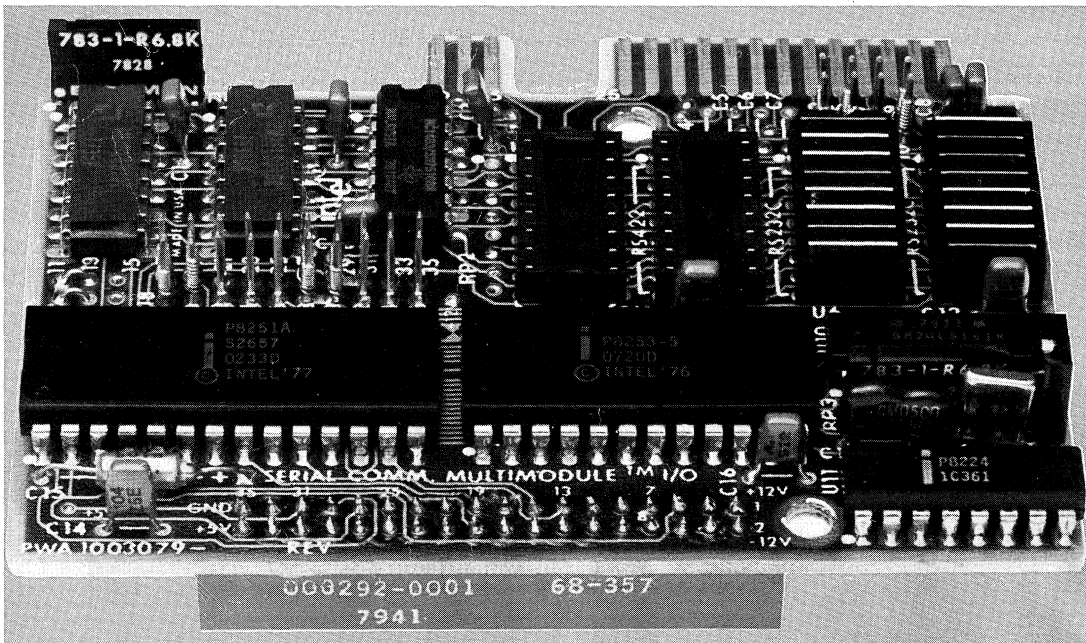
Part Number	Description
SBX 350	Parallel I/O MULTIMODULE Board



## 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

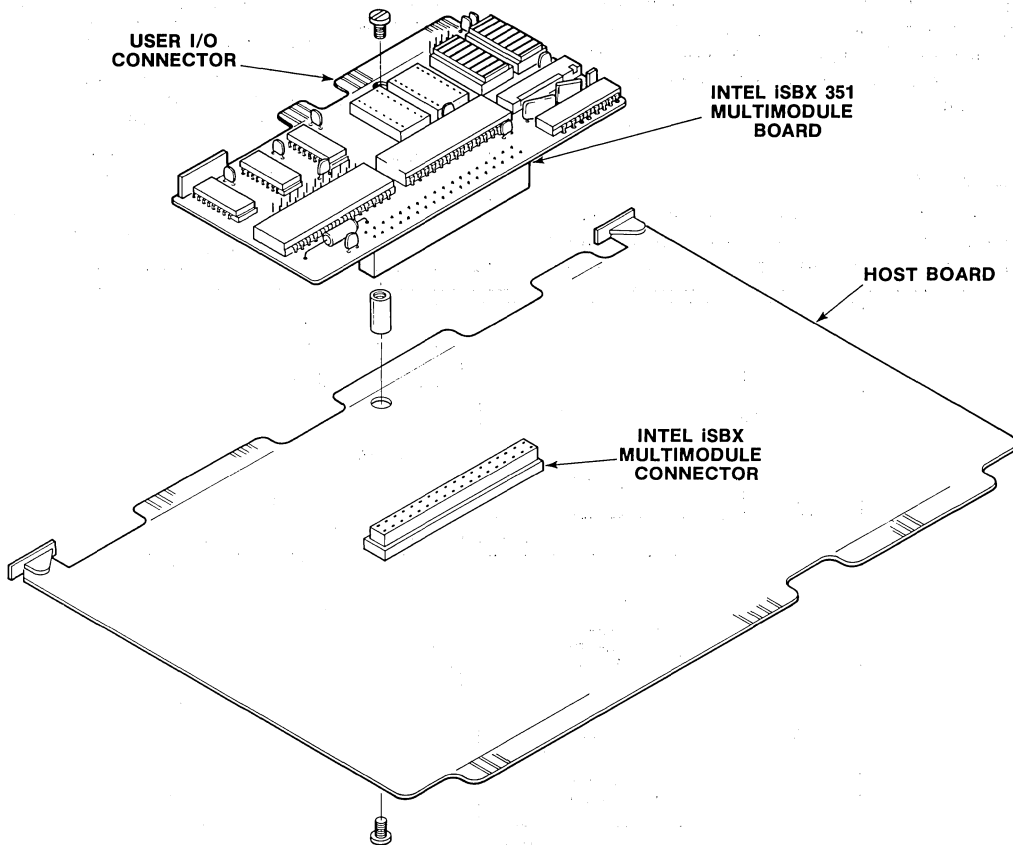


Figure 1. Installation of iSBX 351 Module on a Host Board

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.

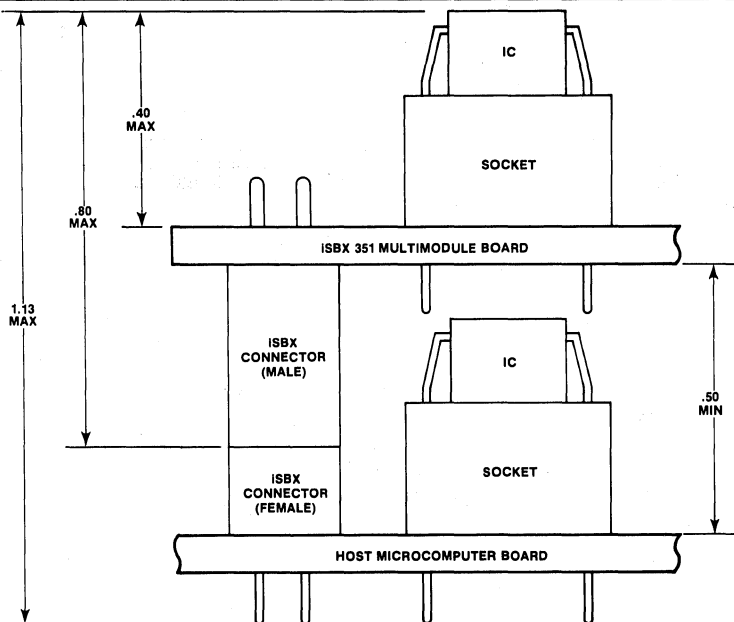
### Installation

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)**

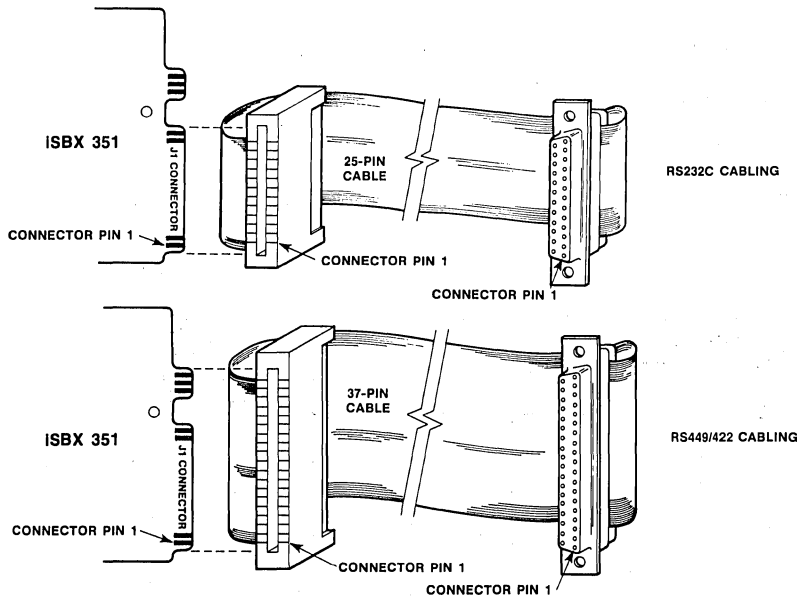


Figure 3. Cable Construction and Installation for RS232C and RS449/422 Interface

**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 <sup>1</sup> (kHz, Software Selectable)	8251 USART Baud Rate (Hz) <sup>2</sup>	
	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  $\pm 0.1\%$  (.813  $\mu$ sec period nominal)

153.6 kHz  $\pm 0.1\%$  (6.5  $\mu$ sec period nominal)

### Output Frequency:

	Rate Generator (Frequency)		Real-Time Interrupt (Interval)	
	Min.	Max.	Min.	Max.
Single Timer <sup>1</sup>	18.75 Hz	614.4 kHz	1.63 $\mu$ sec	53.3 msec
Single Timer <sup>2</sup>	2.34 Hz	76.8 kHz	13.0 $\mu$ sec	426.7 msec
Dual Timer <sup>3</sup> (Counters 0 and 1 in series)	0.000286 Hz	307.2 kHz	3.26 $\mu$ sec	58.25 min
Dual Timer <sup>4</sup> (Counters 0 and 1 in series)	0.0000358 Hz	38.4 kHz	26.0 $\mu$ sec	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 <sup>2</sup>	MULTIMODULE Edge Connector	Cable	Connector <sup>8</sup>
RS232C	DTE	26-pin <sup>5</sup> , 3M-3462-0001	3M <sup>3</sup> -3349/25	25-pin <sup>7</sup> , 3M-3482-1000
RS232C	DCE	26-pin <sup>5</sup> , 3M-3462-0001	3M <sup>3</sup> -3349/25	25-pin <sup>7</sup> , 3M-3483-1000
RS449	DTE	40-pin <sup>6</sup> , 3M-3464-0001	3M <sup>4</sup> -3349/37	37-pin <sup>1</sup> , 3M-3502-1000
RS449	DCE	40-pin <sup>6</sup> , 3M-3464-0001	3M <sup>4</sup> -3349/37	37-pin <sup>1</sup> , 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.

## iSBX 351

---

### Electrical Characteristics

#### DC Power Requirements

Mode	Voltage	Amps (Max.)
RS232C	+5V $\pm 0.25V$	460 mA
	+12V $\pm 0.6V$	30 mA
	-12V $\pm 0.6V$	30 mA
RS449/422	+5V $\pm 0.25V$	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



1. The first part of the document discusses the importance of maintaining accurate records of all transactions and activities. It emphasizes the need for transparency and accountability in financial reporting.

2. The second part of the document outlines the various methods and techniques used to collect and analyze data. It includes a detailed description of the experimental procedures and the statistical tools employed to interpret the results.

3. The third part of the document presents the findings of the study, including a comparison of the results with previous research. It highlights the key observations and discusses the implications of the findings for future research and practice.

4. The fourth part of the document provides a comprehensive conclusion, summarizing the main points of the study and offering recommendations for further investigation. It also includes a list of references and a bibliography of the sources consulted during the research process.

5. The fifth part of the document contains a detailed appendix, which includes all the raw data, calculations, and supporting materials used in the study. This section is intended to provide a complete and transparent record of the research process for the benefit of other researchers and stakeholders.

6. The sixth part of the document is a glossary of terms, defining the key concepts and terminology used throughout the document. This section is designed to ensure clarity and consistency in the use of language and to facilitate the understanding of the document's content.

7. The seventh part of the document is a list of references, providing a comprehensive overview of the literature related to the study. This section is intended to acknowledge the contributions of other researchers and to provide a starting point for further exploration of the topic.



*iCS™ Industrial  
Control Series*

---

**3**



THE UNIVERSITY OF CHICAGO  
LIBRARY

UNIVERSITY OF CHICAGO LIBRARY

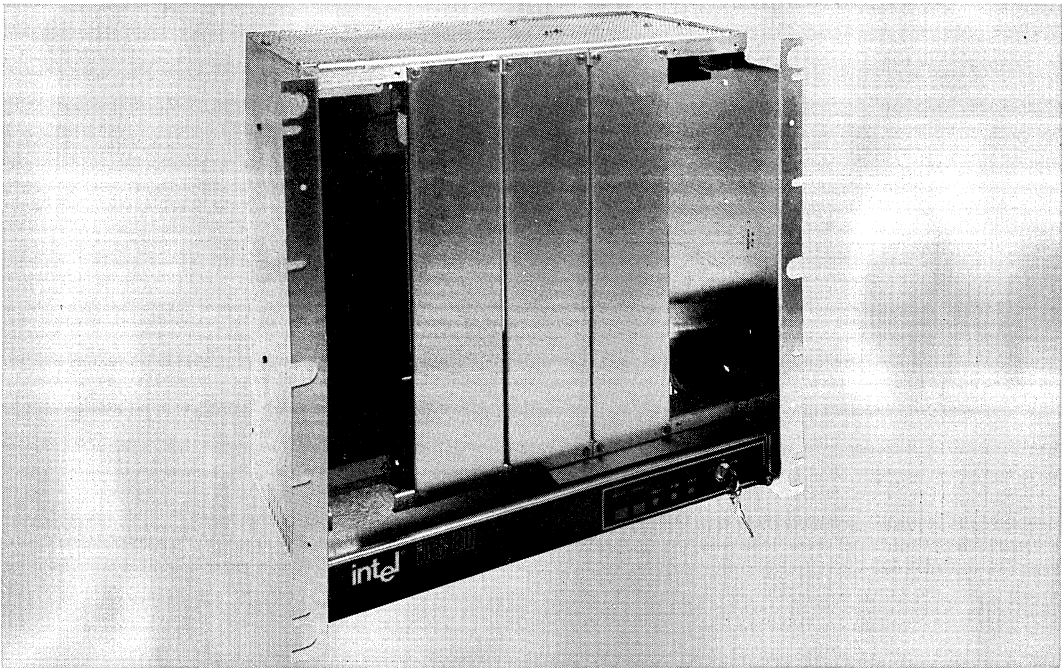
[The remainder of the page contains extremely faint and illegible text, likely bleed-through from the reverse side of the document.]



## 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 86/12 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).

### 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.

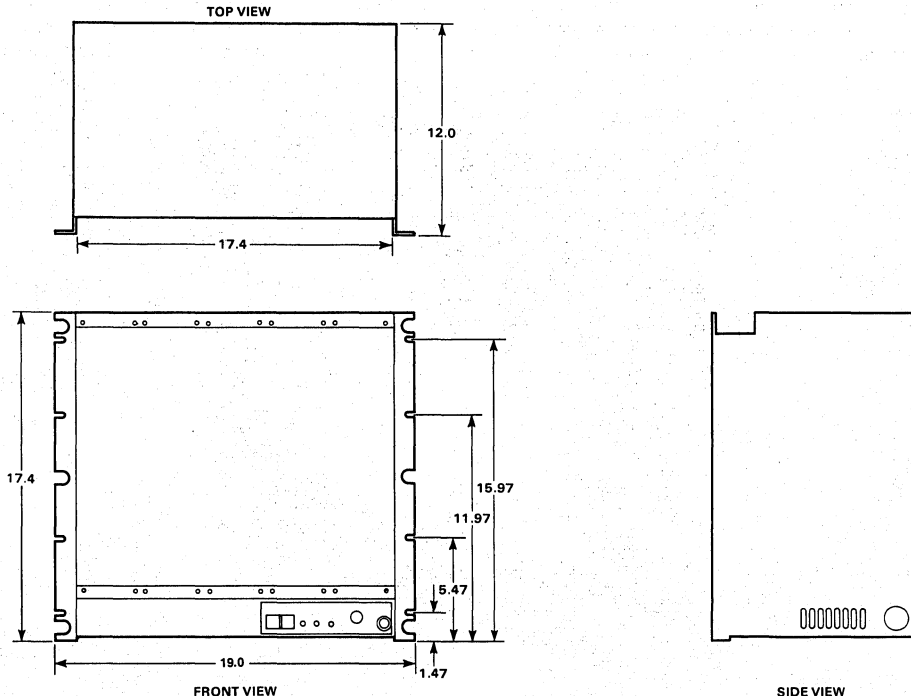
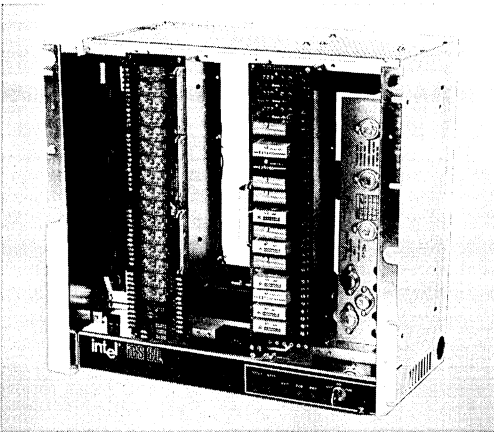


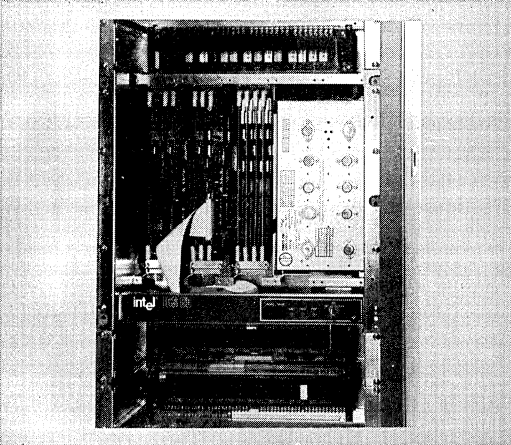
Figure 1. ICS 80 Chassis Dimensions



**Typical Small Configuration**

- 8-bit 8080 processor (iSBC 80/20-4)
- 2K bytes RAM
- 8K bytes ROM/EPROM
- 16 analog inputs
- 2 analog outputs
- 8 isolated digital inputs
- 8 isolated digital outputs
- 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 86/12)
- 96K bytes RAM
- 16K bytes EPROM
- 64 analog inputs
- 4 analog voltage outputs
- 4 analog current outputs (4-20 mA)
- 132 isolated digital inputs
- 48 TTL digital inputs
- 52 isolated digital outputs

(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

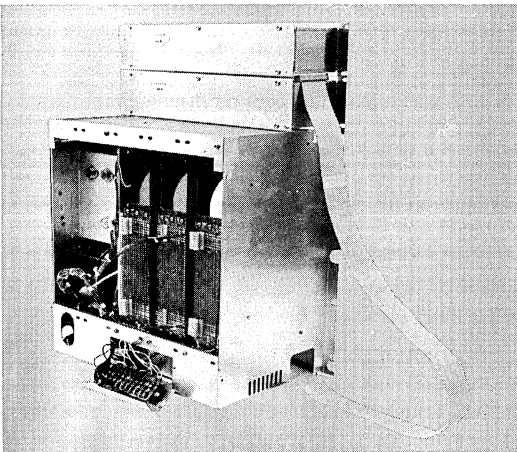


Figure 4. Rear View ICS 80 Chassis Showing Power Distribution Panel, and Cabling from ICS 80 Chassis to ICS 9XX RETMA Mounted Signal Conditioning Panels (Top of ICS 80 Chassis)

### 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.

---

## 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)

### Submitted for UL Recognition

The ICS 80 chassis has been submitted to Underwriters Laboratories for approval 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 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.

### 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.

### 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.

Current: (including fans)	With iSBC 635	With iSBC 640	Input Voltage
	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** —  $\pm 1\%$  at  $\pm 10\%$  static line change and  $\pm 50\%$  static load change, measured at the output connector ( $\pm 0.2\%$  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  $\mu$ sec for  $\pm 50\%$  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

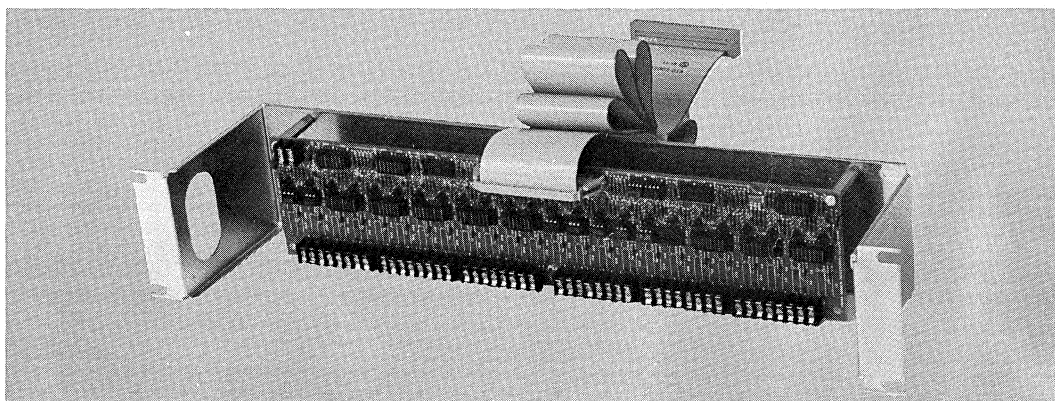




## iCS 910/920/930 SIGNAL CONDITIONING/TERMINATION PANELS

- 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 filters
  - 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**

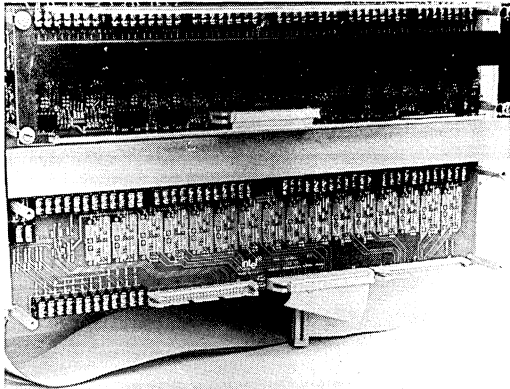
The barrier strip screw terminations on the iCS 910/920/930 panels provide familiar connection points for factory electricians to terminate the heavier gauge wiring often pulled through conduits from sensors or control elements. These screw terminals securely connect up to 14 AWG gauge wire size (16-gauge on iCS 910/920 panels). Alternately, spade lugs can be crimped on field wiring and inserted under the screw terminals.

**Mounting Flexibility and Serviceability**

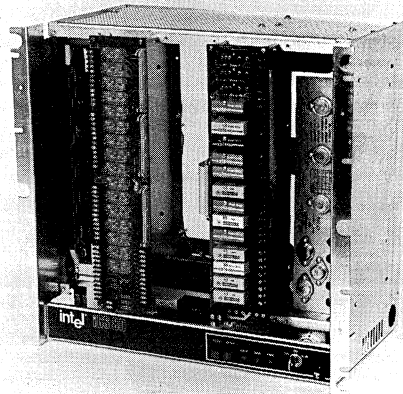
The iCS 910/920/930 panels were designed to be physically separate from iSBC boards or the iCS 80 chassis to allow maximum mounting flexibility and ease of serviceability. The panels and field wiring can be mounted in one area of the cabinet where electricians have access. Flat ribbon cable can then 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).

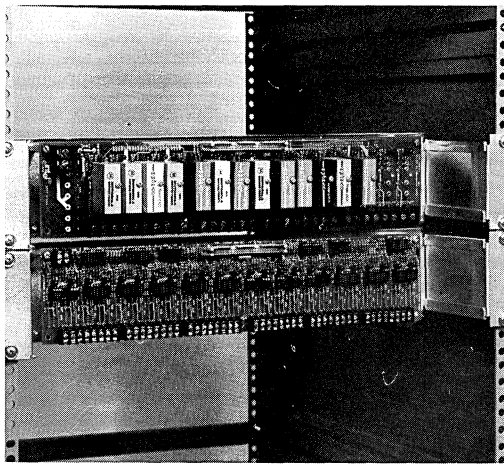
A black metal labelling strip is provided with each iCS 910/920/930 panel. White, blank gummed labels are included so that users can custom identify each input or output channel. A clear plastic cover is provided to protect against inadvertent touching or damage to the screw terminals or customer mounted components.



**Figure 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**



**Figure 1. iCS 930 AC and iCS 920 Digital Signal Conditioning/Termination Panel Mounted on a 19" Width RETMA Rack**

**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).

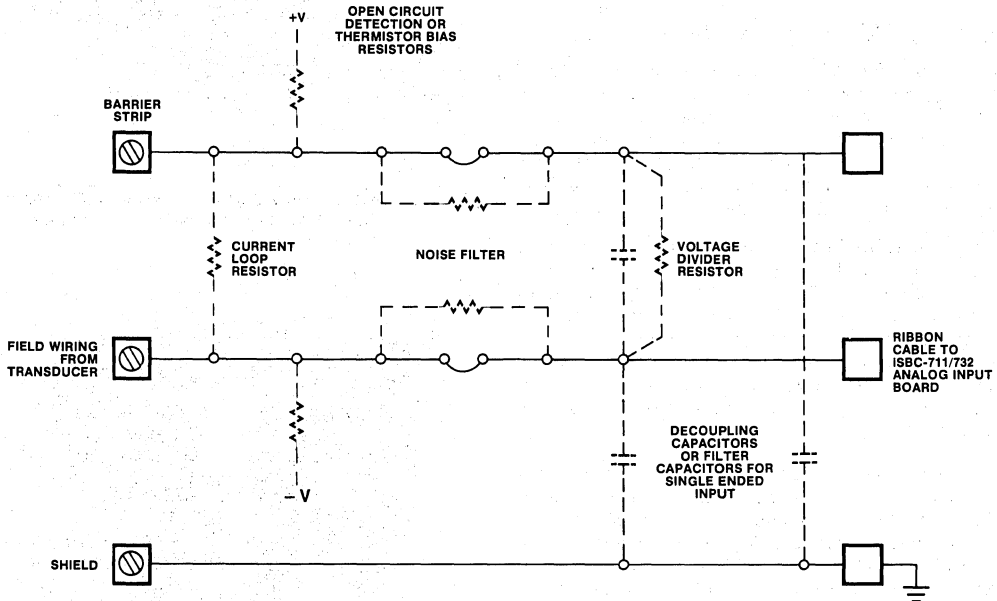


Figure 4. iCS 910 Analog Input Signal Conditioning Examples

## 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

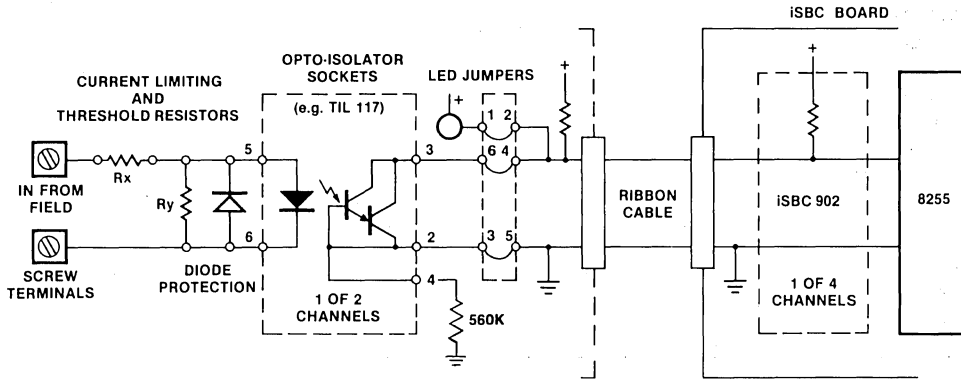
Dual-in-line sockets are in series with each channel (see Figure 5) to allow customer jumpering for straight through connections (TTL I/O), or for insertion of popular DIP packaged opto-isolators or digital output high current driver transistors. Circuit pads are available for

mounting voltage divider/threshold resistors and protection diodes.

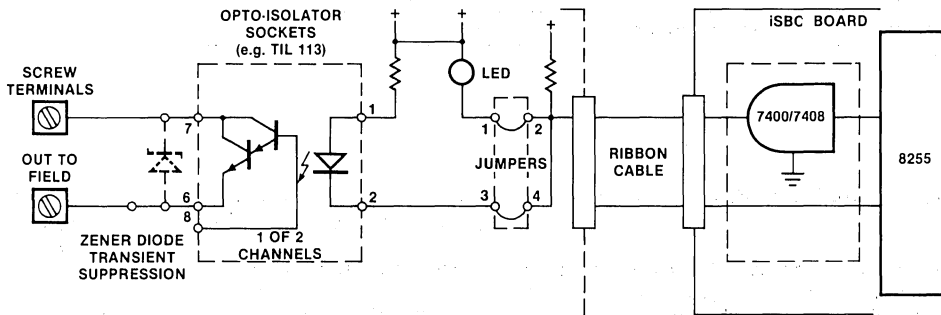
Groups of four inputs can have mixed voltage levels, opto-isolation, or straight through connections in groups of two. Output groups of four can be mixed opto-isolated or high current drive in groups of two. DIP components from a wide variety of vendors are selected and inserted by users based on their 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

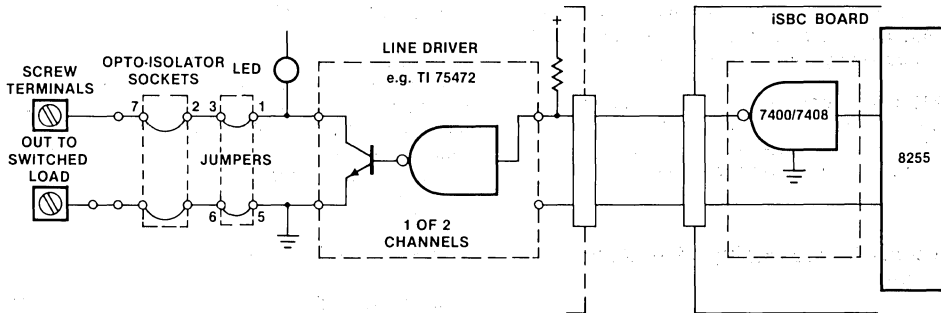
Light emitting diodes (LEDs) are mounted adjacent to each channel's screw terminals and may be jumpered in to indicate the Hi-Lo status of each of the 24 input or output channels.



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.

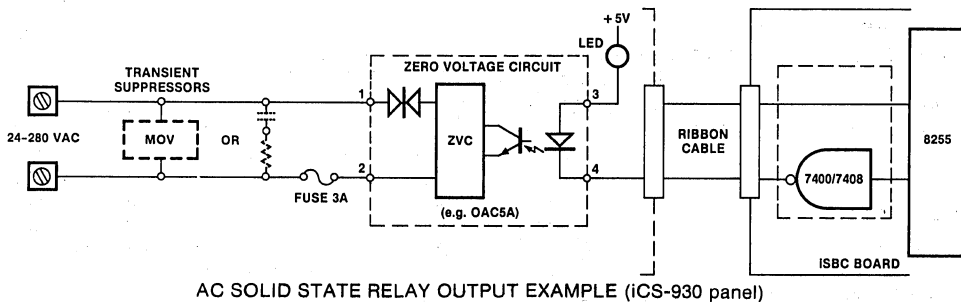
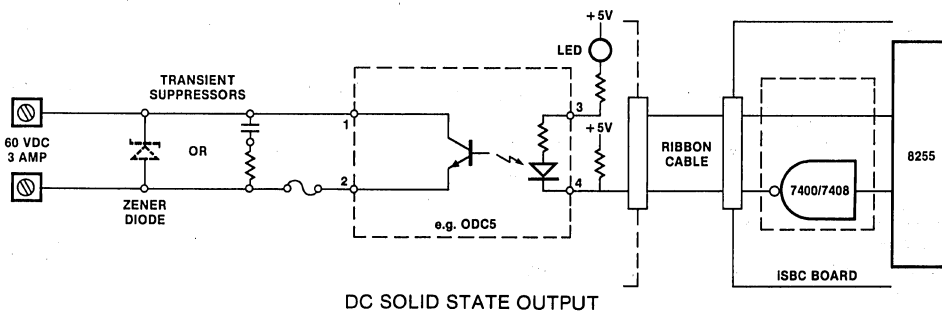
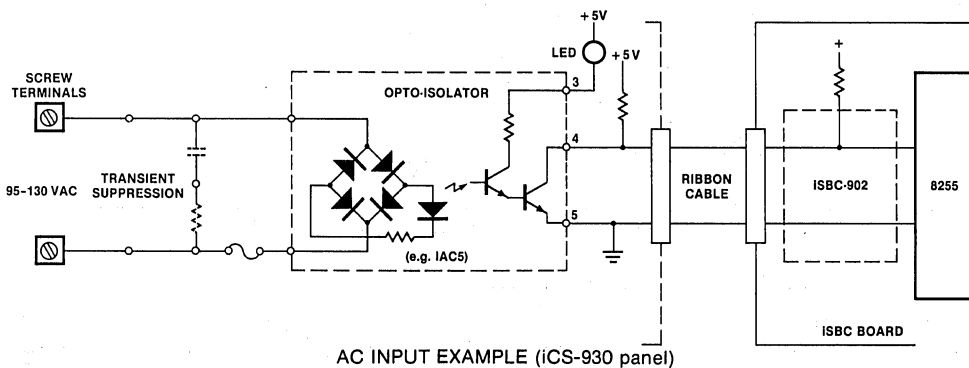
Examples of ICS 930 input and output schematics are shown in Figure 6.

# ICS 910/920/930

**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 220 VAC	95 to 130 VAC 180 to 280 VAC	10 mA 10 mA	IAC5 IAC5A	IAC5
DC Input — 5 $\mu$ sec Filter Fast, 50 $\mu$ sec On	10 to 32 VDC 4 to 16 VDC	32 mA 14 mA	IDC5 IDC5B	IDC5
<b>Output Current Rating</b>				
AC Output	12 to 140 VAC 24 to 280 VAC	3A 3A	OAC5 OAC5A	OAC5
DC Output	10 to 60 VDC 200 VDC	3A 1A	ODC5 ODC5A	ODC5

\* Motorola and Opto-22 sales offices are located in North America, Europe, and Japan.



**Figure 6. Typical ICS 930 Signal Conditioning Examples**

# ICS 910/920/930

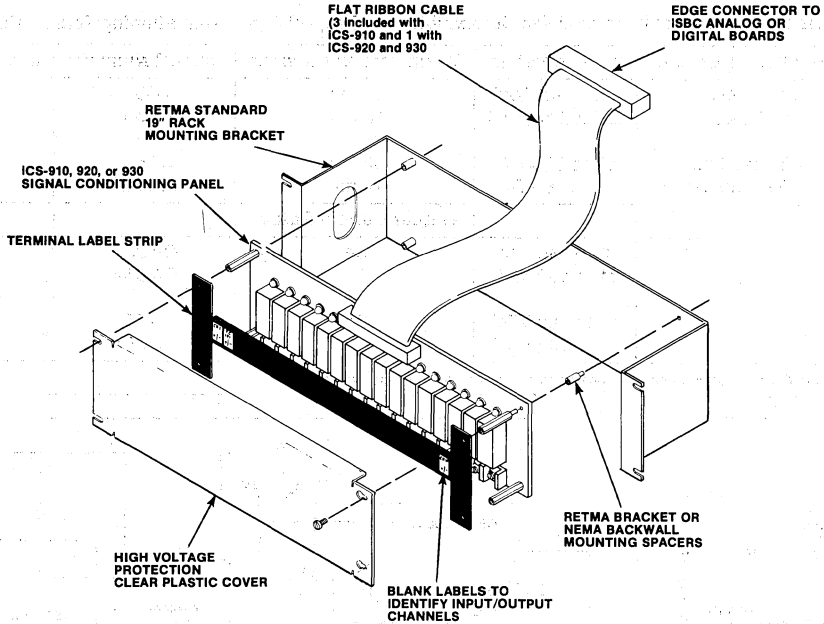


Figure 7. Mounting Arrangements for Signal Conditioning/Terminal Panels

## 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
<b>Weight:</b>	(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)
<b>Depth:</b>	(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.)
<b>Connectors:</b>	(Barrier strip)		
	2/56 screws	2/56 screws	6/32 screws
	48 AI	48 DI/DO	32 DI/DO
	12 AO	2 +5V power	2 +5V power
	2 power		
	(J1, J2, J3 to iSBC boards)		
	50-pin	50-pin	50-pin
	0.1 in. centers (2.54 mm)	0.1 in. centers (2.54 mm)	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: 12 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



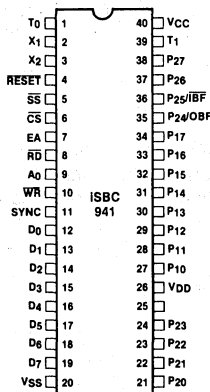


# iSBC 941 INDUSTRIAL DIGITAL PROCESSOR

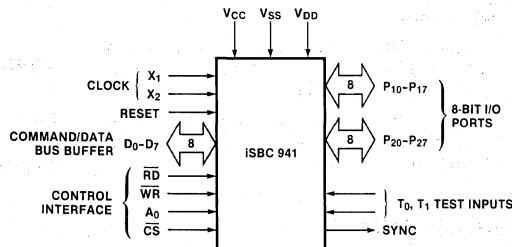
- Provides measurement and control of common industrial digital input and output signals
  - Sense change state
  - Pulse counting
  - Pulse generation
  - Period measurement
  - Frequency measurement
- Off-loads host processor from time-consuming task of digital I/O processing
- Simplified command protocol with MCS 80/85/86 "Master" Processor
- Compatible with 8041A Universal Peripheral Interface (UPI-41A) sockets such as those provided on Intel iSBC 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

## iSBC 941

	first-in-first-out (FIFO) buffer (used by SCAN and SEROUT Primary Functions).	(5) Enable selected SCAN input lines (Parameter byte) (enabled in groups of eight)
RDHR	— Enables the host processor to read the contents of the iSBC 941 holding register specified (used by PULSE, PERIOD, and EVENT Primary Functions).	(6) Terminate parameter list (Command byte)
RDIDV	— Enables the host processor to read the iSBC 941 processor's input-data-valid (IDV) flags.	(7) Enable iSBC 941 processor interrupt outputs (Command byte)
RDLC	— Enables host processor to read one of eight user specified EVENT counters previously stored by LATCH command.	(8) Request execution (Command byte)
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.	
SETS F	— 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 (Parameter byte) and select internal or external time reference
- (3) Program Time Reference Period (Parameter byte) (scan rate — this byte is required only if internal time reference was selected in (2) )
- (4) Define return message format (Parameter byte)

- (5) Enable selected SCAN input lines (Parameter byte) (enabled in groups of eight)
- (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 <sub>0</sub> -D <sub>7</sub>	Three-state, bidirectional COMMAND/DATA BUS BUFFER lines used to interface the iSBC 941 processor to an 8-bit master system data bus.
P <sub>10</sub> -P <sub>17</sub>	8-bit, PORT 1 quasi-bidirectional I/O lines.
P <sub>20</sub> -P <sub>27</sub>	8-bit, PORT 2 quasi-bidirectional I/O lines. Control can configure P <sub>24</sub> as OBF (Output Buffer Full), P <sub>25</sub> 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 <sub>0</sub>	Address input used by the master processor to indicate whether byte transfer is data or command.
T <sub>0</sub> , T <sub>1</sub>	Input pins used by various iSBC 941 processor routines.
X <sub>1</sub> , X <sub>2</sub>	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 <sub>CC</sub>	+5V power supply pin.
V <sub>DD</sub>	+5V during normal operation.
V <sub>SS</sub>	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<sub>A</sub> = 0°C to 70°C, V<sub>SS</sub> = 0V, V<sub>CC</sub> = + 5V ± 5%

Symbol	Parameter	Min.	Max.	Unit	Test Conditions
V <sub>IL</sub>	Input Low Voltage (All Except X <sub>1</sub> , X <sub>2</sub> )	-0.5	0.8	V	
V <sub>IH1</sub>	Input High Voltage (All except X <sub>1</sub> , X <sub>2</sub> , $\overline{\text{RESET}}$ , $\overline{\text{WR}}$ , $\overline{\text{CS}}$ )	2.0	V <sub>CC</sub>	V	
V <sub>IH2</sub>	Input High Voltage (X <sub>1</sub> , X <sub>2</sub> , $\overline{\text{RESET}}$ )	3.0	V <sub>CC</sub>	V	
V <sub>IH3</sub>	Input High Voltage ( $\overline{\text{WR}}$ , $\overline{\text{CS}}$ )	2.2	V <sub>CC</sub>	V	
V <sub>OL1</sub>	Output Low Voltage (D <sub>2</sub> -D <sub>7</sub> , Sync)		0.45	V	I <sub>OL</sub> = 2.0 mA
V <sub>OL2</sub>	Output Low Voltage (All Other Outputs Except Prog)		0.45	V	I <sub>OL</sub> = 1.6 mA
V <sub>OH1</sub>	Output High Voltage (D <sub>0</sub> -D <sub>7</sub> )	2.4		V	I <sub>OH</sub> = - 400 μA
V <sub>OH2</sub>	Output High Voltage (All Other Outputs)	2.4		V	I <sub>OH</sub> = - 50 μA
I <sub>IL</sub>	Input Leakage Current (T <sub>0</sub> , T <sub>1</sub> , $\overline{\text{RD}}$ , $\overline{\text{WR}}$ , $\overline{\text{CS}}$ , A <sub>0</sub> )		± 10	μA	V <sub>SS</sub> ≤ V <sub>IN</sub> ≤ V <sub>CC</sub>
I <sub>OZ</sub>	Output Leakage Current (D <sub>0</sub> -D <sub>7</sub> , High Z State)		± 10	μA	V <sub>SS</sub> + 0.45 ≤ V <sub>IN</sub> ≤ V <sub>CC</sub>
I <sub>LI1</sub>	Low Input Load Current (P <sub>10</sub> P <sub>17</sub> , P <sub>20</sub> P <sub>27</sub> )		0.5	mA	V <sub>IL</sub> = 0.8V
I <sub>LI2</sub>	Low Input Load Current ( $\overline{\text{RESET}}$ , SS)		0.2	mA	V <sub>IL</sub> = 0.8V
I <sub>DD</sub>	V <sub>DD</sub> Supply Current		15	mA	
I <sub>CC</sub> + I <sub>DD</sub>	Total Supply Current		125	mA	

**A.C. Characteristics**

T<sub>A</sub> = 0°C to 70°C, V<sub>SS</sub> = 0V, V<sub>CC</sub> = V<sub>DD</sub> = + 5V ± 5%

**DBB READ**

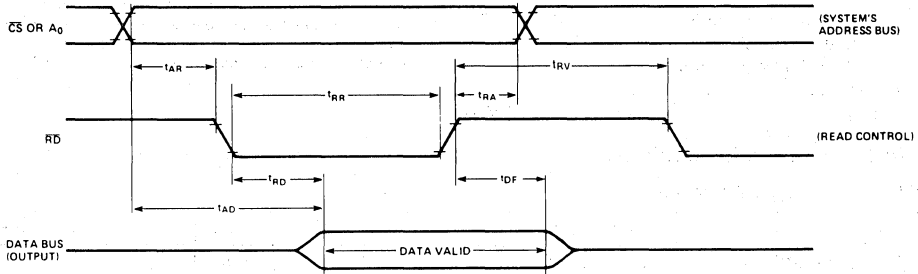
Symbol	Parameter	Min.	Max.	Unit	Test Conditions
t <sub>AR</sub>	$\overline{\text{CS}}$ , A <sub>0</sub> Setup to $\overline{\text{RD}}\downarrow$	0		ns	
t <sub>RA</sub>	$\overline{\text{CS}}$ , A <sub>0</sub> Hold After $\overline{\text{RD}}\uparrow$	0		ns	
t <sub>RR</sub>	$\overline{\text{RD}}$ Pulse Width	250		ns	
t <sub>AD</sub>	$\overline{\text{CS}}$ , A <sub>0</sub> to Data Out Delay		225	ns	C <sub>L</sub> = 150 pF
t <sub>RD</sub>	$\overline{\text{RD}}\downarrow$ to Data Out Delay		225	ns	C <sub>L</sub> = 150 pF
t <sub>RDF</sub>	$\overline{\text{RD}}\uparrow$ to Data Float Delay		100	ns	
t <sub>RV</sub>	Recovery Time Between Reads And/Or Write	300		ns	
t <sub>CY</sub>	Cycle Time	2.5	15	μs	

**DBB WRITE**

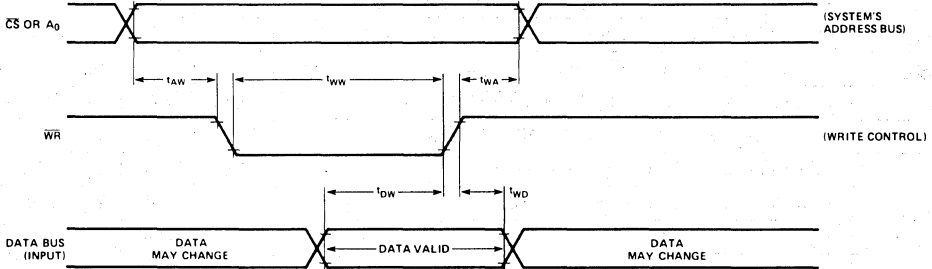
Symbol	Parameter	Min.	Max.	Unit	Test Conditions
t <sub>AW</sub>	CS, A <sub>0</sub> Setup to WR $\downarrow$	0		ns	
t <sub>WA</sub>	CS, A <sub>0</sub> Hold After WR $\uparrow$	0		ns	
t <sub>WW</sub>	WR Pulse Width	250		ns	
t <sub>DW</sub>	Data Setup to WR $\uparrow$	150		ns	
t <sub>WD</sub>	Data Hold After WR $\uparrow$	0		ns	

# iSBC 941

## 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*

**4**

---

THE UNIVERSITY OF CHICAGO  
DEPARTMENT OF CHEMISTRY  
5800 S. UNIVERSITY AVENUE  
CHICAGO, ILLINOIS 60637

Blank page with faint horizontal lines and scanning artifacts.



## 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.

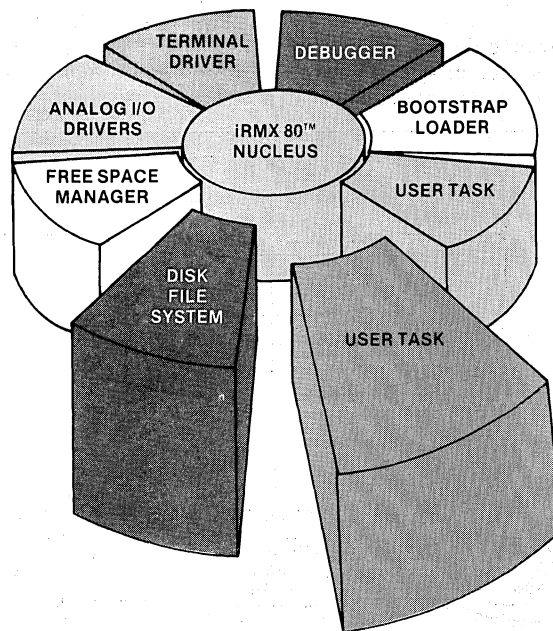


Figure 1. Structure Diagram



# iRMX 80

## 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

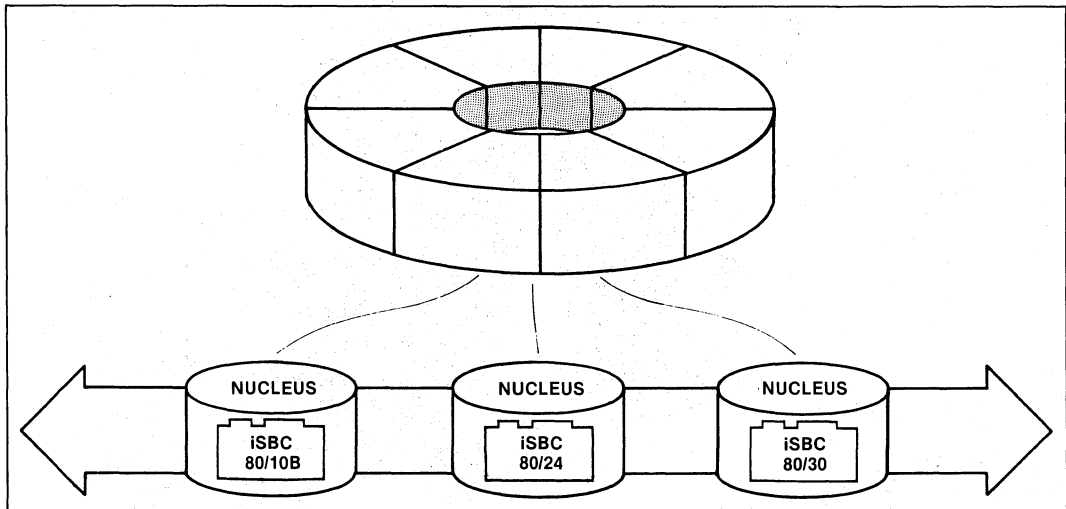
The iRMX 80 scheduler insures that the highest priority task which is ready to execute is given system control, allowing the application system to be responsive to its external world.

### 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 Intel 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.

# iRMX 80

## 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 Intel Development System and elicits the configuration information about the application system, e.g. CPU TYPE: *80/30*, or TERM HANDLER: *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.

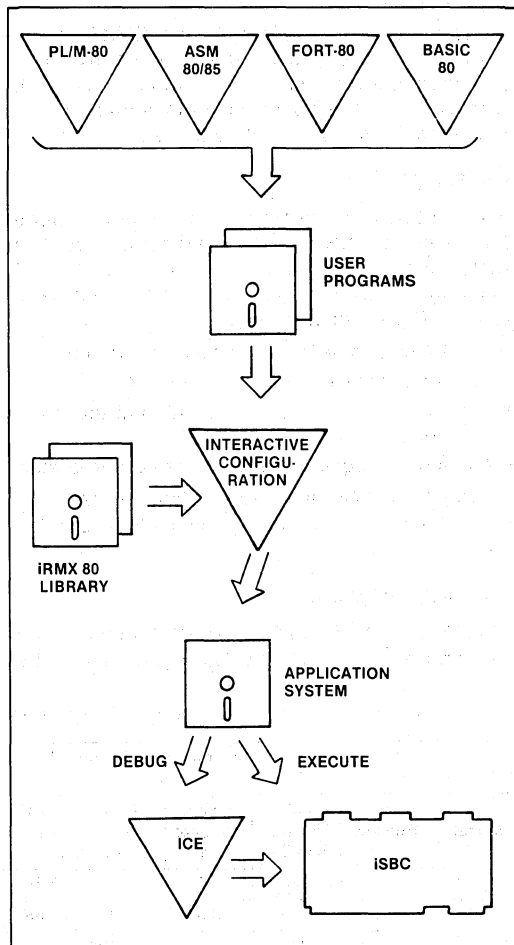


Figure 3. The System Generation Process

# iRMX 80

---

## 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 201  
iSBC 202  
iSBC 204  
iSBC 206

#### 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  
Registration Card  
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)

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

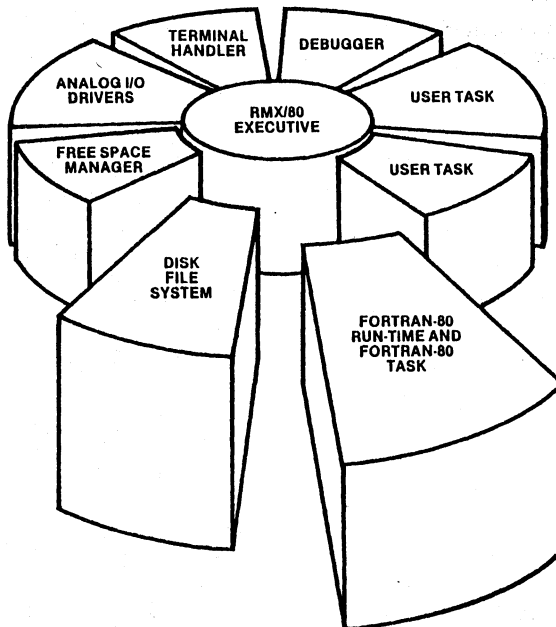
Part Number	Description
RMX 80	Real-Time Multi-Tasking Executive



## iSBC 801 FORTRAN-80 RUN-TIME PACKAGE FOR RMX/80 SYSTEMS

- The ideal iSBC 80 Single Board Computer run-time environment for programs written in Intel's ANSI FORTRAN 77 standard compiler
- Fully compatible with Intel's RMX/80 Real-Time Multitasking Executive software
- Includes FORTRAN-80 disk I/O via RMX/80 Disk File System
- Includes FORTRAN-80 terminal I/O via RMX/80 mini or full Terminal Handler
- Software floating point libraries are compatible with Intel floating point standard
- iSBC 310 High Speed Math Unit drivers allow acceleration of floating point operations

The iSBC 801 FORTRAN-80 Run-Time Package is a complete, ready-to-use set of linkable object modules which are fully compatible with RMX/80 systems. The modules, when combined with the FORTRAN-80 coded application, provide the appropriate interfaces to the disk file system and terminal handler of RMX/80. The FORTRAN-80 application, in addition to usage of the disk file or terminal I/O, can take advantage of the software floating point routines. In those cases where accelerated operations are necessary, the FORTRAN-80 Run-Time Package provides the necessary interface to use the iSBC 310 High Speed Math Unit.



**FUNCTIONAL DESCRIPTION**

The FORTRAN-80 Run-Time Package is a complete, ready-to-use set of linkable object module libraries on both a single and a double density diskette. It can be combined with the RMX/80 modules and user-supplied application software modules on the Intellec Development System. The FORTRAN-80 Run-Time Package for RMX/80 systems is the ideal environment for executing the FORTRAN-80 code in an ISBC 80 system. It provides full data formatting and storage capability through the RMX/80 Disk File System. The package allows utilization of the RMX/80 Terminal Handler for entry and display of information on a TTY or CRT. In addition, the FORTRAN-80 Run-Time Package facilitates effective use of the ISBC 310 High Speed Math board to optimize floating point calculations.

**Features and Benefits of the FORTRAN-80 Run-Time Package**

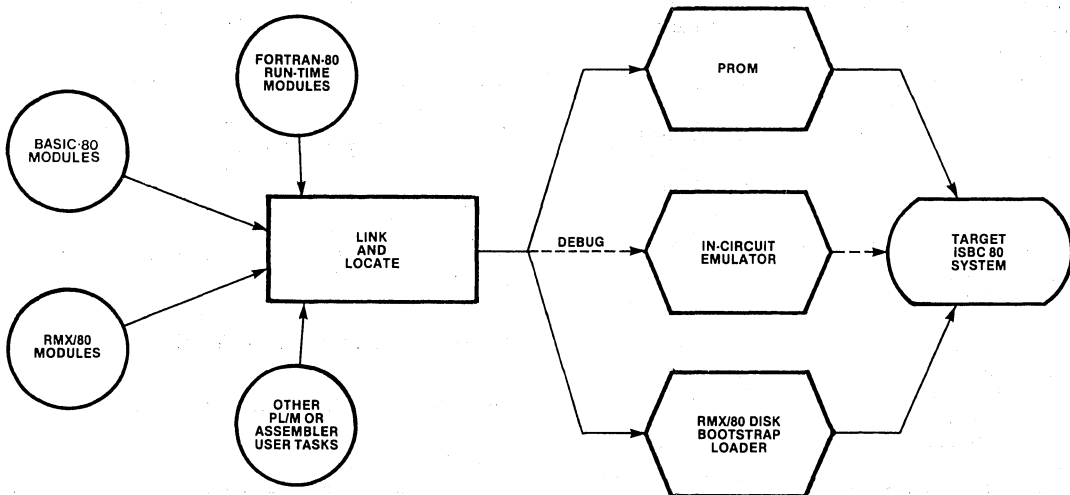
The modules included on the diskettes are readily usable by linking with the user-coded FORTRAN-80 relocatable object modules for operation in the RMX/80 environment. Advantages for the ISBC user include:

**Direct communication** with the mini or full Terminal Handler and Disk File System by using the formatted I/O facility in FORTRAN-80 coded application tasks. The user is able to specify the external format of data (integer, floating point, and character) as either input or output on those devices. The universality of formatted data statements in FORTRAN-80 facilitates storing relevant data for later retrieval, as well as summary display of the information. For example, to read two variables:

```
READ (5,88) X, Y
88 FORMAT (F10.3, F6.2)
```

**The run-time software support modules** are built for selective utilization at link time with FORTRAN-80 coded tasks. Selective linking can be profitably used to insure that only those modules which exactly match FORTRAN-80 application requirements are included in the run time systems. This insures that the minimum memory is required in the final system.

**Software modules** provided with the package are conveniently linkable with the RMX/80 library modules. This allows use of the right software tool (whether 8080/8085 Assembler, PL/M-80, or FORTRAN-80) for the right job. The user is also able to combine all the software elements into a highly effective RMX/80 based applica-



The RMX/80 executive based environment is specified by configuration parameters supplied at the time of system linking. These parameters define expected system characteristics and the relationship of FORTRAN-80 coded tasks to other RMX/80 modules. The linker output is the object code which may be placed in PROM or loaded into RAM via the bootstrap loader. Configuration parameters are discussed in detail in the RMX/80 User's Guide, Chapter 3.

**Figure 1. Configuration Flow Diagram**

# iSBC 801

tion. FORTRAN routines can be interfaced to PL/M or assembler coded routines so portable software can be run on all RMX/80 based systems.

**Application writing** with the floating point algorithms of FORTRAN-80 is enhanced thru the accuracy insured by FORTRAN-80 adherence to the Intel Floating Point Standard. Accuracy is insured thru software; it can be duplicated and accelerated thru the interface modules for the iSBC 310 High Speed Math Board. The iSBC customer now can effectively perform high speed, accurate computations with pre-coded and debugged mathematical functions.

## Generation Considerations

The FORTRAN-80 Run-Time Package consists of several relocatable library modules. The modules are configurable with the RMX/80 supplied library modules and user coded FORTRAN-80, PL/M-80 or 8080/8085 system. Through the proper linkage and location of the appropriate modules, a complete software solution for the target iSBC 80 system can be generated. This software then can be quickly loaded into the target iSBC 80 system through ICE-80 or 85 during debug. The final application code is placed in EPROM or loaded directly from disk via the RMX/80 bootstrap loader.

## SPECIFICATIONS

### Generation Environment

Intellec Microcomputer Development System with ISIS-II Diskette Operating System.

Minimum of 64K bytes RAM memory.

Dual Floppy Diskettes.

RMX/80 Real-Time Multi-Tasking Executive library diskettes.

FORTRAN-80 (V2.0 or later) Compiler diskettes.

### Compatible Run-Time Hardware

#### Single Board Computers:

iSBC 80/10A  
iSBC 80/20-4  
iSBC 80/30

#### Disk Controllers:

iSBC 202  
iSBC 204  
iSBC 206

#### Other:

iSBC 310 Math Unit

### Physical Characteristics

Product distributed on both a single and a double density diskette.

## Description and Capabilities Summary

To run FORTRAN-80 programs under RMX/80, libraries are selected from two packages: the RMX/80 Real-Time Multitasking Executive and the FORTRAN-80 Run-Time Package for RMX/80 systems. The FORTRAN-80 Run-Time Package provides libraries to support FORTRAN-80 coded programs running under RMX/80:

F80RMX.LIB	FORTRAN-80 input/output formatting and interface routines for the RMX/80 environment.
FPSFTX.LIB	Software floating point arithmetic routines for the RMX/80 environment.
FPHRDX.LIB	Software interface routines for the iSBC 310 facilitate floating point calculations in the RMX/80 environment of an iSBC 80/20 or iSBC 80/30 hardware based system. FPHX10.LIB is another library provided specifically for the iSBC 80/10 based RMX/80 system.

## Reference Manuals

The following material is shipped with the product:  
#9800480 ISIS-II FORTRAN-80 Compiler Operator's Manual

Additional information for FORTRAN-80 and RMX/80 can be found in the

#9800481 FORTRAN-80 Programming Manual

#9800522 RMX/80 User's Guide

#9800547 8080/8085 FORTRAN-80 Reference Card

Additional 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 801 FORTRAN-80 Run-Time Package for RMX/80 Systems

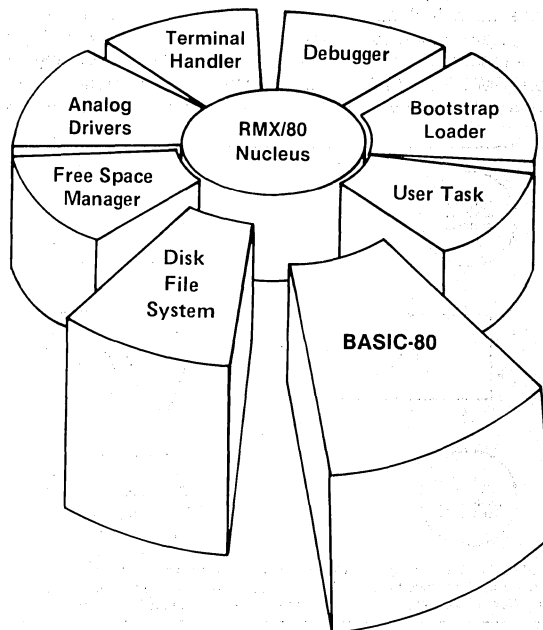
FORTRAN-80 Run-Time Package is copyrighted and licensed by Intel Corporation. It can be purchased only under a required license agreement with Intel. Software may be developed with the FORTRAN-80 Run-Time Package to run on Intel RMX/80 based systems.



## iSBC 802 BASIC-80 CONFIGURABLE RMX/80 DISK BASED INTERPRETER

- Resident BASIC interpreter for iSBC 80 Single Board Computers
- Sequential or random access diskette storage for programs and data
- Single and double precision software floating point arithmetic
- Direct port I/O instructions for special input and output requirements
- Interface up to 25 user defined software functions written in 8080/8085 assembler, PL/M-80, or FORTRAN-80
- Totally compatible with Intel's RMX/80 Real-Time Multitasking Executive software
- Exceeds the ANS X3.60-1978 and the ECMA-55 minimal BASIC standard
- Interpreter supplied in two forms:
  - A relocatable module for tailoring systems to exact requirements
  - A predefined version which incorporates the RMX/80 Nucleus, Terminal Handler, and Disk File System for an iSBC 80/30 and iSBC 204 based system

The iSBC 802 BASIC-80 Configurable RMX/80 Disk Based Interpreter has a complete ready-to-use set of linkable object modules which are fully compatible with Intel's RMX/80 Real-Time Multitasking Executive Software. The modules combine with RMX/80 to provide a BASIC-80 interactive interpreter system. BASIC-80 programs may be created, stored, and interpreted on the iSBC 80 based system. These programs may use the disk file and terminal I/O, software floating point, or interface to other routines provided by the user to tailor the BASIC-80 system.





**FUNCTIONAL DESCRIPTION**

The RMX BASIC-80 Interpreter provides a high level language interpreter with extended disk capabilities which operates under the RMX/80 Real-Time Multitasking Executive and translates BASIC-80 source programs into an internally executable form. This universally accepted language is designed for the OEM who requires a pass through programming language suitable for usage by both beginners and experienced programmers. BASIC-80 offers users an expedient method of utilizing the powerful computational and input/output capabilities of an iSBC 80 microcomputer system and applying them to solving a wide range of application problems. 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 bitwise Boolean logical operators. In addition, Intel's BASIC-80 meets and exceeds the requirements of ANS X3.60-1978 Minimal BASIC Standard and, similarly, the Standard ECMA-55 Minimal BASIC.

**Dual Solution**

The iSBC 802 provides BASIC-80 to the OEM in two forms. First, the product is supplied as a set of linkable object modules on both a single and a double density diskette which can readily be combined with RMX/80 library and user application modules. The Intellec Development System is used to form a complete software solution for the iSBC 80 based system from this configurable version. Second, for users who want an "instant-on" BASIC system and have no need to add additional software routines for their end product, a predefined BASIC-80 software system has been included in the iSBC 802. The predefined system includes 1) the RMX/80 and BASIC-80 software modules configured into an executable object module (RMXSYS) which is supplied on the same single and double density diskettes

as the relocatable BASIC-80 modules, 2) a set of bootstrap PROMs that will load the executable RMXSYS module from the diskette into the iSBC main memory, and 3) a cable that connects the iSBC 204 Floppy Diskette Controller to the bulkhead connector on any Intellec diskette drive chassis. The predefined system offers an "instant-on" software capability for the user's supplied hardware system consisting of the following items:

- iSBC 80/30 Single Board Computer
- iSBC 204 Universal Flexible Diskette Controller
- 32K bytes additional RAM memory (either iSBC 032 or two iSBC 016s)
- Compatible teletypewriter terminal
- Compatible diskette drives

**RMX/80 BASIC-80 FEATURES AND BENEFITS**

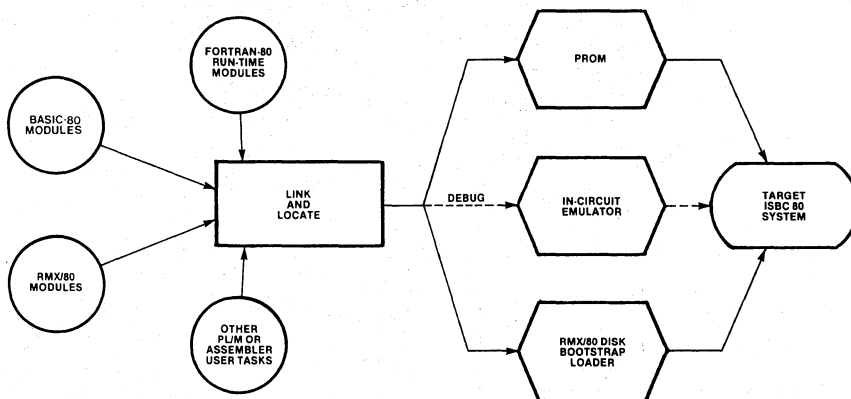
**Meets New Industry Standard**

BASIC-80 provides the OEM and end-user with a resident programming language for the iSBC 80/10A, iSBC 80/20, iSBC 80/20-4, or iSBC 80/30 based systems. A desirable and cost effective language feature is its PROM or disk resident capability. BASIC-80 also complies with an industry standard that defines the Minimal BASIC language specification and provides a level of confidence and structure not previously available to the OEM with BASIC interpreters. Since Intel's BASIC-80 meets this standard, the user can be assured of a maximum amount of compatibility with other BASIC interpreters.

**Major Features of ANS X3.60-1978 Include:**

**Constants and Variables**—The user can describe and manipulate constants and variables in a form most suited to the application. The numeric expression can be constructed from variables, numeric constants, and function references. The string expression is composed of a string variable or a string constant.

**Numeric Functions**—Commonly used single precision scientific numeric functions are provided for cosine, sine,



The RMX/80 based environment is specified by configuration parameters supplied at the time of system linking. These parameters define expected system characteristics and the relationship of the BASIC-80 tasks to other RMX/80 modules. The linker output is the object code which may be placed in EPROM or loaded into RAM via the bootstrap loader. Configuration parameters are discussed in detail in the RMX/80 User's Guide, Chapter 3.

**Figure 1. Configuration Flow Diagram**

tangent, arctangent, square root, exponentiation, absolute value, integer, sign of a number, log, and random number generation. In addition, RANDOMIZE can be used to override the predefined pseudo-random number values. All underflow and overflow conditions are recognized and reported in a consistent and defined manner.

**Program Control and Sequencing**—Specific program control and the sequencing of statement execution is provided.

**Assigned Variable Values**—Variables can be assigned values by the use of DATA statements which can be READ and manipulated through the RESTORE statement.

**Custom Functions**—The user can define custom functions for use during program execution.

**Indexed Arrays**—Arrays can be indexed starting with 0 or 1 and the array's size is established through a dimension (DIM) statement.

**Interactive I/O**—Program directed prompts can be sent to the terminal through the INPUT statement in addition to accepting input data values. Effective tabular output is accomplished through the PRINT statement which will cause the transmission of a character string to an external device.

## EXTENSION BEYOND INDUSTRY STANDARD

Intel's BASIC-80 has enhancements and extensions beyond the standard. These include the following:

**PROM Storage**—BASIC-80 interpreter and program source can be stored in PROM and then executed at system startup. This allows users to take advantage of special routines they have written in other languages, PL/M-80, FORTRAN-80, or 8080/8085 Assembler.

**Floating Point Standardization**—The OEM can code floating point algorithms in BASIC-80 because the accuracy is insured by the BASIC-80 interpreter's adherence to the Intel Floating Point Standard. Single precision is accurate to a maximum value  $3.40282E + 38$  and double precision is accurate to a maximum value of  $1.79769313486231 D + 308$  for the utmost in creditability for scientific and mathematical computation.

**File Management Statements**—Abundant file managing statements aid in the orderly usage of programs and data stored on disk files.

**Direct Port I/O**—Communicate directly to the I/O ports of the iSBC product line through the byte oriented INP and OUT instructions. Use the WAIT statement to mask inputs from a port until the indicated bit pattern appears. Specific memory locations can be examined with the PEEK statement or modified via the POKE statement.

**Formatted Output**—Valuable output formatting of data can be accomplished through the PRINT USING formatting characters, i.e. asterisk (\*) fill, floating dollar sign (\$), leading or trailing sign (+, -), comma (,), insertion and scientific notation (exponent display).

**Enhanced Interactive Programming**—Interactive programming is enhanced through easy to use source line modification commands. Program modules stored on disk can be readily MERGED with statements in memory.

## GENERATION CONSIDERATIONS

### Generation of Configurable Version

The configurable portion of the iSBC 802 BASIC-80 package consists of several relocatable library modules. These modules are configurable with the appropriate RMX/80 library routines (Minimal Terminal Handler, Nucleus, Disk File System and Loader) and the other application tasks to form a full system. Through the proper linkage and location of the appropriate modules, a complete software solution for the iSBC 80 system can be generated. This software then can be quickly loaded into the iSBC 80 system through ICE 80 or 85 during debug. The final application code is placed in EPROM or loaded directly from disk via the RMX/80 bootstrap loader.

## DESCRIPTION AND CAPABILITIES SUMMARY

### Configurable Version

To create a tailored BASIC-80 System for an iSBC 80 based product, libraries are selected from two packages: the RMX/80 Real-Time Multitasking Executive and the BASIC-80 Configurable RMX/80 Disk Based Interpreter package. The BASIC-80 Configurable RMX/80 Disk Based Interpreter package provides the following files for use with RMX/80 (Version 1.3 and above).

BASIC.LIB	Necessary modules for linkage with RMX/80
BQOPS.ASM	"Include" source file for BASIC-80 options related to the creation of a tailored system
BASCM.ASM	The macro assembler source for the Configuration Module used for RMX/80
BQMEM.ASM	Module to describe the memory allocation for the BASIC-80 task
BOOTCM.ASM	Assembler source for the Configuration Module used by the bootstrap loader
BQBMEM.ASM	Module to describe the memory allocation for the system to the bootstrap loader

### Predefined Version

The following modules are related to the predefined BASIC-80 System which can be loaded via the supplied bootstrap loader.

RMXSYS	Predefined BASIC-80 System software configured, linked, and located to support the required configuration
BQBOOT	Exact copy of the bootstrap loader as supplied on the ROMs for the predefined system

---

## SPECIFICATIONS

### Configurable Version Generation Requirements

Intellec® Microcomputer Development System with ISIS-II Diskette Operating System

Minimum of 48K bytes RAM memory

Dual Floppy Diskettes

RMX/80 Real-Time Multitasking Executive library diskettes (Version 1.3 and above)

### Compatible Run-Time Hardware

Single Board Computers

iSBC 80/10A

iSBC 80/20-4

iSBC 80/30

Disk Controllers

iSBC 202

iSBC 204

iSBC 206

### Predefined Version Generation Requirements

No system generation is required as the software system has been predefined to execute in the run-time environment specified below:

iSBC 80/30

iSBC 204 disk controller

Additional 32K bytes memory (either iSBC 032 or two iSBC 016 memory boards)

Compatible Intellec diskette drives and cable (cable supplied with iSBC 802)

RS232C compatible terminal

### Physical Characteristics

Product distributed on both a single and double density diskette. Each diskette contains both the modules for configuring a tailored system and the predefined software system.

Two ROMs which contain the bootstrap loader.

One cable for the predefined system which provides a connection from the iSBC 204 to the user's Intellec Diskette Drive Chassis (MDS DDS, MDS DDR, and MDS 2DS, MDS 710, MDS 720, MDS 730).

### Reference Manuals

The following material is shipped with the product:

9800758 BASIC-80 Reference Manual (SUPPLIED)

9800774 BASIC-80 Pocket Reference (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 802	BASIC-80 Configurable RMX/80 Disk Based Interpreter

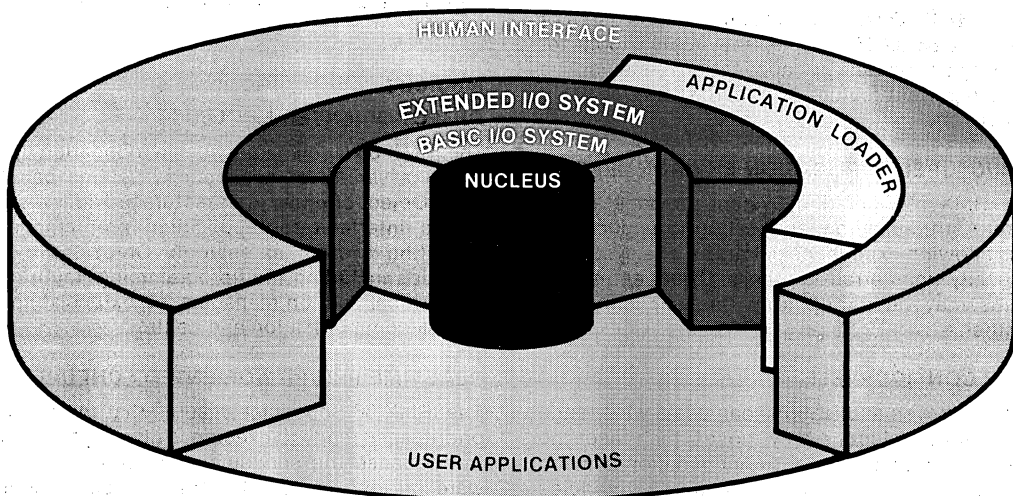
BASIC-80 Configurable RMX/80 Disk Based Interpreter is copyrighted and licensed by Intel Corporation. It is available only under a license agreement with Intel. Software may be developed with the BASIC-80 to run on Intel RMX/80 based systems.



## iRMX 86 OPERATING SYSTEM

- Structured application environment
- User configurable
- User extensible
- PROM or RAM based
- Object-oriented nucleus
- Real-time priority-oriented scheduling
- Powerful error management
- Interactive system debugger
- Comprehensive I/O system
- Extensive human interface

The Intel® iRMX 86 Real-Time Operating System is an easy-to-use, sophisticated software system which operates on Intel iSBC 86/12 boards and user iAPX 86 (8086)-based boards. The iRMX 86 Operating System extends the iAPX 86 (8086) architecture, providing a structured, efficient environment for many applications, including process control, intelligent terminals, office systems, medical electronics, and data communications.



iRMX 86™ Operating System Layers

## 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; the means to concurrently monitor and control multiple external events; real-time clock control; interrupt management and task dispatching. The iRMX 86 Operating System contains the following modules: a Nucleus, a device independent Input/Output Subsystem, a Human Interface Subsystem with command language interpreter and ASCII console interface, a Terminal Handler, and an interactive object-oriented Debugging Subsystem.

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 thus eliminates the need for custom operating system design and hence reduces development time, cost and risk.

### Feature Overview

The iRMX 86 Operating System provides users of the iAPX 86 (8086) simple, easy-to-use tools for creating a wide range of application systems. The most important features of the iRMX 86 Operating System are:

#### STRUCTURED APPLICATION ENVIRONMENT

The iRMX 86 Operating System provides a consistent structure from application to application, thus allowing experience gained on one system to be easily transferred to others. Often, entire programs may be ported from one application to another.

#### USER CONFIGURABLE

iRMX 86-based applications can use a wide range of facilities, selecting only those which meet the specific requirements of the application system. The resultant system contains only the modules necessary for its use, allowing the iRMX 86 Operating System to be cost-effectively applied in a wide range of application environments from performance-oriented industrial applications, to security-conscious data-processing systems.

The iRMX 86 Operating System is constructed in a thoroughly modular manner with the full range of

facilities being offered in library modules, allowing easy selection of the exact features required. Overhead for unused facilities is then eliminated.

#### USER EXTENSIBLE

The iRMX 86 Operating System provides a framework in which to extend the system and have these extensions look like facilities provided by Intel. These extensions include not only the ability to add custom system calls, but also the ability to add operating system data structures. Check-out of these extensions is easily accomplished by using the Debugging Subsystem since extensions become an integral part of the operating system. Because the operating system is designed for user extensibility, the extensions can be added in a symmetric manner, resulting in a homogeneous customized operating system.

#### EPROM OR RAM BASED

The iRMX 86 system can be EPROM resident or loaded from a mass storage device into RAM, depending upon application requirements. Being able to place all of the software in EPROM offers two benefits. First, if the application is in a harsh environment, mass storage devices cannot be used because of the danger of contamination. Second, if the application is small, the expense and overhead of mass storage devices is eliminated.

#### Nucleus

##### OBJECT-ORIENTED NUCLEUS

The iRMX 86 Nucleus provides a foundation upon which a variety of application systems can be built. Object-oriented architecture provides a symmetric interface for application programs and operating system extensions. Objects provide facilities including multiprogramming, multitasking, critical section management, extensive task-to-task communication and control.

##### REAL-TIME PRIORITY-ORIENTED SCHEDULER

The iRMX 86 Scheduler ensures that the highest priority task ready to execute is given system control because the scheduler recognizes 255 software priority levels. Also, the system supports eight hardware priority levels, allowing the application system to be responsive to its external environment.

##### ERROR MANAGEMENT SUBSYSTEM

The iRMX 86 Operating System provides extensive error management and reporting mecha-

nisms. Both excessive system loading and user programmer errors can be reported, reducing system debug time. The flexibility of the Error Management Subsystem allows errors to be serviced directly by the user task or sent to a specific error handler.

## Debugging System

The iRMX 86 Operating System provides interactive software debugging. The Debugging Subsystem has two capabilities that greatly simplify the process of debugging a multitasking system. First, the Debugger allows you to debug several tasks while the balance of the application system continues to run in real-time. Second, the Debugger lets the programmer interactively view and modify system constructs as well as RAM and iAPX 86 (8086) registers.

## I/O System

The iRMX 86 device-independent I/O Subsystem provides a standard interface for application programs to communicate with all I/O devices. File management systems provide powerful features like a hierarchical directory structure for quick, efficient file access. A standardized device driver interface allows users to easily create custom device drivers. A wide range of standard drivers are available including; iSBC 204 Single Density Diskette Controller, iSBC 206 Hard Disk Controller, and terminal handler (via on-board 8251 USART device).

## Human Interface

The iRMX 86 Human Interface provides a powerful man-machine interface for interactive control of system resources and utilities. iRMX 86 system utilities include display file directories, copy files, rename files, etc. The Intel supplied command line interpreter is table driven, allowing easy modification by the user for the creation of application-oriented commands.

## SYSTEM LAYERS AND FACILITIES

### Nucleus

#### FEATURES

- Object-oriented architecture
  - Dynamic memory manager
  - Multitasking
  - Multiprogramming
  - Interprogram and intertask communication and control

- Critical section management
- User extensible

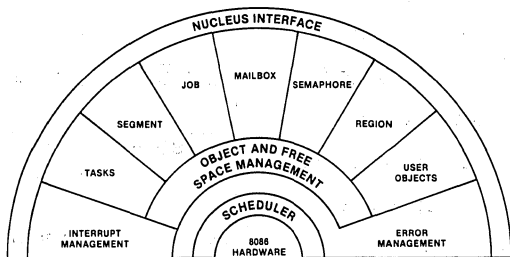
- Real-time priority-oriented scheduling
- Interrupt management
- Extensive error management

## OVERVIEW

The Nucleus is the heart of the iRMX 86 Operating System. All other subsystems require the Nucleus.

Embedded in the Nucleus are two main facilities: first, the facility for concurrent program execution; second, the facility for handling simultaneous asynchronous events. iRMX 86 systems solve problems that require real-time response. For example, interrupts coming from specialized peripheral devices can be serviced in an efficient manner. iRMX 86 systems allow the hardware to be used by more than one application, thus reducing the overall system size and hardware cost.

The object-oriented architecture provides the power necessary to solve the complex problems of today and tomorrow. The system is comprised of uniform system calls ensuring that the Nucleus is both easy to learn and easy to extend. Only four concepts must be mastered to learn the system: object management, interrupt management, the scheduling algorithm, and error management.



### Nucleus Application Code Interface

## OBJECT-ORIENTED ARCHITECTURE

The Nucleus is composed primarily of "objects." Objects are data structures with a fixed set of attributes. Just as a floating point number is a data structure with operators to manipulate it, like add or multiply, iRMX 86 objects have system calls to manipulate them. Because of the uniform struc-

# IRMX 86

ture of the system, users have a foundation on which to tailor the Nucleus to the application by adding application-specific objects and system calls.

iRMX 86 objects include:

- SEGMENTS, for dynamic buffering
- MAILBOXES, for intertask and interprogram object and data transfer.
- SEMAPHORES, for intertask and interprogram synchronization and mutual exclusion
- REGIONS, for critical section management
- TASKS, for code execution
- JOBS, for program isolation
- USER OBJECTS, for system extensions

All iRMX 86 objects are dynamic; they can be created and deleted. The free space memory man-

ager automatically allocates and deallocates available RAM memory when objects are created and deleted.

## Object Description

**SEGMENT** — Dynamically created RAM buffer with a specified length used to store data.

**MAILBOX** — Provides intertask and interprogram object and data transfer. Mailboxes are locations for objects to be sent and retrieved. For example, 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 from task to task, although any object (user- or system-provided) may be passed.

**SEMAPHORE** — Manages mutual exclusion and synchronization. A semaphore, an integer be-

## Nucleus Object Management System Calls

System Calls for All Objects	O.S. Objects	Attributes	Object-Specific System Calls
CATALOG\$OBJECT UNCATALOG\$OBJECT LOOKUP\$OBJECT ENABLE\$DELETION DISABLE\$DELETION FORCE\$DELETE GET\$TYPE	JOBS	Tasks Memory pool Object directory Exception handler	CREATE\$JOB DELETE\$JOB SET\$POOL\$MIN GET\$POOL\$ATTRIB OFFSPRING
	TASKS	Priority Stack Code State Exception handler	CREATE\$TASK DELETE\$TASK SUSPEND\$TASK RESUME\$TASK GET\$EXCEPTION\$HANDLER SET\$EXCEPTION\$HANDLER SLEEP GET\$TASK\$TOKENS GET\$PRIORITY SET\$PRIORITY
	SEGMENTS	Buffer with length	CREATE\$SEGMENT DELETE\$SEGMENT GET\$SIZE
	MAILBOXES	List of objects List of tasks waiting for objects	CREATE\$MAILBOX DELETE\$MAILBOX SEND\$MESSAGE RECEIVE\$MESSAGE
	SEMAPHORES	Semaphore unit value List of tasks waiting for units	CREATE\$SEMAPHORE DELETE\$SEMAPHORE RECEIVE\$UNITS SEND\$UNITS
	REGIONS	List of tasks waiting for critical section	CREATE\$REGION DELETE\$REGION RECEIVE\$CONTROL ACCEPT\$CONTROL SEND\$CONTROL
	USER OBJECTS	License rights to a given extension type	CREATE\$EXTENSION DELETE\$EXTENSION
		New object template	CREATE\$COMPOSITE DELETE\$COMPOSITE INSPECT\$COMPOSITE ALTER\$COMPOSITE

tween.0 and 64K, is used to signal another task when processing has been completed or when resources are available. A semaphore provides a low overhead signalling mechanism.

**REGION** — Controls access to critical sections. Regions allow only one task at any given time to access a portion of code. Examples are a non-reentrant procedure or code for controlling a peripheral device that can only service one request at a time. In addition, regions can also be used to protect data structures from being manipulated by more than one procedure at a time.

**TASK** — Software modules that execute sequentially. Each task in the system has the characteristics of a processor. It has its own code, priority level, stack area, data area, and status. Software priority levels determine the task's eligibility to execute; there are 255 levels. Task execution is based on an event-driven priority-oriented scheduling algorithm.

**JOB** — Permits isolation of application tasks, objects, and memory, providing a multiprogramming environment. Jobs encapsulate an application and limit the degree of interaction between one set of tasks and another.

**USER OBJECTS** — System programmer's facility for creating objects not found in the iRMX 86 system. These new objects appear to other facilities in the iRMX 86 system as if part of the original operating system. This means that the system calls that manipulate system objects also manipulate user defined objects.

## NUCLEUS INTERRUPT MANAGEMENT

### Features

- Two levels of interrupt management
- User selectable priority for interrupt response

### Overview

The iRMX 86 Operating System is responsive to external events occurring asynchronously in real time.

### Interrupt Management

The iRMX 86 Operating System provides two levels of interrupt management; an interrupt handler or an interrupt task. One provides speed, the other power. The interrupt task permits all iRMX 86 system calls to be issued and masks only lower priority interrupts. The interrupt handler permits only iRMX 86 interrupt system calls to be issued and masks all interrupts.

## Priority Levels

Handling an interrupt with respect to other work in the system can be completely user controlled. iRMX 86 external interrupt levels are directly related to task priorities. The priority of the executing task determines which interrupts are masked. Generally, software background tasks will be assigned a lower priority than interrupt tasks to avoid blocking interrupts.

## NUCLEUS SCHEDULING

### Features

- Real-time priority-oriented
- Timed wait for system resources
- Dynamic control of task execution

### Overview

The iRMX 86 scheduler executes tasks on the basis of three criteria: relative importance of the task, elapsed time, and task state.

### Real-Time Priority Scheduling

The iRMX 86 Nucleus offers a priority-oriented event-driven scheduling mechanism. The scheduler uses the task priority to determine which task receives CPU time. The scheduler ensures that the highest priority task ready to execute is given system control. That task will continue to run until a higher priority interrupt occurs, or until the running task requests resources that are not available. Priority scheduling allows the system to be responsive to the external environment while still devoting resources only to tasks that have work to be performed.

### Timed Wait

A task can specify the time it will wait for an event to occur or wait for system resources to become available. A task can limit the time it will wait to receive an object at a mailbox, to receive units from a semaphore, or to look-up an object in the directory. In addition, a task can specify how long it will remain dormant before executing any further instructions. This allows the user to more finely control the execution of the task and guard against possible system deadlock.

### Dynamic Control of Task Execution

Dynamic control over task execution can be completely user controlled. Often, it is important to temporarily halt the execution of a task. (This is known as suspending a task.) If a task has been suspended N times, then it must be resumed N



times before it will be permitted to run. The system is thus protected against a suspended task executing before it has been resumed the proper number of times.

## ERROR MANAGEMENT

### Features

- Hierarchical error handling
- Selective error processing

### Overview

When a task issues an iRMX 86 system call the results may not be what the task is trying to achieve. For example, the task may request memory that is not available, or it may use an invalid parameter. When these conditions occur, they are reported by the iRMX 86 error management system.

### Hierarchical Error Handling

There is a hierarchy of error handlers within the iRMX 86 system. If errors are to be handled uniformly, then a system error handler is used. If a job has need of a different error handling facility, then the job's errors are handled at the job level. In addition, if a task has need for a unique error handler, then an error handler can be specified for that task. This flexible error handling means global error handlers can be created for the majority of the errors, reducing the amount of application software to be written. Yet, jobs or tasks with specific needs can trap and handle errors in a manner unique to them.

### Selective Error Processing

Errors can be processed according to two failure modes: programmer errors and environmental conditions. "Insufficient memory" for example would be classified as an environmental condition. An invalid parameter in a system call would be classified as a programmer error. Whether a programmer error or an environmental condition, errors of a given category can be serviced in-line or cause the system to vector to the specified error handler.

## Terminal Handler

### Features

- Line buffering and character echoing
- Keystroke control for line editing

### Overview

The Terminal Handler supports real-time, asynchronous I/O between the operator's terminal and

tasks running under the iRMX 86 Nucleus. It is intended for use in applications which require only limited I/O through a terminal and is generally used in applications that do not include an iRMX 86 I/O Subsystem. An output-only version of the Terminal Handler is for use in applications that send output to a terminal but do not receive input.

### Line Buffering and Character Echoing

The Terminal Handler provides a line buffer which stores ASCII characters as they are input to the console. Special editing characters are not placed in the line buffer. Characters are echoed to the terminal when they are stored in the line buffer.

### Keystroke Control

The Terminal Handler supports operator control over both input and output. A table of special characters is shown.

**Terminal Handler Editing Characters**

Special Character	Effect
Rubout	Deletes previously entered character.
Carriage Return	Signals end of line.
Line Feed	Signals end of line.
Escape	Signals end of line.
control-C	Aborts an application program.
control-O	Kills or restarts output.
control-Q	Resumes suspended output.
control-R	Displays current line with editing.
control-S	Suspends output.
control-X	Deletes the current line.
control-Z	Sends empty message.

## Debugging Subsystem

### Features

- Object-oriented debugger
- User-oriented interactive on-line debugging facility
- Task execution breakpoint facility

### Overview

A powerful Debugging Subsystem which is both object-oriented and interactive is provided to ease the chore of debugging application systems.

### Object-Oriented Debugger

The iRMX 86 Operating System provides a special debugger that recognizes iRMX 86 objects. Both

system objects and user objects can be viewed using the debugger.

**Interactive On-Line Facilities**

The Debugging Subsystem's own terminal handler includes all the features available in the stand-alone Terminal Handler. Interactive system checkout allows the user to view iRMX 86 system lists including the lists of the jobs, the ready tasks, the suspended tasks, the tasks queued at mailboxes, semaphores, regions, and the objects queued at mailboxes.

**Task Execution Breakpoint Facility**

The Debugging Subsystem allows the application programmer to debug several tasks while the balance of the application system continues to run in real time using the task breakpoint facility. Task breakpoints can be set, viewed and changed.

**Basic I/O Subsystem**

**FEATURES**

- Device-independent file driver interface
- Device driver interface

**OVERVIEW**

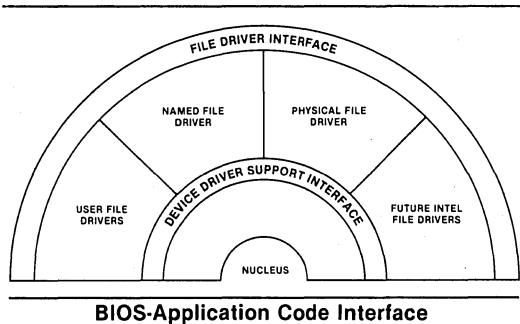
The Basic Input/Output Subsystem, BIOS, is an optional subsystem that provides input and output facilities for mass storage devices and communication with I/O devices in a device-independent manner. For example, the BIOS translates a READ request into device-specific instructions, freeing the application programmer from these details. Because the BIOS provides a standard interface for all I/O requests, programmers need not rewrite application code when adding or changing I/O devices. To incorporate application-specific peripheral devices in an iRMX 86 system only a small amount of code, a device driver, needs to be written.

The BIOS has two interfaces: one for performing device-independent application I/O requests (file driver interface) and one for adding device drivers (device driver interface).

**DEVICE-INDEPENDENT FILE DRIVER INTERFACE**

**Features**

- Two file manager support options
- Logical device naming for constant I/O application interface



**Overview**

All I/O requests are performed on "files". Files can be created, deleted, opened, closed, read, and written. The system provides two file manager support options, named file support and physical file support. These file managers offer different I/O support and I/O-application interfaces. I/O requests to a device can be issued using the file managers supported by the actual driver.

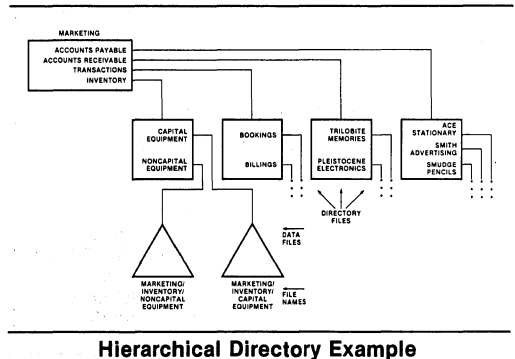
**Named File Manager**

**FEATURES**

- Hierarchical directory support for mass storage devices
- File access protection and control
- Control over file fragmentation

**OVERVIEW** — The named file manager provides support for mass storage devices. Application programs reference data through a directory rather than by physical track and block location, thus allowing data to be given a name and referenced by that name.

**HIERARCHICAL DIRECTORY SUPPORT** — The BIOS supports a hierarchical directory structure, allowing data to be grouped logically and allowing quick directory lookup. An example is shown.



**FILE ACCESS PROTECTION AND CONTROL** — Besides logical data grouping and naming, both access mode and password protection are supported to ensure file integrity. When a file is created the creator determines access rights: read, write, update, and append. The file is also password protected and only the creator of a file can grant another user access rights. These rights may be different from the creator's. For example, a creator who can "read" or "write" a file has the ability to grant "read-only" access to another user.

**CONTROL OVER FILE FRAGMENTATION** — Application programs specify file granularity at the time a file is created. Because files are stored on random locations on the disk, wasted space is eliminated by using small file granularity. When file granularity is increased, access time for data retrieval is improved. The BIOS allows the user to decide whether time or space is more important.

### Physical File Manager

A direct interface to the device driver support facilities is provided via the physical file manager. No file directory structure is associated with the physical file manager. Line printers, A/D and D/A

converters are typical kinds of devices that use the physical file manager.

### Logical Device Naming

Logical names are used when specifying a particular device rather than a physical address, causing physical device changes not to require a rewrite of application code.

### DEVICE DRIVER INTERFACE

#### Features

- Mass storage device drivers
- Support for custom device drivers
- Run-time device driver binding

#### Overview

The iRMX 86 system offers extensive device driver support, enabling users to easily add device drivers to the system. The same device driver code can be used for more than one physical device.

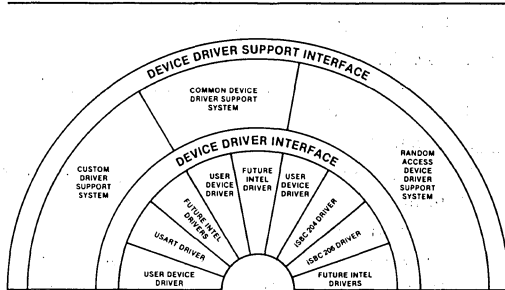
### Mass Storage Device Drivers

The iRMX 86 system includes two mass storage device drivers. The iSBC 204 Universal Flexible Diskette Controller for standard flexible diskette

## BIOS-Application System Call Interface

Universal File Calls	File Management Options	File Types	File Specific Calls	Access Mode Protection
ATTACH\$FILE DELETE\$CONNECTION GET\$FILE\$STATUS GET\$CONNECTION\$STATUS	<b>Named</b> <ul style="list-style-type: none"> <li>• Hierarchical Directories</li> <li>• Access Mode Protection</li> <li>• Password Protection</li> </ul>	<b>Named Data</b>	CREATE\$FILE DELETE\$FILE OPEN CLOSE READ WRITE SEEK RENAME\$FILE CHANGE\$ACCESS GET\$PATH\$COMPONENT	Delete Read Write Update Append
			<b>Named Directory</b>	CREATE\$DIRECTORY DELETE\$FILE GET\$DIRECTORY\$ENTRY RENAME\$FILE CHANGE\$ACCESS GET\$PATH\$COMPONENT
	<b>Physical</b> <ul style="list-style-type: none"> <li>• Direct Device Driver Support Interface</li> </ul>	None	CREATE\$FILE OPEN CLOSE READ WRITE SEEK	None

drives is supported. For larger mass storage requirements, the iSBC 206 Hard Disk Controller offers 10 MB on-line storage.



**Device Driver Interface**

### Support for Custom Device Drivers

Custom drivers can also be added to the iRMX 86 system. There are three levels of device driver support: random access device driver support, common device driver support, and custom device driver support. Regardless of the driver support option and the file management option, only four routines need to be written for the driver. Also, an interrupt routine must be written for common and random access drivers. The semantics of these routines differ according to the device driver support option used. These are:

**RANDOM ACCESS DRIVER SUPPORT** — High-level support for random access devices. Assumptions for its use are: deblocking of requests is necessary, one interrupt level is sufficient, and random-peek must be supported.

**COMMON DRIVER SUPPORT** — High-level support for common devices. Common driver support assumes no deblocking of requests is necessary, FIFO I/O requests are sufficient, and one interrupt level is sufficient.

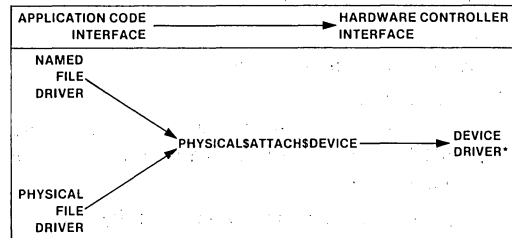
**CUSTOM DRIVER SUPPORT** — Complete flexibility to the device driver writer. No I/O request scheduling is provided, giving complete I/O request queuing responsibility to the device driver. No assumptions are made regarding interrupt levels or blocking or deblocking of requests.

### Run-Time Device Driver Binding

The application program specifies the file management system and device driver; device drivers are attached to the system at run-time. The `PHYSICAL$ATTACH$DEVICE` system call estab-

lishes the binding and thereby the type of file support used for a given device.

### Run-Time Device Driver Binding



## iRMX 86 System Calls

### NUCLEUS OBJECT MANAGEMENT SYSTEM CALLS

#### CATALOG\$OBJECT

Catalogs the specified object in the given job's object directory.

#### UNCATALOG\$OBJECT

Uncatalogs object from the directory.

#### LOOKUP\$OBJECT

Returns access to the specified object.

#### ENABLE\$DELETION

Enables an object to be deleted based on the disable-deletion depth.

#### DISABLE\$DELETION

Disables an object from being deleted by incrementing the disable-deletion depth.

#### FORCE\$DELETE

Decrements disable-deletion depth by one, then proceeds to delete the object.

#### GET\$TYPE

Returns the specified object's type code.

#### CREATE\$JOB

Creates the job with an initial task, memory pool, an empty object directory, and an exception handler.

#### DELETE\$JOB

Deletes the specified job.

#### SET\$POOL\$MIN

Defines new minimum for the given job's memory pool.

#### GET\$POOL\$ATTRIB

Returns information regarding the given job's memory pool.

## OFFSPRING

Returns the children jobs of the specified job.

## CREATE\$TASK

Creates a task with a specified priority, stack, code location, and the job's exception handler.

## DELETE\$TASK

Deletes the specified task.

## SUSPEND\$TASK

Suspends the specified task or, if already suspended, increments suspension depth by one.

## RESUME\$TASK

Decrements the suspension depth by one, and if zero resumes the task.

## GET\$EXCEPTION\$HANDLER

Returns task's exception handler information.

## SET\$EXCEPTION\$HANDLER

Sets a task's exception handler to the specified location and mode. The mode determines which classes of errors will be sent to the exception handler.

## SLEEP

Places the caller task in the asleep state for a specified number of milliseconds.

## GET\$TASK\$TOKENS

Returns requested system information.

## GET\$PRIORITY

Returns the priority of the calling task.

## SET\$PRIORITY

Dynamically alters the specified task's priority.

## CREATE\$SEGMENT

Creates a segment with a specified length.

## DELETE\$SEGMENT

Deletes the specified segment.

## GET\$SIZE

Returns the segment size.

## CREATE\$MAILBOX

Creates a mailbox for objects to be sent and received.

## DELETE\$MAILBOX

Deletes the specified mailbox.

## SEND\$MESSAGE

Sends the specified object's access to the given mailbox.

## RECEIVE\$MESSAGE

Receives an object's access from the specified mailbox.

## CREATE\$SEMAPHORE

Creates an integer semaphore with an initial value.

## DELETE\$SEMAPHORE

Deletes the specified semaphore.

## RECEIVE\$UNITS

Receives the number of units requested from the semaphore.

## SEND\$UNITS

Sends to the given semaphore the specified number of units.

## CREATE\$REGION

Creates an access mechanism which controls code that is only to be accessed by one task at a time.

## DELETE\$REGION

Deletes the specified region.

## RECEIVE\$CONTROL

Gains access to the mutually exclusive code even if it involves waiting.

## ACCEPT\$CONTROL

Gains access to the mutually exclusive code only if it is immediately available.

## SEND\$CONTROL

Relinquishes the region when finished using it.

## CREATE\$EXTENSION

Creates the license rights to a specified extension type.

## DELETE\$EXTENSION

Deletes all objects of the extension type specified.

## CREATE\$COMPOSITE

Creates a new object template under the specified extension object type.

## DELETE\$COMPOSITE

Deletes the specified composite object.

## INSPECT\$COMPOSITE

Inspects the specified composite object's template.

## ALTER\$COMPOSITE

Changes the composite object's template.

## INTERRUPT MANAGEMENT SYSTEM CALLS

### SET\$INTERRUPT

Assigns an interrupt handler and, if desired, an interrupt task.

## RESET\$INTERRUPT

voids the SET\$INTERRUPT call.

## GET\$LEVEL

Returns interrupt level of executing interrupt handler.

## EXIT\$INTERRUPT

Interrupt handler's "END OF INTERRUPT" statement.

## SIGNAL\$INTERRUPT

Used within interrupt handler to invoke interrupt task.

## WAIT\$INTERRUPT

Puts calling interrupt task in suspension until reinvoked by the interrupt handler.

## ENABLE

Enables an external interrupt level.

## DISABLE

Disables an external interrupt level.

## BASIC I/O SUBSYSTEM CALLS

### CREATE\$FILE

Creates a file and establishes an I/O line or connection. If named, it also establishes access mode.

### ATTACH\$FILE

Establishes an I/O connection to an existing file.

### DELETE\$CONNECTION

Deletes an I/O connection to the specified file.

### GET\$CONNECTION\$STATUS

Returns I/O connection status.

### GET\$FILE\$STATUS

Returns file status.

### PHYSICAL\$ATTACH\$DEVICE

Binds file driver to device driver.

### OPEN

Opens a file for I/O. The file pointer is set to the beginning of the file.

### CLOSE

Undoes an OPEN.

### READ

Initiates an I/O read starting at the file pointer.

### WRITE

Initiates an I/O write starting at the file pointer.

### SEEK

Moves the file pointer.

## DELETE\$FILE

Deletes the file from the directory after all connections to the file have been deleted.

## RENAME\$FILE

Renames the file in the directory.

## CHANGE\$ACCESS

Changes access rights to a given file. User rights may be added or changed.

## TRUNCATE

Truncates the file at the present file pointer.

## GET\$PATH\$COMPONENT

Returns the hierarchical directory tree traversal from the root to the file.

## GET\$DIRECTORY\$ENTRY

Returns the file's directory name.

## CREATE\$DIRECTORY

Creates a directory file.

## Supported Hardware

### SINGLE BOARD COMPUTERS

#### iSBC 86/12A Single Board Computer

Complete computer system on a single 6.75 x 12-inch printed circuit board; includes an iAPX 86 (8086) CPU, system clock, 32K bytes of dual port RAM, sockets for 4K bytes (using Intel 2758), 8K bytes (using Intel 2716 or 2316E) or 16K bytes (using Intel 2732) of read only memory, multi-master MULTIBUS™ arbitration logic, 9-level programmable vectored interrupt control, 2 programmable BCD binary timers/counters, 24 programmable I/O lines with sockets for I/O line drivers and terminators, and a USART 8251A with associated RS232C drivers and receivers.

### MASS STORAGE CONTROLLERS

#### iSBC 204 Diskette Controller

Universal Flexible Diskette Controller; plugs into one MULTIBUS system slot. Controls most single density standard flexible diskette drives and mini-drives.

#### iSBC 206 Disk Controller

Disk Controller for 5440-type disk drives with 10M bytes per drive and up to four drives per controller; adapts to industry standard drives.

### USER iAPX 86 (8086)-BASED SYSTEMS

The iRMX 86 system runs on user designed boards with the following components:

- iAPX 86 (8086) — 16-Bit Microprocessor

# iRMX 86

---

- 8253 — Programmable Interval Timer (counter 0 mapped to I/O location 0D0H)
- 8259A — Programmable Interrupt Controller (base port mapped to I/O location 0C0H connecting INT 2 to counter 0 of the 8253)
- 8251A — USART (base port mapped to I/O location 0D8H when using the Terminal Handler)

## iRMX 86 Shipping Package

ISIS-II compatible diskettes containing:

- iRMX 86 Nucleus
- iRMX 86 Terminal Handler
- iRMX 86 Debugging Subsystem
- iRMX 86 Basic Input/Output Subsystem
- iSBC 204 Universal Flexible Diskette Device Driver
- iSBC 206 Hard Disk Device Driver
- Preconfigured iRMX 86 System

One man-week of training at any Intel Training Center offering iRMX 86 training. (Good for 6 months.)

Documentation Package including:

- Introduction to iRMX 86 Operating System (9803124-01)
- iRMX 86 Nucleus, Terminal Handler, and Debugger Reference Manual (9803122-01)
- iRMX 86 I/O System Reference Manual (9803123-01)
- iRMX 86 System Programmer's Reference Manual (142721-001)
- iRMX 86 Installation Guide for ISIS-II Users (9803125-01)
- iRMX 86 Configuration Guide for ISIS-II Users (9803126-01)

Registration Form

Software Problem Reports

One year of updates.

iSBC 957A Intellex-iSBC 86/12A Interface and Execution Package

---

## ORDERING INFORMATION

Part Number	Description
RMX 86	Operating System

## iRMX 88 REAL-TIME MULTITASKING EXECUTIVE

- Designed for iAPX 88, iAPX 86-based applications
- EPROM or RAM based
- Systems tailored through Interactive Configuration Utility
- Effective system debugger
- Performance-oriented nucleus
- Easy-to-use interfaces
- Real-time priority-oriented scheduling
- Disk file system software

The iRMX 88 Real-Time Multitasking Executive is a small, performance-oriented executive system which can be used on Intel's 16-bit single board computers, iAPX 88 and iAPX 86-based boards. Based on the iRMX 80 Executive's field-proven and reliable architecture, the iRMX 88 Executive supports upgrading 8-bit based applications and helps the first-time microcomputer customer with simple, easy-to-use interfaces for both the iSBC and component-based applications. The iRMX 88 Executive Software provides the innermost, software control layer for the CPU that supports real-time application requirements for inter-task communication, asynchronous I/O control, priority-based resource allocation, and standard iSBC disk controller interfaces.

The iRMX 88 Executive offers features that are suitable for reliable, performance-critical process control applications, production test stand units, sophisticated laboratory analysis, instrumentation, or specialized data acquisition and monitoring stations. Now, previous iRMX 80-based designs can upgrade to small size and high performance solution using the iRMX 88 Executive. The application will be easily tailored using the Interactive Configuration Utility.

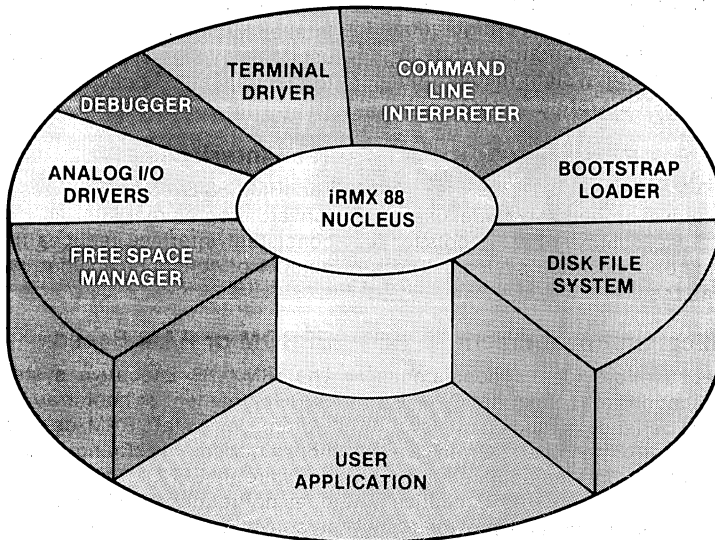


Diagram 1. Module Representation



## FUNCTIONAL DESCRIPTION

The iRMX 88 Executive provides users of Intel Single Board Computers and user-designed boards utilizing iAPX 86, iAPX 88 CPUs, with facilities for executing concurrently, managing resources, and servicing asynchronous events. The iRMX 88 Executive provides simple, easy-to-use tools for creating a range of reliable high quality application systems. The foundation modules include support for real-time facilities including priority-based task scheduling, provide the means to concurrently monitor and control multiple events, provide real-time clock control, manage interrupts, and dispatch tasks. By staying within the 64K byte code and 64K data segment model, the operating system can maintain performance and minimize memory requirements. However, user tasks can access the full megabyte of data memory. The iRMX 88 Executive, as shown in Diagram 1, contains the following modules: a Nucleus, Terminal Handler, Free Space Manager, Disk File System, Bootstrap Loader, Analog Handlers, Interactive System Debugger, and a Command Line Interpreter.

Because of the modularity and variety of services available with the iRMX 88 product, the user selects only those features needed for an effective application-specific operating system. This process of feature selection for a custom operating system is aided by a development tool, the Interactive Configuration Utility. After the user responds to a set of feature and configuration questions, the Interactive Configuration Utility integrates and configures the system readily from its library of quality, tested system features.

## FEATURE OVERVIEW

The iRMX 88 Executive products provide users with simple, easy-to-use, quality software tools for ceating a wide range of application systems. The most important features are:

### Structured Application Environment

The iRMX 88 Executive provides a consistent structure from application to application, CPU to CPU, thus allowing experience gained on one system to be easily transferred to others. Often, entire tasks may be ported from one application to another. As shown in Diagram 2, the Executive manages the hardware interface and provides a standard interface to the application. When the Executive supports future hardware, the Executive may have new control software, but the application layer interface remains the same.

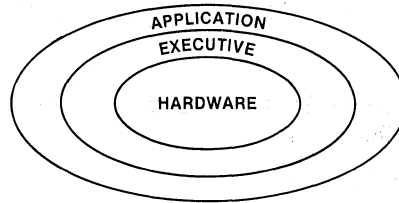


Diagram 2. Consistent Structure

### User Configurable

The iRMX 88-based applications can utilize a wide range of features or select only those which meet the specific requirements of the application system. The resultant system contains only the modules necessary for its use, allowing the iRMX 88 application system to be tailored to the application demands.

The iRMX 88 Executive is constructed in a thoroughly modular manner with the full range of facilities being offered in library modules, allowing easy selection of the exact features required. Overhead for unused features is then eliminated, minimizing memory requirements.

An interactive configuration development tool provides an easy-to-use method of configuring the iRMX 88-based application. Responding to questions from the utility program executing on the Intellec Microcomputer Development System, the user quickly tailors the operating system features to meet specific application needs.

### Task Interfaces

The iRMX 88 Executive utilizes a simple, straightforward programmer interface for user tasks. This consistent interface reduces the complexity and number of concepts which must be learned.

### EPROM or RAM Based

The iRMX 88 Executive system can be totally EPROM resident or bootstrap loaded from a mass storage device into RAM, depending upon application requirements. Being able to place all of the software in EPROM offers two benefits. First, it does not require a disk for bootstrap loading. This is convenient when mass storage devices cannot be used because of the danger of contamination. Second, the overhead expense of mass storage devices is eliminated when the application can be prommed.

## Nucleus

The iRMX 88 Executive provides control software for operation on various ISBC 16-bit Single Board Computers, and can be used for user iAPX 86, iAPX 88-based boards. The iRMX 88 Nucleus provides a foundation upon which a variety of application systems can be built.

The iRMX 88 Nucleus is the heart of an iRMX 88-based operating system. All other subsystems require the Nucleus.

Embedded in the Nucleus are two main facilities: first, the facility for concurrent program execution; second, the facility for handling simultaneous asynchronous events. The typical iRMX 88-based application solution requires real-time response. For example, interrupts (samples per second) coming from specialized peripheral devices must be serviced in an efficient manner. Since the concepts are portable from CPU to CPU, iRMX 88-based systems allow the hardware and software to be used by more than one application, thus reducing the overall system and hardware costs. Designed toward performance-oriented solutions, the multitasking architecture provides the power necessary to solve the complex problems of today and tomorrow. The system comprises uniform system calls ensuring that the Nucleus is both easy to learn and easy to extend. Only two concepts must be mastered to learn the system: the user-task requests to the Nucleus for service; and the definition of the system configuration. These concepts require the understanding of the terms: task, messages, and exchanges. These three items can be created and deleted dynamically.

The Nucleus provides two levels of interrupt management, an interrupt handler and an interrupt task. One provides speed for high performance samples per second requirements and the other provides flexibility. The interrupt handler permits only Nucleus interrupt system calls to be issued and masks all interrupts. The interrupt task permits all Nucleus system calls to be issued and masks only lower priority interrupts; this is very effective for interrupt management at a lower frequency.

## Priority Levels

Handling an interrupt with respect to other work in the system can be completely user controlled. Nucleus external interrupt levels are directly related to task priorities. The priority of the executing task determines which interrupts are masked. Generally, software background tasks will be as-

signed a lower priority than interrupt task priorities, to avoid blocking interrupts.

**Table 1. Nucleus Primitives**

Name	Operands	Return Value	Function
RQACPT	exch-addr	addr	Accept a message from specified exchange. Returns message address if available, zero otherwise.
RQCTSK	STD-addr	none	Create task by building new Task Descriptor based on specified Static Task Descriptor.
RQCXCH	RAM-addr	none	Create exchange at specified RAM address.
RQDLVL	level	none	Disable specified interrupt level.
RQDTSK	TD-addr	none	Delete task specified by Task Descriptor.
RQDXCH	exch-addr	byte	Delete specified exchange.
RQELVL	level	none	Initialize message portion of the Interrupt Exchange Descriptor associated with the specified interrupt level (the first time called only), and enable specified interrupt level.
RQENDI	none	none	Signals end-of-interrupt in user-supplied interrupt service routine.
RQISND	IED-addr	none	Send an interrupt message to the specified interrupt exchange.
RQRESM	TD-addr	none	Resume a task that has previously been suspended.
RQSEND	exch-addr, msg-addr	none	Send the message located at "msg-addr" to the exchange specified by "exch-addr."
RQSETV	addr, level	none	Set interrupt vector. Interrupts at the specified level are serviced by the user-supplied routine starting at "addr," thus bypassing Nucleus interrupt software.
RQSUSP	TD-addr	none	Suspend execution of the task specified by the Task Descriptor.
RQWAIT	exch-addr, time-limit	addr	Wait at the specified exchange until a message is available or time limit expires. Return address of system time-out message or user message.

## Free Space Manager

The iRMX 88 Free Space Manager provides the capability of dynamically allocating RAM space based upon user requests. The Free Space Manager allows efficient use of RAM in the iRMX 88-based system by reclaiming RAM space that is no longer needed by a task, thereby making it available for use by other tasks. This allows RAM space to be treated as a common resource, shared dynamically by a number of tasks.

## System Debugger

The iRMX 88 product provides an interactive software debugger. The System Debugger has two capabilities that greatly simplify the process of debugging a multitasking system. First, the Debugger allows examination of the task while the system continues to operate. Second, the Debugger lets the programmer interactively view and modify system constructs and RAM data space.

## Comprehensive I/O Support

The iRMX 88 software libraries contain support for a wide range of iSBC I/O boards manufactured by Intel. This support simplifies the addition of peripherals to MULTIBUS-compatible application systems. For applications which require custom boards the iRMX 88 architecture allows easy addition of user-written device handlers.

The iRMX 88 Disk File System (DFS) provides disk access capabilities to iRMX 88 operating systems. The term "disk" will refer generally to both hard disk and floppy disk products. The services operate in the real-time environment supported by the Nucleus. Files may be created, deleted or changed. Data may be accessed sequentially and directly ("randomly").

Many applications do not need all the services which DFS offers. The modular design of the system allows DFS functions to be implemented selectively, thereby keeping memory requirements consistent with the needs of the application.

DFS performs most of the activities required to manage real-time disk operations. (See Table 1.) The user does not have to be concerned with scheduling and coordinating the multiple and sometimes conflicting requests that typify real-time operation. By taking responsibility for these activities, and for the maintenance of system integrity, DFS encourages the user to concentrate on solving the application problem at hand.

In addition to presenting a complex control situation, real-time applications often have demanding performance requirements. DFS helps maximize system throughput by allowing user tasks to overlap disk operations with processor operations and by allowing multiple disk operations to proceed in parallel. For example, files can be "double buffered" so that while DFS is filling (or emptying) one buffer, the user task can be processing the data in the other, reducing the time the task spends waiting for I/O.

Table 2. Disk File System Services

Service	Function
<b>Data Transfer Services</b>	
OPEN	Prepare a file for processing.
READ	Transfer data from an open file to memory.
WRITE	Transfer data from memory to an open file.
SEEK	Set or return value of disk file marker.
CLOSE	Terminate processing of an open file.
<b>Directory Maintenance Services</b>	
DELETE	Remove a file from the directory and release its space.
RENAME	Change the name of a file in the directory.
ATTRIB	Change an attribute of a file in the directory.
<b>Other Services</b>	
FORMAT	Initialize a new disk.
LOAD	Read a file of executable code into memory.
DISKIO	Perform fundamental disk sector operations.

## Bootstrap Loader

The iRMX 88 Bootstrap Loader is an extension to the operating system that adds considerable flexibility to iSBC disk controller-based systems. The 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 EPROM resident. This offers greater flexibility in building and maintaining an application that is disk-based.

- Since application software is stored on disk rather than in ROM, changes are easier to implement. A modification to application software entails changing the contents of the disk rather than replacing the ROM.
- Because the disk has a much larger storage capacity than does the ROM, multiple tasks can be stored on disk with the Bootstrap Loader bringing them into RAM on demand. This feature allows a single iSBC-based solution to be used for more than one application.

## Terminal Handler

The iRMX 88 product includes a Terminal Handler module that provides real-time asynchronous I/O between an operator's terminal and tasks executing under the iRMX 88 Nucleus. User tasks can utilize input data obtained from the Terminal Handler. Output may be at the level of a single character, a logical line, or a set of lines.

## Command Line Interpreter

The iRMX 88 Command Line Interpreter (CLI) is a task that provides a powerful man-machine interface for interactive control of system resources and utilities. The CLI can be readily configured with the iRMX 88 Terminal Handler. Examining an input message, the CLI will pass the message to the appropriate utility task. The utility tasks provided with the system include FORMAT, DIR(ec-tory), and COPY.

In addition, a greater degree of flexibility for tailored application is available; the user can add application-oriented commands to the vocabulary

for the CLI, thus satisfying a demand for unique input command requests.

## Analog Handlers

The iRMX 88 Executive offers Analog Handlers which provide a convenient mechanism for performing A/D and D/A conversions, utilizing Intel's iSBC Analog Boards. The Analog Handlers perform repetitive, sequential, single or variable gain, random channel input and random channel output. The input and output modules are individually configurable, allowing greater application flexibility for minimum size systems.

### iSBC iAPX 88 and iAPX 86 Memory Requirements

Memory Requirements** (Bytes)								Minimum Development System Requirements		
Module	Nucleus	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*	3.5K	1.2K	2K	11K	1.4K	1.6K	1.2K	ISIS-II	64K RAM	2
RAM	270	140	500							

\*Indicates amount of code which can be configured in PROM.

\*\*All figures are approximate for education purposes only.

## SPECIFICATIONS

### iSBC Supported Hardware

#### Single Board Computers

iSBC 88/40  
iSBC 86/05  
iSBC 86/12A

#### Mass Storage

iSBC 208 flexible disk controller  
iSBC 215A/215B 8-inch Winchester disk controller  
iSBC 220 SMD disk controller

#### Analog Boards

iSBX 311  
iSBX 328  
iSBC 711/711A

iSBC 724/724A

iSBC 732/732A

#### MULTIMODULE Boards

iSBX 337 numerical data processor  
iSBX 351 serial I/O

### User iAPX 86 and iAPX 88- Based Systems

The iRMX 88 system runs on user-designed boards with the following components:

- 8253—Programmable Interval Timer (counter 0)
- 8259A—Programmable Interrupt Controller (INT1 to counter 0 of 8253)
- 8251A—USART or iSBX 351 when using Terminal Handler

**ORDERING INFORMATION**

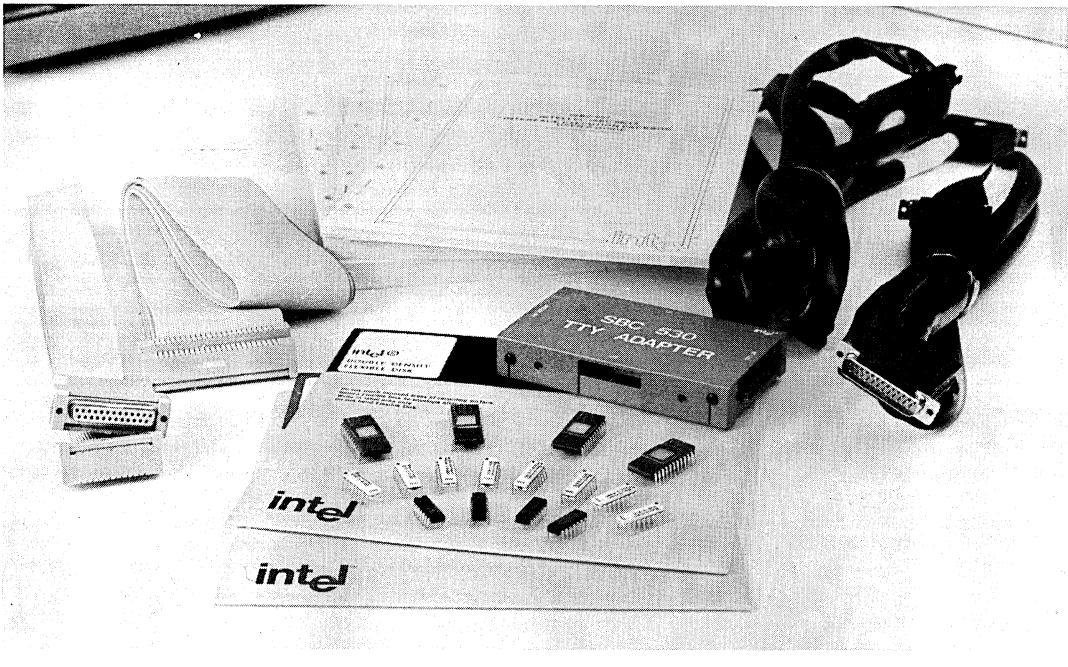
Part Number	Description	Availability	Details
iRMX 88	NUCLEUS, Terminal Handler, Free Space Manager for iAPX 88 (8088), iAPX 86 (8086) support including iSBC 86/12A, iSBC 88/40. Package also includes Interactive Configuration Utility program for execution on an Intellec Microcomputer Development System.	Not Available	The modules which include the debugger, CLI, and utility software modules COPY, DIR, and FORMAT. Analog Handler modules for support of iSBC 711/711A, iSBC 724/724A, iSBC 732/732A, iSBX 311, iSBX 328.
Not Available	Disk File System, modules for iSBC 208, iSBC 215A/215B, iSBC 220 disk controllers, and supports user-defined controllers.	iRMX 88 DOC	Documentation Package including <ul style="list-style-type: none"> <li>• Introduction to the iRMX 88 Executive</li> <li>• iRMX 88 Systems Reference Manual</li> <li>• iRMX 88 Installation Guide</li> <li>• iRMX 88 Configuration Utility Guide</li> <li>• iRMX 88 Application Guide</li> </ul>
Not Available		Not Available	
Not Available		Not Available	



## iSBC 957A INTELLEC® -iSBC 86/12A INTERFACE AND EXECUTION PACKAGE

- Establishes communication between the iSBC 86/12A board and the Intellec® Development Systems to aid in iAPX 86/8086 software development
- Allows full speed execution of iAPX 86/8086 programs
- Includes EPROM resident system monitor for iSBC 86/12A Single Board Computer
- Allows Intellec® ISIS-II files to be transferred between iSBC 86/12A board and Intellec® Microcomputer Development System
- Offers "virtual terminal" capability which permits the Intellec® console to access the iSBC 86/12A monitor
- Provides powerful console commands for software debug
- Allows access to all iSBC 86/12A memory, registers, flags and I/O ports
- Includes all necessary hardware, software and documentation

The Intel® iSBC 957A Intellec® -iSBC 86/12A Interface and Execution Package contains all the necessary hardware, software, cables and documentation required to interface an iSBC 86/12A Single Board Computer to an Intellec Microcomputer Development System for software development and full-speed execution.



## FUNCTIONAL DESCRIPTION

### Overview

The iSBC 957A Inteltec-iSBC 86/12A Interface and Execution Package extends the software development capabilities of the Inteltec Microcomputer Development Systems to the iSBC 86/12 and iSBC 86/12A Single Board Computers. It allows software modules developed under the Inteltec resident ISIS-II Operating Systems to be down-loaded to the iSBC 86/12A board for full-speed execution and debug. In addition, the iSBC 957A package allows segments of iSBC 86/12A memory to be saved on floppy disk files. Special communication software allows transparent access to the powerful debug commands in the iSBC 86/12A monitor from the Inteltec console terminal.

### Software Capabilities

The software included in the iSBC 957A package consists of the iSBC 86/12A monitor residing on four Intel EPROMs which are inserted into sockets on the iSBC 86/12A board. A diskette is also included which contains the Inteltec resident communications software that links the iSBC 86/12A board with the Inteltec Microcomputer Development System. The EPROM resident software creates an execution environment in which object modules may be loaded into the iSBC 86/12A memory, executed at full speed, modified if necessary and saved on the Inteltec system floppy disk. The monitor provides the ability to execute selected program segments with breakpoints or by single stepping, examine and modify

registers and memory, perform port I/O, move a block of memory, compare blocks of memory, search for a word/byte value, and perform hex arithmetic. In addition, the monitor provides for the recognition of interrupts via a user-defined table. The program on the diskette contains communication software which passes appropriate console commands to the iSBC 86/12A resident monitor and also interfaces with the ISIS-II operating system to transfer files between the development system diskettes and the iSBC 86/12A board.

### System Interfacing

The physical interface between the Inteltec Microcomputer Development System and the iSBC 86/12A board is accomplished with cables supplied with the iSBC 957A package. The cabling arrangement varies depending on whether the system is a member of the Inteltec MDS-800 family or one of the Inteltec Series II family.

**Inteltec Series II Interface** — For Inteltec Series II Development Systems the connection between it and the iSBC 86/12A board is accomplished with either a single serial line interconnecting the iSBC 86/12A serial port with serial port 1 on the Inteltec system, or a parallel cable from the UPP port to the parallel port on the iSBC 86/12A board. All communication including command and data transfer occurs over this line.

### Inteltec® Environment

An Inteltec Microcomputer Development System to be used in conjunction with the iSBC 957A package and an iSBC 86/12A board must have the following necessary

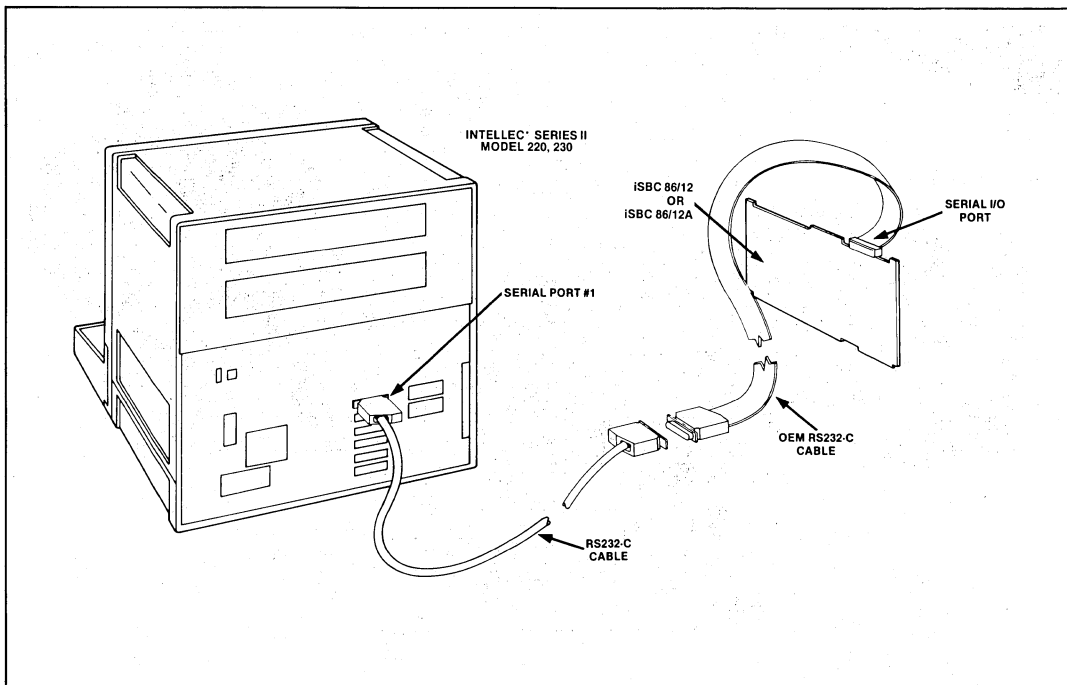


Figure 1. Inteltec® Series Models 220, 230 Serial Connection

## iSBC 957A

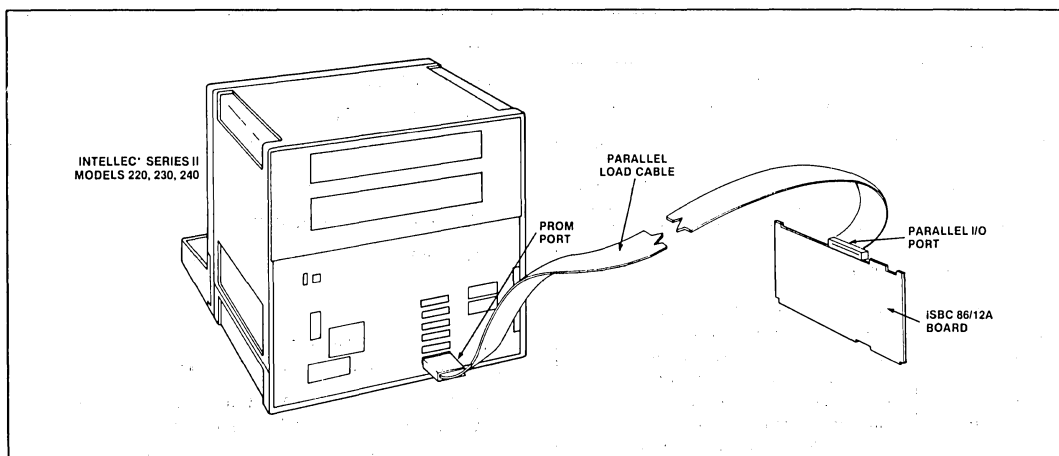


Figure 2. Intellec® Series Models 220, 230, 240 Parallel Connection

functionality to support program development and storage:

1. Intellec Development System with 64K bytes of RAM.
2. Console CRT or TTY terminal.
3. Intellec MDS-DDS Dual Double Density Diskette Drive and ISIS-II Operating System or Intellec MDS-2DS Dual Single Density Diskette Drive and ISIS-II Operating System.
4. User-selected language translators.

**Note:** The Intellec Series II Model 230 Microcomputer Development System and the Intellec MDS-800 Microcomputer Development Center contain all necessary hardware and operating system software to be used with the iSBC 957A package and the iSBC 86/12A board.

### Execution Environment

A full capability iSBC 86/12A execution environment

should include the following components for effective utilization:

1. An iSBC 86/12A Single Board Computer.
2. An iSBC 957A Intellec-iSBC 86/12A Interface and Execution Package.
3. An iSBC 655 or iSBC 660 System Chassis for power and MULTIBUS expansion.
4. One or more iSBC 032, 048, or 064 RAM boards for programs requiring more than 32K bytes of RAM.

**Note:** The iSBC 86/12A cannot be mounted in the Intellec system and requires a separate operating environment.

Additional memory boards, analog and digital I/O boards, and peripheral controllers can be included in the iSBC 660 System Chassis with the iSBC 86/12A to allow the execution environment to be equivalent to the expected final product configuration.

## SPECIFICATIONS

### Hardware

#### Cables

- (1) OEM RS232-C cable — Mates with serial I/O port on iSBC 86/12A
- (1) RS232-C port cable — Mates with RS232-C port on Intellec system
- (1) TTY port cable — Mates with TTY port on Intellec system
- (1) Parallel load cable — Mates with UPP port on Intellec system and parallel I/O port on iSBC 86/12A

All cables allow separation of Intellec system and iSBC 86/12A of up to 6 feet.

#### I/O Drivers and Terminators

- (1) 7437 48 mA open collector drivers
- (4) iSBC 901 220Ω/330Ω terminator packs
- (4) iSBC 902 1 kΩ terminator packs

Drivers and terminators needed when parallel load cable is required

#### Interface Adapters

- (1) iSBC 530 TTY adapter — Used when serial I/O line connects with TTY port on Intellec system
- (1) Parallel port status adapter — Mounts on iSBC 86/12A when parallel load cable is required

**Miscellaneous** — Attachment screws for Intellec mounted connectors

#### Software

- (4) EPROMs with iSBC 86/12A system monitor
- (1) Single density floppy diskette with iSBC 86/12A ISIS-II communication software
- (1) Double density floppy diskette with iSBC 86/12A ISIS-II communication software

#### System Monitor

Addresses: RAM: 00000-006FF<sub>H</sub>; ROM: FE000-FFFF<sub>H</sub>



# ISBC 957A

## Commands

Command	Function
L Load	Loads absolute object file from Intellec into ISBC 86/12A memory.
G Go	Transfers control of the 8086 CPU to the user program.
R Load and Go	Loads absolute object file from Intellec into ISBC 86/12A memory and begins execution.
T Upload	Loads a block of ISBC 86/12A memory into an Intellec file.
N Single Step	Displays and executes one instruction at a time.
X Examine	Displays or modifies 8086 registers.
D Display	Displays contents of a memory block.
S Substitute	Displays/modifies memory locations.
M Move	Moves the contents of a memory block.
F Find	Searches a memory block or a constant.
C Compare	Compares two memory blocks.
I Input	Inputs and displays data from input port.
O Output	Outputs data to output port.
P Print	Prints values of literals.
E Exit	Exits the loader program and returns to ISIS-II.
* Comment	Rest of line is a comment.

## Transfer Rates

Intellec MDS-800 Family

*Serial transfer:* 37K bytes/minute

*Parallel transfer:* 96K bytes/minute

Intellec Series II Family

*Serial transfer:* 37K bytes/minute

*Parallel transfer:* 29K bytes/minute

## Reference Manuals

iSBC 86/12A Hardware Reference Manual

**142849-01** — iSBC 957A Intellec-iSBC 86/12 Interface and Execution Package User's Guide

**9800640** — 8086 Assembly Language Manual

## ORDERING INFORMATION

Part Number	Description
SBC 957A	Intellec-iSBC 86/12A Interfacing and Execution Package.

---

*Peripheral  
Controllers*

THE UNIVERSITY OF CHICAGO  
DEPARTMENT OF CHEMISTRY  
5800 S. UNIVERSITY AVENUE  
CHICAGO, ILLINOIS 60637  
TEL: 773-936-3700  
FAX: 773-936-3701  
WWW: WWW.CHEM.UCHICAGO.EDU

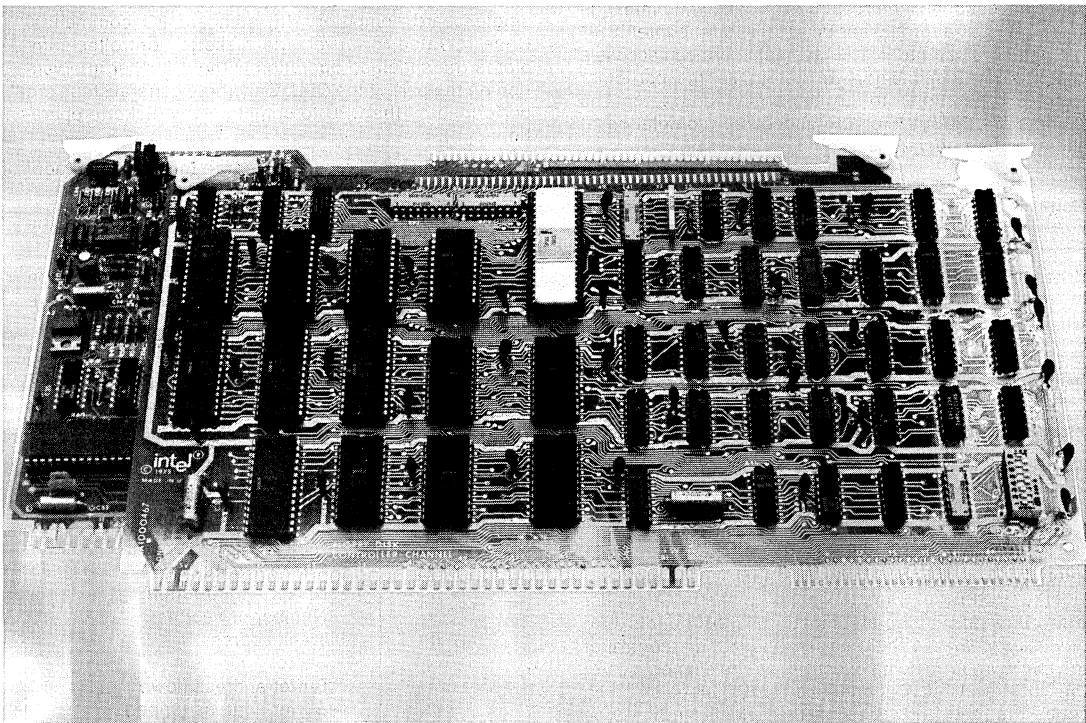
[The remainder of the page is extremely faint and illegible.]



## iSBC 202 DOUBLE DENSITY DISKETTE CONTROLLER

- iSBC 80, iSBC 86 and iSBC 88 compatible interface, control and DMA logic for high speed, high capacity random access bulk storage
- Provides control of up to four single sided flexible diskette drives
- Soft-sectored format allows 500K-byte data storage capacity per double density diskette
- DMA channel allows single board computer to process in parallel with block transfer between diskettes and memory
- Complete CRC data checking
- Compatible with Shugart SA 800-1 and other similar double density diskette drives

The iSBC 202 Double Density Diskette Controller is an interface between the MULTIBUS system bus and double density flexible diskette drives via a direct bus interface. Designed with Intel's powerful 3000 Series of microprogrammable bit-slice microprocessors, the iSBC 202 provides a high speed, efficient, and easy to use high capacity random access bulk storage interface to Intel's family of single board computers. Software support for the iSBC 80 Series of 8-bit single board computers is provided through RMX/80, the Real-Time Multitasking Executive.



## FUNCTIONAL DESCRIPTION

The iSBC 202 Double Density Diskette Controller provides an easy to use interface for OEM use of Intel single board computers and other manufacturers' flexible diskette drives. All DMA logic is provided so no additional boards or circuitry are required, and up to four double density flexible diskette drives may be interfaced with each iSBC 202. The controller facilitates recording of all data in soft-sector format. The controller consists of two boards which may reside in System Chassis, the iSBC 604 or 614 Modular Cardcage/Backplane, or in an OEM custom designed iSBC 80 bus-compatible with a majority of double density specified flexible diskette drives. The microprogrammed track format consists of 52 sectors with 128 bytes per sector. The Shugart SA 800-1 drive is fully compatible with this dense track format due to its "straddle-erase" magnetic head. Use of other manufacturers' flexible disk drives is accommodated also, with the limitation that after any "write data" operation, the CPU must delay 500  $\mu$ s before issuing another read or write command (due to the "delayed-erase" magnetic head). Therefore, use of multi-sector "write data" commands is only possible with the SA 800-1 drive.

### Channel Board

**Channel Board Function** — The channel board is the primary control module within the diskette controller. It receives, decodes, and responds to channel commands from the central processor unit (CPU) on the Intel iSBC single board computer. The channel board can access a block of system memory to determine the particular diskette operations to be performed and fetch the parameters required for the successful completion of the specified operation.

**Control Function** — 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 3604A 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

**Interface Board Function** — The interface board provides the iSBC 202 Double Density Diskette Controller with a means of communicating with the diskette drives, as well as with the 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 diskette platter), cause the head to move to the proper track and verify successful operation. The interface board accepts the data being read off the diskette, 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. The diskette controller is capable of performing seven different operations: recalibrate, seek, format track, write data, write deleted data, read data, and verify CDC.

**Write Operations** — 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.

**Memory Operations** — When the diskette controller requires access to the system memory, the interface board requests and maintains DMA transfer control of the system bus, and generates the appropriate memory command. The interface board also acknowledges I/O commands as required by the MULTIBUS system bus.

### Programming Capability

**IOPB Function** — The I/O Parameter Block (IOPB) has been designed to allow simplified I/O programming where necessary, and to facilitate a high level of sophistication in system-level I/O drivers in more complex systems. All diskette operations are initiated by an Intel single board computer with standard I/O commands. Once initiated, however, the diskette controller completes the specified operation without further intervention on the part of the CPU. Only three general steps are performed by the CPU to complete any diskette operation:

1. The CPU must prepare and store in system memory an IOPB for each operation to be performed.
2. The CPU then passes the memory address of the IOPB to the diskette controller.
3. The CPU must process the resultant information from the diskette controller upon completion of the operation.

**IOPB Format** — In preparing the IOPB, the CPU requires no interaction with the diskette controller. The IOPB is prepared as any block of data in memory would be prepared, utilizing the following format for the 7-byte parameter block:

Byte	Command
1	Channel word
2	Diskette instruction
3	Number of sectors
4	Track address
5	Sector address
6	Buffer address (lower)
7	Buffer address (upper)

## SPECIFICATIONS

### Media

Flexible diskette  
One recording surface  
77 tracks/diskette  
52 sectors/track  
128 bytes/sector  
512,512 bytes/diskette

### Physical Characteristics

**Mounting** — Occupies two slots of System 80 chassis or iSBC 604/614 cardcage, uses P2 connectors for inter-board communication.

**Height (ea)** — 6.75 in. (17.15 mm)

**Width (ea)** — 12.00 in. (30.48 mm)

**Depth (ea)** — 0.50 in. (1.27 mm)

**Weight (ea)** —

### Electrical Characteristics

#### DC Power Requirements

Channel Board: 5V @ 3.75A typ, 5A max

Interface Board: 5V @ 1.5A typ, 2.5A max; -5V @ 0.1A typ, 0.2A max

### Environmental Characteristics

#### Temperature

Operating: 0°C to 55°C

Non-Operating: -55°C to +85°C

#### Humidity

Operating: Up to 90% relative humidity without condensation

Non-Operating: All conditions without condensation of water or frost

### Equipment Supplied

DDFDC channel board

DDFDC interface board

Dual auxiliary board connector (P2 socket)

### Reference Manuals

**9800420** — iSBC 202 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
-------------	-------------

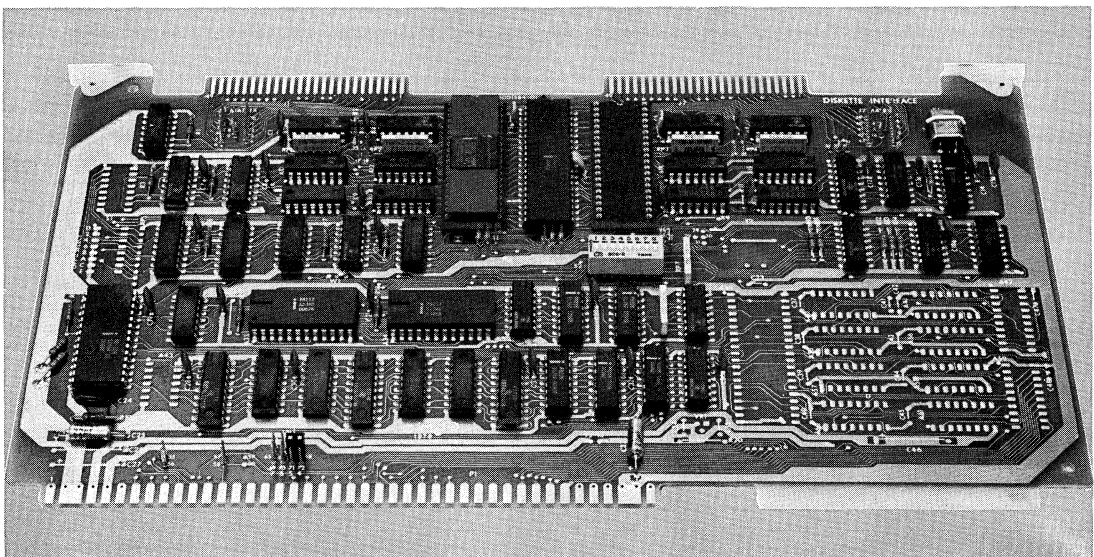
SBC 202	Double Density Diskette Controller
---------	------------------------------------



## 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 (5 1/4 ") flexible diskette drives
- Software supported by RMX/80 and RMX/86 Real-Time Multitasking 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

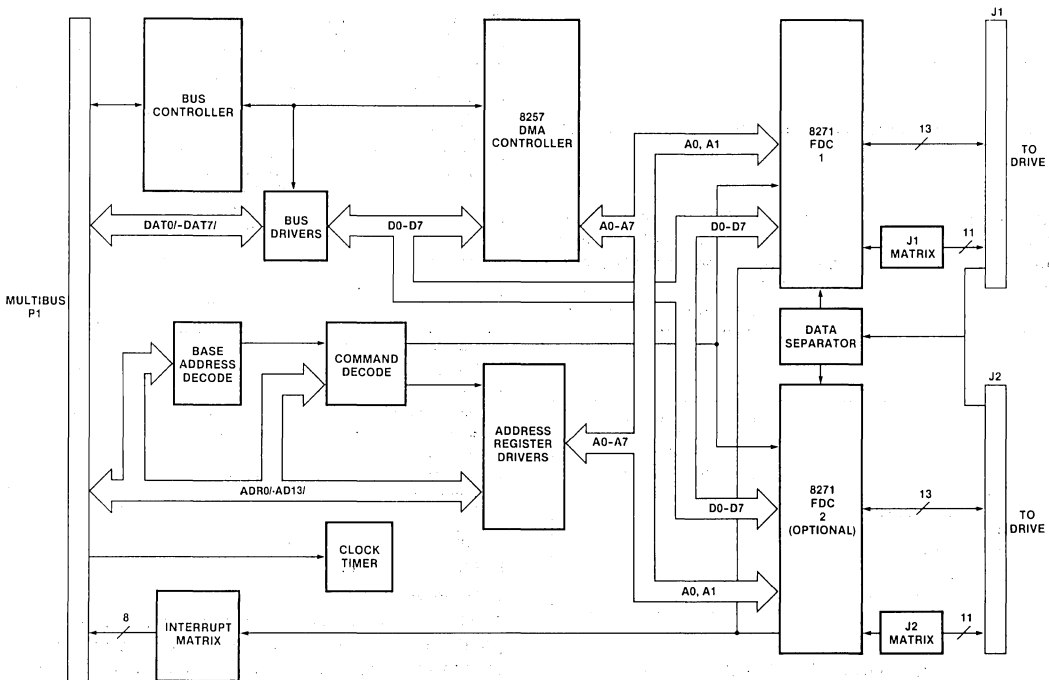


Figure 1. iSBC 204 Single Density Diskette Controller Block Diagram



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.

**RMX/80 and RMX/86 Real-Time Executive Software Support**

The iSBC 204 Controller is supported by the disk file system of RMX/80 and RMX/86, Intel's real-time operat-

ing systems. Files are named symbolically and may be created, deleted, or changed.

Data may be accessed sequentially or directly. An unlimited number of files can be open simultaneously and multiple tasks may read the same file concurrently. A listing of service functions included under RMX/80 and RMX/86 is shown in Table 1.

**Table 1. Services Available Under RMX/80 and RMX/86**

Function	Operation
Open	Prepares a file for processing.
Read	Transfers data from open file to memory.
Write	Transfers data from memory to open file.
Seek	Enables direct access.
Close	Terminates processing of file.
Delete	Removes file from directory, releases its space.
Rename	Changes name of file.
Format	Initializes a new disk.
Load	Reads a file of executable code into memory.
Disk I/O	Performs basic I/O operations.

**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		
	128	256	512	1024	2048	4096
Bytes per sector						
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 ms 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.

## iSBC 204

---

### 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 ( $\pm 5\%$ ), 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

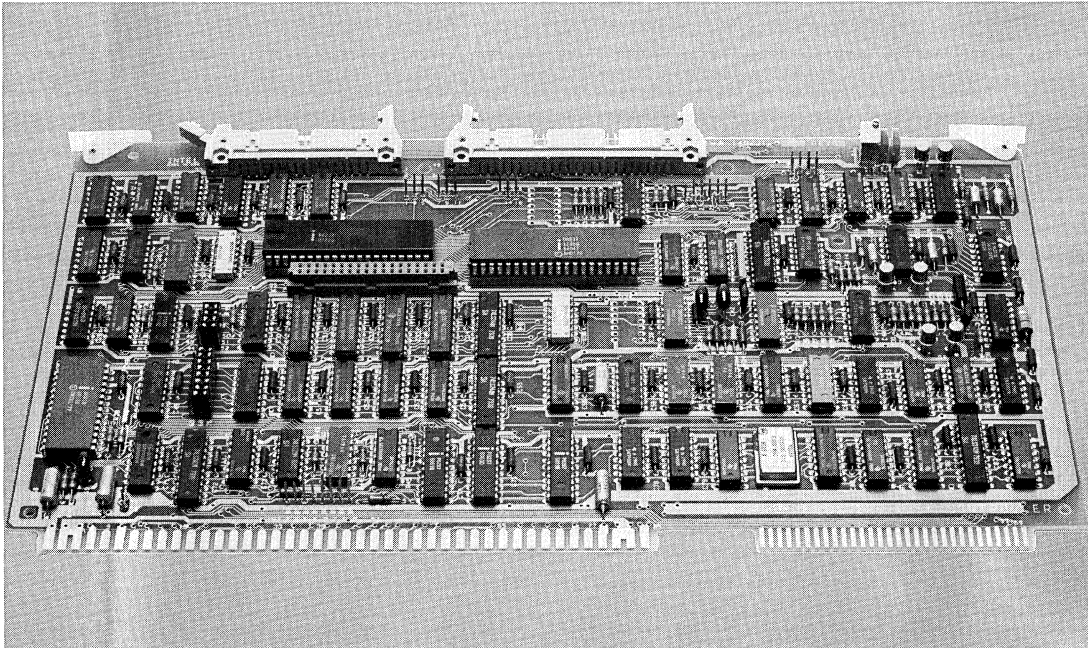
SBC 204	Universal Flexible Diskette Controller
---------	--



## 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, write, verify, and search 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.



# iSBC 208

## 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.

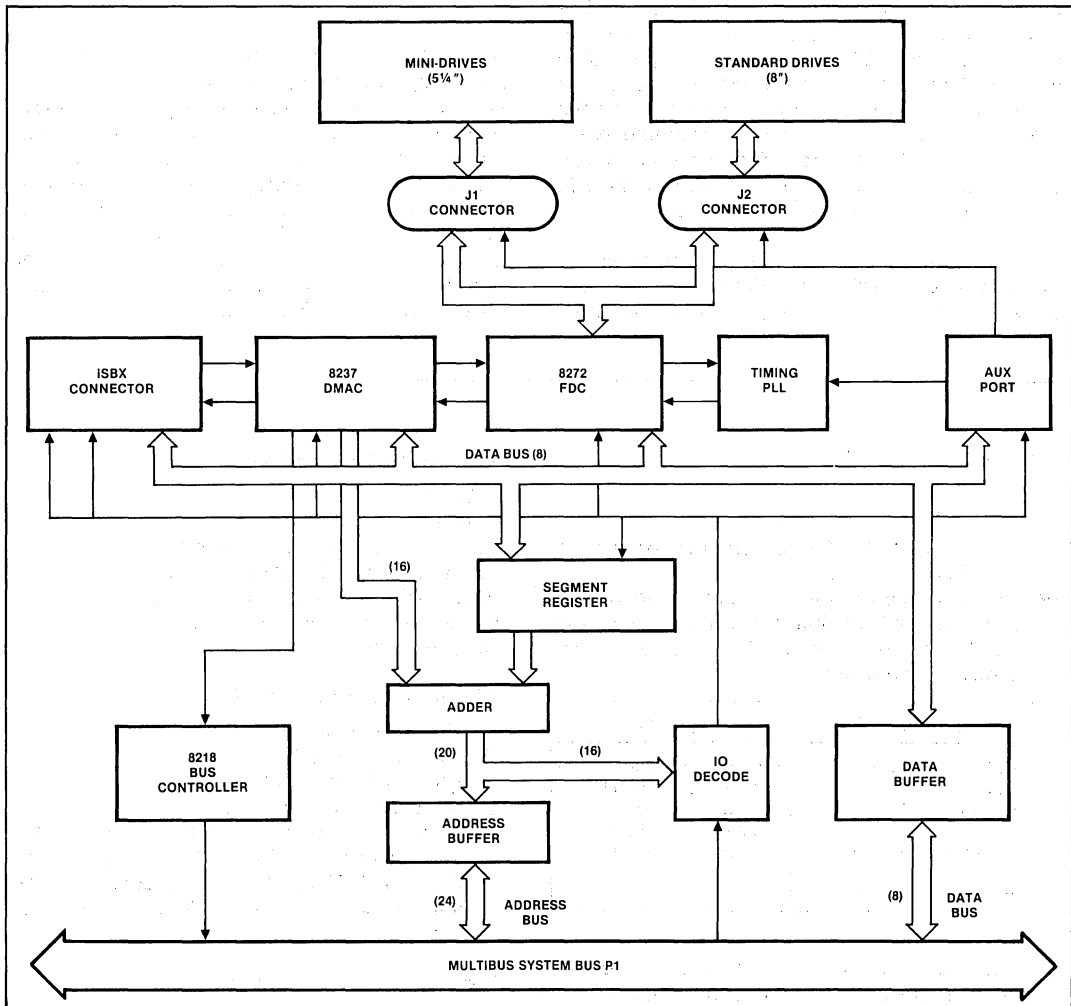


Figure 1. iSBC 208 Flexible Disk Controller Block Diagram

**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, write, and verify 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 SA	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		
Pertec	FD 650		

**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

# ISBC 208

## 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			77			77			77		
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 WINCHESTER DISK CONTROLLER

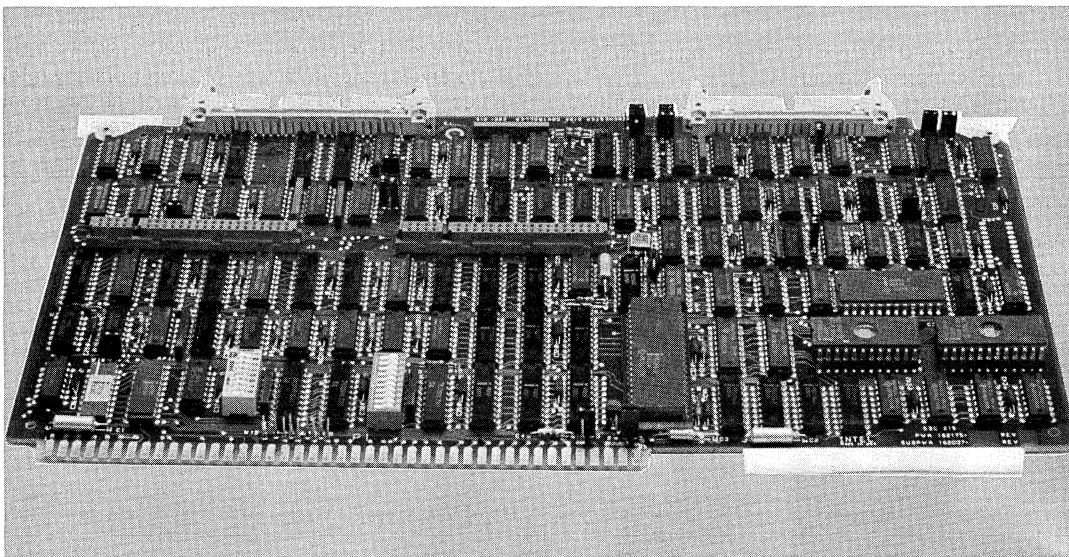
- Controls up to four 8" or 14" Winchester disk drives
- Over 100 MB of storage per controller
- Two iSBX connectors on-board
- Removable back-up storage available through the iSBX 218 Flexible Disk Controller
- Will provide proposed X3T9.3 ANSI standard interface
- Intel 8089 I/O Processor provides DMA channels plus user-programmable intelligence
- On-board diagnostics and ECC
- Full sector buffering on-board
- iSBC 215A board controls open-loop drives; iSBC 215B board controls closed-loop drives
- 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 8" 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 8" or 14" drives and is designed to conform to the proposed X3T9.3 ANSI standard (currently in the final definition phase).

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**

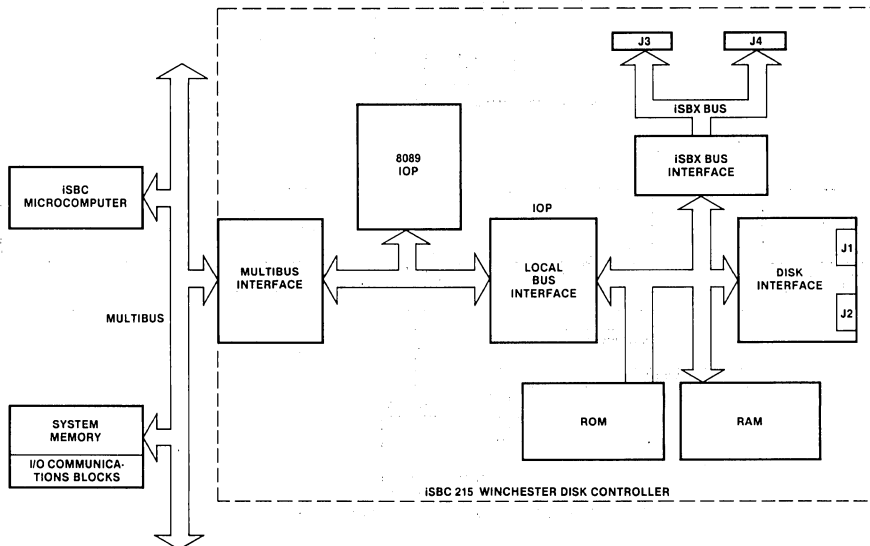
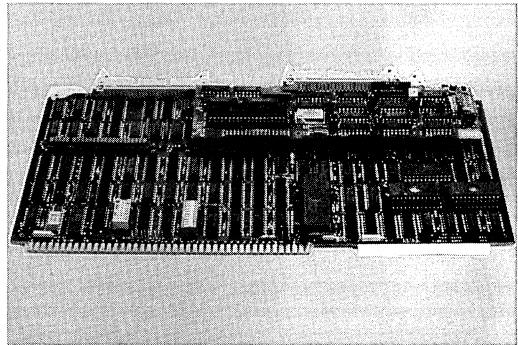
High data integrity is provided by on-board Error Checking Code (ECC) logic. When writing sector ID or data fields, a 32-bit Fire code, for burst error correction, is appended to the field by the controller. During a Read operation, the same logic regenerates the ECC polynomial and compares this second 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.

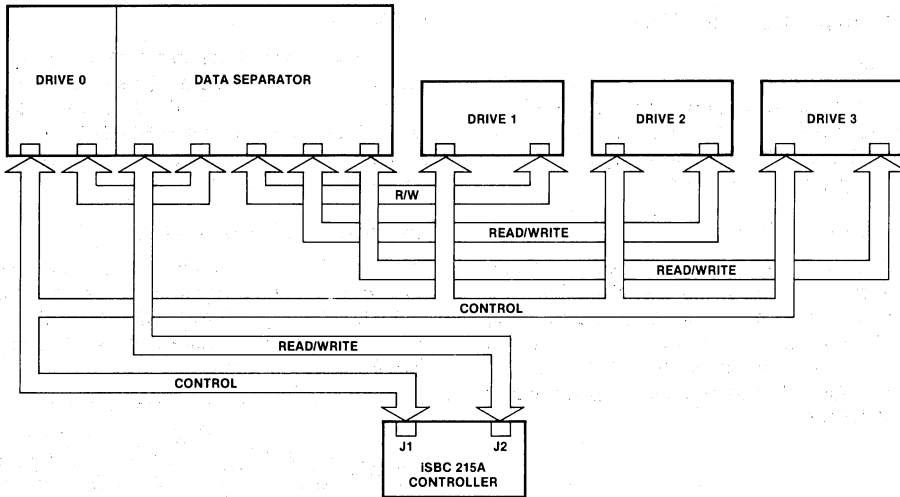
**Expanded I/O Capability**

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.

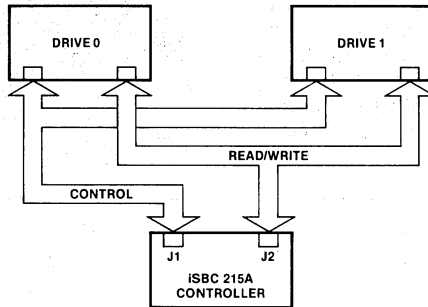


**Block Diagram of iSBC 215 Winchester Disk Controller**

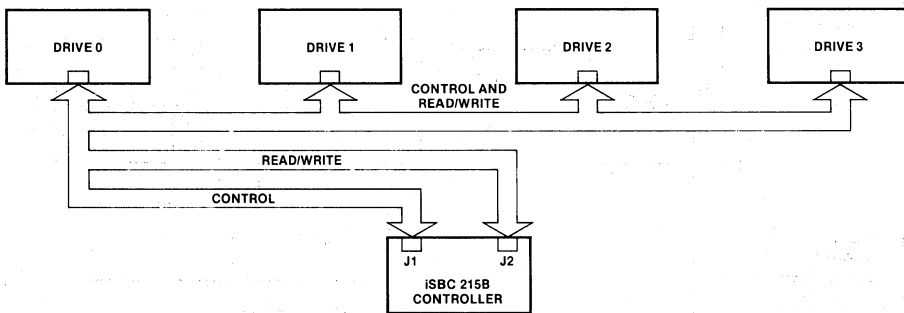




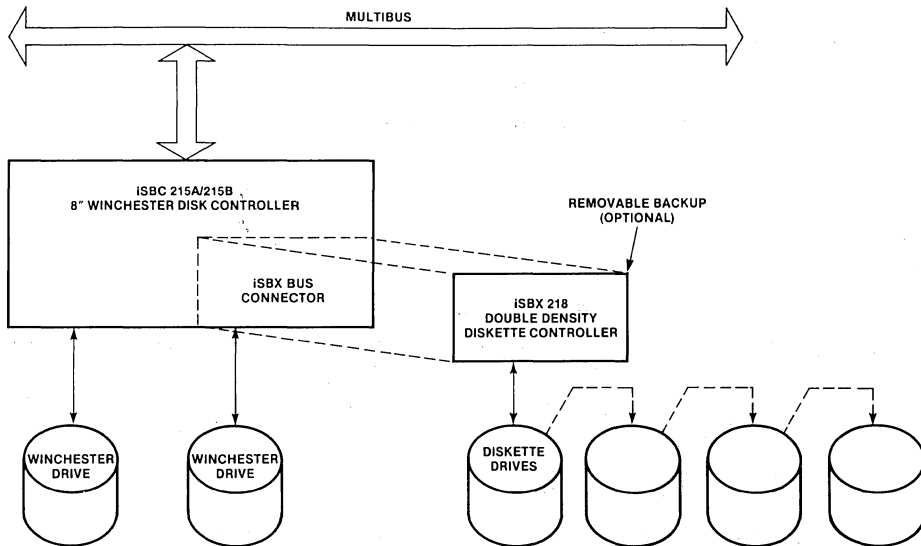
INTERFACE WITH SHUGART/QUANTUM DRIVES



INTERFACE WITH MEMOREX/SHUGART DRIVES



INTERFACE WITH PERTEC AND PRIAM DRIVES



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:

<b>Open-Loop (iSBC 215A)</b>
Shugart SA 1000 Series Shugart SA 4000 Series Memorex 100 Series Quantum Q2000 Series
<b>Closed-Loop (iSBC 215B)</b>
Pertec D8000 Series Priam 8" and 14" Drive Series
<b>iSBX MULTIMODULE Boards</b>
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<sup>1</sup>
- + 12 VDC @ 0.15A max<sup>2</sup>
- 12 VDC @ 0.03A max<sup>2</sup>

**Notes:**

1. On-board regulator and jumper allows - 12 VDC usage from MULTIBUS.
2. Required for some iSBX MULTIMODULE boards.

**Data Organization and Capacity**

**Sectors/Track<sup>1</sup>**

Bytes/Sector	Priam 8"	Priam 14"	Shugart/Quantum	Memorex	Pertec
128	70	104	54	68	73
256	42	62	31	39	44
512	23	34	17	21	24
1024	12	18	9	11	13

**Note 1.** Maximum allowable for corresponding selection of bytes per sector.

**Formatted Capacity/Drive<sup>2</sup>**

Bytes/Sector	Shugart	Quantum	Pertec	Priam	Memorex
128	7.08 MB	7.08 MB	13.05 MB	23.29 MB	8.49 MB
256	8.12	8.12	15.74	27.94	9.74
512	8.91	8.91	17.17	30.62	10.49
1024	9.43	9.43	18.60	31.95	10.98

**Note 2.** Shugart SA 1004, Quantum Q2010, Priam 3450, Pertec D8000, Memorex 101.

**Drives per Controller**

**8" Winchester Disk Drives** — Up to four Shugart, Pertec, Quantum or Priam drives; up to two Memorex 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**

**1215193** — 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

**ORDERING INFORMATION**

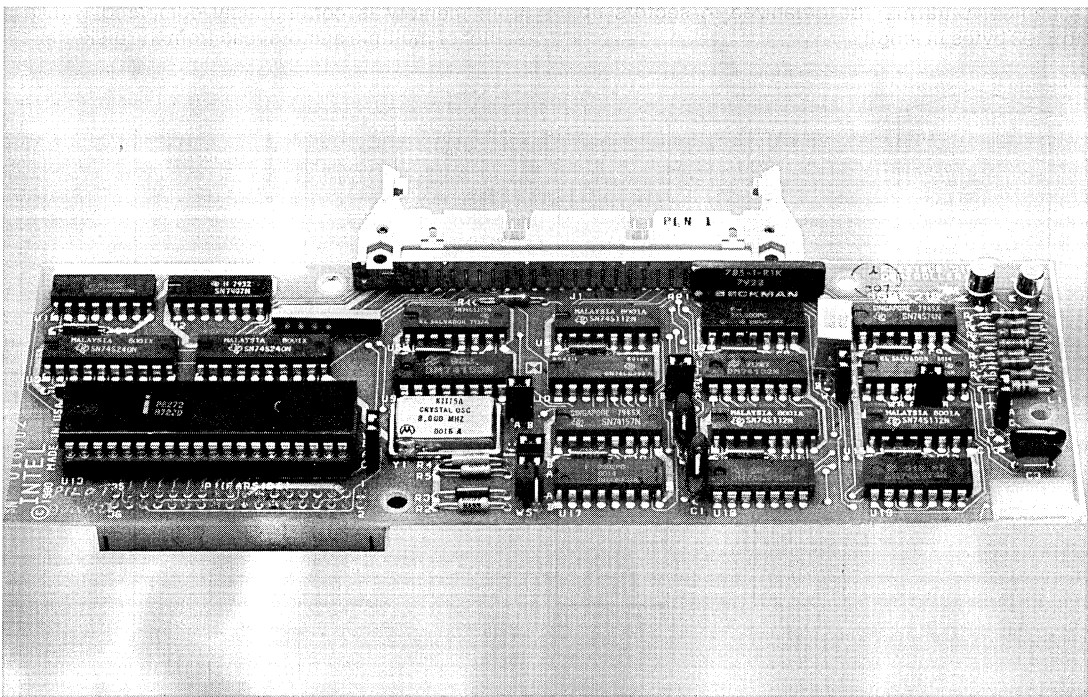
Part Number	Description
SBC 215A	Winchester Disk Controller (open-loop)
SBC 215B	Winchester Disk Controller (closed-loop)



## 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, write, verify, and search 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, write, and verify 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.

**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 read from the diskette.

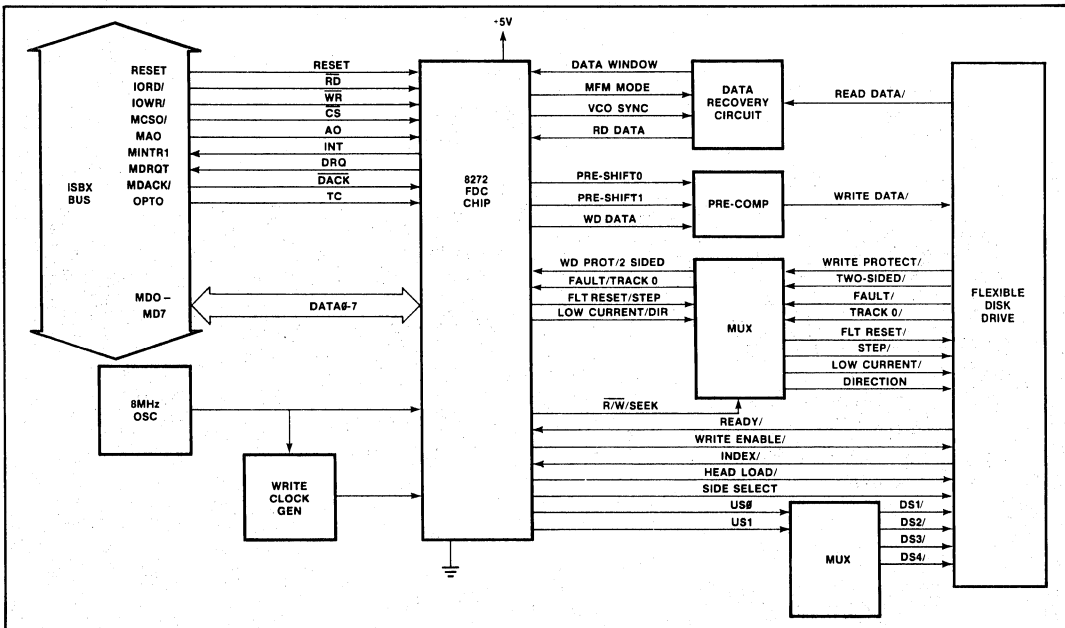


Figure 1. Block Diagram of iSBX 218 Board

## iSBX 218

**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 (5½") flexible disk drives. The drives may be single or double sided. Drives known to be compatible are:

Standard (8")		Mini (5½")
Caldisk	143M	Shugart 450
Remex	RFD 4000	
Memorex	550	
MFE	700	
Siemens	FDD 200-8	
Shugart	SA 850	
Pertec	FD 650	

**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			77			77			77		
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

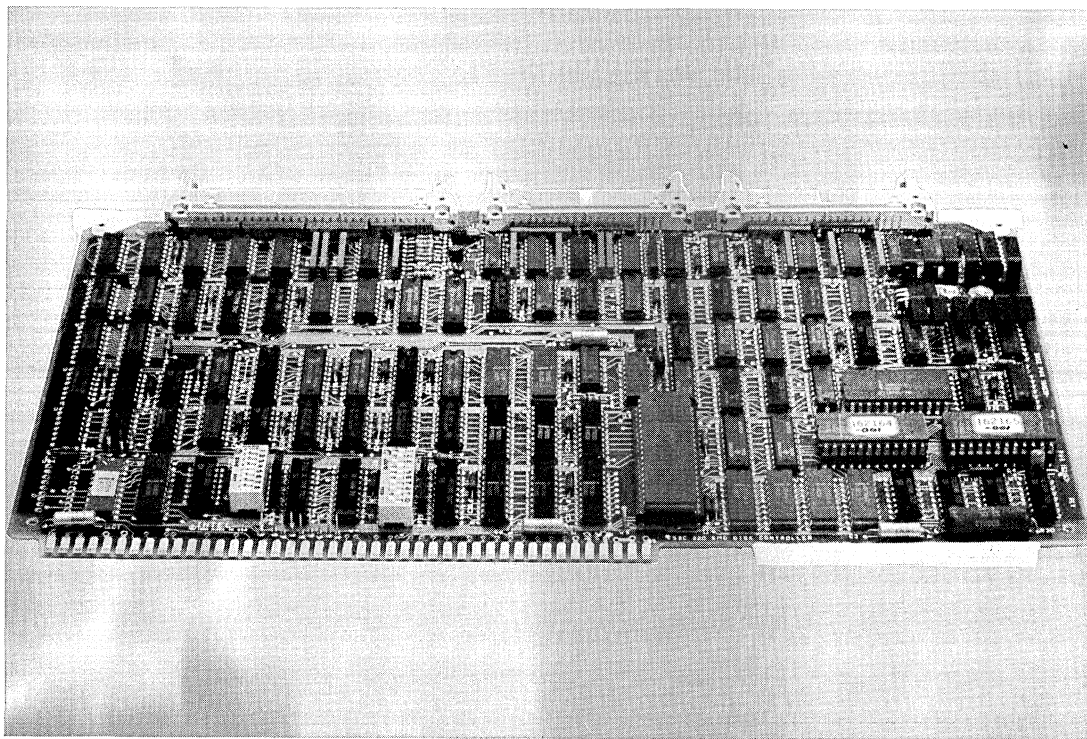


PRELIMINARY

## 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
- 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.





## 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

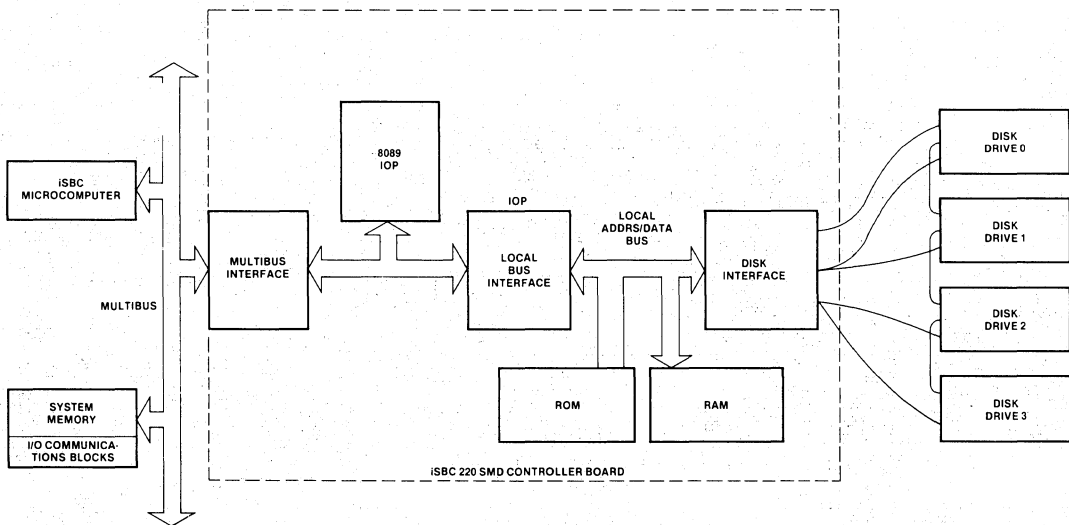
to 32 bits in length and using an 8089 algorithm can correct an erroneous burst up to 11 bits in length.

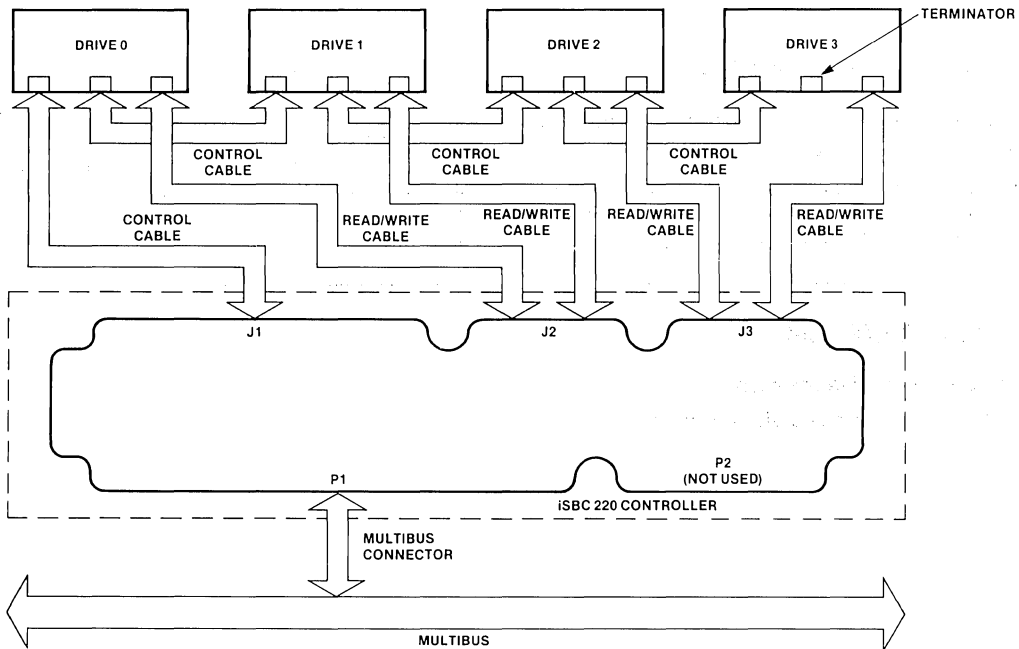
### SMD Interface

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.





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<sup>1</sup>

**Note 1:** On-board voltage regulator allows optional – 12 VDC usage from MULTIBUS.

## Data Organization and Capacity

**Bytes per Sector**<sup>2</sup> — 128 256 521 1024

**Sectors per Track**<sup>2</sup> — 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**

**12159** — 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**

<b>Part Number</b>	<b>Description</b>
SBC 220	SMD Disk Controller



*Memory Expansion  
Boards*

---

**6**

THE UNIVERSITY OF CHICAGO  
DEPARTMENT OF CHEMISTRY  
5800 S. UNIVERSITY AVENUE  
CHICAGO, ILLINOIS 60637

RECEIVED

NOV 15 1964

1964

NOV 15 1964

NOV 15 1964

NOV 15 1964

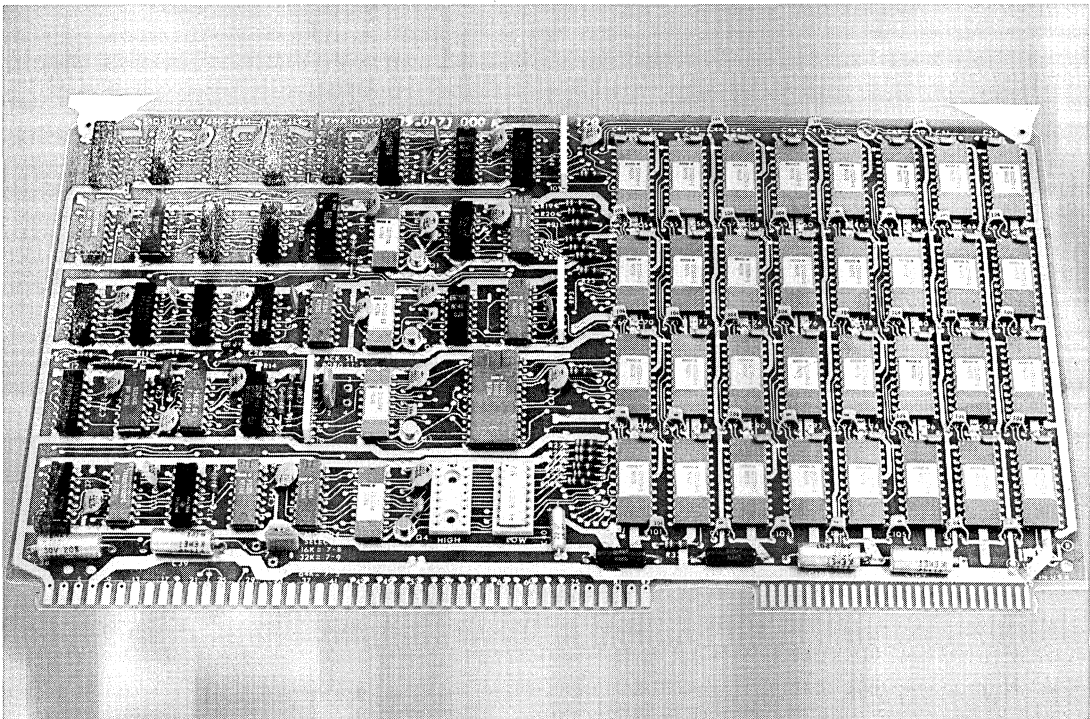
NOV 15 1964



## iSBC 016 16K-BYTE RAM MEMORY BOARD

- iSBC 80 RAM memory expansion through the MULTIBUS interface
- 16K-byte read/write memory capacity
- On-board hardware refresh for all dynamic memory elements
- Jumper selectable starting address for 16K contiguous addresses
- Read/write data buffers
- TTL compatible data, address, and command signal interface

The iSBC 016 16K-Byte RAM Memory Board is a member of Intel's complete line of iSBC memory and I/O expansion boards. The iSBC 016 interfaces directly to any iSBC 80 single board computer via the system bus to expand RAM memory capacity. The board contains 16K bytes of read/write memory, implemented using 2107 dynamic RAM memory components. On-board refresh hardware refreshes 64 bit positions of all 32 RAM elements every 14 microseconds. Each refresh cycle utilizes memory for 735 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. The iSBC 016 contains a jumper used to select contiguous 16K-byte address segments starting in locations 0000, 4000, 8000, or C000. Read/Write buffers reside on the 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.



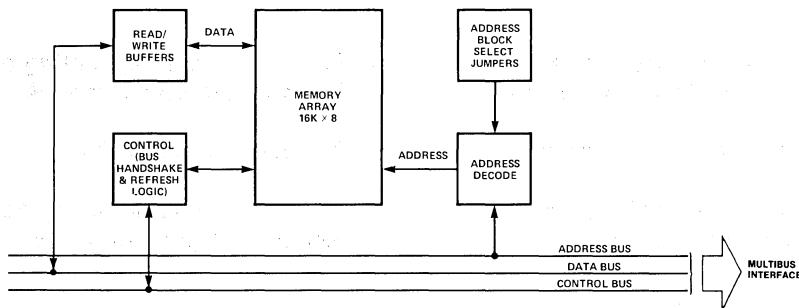


Figure 1. ISBC 016 16K RAM Memory Expansion Board Block Diagram

**SPECIFICATIONS**

**Word Size**

8 bits

**Memory Size**

16,384 bytes

**Cycle Times**

Read Cycle — 735 ns max

Write Cycle — 1360 ns max

Refresh Cycle — 735 ns max

**Interface**

All address, data, and command signals are TTL compatible.

**Address Selection**

Jumper selection of base address of 16K contiguous memory block to reside in locations 0000, 4000, 8000, or C000.

**Connectors**

Edge 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 (415.2 gm)

**Electrical Characteristics**

**DC Power Requirements**

V<sub>CC</sub> = +5V DC ± 5%

I<sub>CC</sub> = 1.2A typ; 1.5A max

V<sub>DD</sub> = +12V DC ± 5%

I<sub>DD</sub> = 0.7A typ; 1.0A max

V<sub>BB</sub> = -5V DC ± 5%

I<sub>BB</sub> = 0.2 mA typ; 3.2 mA max

**Environmental Characteristics**

Operating Temperature — 0°C to 55°C

**Reference Manual**

9800279A — ISBC 016 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**

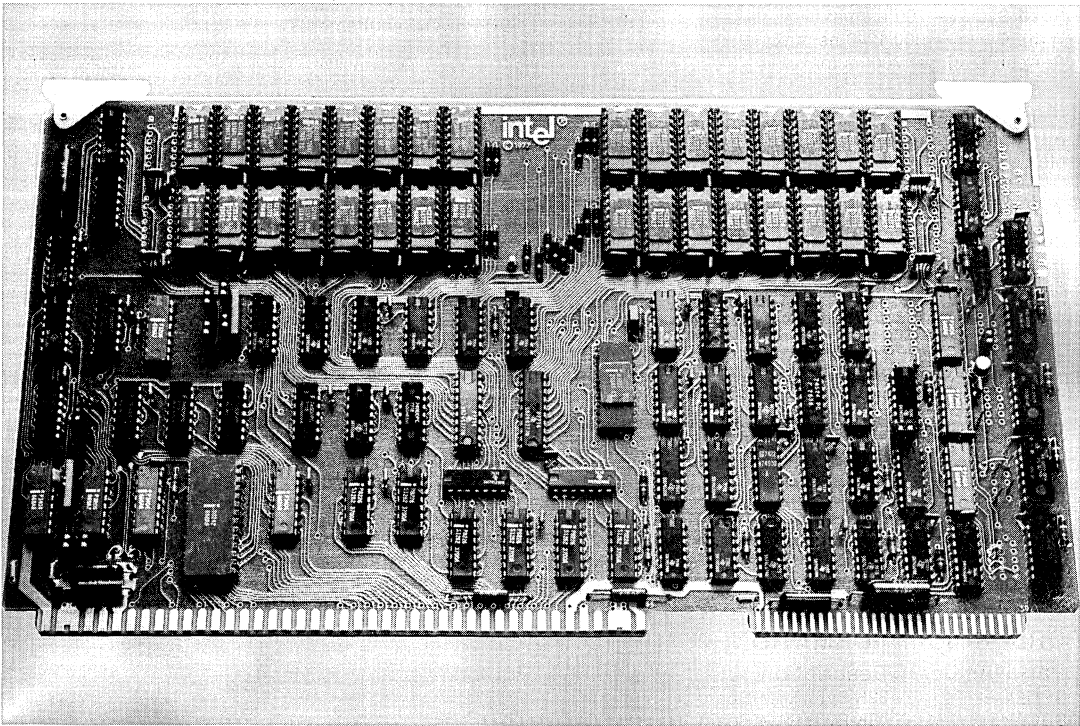
Part Number	Description
SBC 016	16K-Byte RAM Memory Board



## iSBC 032/048/064 RAM MEMORY BOARDS

- iSBC 80 and iSBC 86 RAM memory expansion through direct MULTIBUS interface
- 32K, 48K, 64K bytes of read/write memory (iSBC 032, iSBC 048, iSBC 064 boards, respectively)
- On-board hardware for refresh of all dynamic memory elements
- Auxiliary power bus and memory protect control logic provided for battery backup RAM requirements
- Jumper selectable starting address for independent 16K-byte memory segments
- Read/write data buffers
- TTL compatible data, address, and command signal interface

The iSBC 032, iSBC 048, and iSBC 064 RAM Memory Boards are members of Intel's complete line of iSBC memory and I/O expansion boards. Each board interfaces directly to any Intel iSBC 80 or iSBC 86 single board computer via the MULTIBUS interface to expand RAM memory capacity. The iSBC 032 contains 32K, the iSBC 048, 48K, and the iSBC 064, 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. The iSBC 032 contains jumpers used to individually select two independent 16-byte memory segments, and the iSBC 048 contains jumpers used to individually select three independent 16K-byte memory segments starting on 16K-byte boundaries in one of sixteen 64K-byte pages. 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.





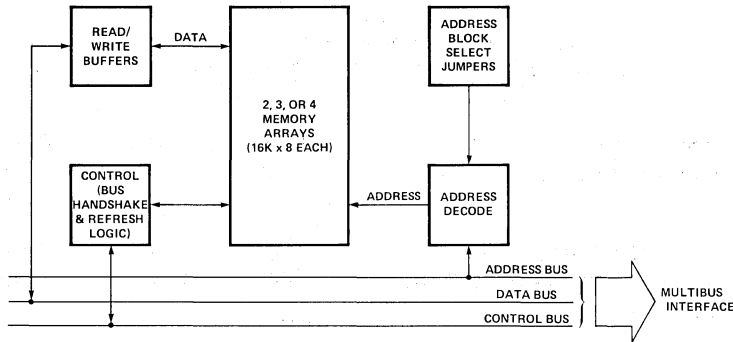


Figure 1. RAM Memory Expansion Boards Block Diagram

## SPECIFICATIONS

### Word Size

8 bits and 16 bits

### Memory Size

32,768 bytes (iSBC 032), 49,152 bytes (iSBC 048), 65,536 bytes (iSBC 064)

### 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 for independent 16K-byte memory blocks starting on 16K-byte boundaries in one of sixteen 64K-byte pages.

### 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) <sup>1</sup>	AUX Power RAM Access (max) <sup>2</sup>	AUX Power No RAM Access (max)
$V_{CC} = +5V \pm 5\%$	$I_{CC} = 3.2A$	1.7A	1.7A
$V_{DD} = +12V \pm 5\%$	$I_{DD} = 600\text{ mA}$	600 mA	120 mA
$V_{BB} = -5V \pm 5\%$	$I_{BB} = 10\text{ mA}$	10 mA	3 mA

### Notes

- All current values apply to the iSBC 032, iSBC 048 or iSBC 064 boards and include AUX power.
- RAM chips and RAM control logic powered via auxiliary power bus.
- 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 032	32K-Byte RAM Board
SBC 048	48K-Byte RAM Board
SBC 064	64K-Byte RAM Board



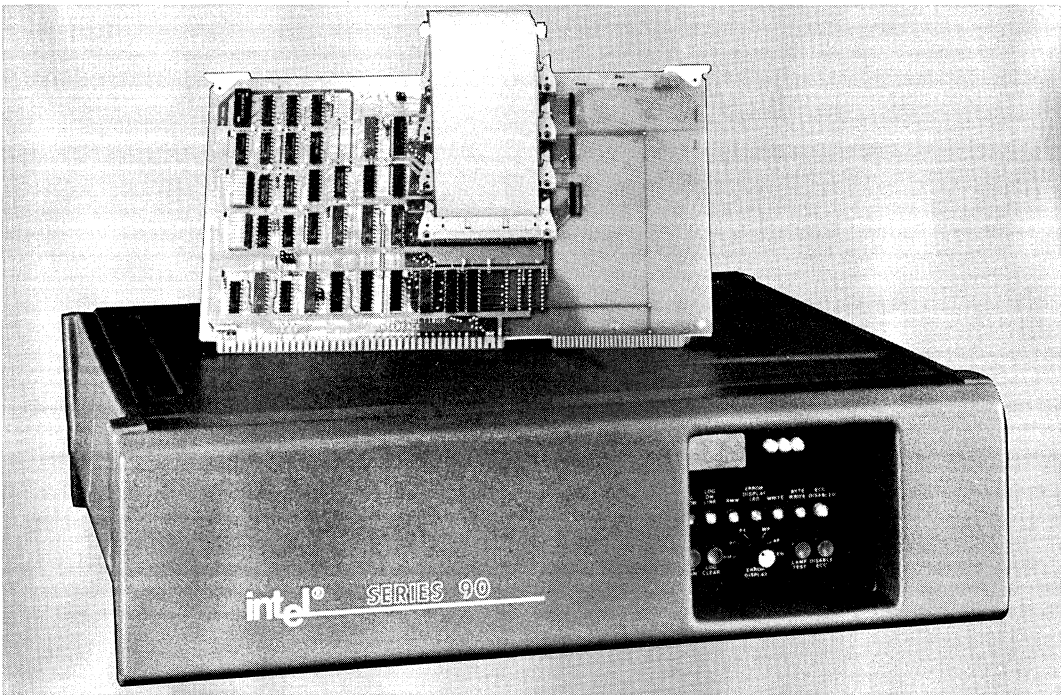
## iSBC 090™ MEMORY SYSTEM

- 128K, 256K, 384K, 512K, 768K or 1024 K-byte Capacities
- Error Correcting Circuitry (ECC) 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 by 19.00 inches by 19.5 inches. 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 rack mount system or 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, Inteltec, Library Manager, Megachassis, Micromap, MULTIBUS, PROMPT, RMX, UPI,  $\mu$ Scope, 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, 1980

**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, or up to 512K-bytes in 128K increments. Each 128K-byte or 256K-byte increment is provided by adding one memory module in the memory. Five standard system capacities are offered: 128K, 256K, 384K, 512K, 768K and 1024K-bytes. The 128K system can be expanded to 512K in 128K increments; the 256K and larger systems to 1024K-bytes in 256K-byte increments.

**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.

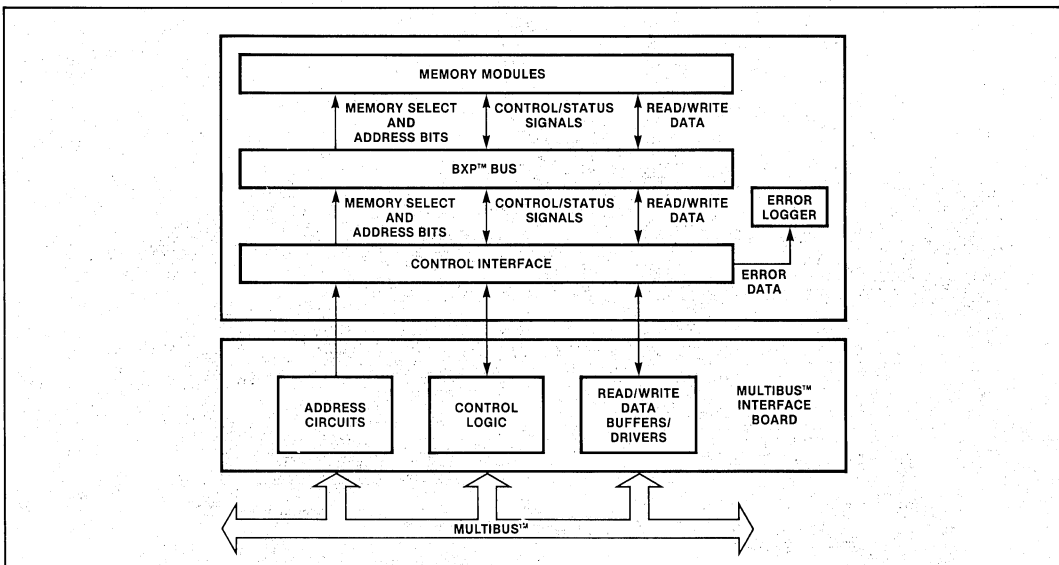


Figure 1. iSBC 090™ Memory System, Block Diagram

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

128K, 256K, 384K, 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 — 19-inch rack/table top (optional)

### 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	Model Number	Description
SBC-090-x*3F	128K Byte Multibus-compatible dynamic RAM memory system with ECC. Expandable to 512K Bytes using CM-90200-F22 memory module (below). NOTE: "x" in model no. must be specified per table below to define input voltage and chassis configuration.	SBC-090-x*1H	256K Byte Multibus-compatible dynamic RAM memory system with ECC. Expandable to 1024K Bytes using CM-90100-H22 memory module (below). NOTE: "x" in model no. must be specified per table below to define input voltage and chassis configuration.
SBC-090-x*3H	256K Byte memory system. Otherwise identical to above SBC-090-x*3F.	SBC-090-x*1K	512K Byte memory system. Otherwise identical to above SBC-090-x*1H.
SBC-090-x*3J	384K Byte memory system. Otherwise identical to above SBC-090-x*3F.	SBC-090-x*1L	768K Byte memory system. Otherwise identical to above SBC-090-x*1H.
SBC-090-x*3K	512K Byte memory system. Not expandable. Otherwise identical to above SBC-090-x*3F.	SBC-090-x*1M	1024K Byte memory system. Not expandable. Otherwise identical to above SBC-090-x*1H.
CM-90200-F22	128K Byte memory module for expansion of SBC-090-x*3F, H, or J.	CM-90100-H22	256K Byte memory module for expansion of SBC-090-x*1H, K, or L.

**NOTE**

To order SBC 090, the "x" in model no. *must* be specified per the following table to define input voltage and chassis configuration.

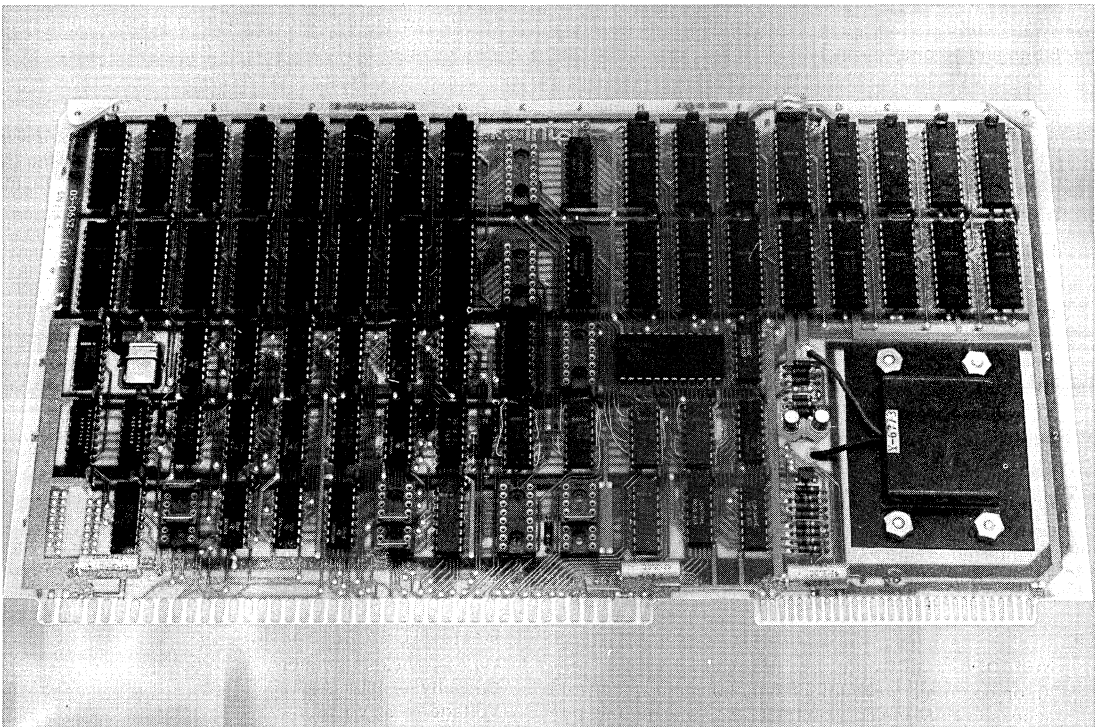
- x = 0 for rack mount unit, 110 VAC
- x = 1 for rack mount unit, 220 VAC
- x = 2 for table-top unit, 110 VAC
- x = 3 for table-top unit, 220 VAC



## iSBC 094 4K-BYTE CMOS RAM MEMORY BATTERY BACKUP BOARD

- iSBC 80 and iSBC 86 nonvolatile 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.



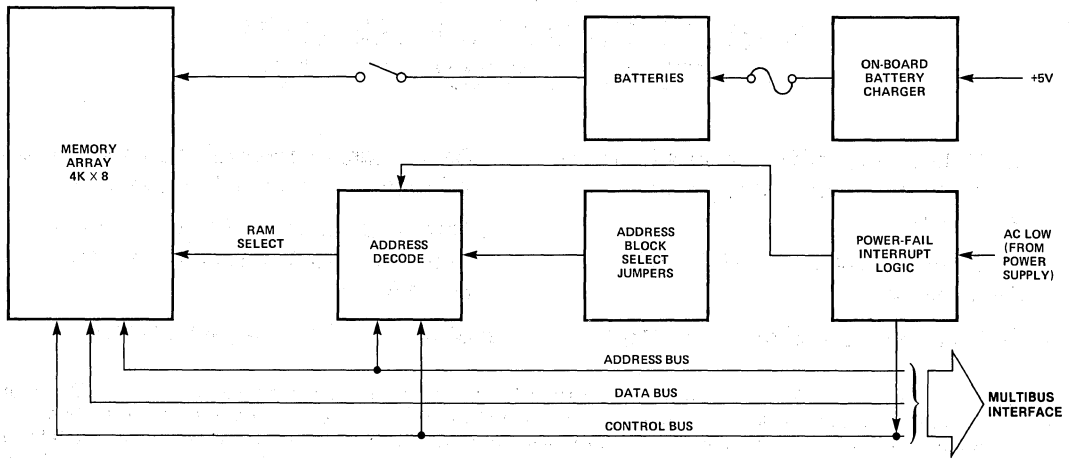


Figure 1. iSBC 094 Memory Backup Board Block Diagram

**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 pro-

vided 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<sub>H</sub>, 1000<sub>H</sub>, ... F000<sub>H</sub>). 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 <sup>1</sup>	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  $\pm 5\%$

$I_{CC} = 0.8A$  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





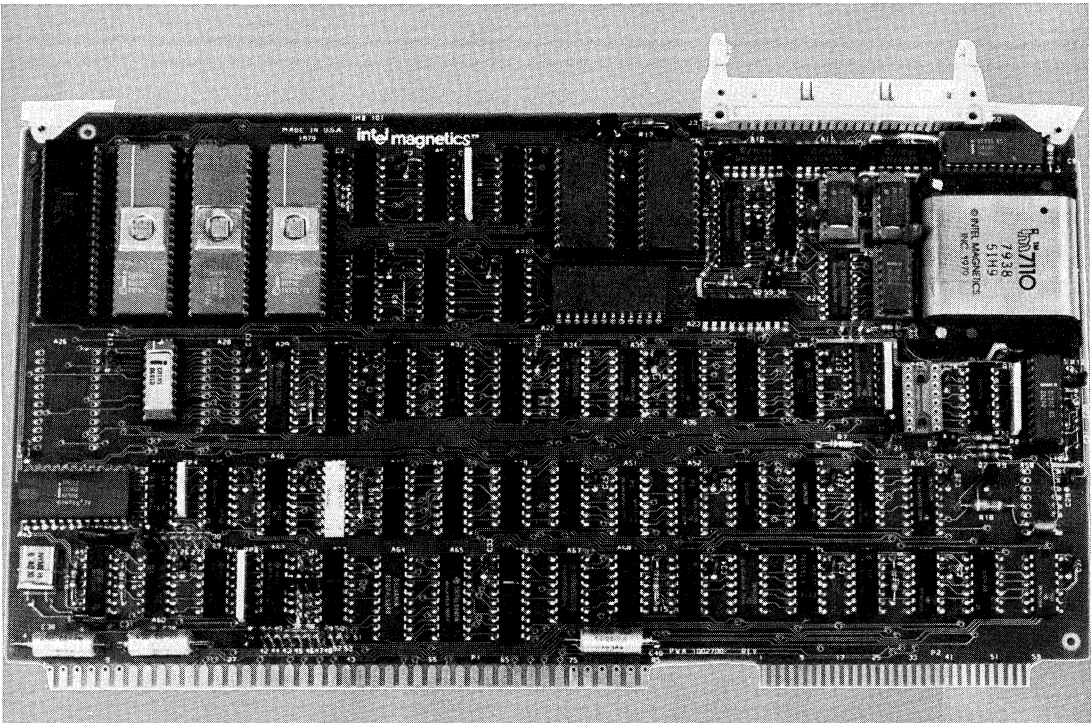
## iSBC 250™ 1 Megabit Bubble Memory Board

- Completely Assembled 1 Megabit Bubble Memory System
- Direct MULTIBUS™ Interface
- Standard 6.75 x 12-inch SBC Printed Circuit Board
- Non-Volatile Solid-State Memory
- Average Access time 48 ms
- Automatic Error Correction
- Power Fail Circuit Protects Bubble Memory and Stored Data
- Fully Compatible with iSBC-80™ and iSBC-86™ Single Board Computers

The iSBC 250 is a completely assembled 128K byte non-volatile memory utilizing the Intel Magnetics 7110 one megabit bubble memory, 7230 current pulse generator, 7242 dual formatter/sense amplifier, and 7250 and 7254 coil drivers. Designed for interface with MULTIBUS systems, the iSBC 250 can be plugged into Intellec® Microcomputer Development Systems.

Software provided on single- and double-density diskettes enables the user to perform maintenance functions or to perform reliability evaluation. Source code in 8080 Assembler Language for sample driver routine is supplied on each diskette.

Error correction and power fail protection capabilities enable the iSBC 250 to be used in OEM systems as well as in development applications.



## iSBC 250™ FUNCTIONS

The iSBC 250 megabit Bubble Memory Board consists of a one megabit bubble memory cell and a bubble memory controller based on integrated circuits.

The controller for the iSBC board is made up of an 8085A microprocessor, PROMs and EPROMs. The 8085A communicates with the host CPU through a set of registers on the board via I/O commands. By interpreting these registers, the 8085A microprocessor controls timing and analog circuits to access the bubble memory. Data are passed via a FIFO on the board. An on-board clock generator (a crystal-controlled oscillator and driver) allows the board to function independently of the host CPU clock.

The iSBC 250 board is address-decoded by an 8205 decoder. The controller address is selected by a jumper option. Normal jumper configurations provide compatibility with MULTIBUS structures. The host addressable registers are a 16 x 8 RAM, and the addresses are controlled either by the host bus or the 8085A microprocessor.

The bubble memory cell operates as follows:

The 7250 coil predriver translates 5V signals to 12V, to drive the 7254 quad transistor packs. These transistors in turn, drive the X and Y coils at the 7110 bubble memory.

The 7230 current pulse generator provides pulses to the 7110 bubble memory and serves as a voltage supply monitor. The 7230 device issues a POWER-FAIL signal if either +5V or +12V power falls below a preset safe operating threshold value.

The 7242 dual formatter/sense amplifier communicates with the 8085A-based controller, the 7230 current pulse generator and the 7110 bubble memory. The 7242 formatter also provides internal error correction by means of a 14-bit Fire code. This code can correct burst errors up to 5 bits in length, in each page. This error correction system significantly improves the read error rate. A zero-suppression and filter system is also incorporated between the 7242 inputs and the 7110 detector outputs to further minimize errors.

The board includes circuits to recreate seed bubbles in the 7110 bubble memory as needed. The iSBC 250 is designed to utilize the transparent redundancy inherent in the 7110 magnetic bubble memory device. The bootloop map information of the 7110 is

written into the 7242 during system initialization. The 7242 uses the bootloop map to gate the data to insure that bad loops are not written into and that data from these loops are not read out.

## SOFTWARE

A set of object programs for exercising and maintaining the iSBC 250 is supplied on diskettes. These programs can be used on any Intellect® Microcomputer Development System with ISIS-II.

The programs are:

1. MASK—To create a bad loop map in a format suitable for writing to a bubble memory chip.
2. UNMASK—To determine the correspondence between a given bit in an input/output buffer and the physical minor loop, given a bad loop map.
3. VERIFY—To write bubble memory bootloop.
4. BUBBLE—To transfer data between bubble, RAM or other devices in the system.
5. SEED—To regenerate seeds on the bubble.
6. TEST—To check the operation of the board. It enables the user to perform device evaluations by writing data in the 7110, reading the stored data and indicating errors.

In addition, a stand alone monitor is provided in source code form. It allows the user to execute a set of commands, such as, INITIALIZE, READ, WRITE, ABORT AND SEEK for the purpose of demonstrating the functions of the iSBC 250 in a Single Board Computer or an Intellect® Microcomputer Development System environment.

Since the monitor software is in source code form, it may be modified by the user to reside on any SBC processor board or may be used as a model for the user's OEM application software.

## SYSTEM PROGRAMMING

The iSBC 250 appears as a peripheral device to the processor. The processor communicates with the iSBC 250 through two ports. The command, address and status information through one of the ports, and data is passed in the second port.

Data is organized as 2048 logical pages with 64 bytes per page. Bubble addressing is done by page number.

Data is transferred in byte mode. Interrupt can be set on completion of command or on encountering error conditions.

## SPECIFICATIONS

### Memory Size

128K bytes.

### Interface

All address, data and control signals are Intel MULTIBUS compatible.

### Electrical Characteristics

D.C. Power (Max)

- +5 Volts D.C. ±5%, 3.0A Max.
- +12 Volts D.C. ±5%, 0.5A Max.
- 12 Volts D.C. ±5%, 0.1A Max.

### Performance

- Rotating Rate Field: 50kHz
- Maximum Data Rate: 100K bits/sec
- Nominal Data Rate: Read 35K bits/sec,  
Write 41K bits/sec
- Average Access Time: 48 ms

### Error Rates

	Typical Without Correction	Calculated With Correction
Read	$10^{-9}$	$10^{-16}$
Data	$10^{-11}$	$10^{-20}$

## Connector

Mating Connector: Control Data VPB01E43D00A1 or Viking 2VH43/1AV5. 86-pin double-sided PC edge connector with 0.40 cm (0.156 inch) contact centers.

## Physical Characteristics

- Length: 30.48 cm (12 inches)
- Height: 17.15 cm (6.75 inches)
- Depth: 1.45 cm (0.57 inch)
- Weight: 447g (16 ounces)

## Environment

- Operating Temperature: 0–50°C for iSBC-250  
10–50°C for iSBC-250-2

## Equipment Supplied

- iSBC 250 Bubble Memory Board
- iSBC 250 Operation Manual
- iSBC 250 Software (single and double density diskettes).

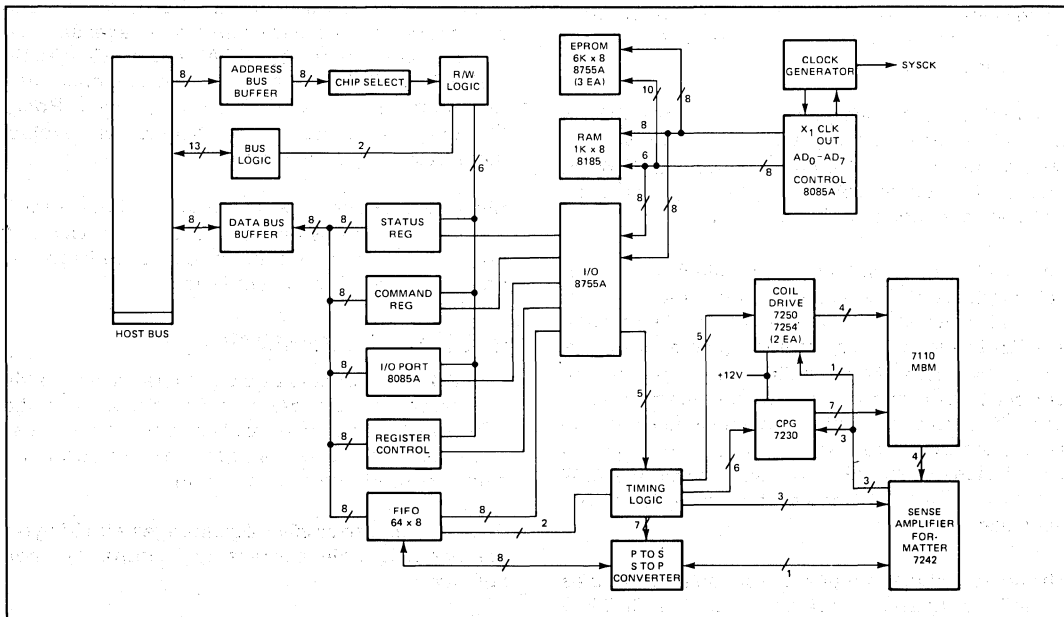


Figure 1. Block Diagram

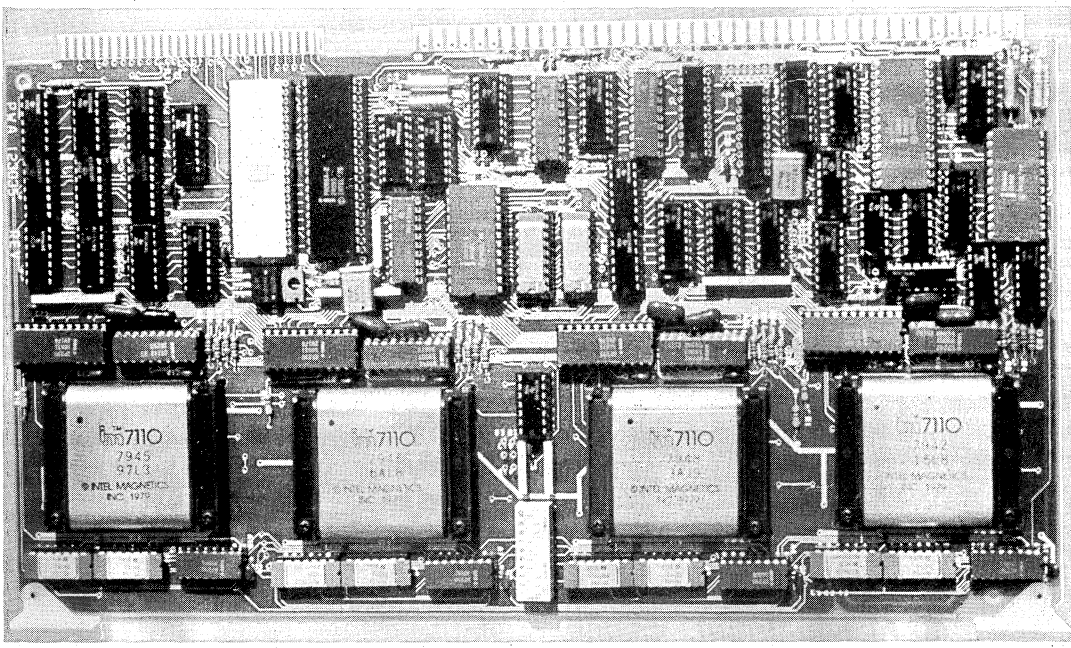
## iSBC 254™ BUBBLE MEMORY BOARD

- 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.

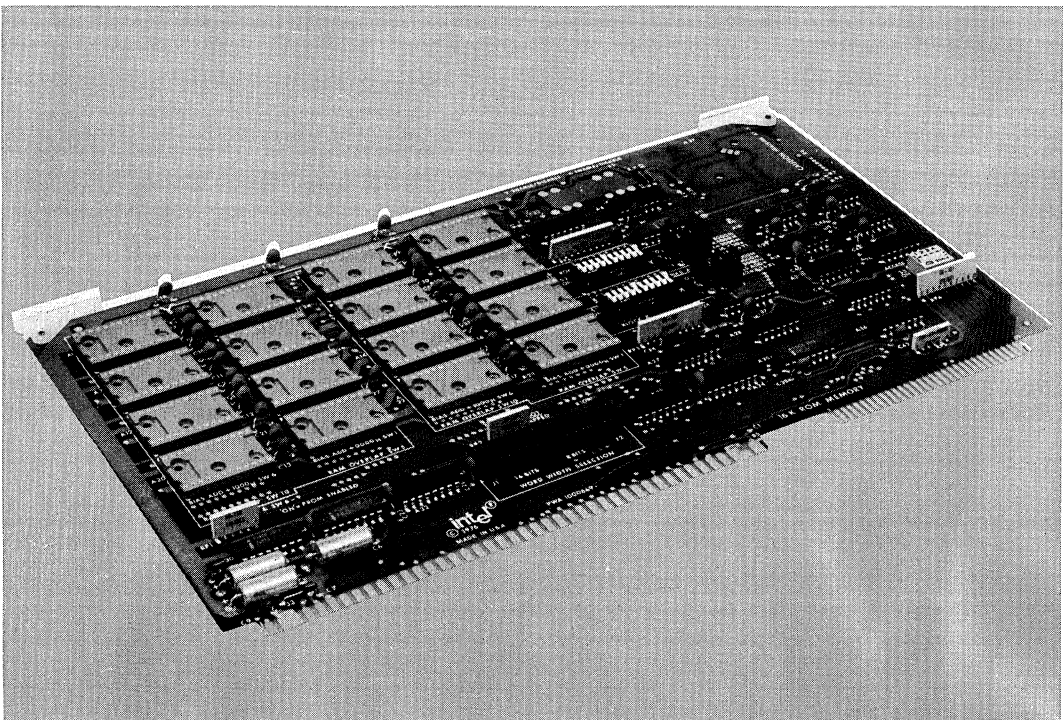




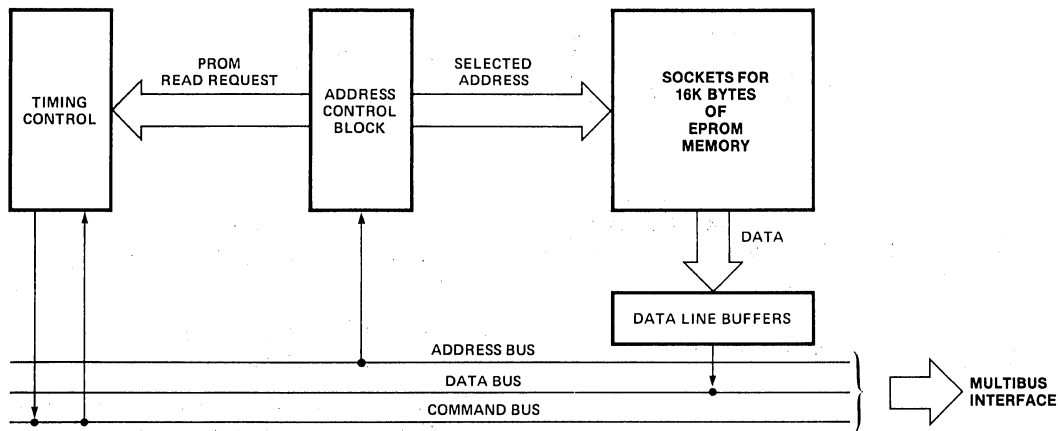
## iSBC 416 16K EPROM EXPANSION BOARD

- 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.



## iSBC 416



**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

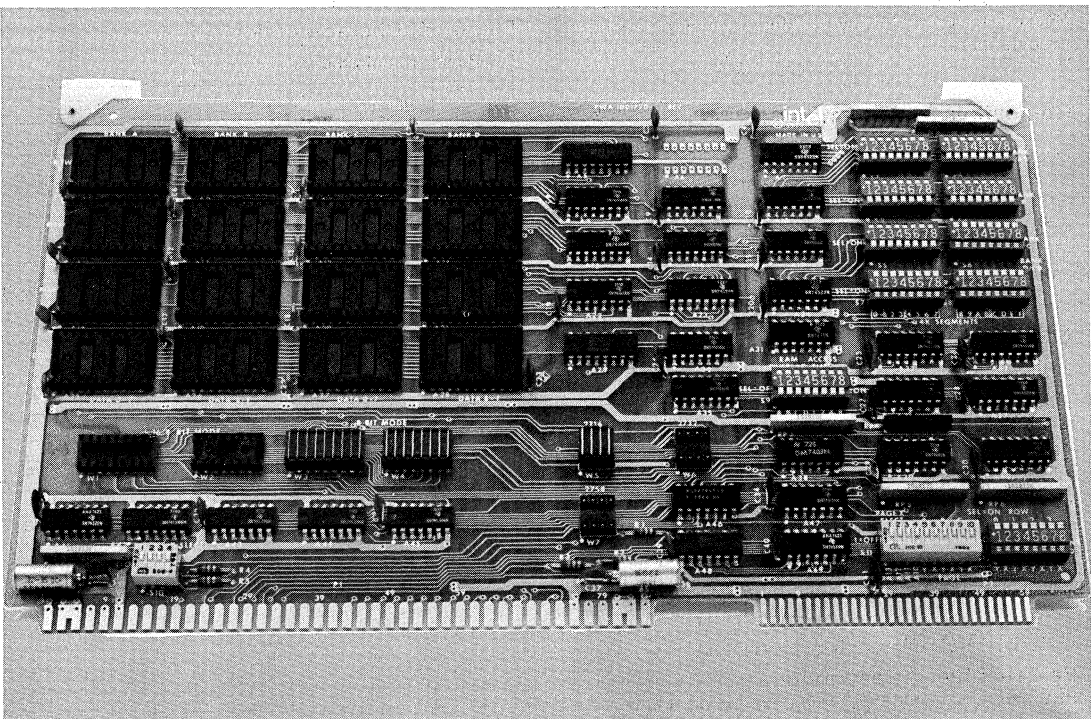
<b>Part Number</b>	<b>Description</b>
SBC 416	16K EPROM Expansion Board



## iSBC 464 or (pSBC 464\*) 64K-BYTE EPROM EXPANSION BOARD

- Provides EPROM/ROM expansion of iSBC 80 and iSBC 86 systems via direct MULTIBUS interface
- Sockets for up to 64K bytes of EPROM/ROM
- Compatible with Intel 2758, 2716 or 2732 erasable PROMs and 2136E masked ROMs
- Switch selectable base address on 4K-byte boundaries for each memory bank
- Assignable anywhere within a 1 megabyte address space
- EPROM/ROM components which are not enabled are placed in standby power mode
- Requires a single +5V power supply

The iSBC 464 64K-Byte EPROM Expansion Board 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 or iSBC 86 single board computers via the MULTIBUS system bus, to expand system EPROM/ROM memory capacity.



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

**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/ROM components installed by the user. The board is compatible with three different sizes of Intel EPROM/ROM devices. These are the 1K byte 2758 EPROM, the 2K byte 2716 EPROM/2316E ROM and the 4K byte 2732 EPROM/2332 ROM. Although only one device size may be used, EPROM and ROM may be mixed on the same board.

**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.

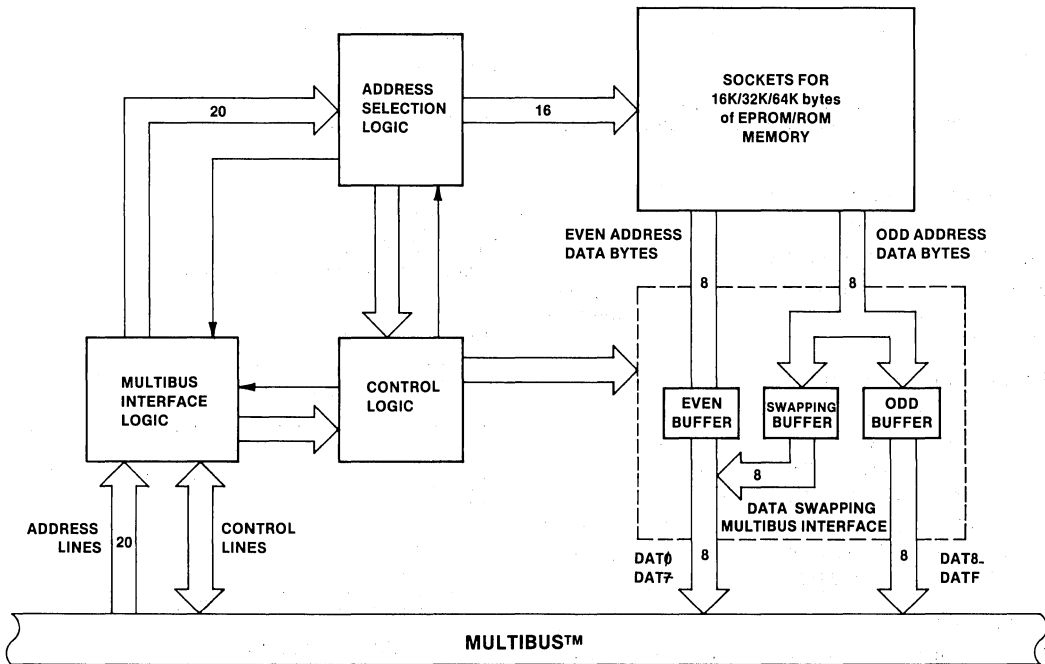


Figure 1. iSBC 464 Block Diagram



## 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, 2332, 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 accessible RAM in the system from responding to the current address. If an EPROM/ROM 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

**ROM** — 2316E or 2332

**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  $\pm$  5%

$I_{CC}$ : 1.1 amps without EPROM/ROMs

$I_{CC}$ : 1.6 amps with (16) 2716s or 2758s

$I_{CC}$ : 1.3 amps with (16) 2732s

$I_{CC}$ : 1.35 amps with (16) 2332s

$I_{CC}$ : 3.0 amps with (16) 2316s

### 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/ROM

### 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

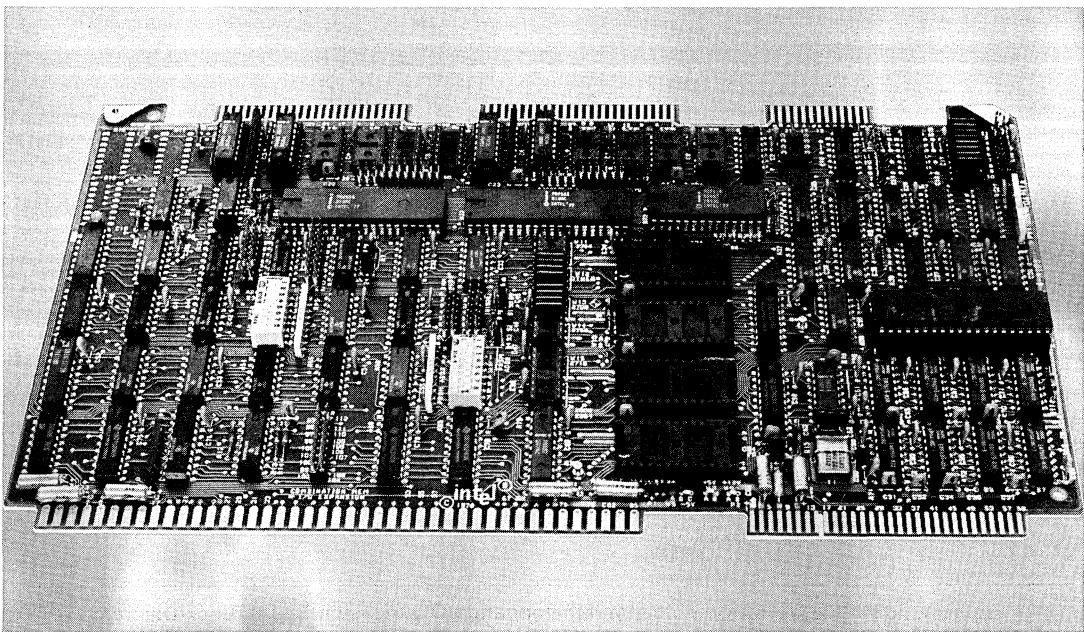
Part Number	Description
SBC 464	64K EPROM Expansion Board



## 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 1 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™ or iSBC 86™ 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, or 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 1 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. The boards have been designed to also be compatible with the forthcoming 64K EPROMs.

**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-

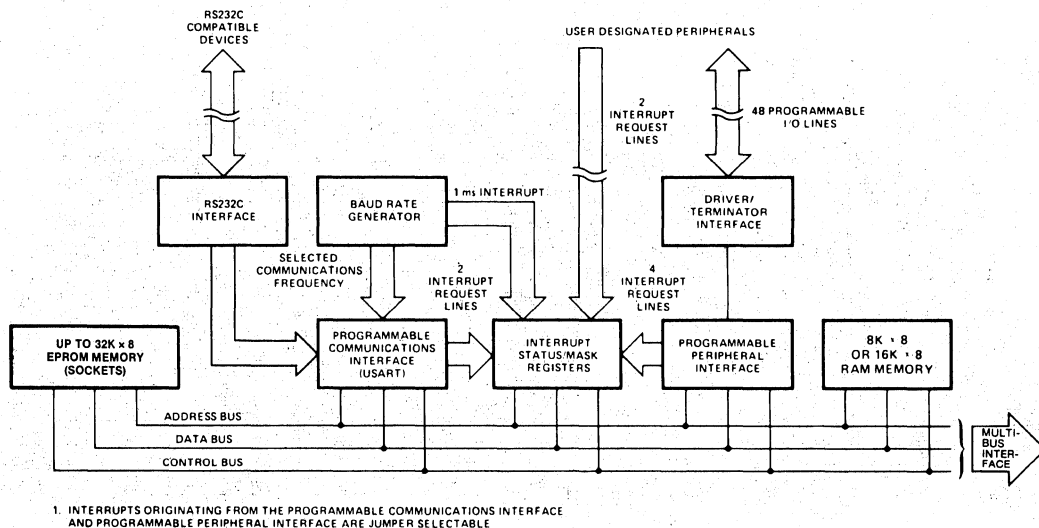


Figure 1. iSBC 108A/116A Combination Memory and I/O Expansion Board Block Diagram

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.

**Optically Isolated Interface** — The iSBC 530 Teletypewriter Adapter provides an optically isolated interface for those systems requiring a 20 mA current loop. The iSBC 530 may be used to interface the iSBC 108A/116A combination boards to teletypewriters and other 20 mA current loop equipment.

**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 80/10B, 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 <sup>1</sup>
	4	X		X			X <sup>1</sup>
4	8	X	X	X	X	X	
5	8	X	X	X	X		
6	4	X		X			X <sup>2</sup>
	4	X		X			X <sup>2</sup>

**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 1 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

### 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.

### 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.

## iSBC 108A/116A

### 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

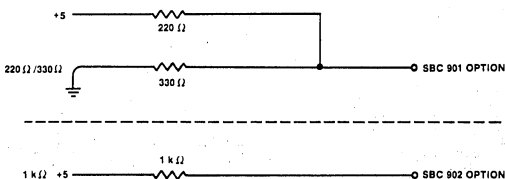
**I/O Drivers** — The following line drivers and terminators are all compatible with the I/O driver sockets on the iSBC 108A/116A board. Ports 1 and 4 have 25 mA totem-pole drivers and 1 k $\Omega$  terminators.

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** — 220 $\Omega$ /330 $\Omega$  divider or 1 k $\Omega$  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 <sub>DD</sub> = +12 $\pm$ 5%	V <sub>CC</sub> = +5 $\pm$ 5%	V <sub>BA</sub> = -5 $\pm$ 5%	V <sub>AA</sub> = -12 $\pm$ 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

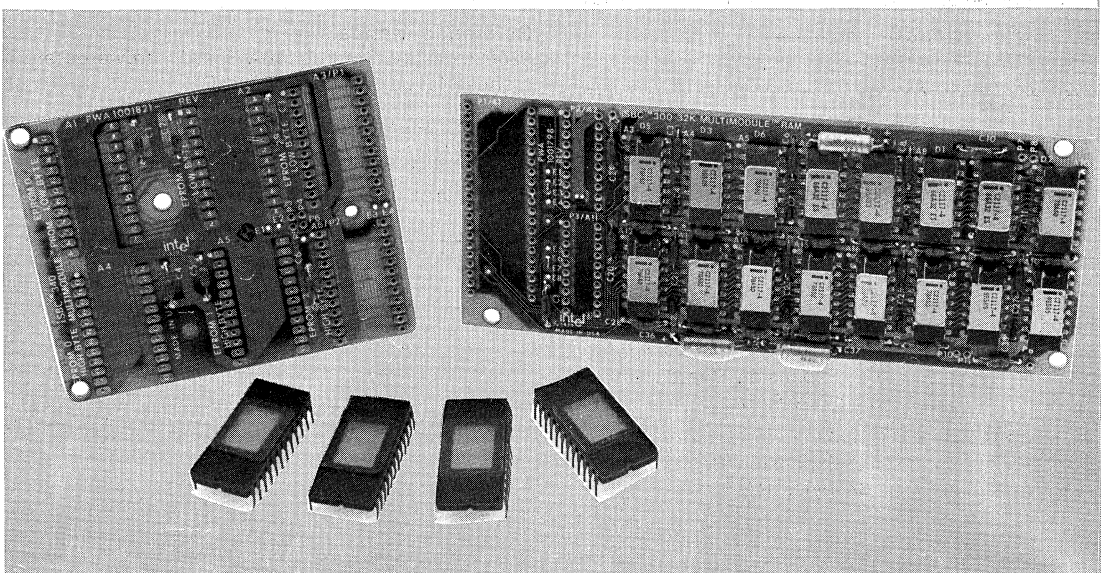
Part Number	Description
SBC 108A	Combination Memory and I/O Expansion Board with 8K bytes RAM
SBC 116A	Combination Memory and I/O Expansion Board with 16K bytes RAM



## 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**

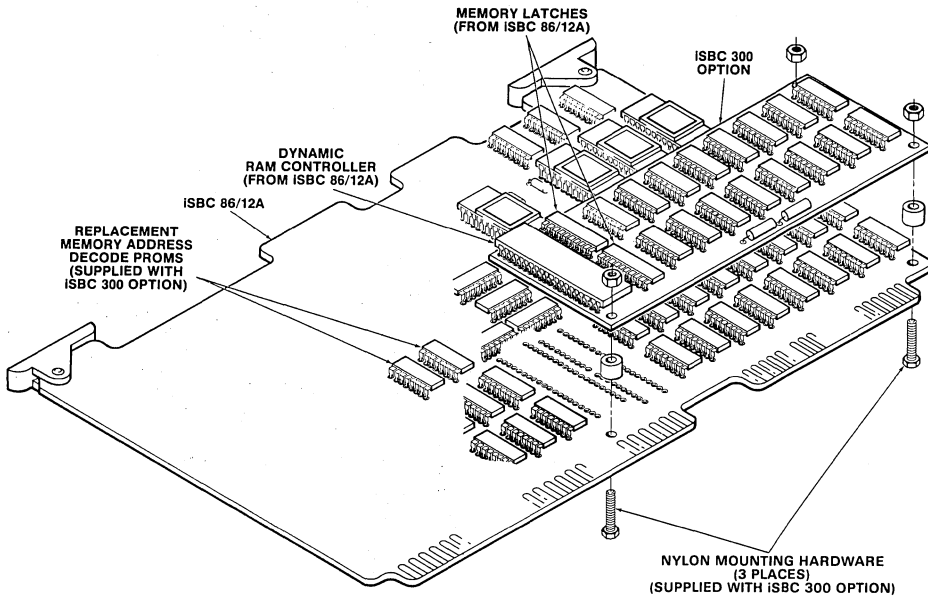
The iSBC 300 module contains sixteen 16K-byte dynamic RAM devices, sockets for the Intel® 8202A Dynamic computer. It expands the iSBC 86/12A board's on-board dual port RAM capacity from 32K bytes to 64K bytes. The iSBC 300 module contains sixteen 16K-byte dynamic RAM devices, sockets for the Intel® 8202 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  $\mu$ sec, write: 1.2  $\mu$ sec

**iSBC 340 Module** — Standard EPROMs (450 nsec): 1  $\mu$ sec, 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 segments. 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.

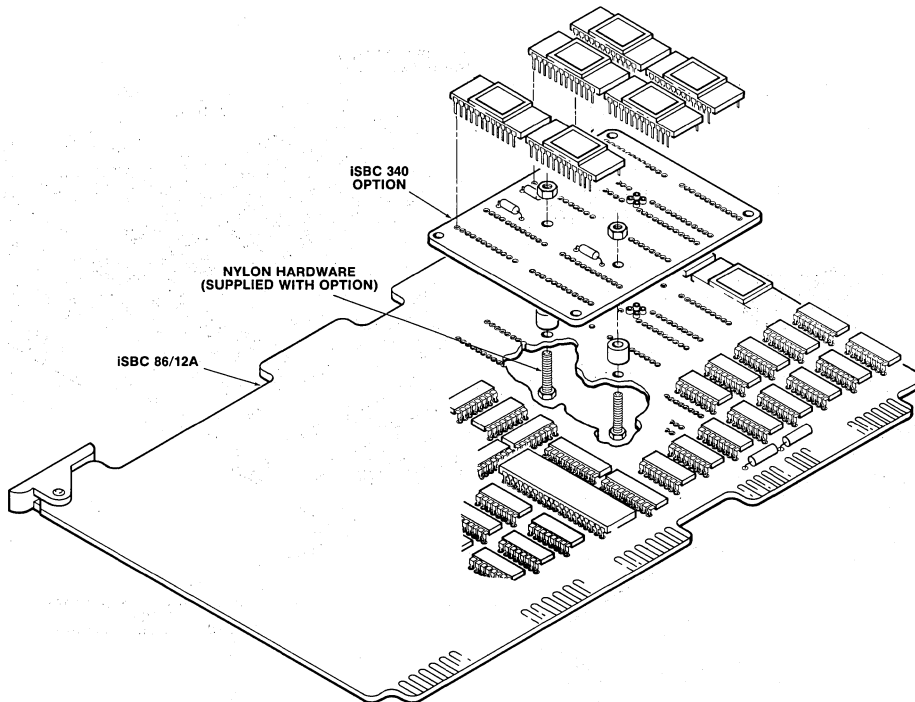


Figure 2. Installation of iSBC 340 MULTIMODULE EPROM Option on iSBC 86/12A Single Board Computer

## ISBC 300/340

### Electrical Characteristics

DC power requirements:

Voltage	ISBC 300	ISBC 340
+5 ±5%	1 mA	120 mA <sup>1</sup>
+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

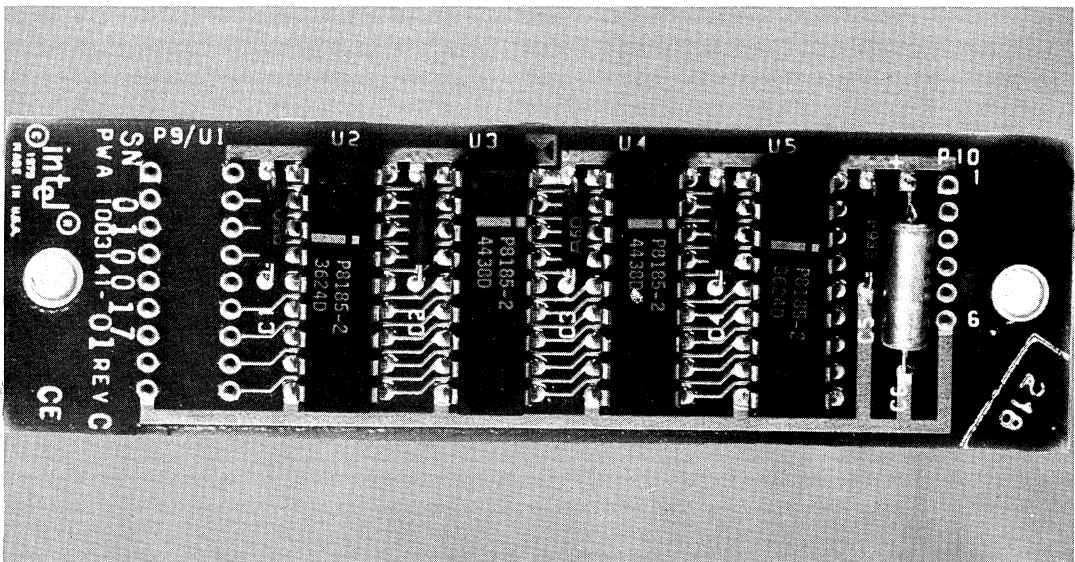
SBC 300 32K byte MULTIMODULE RAM  
SBC 340 16K byte MULTIMODULE EPROM



## iSBC 301 4K-BYTE RAM MULTIMODULE BOARD

- On-board memory expansion for iSBC 80/24 Single Board Computer to 8K bytes
- Provides 4K bytes of static RAM directly on the iSBC 80/24 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

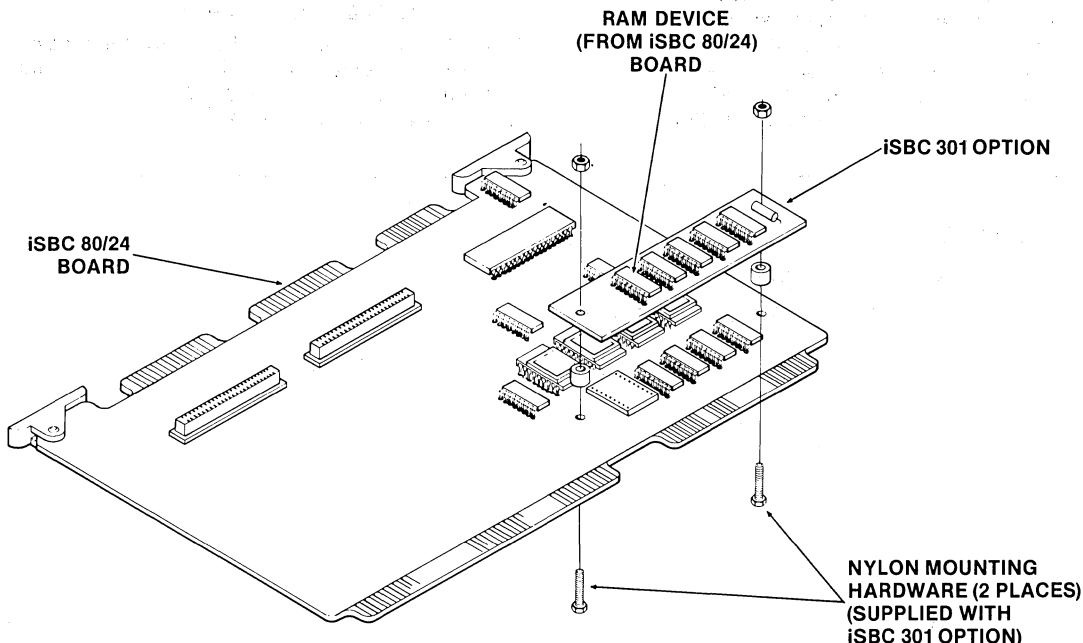
The Intel® iSBC 301 4K-Byte RAM MULTIMODULE Board provides simple, low cost expansion to double the RAM capacity on the iSBC 80/24 Single Board Computer to 8K bytes. This offers system designers a new level of flexibility in defining and implementing system memory requirements. Because memory is configured on-board, it can be accessed as quickly as the existing iSBC 80/24 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 single board computer. It expands the iSBC 80/24 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 board. To install the iSBC 301 MULTIMODULE board, one of the RAMs is removed from the iSBC 80/24 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 iSBC 80/24 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 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 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 on the iSBC 80/24 Single Board Computer**

**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 MULTIMODULE board is controlled by the host board via the address and chip select signal lines.

With the iSBC 80/24 board:

The 4K bytes of RAM on the iSBC 301 board occupy the 4K byte address space immediately preceding that of the iSBC 80/24 boards 4K RAM (i.e., default configuration —

iSBC 301 board's RAM 2000-2FFF  
iSBC 80/24 board's RAM 3000-3FFF)

**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 +  
iSBC 80/24 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 iSBC 80/24 Hardware Reference Manual; Order #142648-001. (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.

---

**SPECIFICATIONS**

**Part Number Description**

SBC 301 4K Byte RAM MULTIMODULE Board

---

*Digital I/O Expansion  
and Signal  
Conditioning Boards*

---

**7**

MEMORANDUM FOR THE RECORD

DATE: 10/15/54

TO: SAC, NEW YORK

FROM: SAC, NEW YORK

SUBJECT: [Illegible]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

[Illegible text]

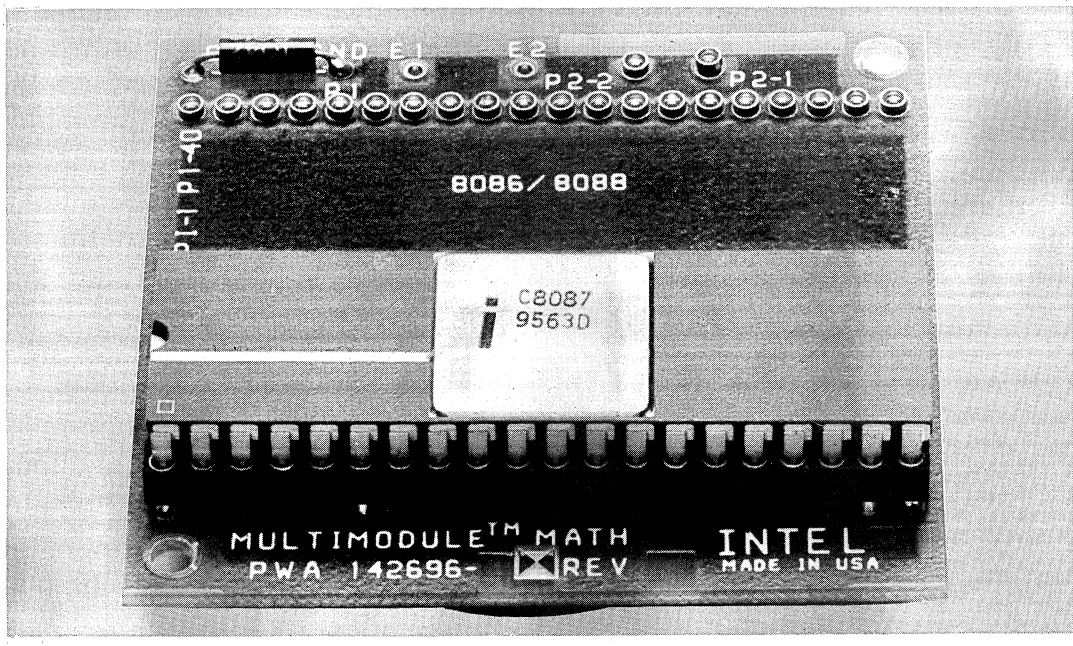
[Illegible text]



## 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.





## 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

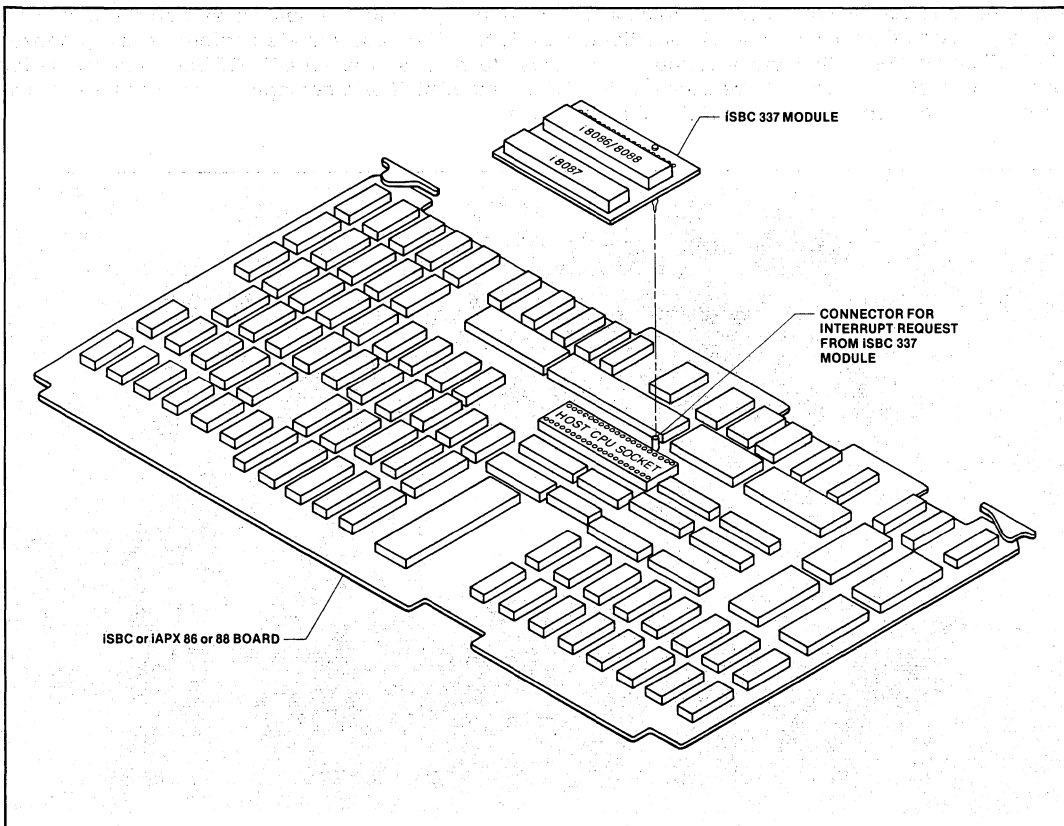


Figure 1. iSBC 337 Module Installation

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

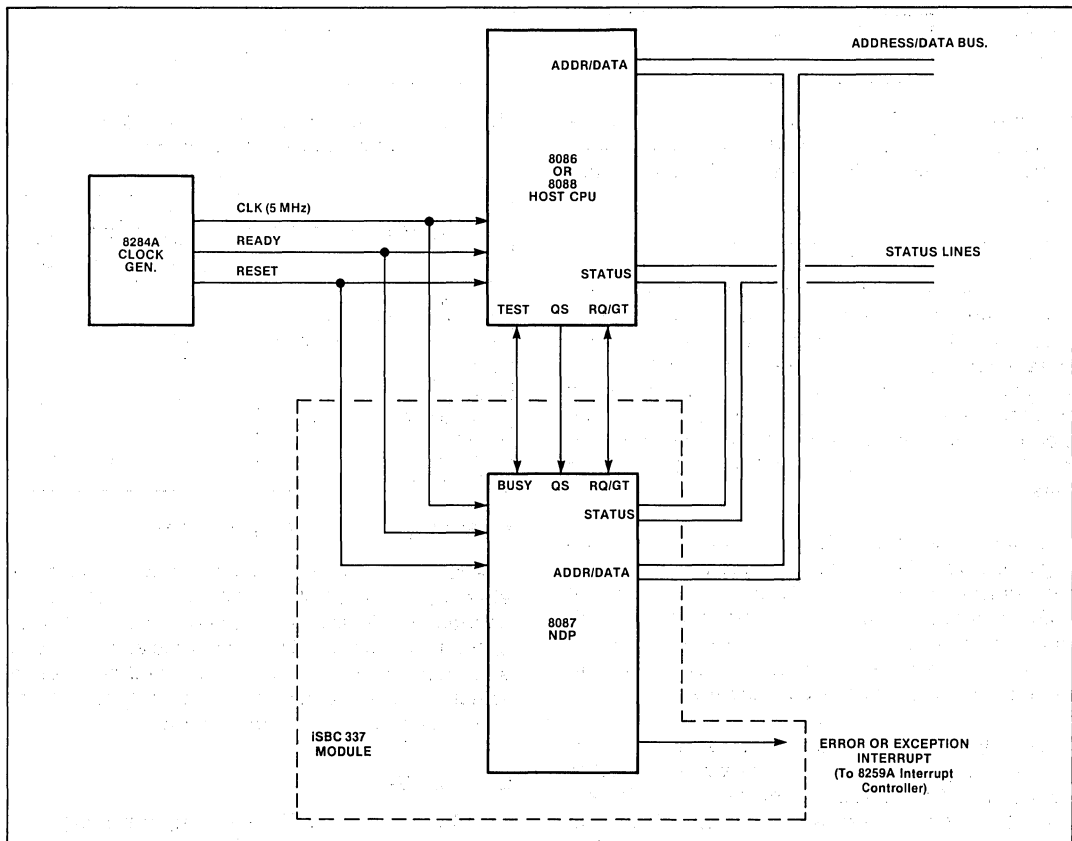


Figure 2. iSBC 337 System Configuration

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 <sub>15</sub> — I <sub>0</sub>								Two's Complement																									
Short Integer	$10^9$	32 Bits	I <sub>31</sub> — I <sub>0</sub>																Two's Complement																	
Long Integer	$10^{19}$	64 Bits	I <sub>63</sub> — I <sub>0</sub>																																Two's Complement	
Packed BCD	$10^{18}$	18 Digits	S	—	D <sub>17</sub>	D <sub>16</sub>											D <sub>1</sub>	D <sub>0</sub>																		
Short Real	$10^{\pm 38}$	24 Bits	S	E <sub>7</sub>	E <sub>0</sub>	F <sub>1</sub>											F <sub>23</sub>	F <sub>0</sub> Implicit																		
Long Real	$10^{\pm 308}$	53 Bits	S	E <sub>10</sub>	E <sub>0</sub>	F <sub>1</sub>											F <sub>52</sub>	F <sub>0</sub> Implicit																		
Temporary Real	$10^{\pm 4932}$	64 Bits	S	E <sub>14</sub>	E <sub>0</sub>	F <sub>0</sub>											F <sub>63</sub>																			

Note:

Integer: I  
 Fraction: F  
 Exponent: E

Sign: S  
 BCD Digit (4 Bits): D

Packed BCD:  $(-1)^S(D_{17} \dots D_0)$   
 Real:  $(-1)^S(2^{E-BIAS})(F_0 F_1 \dots)$

Bias = 127 for Short Real  
 1023 for Long Real  
 161383 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 (ms)	
	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		Arithmetic Instructions		Processor Control Instructions	
<b>Real Transfers</b>		<b>Addition</b>		<b>FINIT/FNINIT</b> Initialize processor	
FLD	Load real	FADD	Add real	FDISI/FNDISI	Disable interrupts
FST	Store real	FADDP	Add real and pop	FENI/FNENI	Enable interrupts
FSTP	Store real and pop	FIADD	Integer add	FLDCW	Load control word
FXCH	Exchange registers	<b>Subtraction</b>		FSTCW/FNSTCW	Store control word
<b>Integer Transfers</b>		FSUB	Subtract real	FSTSW/FNSTSW	Store status word
FILD	Integer load	FSUBP	Subtract real and pop	FCLEX/FNCLEX	Clear exceptions
FIST	Integer store	FISUB	Integer subtract	FSTENV/FNSTENV	Store environment
FISTP	Integer store and pop	FSUBR	Subtract real reversed	FLDENV	Load environment
<b>Packed Decimal Transfers</b>		FSUBRP	Subtract real reversed and pop	FSAVE/FNSAVE	Save state
FBLD	Packed decimal (BCD) load	FISUBR	Integer subtract reversed	FRSTOR	Restore state
FBSTP	Packed decimal (BCD) store and pop	<b>Multiplication</b>		FINCSTP	Increment stack pointer
<b>Comparison Instructions</b>		FMUL	Multiply real	FDECSTP	Decrement stack pointer
FCOMP	Compare real	FMULP	Multiply real and pop	FFREE	Free register
FCOMP	Compare real and pop	FIMUL	Integer multiply	FNOP	No operation
FCOMP	Compare real and pop twice	<b>Division</b>		FWAIT	CPU wait
FICOM	Integer compare	FDIV	Divide real		
FICOMP	Integer compare and pop	FDIVP	Divide real and pop		
FTST	Test	FIDIV	Integer divide		
FXAM	Examine	FDIVR	Divide real reversed		
		FDIVRP	Divide real reversed and pop		
		FIDIVR	Integer divide reversed		
<b>Transcendental Instructions</b>		<b>Other Operations</b>			
FPTAN	Partial tangent	FSQRT	Square root		
FPATAN	Partial arctangent	FSCALE	Scale		
F2XM1	$2^x - 1$	FPREM	Partial remainder		
FYL2X	$Y \cdot \log_2 X$	FRNDINT	Round to integer		
FYL2XP1	$Y \cdot \log_2(X + 1)$	FTRACT	Extract exponent and significant		
		FABS	Absolute value		
		FCNS	Change sign		

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.

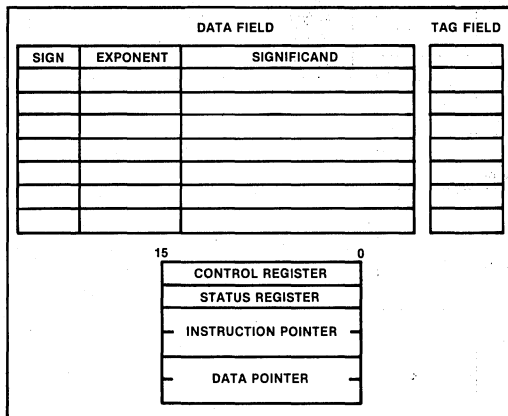


Figure 3. 8087 Register Set

### 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<sub>0</sub>-C<sub>3</sub>) 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.

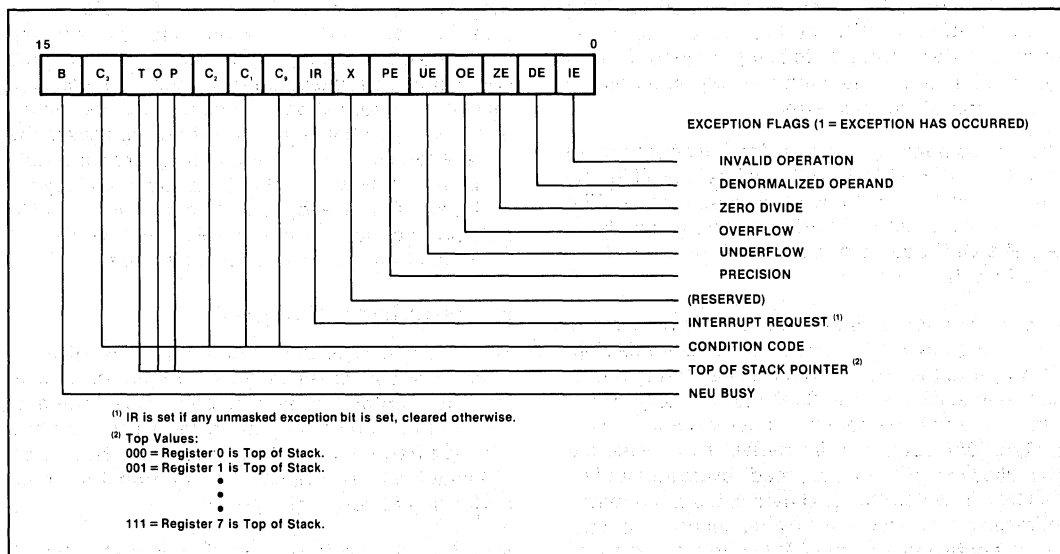


Figure 4. 8087 Status Word

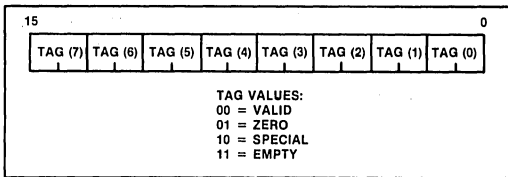


Figure 5. 8087 Tag Word

**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.

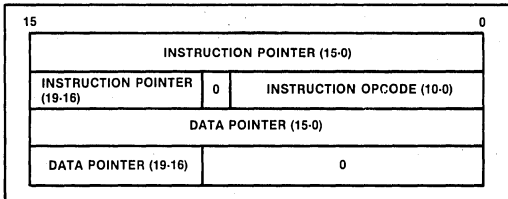


Figure 6. 8087 Instruction and Data Pointers

**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.

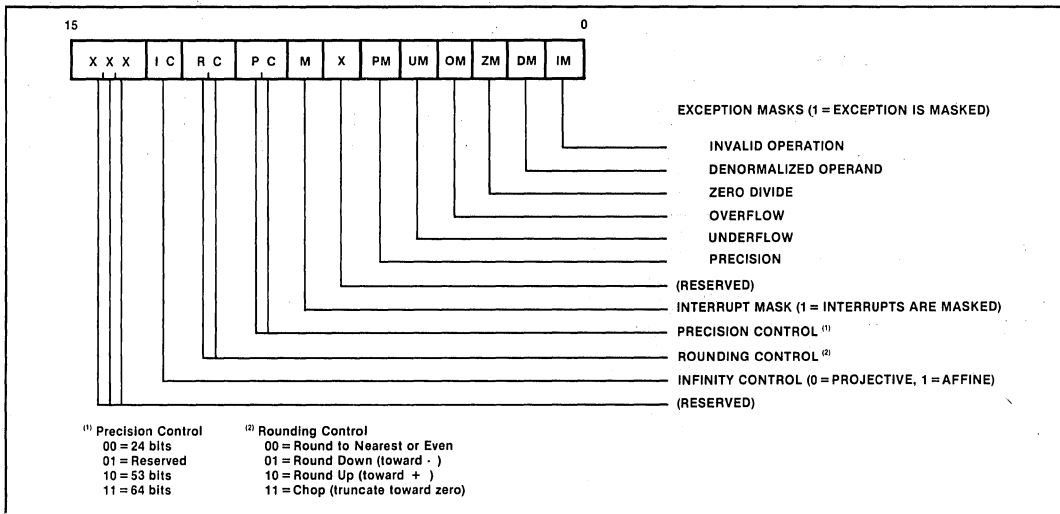


Figure 7. 8087 Control Word

**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.

**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

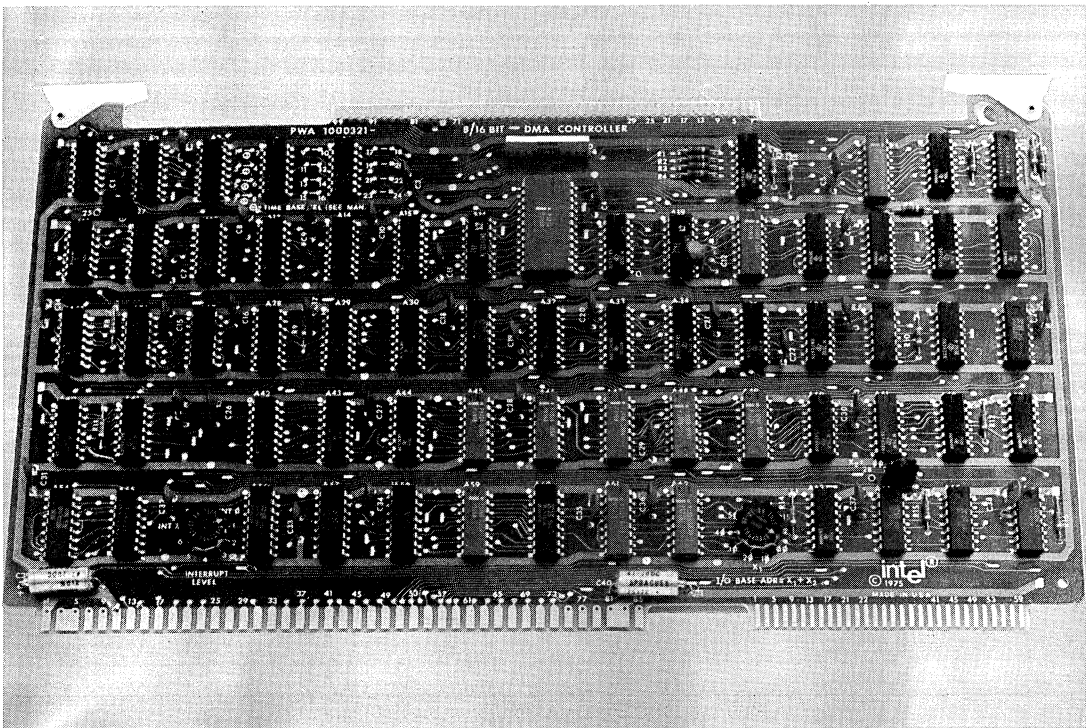
Part Number	Description
SBC 337	MULTIMODULE Numeric Data Processor



## iSBC 501 DIRECT MEMORY ACCESS CONTROLLER

- Directly compatible with Intel iSBC 80 and iSBC 86 Single Board Computers
- Block length up to 65,536 words
- Directly addresses up to 65,536 memory locations
- Transfer rate up to 1 million words per second for block transfers
- Transfers initialized via software
- Transfers data up to 330K words per second for interleaved transfers
- Software selectable/maskable interrupt operations
- Interrupt priority switch selectable

The iSBC 501 Direct Memory Access Controller is a member of Intel's complete line of iSBC OEM computer systems. The iSBC 501 interfaces directly with any iSBC single board computer-based system via the MULTIBUS interface. High speed, direct memory access control and interfacing for transfers between iSBC expansion board memory and up to 16 peripheral devices is provided.





**FUNCTIONAL DESCRIPTION**

**Transfer Capability**

Block lengths up to 65,536 bytes long may be transferred directly to or from RAM memory in iSBC systems at rates of up to 1 million words per second. The iSBC 501 16-bit addressing capability allows transfers to take place at any location within memory. It is designed to control the direct transfer of data to or from Intel iSBC memory expansion or combination memory and I/O boards. Two transfer modes of operation are included. System software is used to select the desired mode. Transfer rates up to 330K words per second may be achieved in the shared bus mode, wherein the iSBC 501 request access to the system bus for 600 ns to perform a transfer of one word to or from memory. The second mode, the override mode, establishes the DMA controller as the only master which may access the system bus during the transfer period, thereby providing rapid block transfer capability. This mode provides transfer rates up to 1 million words per second. The iSBC 80/10B single board computer may only interact with the iSBC 501 in

the shared bus mode. Either mode may be used with the other iSBC single board computers. Four timing strobes are provided for the control of data input transfers and four timing strobes are provided for output transfer operations. Strokes are initiated and selected via system software, and strobe pulses are jumper selectable to 100, 200, 400, 800, or 1600 ns widths.

**Interrupt Requests**

Interrupt requests originating from the DMA controller are software maskable, active-low, and switch selectable to any one of eight priority levels. User selected DMA interrupt requests may originate automatically upon completion of a transfer operation, from an external DMA device, or from a software command to the DMA controller (for system testing purposes).

**Peripheral Interface**

A 4-bit tag register is provided which may be used as a device select port to provide selection for four (up to 16 with external decoding) high speed peripheral devices interfacing through the iSBC 501.

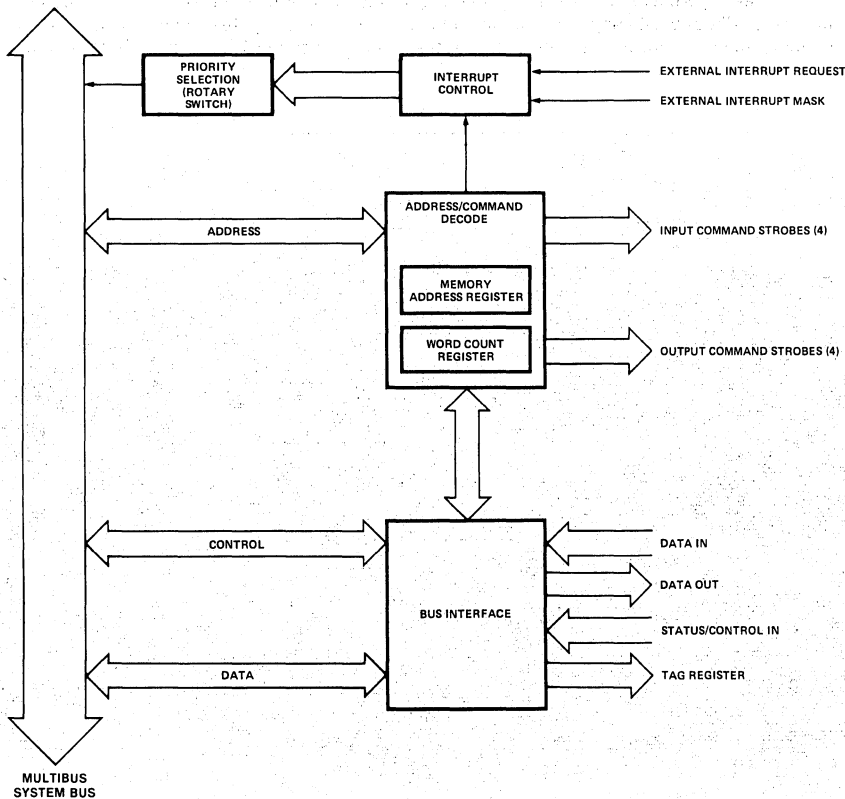


Figure 1. iSBC 501 DMA Controller Block Diagram

## SPECIFICATIONS

### Word Size

8 bits

### Block Size

65,536 words, max

### Address Capability

65,536 words

### Transfer Rates

Mode	Transfer Rates (K bytes/sec) <sup>1</sup>			
	Memory Read Operations		Memory Write Operations	
	Typical	Worst Case <sup>3</sup>	Typical	Worst Case <sup>3</sup>
Shared bus, CPU halted <sup>2</sup>	330	270	330	270
Shared bus, CPU executing code <sup>2</sup>	180	160	180	160
Override	1000	660	1000	660

#### Notes:

- Transfer rates given are to and from RAM memory on iSBC 108A combination memory and I/O board.
- Shared bus mode may be used with Intel iSBC 80/10B.
- Assumes every DMA transfer must wait for RAM refresh cycle to be completed, worst case memory cycle times.

### Interrupts

Interrupt requests originating from the DMA controller are software maskable, active-low, and switch selectable to any one of eight priority levels. User selectable DMA interrupt requests may originate automatically upon completion of a transfer operation, from an external DMA device, or from a software command to the DMA controller (for system testing purposes).

### Key Registers

**Control Register (6 bits)** — The contents of the control register specify the busy status of the DMA board, the type of operation to be performed (transfer or non-transfer), the transfer direction (to or from memory), the interrupt condition (enabled or disabled), and the means by which the DMA board is using the system bus (shared mode or override mode).

**Memory Address Register (16 bits)** — Contains the address of the next memory location to be accessed by the iSBC 501. Loaded from the CPU, prior to a transfer operation, with the address of the first memory location to be accessed. The address is gated onto the system address bus during each transfer, and incremented by one for each word transferred.

**Length Register (16 bits)** — Contents of this register specify the total number of words to be transferred. This word count is decremented by one after each word is transferred. The transfer stops when the word count equals zero.

**Tag Register (4 bits)** — The contents of the tag register are used as control/select lines to the external peripheral devices being interfaced by the iSBC 501 (e.g., as the “go” command line to each of four devices), or the tag register outputs may be used with external decoding to expand the maximum number of DMA peripherals to 16.

**Status Register (8-bits)** — Provides 4 bits of DMA controller status: software interrupt, memory read/write operation requested, external/end-of-transfer interrupt, and DMA controller busy. The status register also provides four status/control bits directly from user peripheral devices.

### Address Selection

iSBC 501 registers are located in a jumper selectable block starting at any 16-word boundary in the I/O address space.

### Register Locations

Address <sup>1</sup>	I/O Operation	Function
X0	Output	Output strobe 0
X1	Output	Output strobe 1
X2	Output	Output strobe 2
X3	Output	Output strobe 3
X4	Output	Output tag strobe
X8	Output	Set interrupt
X9	Output	Reset interrupt
XA	Output	Load control register
XB	Output	Load tag register
XC	Output	Load LSB length register
XD	Output	Load MSB length register
XE	Output	Load LSB memory address register
XF	Output	Load MSB memory address register
X0	Input	Input command strobe 0
X1	Input	Input command strobe 1
X2	Input	Input command strobe 2
X3	Input	Input command strobe 3
X4	Input	Read LSB length register
X5	Input	Read MSB length register
X6	Input	Read DMA status
X7	Input	Invalid command

#### Note

- X is any hex digit, assigned by jumpers.

### Connectors

Interface	Double-Sided Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	Viking 3KH43/9AMK12
I/O	100	0.100	Intel MDS 990 Viking 3VH50/1JN5

### Interface Characteristics

**I/O Line Driver Sink Current** — 48 mA

**I/O Line Terminator Load** — 150Ω pullup

**Input** — Data positive relative to data bus

**Output** — Data positive relative to data bus

**Output Strobes** — Jumper selectable to 100, 200, 400, 800, or 1600 ns pulse widths.

All I/O interface data and control signals are TTL compatible and MULTIBUS interface compatible.

# iSBC 501

## 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

$V_{CC} = 5V \pm 5\%$

$I_{CC} = 3.35A \text{ max}; 2.70A \text{ typ}$

## Environmental Characteristics

Operating Temperature — 0°C to 55°C

## Equipment Supplied

iSBC 501 DMA Controller Board

## Reference Manuals

9800294A — iSBC 501 Hardware Reference Manual (NOT SUPPLIED)

iSBC 501 Schematic (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

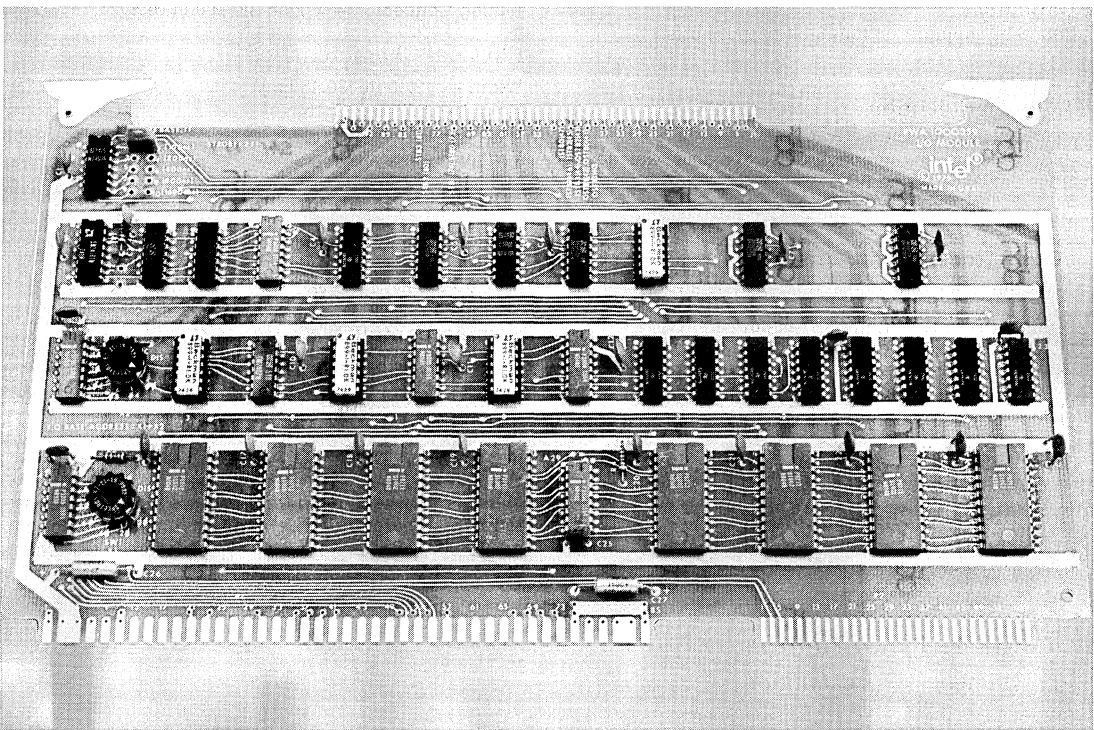
Part Number	Description
SBC 501	Direct Memory Access Controller



## iSBC 508 I/O EXPANSION BOARD

- iSBC I/O expansion via direct MULTIBUS interface
- Four 8-bit terminated input ports
- Four 8-bit output ports with buffered TTL drivers
- Selectable latched or unlatched input ports
- Latched outputs with selectable width strobes
- Switch selectable I/O port addresses

The iSBC 508 I/O Expansion Board is a member of Intel's complete line of iSBC memory and I/O expansion boards. The iSBC 508 interfaces directly to any iSBC single board computer via the system bus to expand input and output port capacity. Four 8-bit terminated input ports are contained on the board. Data is gated into the port while the strobe is present and latched if the strobe is removed. The iSBC 508 contains four 8-bit output ports. All output lines are driven by TTL level buffer drivers residing on the board. Output data is latched. A strobe signal of jumper selectable width is sent to the peripheral device during an output operation. Address selection is accomplished using two resident rotary switches to select one of 64 unique base addresses for all input and output ports. The board operates with a single +5V power supply.



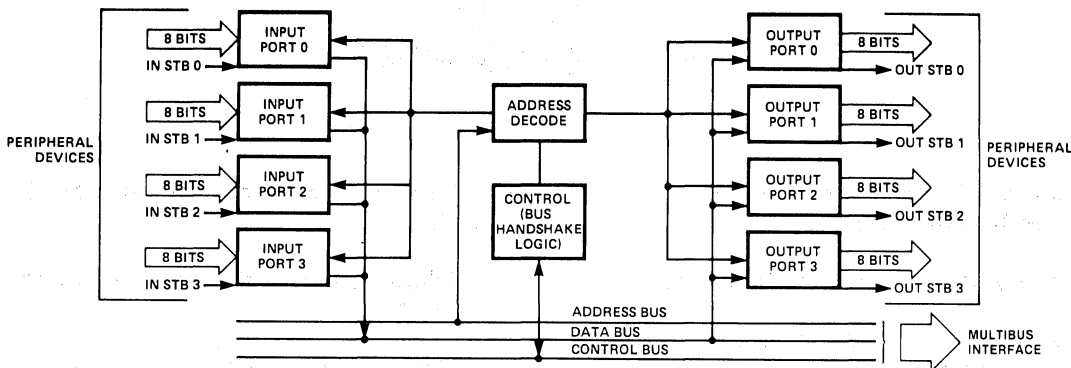


Figure 1. I/SBC 508 I/O Expansion Board Block Diagram

## SPECIFICATIONS

### Word Size

8 bits

### Capacity

Four 8-bit input ports; four 8-bit output ports.

### Address Selection

Input and output ports are accessed as four sequential addresses starting in one of 64 switch selectable locations between 00 and FC<sub>16</sub>.

### I/O Interface Characteristics

All I/O interface data and control signals are TTL levels.

I/O Line Driver Sink Current — 48 mA

I/O Line Terminator Load — 1 kΩ pullup

Inputs — Data positive relative to data bus

Outputs — Data positive relative to data bus

Output Strobe — Jumper selectable to 100, 200, 400, 800, or 1600-ns pulse widths

### Bus Interface Characteristics

All address, data, and control signals are TTL compatible.

### Connectors

**Bus Edge Connector** — 86-pin double-sized PC edge connector with 0.156-in. contact centers

**Mating Connector** — Control Data VPB01E43A00A1

**I/O Edge Connector** — 100-pin double-sided PC edge connector with 0.1 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 (415.2 gm)

### Electrical Characteristics

#### DC Power Requirements

V<sub>CC</sub> = +5V ± 5%

I<sub>CC</sub> = 2.6A max, 1.9A typ

### Environmental Characteristics

Operating Temperature — 0°C to 55°C

### Reference Manuals

9800278 — I/SBC 508 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

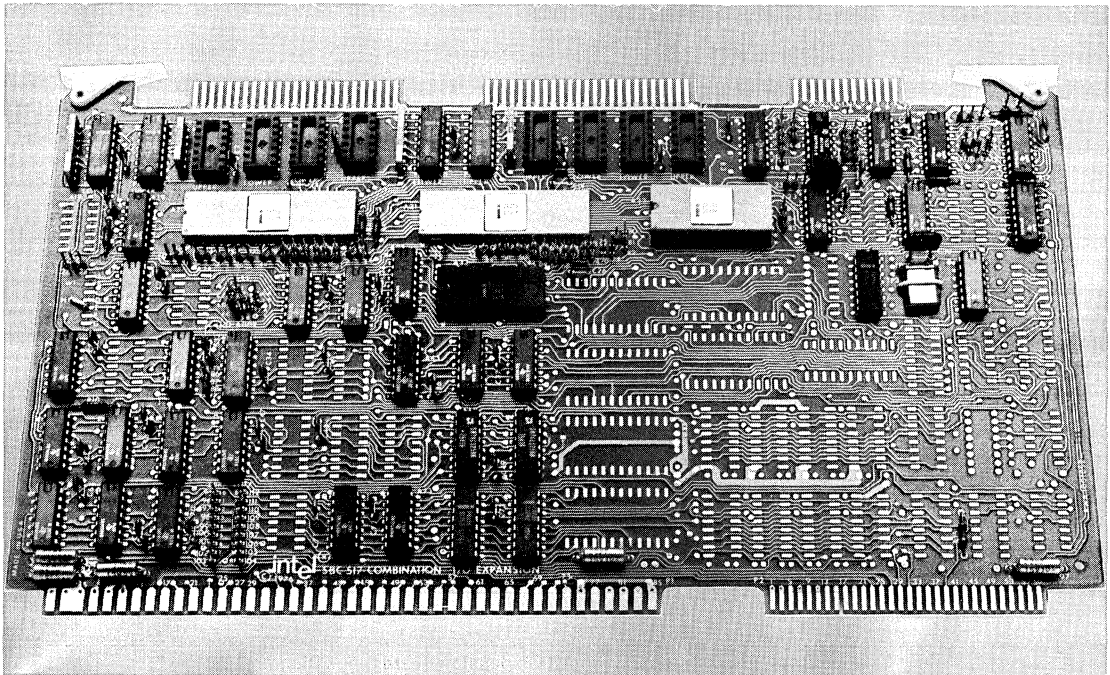
Part Number	Description
SBC 508	I/O Expansion Board



## 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. The iSBC 530 Teletypewriter Adapter provides an optically isolated interface for those systems requiring a 20 mA current loop. The iSBC 530 may be used to interface the iSBC 517 Combination I/O Expansion Board to teletypewriters and other 20 mA current loop equipment.

**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

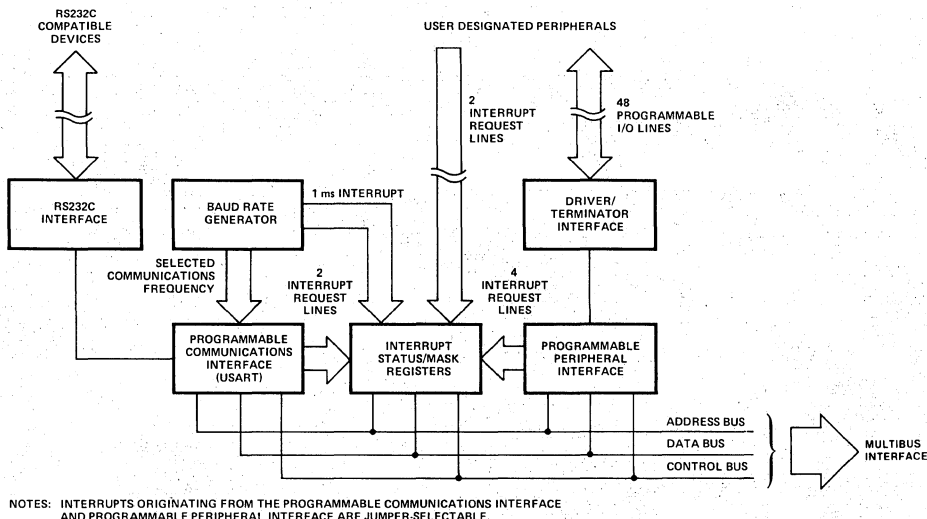


Figure 1. iSBC 517 Combination I/O Expansion Board Block Diagram

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 <sup>1</sup>
	4	X		X			X <sup>1</sup>
4	8	X	X	X	X	X	
5	8	X	X	X	X		
6	4	X		X			X <sup>2</sup>
	4	X		X			X <sup>2</sup>

**Notes**

- 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.
- 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 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



# iSBC 517

## 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 <sup>1</sup>	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:

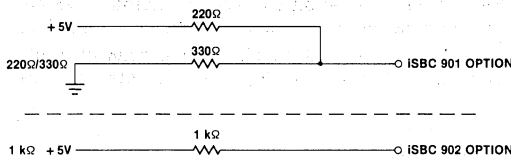
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

### Note

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

Ports 1 and 4 have 25 mA totem-pole drivers and 1 k $\Omega$  terminators.

**I/O Terminators** — 220 $\Omega$ /330 $\Omega$  divider or 1 k $\Omega$  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

$V_{CC} = +5V \pm 5\%$

$V_{DD} = +12V \pm 5\%$

$V_{AA} = -12V \pm 5\%$

$I_{CC} = 2.4 \text{ mA max}$

$I_{DD} = 40 \text{ mA max}$

$I_{AA} = 60 \text{ mA max}$

### Note

Does not include power required for optional I/O drivers and I/O terminators. With eight 220 $\Omega$ /330 $\Omega$  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**

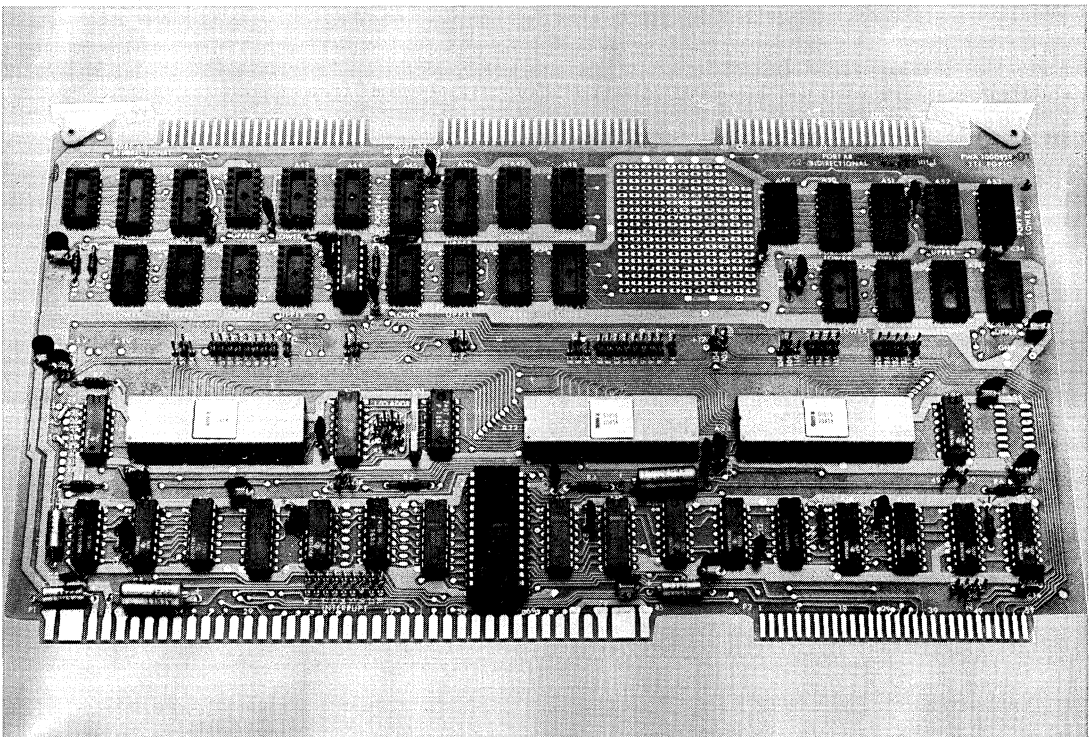
SBC 517 Combination I/O Expansion Board



## 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 iSBC 80 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 80 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				Bidirectional	Control
		Unidirectional					
		Input		Output			
		Unlatched	Latched & Strobed	Latched	Latched & Strobed		
1,4,7	8	X	X	X	X		
2,5,8	8	X	X	X	X		
3,6,9	4	X		X		X <sup>1,2,3</sup>	
	4	X		X		X <sup>1,2,3</sup>	

#### 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.

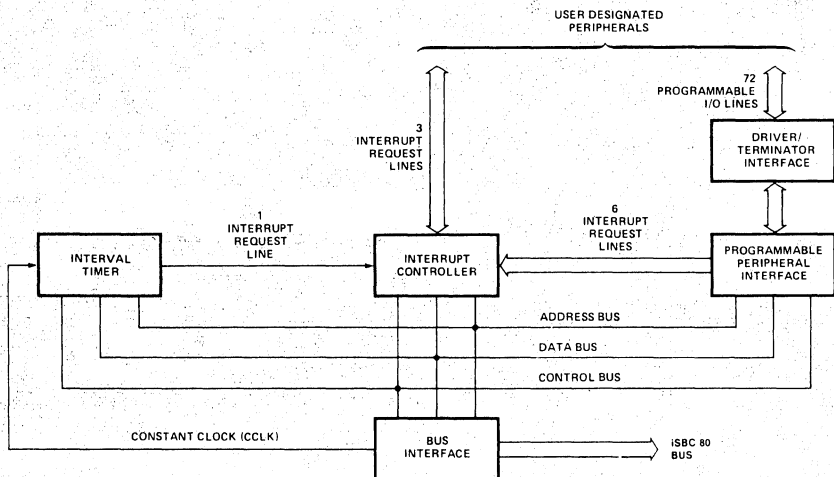


Figure 1. iSBC 519 Programmable I/O Expansion Board Block Diagram

**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				8255 No. 2 Control				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<sup>1</sup>.

**Timing Intervals** — 500, 1,000, 2,000, and 4,000 ms ± 1%; jumper selectable<sup>2</sup>.

**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 <sup>1</sup>	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:

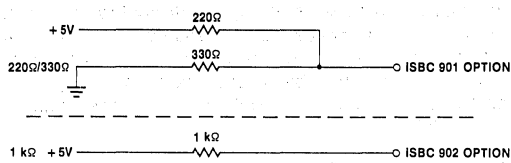
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

**Note**

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

## ISBC 519

### 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 <sup>1</sup>	With Termination <sup>2</sup>
$V_{CC} = +5V \pm 5\%$	$I_{CC} = 1.5A \text{ max}$	3.5A max

#### Note

- Does not include power required for optional I/O drivers and I/O terminators.
- 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

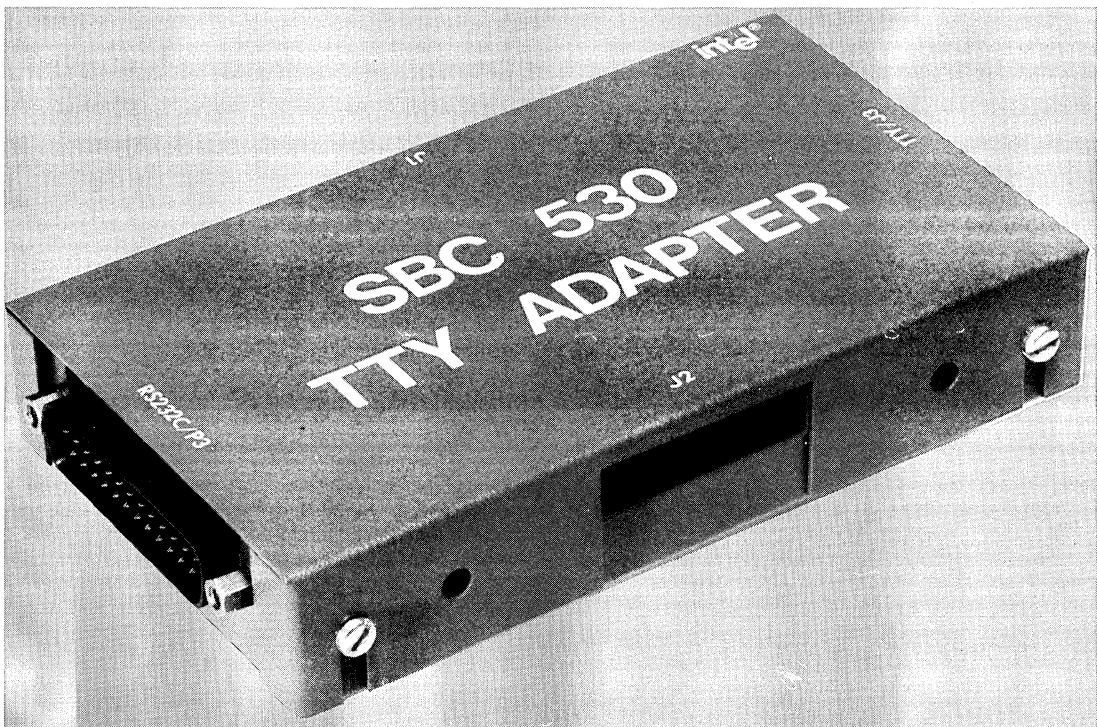
Part Number	Description
SBC 519	Programmable I/O Expansion Board



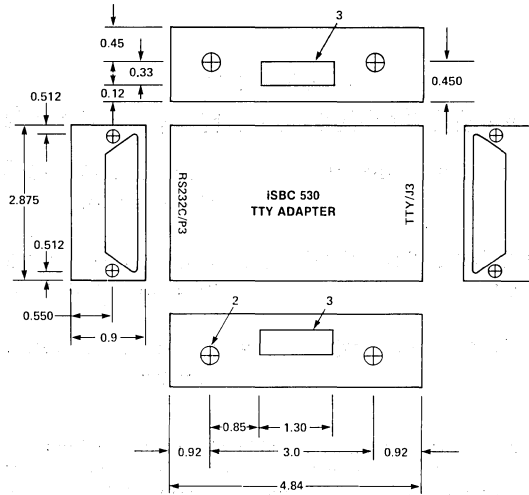
## iSBC 530 TELETYPEWRITER ADAPTER

- Compact, easily mounted package with standard connectors
- General purpose RS232C to 20 mA current loop interface
- Compatibility with iSBC 80 and iSBC 86 Single Board Computers
- Jumper selectable RS232C data set or data terminal configuration
- Compatibility with iSBC 80 combination boards
- Interface opto-isolator for high noise immunity

The iSBC 530 Teletypewriter Adapter provides a compact and flexible means for interfacing Intel iSBC 80 and iSBC 86 Single Board Computers, iSBC combination memory and I/O expansion boards, and most RS232C compatible equipment to teletypewriters and other 20 mA current loop equipment. The iSBC 530 converts RS232C signal levels to an optically isolated 20 mA current loop interface. The iSBC 530 provides signal translation for transmitted data (TxD), received data (RxD), and a teletypewriter paper tape reader relay. The RS232C interfaces are jumper selectable, and may be configured to accept signals from an RS232C data terminal or data set. Threaded holes have been incorporated in the iSBC 530 for ease in system chassis design and multiple units may be mounted together to support multiple serial channels. The units are mountable in any of three planes. When used with an iSBC Single Board Computer, power is provided to the iSBC 530 directly through its RS232C connector. Power may also be provided through either of two auxiliary power connectors for standard current loop interfacing. The noise immunity benefits of total inter-system power isolation may be achieved through the use of both auxiliary power connectors on the iSBC 530.



# iSBC 530



- NOTES: 1. ALL DIMENSIONS IN INCHES.  
 2. ALL FOUR MOUNTING HOLES THREADED FOR 6-32 MACHINE SCREWS.  
 3. CUTOUTS FOR AUXILIARY POWER CONNECTORS.

Figure 1. iSBC 530 Dimensions

## SPECIFICATIONS

### Interface Characteristics

#### RS232C Side

RS232C signal levels in/out<sup>1</sup>

#### TTY Side

20 mA optically isolated current loop

#### Note

1. RS232C data set ready line controls 20 mA paper tape reader relay driver line.

### Mating Connectors

RS232C	Cinch	DB-25S	
	ITT Cannon	DB-25S	
20 mA (TTY)	Cinch	DB-25P	
	ITT Cannon	DB-25P	
Auxiliary power	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 Characteristics

Width — 2.876 in. max (7.31 cm)

Height — 4.850 in. max (12.32 cm)

Depth — 0.920 in. max (2.34 cm)

Weight — 9 oz (255.4 gm)

### Electrical Characteristics

Power connectors for ground, +12V and -12V, are jumper selectable. Power may be provided via 25-pin RS232C connector or via two separate auxiliary power connectors. Auxiliary connectors allow total power system isolation at iSBC 530 opto-coupler interface.

#### Power Requirements

$V_{DD} = +12V \pm 5\%$

$V_{AA} = -12V \pm 5\%$

$I_{DD} = 98 \text{ mA max}$

$I_{AA} = 98 \text{ mA max}$

### Environmental Characteristics

Operating Temperature — 0°C to 55°C

### Reference Manuals

None 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

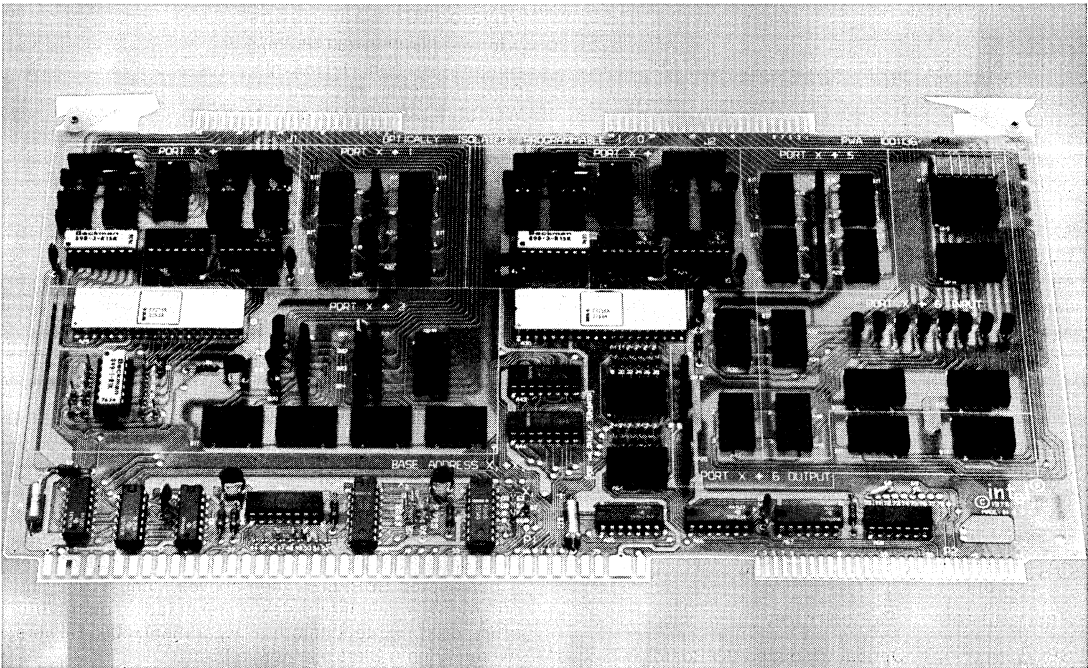
SBC 530 Teletypewriter adapter



## 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
- Voltage/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.





# ISBC 556

**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						

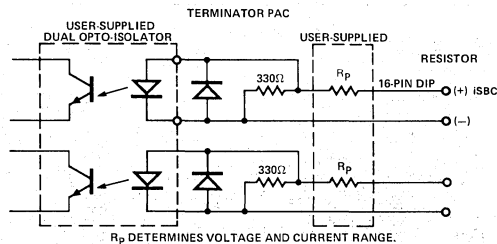
## SPECIFICATIONS

### 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)

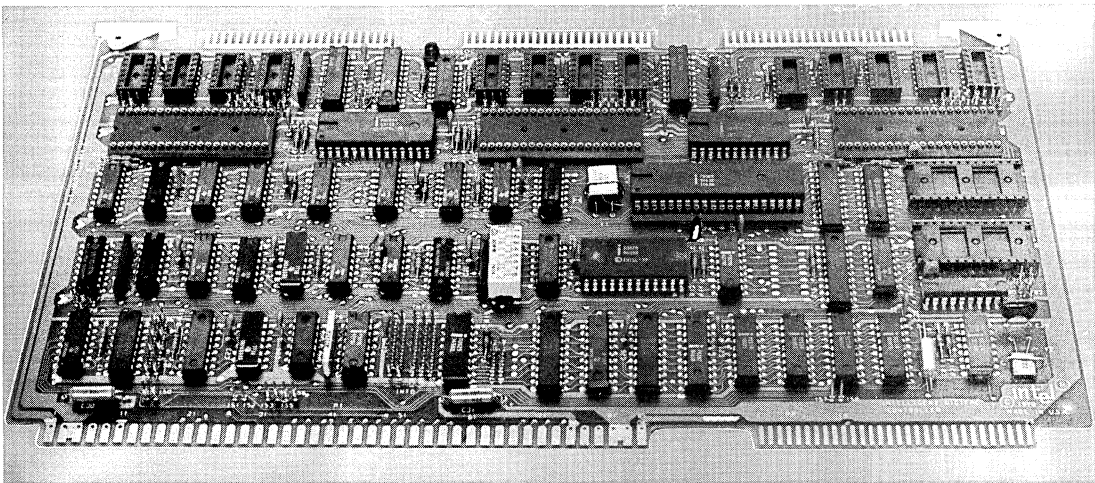
Reference manuals are shipped with each product only if designated SUPPLIED (see above). Manuals may be ordered from any Intel sales representative, distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, California 95051.



## 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-alone 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.

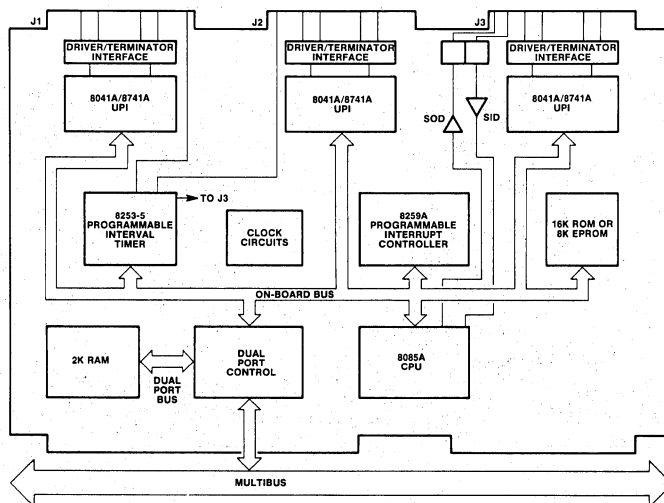


Figure 1. iSBC 569 Intelligent Digital Controller Block Diagram

## 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 IIBC 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 IIBC 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 IIBC 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 IIBC 569 boards may be included in an

expanded system using one IIBC 569 Intelligent Digital Controller as the system master and additional controllers as intelligent slaves.

### Intelligent Slave Programming

When used as an intelligent slave, the IIBC 569 controller appears as an additional RAM memory module. System bus masters communicate with the IIBC 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 IIBC 569 RAM. This interrupt provides a means for the master CPU to notify the IIBC 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 IIBC 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 IIBC 569 Intelligent Digital Controller board is supported by the Intellec® Micro-computer 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 IIBC 569 board. A unique in-circuit emulator (ICE-85™) option provides the capability of developing and debugging software directly on the IIBC 569 board. This greatly simplifies the design, development, and debug of IIBC 569 system software.

## SPECIFICATIONS

### 8085A CPU

**Word Size** — 8, 16 or 24 bits

**Cycle Time** — 1.30  $\mu$ sec  $\pm$  .1% for fastest executable instruction; i.e., four clock cycles.

**Clock Rate** — 3.07 MHz  $\pm$  .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); 0-0FFF (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 <sup>1</sup>	Max <sup>1</sup>
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

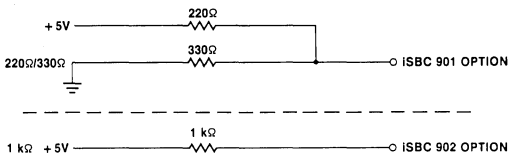
**Line Drivers and Terminators**

**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

Dear Mr. [Name],

I am writing to you regarding the [Topic] that we discussed in our meeting on [Date].

The information you provided is being reviewed, and we will contact you again once a decision has been reached.

We appreciate your patience and understanding during this process.

Yours faithfully,  
[Signature]

[Name]  
[Title]  
[Company Name]  
[Address]  
[City, State, Zip]

[Phone Number]  
[Email Address]

[Additional Information]

[Closing Remarks]

[Signature]

[Name]  
[Title]  
[Company Name]  
[Address]  
[City, State, Zip]

[Phone Number]  
[Email Address]

[Additional Information]

[Closing Remarks]

Dear Mr. [Name],

I am writing to you regarding the [Topic] that we discussed in our meeting on [Date].

The information you provided is being reviewed, and we will contact you again once a decision has been reached.

We appreciate your patience and understanding during this process.

Yours faithfully,  
[Signature]

[Name]  
[Title]  
[Company Name]  
[Address]  
[City, State, Zip]

[Phone Number]  
[Email Address]

[Additional Information]

[Closing Remarks]

[Signature]

[Name]  
[Title]  
[Company Name]  
[Address]  
[City, State, Zip]

[Phone Number]  
[Email Address]

[Additional Information]

[Closing Remarks]



*Communication  
Controllers*

---

**8**



DEPARTMENT OF CHEMISTRY  
LABORATORY OF ORGANIC CHEMISTRY

RESEARCH REPORT  
NO. 1000

BY

ROBERT H. WOODWARD

AND

ROBERT B. WOODWARD

AND

ROBERT B. WOODWARD

AND

ROBERT B. WOODWARD

AND

ROBERT B. WOODWARD

AND

ROBERT B. WOODWARD

AND

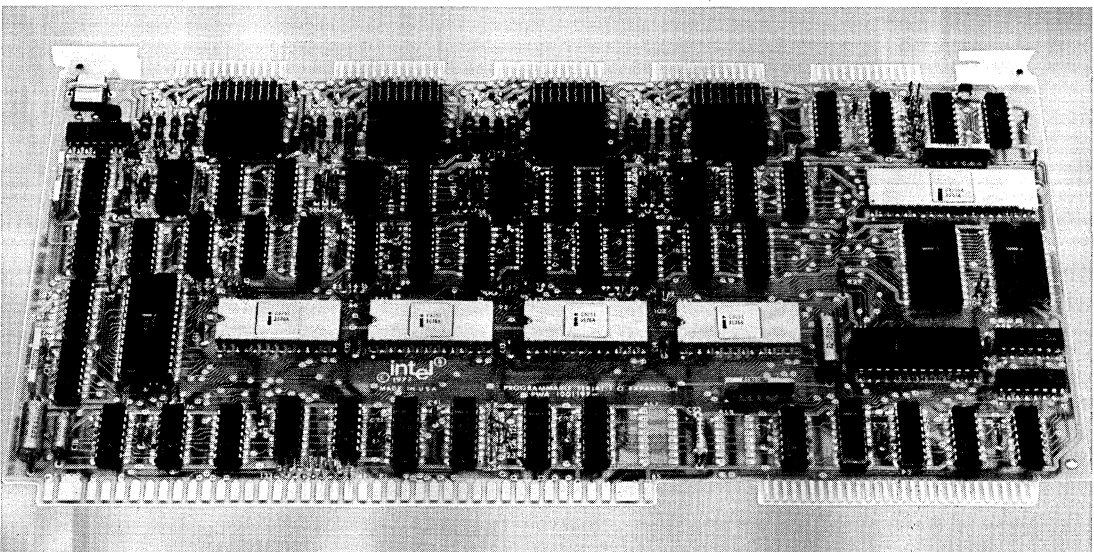
ROBERT B. WOODWARD



## iSBC 534 or (pSBC 534\*) FOUR CHANNEL COMMUNICATION EXPANSION BOARD

- iSBC 80 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 iSBC 80 memory and I/O expansion boards. The iSBC 534 interfaces directly to any iSBC 80 single board computer via the system bus 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 iSBC 80 and System 80 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 8251 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 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. Each 8251 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 8259 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

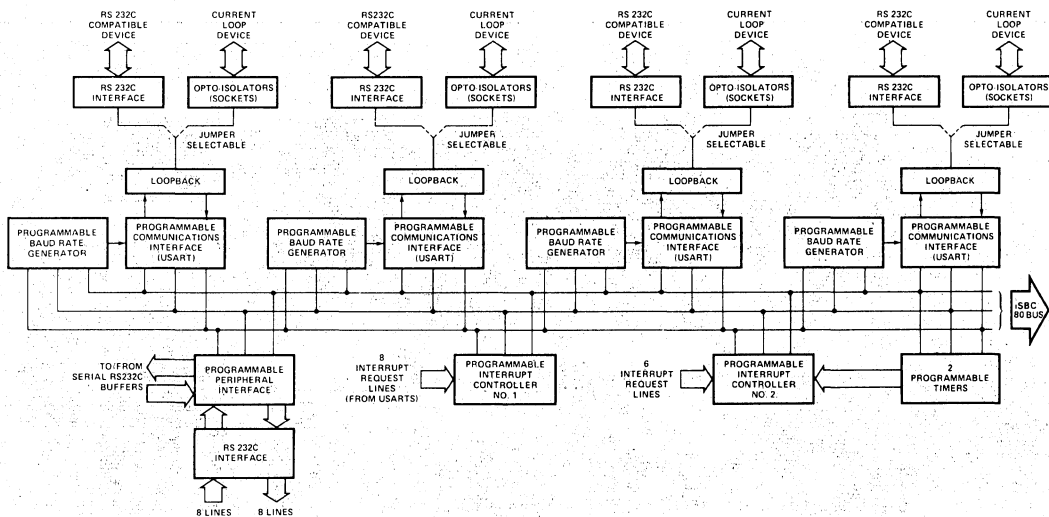


Figure 1. ISBC 534 Four Channel Communications Expansion Board Block Diagram

output line may be jumper selected to drive any of the nine interrupt lines on the iSBC 80 bus.

**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 iSBC 80 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.

**Systems Compatibility**

The iSBC 534 provides 16 RS232C buffered parallel I/O lines implemented utilizing an Intel 8255 programmable

**Table 3. Interrupt Assignments**

Interrupt Request Line	PIC 0	PIC 1
0	PORT 0 R <sub>X</sub> RDY	PIT 1 counter 1
1	PORT 0 T <sub>X</sub> RDY	PIT 2 counter 2
2	PORT 1 R <sub>X</sub> RDY	Ring indicator (all ports)
3	PORT 1 T <sub>X</sub> RDY	Present next digit
4	PORT 2 R <sub>X</sub> RDY	Carrier detect port 0
5	PORT 2 T <sub>X</sub> RDY	Carrier detect port 1
6	PORT 3 R <sub>X</sub> RDY	Carrier detect port 2
7	PORT 3 T <sub>X</sub> RDY	Carrier detect port 3

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 iSBC 80 and System 80-based systems 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 8251 USART, the Intel 8253 Programmable Interval Timer, the Intel 8255 Programmable Peripheral Interface, and the Intel 8259 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<sup>1</sup>**

Frequency <sup>2</sup> (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:**

- 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).
- 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 ± 10% (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 iSBC 80 bus compatible

## ISBC 534

### 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 3KH43/9AMK12	N/A
Serial and parallel I/O	26	0.1	3M 3462-000 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 <sup>1</sup>
V <sub>CC</sub> = +5V	1.9 A, max	1.9 A, max
V <sub>DD</sub> = +12V	275 mA, max	420 mA, max
V <sub>AA</sub> = -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

**9800450-12** — 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

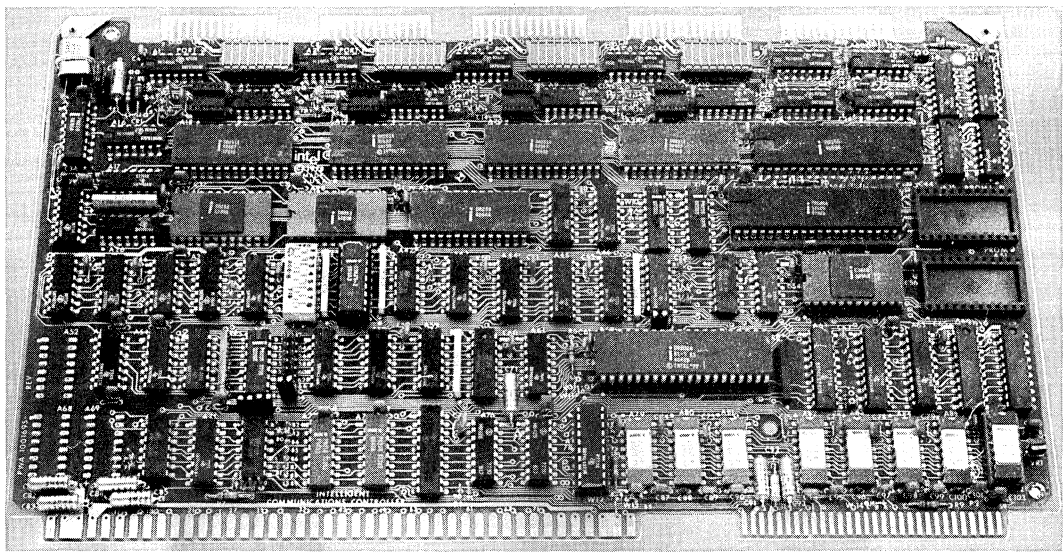
Part Number	Description
SBC 534	Four Channel Communication Expansion Board



## 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 Micro-processor 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 x 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, 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-

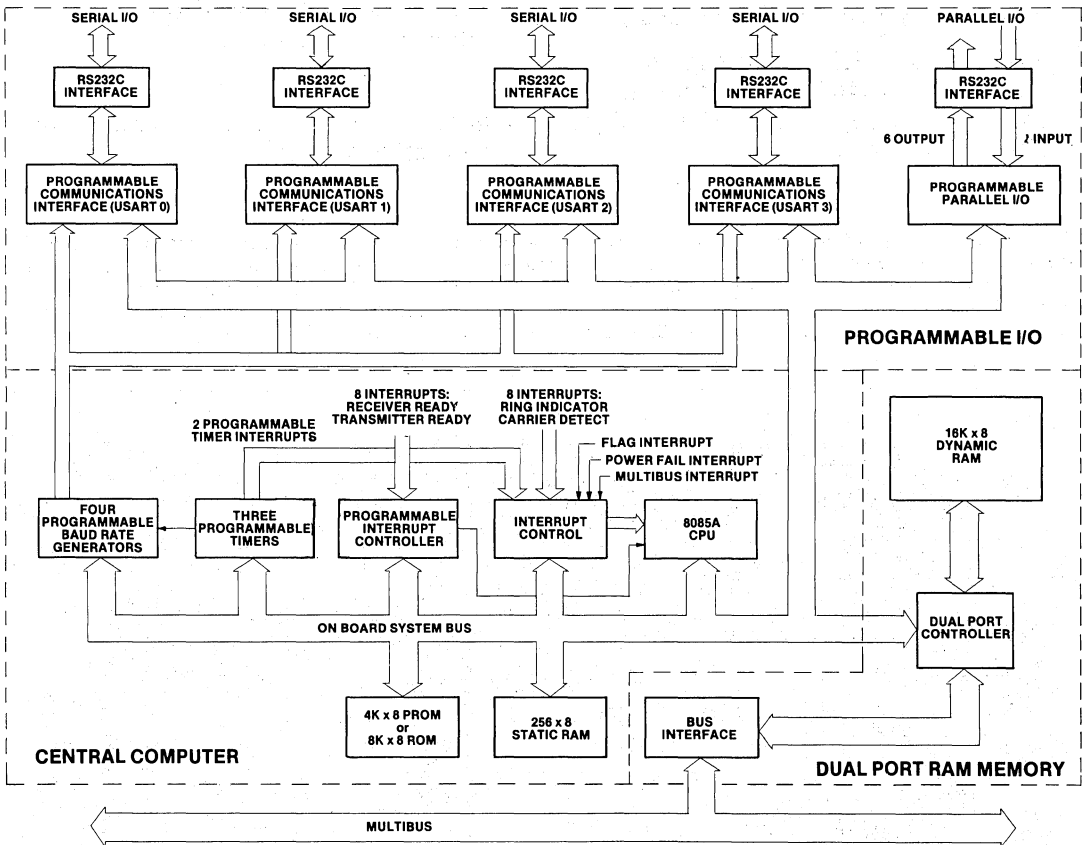


Figure 1. iSBC 544 Intelligent Communications Controller Block Diagram

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/counters 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 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. 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. An optional iSBC 530 teletypewriter adapter provides an optically isolated interface for those systems requiring a 20mA current loop.

### 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 sub-routine 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 Intel 2316E 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 address 7F00<sub>H</sub>-7FFF<sub>H</sub>.



**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 PIT 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 8259 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 accessible 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 Baud rate clocked to the USARTs.	8253 BDGO-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 <sub>H</sub>
8253 TINT1	RST 7.5	3C <sub>H</sub>
8255 TINT0		
Ring Indicator (1) Carrier Detect	RST 6.5	34 <sub>H</sub>
Flag Interrupt	RST 5.5	34 <sub>H</sub>
INT0/-INT7/ (1 of 8)		
RXRDY0 TXRDY0 RXRDY1 TXRDY1 RXRDY 2 TXRDY2 RXRDY3 TXRDY3	INTR	Program- mable
		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.

**8259 Interrupts** — Eight interrupt sources signaling transmitter and receiver ready from the four USARTs are channeled directly to the Intel 8259 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 addresses 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 any time 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 locator, 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) <sup>1</sup> (Software Selectable)	Baud Rate (Hz) <sup>2</sup>	
	Synchronous	Asynchronous
307.2		+ 16 + 64 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	---

#### 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 PROM.
- 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 Dynamic 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 2316E 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) EA (Port B) EB (Port C)	E8

### Interrupts

#### Addresses for 8259 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 (MSB) TINT0
- ED Timer (LSB) 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	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
	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 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

**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**

<b>Width:</b>	30.48 cm (12.00 inches)
<b>Depth:</b>	17.15 cm (6.75 inches)
<b>Thickness:</b>	1.27 cm (0.50 inch)
<b>Weight:</b>	3.97 gm (14 ounces)

**Electrical Characteristics**

**DC Power Requirements**

Configuration	Current Requirements			
	V <sub>CC</sub> = +5V ± 5% (max)	V <sub>DD</sub> = ± 12V ± 5% (max)	V <sub>BB</sub> = - 5V(3) ± 5% (max)	V <sub>AA</sub> = - 12V ± 5% (max)
With 4K EPROM (using 2716)	I <sub>CC</sub> = 3.4 max max	I <sub>DD</sub> = 350mA max	I <sub>BB</sub> = 5mA max	I <sub>AA</sub> = 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<sub>BB</sub> is normally derived on-board from V<sub>AA</sub>, eliminating the need for a V<sub>BB</sub> supply. If it is desired to supply V<sub>BB</sub> 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

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

...the ...  
...the ...  
...the ...

---

*Analog I/O Expansion  
and Signal  
Conditioning Boards*

---

**9**

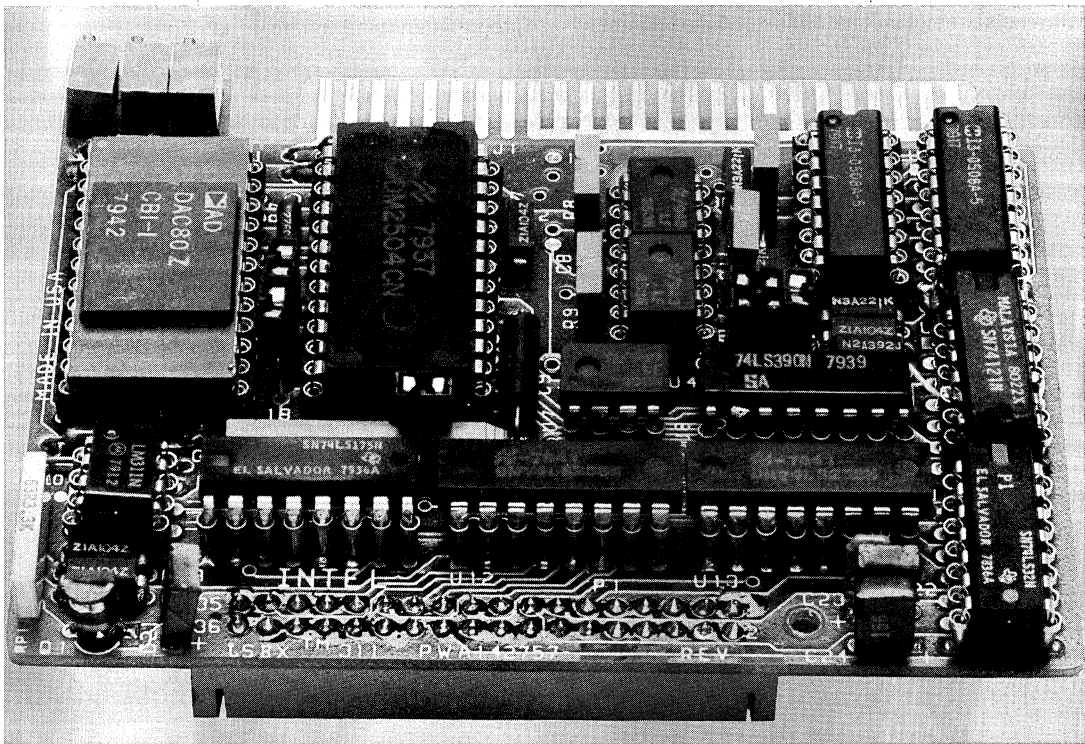




## ISBX 311 ANALOG INPUT MULTIMODULE BOARD

- 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.

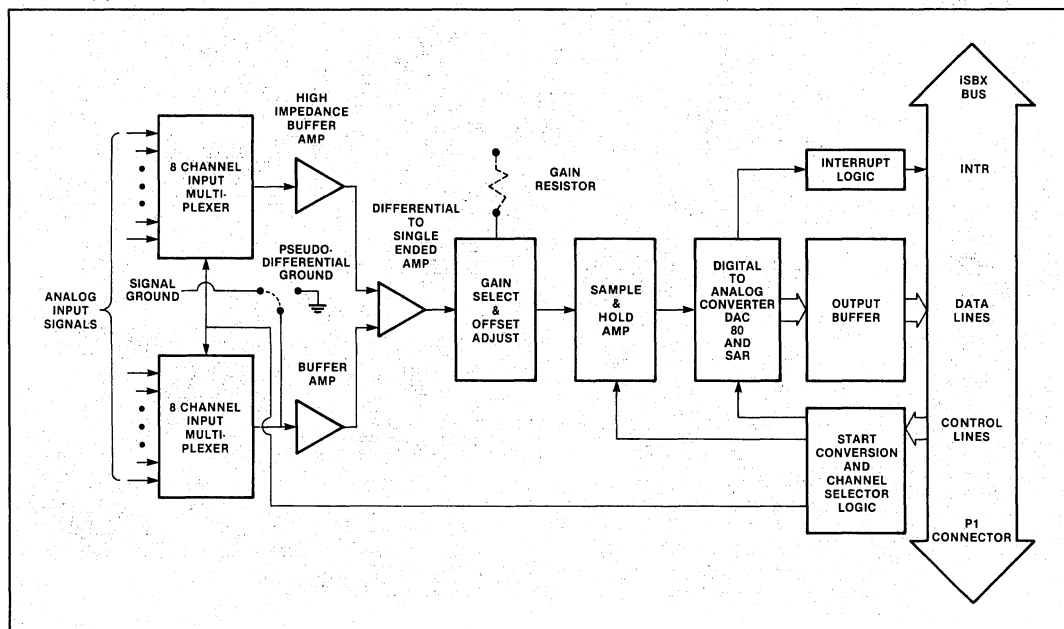


Figure 1. ISBX 311 Analog Input MULTIMODULE Board

# iSBX 311

## Accuracy

High quality components are used to achieve 12 bits resolution and accuracy of .035% full scale range  $\pm 1/2$  LSB. Offset and gain are adjustable to  $\pm 0.024\%$  FSR  $\pm 1/2$  LSB accuracy at any fixed temperature between 0°C (gain = 1). See specifications for other gain accuracies.

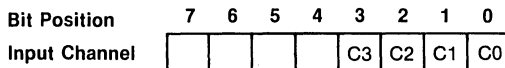
## Gain

To allow sampling of millivolt level signals such as strain gauges and thermocouples, gain is made configurable via user inserted gain resistors up to  $250 \times$  (20 millivolts, full scale input range). User can select any other gain range from 1 to 250 to match his application.

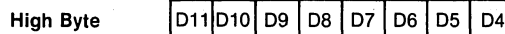
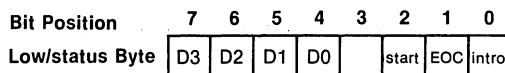
## OPERATIONAL DESCRIPTION

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.



**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/).



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 Intro/, 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; gain about 250 not recommended.

**Resolution** — 12 bits over full scale range (1.22 mv at 0-5 v, 5  $\mu$ v at 0-20 mv)

### Accuracy —

Gain	Accuracy at 25°C	Accuracy at 0° to 60°C
1	$\pm 0.035\% \pm 1/2$ LSB	$\pm 0.23\% \pm 1/2$ LSB
5	$\pm 0.035\% \pm 1/2$ LSB	$\pm 0.28\% \pm 1/2$ LSB
50	$\pm 0.035\% \pm 1/2$ LSB	$\pm 0.76\% \pm 1/2$ LSB
250	$\pm 0.035\% \pm 1/2$ LSB	$\pm 2.70\% \pm 1/2$ LSB

### NOTE:

Figures are in percent of full scale reading. At any fixed temperature between 0° and 60°C, the accuracy is adjustable to  $\pm 0.035\%$  of full scale.

**Dynamic Error** —  $\pm 0.015\%$  FSR for transitions

**Gain TC (at Gain = 1)** — 54 PPM per degree centigrade.

# ISBX 311

**Offset TC (in percent of FCR/°C) — 0.0016%.**

Gain	Offset <sup>1</sup>
1	0.0016%
5	0.0022%
50	0.016%
250	0.08%

<sup>1</sup>Offset is measured in PPM/°C with user-supplied 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	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 — 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**

**Depth — 9.40 cm (3.7 inches)**

**Weight — 68.05 gm (2.4 ounces)**

## Electrical Characteristics (from ISBX connector)

$V_{CC} = \pm 5$  volts ( $\pm 0.25V$ ),  $I_{CC} = 250$  mAmax

$V_{DD} = +12$  volts ( $\pm 0.6V$ ),  $I_{DD} = 50$  mAmax

$V_{SS} = -12$  volts ( $\pm 0.6V$ ),  $I_{SS} = 55$  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

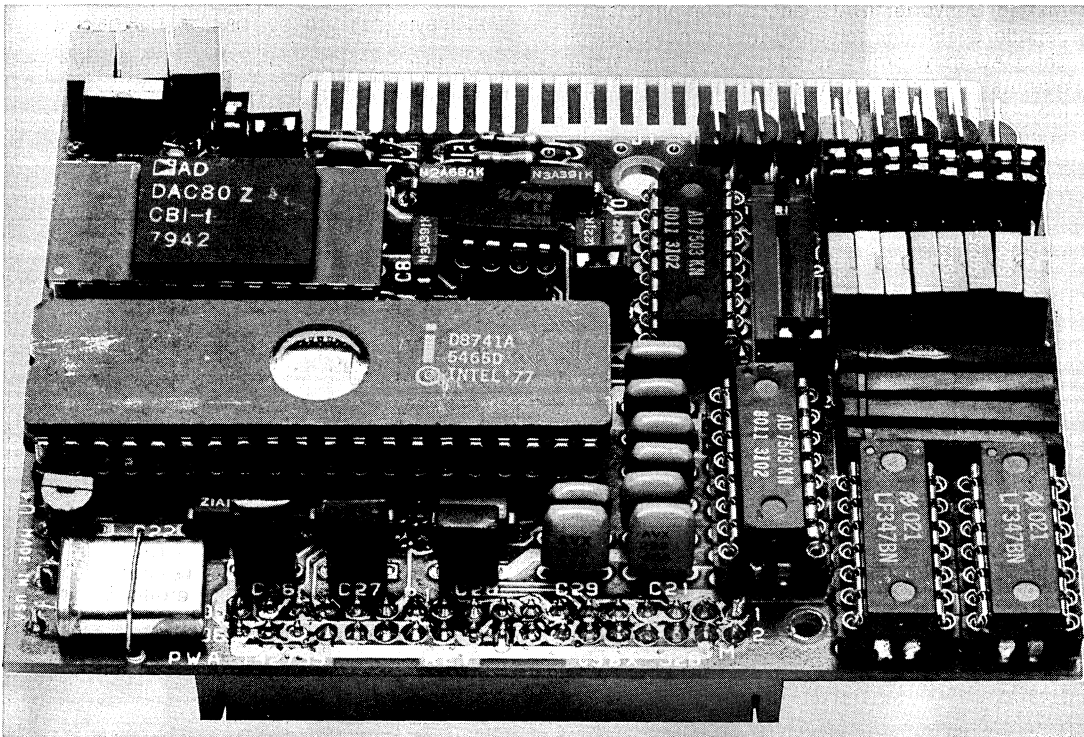
Part Number	Description
SBX 311	Analog Input MULTIMODULE Board



## 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 compatible 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 MUL-

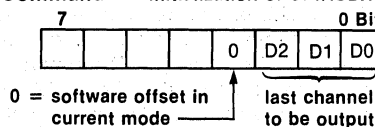
TIMODULE board status is available via the ISBX bus connector, to determine if the UPI is ready to receive updates to analog output channels.

**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**



**OUTPUT Command — Data Bytes**

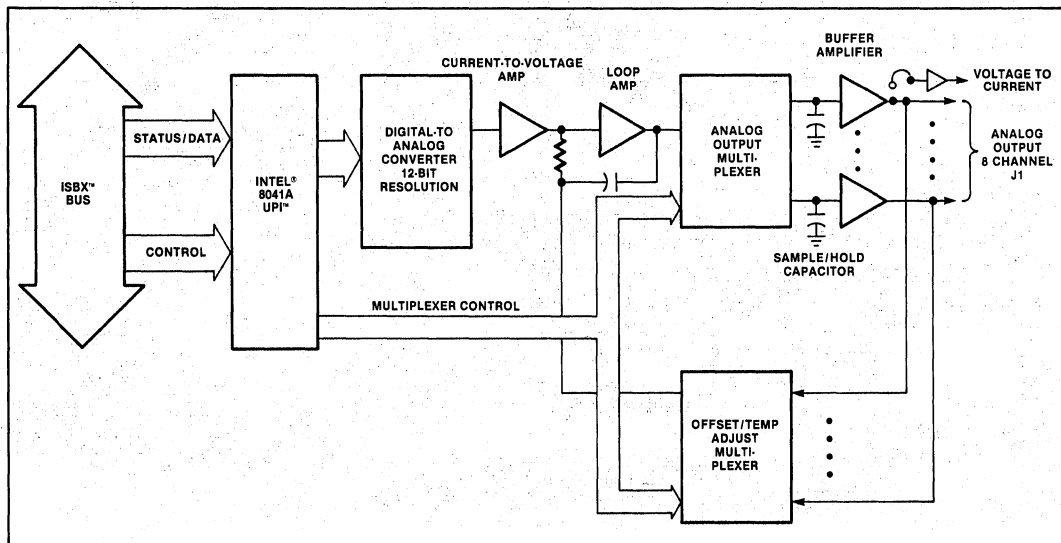
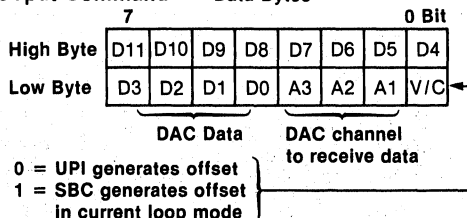
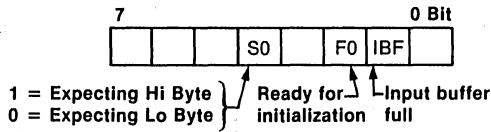


Figure 1. ISBC 328 Analog Output MULTIMODULE Board Block Diagram

# iSBX 328

## 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	± 0.11% FSR	@ 0° to 60°C
Voltage-Unipolar, maximum	± 0.22% FSR	@ 0° to 60°C
Voltage-Bipolar, typical	± 0.025% FSR	@ 25°C
Voltage-Bipolar, maximum	± 0.035% FSR	@ 25°C
Voltage-Bipolar, typical	± 0.10% 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	± 0.25% FSR	@ 0° to 60°C
Current Loop, maximum	± 0.36% 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**—6.35 cm (2.5 inches)

**Height**—1.4 cm (0.56 inch) MULTIMODULE board only

**Depth**—2.82 cm (1.13 inches) MULTIMODULE and iSBC board.

**Depth**—9.40 cm (3.7 inches)

**Weight**—85.06 gm (3.0 ounces)

## Electrical Characteristics

$V_{cc}$  = ±5 volts (±0.25V),  $I_{cc}$  = 140 ma max

$V_{dd}$  = ±12 volts (±0.6V),  $I_{dd}$  = 45 ma max (voltage mode)

= 200 ma max (current loop mode)

$V_{ss}$  = -12 volts (±0.6V),  $I_{ss}$  = 55 ma max

# ISBX 328

## 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

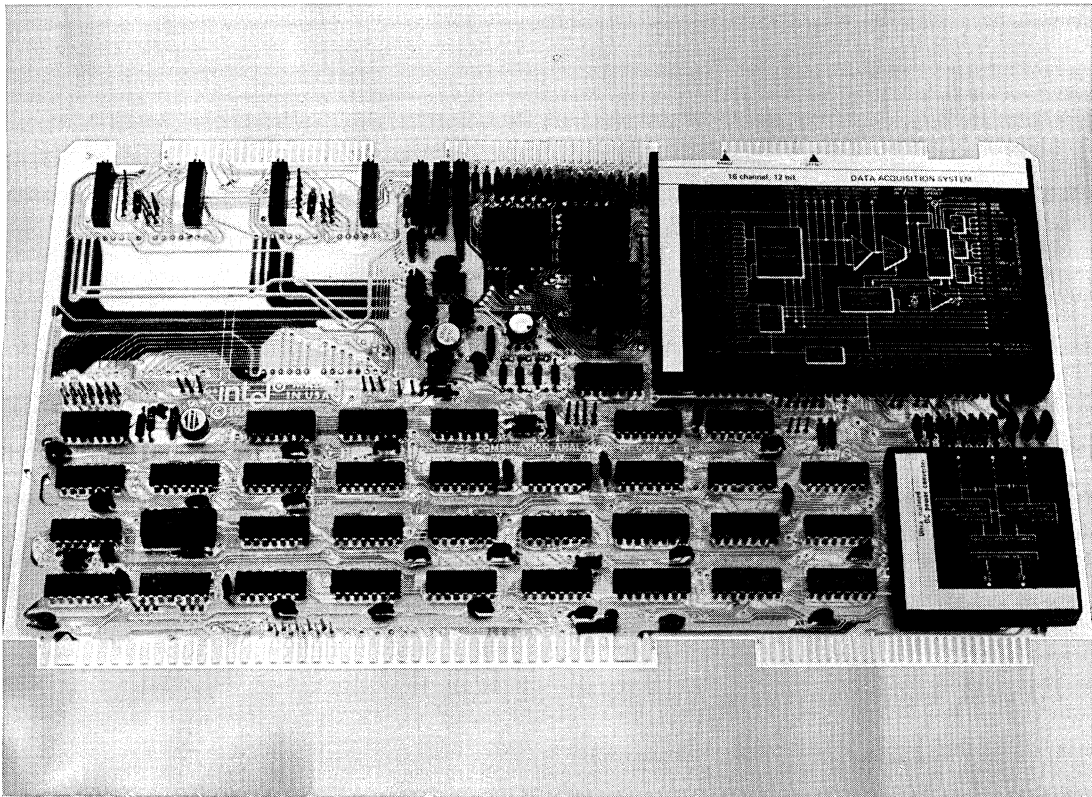
Part Number	Description
SBX 328	Analog Output MULTIMODULE Board



## iSBC 711 ANALOG INPUT BOARD

- iSBC 80 and MULTIBUS compatible
- Programmable gain amplifier
- Up to 16 differential/32 single-ended non-isolated inputs
- 12-bit A/D converter
- Fault protection on all inputs
- Memory mapped I/O
- Voltage or current loop input
- Single +5V supply

The iSBC 711 Analog Input Board is a complete single board input subsystem which allows easy interfacing of high level analog input signals to Intel's complete line of iSBC 80 single board computers. The iSBC 711 performs the basic functions of data acquisition of analog input signals under microprocessor control. The iSBC 711 consists of 8 differential/16 single ended channel multiplexer, input protection circuits, programmable gain amplifier, sample and hold amplifier, 12-bit A/D converter, DC to DC converter, memory mapped interface, and control logic.





## FUNCTIONAL DESCRIPTION

The iSBC 711 is electrically and mechanically compatible with the iSBC 80 series single board computers. Programming compatibility is also maintained with the iSBC 732 Analog Combination I/O Board. A block diagram of the iSBC 711 Analog Input Board is shown in Figure 1.

### Input Capacity

The iSBC 711 contains an 8 differential or 16 single ended analog input multiplexer. Optionally, the iSBC 711 can be expanded up to 16 differential or 32 single ended, non-isolated input channels via two plug-in multiplexers (Part No. Harris H1818A).

### Fault Protection

All input channels are protected up to  $\pm 28V$  via diode clamping, together with fusible current limit resistors, limiting potentially destructive overloads under fault conditions.

### Current Loop Input

The differential input channels have provisions for up to 16 user supplied 250-ohm resistors to accept 4 to 20 mA current loop inputs.

### Programmable Gain Amplifier

The programmable gain amplifier allows gains of 1, 2, 4, and 8 to be specified under program control.

## 12-Bit A/D Converter

The A/D converter is a 12-bit 35  $\mu s$  successive approximation device with sample and hold amplifier which can be jumper selected for 0 to +5V, 0 to +10V,  $\pm 5V$ , and  $\pm 10V$  input ranges. A/D conversion can be initiated by external trigger, pacer clock, or programmed I/O. Interrupt on conversion or end of scan is a standard feature. The external trigger is useful when the A/D conversion is to be synchronized to some external event. The internal pacer clock (10 ranges jumper selectable from 976  $\mu s$  to 1 sec) allows precise, evenly spaced A/D conversions where signal reconstruction is required.

## OPERATIONAL DESCRIPTION

### Analog Input

The iSBC 711 has three modes of operation for acquisition of analog inputs. These are:

1. Repetitive single channel input
2. Sequential input scan
3. Random channel input

**Repetitive Single Channel Input** — The channel is selected by a write to the multiplexer address register (MAR). The initiation of the first conversion is stated by a write to the command register (CR). Setting the appropriate bits in the command register allows stop/start, pacing, external trigger. Subsequent conversions are initiated by read of the A/D register (ADCR), or by another write to the command register.

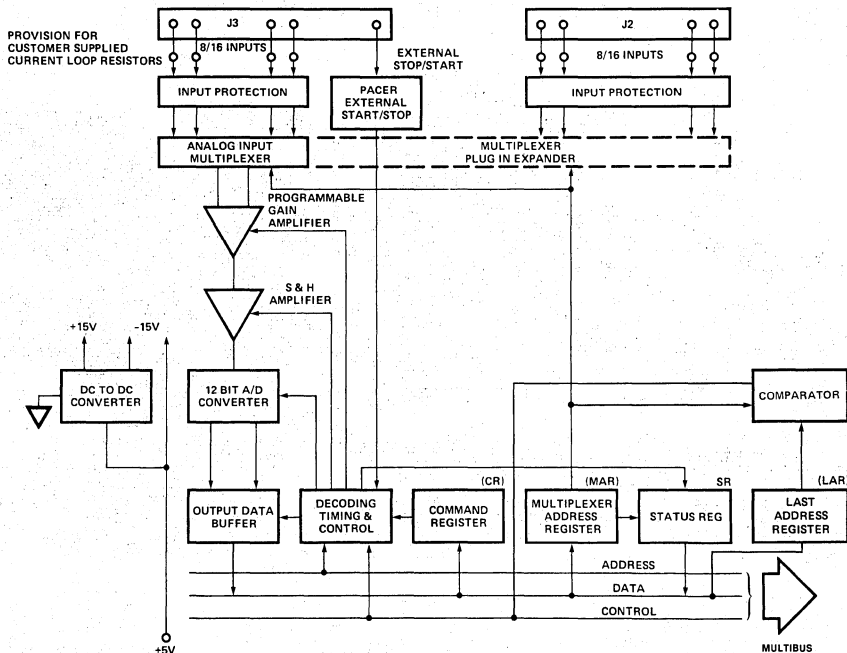


Figure 1. iSBC 711 Analog Input Board Block Diagram

# ISBC 711

**Sequential Input Scan** — This mode is initiated in the same manner as the repetitive single channel, except bit 1 of the command register (CR) is set to a "1" on the write command.

**Random Input Scan** — This mode is initiated in the same manner as repetitive single channel input; subsequent channels are selected by a write to the multiplexer address register (MAR) before a read of the current converted value from the ADCR register.

## Addressing

The ISBC 711 utilizes a memory mapping technique which simplifies programming in transferring 12-bit data. Since the ISBC 711 interfaces as memory, any of

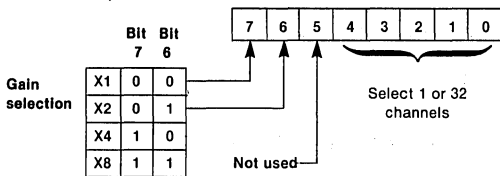
the 8080's 16-bit memory reference instructions can be used. The A/D converter is addressed as memory: base plus specific address. The base address is factory set at F700H and is jumper selectable. Table 1 shows address:

Address	Command	Function
Base + 0	Write	Write to command register
Base + 0	Read	Read status register
Base + 1	Write	Write to multiplexer address register
Base + 1	Read	Read multiplexer address register
Base + 2	Write	Write to last channel register
Base + 3	Write	Clear interrupts
Base + 4	Read	Read LS byte of A/D data
Base + 5	Read	Read MS byte of A/D data

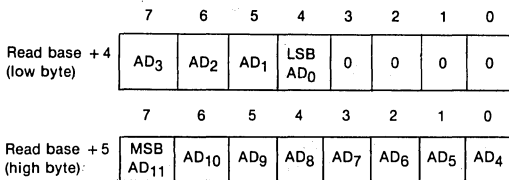
**Table 1. Address Commands and Functions**

## SPECIFICATIONS

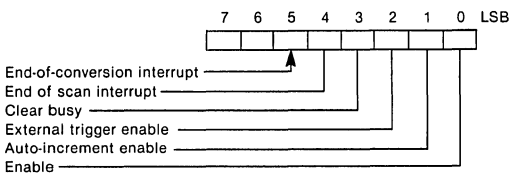
### Multiplexer Address Register (MAR) (Write Base + 1)



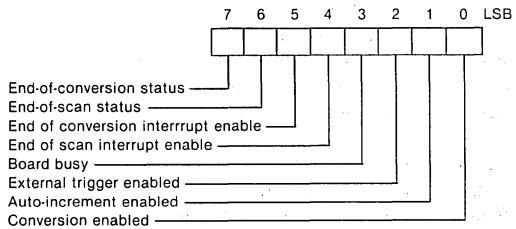
### A/D Converter Register (ADCR)



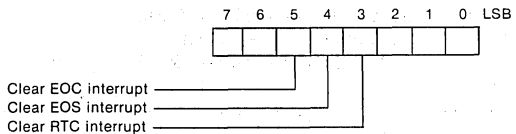
### Command Register (CR) (Write Base + 0)



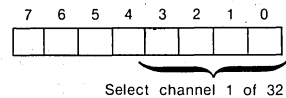
### Status Register (Read Base + 0)



### Clear Interrupt (Write Base + 3)



### Last Address Register (LAR) (Write Base + 2)



## Analog Input

**Number of Input Channels** — 8 differential or 16 single-ended (jumper selectable); expandable from 8/16 to 16/32 using two plug-in multiplexers (Part No. Harris H1818A).

**Resolution** — 12 bit bipolar or unipolar

**Sample and Hold Aperture Time** — <20 ns

**Sample and Hold Uncertainty** — 5 ns

**Multiplexer Input Voltage Ranges**

# ISBC 711

Gain X	A/D Input Range			
	+5V	+10V	±5V	±10V
1	+5V	+10V	±5V	±10V
2	+2.5V	+5V	±2.5V	±5V
4	1.25V	+2.5V	±1.25V	±2.5V
8	0.625V	+1.25V	±0.625V	±1.25V

Jumper  
selectable

Programmable

## Input Impedance

Power Off: 680 ohms  
Power On: >100M ohms

**Input Current Range** — 0-20 mA using 250-ohm user-installed resistors

## Source Impedance

Balanced: <5000 ohms  
Unbalanced: <1000 ohms

## Overall Accuracy @ 25°C

0.05% FSR ± ½ LSB (gain × 1)  
0.07% FSR ± ½ LSB (gain × 2, × 4, × 8)

(Includes 3 sigma noise, linearity, offset gain, and dynamic response errors.)

## Temperature Coefficient

0.0025% FSR/°C (gain × 1)  
0.0030% FSR/°C (gain × 2, × 4, × 8)

A/D Conversion Speed — 28 kHz

## Throughput

**Sample Rate** (Single Channel) — 17 kHz

**Channel to Channel Rate** — 16kHz

**Common Mode Rejection (CMR)** — 60 dB differential input

**Common Mode Voltage (CMV)** — ±10.24V (signal and common mode)

**Input Over-Voltage Protection** — ±28V DC, peak AC continuous

**External Trigger** — TTL compatible, 1.5 μs (min) better than 50 ns risetime

## Pacer Clock

Crystal controlled accuracy 0.05%

Divider gives range of  $\frac{1000}{2^n}$  ms

n=0 through 10

## Connectors

Interface	Pins (qty)	Centers		Mating Connectors
		in.	cm	
P1 ISBC Bus	86	0.156		Viking 3KH43/9AMK12
P2 ±15V Auxiliary power	60	0.1		
J1 Not used	50	0.1		3M3415-000 or TIH312125
J2 1st 8/16 input channels	50	0.1		3M3415-000 or TIH312125
J3 Expander 8/16 input channels	50	0.1		3M3415-000 or TIH312125

## Compatible Boards and Systems

iSBC 80/05  
iSBC 80/10  
iSBC 80/20  
System 80/10, and  
System 80/20-4

## Physical Characteristics

**Width** — 12.00 in. (30.48 cm)

**Height** — 6.75 in. (17.15 cm)

**Depth** — 0.5 in. (1.27 cm)

**Weight** — 1 lb (454 gm)

## Electrical Characteristics

**Power supply** — +5V ± 5%, 1.7A max

## Environmental Characteristics

	Reference Operating Conditions	Operative Limits	Transportation and Storage Limits
Temperature	25 ± 0.2°C	0 to 55°C	-25 to 85°C
Relative humidity		0 to 95% not to exceed upper limit 40°C dewpoint	0 to 95%
DC supply	+5V ± 5%	+5V ± 5%	N/A

## Reference Manuals

**9800485A** — iSBC 711 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

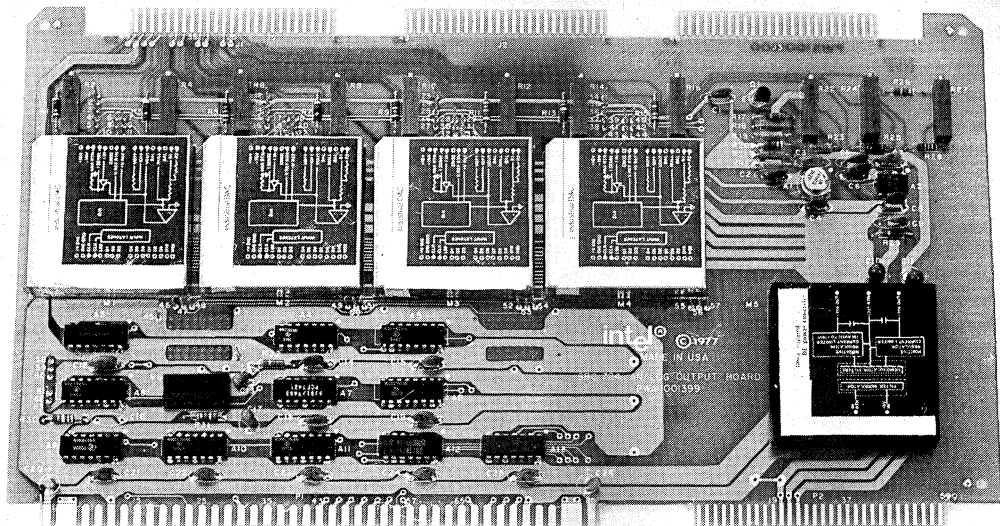
Part Number	Description
SBC 711	Analog Input Board



## iSBC 724 ANALOG OUTPUT BOARD

- Compatible with iSBC 80 series and MULTIBUS bus
- Four independent 12-bit D/A converters
- Short circuit protection on voltage output
- Memory Mapped I/O
- Accuracy: 0.05% FSR
- Unipolar or bipolar with 0 to +10V, 0 to +5V,  $\pm 10V$  ranges
- Single +5V supply

The iSBC 724 Analog Output Board is a complete single board analog output subsystem which allows easy interfacing of analog outputs from Intel's complete line of iSBC 80 series single board computers to voltage actuated devices or control elements. The iSBC 724 board contains four independent 12-bit digital to analog (D/A) converters and associated voltage output amplifiers, 8-bit holder register, DC to DC converter, memory mapped interface, and control logic.



# ISBC 724

## FUNCTIONAL DESCRIPTION

The iSBC 724 is electrically and mechanically compatible with the iSBC 80 series single board computers. Programming compatibility is also maintained with the analog output provided on the iSBC 732 analog I/O board. Each individual D/A converter can be jumper selected for 0 to +5V, 0 to +10V,  $\pm 5V$ ,  $\pm 10V$ . Short circuit protection for voltage outputs is a standard feature suitable for many display and control applications. A block diagram of the iSBC 724 analog output board is shown in Figure 1.

## OPERATIONAL DESCRIPTION

### Data Transfer

Data transfer to a 12-bit D/A converter is accomplished by a write of the least significant 4 bits to the holding register. When the 8 remaining bits of the upper byte are transferred, the 12 bits of data are immediately loaded into the specific D/A converter channel.

## Addressing

The iSBC 724 utilizes a memory mapping technique which simplifies programming in transferring 12-bit data. Since the iSBC 724 interfaces as memory, any of the 8080's memory reference instructions can be used. Each independent D/A converter is addressed as memory; base plus specific address. The base address is factory set at F708H which is jumper selectable. Table 1 shows addressing:

Table 1. Address Commands and Functions

Address	Command	Function
Base + 0	Write	Write to DAC holding register
Base + 1	Write	Write to DAC 0
Base + 2	Write	Write to DAC holding register
Base + 3	Write	Write to DAC 1
Base + 4	Write	Write to DAC holding register
Base + 5	Write	Write to DAC 2
Base + 6	Write	Write to DAC holding register
Base + 7	Write	Write to DAC 3

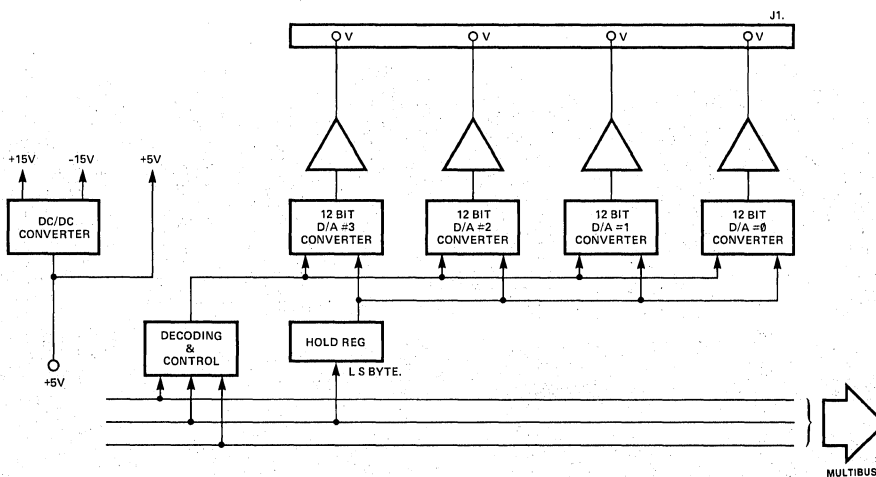
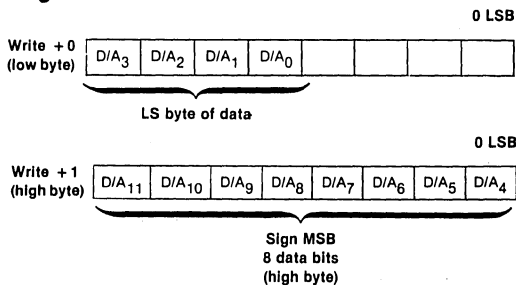


Figure 1. ISBC 724 Analog Output Board Block Diagram

# iSBC 724

## SPECIFICATIONS

### Registers



### Analog Output

**Number of Channels** — 4 non-isolated

**Resolution** — 12 bits bipolar or unipolar (switch selectable)

### Voltage Output Characteristics

**Voltage Output Ranges** — 0 to +5V, 0 to +10V, ±5V, ±10V (jumper selectable)

**Output Current** — ±5 mA @ ±10V

**Output Impedance** — 0.2 ohm

**Slew Rate** — 10V/μs

**Accuracy @ 25°C** — 0.05% FSR (includes linearity and noise)

**Temperature Coefficient** — 0.005/°C

### Connectors

Interface	Pins (qty)	Centers		Mating Connectors
		in.	cm	
P1 iSBC Bus	86	0.156	0.396	Viking 3KH43/9AMK12
P2 ±15V auxiliary power	60	0.1	0.254	
J1 4 channels of analog output	50	0.1	0.254	3M 3415-000 or TI H312125

### Compatible Boards and Systems

iSBC 80/05  
iSBC 80/10  
iSBC 80/20  
System 80/10  
System 80/20-4

### Physical Characteristics

**Width** — 12.00 in. (30.48 cm)

**Height** — 6.75 in. (17.15 cm)

**Depth** — 0.50 in. (1.27 cm)

**Weight** — 1 lb, 2 oz (511 gm)

### Electrical Characteristics

**Power Supply** — +5V ±5%, 2.0A max

### Environmental Characteristics

	Reference Operating Conditions	Operative Limits	Transportation and Storage
Temperature	25 ± 2.0°C	0 to 55°C	- 25 to 85°C
Relative humidity		0 to 95% not to exceed upper limit 40°C dewpoint	0 to 95%
DC supply	+5 ± 5%	+5V ± 5%	N/A

### Reference Manuals

**9800486A** — iSBC 724 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

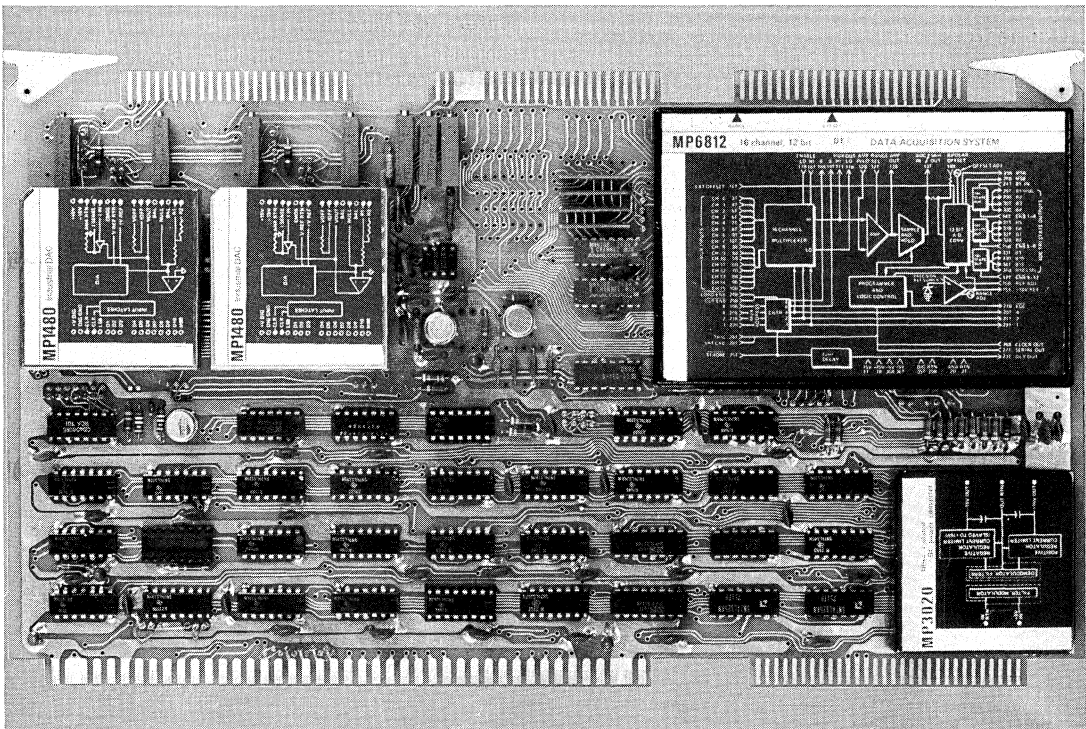
Part Number	Description
SBC 724	Analog Output Board



## iSBC 732 ANALOG COMBINATION I/O BOARD

- iSBC 80 and MULTIBUS compatible
- Up to 16 differential/32 single ended non-isolated inputs
- Fault protection on all inputs
- Voltage or current loop input/output
- Programmable gain amplifier
- 12-bit, 28 kHz A/D converter
- Two 12-bit D/A converter output channels
- Memory mapped I/O
- Single +5V supply

The iSBC 732 is a complete single board analog input/output subsystem which allows easy interfacing of high level analog input and output signals to Intel's complete line of iSBC 80 series single board computers. The iSBC 732 performs the basic functions of data acquisition of analog inputs and controlled analog output signals under microprocessor control.



# ISBC 732

## FUNCTIONAL DESCRIPTION

The iSBC 732 is electrically and mechanically compatible with the iSBC 80 series single board computers. Programming compatibility is also maintained with the iSBC 711 Analog Input Board and the iSBC 724 Analog Output Card. A block diagram of the iSBC 732 Combination Analog I/O Board is shown in Figure 1.

### Input Capacity

iSBC 732 contains an 8 differential or 16 single ended analog input multiplexer. Optionally, the iSBC 732 can be expanded up to 16 differential or 32 single ended, non-isolated input channels via two plug-in multiplexers (Part No. Harris H1818A).

### Fault Protection

All input channels are protected up to  $\pm 28V$  via diode clamping, together with fusible current limit resistors, limiting potentially destructive overloads under fault conditions.

### Current Loop Input

The differential input channels have provisions for up to 16 user supplied 250-ohm resistors to accept 4 to 20 mA current loop inputs.

### Programmable Gain Amplifier

The programmable gain amplifier allows gains of 1, 2, 4, and 8, to be specified under program control.

### 12-Bit A/D Converter

The A/D converter is a 12-bit, 34  $\mu s$  successive approximation device with sample and hold amplifier which can be jumper selected for 0 to +5V, 0 to +10V,  $\pm 5V$  and  $\pm 10V$  input ranges. A/D conversion can be initiated by external trigger, pacer clock, or programmed I/O. Interrupt on conversion or end of scan is a standard feature. The external trigger is useful when the A/D conversion is to be synchronized to some external event. The internal pacer clock (10 ranges jumper selectable from 976  $\mu s$  to 1 sec) allows precise, evenly spaced A/D conversions where signal reconstruction is required.

### D/A Converter

The dual 12-bit non-isolated D/A converter can be configured for voltage or current loop output. Each D/A converter voltage output can be jumper selected for 0 to +5V, 0 to +10V,  $\pm 5V$ ,  $\pm 10V$ , or optionally 0 to 20 mA current loop output. Short circuit protection for voltage outputs is a standard feature and user supplied compliance voltage of up to 30V make the current loop output suitable for many control applications.

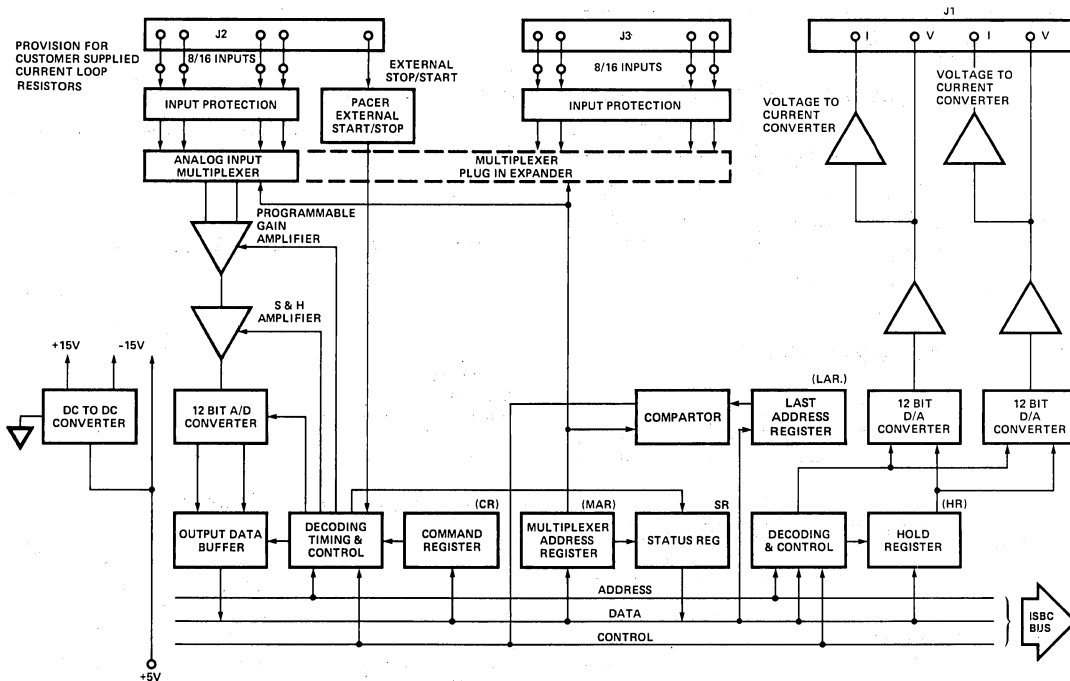


Figure 1. ISBC 732 Analog Combination I/O Board Block Diagram



## OPERATIONAL DESCRIPTION

### Analog Input

The iSBC 732 has three modes of operation for acquisition of analog inputs. These are:

1. Repetitive single channel input
2. Sequential input scan
3. Random channel input

**Repetitive Single Channel Input** — The channel is selected by a write to the multiplexer address register (MAR). The initiation of the first conversion is started by a write command to the command register (CR). Setting the appropriate bits in the command register allow stop/start, pacing, external trigger. Subsequent conversions are initiated by a read command of the A/D register (ADCR).

**Sequential Input Scan** — This mode is initiated in the same manner as the repetitive single channel, except bit 1 of the command register (CR) is set to a "1" on the write command.

**Random Input Scan** — This mode is initiated in the same manner as repetitive single channel input. Subsequent channels are selected by a write command to the multiplexer address register (MAR) before a read command of the current converted value from the ADCR register.

### Analog Output

Data transfer to a D/A converter is accomplished by a write command of the least significant 4 bits to the holding register. When the upper byte is transferred, the 12 bits of data are immediately loaded to the specific D/A channel.

### Addressing

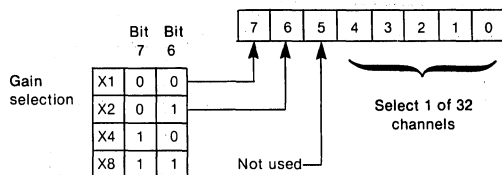
The iSBC 732 utilizes a memory mapping technique which simplifies programming in transferring 12-bit data. Since the iSBC 732 interfaces as memory, any of the 8080's memory reference instructions can be used. The A/D and independent D/A converters are addressed as memory: base plus specific address. The base address is set at F700H which is jumper selectable. Table 1 shows addressing.

Table 1. Address Commands and Functions

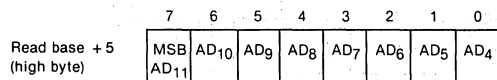
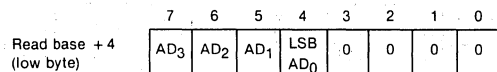
Address	Command	Function
Base + 0	Write	Write to command register
Base + 0	Read	Read status register
Base + 1	Write	Write to multiplexer address register
Base + 1	Read	Read multiplexer address register
Base + 2	Write	Write to last channel register
Base + 3	Write	Clear interrupts
Base + 4	Read	Read LS byte of A/D data
Base + 5	Read	Read MS byte of A/D data
Base + 8	Write	Write to DAC holding register (low byte)
Base + 9	Write	Write to DAC 0 (high byte)
Base + A	Write	Write to DAC holding register (low byte)
Base + B	Write	Write to DAC 1 (high byte)

## SPECIFICATIONS

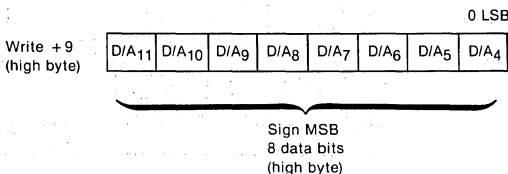
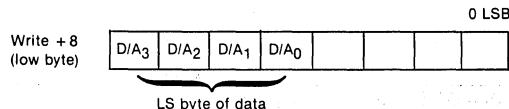
### Multiplexer Address Register (MAR) Write



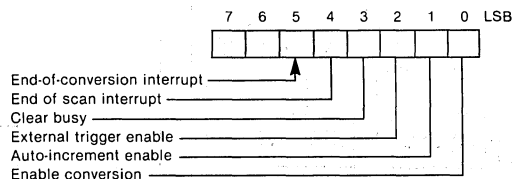
### A/D Converter Register (ADCR)



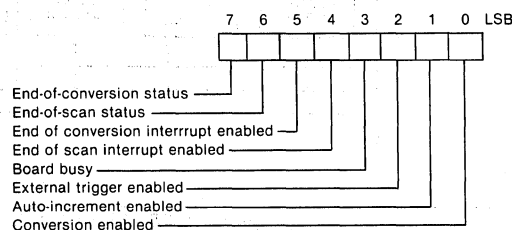
### D/A Hold Register



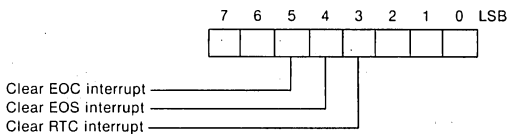
### Command Register (CR) (Write Base + 0)



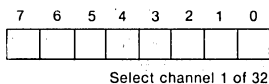
### Status Register (Read Base + 0)



**Clear Interrupt (Write Base + 3)**



**Last Address Register (LAR) (Write Base + 2)**



**Analog Input**

**Number of Input Channels** — 8 differential or 16 single ended (jumper selectable); expandable from 8/16 to 16/32 using two plug-in multiplexers (Part No. Harris H1818A).

**Resolution** — 12-bit bipolar or unipolar  
**Sample and Hold Aperture Time** — <20 ns  
**Sample and Hold Uncertainty** — 5 ns  
**Multiplexer Input Voltage Ranges**

Gain X	A/D Input Range				Jumper selectable
	+5V	+10V	±5V	±10V	
1	+5V	+10V	±5V	±10V	Software programmable
2	+2.5V	+5V	±2.5V	±5V	
4	1.25V	+2.5V	±1.25V	±2.5V	
8	0.625V	+1.25V	±0.625V	±1.25V	

**Input Impedance**

Power Off: 680 ohms  
 Power On: >100M ohms

**Input Current Range** — 0 to 20 mA using 250-ohm user installed resistors

**Source Impedance**

Balanced: <5000 ohms  
 Unbalanced: <1000 ohms

**Overall Accuracy @ 25°C**

0.05% FSR ± ½ LSB (gain × 1)  
 0.07% FSR ± ½ LSB (gain × 2, × 4, × 8)  
 (Includes 3 sigma noise, linearity, offset gain, and dynamic response errors)

**Temperature Coefficient**

0.0025% FSR/°C (gain × 1)  
 0.0030% FSR/°C (gain × 2, × 4, × 8)

**A/D Conversion Speed** — 28 kHz

**Throughput**

**Sample Rate (Single Channel)** — 17 kHz  
**Channel to Channel Rate** — 16 kHz  
**Common Mode Rejection (CMR)** — 60 dB differential input

**Common Mode Voltage (CMV)** — ± 10.24V (signal and common mode)

**Input Over-Voltage Protection** — ± 28V DC, peak AC continuous

**External Trigger** — TTL compatible, 1.5 μs (min) better than 50 ns risetime

**Pacer Clock**  
 Crystal controlled accuracy 0.05%

Divider gives range of  $\frac{1000}{2^n}$  ms  
 n = 0 through 10

**Analog Output**

**Number of Channels** — 2 Non-isolated  
**Resolution** — 12 bits bipolar or unipolar (switch selectable)

**Voltage Output Characteristics**

**Voltage Output Ranges** — 0 to +5V, 0 to +10V, ±5V, ±10V (jumper selectable)  
**Output Current** — ± 5 mA @ ± 10V  
**Output Impedance** — 0.2 ohm  
**Slew Rate** — 10V/μs with no external capacitance  
**Accuracy @ 25°C** — 0.05% FSR (includes linearity and noise)  
**Temperature Coefficient** — 0.005°C

**Current Loop Characteristics**

**Current Output** — 0 to 20 mA  
**Load Resistance** — 0 to 500 ohms  
**Compliance Voltage** — Up to 30V DC (provided by user)  
**Accuracy @ 25°C** — 0.075% FSR (includes linearity and noise)  
**Temperature Coefficient** — 0.005/°C

**Connectors**

Interface	Pins (qty)	Centers		Mating Connectors
		in.	cm	
P1 IIBC Bus	86	0.156	0.396	Viking 3KH43/9AMK12
P2 ± 15V auxiliary power	60	0.1	0.254	
J1 2 channels of analog output	50	0.1	0.254	3M 3415-000 or TI H312125
J2 1st 8/16 input channels	50	0.1	0.254	3M 3415-000 or TI H312125
J3 Expander 8/16 input channels	50	0.1	0.254	3M 3425-000 or TI H312125

## iSBC 732

### Compatible Boards and Systems

iSBC 80/05  
iSBC 80/10  
iSBC 80/20  
System 80/10  
System 80/20-4

### Physical Characteristics

**Width** — 12.00 in. (30.48 cm)  
**Height** — 6.75 in. (17.15 cm)  
**Depth** — 0.5 in. (1.27 cm)  
**Weight** — 1 lb 4 oz (568 gm)

### Electrical Characteristics

**Power Supply** — +5V ± 5%, 2.3A max

### Environmental Characteristics

	Reference Operating Conditions	Operative Limits	Transportation and Storage Limits
Temperature	25 ± 0.2 °C	0 to 55 °C	- 25 to 85 °C
Relative humidity		0 to 95% not to exceed upper limit 40°C dewpoint	0 to 95%
DC supply	+5V ± 5%	+5V ± 5%	N/A

### Reference Manuals

**9800487-02** — iSBC 732 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 732 Analog Combination I/O Board

---

*System Packaging and  
Power Supplies*

**10**

---

1. The first part of the document discusses the importance of maintaining accurate records of all transactions and activities. It emphasizes that this is crucial for ensuring transparency and accountability in the organization's operations.

2. The second part of the document outlines the various methods and tools used to collect and analyze data. It highlights the need for consistent and reliable data collection processes to support effective decision-making.

3. The third part of the document focuses on the role of technology in data management and analysis. It discusses how modern software solutions can streamline data collection, storage, and reporting, thereby improving efficiency and accuracy.

4. The fourth part of the document addresses the challenges associated with data management, such as data quality, security, and privacy. It provides strategies to mitigate these risks and ensure that data is used responsibly and ethically.

5. The fifth part of the document concludes by summarizing the key findings and recommendations. It stresses the importance of ongoing monitoring and evaluation to ensure that data management practices remain effective and up-to-date.

6. The sixth part of the document provides a detailed overview of the data collection process, including the identification of data sources, the design of data collection instruments, and the implementation of data collection procedures.

7. The seventh part of the document discusses the various methods used for data analysis, such as descriptive statistics, inferential statistics, and qualitative analysis. It explains how these methods are used to interpret the data and draw meaningful conclusions.

8. The eighth part of the document focuses on the importance of data visualization in presenting complex information in a clear and concise manner. It discusses various visualization techniques and their applications in data analysis.

9. The ninth part of the document addresses the ethical considerations surrounding data management and analysis. It discusses the need for transparency, informed consent, and data protection to ensure that the use of data is ethical and lawful.

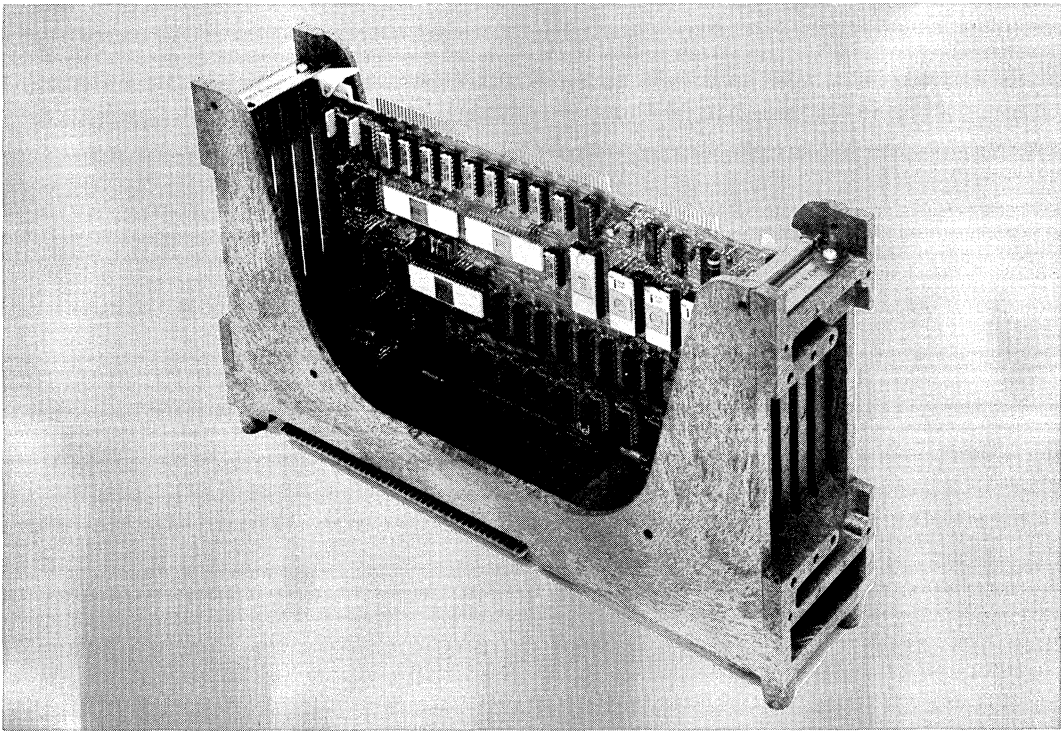
10. The tenth part of the document provides a final summary and concludes the report. It reiterates the key findings and emphasizes the importance of data management and analysis in achieving organizational success.



## iSBC 604/614 or (pSBC 604/614\*) MODULAR CARD CAGE/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 than 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.

# iSBC 604/614

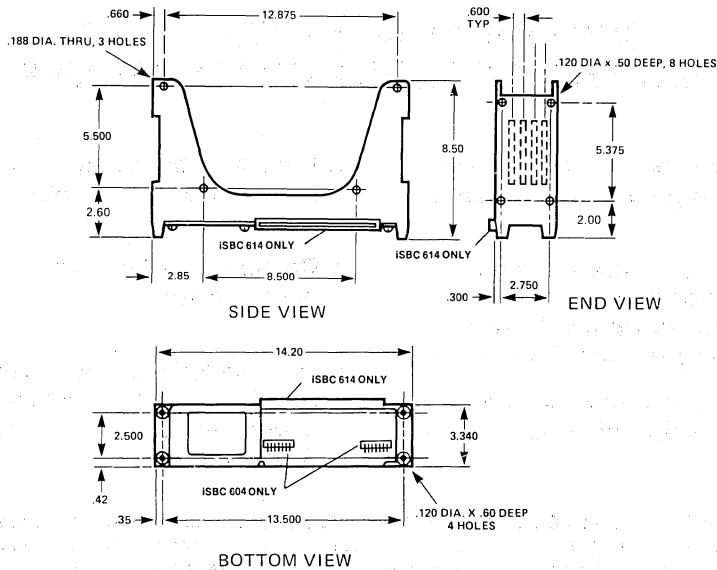


Figure 1. iSBC 604/614 Modular Backplane and Cardcage Dimensions

## 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

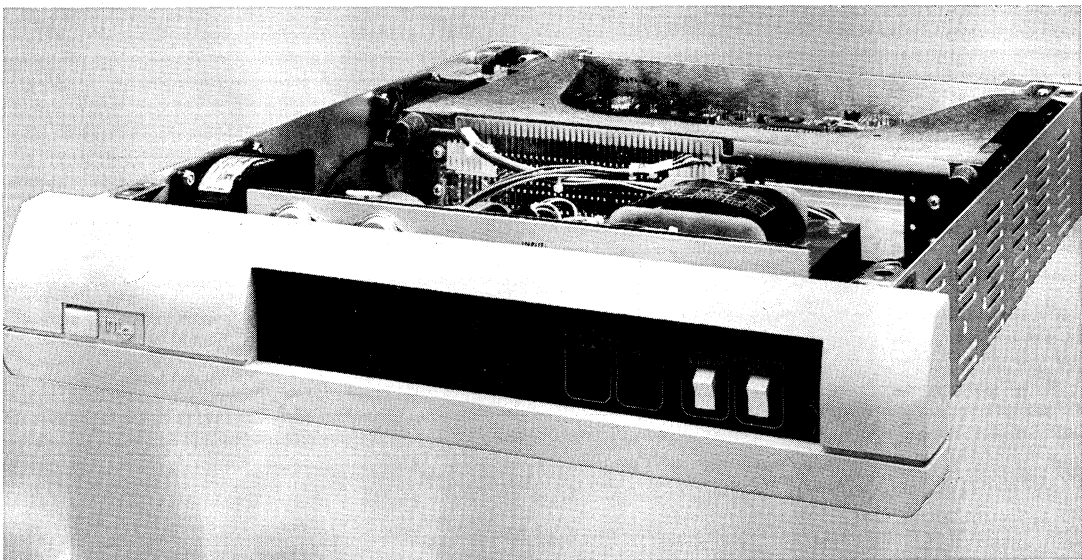
Part number	Description	Part Number	Description
SBC 604	Modular Cardcage/Backplane (Base Unit)	SBC 614	Modular Cardcage/Backplane (Expansion Unit)



## iSBC 655 SYSTEM CHASSIS

- A rack-mountable package for Intel microcomputer systems
- Provides the Intel MULTIBUS structure used on the single board computers
- Compact single chassis power supply with all standard iSBC board voltages
- Forced-air cooling
- Attractive front panel with control switches and indicator lights
- 110V or 220V operation at 50/60 Hz
- 19-inch rack mountable
- Parallel I/O connectors and RS232C cable included

The iSBC 655 System Chassis is an attractive 3.5" high unit designed for use in Intel Microcomputer Systems. The Chassis' four slots accommodate both single board computers and expansion boards which provide additional I/O, memory, or peripheral controller functions. The iSBC 655 System Chassis will accept all Intel boards using the Multibus architecture. DC power is provided at  $\pm 5$ VDC and  $\pm 12$ VDC levels, at current levels commensurate with typical combinations of four boards. The chassis is designed to provide adequate cooling to both power supply and circuit boards over external temperatures ranging from 0°C to 50°C. Current limiting and over-voltage protection are provided on all outputs. The power supply recognizes an AC power failure condition and provides a TTL signal sufficiently in advance of DC power failure to allow orderly system shut-down. For user convenience, system RESET and INTERRUPT switches are provided on the front panel to facilitate system restarts and provide for operator intervention. RUN and HALT LED indicators are driven to indicate the operational status of the single board computer.





**SPECIFICATIONS**

**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 -16 V

**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 ±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, 7.5 ms typical) after AC LOW goes true.

**Physical Characteristics**

**Height** — 3.5 inches (8.9 cm)

**Width** — 19 inches (48.3 cm) at Front Panel, 17 inches (43.2 cm) behind Front Panel

**Depth** — 20 inches (50.8 cm) with all protrusions

**Weight** — 37 pounds (17 Kg)

**Environmental Characteristics**

**Temperature** — Operating: 0°C to 50°C. Non-Operating: -40°C to 85°C

**Relative Humidity** — Up to 90%, non-condensing

**Equipment Supplied**

iSBC 655 System Chassis with iSBC 635 Power Supply, iSBC 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 (Not Supplied)**

**9800709** — iSBC 655 System Chassis Hardware Reference Manual

**9800298** — iSBC 635 Power Supply Hardware Reference Manual

**9800708** — iSBC 604/614 Cardcage Hardware Reference Manual

Manuals may be ordered from any Intel Sales Representative, Distributor office or from Intel Literature Department, 3065 Bowers Avenue, Santa Clara, CA 95051

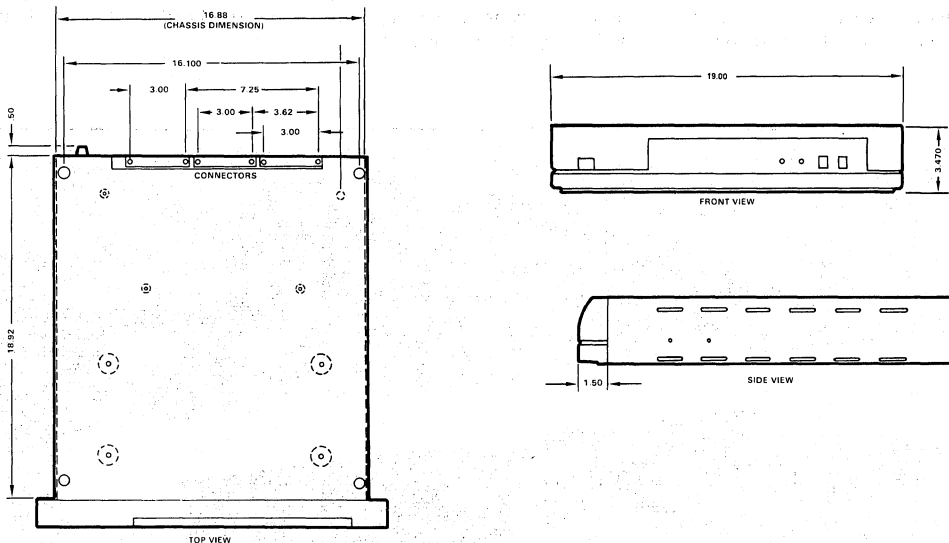


Figure 1. iSBC 655 Dimensions (inches)

**ORDERING INFORMATION**

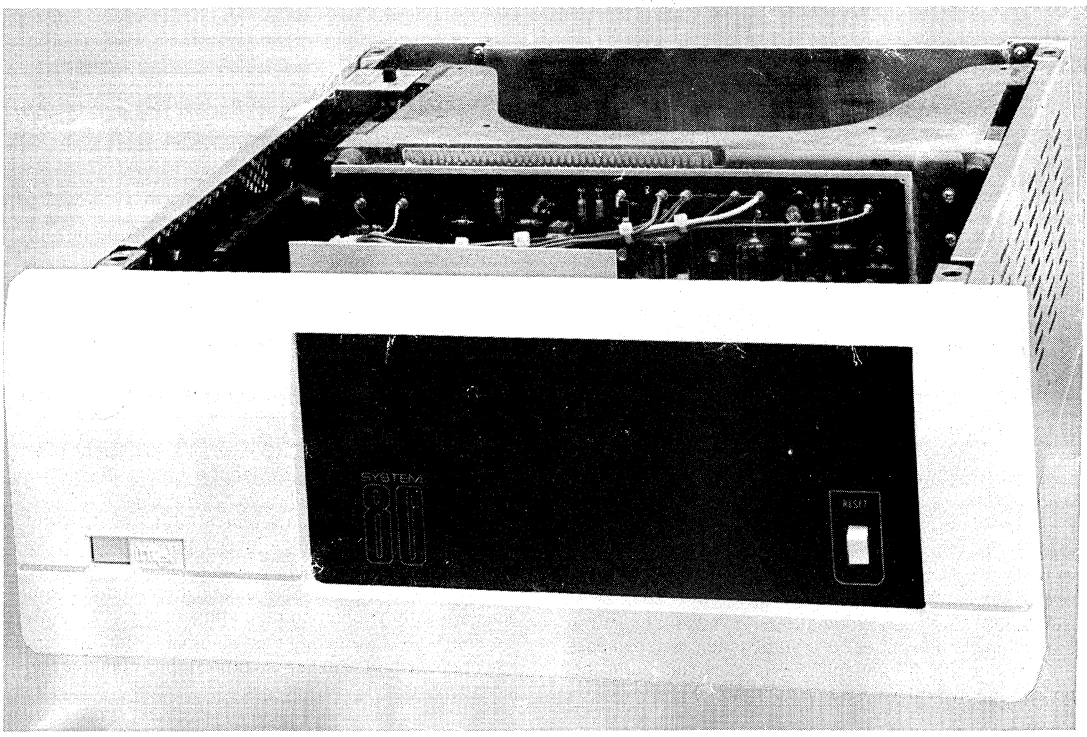
Part Number	Description
SBC 655	iSBC 655 System Chassis



## 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.



# ISBC 660

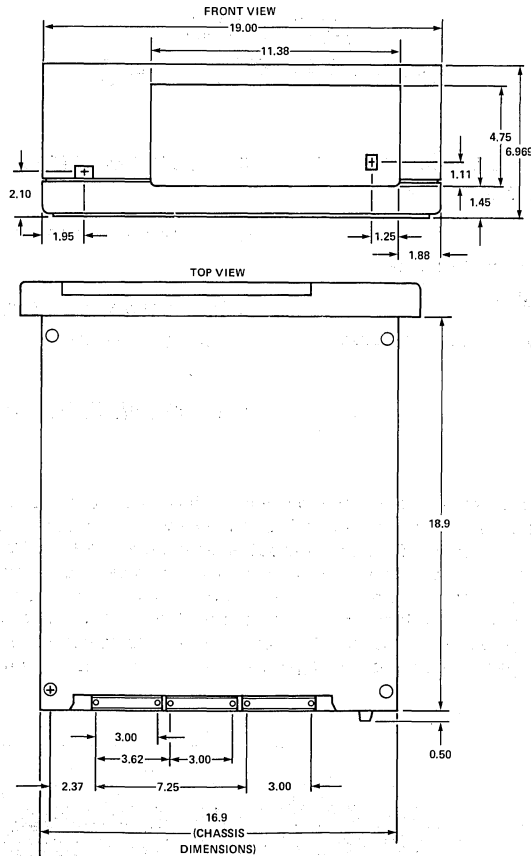


Figure 1. ISBC System Chassis Dimensions

## SPECIFICATIONS

### Electrical Characteristics

#### Input Power

Frequency: 50 Hz  $\pm$  5%, 60 Hz  $\pm$  5%

Voltage: 115V  $\pm$  10%, 230V  $\pm$  10%, 215 VAC  $\pm$  10%,  
100 VAC  $\pm$  10% via user configured wiring options

#### Output Power

Power	Output Current (Max)	Current Limit (Amps)	Over-Voltage Protection
+ 12V	4.5A	5.4	15V $\pm$ 1V
+ 5V	30A	3.6	6.2V $\pm$ 0.4V
- 5V	1.75A	2.1	-6.2V $\pm$ 0.4V
- 12V	1.75A	2.1	-15V $\pm$ 1V

**Combined Line/Load Regulation** —  $\pm$  1% at  $\pm$  10% static line change and  $\pm$  50% static load change, measured at the output connector ( $\pm$  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  $\mu$ s for  $\pm$  50% load change.

**Output Transient Deviation** — Less than  $\pm$  5% of initial voltage for  $\pm$  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 power-down performance at other input combinations (i.e. 100 VAC, 215 VAC, 50 Hz).

## iSBC 660

---

**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

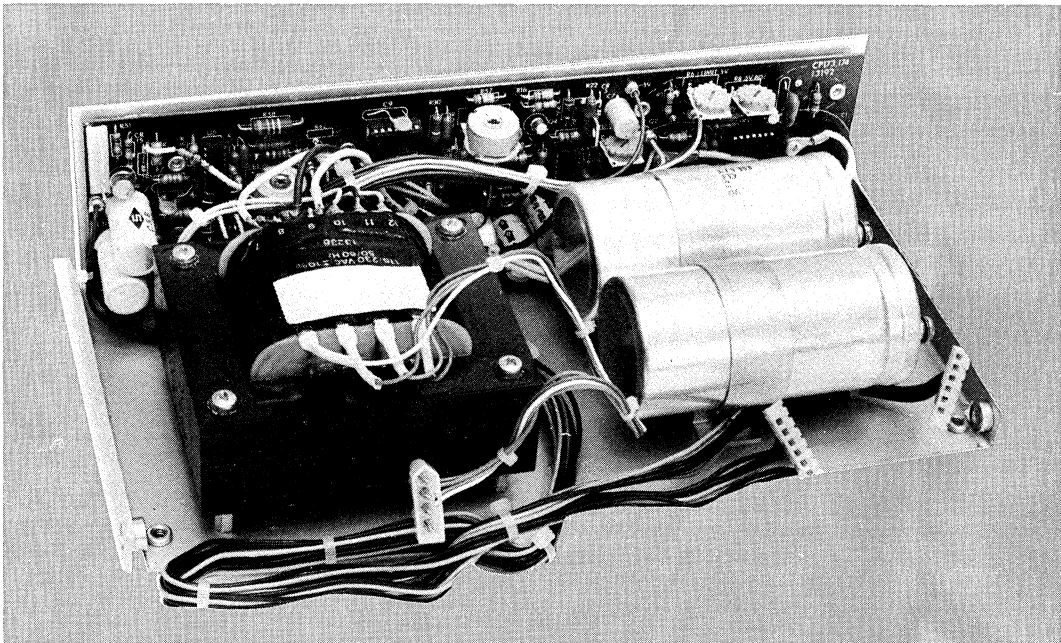
SBC 660	iSBC 660 system chassis
---------	-------------------------



## iSBC 635 POWER SUPPLY

- Compact single chassis
- $\pm 5V$  and  $\pm 12V$  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, -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 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.



# ISBC 635

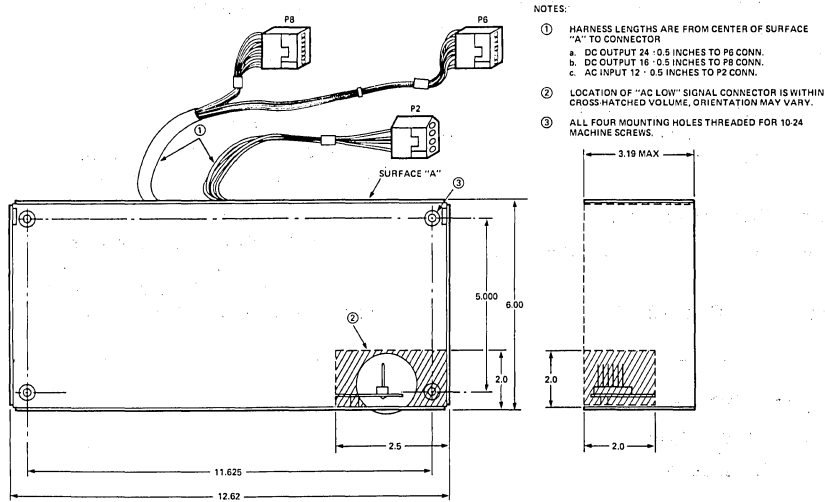


Figure 1. ISBC 635 Mounting Information

## SPECIFICATIONS

### Mating Connectors<sup>1</sup>

#### AC Input

Connector	Molex	03-09-1042 or equivalent
Pin	Molex	02-09-1118 or equivalent (18 to 22 gauge wire)

#### DC Output<sup>2</sup>

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** —  $\pm 1\%$  at  $\pm 10\%$  static line change and  $\pm 50\%$  static load change, measured at the output connector ( $\pm 0.2\%$  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  $\mu$ sec for  $\pm 50\%$  load change

**Output Transient Deviation** — Less than  $\pm 5\%$  of initial voltage for  $\pm 50\%$  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.

## iSBC 635

---

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

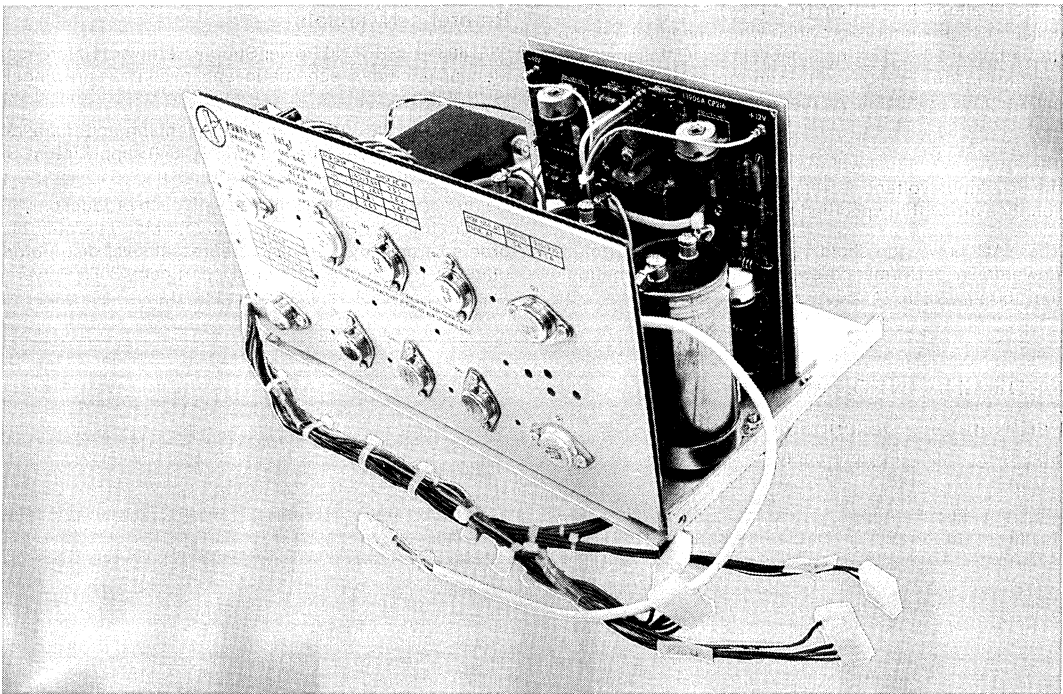
SBC 635 Power supply



## iSBC 640 POWER SUPPLY

- $\pm 5V$  and  $\pm 12V$  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 Modular Backplane/Cardcage assembly. 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  $\pm$  5%, 60 Hz  $\pm$  5%

**Voltage:** 115V  $\pm$  10%, 230V  $\pm$  10%, 215VAC  $\pm$  10%,  
100VAC  $\pm$  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 $\pm$ 1V
+ 5V	30A	31.5-45.0	15.0	6.2V $\pm$ 0.4V
- 5V	1.75A	1.8- 3.2	0.9	- 6.2V $\pm$ 0.4V
- 12V	1.75A	1.8- 3.2	0.9	- 15V $\pm$ 1V

**Combined Line/Load Regulation** —  $\pm$ 1% at  $\pm$ 10% static line change and  $\pm$ 50% static load change, measured at the output connector ( $\pm$ 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  $\mu$ sec for  $\pm$ 50% load change.

**Output Transient Deviation** — Less than  $\pm$  10% of initial voltage for  $\pm$  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<sup>1</sup>**

**AC Input**

Connector	Molex	03-09-1042 or equivalent
Pin	Molex	02-09-1118 or equivalent (18 to 22 gauge wire)

**DC Output<sup>2</sup>**

Header	Molex	09-66-1071
	AMP	87194-6

**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) (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

*Microcomputer  
Development Systems*

---

**11**

THE UNIVERSITY OF CHICAGO LIBRARY

1010 S. EAST ASIAN BLDG.

CHICAGO, ILL. 60607



## **New dimensions in development solutions.**

Solutions that deliver the support you need to develop microcomputer based products successfully and economically. No other development system product line supports such a broad range of microprocessor and support chip functionality. No other development system offers such a broad selection of high-level software support and de-bug tools. And no other development system today delivers state-of-the-art development support without obsolescence of previous systems.

### **Evolutionary Growth**

Every Intellec® development system delivered since 1975 can easily be upgraded to current standards. Including resident support for 16-bit product development and distributed development solutions.

### **Ease of Entry**

The new Series II/Model 120 Microcomputer Development System delivers full entry level support at the lowest entry price, while maintaining full upgradability.

### **Unprecedented 16-bit Support**

The newly introduced Intellec Series III/Model 286 Microcomputer Development System features a resident iAPX 86 16-bit CPU for direct execution of 16-bit Intel software AND a resident 8-bit CPU for execution of all Intel 8-bit microprocessor software.

### **Distributed Development Solutions**

For larger development projects, Intel delivers NDS-1—the Networked Development System—with up to eight development systems sharing hard disk speed and centralized file resources.

### **Software. Key to Programmer Productivity**

High level languages and powerful macro assemblers produce linkable and relocatable object modules, allowing the designer to match software strengths to project requirements for optimal development speed and product performance. New introductions include PL/M, PASCAL and FORTRAN compilers for the iAPX family of 16-bit microprocessors.

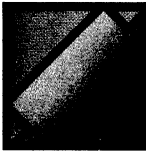
### **ICE™ Emulation. For Every Processor**

Intel's family of microprocessors is supported by ICE in-circuit emulation modules for integration of system hardware and software throughout the design cycle. ICE modules feature real-time and single-step trace and debugging.

### **The Development Investment**

Intellec development tools provide the support you need to manage the complexity of microcomputer product development over the next decade.

You invest in a family of support tools that supports, in turn, a comprehensive and cohesive family of processors and peripherals—from digital signal processors to 8-bit microcontrollers to 16-bit microsystems operating in a multiprocessing environment. And Intel's record of product revolution through product evolution means your development dollars are an investment in today's technology AND tomorrow's technology.



## 8080/8085 PERFORMANCE GUIDE

For developing 8080/8085 programs on an 8-bit based Inteltec® development system.

UPGRADE OPTIONS	CURRENT DEVELOPMENT SYSTEM MODEL 800	SERIES II MODELS 120,220,230	SERIES II/85 MODELS 225,235	SERIES II MODEL 240	SERIES II/85 MODEL 245
TODAY'S PERFORMANCE ASM 80/85 V3.0	1X	1X	1.4X	1.3X	1.8X
ADD NEW ASM 80/85 V4.0	1.7X	1.7X	2.3X	2.0X	3.0X
ADD MODEL 505 ASM 80/85 V4.0	N/A	2.3X	"	3.0X	"
ADD MODEL 740 ASM 80/85 V4.0	2.0X	2.0X	3.0X	"	"
ADD MODEL 505 & MODEL 740 ASM 80/85 V4.0	N/A	3.0X	"	"	"

\*The current system includes this option.

## IAPX 86/88 PERFORMANCE GUIDE

For developing iAPX 86 or 88 programs on either an 8-bit or 16-bit based Inteltec® development system.

UPGRADE OPTIONS	CURRENT DEVELOPMENT SYSTEM	MODEL 800	SERIES II MODELS 120,220,230	SERIES II/85 MODELS 225,235	SERIES II MODEL 240	SERIES II/85 MODEL 245
TODAY'S PERFORMANCE	ASM 86/88	1X	1X	1.2X	1.5X	2.1X
	PL/M 86/88	1X	1X	1.3X	1.4X	2.0X
ADD MODEL 505	ASM 86/88	N/A	1.2X	"	2.1X	"
	PL/M 86/88	N/A	1.3X	"	2.0X	"
ADD MODEL 556	ASM 86/88	1.6X	1.6X	1.7X	2.7X	3.1X
	PL/M 86/88	2.3X	2.3X	2.4X	3.7X	4.0X
ADD MODEL 556 I	ASM 86/88	N/A	1.7X	N/A	3.1X	"
	PL/M 86/88	N/A	2.4X	N/A	4.0X	"
ADD MODEL 740	ASM 86/88	1.5X	1.5X	2.1X	"	"
	PL/M 86/88	1.4X	1.4X	2.0X	"	"
ADD MODEL 505 & MODEL 740	ASM 86/88	N/A	2.1X	"	"	"
	PL/M 86/88	N/A	2.0X	"	"	"
ADD MODEL 556 & MODEL 740	ASM 86/88	2.7X	2.7X	3.1X	"	"
	PL/M 86/88	3.7X	3.7X	4.0X	"	"
ADD MODEL 556 I & MODEL 740	ASM 86/88	N/A	3.1X	"	"	"
	PL/M 86/88	N/A	4.0X	"	"	"

\*The current system includes this option.

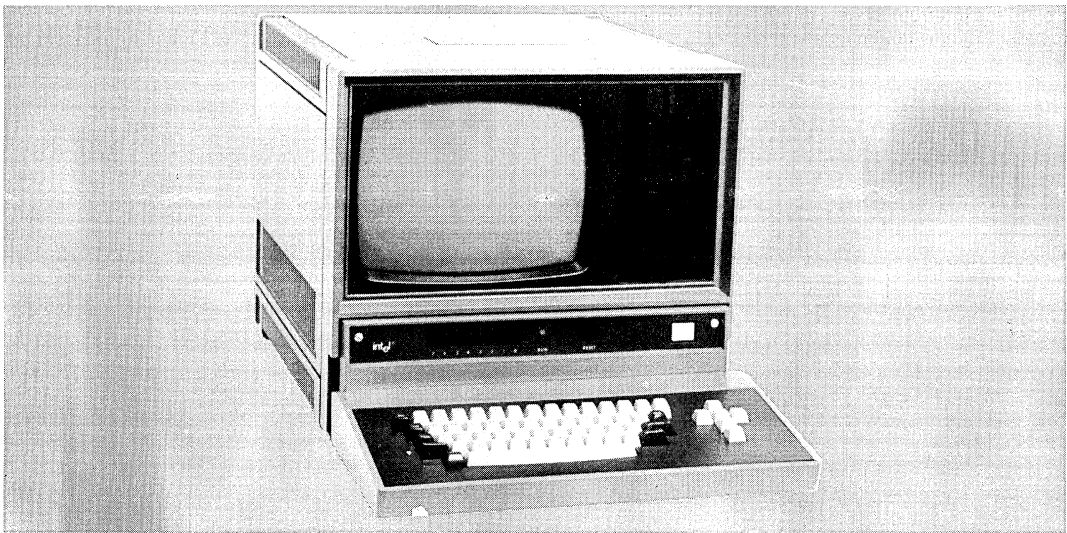
These improvement ratios are based upon Intel benchmarks using current versions of Intel software. The actual performance improvement you receive may vary depending upon your application.



## MODEL 120 INTELLEC® SERIES II MICROCOMPUTER DEVELOPMENT SYSTEM

- Complete microcomputer development system in one package for MCS®.48 and MCS-80/85 microprocessor families
- Single LSI electronics board with CPU, 32K bytes RAM memory, and 4K bytes ROM memory
- Self-test diagnostic capability
- ISIS-II disk operating system
- Built-in interfaces for high speed paper tape reader/punch, printer and Universal PROM Programmer
- Integral 250K-byte floppy disk with total storage capacity expandable to over 2M bytes
- Available with user's choice of MCS®.48 or 8080/8085 macroassembler
- Software compatible with previous Intellec® systems

The Model 120 Intellec Series II Microcomputer Development System is a complete microcomputer development system integrated into one compact package. It includes a CPU with 32K bytes of RAM memory, 4K bytes of ROM memory, a 2000-character CRT, detachable full ASCII keyboard with cursor controls and upper/lower case capability, and a 250K-byte floppy diskette drive. Powerful ISIS-II Operating System software allows the Model 120 to be used quickly and efficiently for assembling, compiling and debugging programs for Intel's MCS-48, MCS-80/85 and all other currently supported Intel 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 in conjunction with an optional in-circuit emulator (ICE™) module or the HSE-49™ High-Speed Emulator, the Model 120 provides all the hardware and software development tools necessary for the rapid development of a microcomputer-based product.



**FUNCTIONAL DESCRIPTION**

**Hardware Components**

The Intellec Series II Model 120 is a packaged, highly integrated microcomputer development system consisting of a CRT chassis with a 6-slot cardcage, power supply, fans, cables, single floppy diskette drive, and two printed circuit cards. A separate, full ASCII keyboard is connected with a cable.

**CPU Cards** — The master CPU card contains its own microprocessor, memory, I/O, interrupt, and bus interface circuitry, fashioned from Intel's high-technology LSI components. Known as the integrated processor board (IPB), 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 and floppy disk control. This card, mounted on the rear panel, also contains its own microprocessor, RAM and ROM memory, and I/O interface, thus in effect creating a dual processor environment. Known as the I/O controller (IOC), the slave CPU card communicates with the IPB over an 8-bit bidirectional data bus, thus leaving the remaining 5 slots in the cardcage available for system expansion. A block diagram of the IOC is shown in Figure 1.

**System Components**

The heart of the IPB is an Intel NMOS 8-bit microprocessor, the 8080A-2, running at 2.6 MHz. 32K bytes of RAM memory are provided on the board using Intel 16K RAMs. 4K of ROM is provided, preprogrammed with system bootstrap "self-test" diagnostics and the Intellec Series II 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**

**IPB Serial Channels** — The I/O subsystem in the Model 120 consists of two parts: the IOC card and two serial channels on the IPB 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 data terminal. One channel contains current loop adapters. Both channels are implemented using Intel's 8251 USART. They can be programmatically selected to perform a variety of I/O functions. Baud rate selection is accomplished programmatically 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 8259 interrupt controller, operating in a polled mode, nested to the primary 8259.

**IOC Interface** — The remainder of system I/O activity takes place in the IOC. The IOC provides interfaces for the CRT, keyboard, integral floppy disk and other peripherals, including a printer, high speed paper tape reader/punch, and Intel's Universal PROM Programmer. The IOC contains its own independent microprocessor, also an 8080A-2. This CPU controls all I/O operations, as well as supervising communications with the IPB. 8K bytes of ROM contain all I/O control firmware. 8K bytes of RAM are used for CRT screen refresh storage and the floppy disk buffer. These do not occupy any 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 15.5 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

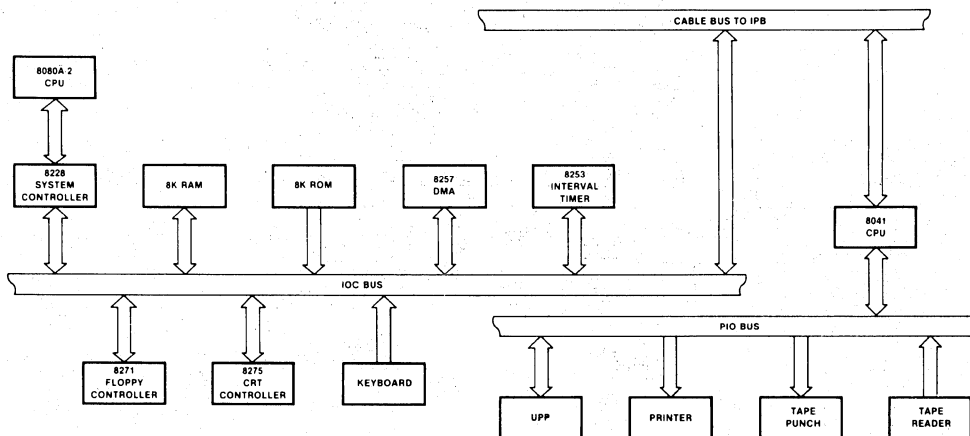


Figure 1. I/O Controller (IOC) Block Diagram for the Model 120 Intellec® Series II Microcomputer Development System

the IPB 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 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-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.

### Floppy Disk Drive

The floppy disk drive 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.

### Peripheral Interface

A UPI-41 Universal Peripheral Interface on the IOC board provides interface for other peripheral devices, including a printer, high speed paper tape reader, high speed paper tape punch, and Intel's Universal PROM

Programmer. Communication between the IPB and IOC is maintained over a separate, 8-bit bidirectional data bus. Connectors for the 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 IPB, allowing the eight interrupt switches to connect to the primary 8259A, as well as to the Intellec Series II bus.

### MULTIBUS™ Interface Capability

All Intellec Series II 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.

### Expansion

The Model 120 may be expanded up to 2.25M bytes of on-line floppy diskette storage, plus 7.3M bytes of on-line hard-disk storage capacity; system RAM may be expanded beyond the 32K bytes provided by adding iSBC memory expansion boards.

## SPECIFICATIONS

### Host Processor (IPB)

8080A-2 based, operating at 2.600 MHz.

**RAM** — 32K, expandable to 64K with iSBC 032 RAM boards (system monitor occupies 62K through 64K)

**ROM** — 4K (2K in monitor, 2K in boot/diagnostic)

**Bus** — MULTIBUS architecture, maximum transfer rate of 5 MHz

**Clocks** — Host processor, crystal controlled at 2.6 MHz; bus clock, crystal controlled at 9.8304 MHz.

### I/O Interfaces

2 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 bus; implemented for user selected DMA devices through optional DMA module — maximum transfer rate of 5 MHz.

### Memory Access Time

**RAM** — 585 ns max

**PROM** — 450 ns max

### Diskette

**Diskette System Capacity** — 250K bytes (formatted)

**Diskette System Transfer Rate** — 160K bits/sec

**Diskette System Access Time**

Track-to-Track: 10 ms max

Average Random Positioning: 260 ms max

Rotational Speed: 360 rpm

Average Rotational Latency: 83 ms max

Recording Mode: FM

### Physical Characteristics

**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. (22.0 cm)

**Weight** — 6 lb (3 kg)

**Electrical Characteristics**
**DC Power Supply**

Volts Supplied	Amps Supplied	Typical System Requirements
+ 5 ± 5%	30.0	7.5
+ 12 ± 5%	2.5	0.2
- 12 ± 5%	0.3	0.05
- 10 ± 5%	1.5	0.15
+ 15 ± 5%	1.5	1.3*
+ 24 ± 5%	1.7	1.2*

\*Not available on bus.

**AC Requirements**
**110V, 60Hz** — 5.9 Amp

**220V, 50 Hz** — 3.0 Amp

**Environmental Characteristics**
**Operating Temperature** — 16°C to 32°C (61°F to 90°F)

**Operating Humidity** — 20% to 80% relative humidity

**Equipment Supplied**

Model 120 chassis

Integrated processor board (IPB)

I/O controller board (IOC)

CRT and keyboard

250K-byte floppy disk drive

ROM resident system monitor

ISIS-II system diskette

MCS-48 macroassembler diskette (supplied with MCI-120/48-Kit)

MCS-80/MCS-85 macroassembler diskette (supplied with DS-120/80-Kit)

**Reference Manuals**
**9800558** — A Guide to Microcomputer Development Systems (SUPPLIED)

**9800559** — Intellec Series II Installation and Service Manual (SUPPLIED)

**9800306** — ISIS-II System User's Guide (SUPPLIED)

**9800556** — Intellec Series II Hardware Reference Manual (SUPPLIED)

**9800555** — Intellec Series II Hardware Interface Manual (SUPPLIED)

**9800605** — Intellec Series II System Monitor Source Listing (SUPPLIED)

**9800554** — Intellec Series II Schematic Drawing (SUPPLIED)

**9800255** — MCS-48 and UPI-41 Assembly Language Programming Manual (SUPPLIED with MCI-120/48-Kit)

**9800301** — 8080/8085 Assembly Language Programming Manual (SUPPLIED with DS-120/80-Kit)

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
MCI-120/48-Kit (110V/60 Hz)	Intellec Series II Model 120 micro-computer development system with MCS-48 macroassembler
DS-120/80-Kit (110V/60 Hz)	Intellec Series II Model 120 micro-computer development system with MCS-80/MCS-85 macroassembler
MDS-120* (110V/60 Hz)	Intellec Series II Model 120 micro-computer development system
MDS-121* (220V/50 Hz)	Intellec Series II Model 121 micro-computer development system

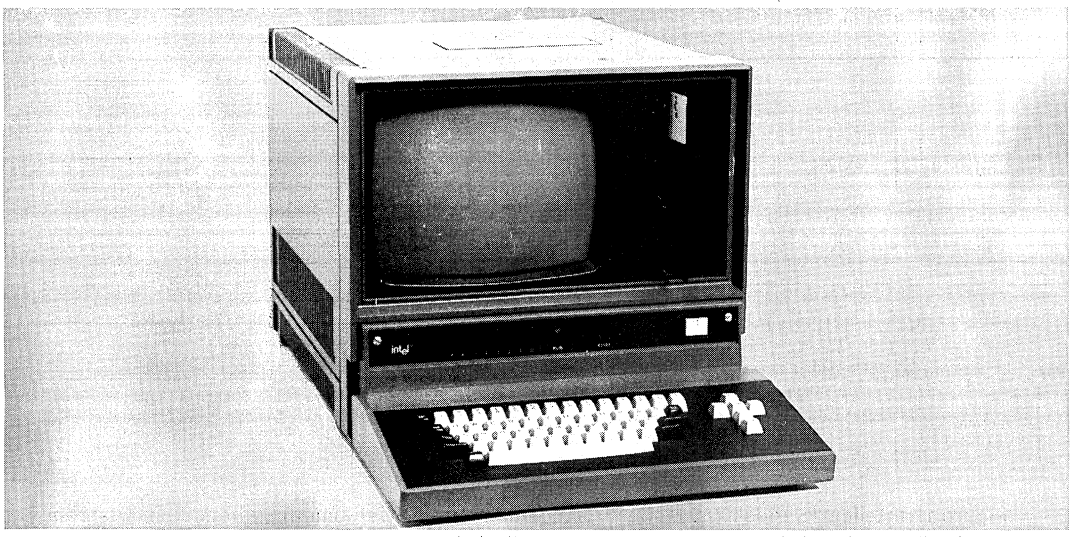
\*\*"MDS" is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.



# MODEL 225 INTELLEC<sup>®</sup> SERIES II/85 MICROCOMPUTER DEVELOPMENT SYSTEM

- Complete microcomputer development system for MCS<sup>™</sup> -86, MCS<sup>™</sup> -85, MCS<sup>™</sup> -80, and MCS<sup>™</sup> -48 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, locator, and CRT based editor CREDIT
- Supports PL/M, FORTRAN, BASIC, PASCAL and COBOL high level languages
- Software compatible with previous Intellec<sup>™</sup> 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, or MCS-48 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<sup>™</sup>) 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.



**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.

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.

**CPU Cards** — The master CPU card contains its own microprocessor, memory, I/O, interrupt and bus interface circuitry implemented with Intel's

**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.

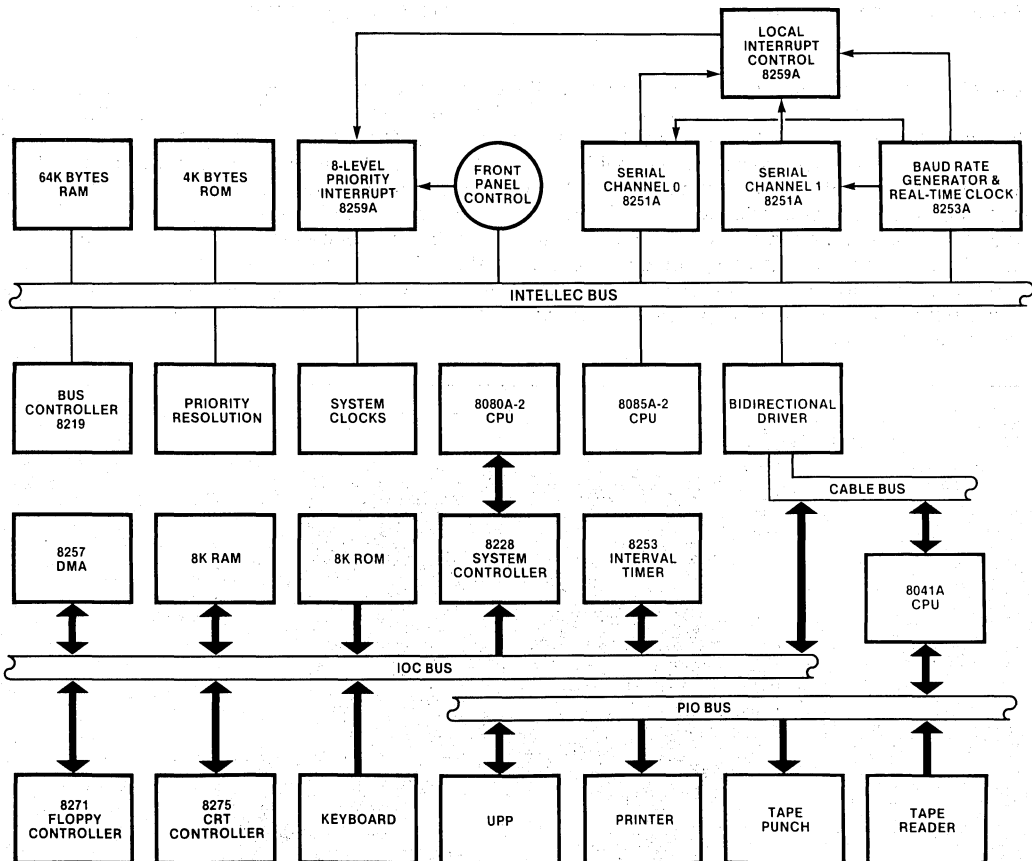


Figure 1. Intellec® Series II/85 Model 225 Microcomputer Development System Block Diagram

## 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, pre-programmed 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 con-

troller. 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)

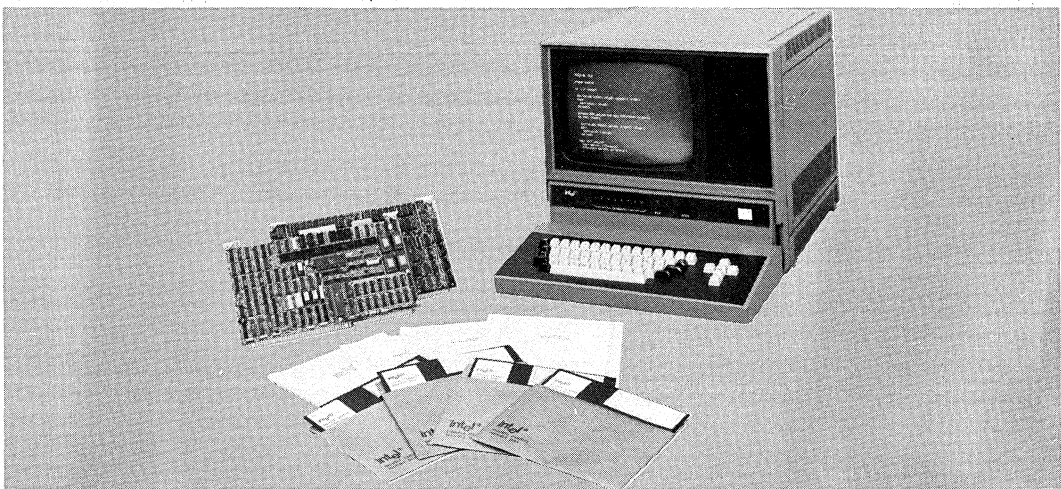
\*\*"MDS" is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.



## MODEL 286 INTELLEC® SERIES III MICROCOMPUTER DEVELOPMENT SYSTEM

- Complete 16-bit High Performance, Microcomputer Development Solution for Intel iAPX 88/86 Applications. Also Supports MCS-85™, MCS-80™ and MCS-48™ Families
- Full Range of iAPX 88/86-resident, High-level Languages: PL/M-88/86, PASCAL-88/86, and FORTRAN-88/86
- 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 88/86 Programs
- Upgradeable from Intellec® Model 800, Series II/80 and Series II/85
- Compatible with Intellec® Distributed Development Systems
- Software Compatible with Previous Intellec® Systems
- Software Applications Debugger for User iAPX 88/86 Programs
- Integral 250K Byte Floppy Disk Drive with Total Storage Capacity Expandable to Over 2M Bytes of Floppy Disk Storage and 7.3 M Bytes of Hard Disk Storage (formatted)

The Intellec® Series-III Microcomputer Development System is a high performance system solution designed specifically for iAPX 88/86 microprocessor development. It contains two host CPUs, and 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 88/86 user RAM memory, a 2000-character CRT, detachable full ASCII keyboard with cursor controls and upper/lower case capability, 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 88/86 resident software and the high-level languages PL/M-88/86, PASCAL-88/86, and FORTRAN-88/86 are also available. A ROM resident software debugger not only provides self-test diagnostic capability, but also gives the user a powerful iAPX 88/86 applications debugger.



**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.

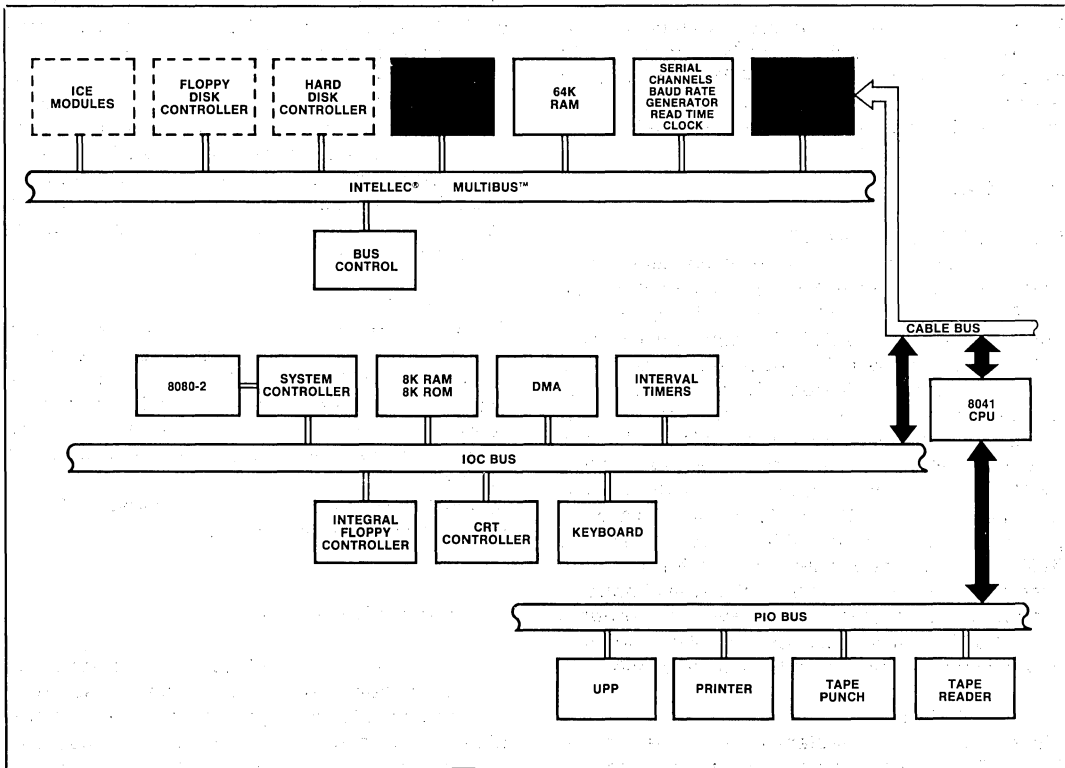


Figure 1. INTELLEC® Series III Block Diagram



## 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 intelligent 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-compatible 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-compatible 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. The controller provides an interface to the Intellec Series III system bus, as well as supporting up to four diskette drives to allow more than 2 million bytes of on-line storage.

An additional cable and connectors are also supplied to optionally convert the integral floppy disk from single density to double density.

### FLOPPY DISK CONTROLLER BOARDS

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 validates data during reads using a cyclic redundancy check (CRC) polynomial and generates CRC data during write operations. When the diskette controller requires access to the Intellec system memory, the interface board requests and maintains DMA master control of the system bus, and generates the appropriate memory command.

### 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 Mb/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.

**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<sup>2</sup> 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

**ENVIRONMENTAL CHARACTERISTICS**

System Operating Temperature—0°C to 35°C (32°F to 95°F)

Humidity—20% to 80%

**DOCUMENTATION SUPPLIED**

*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*, 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)

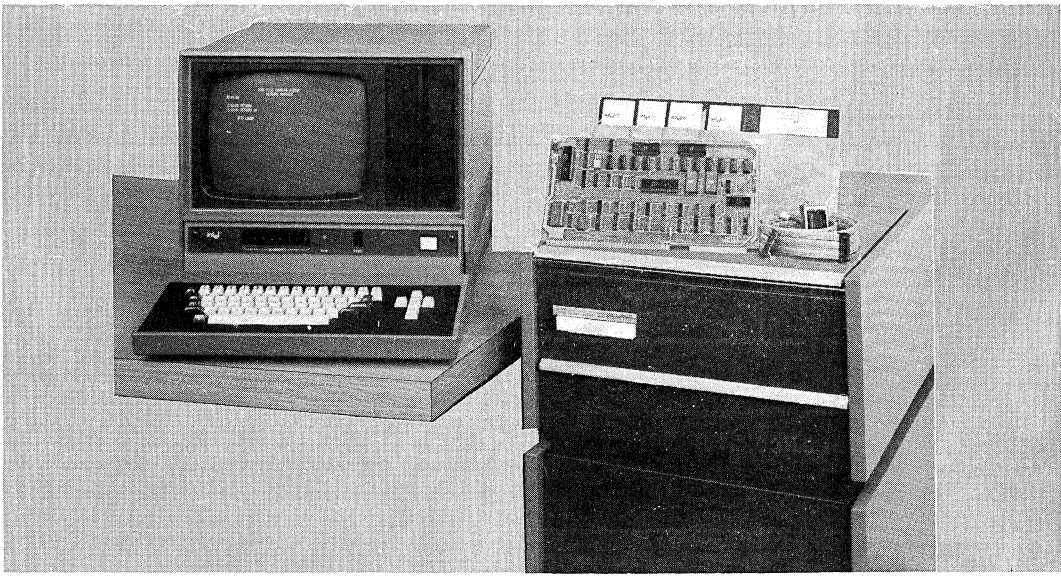
Requires Software License



## MODEL 290 NETWORK MANAGER INTELLEC® NETWORK DEVELOPMENT SYSTEM-I (NDS-I)

- Provides a distributed development system environment for Intellec® microcomputer development system users
- Supports all existing Intellec® development systems as workstations: Intellec® Model 800, Series II/80, Series II/85, and Series III models
- Functions as a Project Management tool to increase programmer productivity and coordinate large program development
- Shared background line printer
- Supports up to 8 workstations operating concurrently
- Files stored on central hard disk shared among workstations
- Provides substantial performance enhancement for floppy disk-based systems
- Distributes the costs of central mass storage among workstations
- Network Manager upgradeable from either Intellec® Model 240 or Intellec® Model 245

The Intellec® Network Development System-I (NDS-I) is designed to provide the user with tools necessary to support a distributed development system environment. NDS-I enables up to eight Intellec development systems to share both a common line printer and disk storage. Disk files may reside on either one or two central hard disks, providing up to fifteen megabytes of storage capacity. The major component of NDS-I is the Network Manager which controls all communications between the workstations and the shared disk. The powerful multitasking operating system of the Network Manager provides public/private file control for all files resident on the hard disk and printer sharing. As a project management tool, NDS-I helps coordinate the numerous program modules common to large multi-man projects. Productivity is increased and development time is shortened. An upgrade package is available to convert an existing Model 240 or 245 into a Network Manager.



**FUNCTIONAL DESCRIPTION**

**Hardware Components**

The NDS-I Network Manager consists of CRT chassis with a 6-slot cardcage, power supply, fans, cables, single floppy diskette drive, a detachable full ASCII keyboard and five printed circuit cards. A free standing pedestal houses the hard disk drive along with power supply, fans, and cables for connection to the main chassis. A block diagram of the Network Manager is shown in Figure 1.

**CPU**

The master CPU card is built around the 8085A-2 and includes 64K bytes of on board memory, I/O, interrupt and bus interface circuiting fashioned from Intel's high technology components. Known as the integrated processor card (IPC), it occupies the first slot in the cardcage. A second slave CPU card is responsible for 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. Known as the input/output controller (IOC), the slave CPU card communicates with the IPC over an 8-bit bidirectional data bus.

**DISK CONTROLLER**

The hard disk controller consists of two boards, the channel board and the interface board, mounted in the system cardcage. The channel board receives,

decodes and responds to channel commands from the 8085A-2 in the Network Manager. The interface board provides the disk controller with a means of communication with the disk drives and with the Intellec system bus. The interface board generates a cyclical redundancy check polynomial and validates data during reads using a CRC polynomial. When the disk controller requires access to Intellec system memory, the channel board requests and maintains DMA master control of the system bus, and generates the appropriate memory command. The channel board also acknowledges I/O commands as required by the Intellec bus. The disk controller supports one or two hard disk drives.

**INTERCONNECT COMMUNICATION**

An Interconnect Board (ICB), which occupies one slot in each workstation and the Network Manager cardcage, provides the communications interface between the workstations and the Network Manager. The ICB is a MULTIBUS™ bus compatible board with an 8085 microprocessor, 1.25 kbytes of RAM, 4 kbytes of ROM and seven I/O ports. The ICB accepts a command from the master CPU, executes the command without intervention by the master CPU, and signals completion by setting a flag and generating a MULTIBUS™ bus interrupt. The ICB moves data between the workstations and the Network Manager with a burst of data transfer rate of 40 kbytes per second.

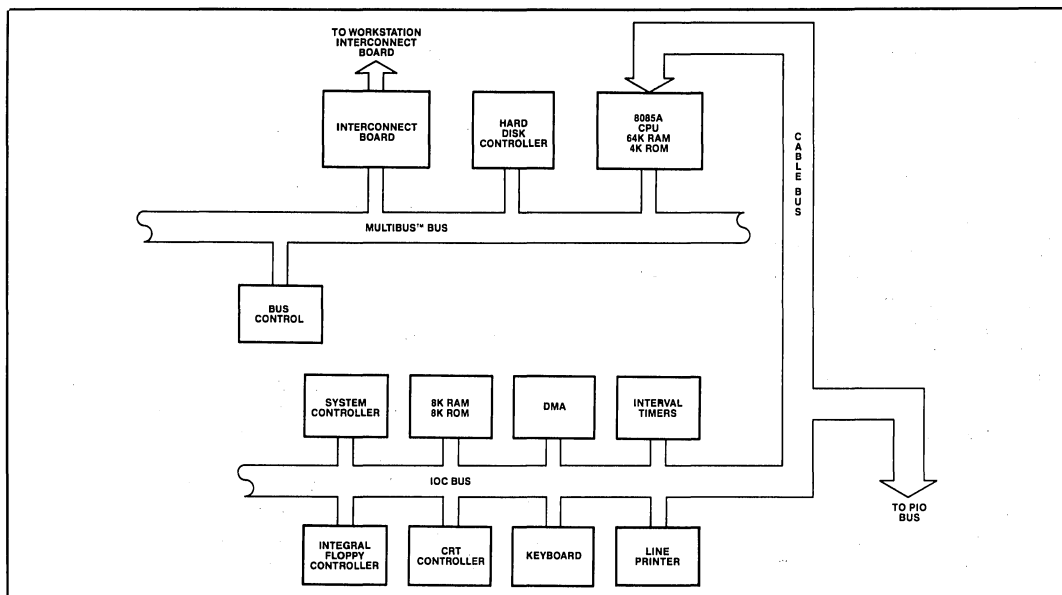


Figure 1. Intellec® NDS-I Network Manager Block Diagram

**INTEGRAL CRT**

The CRT is a 12-inch raster scan type monitor with a 50/60Hz 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 letters.

**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, buffers the characters and provides N-key rollover. 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.

**FLOPPY DISK DRIVE**

The integral single density floppy disk drive is controlled by an Intel 8271 single-chip, programmable floppy disk controller. The 8271 transfers data via an Intel 8257 DMA controller between an IOC RAM buffer and the diskette. The 8271 performs reading and writing of data, formatting diskettes, and reading status commands from the IOC processor.

**Software Components**

Included with the Model 290 is the Network Manager operating system that controls all communications between the workstations and the shared hard disk. The operating system software provides concurrent disk input/output, communications, and file management, and offers substantial workstation performance enhancements compared to standalone, floppy disk-based development systems. The Network Manager operating system provides public/private file control for all files resident on the hard disk. The public/private file control provides a useful project management tool to help coordinate the numerous programmers and modules common to large development projects.

**SPECIFICATIONS****Disk Drive**

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

**Disk System Capacity**

Per Surface—15M bits  
 Per Platter—29M bits  
 Per Drive—59M bits  
 Per Drive—7.3M bytes (formatted)

**Disk Performance**

Disk Transfer Rate—2.5M bits/sec  
 Disk System Access Time  
   Track to Track: 13 ms max  
   Full Stroke: 100 ms  
   Rotational Speed: 2,400 rpm

**Diskette**

Diskette System Capacity—250K bytes (formatted)  
 Diskette System Transfer Rate—160K bits/sec  
 Diskette System Access Time  
   Track to Track: 10 ms max  
   Average Random Positioning: 260ms  
   Rotational Speed: 360 rpm  
   Average Rotational Latency: 83 ms  
   Recording Mode: FM

**Physical Characteristics**

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)

**DISK DRIVE ON PEDESTAL**

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**
**D.C. POWER SUPPLY**

Supply Voltage	Amps Supplied	Typical System Requirements
+ 5 ±5%	30	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

**A.C. REQUIREMENTS FOR MAINFRAME AND 2 DRIVES**

110V, 60Hz—16 Amp (Mainframe =5.9 Amp)  
 (Drive =5.0 Amp)  
 220V, 50Hz—8.6 Amp (Mainframe =3.1 Amp)  
 (Drive =3.0 Amp)

**Environmental Characteristics**

Operating Temperature—16°C to 32°C (90°F)  
 Humidity—20% to 80%

**Equipment Supplied**

Model 225 Chassis  
 Integrated Processor Card  
 I/O Controller Board  
 CRT and Keyboard  
 Model 740 Hard Disk Drive  
 Two Hard Disk Controller Boards with Cables  
 Disk Cartridge  
 NDS-I Interconnect Board with Cable  
 Two NDS-I Line Terminators  
 ROM Resident System Monitor  
 Network Manager Operating System  
 Network Manager Diagnostics

**Reference Manuals**

*Hard Disk Subsystem Operation and Checkout Manual, 9800943.*  
*Network Manager Console Operating Instructions, 121645.*

**Optional Equipment**

Model 595 NDS-I Workstation Interconnect  
 Package: Includes Interconnect board and workstation software to convert any Model 800, Series II/80, Series II/85, or Series III to a NDS-I workstation.

Model 596 NDS-I 10 ft. Interconnect Cable

Model 597 NDS-I 20 ft. Interconnect Cable

Model 743 (110V) Add on hard disk unit with cables and disk cartridge.

Model 744 (220V) Add on hard disk unit with cables and disk cartridge.

Model 746 Box of 5 blank hard disk cartridges.

**ORDERING INFORMATION**
**Part Number Description**

DS-290 (110V) NDS-I Network Manager: Includes  
 DS-291 (220V) network console with hard disk subsystem, interconnect board, internal cable, and Network Manager software.



1. The first part of the document discusses the importance of maintaining accurate records of all transactions and activities. It emphasizes that proper record-keeping is essential for ensuring transparency and accountability in financial operations.

2. The second part of the document outlines the various methods and techniques used to collect and analyze data. It highlights the need for consistent and reliable data collection processes to ensure the validity of the results.

3. The third part of the document focuses on the analysis and interpretation of the collected data. It discusses the various statistical and analytical tools used to identify trends and patterns in the data.

4. The fourth part of the document discusses the implications and conclusions drawn from the analysis. It highlights the key findings and their potential impact on the organization's operations and decision-making.

5. The fifth part of the document provides a summary of the overall findings and recommendations. It emphasizes the need for continuous monitoring and evaluation to ensure the effectiveness of the implemented measures.

6. The sixth part of the document discusses the challenges and limitations encountered during the study. It highlights the need for further research and development to address these challenges and improve the overall quality of the data and analysis.

7. The seventh part of the document provides a detailed description of the methodology used in the study. It outlines the steps and procedures followed to ensure the accuracy and reliability of the data collection and analysis.

8. The eighth part of the document discusses the ethical considerations and safeguards implemented during the study. It emphasizes the importance of protecting the privacy and confidentiality of the data and ensuring the integrity of the research process.

9. The ninth part of the document provides a detailed description of the data sources and collection methods. It outlines the various channels and techniques used to gather the data, ensuring a comprehensive and representative sample.

10. The tenth part of the document discusses the overall findings and conclusions of the study. It highlights the key insights and their implications for the organization's future operations and decision-making.



*Microcomputer  
Development Systems  
Options*

---

**12**

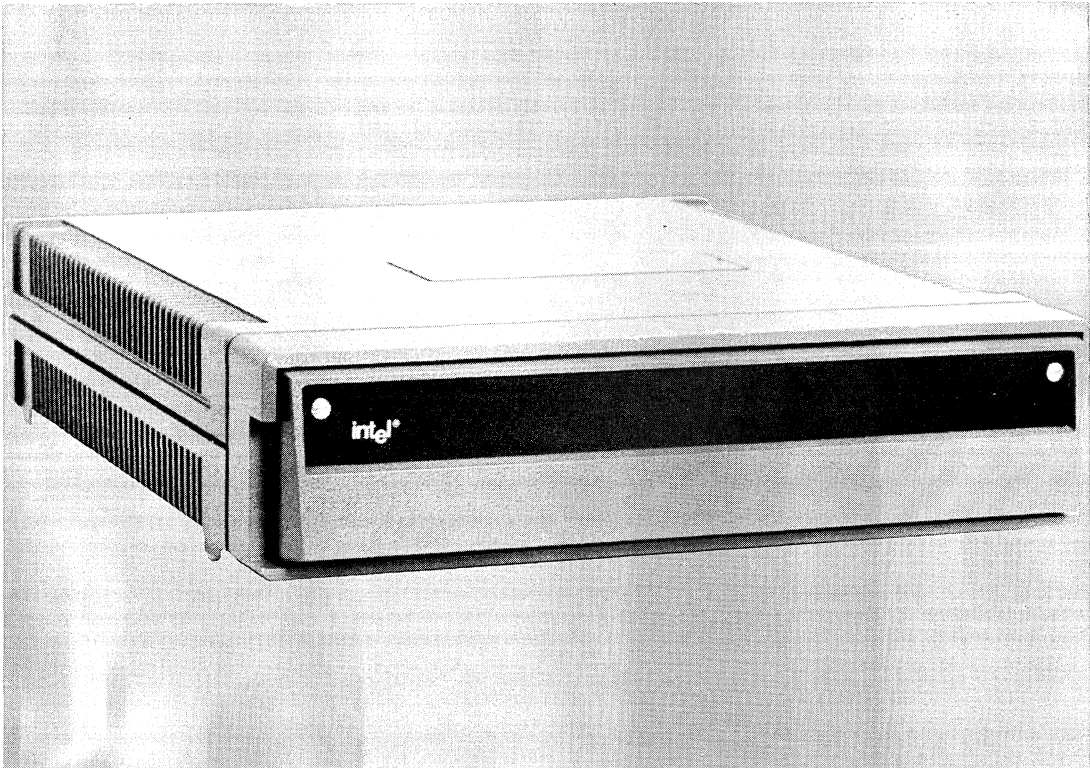




## EXPANSION CHASSIS INTELLEC® SERIES II MICROCOMPUTER DEVELOPMENT SYSTEM

- 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 micro-computer 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.



# EXPANSION CHASSIS

## 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

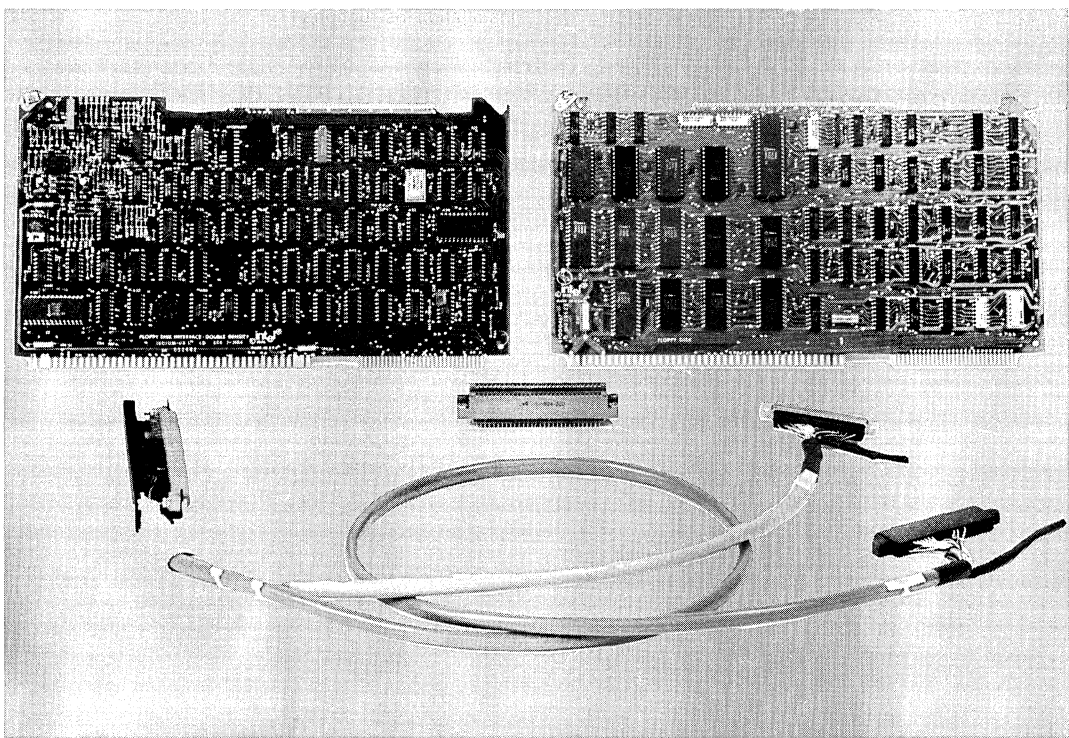
\*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Science.



## 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 compatibility 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.



# MODEL 503 DOUBLE DENSITY UPGRADE KIT

---

## 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 M<sup>2</sup>FM  
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.

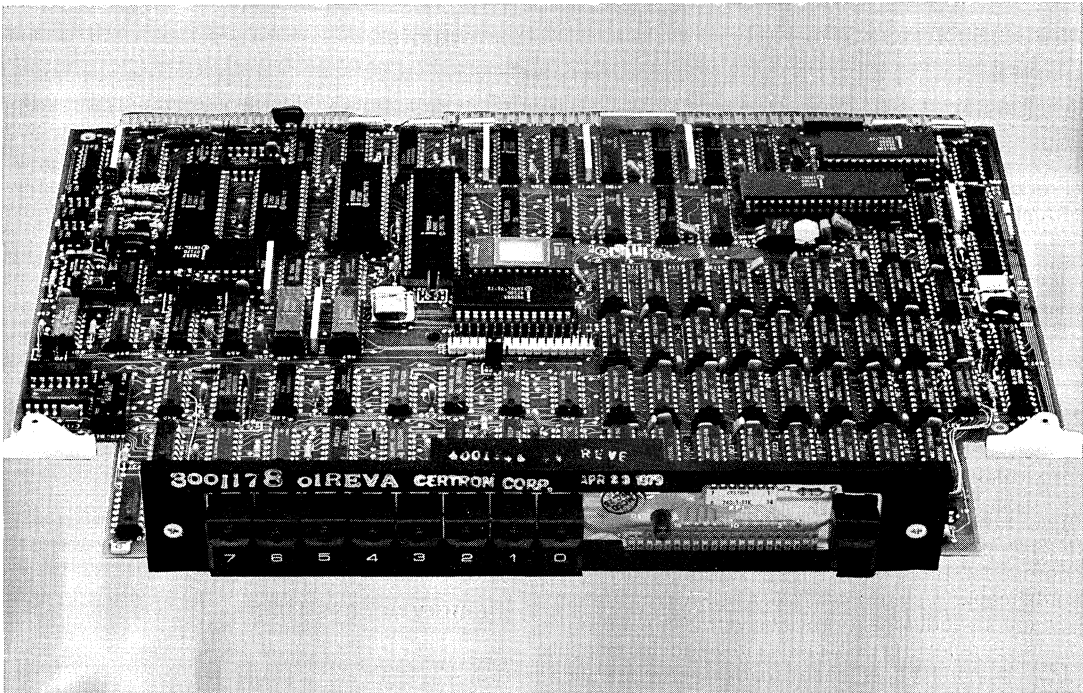
---



## MODEL 505 INTEGRATED PROCESSOR CARD

- Single Electronic Board with 8085A-2 CPU, 64K Bytes RAM and 4K Bytes ROM
- Increases Card Slot Availability on the Inteltec® Series II Microcomputer Development System
- Fully Software Compatible with the 8080-Based Integrated Processor Board
- High Performance 8085A-2 Based Integrated Processor Card Upgrades Inteltec® Series II Microcomputer Development Systems to the Performance of Series II/85 Systems
- Additional Functions Available Through MULTIBUS™ Interface—
  - Local Interrupt Controller
  - Programmable Interval Timer
  - Two Channels of USARTs

The Inteltec® Series II/85 Model-505 Integrated Processor Card (IPC) is a single board upgrade for all 8080-based Inteltec Series II Microcomputer Development Systems. The IPC is an 8085A-2 based CPU board which contains 64K bytes of RAM, 4K bytes of ROM, two 8259 interrupt controllers, two 8251 USARTs, an 8253 interval timer, MULTIBUS™ compatible interface, and special interfaces to the Inteltec Series II Microcomputer Development System. The IPC is fully software compatible with the 8080-based integrated processor board.





## FUNCTIONAL DESCRIPTION

### Hardware 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 RAM chips. 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.

### SYSTEM IMPROVEMENTS

The IPC provides two main advantages over its predecessor, the 8080-based Integrated Processor Board (IPB). The first advantage is the higher processing speed. The second advantage is that it provides twice the memory.

**Higher Speed**—The IPB uses an 8080A-2 microprocessor with a 23.4000 MHz crystal to derive a clock cycle time of 384.6nsec, and executes with one wait state on memory read cycles, and 2 wait states on memory write cycles. The IPC uses an 8085A-2 microprocessor with an 8.0000 MHz crystal to derive a clock cycle time of 250nsec, and runs with the same number of wait states as the IPB. The IPC thus provides an increase in processing speed of approximately 54% over the IPB. The overall system throughput improves correspondingly. However, the amount of improvement is a function of the type and the length of the programs being executed, as well as the type of storage devices attached to the system.

**Saves One Card Slot Space**—The IPC contains an additional 32K bytes of RAM over the IPB, to provide a total of 64K bytes of system memory on one card. The increased on-board RAM size frees up one slot in the card cage of the Series II. For a typical user of ICE-86™ or ICE-88™ emulators, this additional slot will eliminate the need to purchase an expansion chassis.

## SPECIFICATIONS

### Host Processor (IPC)

**Processor**—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

Eight-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

**ELECTRICAL CHARACTERISTICS**

**DC Power Supply**

Voltage Requirements (Volts)	Worst Case Current Requirements (Amperes)
+ 5 ± 5%	4.2
+ 12 ± 5%	0.5
- 12 ± 5%	0.2
- 10 ± 5%	0.02

**ENVIRONMENTAL CHARACTERISTICS**

Operating Temp.:	Board Level	5° - 55°C (41°F - 131°F)
	System Level	16° - 32°C (61°F - 90°F)
Humidity:		20% to 80%

**EQUIPMENT SUPPLIED**

8085 Based Integrated Processor Card (IPC)

**DOCUMENTATION SUPPLIED**

*Intellec® Series II Model 22X/23X Installation Manual, 9800559*

*Intellec® Series II Hardware Reference Manual, 9800556*

*Intellec® Series Monitor Source Listing, 9800605*

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 No.	Description
MDS-505*	Integrated Processor Card upgrade package for Intellec® Series II Microcomputer Development System (110V/60Hz or 220V/50Hz). Upgrades Models 220, 221, 230, 231, 240, and 241 to the performance of the 8085A-2 based systems.

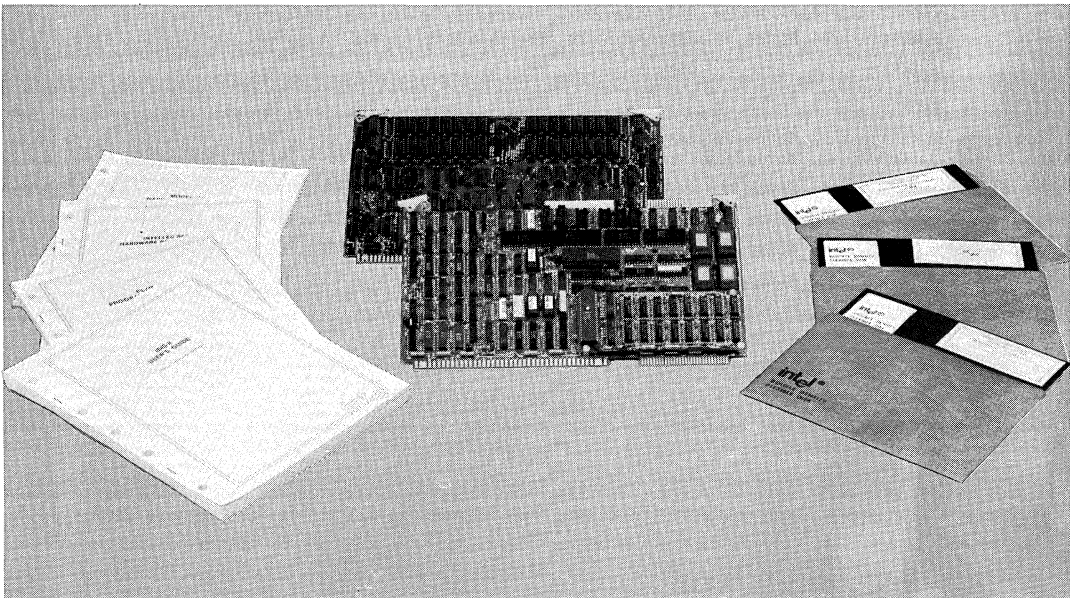
\*"MDS" is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.



## MODEL 556 iAPX 86 RESIDENT PROCESSOR BOARD PACKAGE

- High Performance 8086-Based CPU Board for Increased Intellect® Development System Performance and iAPX 86/88 Development Environment
- Upgrades Intellect® 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, Locator and Librarian
- Dual-Processor Disk Operating System Software with CRT-based Editor

The Model 556 is a performance enhancement package for Intellect® Series II/80, Series II/85 and Model 800 (requires Model 556E) 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, Locator and Librarian; software applications debugger for iAPX 86/88 user programs; and complete user documentation.



## 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 microprocessors. 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 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 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**

*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/80, Series II/85 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.
MDS-556E*	Performance package for Intellec Model 800 Microcomputer Development Systems. Specifically designed for iAPX 86/88 microprocessor development. Upgrades Model 800 to the functionality of an Intellec Series III.

\*"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.



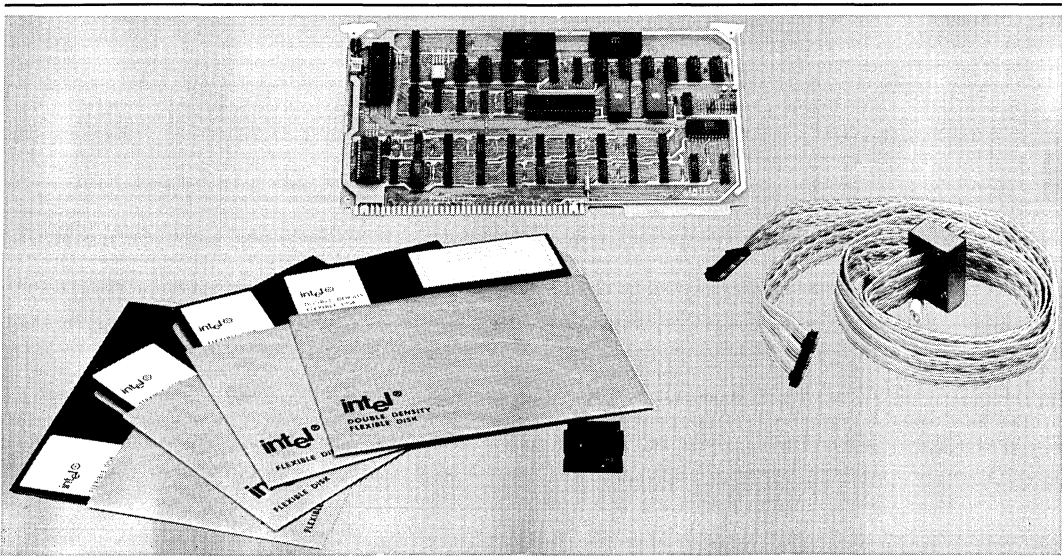
## MODEL 590 NETWORK MANAGER UPGRADE PACKAGE INTELLEC® NETWORK DEVELOPMENT SYSTEM-I (NDS-I)

- Upgrades either Intellec® Model 240 or Intellec® Model 245 to a NDS-I Network Manager
- Provides a distributed development system environment for Intellec® microcomputer development system users
- Supports all existing Intellec® development systems as workstations: Intellec® Model 800, Series II/80, Series II/85, and Series III models
- Shared background line printer
- Functions as a Project Management tool to increase programmer productivity and coordinate large program development
- Supports up to 8 workstations operating concurrently
- Files stored on central hard disk shared among workstations
- Provides substantial performance enhancement for floppy disk-based systems
- Distributes the cost of central mass storage among workstations

The Intellec® Network Development System (NDS-I) is designed to provide the user with tools necessary to support a distributed development system environment. NDS-I enables up to eight Intellec development systems to share both a common line printer and disk storage. Disk files may reside on either one or two central hard disks, providing up to fifteen megabytes of storage capacity.

The major component of NDS-I is the Network Manager which controls all communications between the workstations and the shared disk. The powerful multitasking operating system of the Network Manager provides public/private file control for all files resident on the hard disk. Printer sharing operates as a background task concurrent with other NDS-I operations.

As a project management tool, NDS-I helps coordinate the numerous program modules common to large multi-man projects. Productivity is increased and development time is shortened. This upgrade package converts an existing Model 240 or 245 into a NDS-I Network Manager.



**COMPONENTS**

The NDS-I Network Manager upgrade package consists of an interconnect board, internal cable, line terminators, and Network Manager software. A block diagram of the Network Manager is shown in Figure 1.

**Interconnect Communications**

An Interconnect Board (ICB), which occupies one slot in the Network Manager and each workstation, provides the communications interface between the workstations and the Network Manager. The ICB is a MULTIBUS™ compatible board with an 8085 microprocessor, 1.25 kbytes of RAM and 4 kbytes of ROM. The ICB accepts a command from the master CPU, executes the command without intervention by the master CPU, and signals completion by setting a flag and generating a Multibus interrupt. The

ICB moves data between the workstations and the Network Manager with a burst of data transfer rate of 40 kbytes per second.

**Software Components**

Included with the Model 590 is the Network Manager operating system which controls all communications between the workstations and the shared hard disk. The operating system software provides concurrent disk input/output, communications, and file management. It offers substantial workstation performance enhancement compared to standalone, floppy disk-based development systems. The Network Manager operating system provides public/private file control for all files resident on the hard disk. The public/private file control is a useful project management tool to help coordinate the numerous programmers and modules common to large development projects.

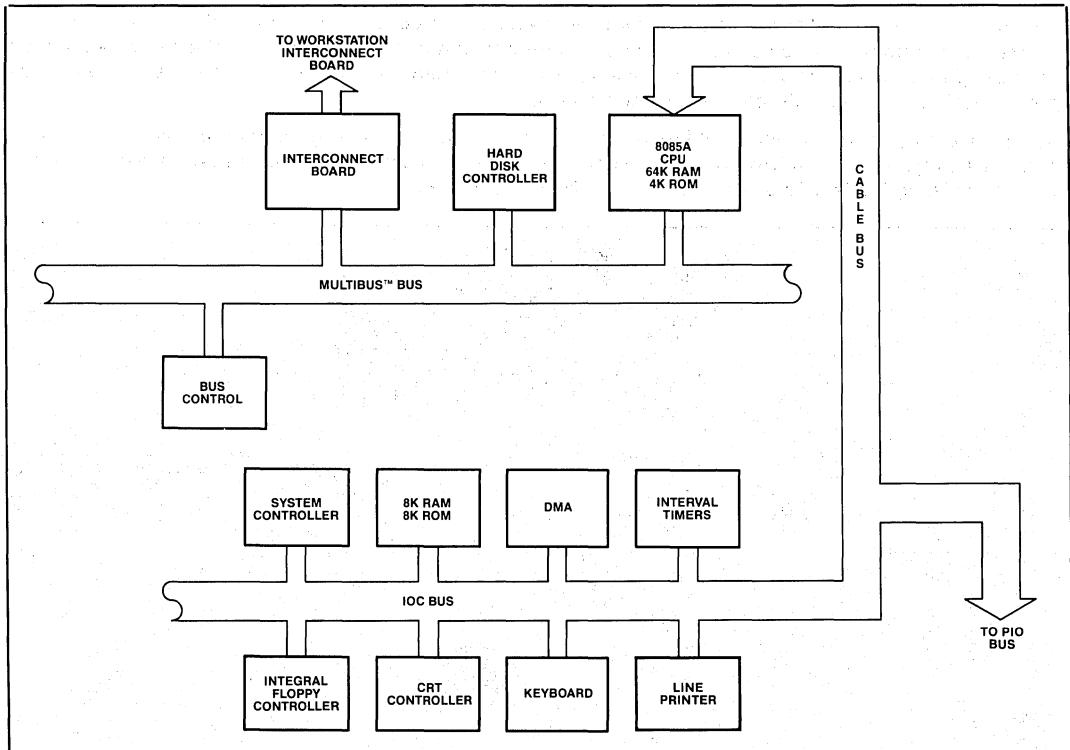


Figure 1. Intellec® NDS-I Network Manager Block Diagram

**SPECIFICATIONS****Equipment Supplied**

Interconnect Board  
Internal Cable  
Two NDS-I Line Terminators  
Network Manager Operating System  
Network Manager Diagnostics

**Optional Equipment**

Model 595 NDS-I Workstation Interconnect  
Package: Includes Interconnect board  
and workstation software to convert any  
Model 800, Series II/80, Series II/85, or  
Series III to a NDS-I workstation.

Model 596 NDS-I 10 ft. Interconnect Cable

Model 597 NDS-I 20 ft. Interconnect Cable

Model 743 (110V) Add on hard disk unit with cables  
and disk cartridge.

Model 744 (220V) Add on hard disk unit with cables  
and disk cartridge.

Model 746 Box of 5 blank hard disk cartridges.

**Reference Manuals**

*Network Manager Console Operating Instructions*,  
121645.

---

**ORDERING INFORMATION****Part Number Description**

MDS*-590	NDS-I Network Manager Upgrade Package: Includes interconnect board, internal cable, and Network Manager software.
----------	--

\*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.



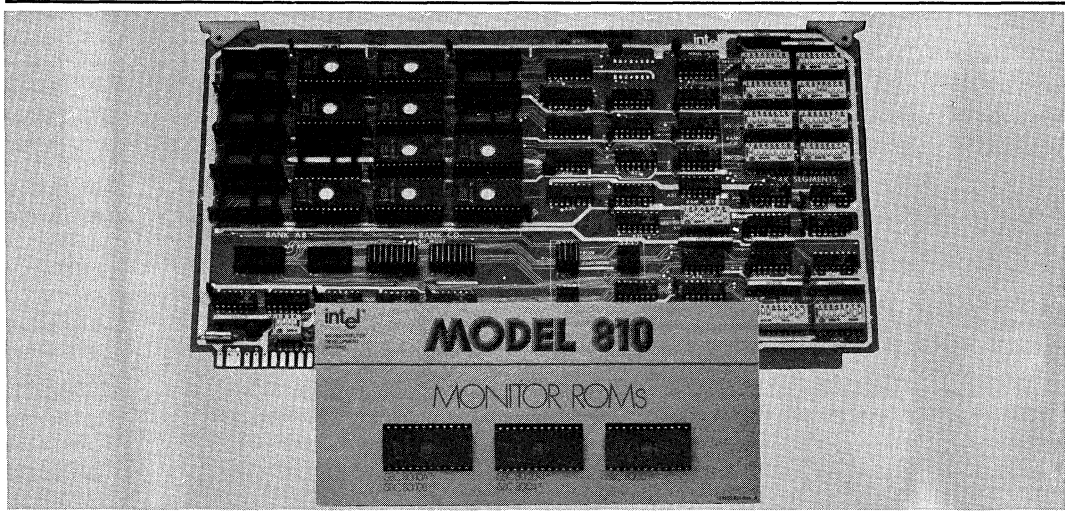


## MODEL 810 SOFTWARE DEVELOPMENT MODULE

- Adds text entry, assembly, and debug to iSBC™ configurations.
- ROM-based Editor, Assembler, and Monitor provide resident software development and debugging.
- User selects standard chassis and Intel CPU board to host development software module.
- Symbolic Assembler produces absolute code for immediate loading and execution.
- Low-cost, software development capability for Intel iSBC™ 80/10B, 80/24, and 80/30 Single Board Computers.
- Development software fits easily into user-defined Intel 8080/8085 iSBC™ configuration by residing on standard iSBC™ 464 Memory Expansion Board.
- Monitor debugger modifies and dumps memory and registers; sets breakpoints.

The Model 810 Software Development Module provides complete software development capability for Intel single board computers. By inserting the standard iSBC™ memory expansion board into an Intel 8080/8085 user-defined iSBC system, you gain text entry, assembly and debug capability in your application system. (See figure 1.) After using the Model 810 to develop and debug your application, the board may be left in the system to be re-used or removed to give a complete, self-contained single board computer application.

The 8080/8085 Symbolic Assembler is a subset of Intel's full 8080/8085 Assembler used on Intellec® Microcomputer Development Systems. Source code is entered from a user-supplied console (ASR 33 Teletype or equivalent), and debugging is done by a ROM-based Monitor for either the iSBC 80/10B, 80/20-4, 80/24, or 80/30 Single Board Computer. After assembly and execution, completed programs may be punched to paper tape for archiving and later programmed into PROM.



## FUNCTIONAL DESCRIPTION

### System Environment

The Software Development Module lets you put together your own single-board-computer-based development system with standard off-the-shelf Intel products: an Intel CPU board, chassis and the Model 810 Software Development Module. The system is ideal for low-cost editing, assembling and debugging of small program modules.

## HARDWARE COMPONENTS

The Model 810 is a modified iSBC 464 Memory Expansion Board that includes nine 2K-byte ROM's containing the Text Editor and 8080/8085 Assembler. Also supplied are three separate 2K-byte ROM's, each containing a debug monitor for one of the three variants of CPU boards supported. The three Monitor ROM's are included to allow selection of the appropriate monitor for the processor board in the user's system. The user must also provide a chassis, 16K RAM (iSBC 016 or on-board 80/30 memory), an ASR 33 Teletype or equivalent, an iSBC 530 Teletype Adapter (for iSBC 80/20-4, 80/24, and 80/30 Single Board Computers), and an iSBC 955 (or equivalent) RS232C Serial I/O Cable Set.

## SOFTWARE COMPONENTS

The Model 810 has three software components:

- Text Editor allows you to enter and update your programs from the TTY terminal; and when you have completed your editing session, your program may be saved on paper tape. All Editor commands are single character to save typing time.
- Assembler allows symbolic program development. The Assembler translates the source code into object code for execution. The object code may be stored on paper tape or in RAM for intermediate testing.
- Debug Monitor controls input and output and provides you with debugging capability. It allows you to examine register and memory locations and interrupt the program at user-specified breakpoints. The three monitor variants are: 1) iSBC 80/10B (8080 CPU), 2) iSBC 80/20-4 (8080 CPU), iSBC 80/24 (8085 CPU), 3) iSBC 80/30 (8085 CPU) Single Board Computers.

## INPUT/OUTPUT

The standard interface enables input/output of data from the TTY keyboard, printer, and paper tape read/punch.

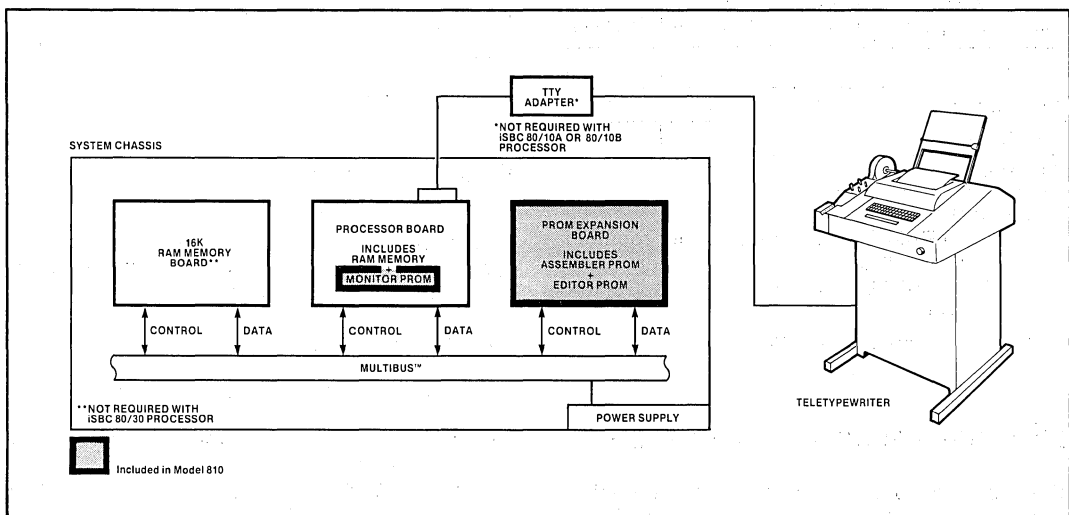


Figure 1. Model 810 Software Development Module in Target System

## MEMORY

The user must provide a minimum of 16K of RAM (iSBC 016 RAM Board, or on-board iSBC 80/30 RAM). All I/O requests are serviced through a common RAM-board jump table stored in RAM. The user may redefine symbol table buffer lengths in RAM to fit a specific application.

## SYSTEM TEST

A Confidence Test is provided to perform a checksum test of Monitor, Editor, and Assembler ROM's, and a read/write test of system RAM.

## SPECIFICATIONS

**ROM:** 20K (2K Monitor, 4K Editor, 14K Assembler)

**Bus:** Multibus interface compatible

## HARDWARE REQUIRED

Choice of iSBC single board computer and iSBC chassis as follows:

Single Board Computer	iSBC 660 Chassis	iSBC 655 Chassis	iCS 80 Chassis
iSBC 80/10B	X	X	X
iSBC 80/20-4	X	X	X
iSBC 80/24	X	X	X
iSBC 80/30	X	X	X

Also required:

iSBC 016 RAM Expansion Board (not required with iSBC 80/30 Single Board Computer)

iSBC 955 RS232C Serial I/O Cable Set (or equivalent)

iSBC 530 Teletype Adapter

## OPTIONAL HARDWARE SUPPORTED

iSBC 80/10A Single Board Computer—interchangeable with iSBC 80/10B Single Board Computer

iSBC 032 RAM Expansion Board (32K RAM)

## MONITOR COMMANDS

Command	Operation
Memory Control:	
Display	Prints contents of specified range of memory.
Insert	Inserts new data into memory.
Move	Copies contents of specified portions of memory into designated RAM locations.
Substitute	Modifies RAM on a byte-by-byte basis.
Register Control:	
X	Prints and allows you to modify contents of specified register(s).
Paper Tape I/O Commands:	
Read	Reads paper tape from reader and loads data into memory at specified location.
Write	Punches on tape the contents of specified memory area.
Program Execution Command:	
Execute (Go)	Transfers control to your program at a specified address. One or two break-points may optionally be set.

## I/O INTERFACES

iSBC 80/10A and iSBC 80/10B boards provide 20 mA current loop for TTY. iSBC 80/20-4, iSBC 80/24, and iSBC 80/30 boards require iSBC 530 TTY Adapter to convert RS232C to 20 mA current loop.

## PHYSICAL CHARACTERISTICS

Weight 294 gm (10.7 oz)  
 Length 30.48cm (12 in)  
 Height 17.15cm (6.75 in)  
 Depth 1.27cm (0.5 in)

## ELECTRICAL CHARACTERISTICS

DC Power (max.)  
 Voltage: +5V DC  $\pm$ 5%  
 Current: 2.0 amps max.

## EQUIPMENT SUPPLIED

Model 810: iSBC 464 board with nine ROM's containing the Text Editor and Assembler, and three loose ROM's containing the Monitors (one ROM for each CPU board monitor variant).

## REFERENCE MANUALS

The following material is shipped with the product:

121581 *Model 810 Software Development Module User's Guide*  
 121582 *Model 810 Software Development Module Installation Manual*  
 9800301 *8080/8085 Assembly Language Programming Manual*

The following manuals for the single board computers and chassis used in the target system can be ordered from Intel Corporation Literature Department.

9800230	<i>iSBC 80/10A™ Single Board Computer Hardware Reference Manual</i>	9800488	<i>iSBC 032™ Random Access Memory Board Hardware Reference Manual</i>
9803119	<i>iSBC 80/10B™ Single Board Computer Hardware Reference Manual</i>	9800643	<i>iSBC 464™ PROM/ROM Board Hardware Reference Manual</i>
9800317	<i>iSBC 80/20-4™ Single Board Computer Hardware Reference Manual</i>	9800708	<i>iSBC 604/614™ Cardcage Hardware Reference Manual</i>
142648	<i>iSBC 80/24™ Single Board Computer Hardware Reference Manual</i>	9800298	<i>iSBC 635™ Power Supply Hardware Reference Manual</i>
9800611	<i>iSBC 80/30™ Single Board Computer Hardware Reference Manual</i>	9800803	<i>iSBC 640™ Power Supply Hardware Reference Manual</i>
9800279	<i>iSBC 016™ 16K RAM Expansion Board Hardware Reference Manual</i>	9800709	<i>iSBC 655™ System Chassis Hardware Reference Manual</i>
		9800505	<i>iSBC 660™ System Chassis Hardware Reference Manual</i>
		9800799	<i>iCS 80™ Industrial Chassis Hardware Reference Manual</i>
		9800798	<i>iCS 80™ Industrial System and Installation Guide</i>

## ORDERING INFORMATION

Part No.	Description
MDS-810*	Software Development Module. ROM based editor, symbolic assembler, and debug monitor provide complete development support when combined with user's target iSBC 8080/8085 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.

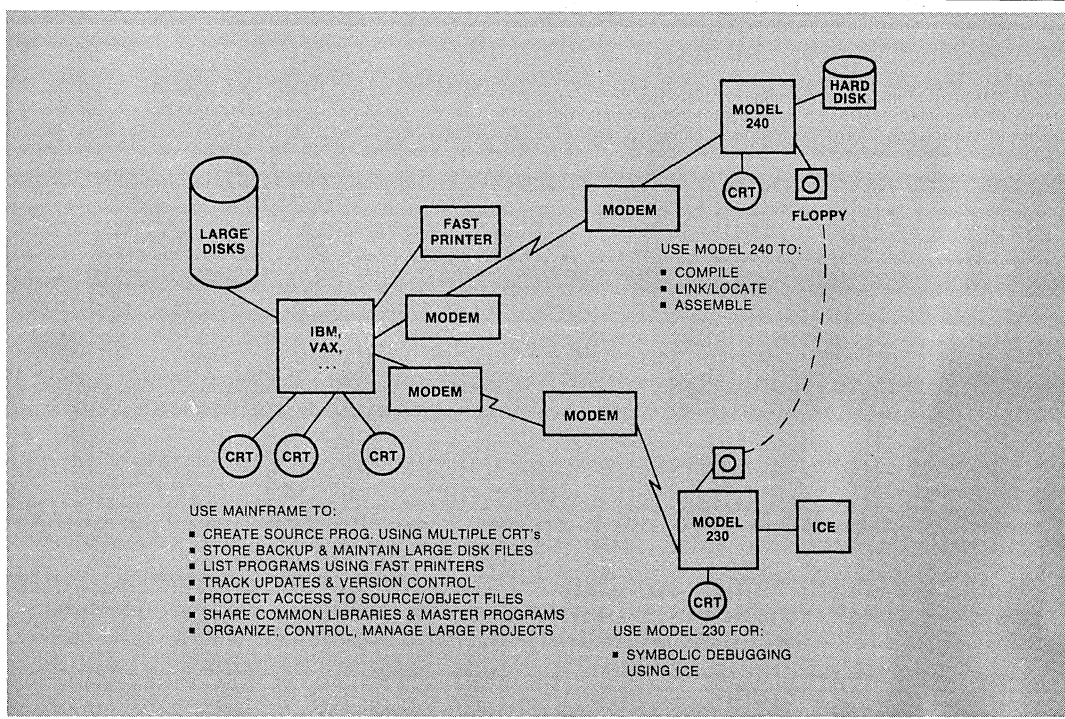


## MAINFRAME LINK FOR DISTRIBUTED DEVELOPMENT

- 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.



**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**

<b>Part Number</b>	<b>Description</b>
*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.



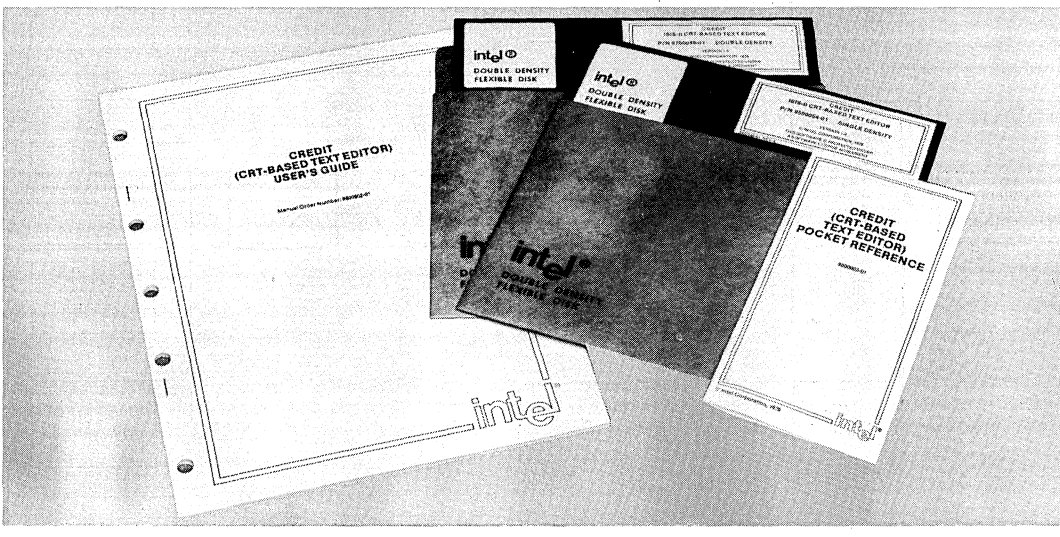
# CREDIT™ CRT-BASED TEXT EDITOR MICROCOMPUTER DEVELOPMENT SYSTEMS

- Provides Interactive Editing of ASCII Text Files
- Displays Full Page of Text
- CRT Screen Display with Cursor-Based Editing Using Single Character Commands for Insertion, Deletion, Page Forward and Backward
- Dynamic Macro Command Definition
- Command Line Editing with String Search, Deletion, Insertion and Move
- 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.





## 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

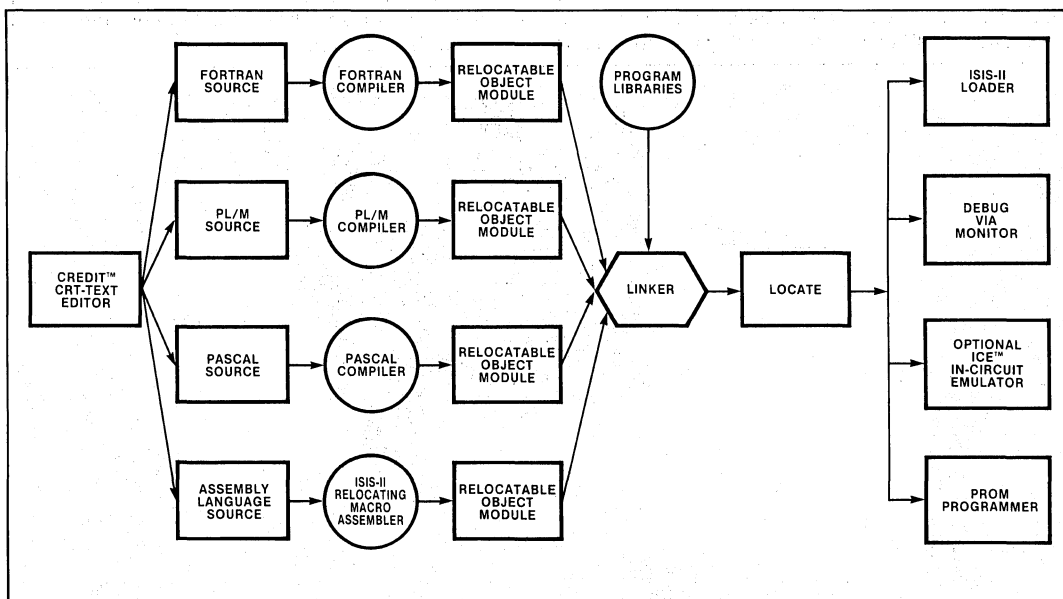
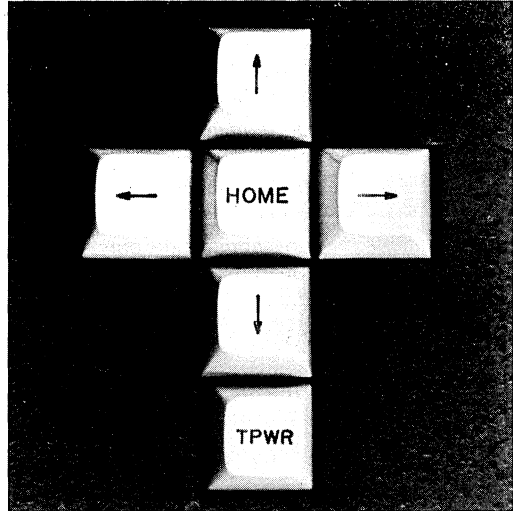


Figure 1. Microcomputer Program Development

**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**

- |  |   |
|--|---|
| <ul style="list-style-type: none"> <li>HELP: Display summary of commands.</li> <li>PRINT: Print n lines or up to tag.</li> <li>JUMP: Move cursor position n characters or to tag.<br/>Move cursor position n lines forward or backward.</li> <li>MOVE: Transfer Copy block of text from tag1, for n lines or through tag2, to cursor position.<br/>Transfer move: like Transfer Copy but the old copy is deleted.</li> </ul> | <ul style="list-style-type: none"> <li>TAGS: Set tag n, n = 0-9. Tag n is referenced as Tn.</li> <li>EXIT: Normal exit.<br/>Exit Quit: Abandon any changes to edit file.</li> <li>INSERT: Insert before CP all text between delimiters.</li> <li>DELETE: Delete n characters, or characters up to tag.<br/>Delete n lines forward or backward.</li> <li>FIND: Search for text; move pointer if found.</li> <li>SUBST: newtext replaces oldtext if oldtext is found. (Optional query to user before replacement.)</li> </ul> |
|--|---|

**ADVANCED EDITING COMMANDS**

- |   |   |
|---|---|
| <ul style="list-style-type: none"> <li>MACROS: Define a macro.<br/>Delete a macro, or all macros if name="".<br/>Expand and execute macro contents, command mode.<br/>Expand and execute macro contents, screen mode.<br/>Print names and definitions of all macros.</li> </ul> | <ul style="list-style-type: none"> <li>FILES: Open file "filename" for Reading or Writing.<br/>Close the current external Read (Write) file.<br/>Go to beginning of current Read file.<br/>Read and insert n lines from the Read file.<br/>Write n lines to the external Write file.</li> </ul> |
|---|---|

---

GET:	Get contents of file into command line.		Do command only if Yes (Search) Flag is False.
QUERY:	Query User: set Query Flag accordingly. Do command only if Query Flag is True. Do command only if Query 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.
YES:	Do command only if Yes (Search) Flag is False.	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 and Hard Disk  
Systems*

**13**

---

1. The first part of the document discusses the importance of maintaining accurate records of all transactions and activities. It emphasizes that this is crucial for ensuring transparency and accountability in the organization's operations.

2. The second part of the document outlines the various methods and tools used to collect and analyze data. It highlights the need for consistent and reliable data collection processes to support informed decision-making.

3. The third part of the document focuses on the role of technology in data management and analysis. It discusses how modern software solutions can streamline data collection, storage, and reporting, thereby improving efficiency and accuracy.

4. The fourth part of the document addresses the challenges associated with data management, such as data quality, security, and privacy. It provides strategies to mitigate these risks and ensure that data is used responsibly and ethically.

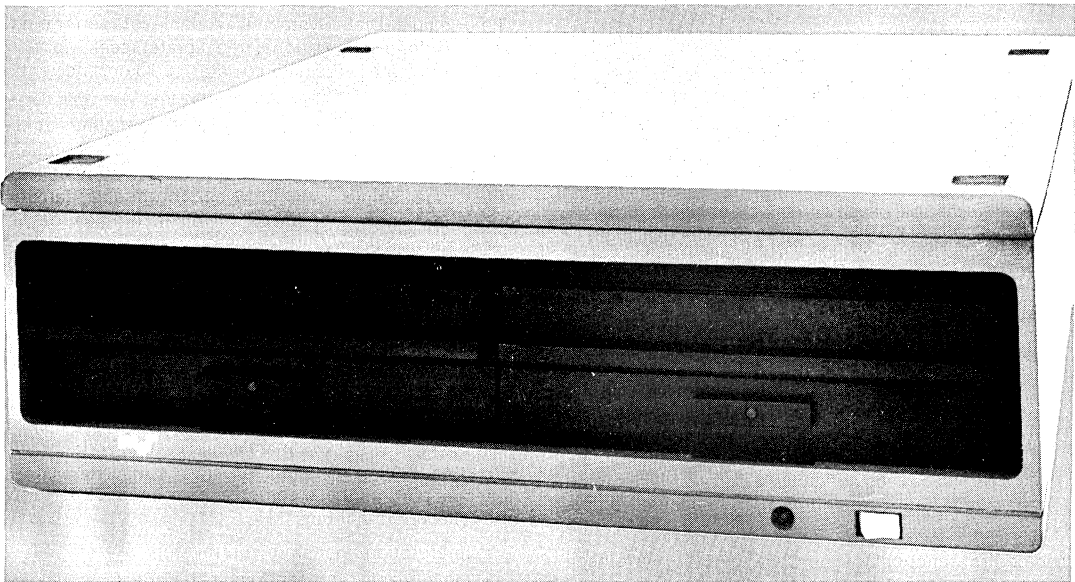
5. The fifth part of the document concludes by summarizing the key findings and recommendations. It stresses the importance of ongoing monitoring and evaluation to ensure that data management practices remain effective and up-to-date.



## INTELLEC® SINGLE/DOUBLE DENSITY FLEXIBLE DISK SYSTEM

- Flexible Disk System Providing High Speed Input/Output and Data Storage for Intellec® Microcomputer Development Systems
- Available in Both Single Density and Double Density Systems
- Data Recorded on Single Density Flexible Disk Is in IBM Soft-Sector Format Which Allows ¼ Million Byte Data Capacity with Up to 200 Files Per Flexible Disk
- 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.

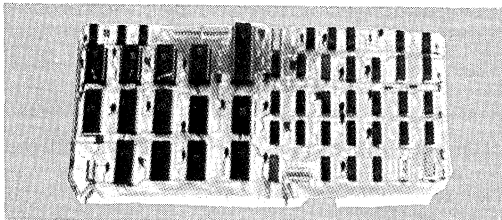


# FLEXIBLE DISK SYSTEM

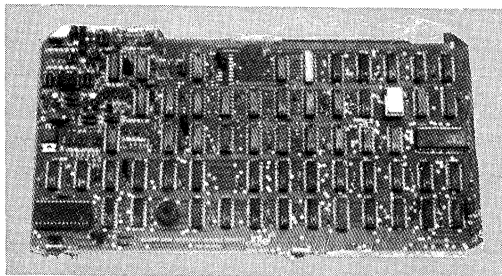
## HARDWARE

The Intellec® flexible disk system provides direct access bulk storage, intelligent controller, and two flexible disk drives. Each single density drive provides ¼ million bytes of storage with a data transfer rate of 250,000 bits/second. 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 single density controller will support two drives. Each double density controller will support up to four drives. The flexible disk system records all data in soft sector format.

The single/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.



SINGLE/DOUBLE DENSITY CHANNEL BOARD



DOUBLE DENSITY INTERFACE BOARD  
(SINGLE DENSITY INTERFACE BOARD  
IS SIMILAR TO THE ONE SHOWN ABOVE)

## 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 Microcom-

puter 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.

This board is the same for either single or double density drives, except that the Series 3000 microcode is different.

## 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 is different for single and double density drives, due to the different recording techniques used. The single density controller boards support one set of dual single density drives. The double density controller boards support 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

## FLEXIBLE DISK SYSTEM

### 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  
 Single or Double density Flexible Disk Drive

### HARDWARE SPECIFICATIONS

#### MEDIA

Single Density	Double Density
Flexible Disk	Double Density Specified Flexible Disk
One Recording Surface	One Recording Surface
IBM Soft Sector Format	Soft Sector Format
77 Tracks/Diskette	77 Tracks/Diskette
26 Sectors/Track	52 Sectors/Track
128 Bytes/Sector	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 Inteltec Development System)

#### CHANNEL BOARD

Single Density	Double Density
5V @ 3.75A (typ), 5A (max)	5V @ 3.75A (typ), 5A (max)

#### INTERFACE BOARD

Single Density	Double Density
5V @ 1.5A (typ), 2.5A (max)	5V @ 1.5A (typ), 2.5A (max) – 10V @ 0.1A (typ), 0.2A (max)

### FLEXIBLE DISK DRIVE PERFORMANCE SPECIFICATION

	Single Density	Double Density
Capacity (Unformatted):		
Per Disk	3.1 megabits	6.2 megabits
Per Track	41 kilobits	82 kilobits
Capacity (Formatted):		
Per Disk	2.05M bits	4.10 megabits
Per Track	26.6K bits	53.2 kilobits
Data Transfer Rate	250 kilobits/sec	500 kilobits/sec
Access Time:		
Track-to-Track	10 ms	10 ms
Head Settling Time	10 ms	10 ms
Average Random Positioning Time	260 ms	260 ms
Rotational Speed	360 rpm	360 rpm
Average Latency	83 ms	83 ms
Recording Mode	Frequency Modulation	M <sup>2</sup> FM

### 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



# FLEXIBLE DISK SYSTEM

---

## EQUIPMENT SUPPLIED

### SINGLE DENSITY

Cabinet, Power Supplies, Line Cord, Two Drives  
Single Density FDC Channel Board  
Single Density FDC Interface Board  
Dual Auxilliary Board Connector  
Flexible Disk Controller Cable  
Flexible Disk Peripheral Cable  
Hardware Reference Manual  
Reference Schematics  
ISIS-II Single Density System Disk  
ISIS-II System User's Guide

### DOUBLE DENSITY

Cabinet, Power Supplies, Line Cord, Two Drives  
Double Density FDC Channel Board  
Double Density FDC Interface Board  
Dual Auxilliary 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-710/110V* 711/220V	Flexible Disk drive unit with two drives, single density drive controller, software, and cables.
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.

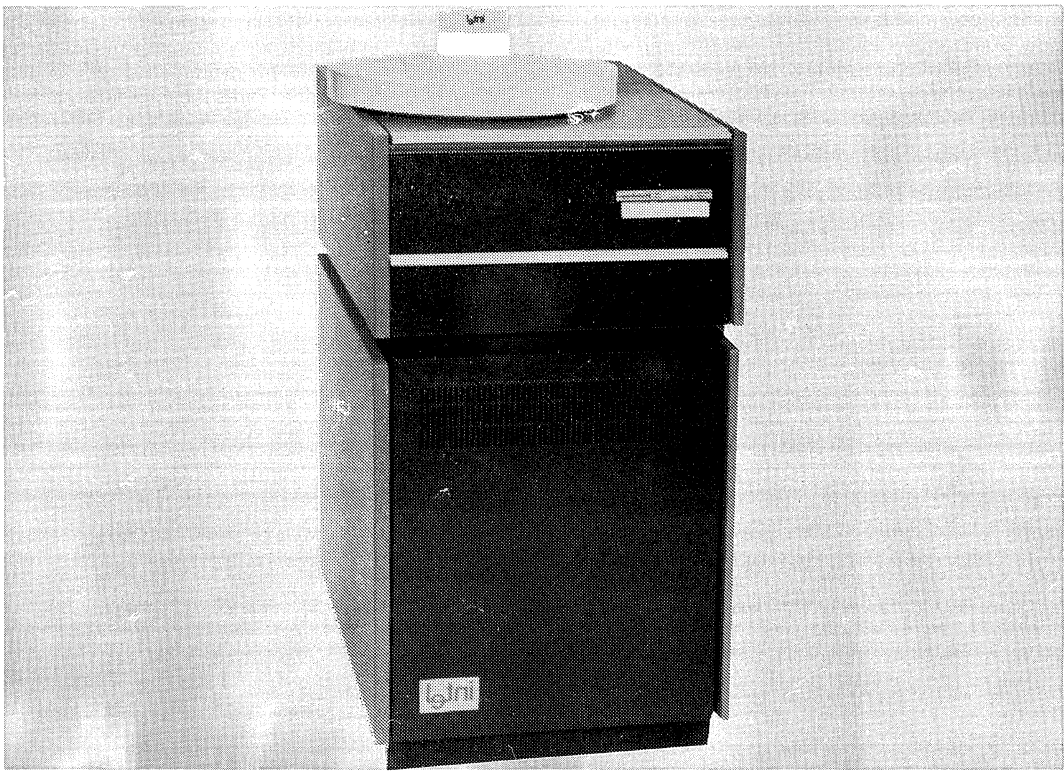
\*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.



## MODEL 740 INTELLEC® HARD DISK SUBSYSTEM

- 7.3 million bytes of on-line hard disk storage
- 5440-type disk cartridge
- Enhanced development system performance
- Fixed and removable mass storage in one drive
- Separate write protect switch for each disk platter
- Intelligent 2-board controller

The Intellec Model 740 Hard Disk Subsystem is a flexible, high capacity, mass storage peripheral for use with Intellec Microcomputer Development Systems. In addition to providing a large fixed and removable storage component, the 740 can significantly improve throughput and reduce development time.



## FUNCTIONAL DESCRIPTION

### Hardware Components

The Intellec Hard Disk Subsystem consists of an intelligent controller and disk drive with one removable cartridge and one fixed platter. Each provides 3.6 million bytes of storage (7.3 M total) with a data transfer rate of 2.5 MBits/second. The disk controller is implemented with Intel's powerful Series 3000 Bipolar Microcomputer Set. The controller provides an interface to the Intellec system bus, as well as the disk drive. Data is recorded in FM format on each surface (2 per platter). Each platter may be separately write protected via a front panel switch. The disk subsystem will perform six specific operations: recalibrate, seek, format track, write data, read data, and verify CRC.

### Disk Controller Boards

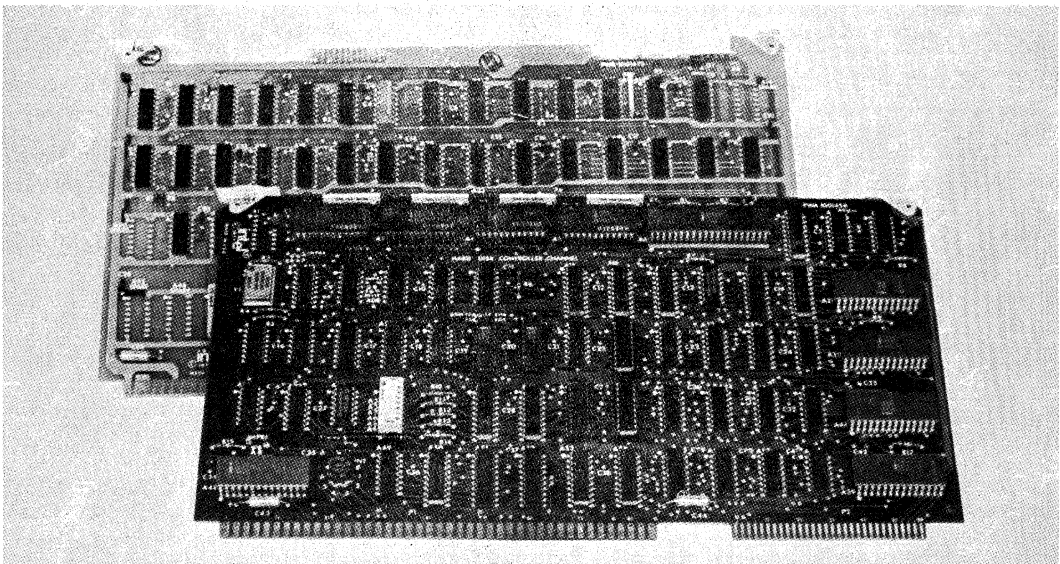
The disk controller consists of two boards, the channel board and the interface board. These two PC boards reside in the Intellec Series II system chassis and constitute the disk controller. The channel board receives, decodes and responds to channel commands from the processor. The interface board provides the disk controller with a means of communication with the disk drive. The interface board validates data during reads using a

cyclic redundancy check (CRC) polynomial and generates CRC data during write operations. When the disk controller requires access to Intellec system memory, the channel board requests and maintains DMA master control of the system bus, and generates the appropriate memory command. The channel board also acknowledges I/O commands as required by the Intellec bus.

### Channel Board

The Channel Board is the primary control module within the hard disk subsystem. 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 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 1K x 32 bits of 3628 programmable-read-only-memory (PROM) which stores the microprogram. It



Controller Boards



## MODEL 740

is the execution of the microprogram by the micro-computer set which actually effects the control capability of the Channel Board.

When the disk controller requires access to Intellec system memory, the Channel Board requests the DMA master control of the system bus, and generates the appropriate memory command. The Channel Board also acknowledges I/O commands as required by the Intellec bus.

### Interface Board

The Interface Board provides the 740 disk controller with a means of communication with the 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 move to the proper track and

verify successful operation. The Interface Board accepts the data being read off the 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.

### Associated Software

**Intel Systems Implementation Supervisor (ISIS-II)** — The Hard Disk Subsystem 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 support, and utility management.

---

## SPECIFICATIONS

### Hardware

#### Disk Drive

**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

#### Disk System Capacity

**Per Surface** — 15M bits

**Per Platter** — 29M bits

**Per Drive** — 59M bits

**Per Drive** — 7.3M bytes (formatted)

#### Disk Performance

**Disk Transfer Rate** — 2.5M bits/sec

**Disk System Access Time** —

Track to Track: 13 ms max.

Full Stroke: 100 ms

Rotational Speed: 2,400 rpm

### Physical Characteristics

Disk Drive on Pedestal

**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

#### Chassis

**DC Power Supplies** — Internal to Cabinet

#### AC Power Requirements

110 VAC; 60 Hz; 5A (max)

220 VAC; 50 Hz; 3A (max)

#### Controller Boards

5V @ 5.5A (typ), 6.5A (max)

### Environmental Characteristics

#### Media, Drive and Chassis

Temperature:

Operating: 16°C to 32°C

Non-operating: -10°C to 60°C

Humidity:

20% to 80% non-condensing

#### Controller Boards

Temperature:

Operating: 0°C to 55°C

Non-operating: -55°C to 85°C

Humidity:

Up to 90% non-condensing



# MODEL 740

## Equipment Supplied

- Hard disk drive pedestal mounted
- Hard disk controller (2 boards)
- Cables
- Disk Cartridge
- ISIS-II System Diskette
- 9800306** — ISIS-II System User's Guide
- 9800943** — Hard Disk Subsystem Operation and Checkout Manual

## Optional Equipment

- MDS-746** Box of 5 blank cartridges specified for use on Model 240/241, 740/741

---

## ORDERING INFORMATION

Part Number	Description
MDS-740/110V 741/220V	Hard disk unit with cables, controller, and disk cartridge

---

*MCS-80/85™  
Development Systems  
and Options*

---

**14**





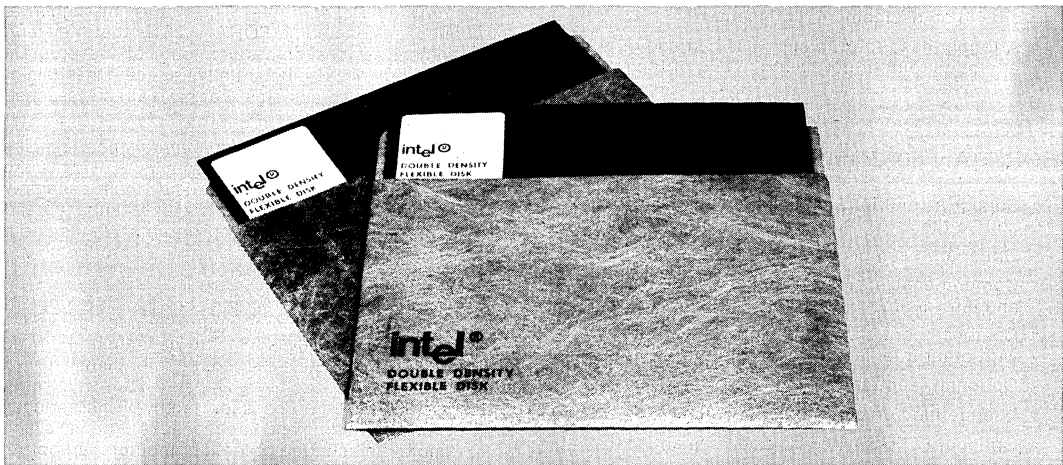
## FORTRAN 80 8080/8085 ANS FORTRAN 77 INTELLEC® RESIDENT COMPILER

- 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<sup>1</sup>. 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.

<sup>1</sup>ANSI X3J3/90





## 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 FORTRAN 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 Intel 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.

# FORTRAN 80

## 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

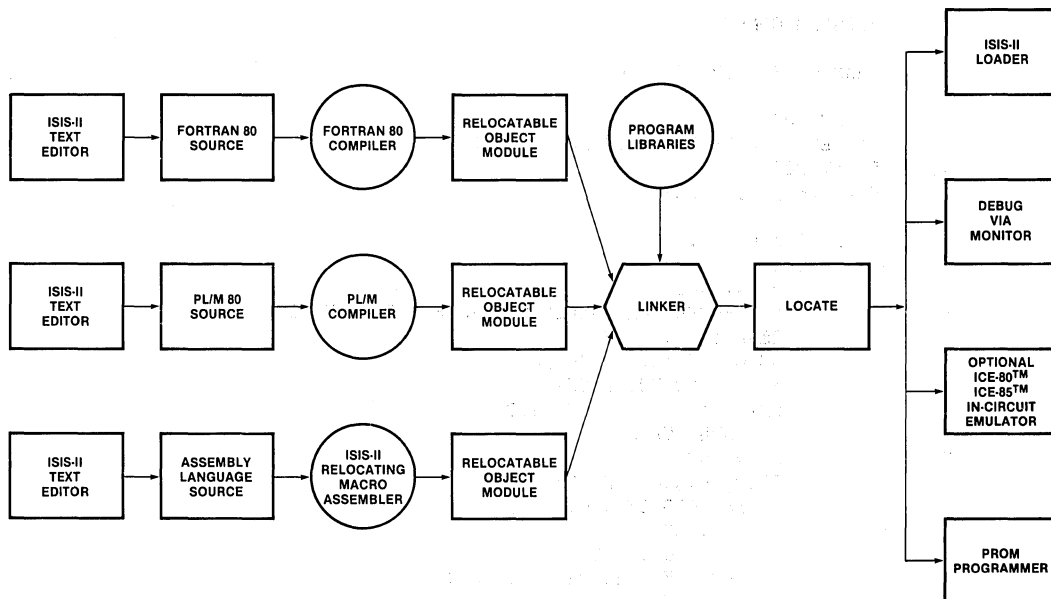
CHARACTER*1 CHOICE, SCALE

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 Inteltec 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.

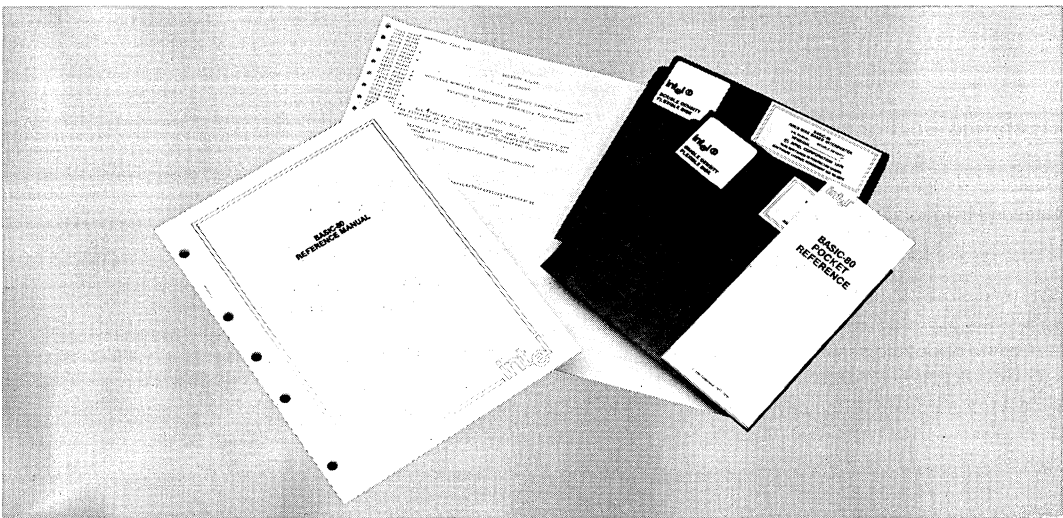


## BASIC-80 EXTENDED ANS 1978 BASIC INTELLEC® RESIDENT INTERPRETER

- 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

## 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

## BASIC-80

---

### 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.

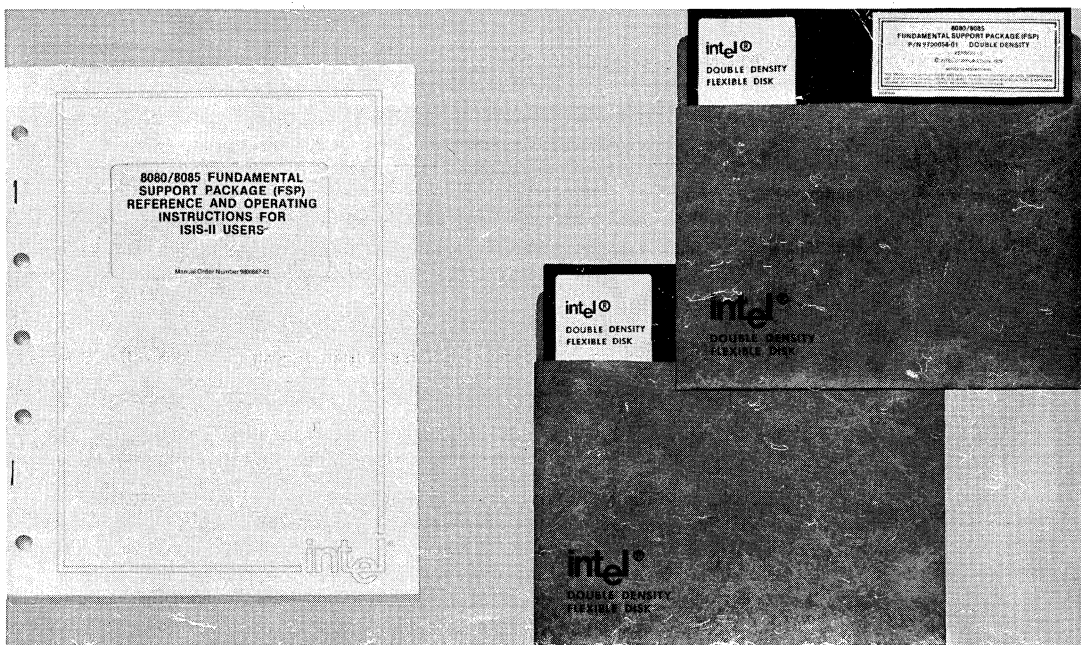


## 8080/8085 FUNDAMENTAL SUPPORT PACKAGE (FSP)

- 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.



## 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.

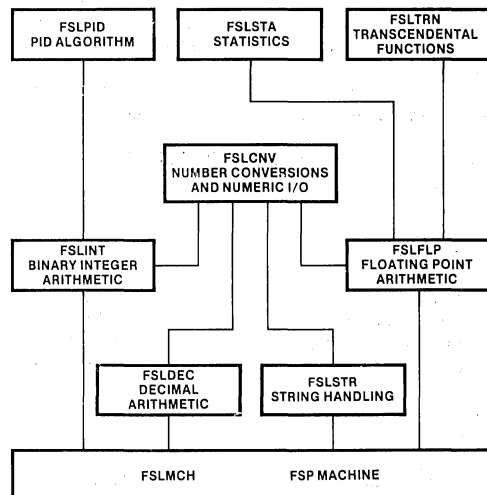


Figure 1. Hierarchy of FSP Modules





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 $2^8-1$
2 bytes, without sign	0 to $2^{16}-1$
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

$$M(t) = B + \frac{100}{P} \left( E(t) + \frac{1}{R} \int_{t_0}^t E(s) ds + D \frac{dE}{dt} \right)$$



## 8080/8085 FUNDAMENTAL SUPPORT PACKAGE (FSP)

---

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.



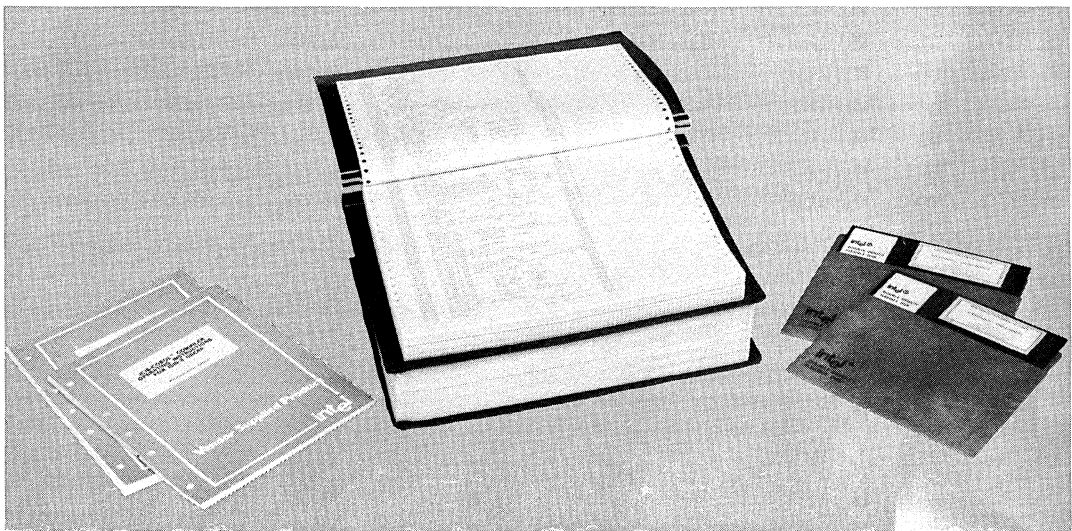
## iCIS-COBOL SOFTWARE PACKAGE

- Meets and Exceeds Minimum ANSI Level 1 Standard for COBOL (X3.23-1974)
- Runs Under ISIS-II on Inteltec® or Inteltec® 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 Inteltec and Inteltec 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 Inteltec 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 Inteltec 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

Screen as a record description

FILLER

REDEFINES

AT line:column

Character highlighting

Clear screen

Numeric vet for PIC fields

### CURSOR CONTROL FACILITIES

HOME to the start of the first data field on the screen

← Forward space

→ Backward space

↓ Forward field

↑ Backward field

C/R Release the screen of data

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.

# iCIS-COBOL

- **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 Inteltec 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 Inteltec development system.
  - COBOL applications programs developed using iCIS-COBOL software package will increase utilization of Inteltec development systems.

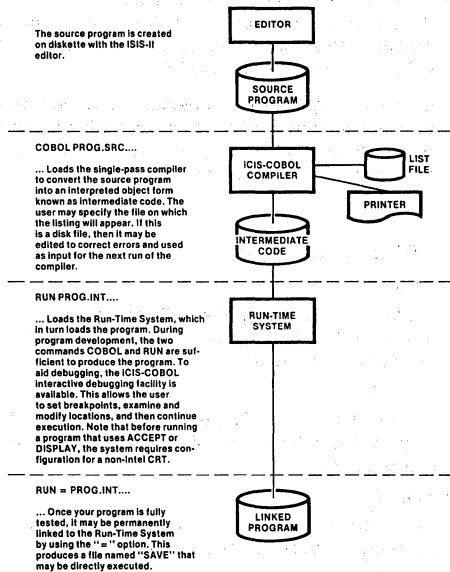


Figure 1. Program Development Cycle

```

**iCIS-COBOL V 1.0                                IF I1STOCK1.CBL                                PAGE1 0001
**
000010 IDENTIFICATION DIVISION.                                000F
000020 PROGRAM-ID, STOCK-FILL=INT-VP.                            000F
000030 AUTHOR, PAIR PIER.                                        000F
000040 ENVIRONMENT DIVISION.                                    000F
000050 CONFIGURATION SECTION.                                    000F
000060 SOURCE-COMPUTER.                                        000F
000070 OBJECT-COMPUTER.                                        000F
000080 INPUT-OUTPUT SECTION.                                    000F
000090 FILE-CONTROL.                                          000F
000100 SELECT STOCK-FILL ASSIGN STOCK-IT.                    000F
000110 ORGANIZATION INCLD=4.                                    000F
000120 ACCESS SEQUENTIAL.                                      000F
000130 RECORD KEY STOCK-CODE.                                  000F
000140 DATA DIVISION.                                        000F
000150 FILE SECTION.                                          000F
000160 FILE STOCK-FILL RECORD 32.                              000F
000170 ST STOCK-ITER.                                          000F
000180   DS STOCK-CODE PIC X(4).                               000F
000190   DS PRODUCT-DESC PIC X(24).                            000F
000200   DS UNIT-PRICE PIC 9(9).                               000F
000210   DS UNIT=SI0KAGE SECTION.                              000F
000220   DS AS=CODE PIC X(4) VALUE STOCK CODE < >.           000F
000230   DS FILLER PIC X(59).                                  000F
000240   DS AS=AS=CODE PIC X(16) VALUE DESCRIPTION < >.       000F
000250   DS SI=DESC PIC X(25) VALUE < >.                       000F
000260   DS FILLER PIC X(19).                                  000F
000270   DS AS=SIZE PIC X(12) VALUE UNIT SIZE < >.           000F
000280   DS UNIT=IT UNPACKED DECIMAL=RAWING.                  000F
000290   DS FILLER PIC X(16).                                  000F
000300   DS CRT=CODE=CODE PIC X(4).                            000F
000310   DS FILLER PIC X(16).                                  000F
000320   DS CRT=PRU=DESC PIC X(4).                             000F
000330   DS FILLER PIC X(20).                                  000F
000340   DS CRT=UNIT=SIZE PIC 9(4).                            000F
000350   DS FILLER PIC X.                                      000F
000360   DS FILLER DIVISION.                                    000F
000370 PAGE=CURR DIVISION.                                    000F
000380 END.                                                    000F
000390 DISPLAY SPAC.                                          000F
000400 OPEN I=O STOCK-FILL.                                    000F
000410 DISPLAY UNPACKED=RAWING.                                000F
000420 UNTIL=INPUT.                                           000F
000430 DO WHILE SPACE TO BRN=IT.                               000F
000440 DISPLAY LIST=IT.                                       000F
000450 CORRECT=MENU.                                          000F
000460 ACCEPT UNIT=IT.                                        000F
000470 IF CRT=SI0KA=CODE = SPALC UU TO BRN=IT.                000F
000480   CRT=SI0K=DESC AND UNIT=UU TO CORRECT=MENU.          000F
000490   MOVE CRT=PRU=DESC TO PRODUCT=DESC.                 000F
000500   MOVE UNIT=SI0K TO UNIT=SIZE.                         000F
000510   MOVE CRT=SI0KA=CODE TO SI0KA=CODE.                  000F
000520   UNIT STOCK-ITER INVALIG UU TO CORRECT=MENU.        000F
000530   GO TO BRN=SI=INPUT.                                  000F
000540 END=IT.                                               000F
000550 CLOSE STOCK-FILL.                                      000F
000560 DISPLAY SPAC.                                          000F
000570 DISPLAY END OF PROGRAM.                                000F
000580 STOP RUN.                                             000F
** END OF LIST
    
```

Annotations for Figure 2:

- COLS 1-6 SEQUENCE NUMBER**: Points to the first column of the listing.
- COLS 7-11 INDICATOR AREA**: Points to columns 7 through 11.
- COLS 12-72 AREA A**: Points to columns 12 through 72.
- COLS 12-72 AREA B**: Points to columns 12 through 72.
- INSERTED BY COMPILER**: Points to the right margin of the listing.

Figure 2. Sample Program Listing Showing Source Format

# ICIS-COBOL

---

## 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.



**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 Intel 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 programmer 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.

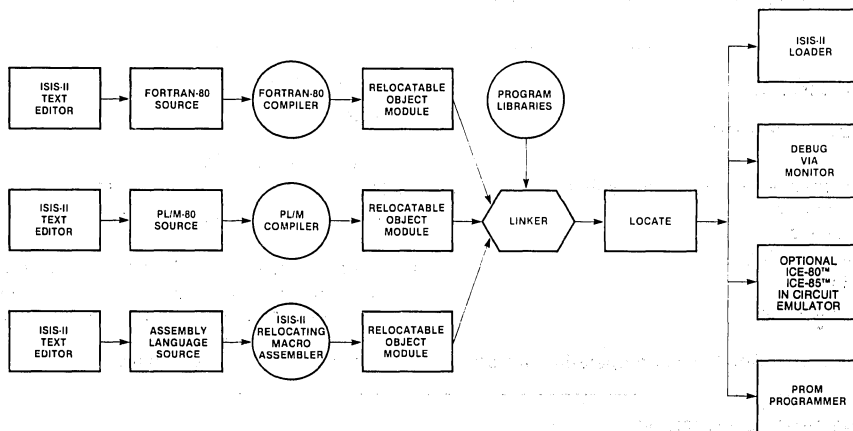


Figure 1. Program Development Cycle Block Diagram



# PL/M 80

**Simpler Project Development** — because the Intellec microcomputer development system with resident PL/M 80 is all that is needed for developing and debug-

ing 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.



## PASCAL 80 SOFTWARE PACKAGE

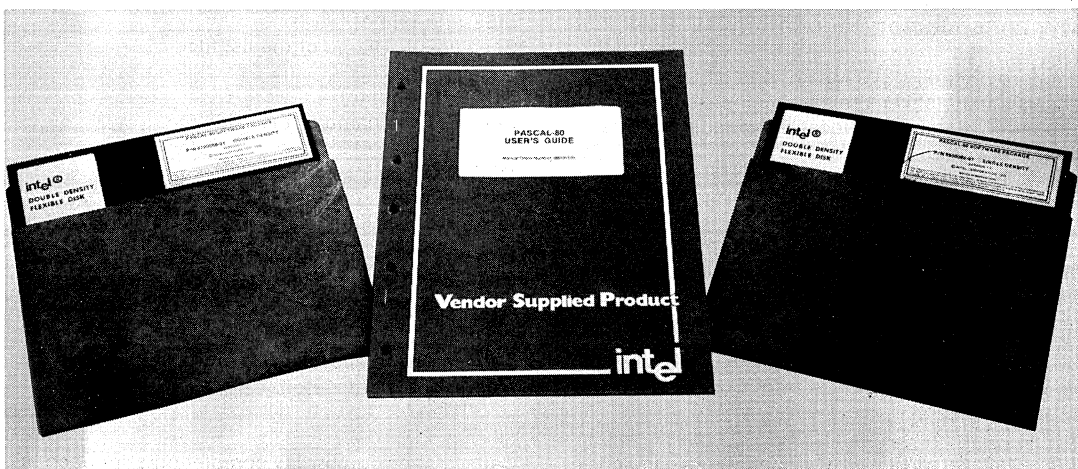
- Offers a Superset of Standard Pascal
- Provides Highly Structured Language with Powerful Data Type Definitions to Suit Applications
- Compiles Pascal Source Code into Intermediate Code to Optimize Execution Speed and Storage
- Executes Compiler and Interprets the Intermediate Code on Intellec® Microcomputer 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
- Includes Data Types and Procedures Consistent with RMX-80™ Environment for Optional RMX-80 Run-Time Interpreter (iSBC 803™)
- 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 or optionally under RMX-80.

\*Pascal language as defined in *PASCAL User Manual and Report*, Second Edition, Kathleen Jensen and Niklaus Wirth.



## 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.

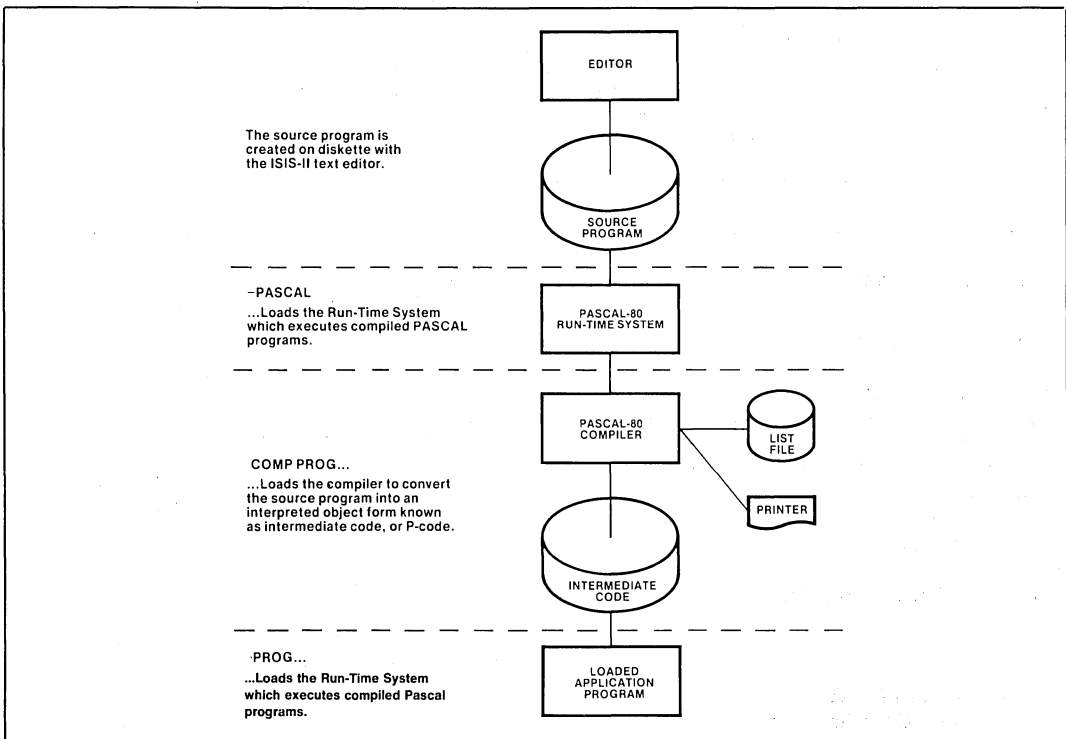


Figure 1. Program Development Cycle

**Table 1. Sample Program Listing Showing Nesting Levels**

					<b>BUFFER.PAS Program Listing</b>
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.

---

Intel Corporation has carefully reviewed this product and believes that the product will operate and perform according to its published user manuals. HOWEVER, INTEL MAKES NO WARRANTIES WITH RESPECT TO THE OPERATION AND USE OF THIS VENDOR SUPPLIED PRODUCT. Successful use depends solely on customer's ability to install and use this product.

This Vendor Supplied Product is licensed on an "as is" basis and Intel Corporation does not guarantee any future enhancements or extensions to this product. The existence of this product does not imply its adaptation in any form as an Intel standard nor its compatibility with any other Intel product except as specifically stated in the published user manuals.

Intel will provide limited telephone assistance to the customer in the understanding of the operation of the product.

In addition, if a problem is encountered which the user diagnosis indicates is caused by a defect in this Vendor Supplied Product, the user is requested to fill out a Problem Report form and mail it to:

Intel Corporation  
MCSD, Marketing  
3065 Bowers Avenue  
Santa Clara, CA 95051

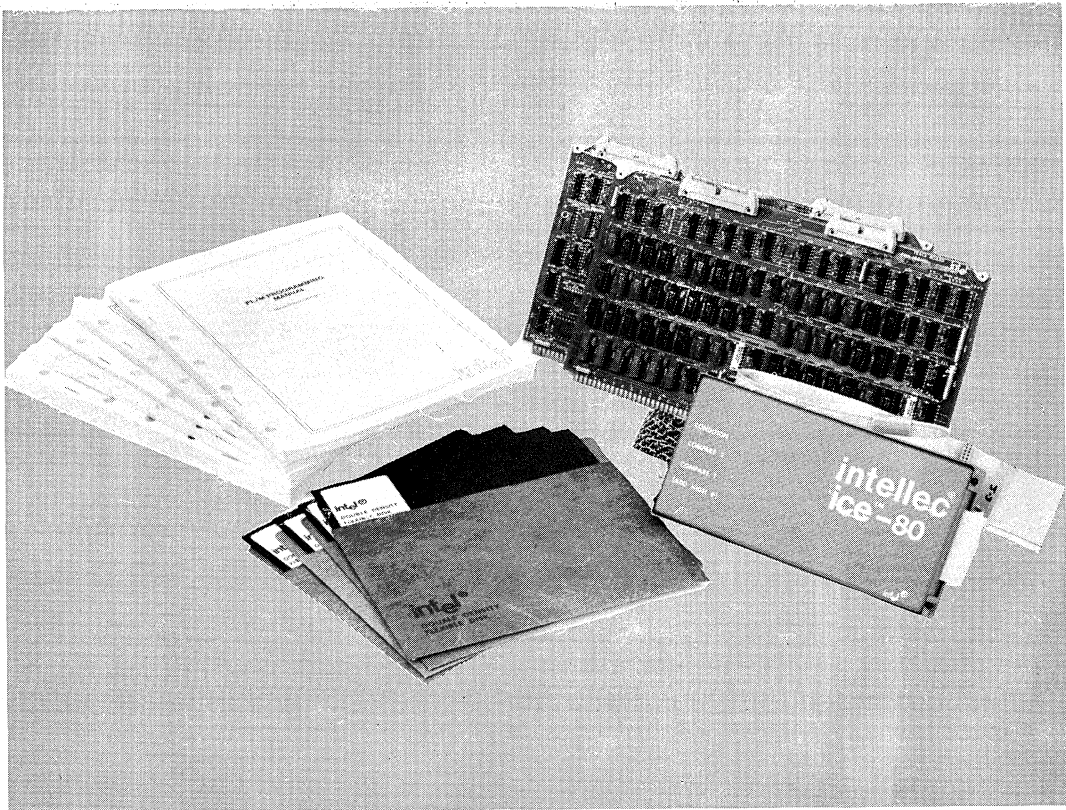
Intel will use its best efforts to respond to Problem Reports in one of the following ways: 1) release information to correct the problem, 2) offer a new revision, when available with corrected code to fix the problem, or 3) issue a notice of availability of a new revision with corrected code.



## SP80 SUPPORT PACKAGE

- Development Support for 8080 Microprocessor Designs, Including:
- PL/M 80 Structured High-Level Programming Language Compiler for System Software Development
- ANS 77 FORTRAN 80 Compiler for Mathematically Oriented Software Development
- ICE-80™ In-Circuit Emulator
- Programs Written in PL/M, FORTRAN, and Assembly Language can be Linked Together and Relocated
- ICE-80™ Emulation Connects Intellec® System Resources to a User's Prototype
- Full Symbolic Debugging is Available for Program Labels or Variables

8080 software development begins using the 8080 macroassembler, PL/M 80, FORTRAN 80 high-level languages and compilers. The compilers operate on Intellec microcomputer development systems under ISIS-II disk operating systems and produces efficient relocatable object modules compatible for linkage with PL/M 80, FORTRAN 80 and 8085 macroassembler modules. After compilation, the object program may be linked to other modules, then located to an in-circuit emulator module (ICE-80) for software execution and debugging in the Intellec environment.



# SP80 SUPPORT PACKAGE

---

## SP80 SUPPORT PACKAGE SPECIFICATIONS

### Operating Environment

#### Required Hardware

DS002 System Package or equivalent is required

DS003 System Package is recommended

### Emulation Clock

User's system clock or ICE-80 adaptor socket.

### Electrical Characteristics

#### DC Power Requirements

$V_{CC} = +5V \pm 5\%$

$I_{CC} = 10A \text{ max}; 7A \text{ typ}$

$V_{DD} = +12V \pm 5\%$

$I_{DD} = 80 \text{ mA max}; 45 \text{ mA typ}$

$V_{BB} = -10V \pm 5\%$

$I_{BB} = 1 \text{ mA max}; 1 \mu A \text{ typ}$

### Environmental Characteristics

Operating Temperature — 0° to 40°C

Operating Humidity — Up to 95% relative humidity without condensation.

### Reference Manuals (supplied)

FORTTRAN-80 Programming Manual (9800481)

ISIS-II FORTTRAN-80 Compiler Operator's Manual  
(9800480)

FORTTRAN-80 Programming Reference Card (9800547)

PL/M-80 Programming Manual (980026)

ISIS-II PL/M-80 Compiler Operator's Manual (9800300)

ICE-80 Operator's Manual (9800185)

---

## ORDERING INFORMATION

Part Number	Description
SP80-KIT	SP80 Support Package Includes ICE-80 in-circuit emulator (MDS-80-ICE), PL/M-80 High-Level Programming Language (MDS-PLM), and FORTTRAN-80 Compiler (MDS-301)

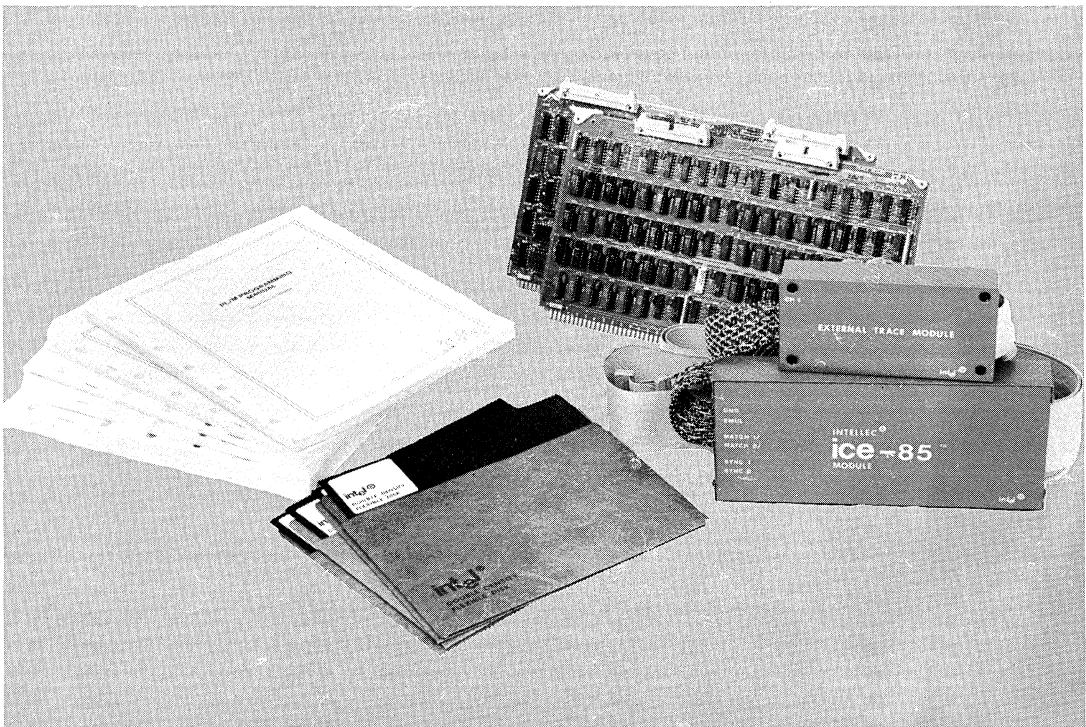




## SP85 SUPPORT PACKAGE

- Development Support for 8085 Microprocessor Designs, Including:
  - PL/M 80 Structured High-Level Programming Language Compiler for System Software Development
  - ANS 77 FORTRAN 80 Compiler for Mathematically Oriented Software Development
  - ICE-85B™ In-Circuit Emulator
- Programs Written in PL/M, FORTRAN, and Assembly Language can be Linked Together and Relocated
- ICE-85B™ Emulation Connects Inteltec® System Resources to a User's Prototype
- Full Symbolic Debugging Is Available for Program Labels or Variables
- External Trace Module Extends Emulation Capability to Prototype System Peripheral Activity

8085 software development begins using the 8085 macroassembler, PL/M 80, FORTRAN 80 high-level languages and compilers. The compilers operate on Inteltec microcomputer development systems under ISIS-II disk operating systems and produces efficient relocatable object modules compatible for linkage with PL/M 80, FORTRAN 80 and 8085 macroassembler modules. After compilation, the object program may be linked to other modules, then located to an in-circuit emulator module (ICE-85B) for software execution and debugging in the Inteltec environment.





## SP85 SUPPORT PACKAGE SPECIFICATIONS

### Operating Environment

#### Required Hardware

DS002 System Package or equivalent is required.

DS003 System Package is recommended.

### Emulation Clock

User's system clock or ICE-85B adaptor socket (6.144 MHz crystal)

### Electrical Characteristics

#### DC Power Requirements

$V_{CC} = +5V \pm 5\%$

$I_{CC} = 12A \text{ max}; 10A \text{ typ}$

$V_{DD} = +12V \pm 5\%$

$I_{DD} = 80 \text{ mA max}; 60 \text{ mA typ}$

$V_{BB} = -10V \pm 5\%$

$I_{BB} = 30 \text{ mA max}; 10 \mu A \text{ typ}$

### Environmental Characteristics

**Operating Temperature** — 0° to 40°C

**Operating Humidity** — Up to 95% relative humidity without condensation.

### Reference Manuals (supplied)

FORTRAN-80 Programming Manual (9800481)

ISIS-II FORTRAN-80 Compiler Operator's Manual  
(9800480)

FORTRAN-80 Programming Reference Card (9800547)

PL/M-80 Programming Manual (980026)

ISIS-II PL/M-80 Compiler Operator's Manual (9800300)

ICE-85 Operator's Manual (9800463)

---

## ORDERING INFORMATION

Part Number	Description
SP85-KIT	SP85 Support Package Includes: ICE-85B™ In-circuit emulator (MDS-85B-ICE) PL/M 80 High-Level Programming Language (MDS-PLM)* and FORTRAN 80 Compiler (MDS-301)

Requires software license

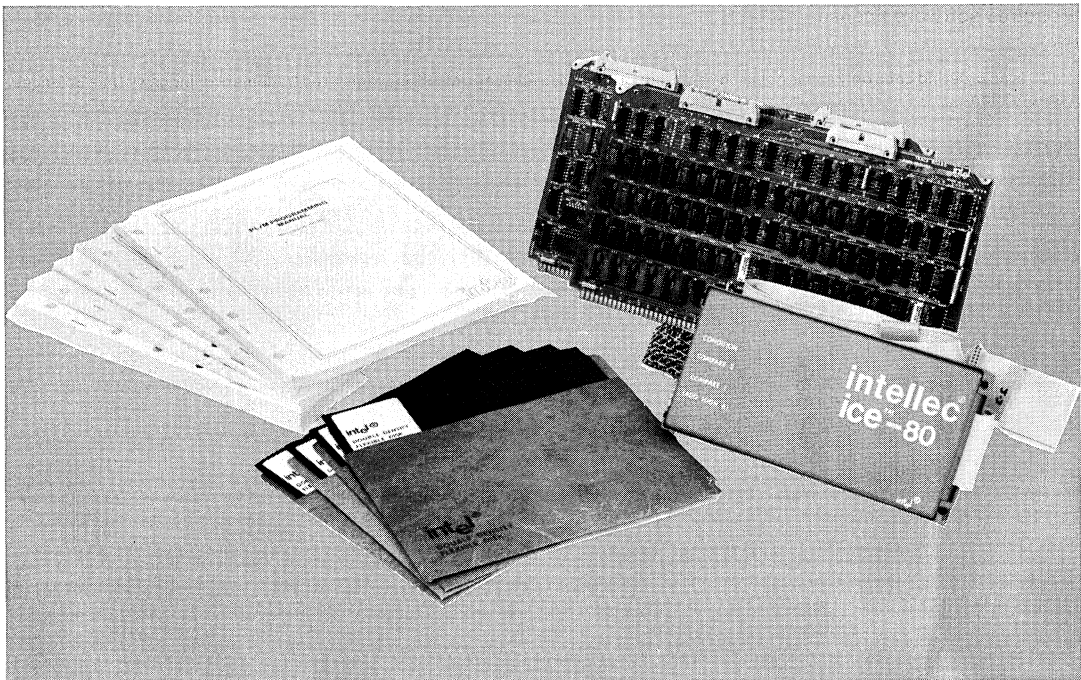
\*MDS is an ordering code only and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corporation.



## ICE-80™ 8080 IN-CIRCUIT EMULATOR

- Connects Inteltec® 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 Inteltec® RAM, ROM, and PROM Memory and Inteltec® 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 Inteltec ICE-80 8080 In-Circuit Emulator is an Inteltec 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 Inteltec memory and I/O for user system equivalents. Powerful Inteltec 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.



# ICE-80™ IN-CIRCUIT EMULATOR

## 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  $\phi_2$  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.

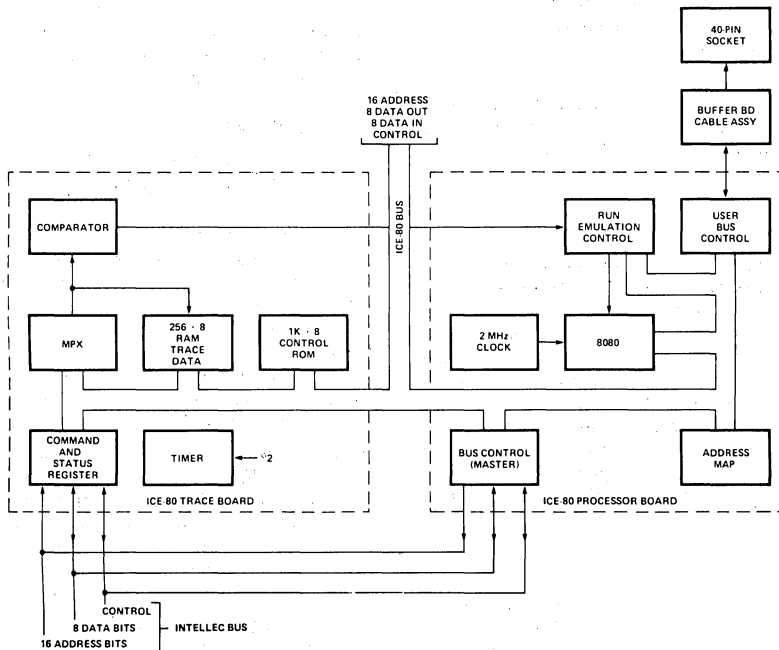


Figure 1. Functional Block Diagram of ICE-80 Module

## ICE-80™ IN-CIRCUIT EMULATOR

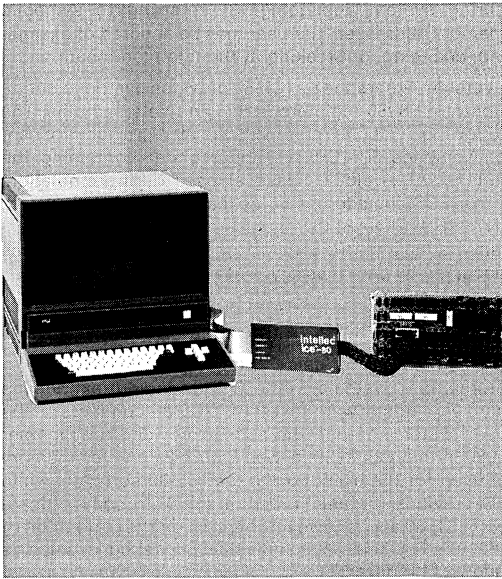


Figure 2. ICE-80 Module Installed in User System

**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 nonexistent 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 comparator 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  $\phi_2$  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

# ICE-80™ IN-CIRCUIT EMULATOR

```

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
③ *LOAD PROG. HEX
  *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 = FFFE H DATA = 29H
⑥ *CHANGE DOUBLE REGISTER SP = 13FFFH
  *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 = 13FFFH
  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-1FFFFH) 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-13FFFH 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 13FFFH, 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

# ICE-80™ IN-CIRCUIT EMULATOR

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
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
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
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

## ICE-80™ IN-CIRCUIT EMULATOR

---

### System Clock

Crystal controlled 2.185 MHz  $\pm$  0.01%. May be replaced by user clock through jumper selection.

$I_{DD}$  = 79 mA max; 45 mA typ  
 $V_{BB}$  = -9V,  $\pm$  5%  
 $I_{BB}$  = 1 mA max; 1  $\mu$ A typ

### 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,  $\pm$  5%

$I_{CC}$  = 9.81A max; 6.90A typ

$V_{DD}$  = +12V,  $\pm$  5%

### 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.



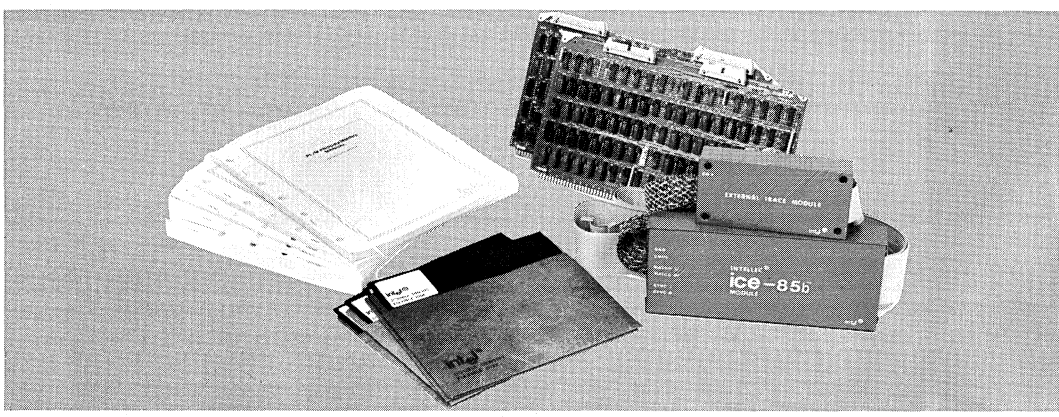


## ICE-85B™ MCS-85™ IN-CIRCUIT EMULATOR WITH MULTI-ICE™ SOFTWARE

- 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.



# ICE-85B™ IN-CIRCUIT EMULATOR

## 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 Inteltec® 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 Inteltec® 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 Inteltec® System equivalents to take the place of devices not yet designed for the user system during prototyping. In addition, Inteltec® 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 Inteltec 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 Inteltec.

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, Inteltec® 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

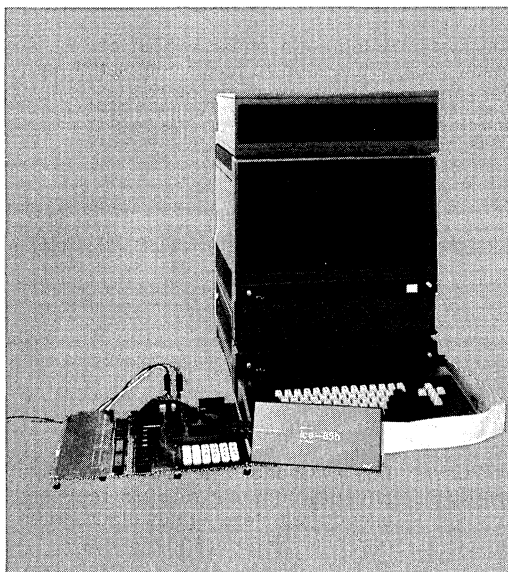
DISPLAY/ CHANGE	Display/Changes the values of symbols and the contents of 8085 registers, pseudo-registers, status flags, interrupt bits, I/O ports and memory.
EVALUATE	Displays the value of an expression in the binary, octal, decimal or hexadecimal.
SEARCH	Searches user memory between locations in a 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 Inteltec® System memory.
EXECUTE	Saves emulated program registers and emulates a user-supplied subroutine to access peripheral chips in the user's system.

# ICE-85B™ IN-CIRCUIT EMULATOR

## 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 Intellect® 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."

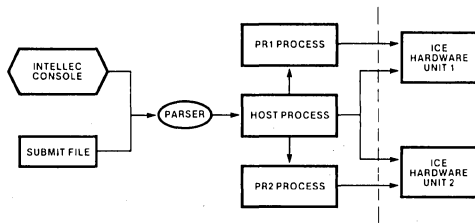
The trace buffer, also 42 entries wide, collects data sampled from 24 8085 processor channels and 18 external channels sampled by the External Trace Module. The signals collected from the 8085 include address lines, data lines, status lines and serial input and output lines. The 18 channels extending 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.

# ICE-85B™ IN-CIRCUIT EMULATOR

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      ;This macro clears the
INITMEM           ;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

# ICE-85B™ IN-CIRCUIT EMULATOR

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.

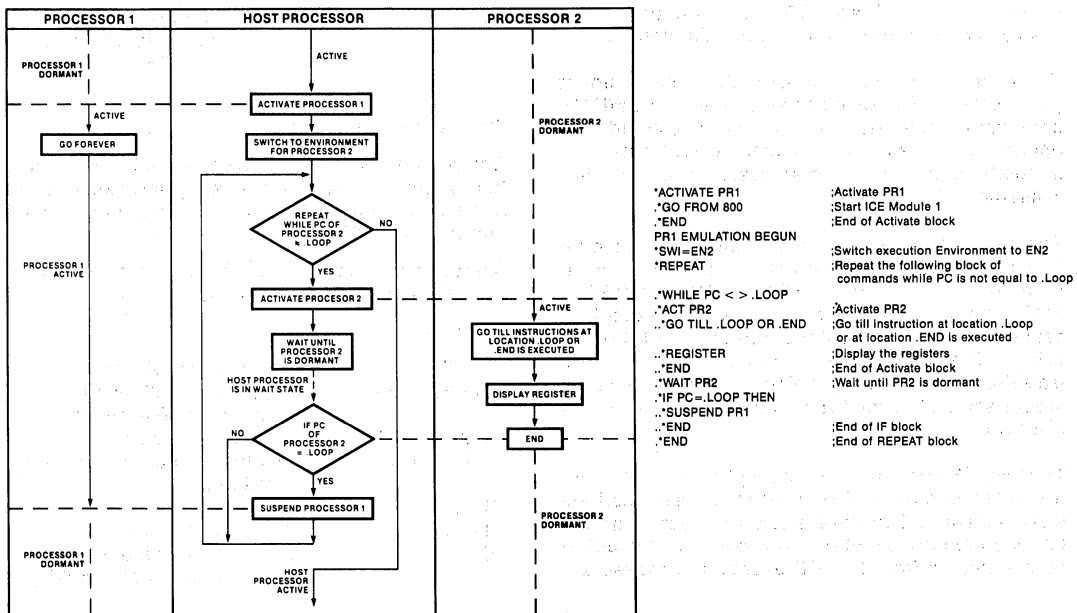
*MAP IO 0 TO FF
=USER
*SWITCH = EN2
*LOAD :F2:LED.HEX
*SWITCH = EN1
* ;End of the file PROG1
* ;After the end of file is
  reached, control is
  returned to console.
```

## 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.



Flowchart of the Example for Demonstrating Multi-ICE™ Synchronization Capability

# ICE-85B™ IN-CIRCUIT EMULATOR

## 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. (17.15 cm)  
 Depth: 0.50 in. (1.27 cm)  
 Packaged Weight: 6.00 lb (2.73 kg)

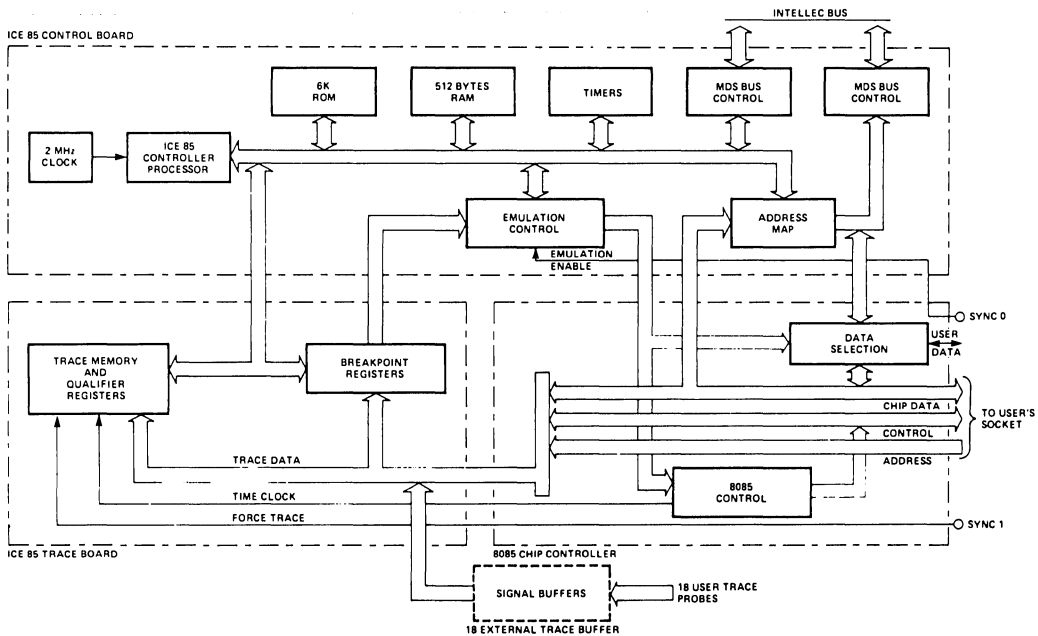
### Electrical Characteristics

#### DC Power:

$V_{CC} = +5V \pm 5\%$   
 $I_{CC} = 12A$  maximum; 10A typical  
 $V_{DD} = +12V \pm 5\%$   
 $I_{DD} = 80$  mA maximum; 60 mA typical  
 $V_{BB} = -10V \pm 5\%$   
 $I_{BB} = 1$  mA maximum; 10  $\mu A$  typical

### Environmental Characteristics

Operating Temperature: 0° to 40°C  
 Operating Humidity: Up to 95% relative  
 humidity without  
 condensation.



ICE-85B™ BLOCK DIAGRAM

# ICE-85B™ IN-CIRCUIT EMULATOR

---

## 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

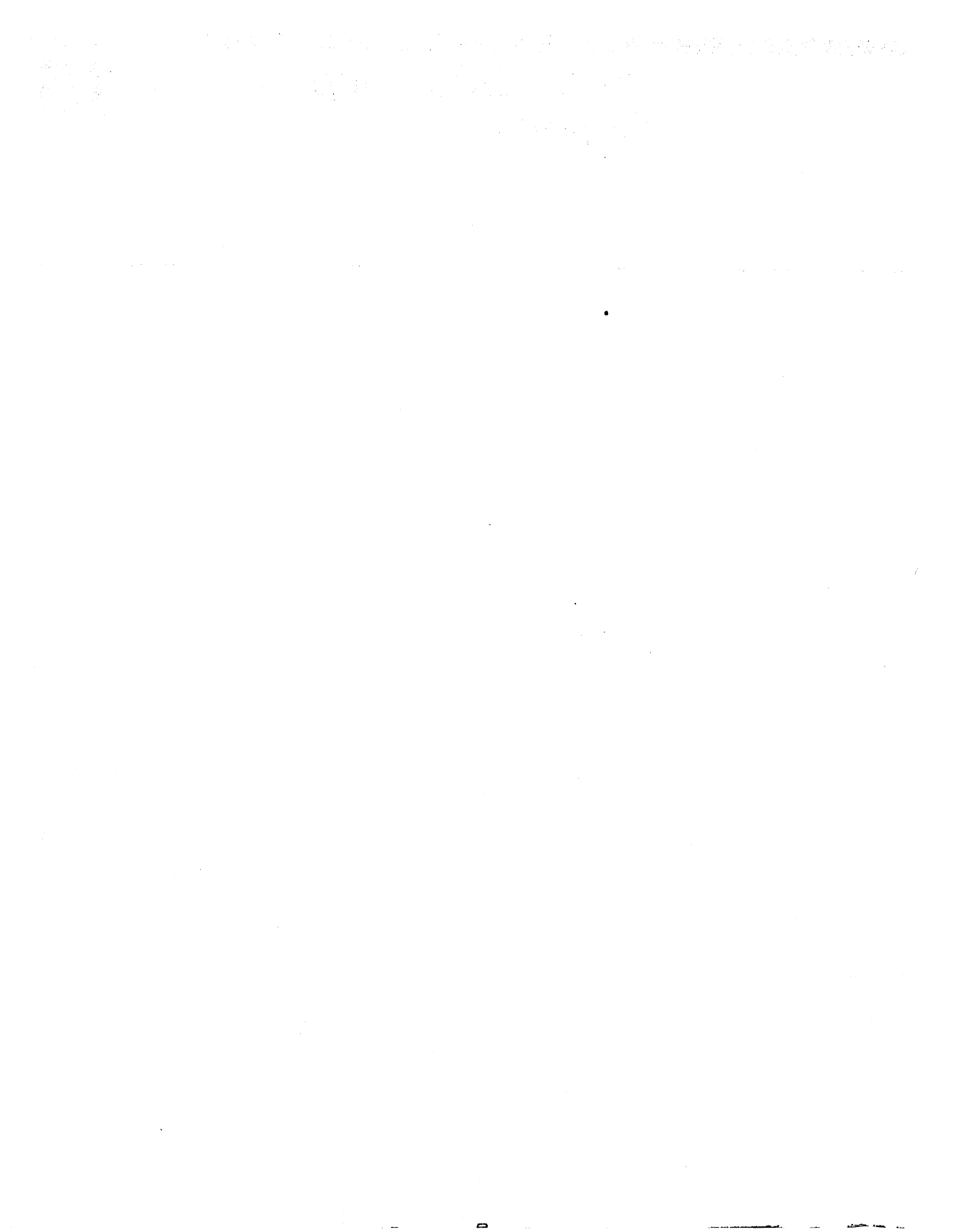
\*\*\*MDS\*\* is an ordering code only, and is not used as a product name or trademark. MDS® is a registered trademark of Mohawk Data Sciences Corp.

*iAPX 86/88 Support  
Options*

---

**15**







## SERIES II 8086/8088 SOFTWARE DEVELOPMENT PACKAGES

- PL/M 86/88 High Level Programming Language
- ASM 86/88 Macro Assembler for 8086/8088 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 8086/8088 Assembly Language Source Code
- OH 86/88 Object-to-Hexadecimal Converter
- LIB 86/88 Library Manager

The Series II 8086/8088 software development packages provide a set of software development tools for the 8086 and the 8088 microprocessors and iSBC 86/12A single board computer. The package operates under the ISIS-II operating system on Intel 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 8086/8088 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 Intel Series II, such as a Model 235, provide total 8086/8088 software development capability.



## SERIES II PL/M 86/88 COMPILER

- **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 8086 and the 8088 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 8086 and 8088 Microprocessors.

PL/M 86/88 has significant new capabilities over PL/M 80 that take advantage of the new facilities provided by the 8086 and the 8088 microprocessors, 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 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.

### 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 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 save 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.

---

## **SERIES II 8086/8088 MACRO ASSEMBLER**

- **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 8086/8088 assembly language. ASM 86/88 translates symbolic 8086/8088 assembly language mnemonics into 8086/8088 relocatable object code.

ASM 86/88 should be used where maximum code efficiency and hardware control is needed. The 8086/8088 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 8086/8088 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 8086/8088 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 8086/8088 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 8086/8088 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 8086/8088 macro assembler allows the extensive capabilities of the 8086/8088 to be fully exploited. In any application, time and space critical routines can be effectively written in ASM 86/88. The 8086/88 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 8086/8088 relocation and linkage utilities.

## CONV 86

### MCS-80/85™ to 86/88 ASSEMBLY LANGUAGE CONVERTER UTILITY PROGRAM

- **Translates 8080/8085 Assembly Language Source Code to 8086/8088 Assembly Language Source Code**
- **Provides a Fast and Accurate Means to Convert 8080/8085 Programs to the 8086 and the 8088, 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 8086 and the 8088. 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 8086/8088 source file with embedded caution messages.
- The OUTPUT file is an 8086/8088 source file.
- CONV 86/88 issues a caution message when it detects a potential problem in the converted 8086/8088 code.
- A transliteration of the 8080/8085 programs occurs, with each 8080/8085 construct mapped to its exact 8086/8088 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 8086/8088 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 8086/8088 assembly language to execute on that respective target machine. The conversion process is porting a source destined for the 8080/8085 to the 8086 or the 8088 via CONV 86/88.

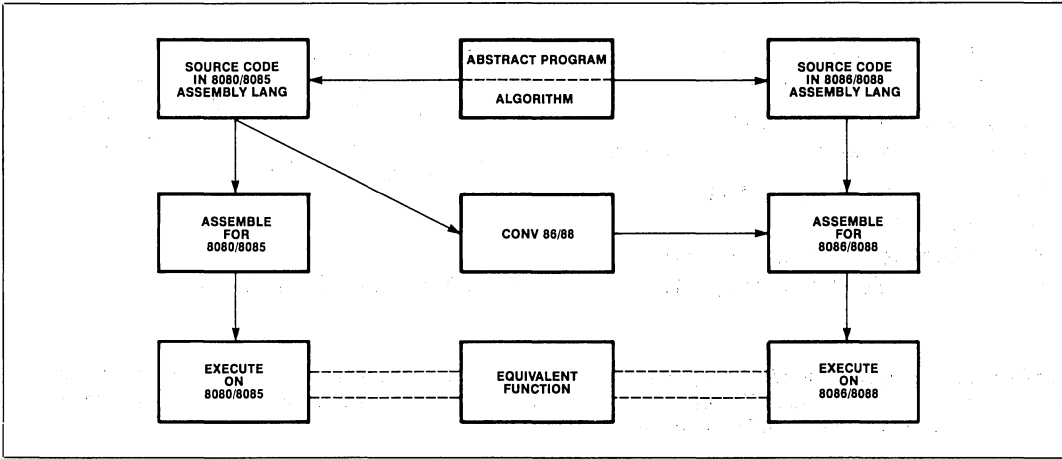


Figure 1. Porting 8080/8085 Source Code to the 8086/8088

## LINK 86

- Automatic Combination of Separately Compiled or Assembled 8086/8088 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 8086 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**
- **Automatic and Independent Relocation of Segments. Segments May Be Relocated to Best Match Users Memory Configuration**
- **Extensive Capability to Manipulate the Order and Placement of Segments in 8086/8088 Memory**
- **Extensive Debug Symbol Manipulation, Allowing Line Numbers, Local Symbols, and Public Symbols to be Purged and Listed Selectively**
- **Abbreviated Control Syntax**

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 Intel development system and then simply relocate the object code to suit your application.

## OH 86/88

- Converts an 8086/8088 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 8086/8088 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.

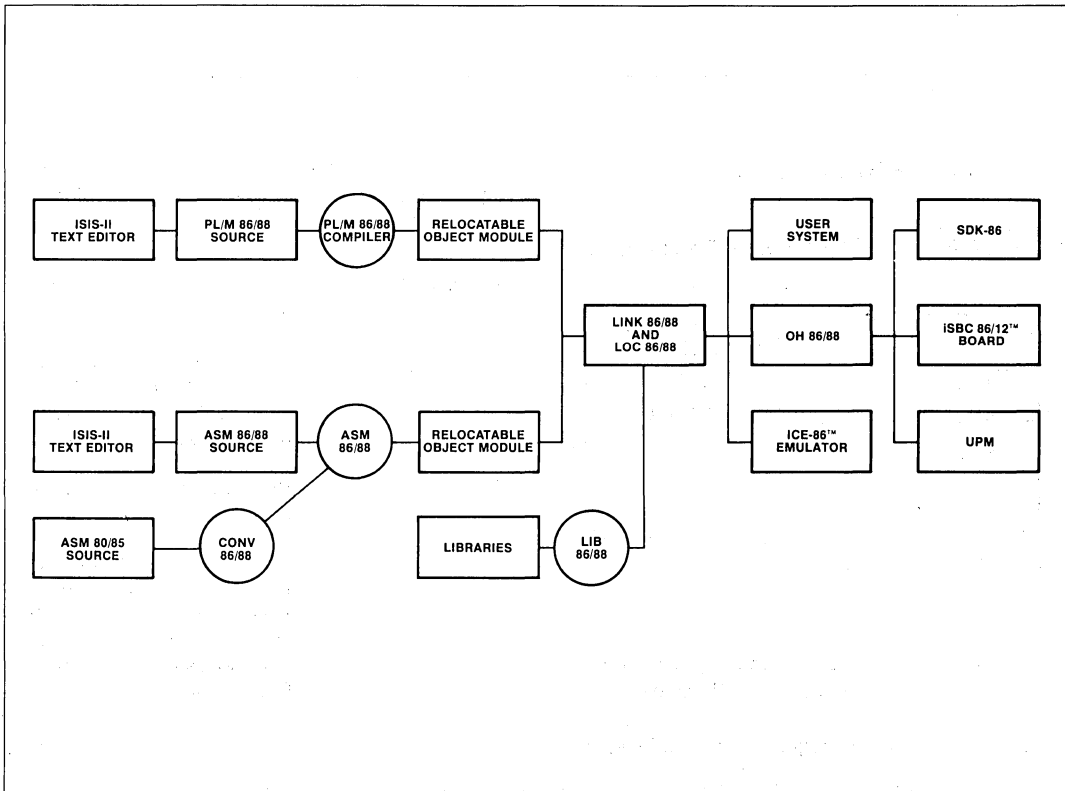


Figure 2. 8086/8088 Software Development Cycle



**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**

**Series II 8086/8088 Software Development Packages:**

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 8086/8088 Software Development Package (MDS-311)
SP86B-KIT	SP86B Support Package (for Series II)
	Includes ICE-86™ In-Circuit Emulator (MDS-86-ICE), 8086/8088 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.



## 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 Intel® 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.

## 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.

```

M:DO: /* Beginning of module */
SORTPROC: PROCEDURE (PTR, COUNT, RECSIZE, KEYINDEX) PUBLIC;
        DECLARE PTR POINTER, (COUNT, RECSIZE, KEYINDEX) INTEGER,
/* Parameters:
PTR is pointer to first record.
COUNT is number of records to be sorted.
RECSIZE is number of bytes in each record—max is 128.
KEYINDEX is byte position within each record of a BYTE scalar
to be used as sort key. */
        DECLARE RECORD BASED PTR(1) BYTE,
                CURRENT (128) BYTE,
                (I, J) INTEGER;

SORT:   DO J = 1 TO COUNT-1;
        CALL MOVB(@RECORD(J*RECSIZE), @CURRENT, RECSIZE);
        I = J;
FIND:   DO WHILE I > 0
        AND RECORD((I-1)*RECSIZE+KEYINDEX)
        < CURRENT(KEYINDEX);
        CALL MOVB(@RECORD((I-1)*RECSIZE),
                @RECORD(I*RECSIZE),
                RECSIZE);
        I = I - 1;
        END FIND;
        CALL MOVB(@CURRENT, @RECORD(I*RECSIZE), RECSIZE);
END SORT;
END SORTPROC;
END M; /*End of module*/

```

PUBLIC and EXTERNAL attributes promote program modularity.

“Based” Variables allow manipulation of external data by passing the base of the data structure (a pointer). This minimizes the STACK space used for parameter passing, and the execution time to perform many STACK operations.

The “AT” operator returns the address of a variable, instead of its contents. This is very useful in passing pointers for based variables.

One of several PL/M built-in procedures for string manipulation.

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.





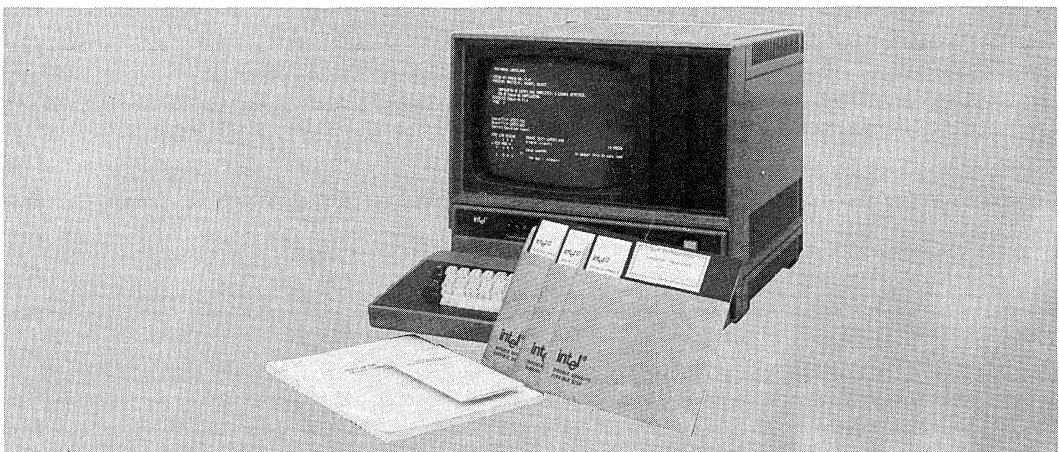
## PASCAL 86/88 SOFTWARE PACKAGE

- 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.

## 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.



## 8087 SOFTWARE SUPPORT PACKAGE

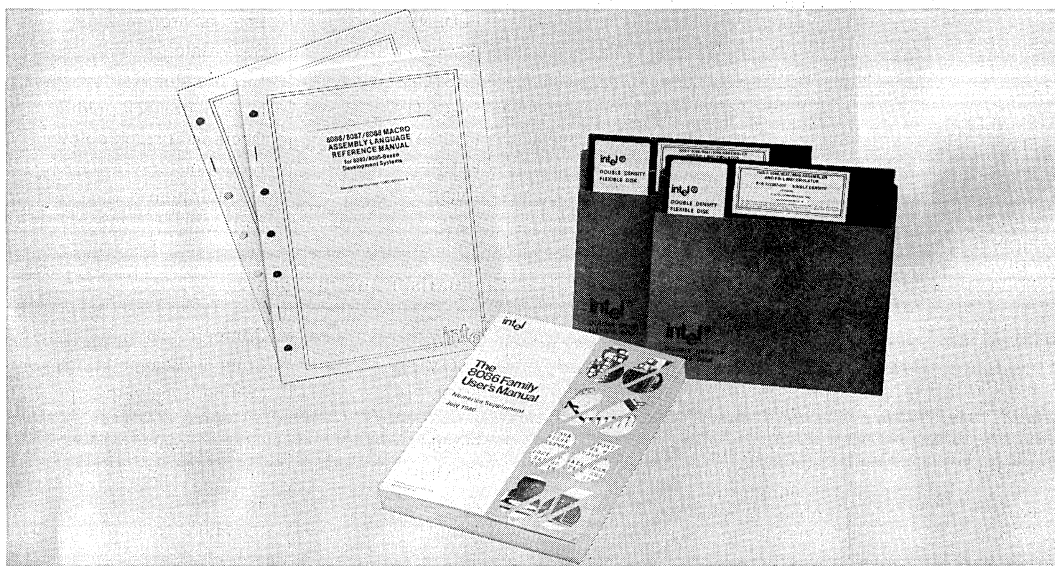
- **Program Generation for the 8087 Numeric Data Processor on the Intellec® Microcomputer Development System**
- **Consists of: 8086/8087/8088 Macro Assembler, 8087 Software Emulator**
- **Macro Assembler Generates Code for 8087 Processor or Emulator, While Also Supporting the 8086/8088 Instruction Set**
- **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 extension 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.



## 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 FADDP FIADD	Add real Add real and pop Integer add
Subtraction	
FSUB FSUBP FISUB FSUBR FSUBRP FISUBR	Subtract real Subtract real and pop Integer subtract Subtract real reversed Subtract real reversed and pop Integer subtract reversed
Multiplication	
FMUL FMULP FIMUL	Multiply real Multiply real and pop Integer multiply
Division	
FDIV FDIVP FIDIV FDIVR FDIVRP FIDIVR	Divide real Divide real and pop Integer divide Divide real reversed Divide real reversed and pop Integer divide reversed
Other Operations	
FSQRT FSCALE FPREM FRNDINT FXTRACT FABS FCHS	Square root Scale Partial remainder Round to integer Extract exponent and significant Absolute value 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 $\pi$
FLDL2T	Load $\log_2 10$
FLDL2E	Load $\log_2 e$
FLDLG2	Load $\log_{10} 2$
FLDLN2	Load $\log_2 2$

**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.

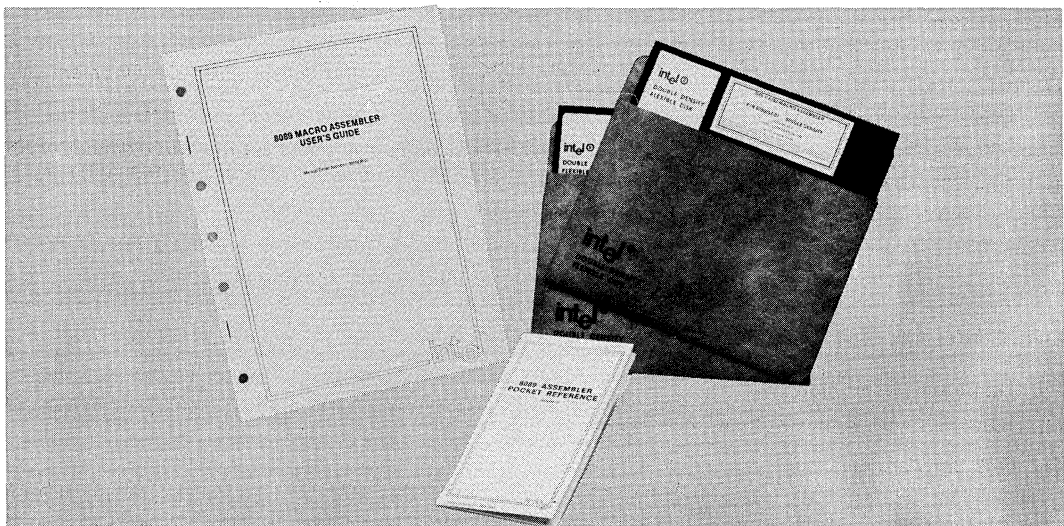


## 8089 IOP SOFTWARE SUPPORT PACKAGE

- **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.



## 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

### 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.

\*Recommended



### Table 1. Sample Program Listing

```

8089 MACRO ASSEMBLER

1313-11 8089 MACRO ASSEMBLER X10S ASSEMBLY OF MODULE TASK
OBJECT MODULE PLACED IN FILE TASK OBJ
ASSEMBLER INVOKED BY: asm89 if1:task a99 gen macro debug pagewidth(132) print(:if1:taskx.lst)

LOC  OBJECT CODE      TIMING  INC MAC  LINE SOURCE
1  * .....
2  * .....
3  * .....
4  * .....
5  * .....
6  * .....
7  name TASK
8  TASK segment
9  * .....
10  * .....
11  * In the first part of this sample program data is moved from
12  * 8086 system RAM to memory local to the 8089 IOP. In the second
13  * part, the data is moved from the local memory to a data port
14  * also in the 8089 I/O space.
15  data@port@0251 equ  Bc000h      ;0251 DP on 8089 local bus
16  command@port@0251 equ  Bc001h   ;0251 CP on 8089 local bus
17  buffer@0809 equ  B200h        ;RAM buffer in 8089 I/O space
18  * .....
19  extrn  buffer@0806           ;RAM buffer in 8086 system memory
20  extrn  y                    ;location of the buffer count
21  * .....
22  %define (macro_1)
23  ( movl  gb,buffer@0809      ; Move buffer address into GB
24  lpd:  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  )
8000  1100  00000000          30  46  32 ONE: lpd:  ga,buffer@0806      ; Load register GA with address
33  ; of 8086 buffer
34  %macro_1
8006  3130  0002              47  71  +1  35  movl  gb,buffer@0809      ; Move buffer address into GB
8009  5100  00000000          77  117 +1  36  lpd:  gc,y                ; Load pointer to count into GC
8010  0802              92  139 +1  37  movb  bc,[gc]             ; Move byte count into BC
8012  0890  00CD            124  185 39 loop00: movb [gb],[gc]      ; Move byte from 8086 to 8089 buffer
8016  0030              134  202 40 inc  ga                    ; Increment pointer into buffer
41 %macro_2(gb,gc)
8018  2030              144  219 +1  42  inc  %PARAM_1           ; Increment pointer into source
801A  403C              154  236 +1  43  gb
44  dec  %PARAM_2           ; Decrement byte count
801C  4040  F3            173  259 +1  45  gc
46  jnz  %PARAM_2           ; Decrement byte count
47  gc,%LOOP
48  LOOP00                ; Loop back if byte count > 0
49
801F  1130  00CD            191+ 294 50 TWO: movl  ga,data@port@0251    ; load GA with address of 0251 DP
8023  1130  01CD            209 309 51  movl  ga,command@port@0251    ; load GC with address of 0251 CP
52 %macro_1
8027  3130  0002              227  334 +1  53  movl  gb,buffer@0809      ; Move buffer address into GB
8028  5100  00000000          262  380 +1  54  lpd:  gc,y                ; Load pointer to count into GC
8031  0802              281  402 +1  55  movb  bc,[gc]             ; Move byte count into BC
56  )
8033  006A  FD              386+ 434 57 loop01: jnbt [gc],0,loop01    ; loop until 0251 transmit ready
8036  0091  00CC            330  400 58  movb  [ga],[gb]           ; move message into buffer
59 %macro_2(gb,gc)
803H  2030              340  497 +1  60  inc  %PARAM_1           ; Increment pointer into source
803C  403C              350  514 +1  61  gb
62  dec  %PARAM_2           ; Decrement byte count
803E  4040  F2            377  537 +1  63  gc
64  jnz  %PARAM_2           ; Decrement byte count
65  gc,%LOOP
66  LOOP01                ; Loop back if byte count > 0
67
8041  2040              399  562 68  hlt
8043  09 TASK ends
70 END

ASSEMBLY COMPLETE, NO ERRORS FOUND

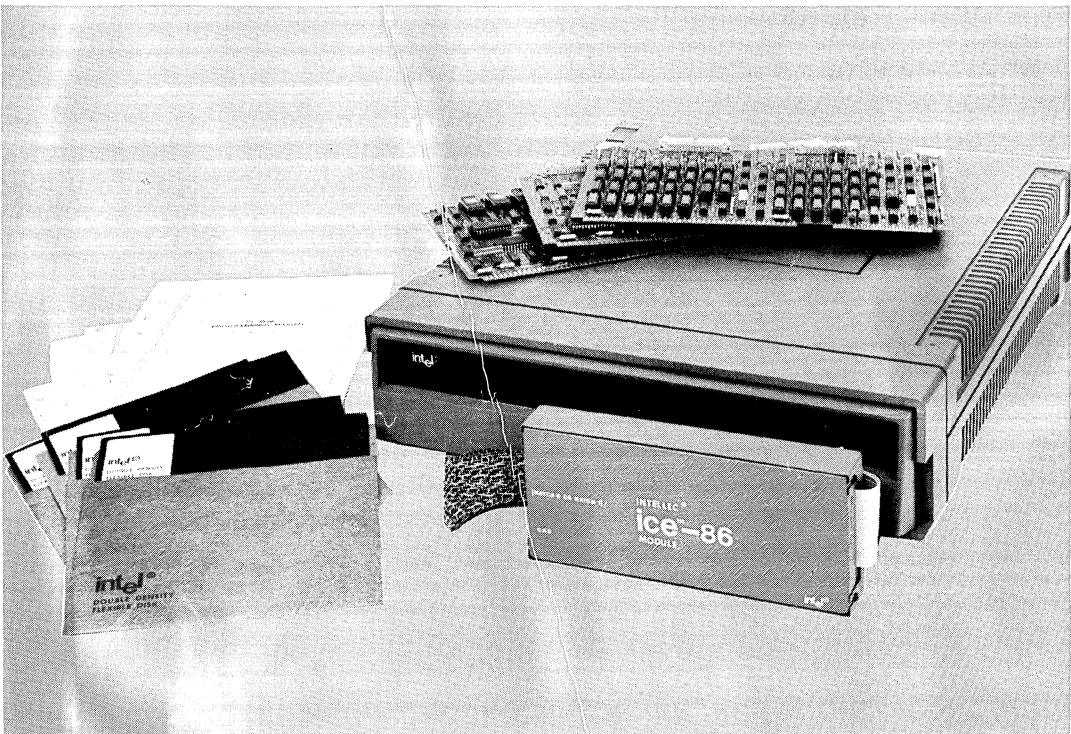
```



## SP86A/SP86B SUPPORT PACKAGE

- Full Development Support for 8086 Microprocessor Designs, Including:
- 8086/8088 Software Development Package for High-Level Language or Assembly Language Programming
- ICE-86™ In-Circuit Emulation
- Expansion Chassis for Development Support on Intellec® Series II Microcomputer Development System
- Programs Written in PL/M and Assembly Language can be Linked Together and Relocated
- Programs Written in 8080/8085 Assembly Language can be Converted to 8086/8088 Assembly Language
- ICE-86™ Emulation Provides Fully Symbolic Debugging for Program Labels or Variables

The SP86A and SP86B Support Packages combine the software and hardware components required for 8086 development using an Intellec Microcomputer Development System. The basic components are an 8086 Software Development Package and ICE-86 in-circuit emulator. The SP86B version of the kit includes an expansion chassis to provide additional board slots and power for the Intellec Series II.



## SP86A/SP86B SUPPORT PACKAGE

### SP86A AND SP86B SUPPORT PACKAGE SPECIFICATIONS

#### Operating Environment

##### Required Hardware

DS003 System Package or equivalent.

#### Environmental Characteristics

**Operating Temperature:** 0°C to 40°C

**Operating Humidity:** Up to 95% relative humidity without condensation.

#### Expansion Chassis Dimensions

**Width** — 17.37 in. (44.12 cm)

**Height** — 4.81 in. (17.22 cm)

**Depth** — 19.13 in. (48.59 cm)

#### Reference Manuals (supplied)

PL/M-86 Programming Manual (9800466)

ISIS-II PL/M-86 Compiler Operator's Manual (9800478)

MCS™-86 User's Manual (9800694)

MCS-86 Software Development Utilities Operating Instructions for ISIS-II Users (9800639)

MCS-86 Assembly Language Reference Manual (9800640)

MCS-86 Assembler Operating Instructions for ISIS-II Users (9800641)

MCS-86 Assembly Language Converter Operating Instructions for ISIS-II Users (9800642)

MCS-86 Absolute Object File Formats (9800821)

Universal PROM Programmer User's Manual (9800819A)

ICE-86 Operating Instructions (9800714A)

#### Electrical Characteristics

##### DC Power Supply

Voltage	Expansion Chassis Amps Supplied	ICE-86 In-Circuit Emulator Amps Required	
		Maximum	Typical
+5 ± 5% - 4%	.24	15	11
+12 ± 5%	2.0	0.12	0.08
-12 ± 5%	0.3	none	none
-10 ± 5%	1.0	0.015	0.012

**AC Requirements** — 50-60 Hz, 115/230 V<sub>AC</sub>

### ORDERING INFORMATION

Part Number	Description
SP86A-KIT	SP86A Support Package for Intellec Model 800 Includes ICE-86 In-Circuit Emulator (MDS-86-ICE) and 8086/8088 Software Development Package (MDS-311).
SP86B-KIT	SP86B Support Package for Intellec Series II Includes ICE-86 In-Circuit Emulator (MDS-86-ICE), 8086/8088 Software Development Package (MDS-311), and Series II Expansion Chassis (MDS-201).



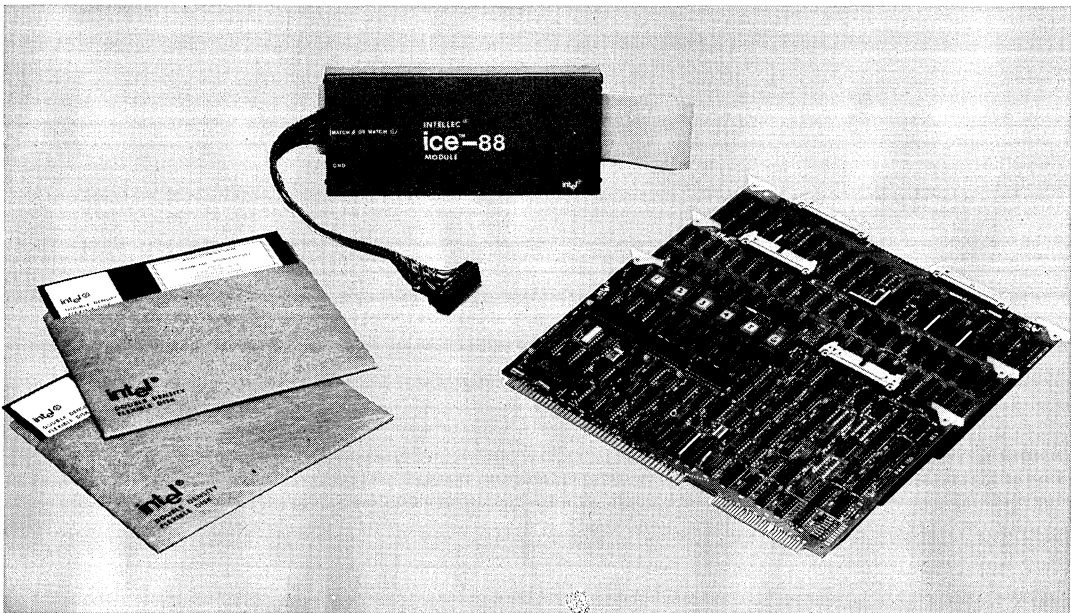
## SP 88 SUPPORT PACKAGE

- Development Support for iAPX 88 Microprocessor Designs
- iAPX 88 Macro Assembler for 8088 Assembly Language Programming
- ICE-88™-iAPX 88 CPU In-Circuit Emulator for Real-Time Emulation and Full Symbolic Debugging
- Link and Locate Utilities Program for Combination of Separately Assembled 8088 Programs into a Relocatable Module
- Utility Program for Conversion of 8080/8085 Source Program into 8088 Source Program
- Object Code Converter Which Translates 8088 Absolute Module to Symbolic HEX Format for Easier Debugging
- Library Manager Program for Creation and Maintenance of Object Files

The SP 88 support package provides both iAPX 88 assembler and ICE-88 emulation which may be combined with the Series II Inteltec Microcomputer Development System.

The iAPX 88 assembler outputs relocatable and linkable object modules. iAPX 88 programs can be developed and debugged in small modules. These modules can be easily linked with other iAPX 88 assembler or PL/M 88 object modules or library routines to form a complete application system.

The assembler translates symbolic assembly language instructions into the appropriate machine code, which can be loaded to an ICE-88 in-circuit emulator module for software execution and symbolic debugging.



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 in place of the user's 8088. Powerful debug functions are thus extended into the user system. Using the ICE-88 module, the designer can execute software on prototype hardware in real-time or single-step mode. Breakpoints allow the user to halt emulation on user-specified conditions. Trace capability gives 1023 frames of the 8088 program execution status prior to the halt. User access to the prototype system software is facilitated with the symbolic debugging feature which allows reference to the source program variables and labels.

The ICE-88 emulator allows hardware and software development to proceed concurrently and 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 it is designed. Software and hardware testing occurs while the product is being developed.

In addition, the SP 88 support package contains a utility program to translate 8080/8085 assembly or PL/M 80 language source code to 8088 assembly language source code. 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.

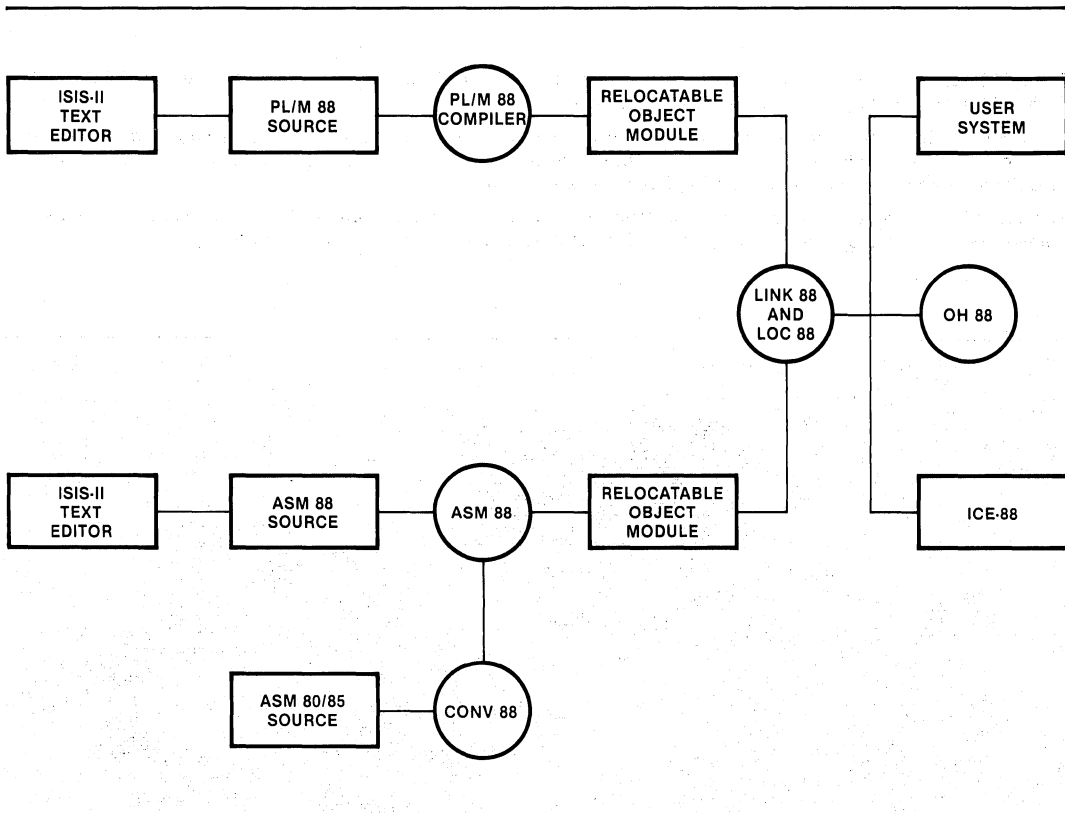


Figure 1. iAPX 88 Microprocessor Development Cycle

## SPECIFICATIONS

### OPERATING ENVIRONMENT

#### Required Hardware

Intellec microcomputer development system Models 800, 225, 235, 245 with:

1. Three adjacent slots for the ICE-88 module. (Series II-220, -230, -240 require 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 with Dual Diskette Drives — single or double density  
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  
ASM 88 software, diskette-based

#### Emulation Clock

User system clock up to 5 MHz or  
2 MHz ICE-88 emulator 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

$V_{CC} = +5V \pm 5\% - 1\%$   
 $I_{CC} = 15A$  maximum; 11A typical  
 $V_{DD} = +12V \pm 5\%$   
 $I_{DD} = 120mA$  maximum; 80mA typical  
 $V_{BB} = -10V \pm 5\%$  or  $-12V \pm 5\%$  optional  
 $I_{BB} = 25mA$  maximum; 12mA typical

### Environmental Characteristics

Operating Temperature: 0° to 40°C  
Operating Humidity: Up to 95% relative humidity without condensation

### DOCUMENTATION SUPPLIED

*PLIM 86 Programming Manual* (9800466)  
*ISIS-II PLIM 86 Compiler Operator's Manual* (9800478)  
*MCS™-86 User's Manual* (9800694)  
*MCS-86 Software Development Utilities Operating Instructions for ISIS-II Users* (9800639)  
*MCS-86 Assembly Language Reference Manual* (9800640)  
*MCS-86 Assembler Operating Instructions for ISIS-II Users* (9800641)  
*MCS-86 Assembly Language Converter Operating Instructions for ISIS-II Users* (9800642)  
*MCS-86 Absolute Object File Formats* (9800821)  
*ICE-88 Emulator Operating Instructions* (9800714A)

## ORDERING INFORMATION

Part Number: SP 88-KIT

Description: SP 88 Support Package includes ICE-88 In-Circuit Emulator (MDS-88\* ICE) and 8088 Software Development Package (MDS-308\*). 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 Corp.



## ICE-86™ 8086 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-86™ Mapped Memory
- Software Debugging with or without User System
- Handles Full 1 Megabyte Addressability of 8086
- Compound Commands
- Command Macros

The ICE-86 module provides In-Circuit Emulation for the 8086 microprocessor and the iSBC 86/12 Single Board Computer. It includes three circuit boards which reside in Inteltec® Microcomputer Development Systems. A cable and buffer box connect the Inteltec system to the user system by replacing the user's 8086. Powerful Inteltec debug functions are thus extended into the user system. Using the ICE-86 module, the designer can execute prototype software in continuous or single-step mode and can substitute blocks of Inteltec 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.



# ICE-86™ IN-CIRCUIT EMULATOR

## INTEGRATED HARDWARE/SOFTWARE DEVELOPMENT

The ICE-86 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-86 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-86 emulator assists three stages of development:

1. It can be operated without being connected to the user's system, so ICE-86 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-86 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-86 module is then used for real time emulation of the 8086 to debug the system as a completed unit.

Thus the ICE-86 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-86 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 "IVAR" displays the value in memory of variable VAR in a format appropriate to its type, while the command "IVAR = IVAR + 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-86 module provides access through symbolic definition to all of the 8086 registers and flags. The READY, NMI, TEST, HOLD, RESET, INTR, and MN/MX pins of the 8086 can also be read. Symbolic references to key ICE-86 emulation information are also provided.

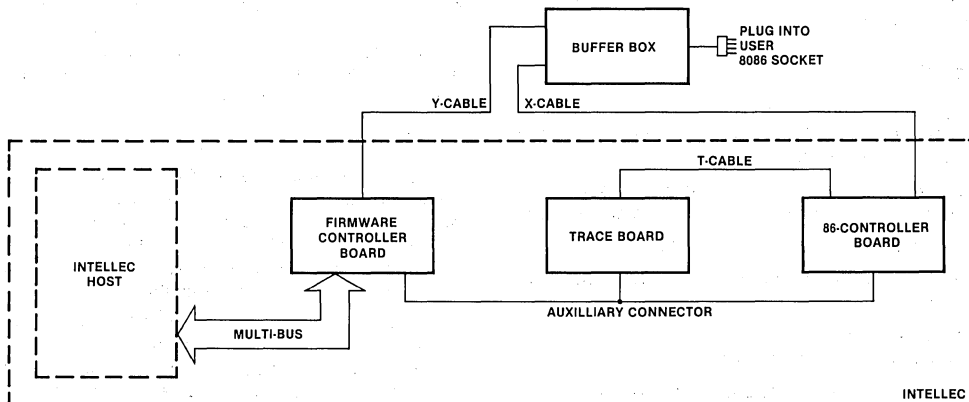
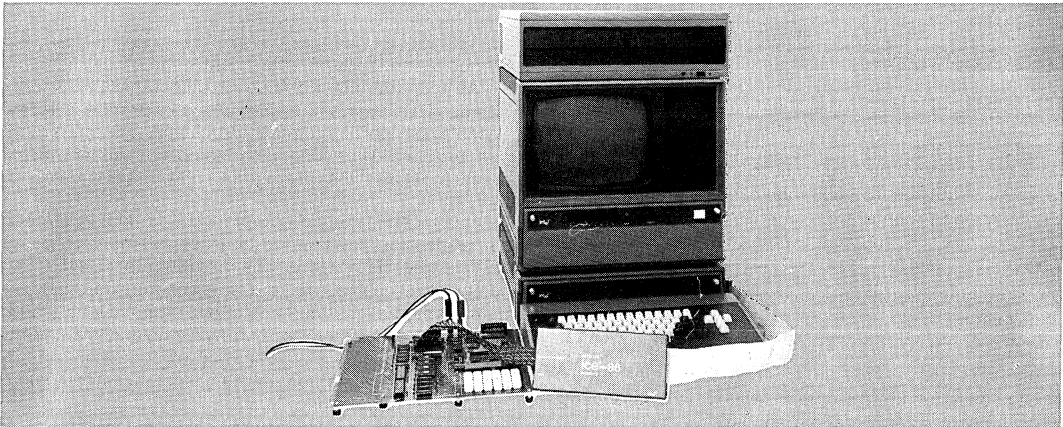


Figure 1. ICE-86™ Emulator Block Diagram



# ICE-86™ IN-CIRCUIT EMULATOR



A typical ICE-86 development configuration. It is based on a Model 230 Development System, which also includes a Double Density Diskette Operating System and a Model 201 Expansion Chassis (which holds the ICE-86 emulator). The ICE-86 module is shown connected to a user prototype system, in this case an SDK-86.

## MACROS AND COMPOUND COMMANDS

The ICE-86 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-86 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 Inteltec System through ICE-86's mapping capability.

The ICE-86 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,
2. Either of two 1K-byte blocks of ICE-86 high speed memory,
3. Inteltec memory,
4. A random-access diskette file.

The user can also designate a block of memory as non-existent. The ICE-86 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-86™ Emulation Commands

## OPERATION MODES

The ICE-86 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-86 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-86 emulator control the process of setting up, running and halting an emulation of the user's 8086. Breakpoints and tracepoints enable ICE-86 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-86 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

# ICE-86™ IN-CIRCUIT EMULATOR

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).

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-86 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 determining 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.

### Memory/Register Commands

Display or change the contents of:

- Memory
- 8086 Registers
- 8086 Status flags
- 8086 Input pins
- 8086 I/O ports
- ICE-86 Pseudo-Registers (e.g. emulation timer)

### Memory Mapping Commands

Display, declare, set, or reset the ICE-86 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-86 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-86 Ready with the user Ready signal for accessing Intellec memory, ICE memory, or diskette memory.

**Table 2. Summary of Basic ICE-86™ Interrogation and Utility Commands**

# ICE-86™ IN-CIRCUIT EMULATOR

## DIFFERENCES BETWEEN ICE-86™ EMULATION AND THE 8086 MICROPROCESSOR

The ICE-86 module emulates the actual operation of the 8086 microprocessor with the following exceptions:

- The ICE-86 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 8086, 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-86 emulator.
- The  $\overline{RQ}/\overline{GT}$  lines in the "maximum" configuration are not supported.

The speed of run emulation by the ICE-86 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 4 MHz clock
ICE	2 wait states per 8086-controlled bus cycle
Intellec	Approximately 0.02% of real time at 4 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-86™ USER CABLE

### 1. Output Low Voltages [ $V_{OL}(\text{Max}) = 0.4V$ ]

	$I_{OL}(\text{Min})$
AD0-AD15	8 mA (24 mA @ 0.5V)
A16/S3-A19/S7, $\overline{BHE}/S7$ , $\overline{RD}$ , LOCK, QS0, QS1, $\overline{S0}$ , $\overline{S1}$ , $\overline{S2}$ , $\overline{WR}$ , M/ $\overline{IO}$ , DT/ $\overline{R}$ , $\overline{DEN}$ , ALE, INTA	8 mA (16 mA @ 0.5V)
HLDA	7 mA
$\overline{MATCH0}$ OR $\overline{MATCH1}$ (on buffer box)	16 mA

### 2. Output High Voltages [ $V_{OH}(\text{Min}) = 2.4V$ ]

	$I_{OH}(\text{Min})$
AD0-AD15	- 2 mA
A16/S3-A19/S7, $\overline{BHE}/S7$ , $\overline{RD}$ , LOCK, QS0, QS1, $\overline{S0}$ , $\overline{S1}$ , $\overline{S2}$ , $\overline{WR}$ , M/ $\overline{IO}$ , DT/ $\overline{R}$ , $\overline{DEN}$ , ALE, INTA, HLDA	- 1 mA
$\overline{MATCH0}$ OR $\overline{MATCH1}$ (on buffer box)	- 0.8 mA

### 3. Input Low Voltages [ $V_{IL}(\text{Max}) = 0.8V$ ]

	$I_{IL}(\text{Max})$
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 $\mu$ f to GND)	- 3.3 mA

### 4. Input High Voltages [ $V_{IH}(\text{Min}) = 2.0V$ ]

	$I_{IH}(\text{Max})$
AD0-AD15	80 $\mu$ A
NMI, CLK	20 $\mu$ A
READY	40 $\mu$ A
INTR, HOLD, $\overline{TEST}$ , RESET	- 0.4 mA
MN/ $\overline{MX}$ (0.1 $\mu$ F to GND)	- 1.1 mA

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.

# ICE-86™ IN-CIRCUIT EMULATOR

---

## SPECIFICATIONS

### Operating Environment

#### Required Hardware

Intellec microcomputer development system with:

1. Three adjacent slots for the ICE-86 module (Series II requires Model 201 Expansion Chassis.)
2. 64K bytes of Inteltec memory. If user prototype program memory is desired, additional memory above the basic 64K is required.

System console  
Intellec diskette operating system  
ICE-86 module

#### Required Software

System monitor  
ISIS-II, version 3.4 or subsequent  
ICE-86 software

### Equipment Supplied

Printed circuit boards (3)  
Interface cable and emulation buffer module  
Operator's manual  
ICE-86 software, diskette-based

### Emulation Clock

User system clock up to 4 MHz or 2 MHz ICE-86 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

$V_{CC} = +5V \pm 5\% - 1\%$   
 $I_{CC} = 15A$  maximum; 11A typical  
 $V_{DD} = +12V \pm 5\%$   
 $I_{DD} = 120$  mA maximum; 80 mA typical  
 $V_{BB} = -10V \pm 5\%$  or  $-12V \pm 5\%$  (optional)  
 $I_{BB} = 15$  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-86-ICE\* 8086 CPU in-circuit emulator, cable assembly and interactive diskette software.

\*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.



## 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.



## 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 "IVAR" displays the value in memory of variable VAR in a format appropriate to its type, while the command "IVAR = !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, TEST, HOLD, RESET, INTR, and MN/MX pins of the 8088 can also be read. Symbolic references to key ICE-88 emulation information are also provided.

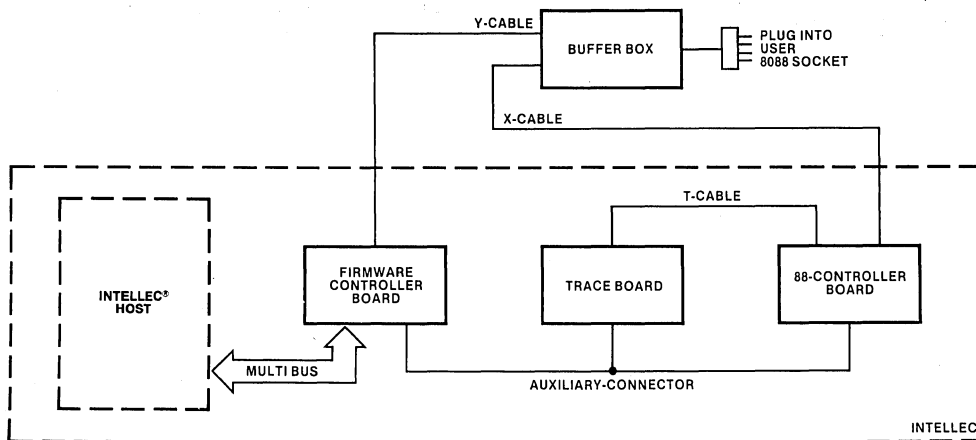


Figure 1. ICE-88™ Emulator Block Diagram  
15-37



## ICE-88™ IN-CIRCUIT EMULATOR

### 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 [ $V_{OL}(\text{Max}) = 0.4\text{V}$ ]

	$I_{OL}(\text{Min})$
$\overline{AD0-AD7}$ , $\overline{A8-A15}$ , $\overline{SS0}$ , $\overline{A16/S3-A19/S7}$ , $\overline{RD}$ ,	8 mA
$\overline{LOCK}$ , $\overline{QS0}$ , $\overline{QS1}$ , $\overline{S0}$ , $\overline{S1}$ , $\overline{S2}$ , $\overline{WR}$ , $\overline{M/I\overline{O}}$ , $\overline{DT/R}$ , $\overline{DEN}$ , $\overline{ALE}$ , $\overline{INTA}$	(16 mA @ 0.5V)
$\overline{HLDA}$	5 mA
$\overline{MATCH0}$ OR $\overline{MATCH1}$ (on buffer box)	16 mA

#### 2. Output High Voltages [ $V_{OH}(\text{Min}) = 2.4\text{V}$ ]

	$I_{OH}(\text{Min})$
$\overline{AD0-AD7}$ , $\overline{A8-A15}$ $\overline{A16/S3-A19/S7}$ , $\overline{SS0}$ , $\overline{RD}$ ,	-2.0 mA
$\overline{LOCK}$ , $\overline{QS0}$ , $\overline{QS1}$ , $\overline{S0}$ , $\overline{S1}$ , $\overline{S2}$ , $\overline{WR}$ , $\overline{M/I\overline{O}}$ , $\overline{DT/R}$ , $\overline{DEN}$ , $\overline{ALE}$ , $\overline{INTA}$	
$\overline{HLDA}$	-3.0 mA
$\overline{MATCH0}$ OR $\overline{MATCH1}$ (on buffer box)	-0.8 mA

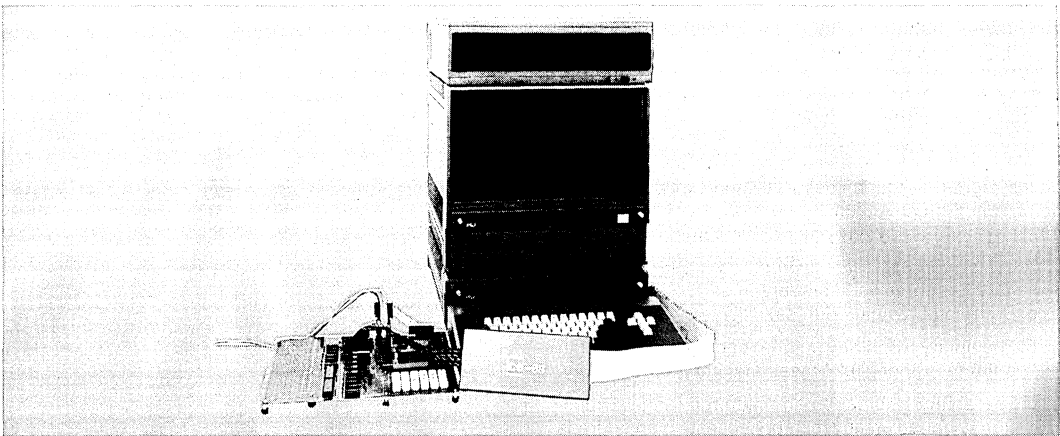
#### 3. Input Low Voltages [ $V_{IL}(\text{Max}) = 0.8\text{V}$ ]

	$I_{IL}(\text{Max})$
$\overline{AD0-AD7}$	-0.2 mA
$\overline{NMI}$ , $\overline{CLK}$	-0.4 mA
$\overline{READY}$	-0.8 mA
$\overline{INTR}$ , $\overline{HOLD}$ , $\overline{TEST}$ , $\overline{RESET}$	-1.4 mA
$\overline{MN/MX}$ (0.1 $\mu\text{F}$ to GND)	-3.3 mA

#### 4. Input High Voltages [ $V_{IH}(\text{Min}) = 2.0\text{V}$ ]

	$I_{IH}(\text{Max})$
$\overline{AD0-AD7}$	80 $\mu\text{A}$
$\overline{NMI}$ , $\overline{CLK}$	20 $\mu\text{A}$
$\overline{READY}$	60 $\mu\text{A}$
$\overline{INTR}$ , $\overline{HOLD}$ , $\overline{TEST}$ , $\overline{RESET}$	-0.4 mA
$\overline{MN/MX}$ (0.1 $\mu\text{F}$ to GND)	-1.1 mA

5.  $\overline{RQ/GT0}$ ,  $\overline{RQ/GT1}$  are pulled up to +5V through a 5.6K ohm resistor. No current is taken from user circuit at  $V_{CC}$  pin.



A typical ICE-88 development configuration. It is based on a Model 230 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





## ICE-88™ IN-CIRCUIT EMULATOR

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 Intellect memory or diskette memory.

#### ENABLE/DISABLE RDY

Enable or disable logical AND of ICE-88 Ready with the user Ready signal for accessing Intellect memory, ICE memory, or diskette memory.

Table 2. Summary of Basic ICE-88 Interrogation and Utility Commands.



## ICE-88™ IN-CIRCUIT EMULATOR

---

### 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

$V_{CC} = +5V \pm 5\% - 1\%$   
 $I_{CC} = 15A$  maximum; 11A typical  
 $V_{DD} = +12V \pm 5\%$   
 $I_{DD} = 120mA$  maximum; 80mA typical  
 $V_{BB} = -10V \pm 5\%$  or  $-12V \pm 5\%$  (optional)  
 $I_{BB} = 25mA$  maximum; 12mA 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.

---

Faint, illegible text covering the page, possibly bleed-through from the reverse side. The text is too light to transcribe accurately.

---

*Prototype  
Microcomputer Kits*

**16**

---



# CONFIDENTIAL

## SECURITY INFORMATION

Page 1

[Faint, illegible text block]

[Faint, illegible text block]

[Faint, illegible text block]

[Faint, illegible text block]

[Faint, illegible text block]

[Faint, illegible text block]

[Faint, illegible text block]

[Faint, illegible text block]

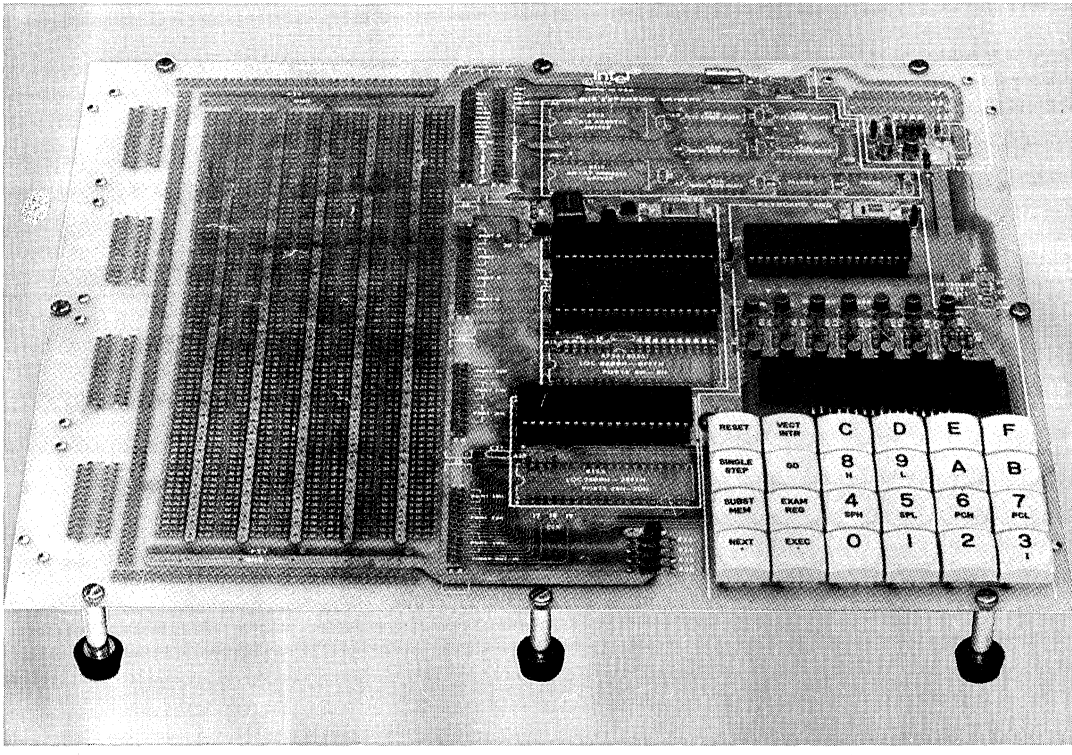
[Faint, illegible text block]



## 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  $\mu$ s 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.



# SDK-85

## 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).

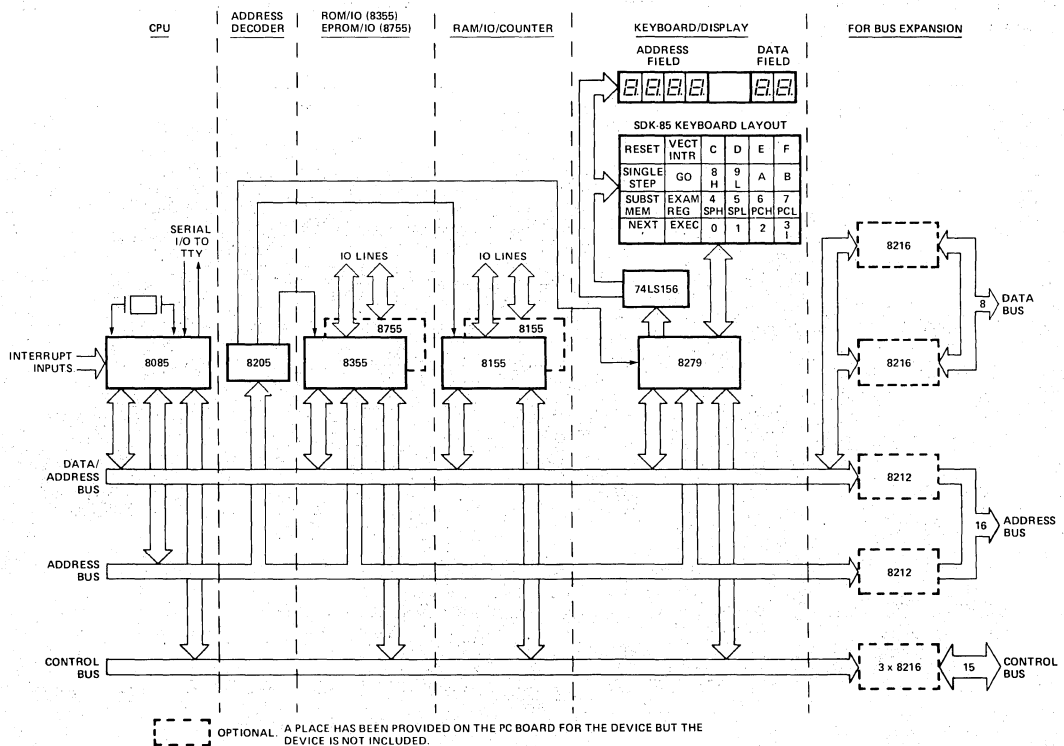
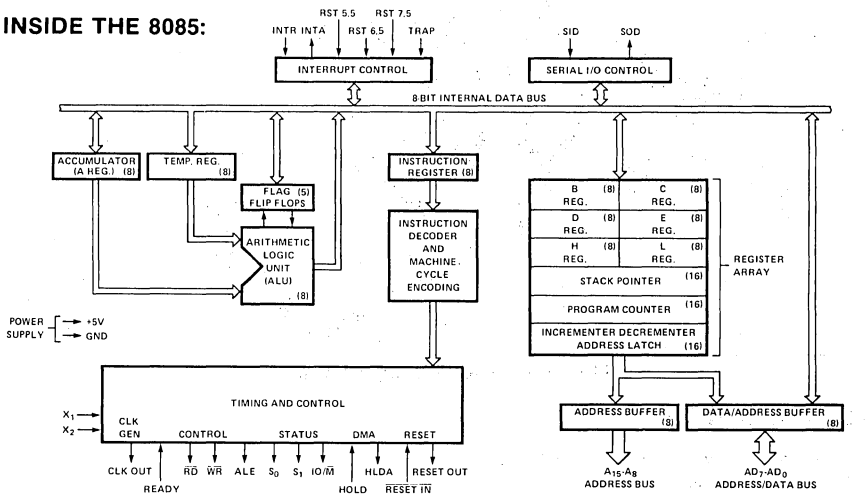


Figure 1. SDK-85 System Design Kit Functional Block Diagram

**INSIDE THE 8085:**



- 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.

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.

**Table 1. Keyboard Monitor Commands**

**Commands** — Keyboard monitor commands and teletype monitor commands are provided in Table 1 and Table 2 respectively.

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.

**Table 2. Teletype Monitor Commands**

**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.

Mnemonic <sup>1</sup>	Description	Instruction Code <sup>2</sup>							Clock <sup>3</sup> Cycles	Mnemonic <sup>1</sup>	Description	Instruction Code <sup>2</sup>							Clock <sup>3</sup> Cycles				
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>				D <sub>0</sub>	D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>		D <sub>1</sub>	D <sub>0</sub>		
<b>MOVE, LOAD, AND STORE</b>																							
MOV r1r2	Move register to register	0	1	D	D	D	S	S	S	4	LXI SP	Load immediate stack pointer	0	0	1	1	0	0	0	1	10		
MOV M,r	Move register to memory	0	1	1	1	0	S	S	S	7	INX SP	Increment stack pointer	0	0	1	1	0	0	1	1	6		
MOV r,M	Move memory to register	0	1	D	D	D	1	1	0	7	DCX SP	Decrement stack pointer	0	0	1	1	1	0	1	1	6		
MVI r	Move immediate register	0	0	D	D	D	1	1	0	7	<b>JUMP</b>												
MVI M	Move immediate memory	0	0	1	1	0	1	1	0	10	JMP	Jump unconditional	1	1	0	0	0	0	1	1	10		
LXI B	Load immediate register Pair B & C	0	0	0	0	0	0	0	1	10	JC	Jump on carry	1	1	0	1	1	0	1	0	7/10		
LXI D	Load immediate register Pair D & E	0	0	0	1	0	0	0	1	10	JNC	Jump on no carry	1	1	0	1	0	0	1	0	7/10		
LXI H	Load immediate register Pair H & L	0	0	1	0	0	0	0	1	10	JZ	Jump on zero	1	1	0	0	1	0	1	0	7/10		
STAX B	Store A indirect	0	0	0	0	0	0	1	0	7	JNZ	Jump on no zero	1	1	0	0	0	0	1	0	7/10		
STAX D	Store A indirect	0	0	0	1	0	0	1	0	7	JP	Jump on positive	1	1	1	0	0	1	0	7/10			
LDAX B	Load A indirect	0	0	0	0	1	0	1	0	7	JM	Jump on minus	1	1	1	1	0	1	0	7/10			
LDAX D	Load A indirect	0	0	0	1	1	0	1	0	7	JPE	Jump on parity even	1	1	1	0	1	0	1	0	7/10		
STA	Store A direct	0	0	1	1	0	0	1	0	13	JPO	Jump on parity odd	1	1	1	0	0	0	1	0	7/10		
LDA	Load A direct	0	0	1	1	1	0	1	0	13	PCHL	H & L to program counter	1	1	1	0	1	0	0	1	6		
SHLD	Store H & L direct	0	0	1	0	0	0	1	0	16	<b>CALL</b>												
LHLD	Load H & L direct	0	0	1	0	1	0	1	0	16	CALL	Call unconditional	1	1	0	0	1	1	0	1	18		
XCHG	Exchange D & E, H & L registers	1	1	1	0	1	0	1	1	4	CC	Call on carry	1	1	0	1	1	1	0	0	9/18		
<b>STACK OPS</b>													CNC	Call on no carry	1	1	0	1	0	1	0	0	9/18
PUSH B	Push register pair B & C on stack	1	1	0	0	0	1	0	1	12	CZ	Call on zero	1	1	0	0	1	1	0	0	9/18		
PUSH D	Push register pair D & E on stack	1	1	0	1	0	1	0	1	12	CNZ	Call on no zero	1	1	0	0	0	1	0	0	9/18		
PUSH H	Push register pair H & L on stack	1	1	1	0	0	1	0	1	12	CP	Call on positive	1	1	1	1	0	1	0	0	9/18		
PUSH PSW	Push A and flags on stack	1	1	1	1	0	1	0	1	12	CM	Call on minus	1	1	1	1	1	1	0	0	9/18		
POP B	Pop register pair B & C off stack	1	1	0	0	0	0	0	1	10	CPE	Call on parity even	1	1	1	0	1	1	0	0	9/18		
POP D	Pop register pair D & E off stack	1	1	0	1	0	0	0	1	10	CPO	Call on parity odd	1	1	1	0	0	1	0	0	9/18		
POP H	Pop register pair H & L off stack	1	1	1	0	0	0	0	1	10	<b>RETURN</b>												
POP PSW	Pop A and flags off stack	1	1	1	1	0	0	0	1	10	RET	Return	1	1	0	0	1	0	0	1	10		
XTHL	Exchange top of stack, H & L	1	1	1	0	0	0	1	1	16	RC	Return on carry	1	1	0	1	1	0	0	0	6/12		
SPHL	H & L to stack pointer	1	1	1	1	1	0	0	1	6	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	0	0	0	0	0	6/12
													RM	Return on minus	1	1	1	1	1	0	0	0	6/12

continued

# SDK-85

Mnemonic <sup>1</sup>	Description	Instruction Code <sup>2</sup>							Clock <sup>3</sup> Cycles	Mnemonic <sup>1</sup>	Description	Instruction Code <sup>2</sup>							Clock <sup>3</sup> Cycles		
		D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>				D <sub>0</sub>	D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>		D <sub>1</sub>	D <sub>0</sub>
RPE	Return on parity even	1	1	1	0	1	0	0	0	6/12	<b>LOGICAL</b>										
RPO	Return on parity odd	1	1	1	0	0	0	0	0	6/12	ANA r	And register with A	1	0	1	0	0	S	S	S	4
<b>RESTART</b>										XRA r	Exclusive Or register with A	1	0	1	0	1	S	S	S	4	
RST	Restart	1	1	A	A	A	1	1	1	12	ORA r	Or register with A	1	0	1	1	0	S	S	S	4
<b>INCREMENT AND DECREMENT</b>										CMP r	Compare register with A	1	0	1	1	1	S	S	S	4	
INR r	Increment register	0	0	D	D	D	1	0	0	4	ANA M	And memory with A	1	0	1	0	0	1	1	0	7
DCR r	Decrement register	0	0	D	D	D	1	0	1	4	XRA M	Exclusive Or memory with A	1	0	1	0	1	1	1	0	7
INR M	Increment memory	0	0	1	1	0	1	0	0	10	ORA M	Or memory with A	1	0	1	1	0	1	1	0	7
DCR M	Decrement memory	0	0	1	1	0	1	0	1	10	CMP M	Compare memory with A	1	0	1	1	1	1	1	0	7
INX B	Increment B & C registers	0	0	0	0	0	0	1	1	6	ANI	And immediate with A	1	1	1	0	0	1	1	0	7
INX D	Increment D & E registers	0	0	0	1	0	0	1	1	6	XRI	Exclusive Or immediate with A	1	1	1	0	1	1	1	0	7
INX H	Increment H & L registers	0	0	1	0	0	0	1	1	6	ORI	Or immediate with A	1	1	1	1	0	1	1	0	7
DCX B	Decrement B & C	0	0	0	0	1	0	1	1	6	CPI	Compare immediate with A	1	1	1	1	1	1	1	0	7
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	<b>ROTATE</b>										
<b>ADD</b>										RLC	Rotate A left	0	0	0	0	0	1	1	1	4	
ADD r	Add register to A	1	0	0	0	0	S	S	S	4	RRC	Rotate A right	0	0	0	0	1	1	1	1	4
ADC r	Add register to A with carry	1	0	0	0	1	S	S	S	4	RAL	Rotate A left through carry	0	0	0	1	0	1	1	1	4
ADD M	Add memory to A	1	0	0	0	0	1	1	0	7	RAR	Rotate A right through carry	0	0	0	1	1	1	1	1	4
ADC M	Add memory to A with carry	1	0	0	0	1	1	1	0	7	<b>SPECIALS</b>										
ADI	Add immediate to A	1	1	0	0	0	1	1	0	7	CMA	Complement A	0	0	1	0	1	1	1	1	4
ACI	Add immediate to A with carry	1	1	0	0	1	1	1	0	7	STC	Set carry	0	0	1	1	0	1	1	1	4
DAD B	Add B & C to H & L	0	0	0	0	1	0	0	1	10	CMC	Complement carry	0	0	1	1	1	1	1	1	4
DAD D	Add D & E to H & L	0	0	0	1	1	0	0	1	10	DAA	Decimal adjust A	0	0	1	0	0	1	1	1	4
DAD H	Add H & L to H & L	0	0	1	0	1	0	0	1	10	<b>INPUT/OUTPUT</b>										
DAD SP	Add stack pointer to H & L	0	0	1	1	1	0	0	1	10	IN	Input	1	1	0	1	1	0	1	1	10
<b>SUBTRACT</b>										OUT	Output	1	1	0	1	0	0	1	1	10	
SUB r	Subtract register from A	1	0	0	1	0	S	S	S	4	<b>CONTROL</b>										
SBB r	Subtract register from A with borrow	1	0	0	1	1	S	S	S	4	EI	Enable interrupts	1	1	1	1	1	0	1	1	4
SUB M	Subtract memory from A	1	0	0	1	0	1	1	0	7	DI	Disable interrupts	1	1	1	1	0	0	1	1	4
SBB M	Subtract memory from A with borrow	1	0	0	1	1	1	1	0	7	NOP	No-operation	0	0	0	0	0	0	0	0	4
SUI	Subtract immediate from A	1	1	0	1	0	1	1	0	7	HLT	Halt	0	1	1	1	0	1	1	0	5
SBI	Subtract immediate from A with borrow	1	1	0	1	1	1	1	0	7	<b>NEW 8085 INSTRUCTIONS</b>										
										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.

**Table 3. Summary of 8085A Processor Instructions**

## SPECIFICATIONS

### Central Processor

**CPU** — 8085A

**Instruction Cycle** — 1.3 μs

**T<sub>cy</sub>** — 330 ns

### Memory

**ROM** — 2K bytes (expandable to 4K bytes) 8355/8755A

**RAM** — 256 bytes (expandable to 512 bytes) 8155

### Addressing

**ROM** — 0000–07FF (expandable 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.

## SDK-85

### 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
$V_{CC} 5V \pm 5\%$	1.3A
$V_{TTY} - 10V \pm 10\%$	0.3A (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

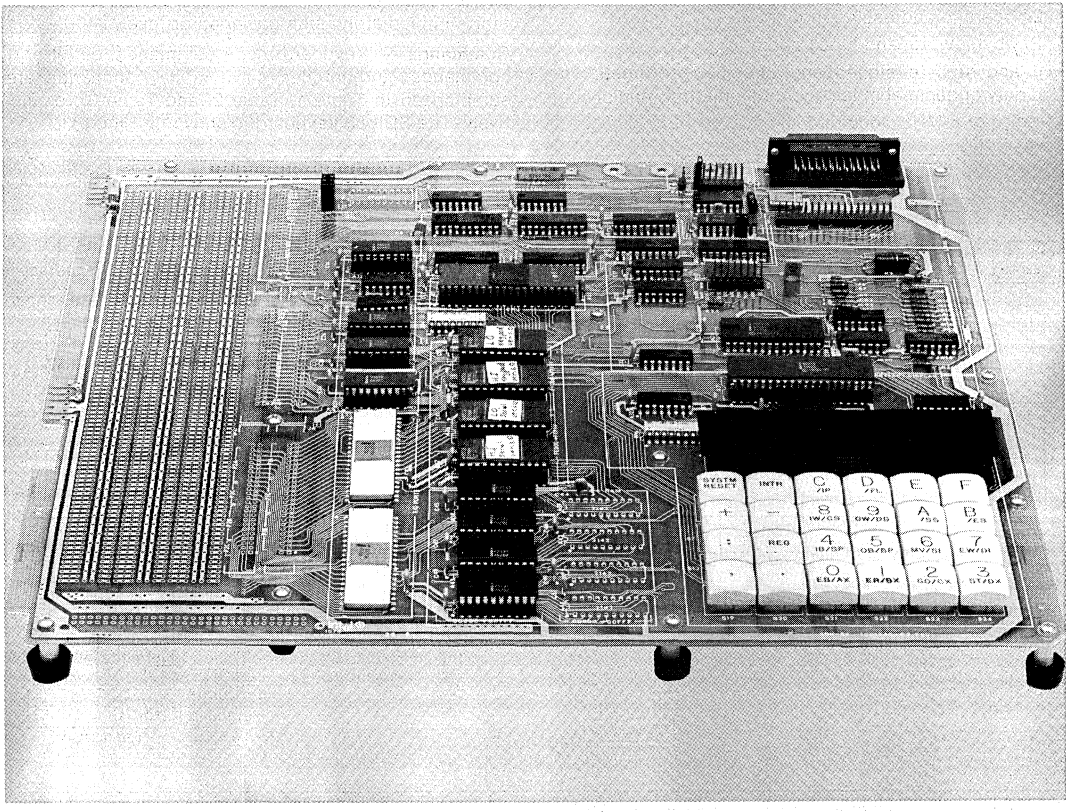
Part Number	Description
SDK-85	MCS-85 system design kit



## SDK-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.

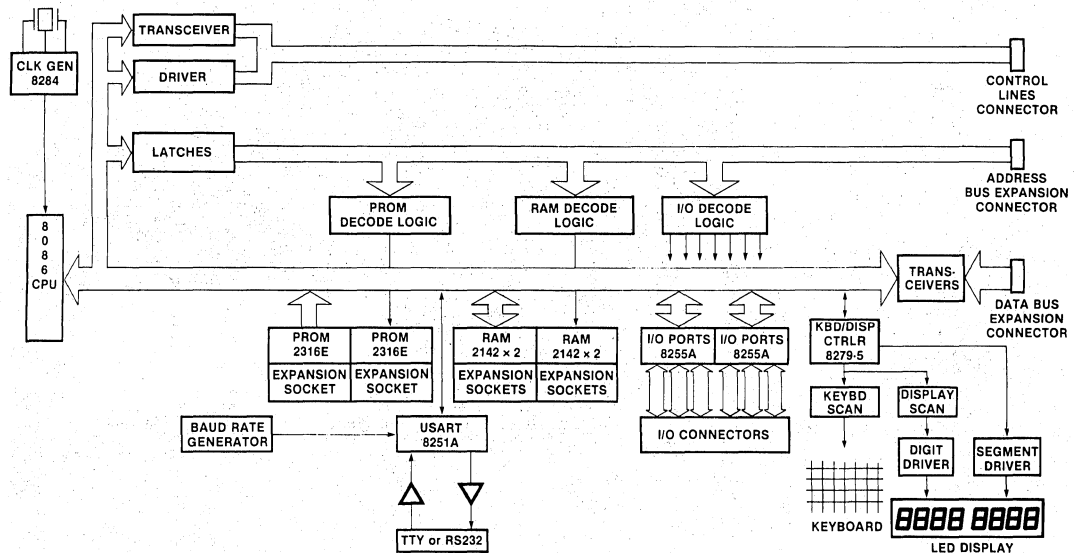
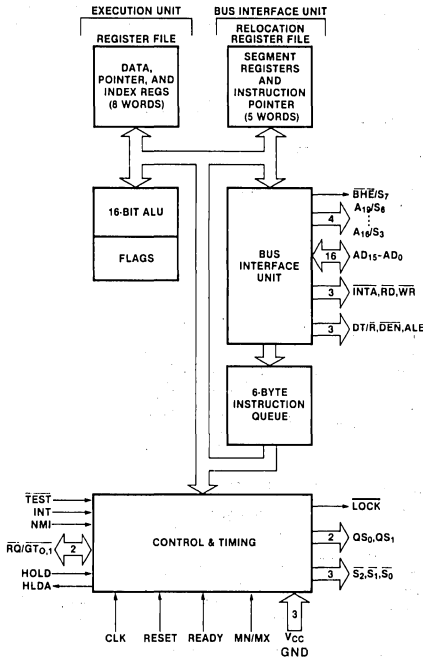


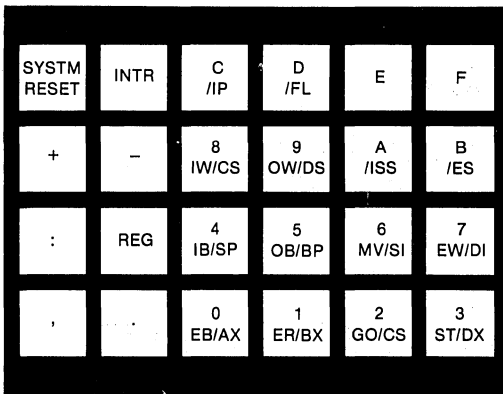
Figure 1. SDK-86 System Design Kit Functional Block Diagram

**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**

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 1. Keyboard 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.

**Table 2. Serial Mode Commands**



# SDK-86

Mnemonic and Description	Instruction Code	Mnemonic and Description	Instruction Code																																					
<b>JMP - Unconditional Jump:</b>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">7 6 5 4 3 2 1 0</td> <td style="text-align: center;">7 6 5 4 3 2 1 0</td> <td style="text-align: center;">7 6 5 4 3 2 1 0</td> </tr> <tr> <td>Direct within segment</td> <td style="text-align: center;">1 1 1 0 1 0 0 1</td> <td style="text-align: center;">disp-low    disp-high</td> </tr> <tr> <td>Direct within segment/short</td> <td style="text-align: center;">1 1 1 0 1 0 1 1</td> <td style="text-align: center;">disp</td> </tr> <tr> <td>Indirect within segment</td> <td style="text-align: center;">1 1 1 1 1 1 1 1</td> <td style="text-align: center;">mod 1 0 0 r/m</td> </tr> <tr> <td>Direct intersegment</td> <td style="text-align: center;">1 1 1 0 1 0 1 0</td> <td style="text-align: center;">offset-low    offset-high</td> </tr> <tr> <td></td> <td style="text-align: center;"></td> <td style="text-align: center;">seg-low    seg-high</td> </tr> <tr> <td>Indirect intersegment</td> <td style="text-align: center;">1 1 1 1 1 1 1 1</td> <td style="text-align: center;">mod 1 0 1 r/m</td> </tr> </table>	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	Direct within segment	1 1 1 0 1 0 0 1	disp-low    disp-high	Direct within segment/short	1 1 1 0 1 0 1 1	disp	Indirect within segment	1 1 1 1 1 1 1 1	mod 1 0 0 r/m	Direct intersegment	1 1 1 0 1 0 1 0	offset-low    offset-high			seg-low    seg-high	Indirect intersegment	1 1 1 1 1 1 1 1	mod 1 0 1 r/m	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">7 6 5 4 3 2 1 0</td> <td style="text-align: center;">7 6 5 4 3 2 1 0</td> </tr> <tr> <td>JNS - Jump on not sign</td> <td style="text-align: center;">0 1 1 1 1 0 0 1</td> <td style="text-align: center;">disp</td> </tr> <tr> <td>LOOP - Loop CX times</td> <td style="text-align: center;">1 1 1 0 0 0 1 0</td> <td style="text-align: center;">disp</td> </tr> <tr> <td>LOOPZ/LOOPPE - Loop while zero/equal</td> <td style="text-align: center;">1 1 1 0 0 0 0 1</td> <td style="text-align: center;">disp</td> </tr> <tr> <td>LOOPNZ/LOOPNE - Loop while not zero/equal</td> <td style="text-align: center;">1 1 1 0 0 0 1 1</td> <td style="text-align: center;">disp</td> </tr> <tr> <td>JCXZ - Jump on CX zero</td> <td style="text-align: center;">1 1 1 0 0 1 1 1</td> <td style="text-align: center;">disp</td> </tr> </table>	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	JNS - Jump on not sign	0 1 1 1 1 0 0 1	disp	LOOP - Loop CX times	1 1 1 0 0 0 1 0	disp	LOOPZ/LOOPPE - Loop while zero/equal	1 1 1 0 0 0 0 1	disp	LOOPNZ/LOOPNE - Loop while not zero/equal	1 1 1 0 0 0 1 1	disp	JCXZ - Jump on CX zero	1 1 1 0 0 1 1 1	disp
7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0																																						
Direct within segment	1 1 1 0 1 0 0 1	disp-low    disp-high																																						
Direct within segment/short	1 1 1 0 1 0 1 1	disp																																						
Indirect within segment	1 1 1 1 1 1 1 1	mod 1 0 0 r/m																																						
Direct intersegment	1 1 1 0 1 0 1 0	offset-low    offset-high																																						
		seg-low    seg-high																																						
Indirect intersegment	1 1 1 1 1 1 1 1	mod 1 0 1 r/m																																						
7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0																																							
JNS - Jump on not sign	0 1 1 1 1 0 0 1	disp																																						
LOOP - Loop CX times	1 1 1 0 0 0 1 0	disp																																						
LOOPZ/LOOPPE - Loop while zero/equal	1 1 1 0 0 0 0 1	disp																																						
LOOPNZ/LOOPNE - Loop while not zero/equal	1 1 1 0 0 0 1 1	disp																																						
JCXZ - Jump on CX zero	1 1 1 0 0 1 1 1	disp																																						
<b>RET - Return from CALL:</b>	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">1 1 0 0 0 0 1 1</td> <td></td> <td></td> </tr> <tr> <td>Within segment</td> <td style="text-align: center;">1 1 0 0 0 0 1 0</td> <td style="text-align: center;">data-low    data-high</td> </tr> <tr> <td>Within seg. adding immed to SP</td> <td style="text-align: center;">1 1 0 0 0 1 0 1</td> <td></td> </tr> <tr> <td>Intersegment</td> <td style="text-align: center;">1 1 0 0 1 0 1 1</td> <td></td> </tr> <tr> <td>Intersegment, adding immediate to SP</td> <td style="text-align: center;">1 1 0 0 1 0 1 0</td> <td style="text-align: center;">data-low    data-high</td> </tr> </table>	1 1 0 0 0 0 1 1			Within segment	1 1 0 0 0 0 1 0	data-low    data-high	Within seg. adding immed to SP	1 1 0 0 0 1 0 1		Intersegment	1 1 0 0 1 0 1 1		Intersegment, adding immediate to SP	1 1 0 0 1 0 1 0	data-low    data-high	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="text-align: center;">1 1 0 0 1 1 0 1</td> <td style="text-align: center;">type</td> </tr> <tr> <td>INT - Interrupt</td> <td></td> </tr> <tr> <td>Type specified</td> <td style="text-align: center;">1 1 0 0 1 1 0 0</td> </tr> <tr> <td>Type 3</td> <td style="text-align: center;">1 1 0 0 1 1 1 0</td> </tr> <tr> <td>INTO - Interrupt on overflow</td> <td style="text-align: center;">1 1 0 0 1 1 1 1</td> </tr> <tr> <td>IRET - Interrupt return</td> <td style="text-align: center;">1 1 0 0 1 1 1 1</td> </tr> </table>	1 1 0 0 1 1 0 1	type	INT - Interrupt		Type specified	1 1 0 0 1 1 0 0	Type 3	1 1 0 0 1 1 1 0	INTO - Interrupt on overflow	1 1 0 0 1 1 1 1	IRET - Interrupt return	1 1 0 0 1 1 1 1											
1 1 0 0 0 0 1 1																																								
Within segment	1 1 0 0 0 0 1 0	data-low    data-high																																						
Within seg. adding immed to SP	1 1 0 0 0 1 0 1																																							
Intersegment	1 1 0 0 1 0 1 1																																							
Intersegment, adding immediate to SP	1 1 0 0 1 0 1 0	data-low    data-high																																						
1 1 0 0 1 1 0 1	type																																							
INT - Interrupt																																								
Type specified	1 1 0 0 1 1 0 0																																							
Type 3	1 1 0 0 1 1 1 0																																							
INTO - Interrupt on overflow	1 1 0 0 1 1 1 1																																							
IRET - Interrupt return	1 1 0 0 1 1 1 1																																							
<b>JE/JZ - Jump on equal/zero</b>	0 1 1 1 0 1 0 0	disp																																						
<b>JL/JNGE - Jump on less/not greater or equal</b>	0 1 1 1 1 1 0 0	disp																																						
<b>JLE/JNB - Jump on less or equal/not greater</b>	0 1 1 1 1 1 1 0	disp																																						
<b>JB/JNAE - Jump on below/not above or equal</b>	0 1 1 1 0 1 0 1	disp																																						
<b>JBE/JNAB - Jump on below or equal/not above</b>	0 1 1 1 0 1 1 0	disp																																						
<b>JP/JNPE - Jump on parity/parity even</b>	0 1 1 1 1 0 1 0	disp																																						
<b>JO - Jump on overflow</b>	0 1 1 1 0 0 0 0	disp																																						
<b>JS - Jump on sign</b>	0 1 1 1 1 0 0 0	disp																																						
<b>JNE/JNZ - Jump on not equal/not zero</b>	0 1 1 1 0 1 0 1	disp																																						
<b>JNL/JNGE - Jump on not less/greater or equal</b>	0 1 1 1 1 1 0 1	disp																																						
<b>JNLE/JB - Jump on not less or equal/greater</b>	0 1 1 1 1 1 1 1	disp																																						
<b>JNB/JNAE - Jump on not below/above or equal</b>	0 1 1 1 0 1 1 1	disp																																						
<b>JNBE/JAB - Jump on not below or equal/above</b>	0 1 1 1 0 1 1 0	disp																																						
<b>JMP/JPO - Jump on not par/par odd</b>	0 1 1 1 1 0 1 1	disp																																						
<b>JNO - Jump on not overflow</b>	0 1 1 1 0 0 0 1	disp																																						
<b>Processor Control</b>																																								
CLC - Clear carry	1 1 1 1 1 0 0 0																																							
CMC - Complement carry	1 1 1 1 1 0 0 1																																							
STC - Set carry	1 1 1 1 1 0 0 1																																							
CLD - Clear direction	1 1 1 1 1 1 0 0																																							
STD - Set direction	1 1 1 1 1 1 0 1																																							
CLI - Clear interrupt	1 1 1 1 1 0 1 0																																							
STI - Set interrupt	1 1 1 1 1 0 1 1																																							
HLT - Halt	1 1 1 1 1 0 1 0																																							
WAIT - Wait	1 0 0 1 1 0 1 1																																							
ESC - Escape (to external device)	1 1 0 1 1 x x x	mod x x x r/m																																						
LOCK - Bus lock prefix	1 1 1 1 0 0 0 0																																							

**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**  
 0 0 1 reg 1 1 0

REG is assigned according to the following table:

16-Bit (w = 1)	8-Bit (w = 0)	Segment
000	AX	00 AL
001	CX	01 CL
010	DX	02 DL
011	EX	03 BL
100	SP	10 AH
101	BP	11 CH
110	SI	12 DH
111	DI	13 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:(OF):(DF):(IF):(TF):(SF):(ZF):X:(AF):X:(PF):X:(CF)

Mnemonics © Intel, 1978

**Table 4. 8086 Instruction Set Summary**

## 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



## SDK-86

### 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
$V_{CC} 5V \pm 5\%$	3.5A
$V_{TTY} - 12V \pm 10\%$	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

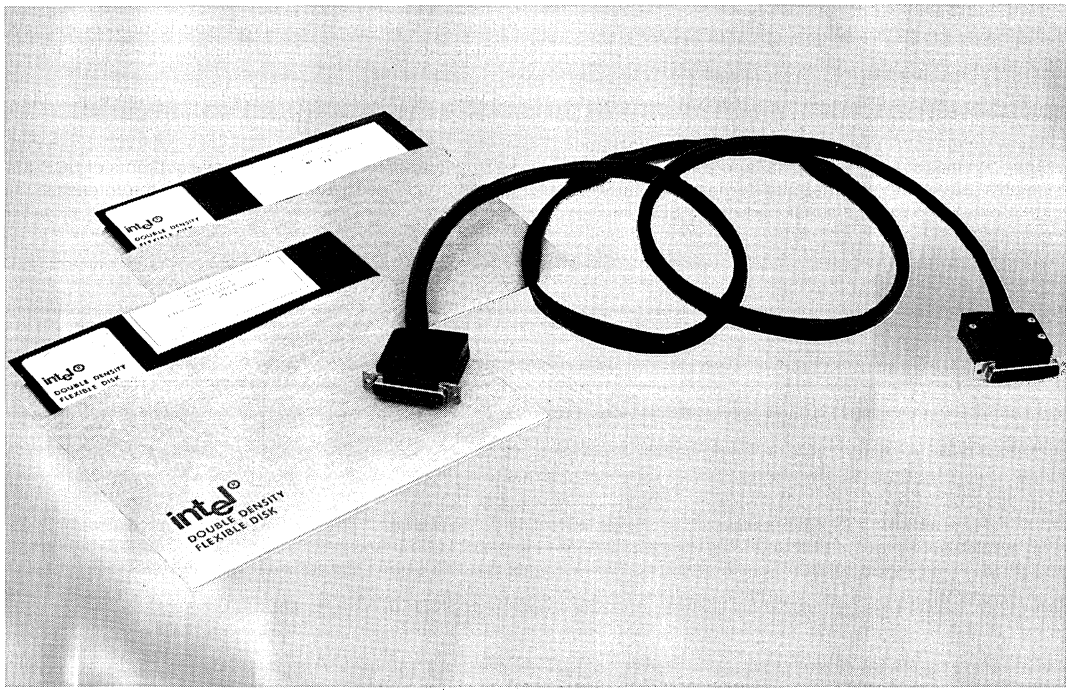
SDK-86 MCS-86 system design kit



## SDK-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.



## SDK-C86

### 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.

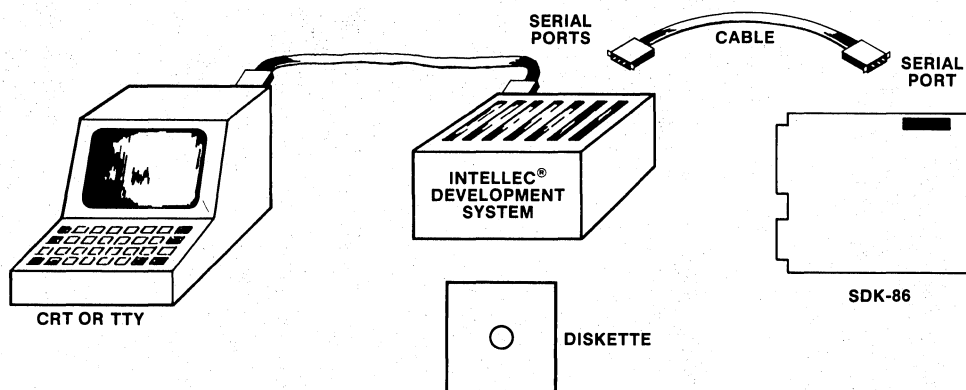
### COMMAND MODES

- **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

---

*MCS-48™ Development  
Systems*

---

**17**

THE UNIVERSITY OF CHICAGO  
DEPARTMENT OF CHEMISTRY  
5800 S. UNIVERSITY AVENUE  
CHICAGO, ILLINOIS 60637

RECEIVED

NOV 15 1964

1964



## MCS-48 DISKETTE-BASED SOFTWARE SUPPORT PACKAGE

**Extends Intellec microcomputer development system to support MCS-48 development**

**Takes advantage of powerful ISIS-II file handling and storage capabilities**

**MCS-48 assembler provides conditional assembly and macro capability**

**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.



# MCS-48™ DISKETTE-BASED SOFTWARE SUPPORT PACKAGE

## 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.

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	FD	22	LP: MOV A,@R0
0108	71	23	ADDC A,@R1
0109	57	24	DA A
010A	A1	25	MOV @R0,A
010B	19	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

Table 1. Sample MCS-48 Diskette-Based Assembly Listing

## 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.



## 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**

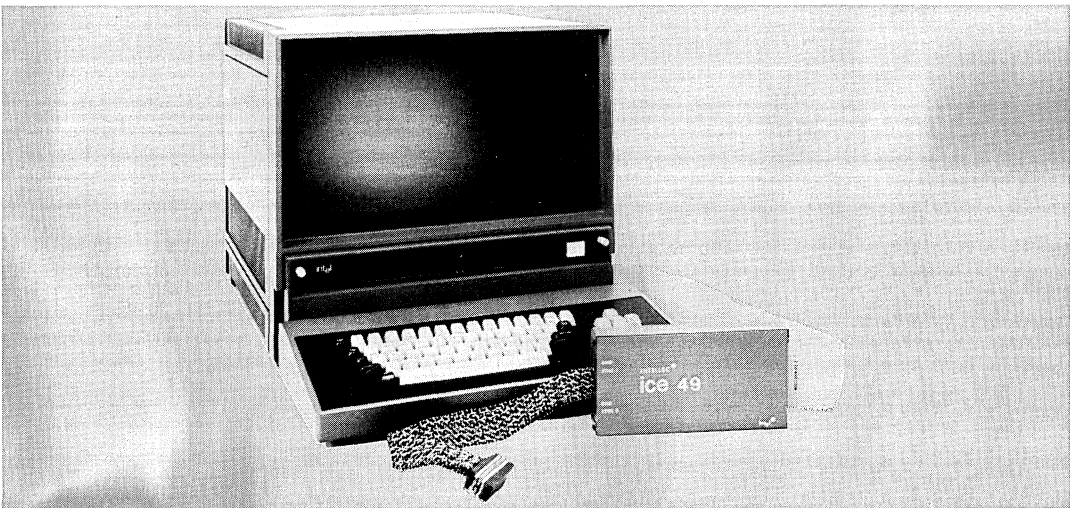
**Collects bus, register, and MCS-48 status information on instructions emulated**

**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 system. 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.

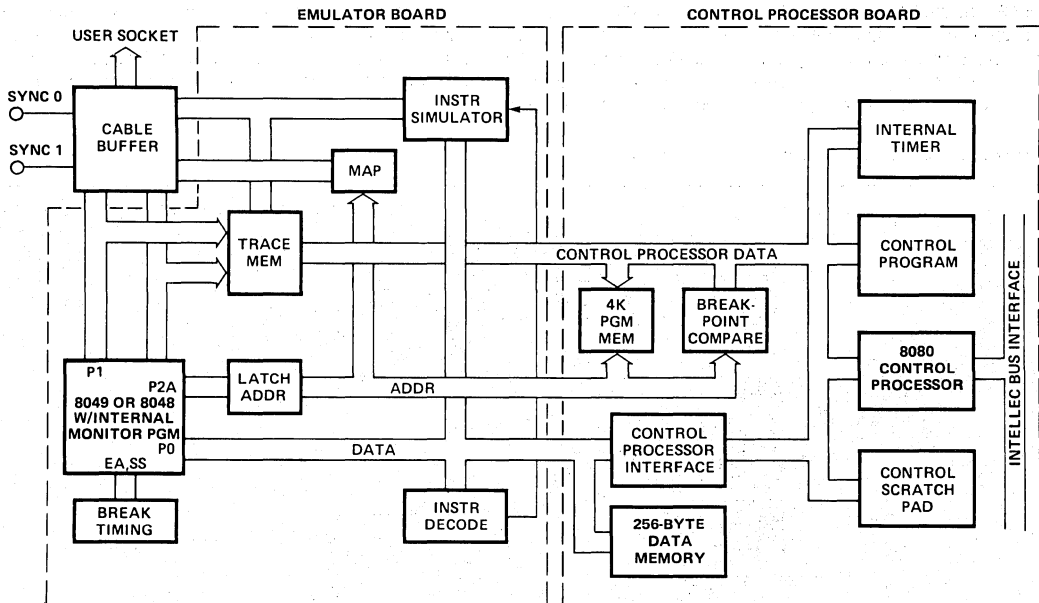


Figure 1. ICE-49 Module Block Diagram

## 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 resettable 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

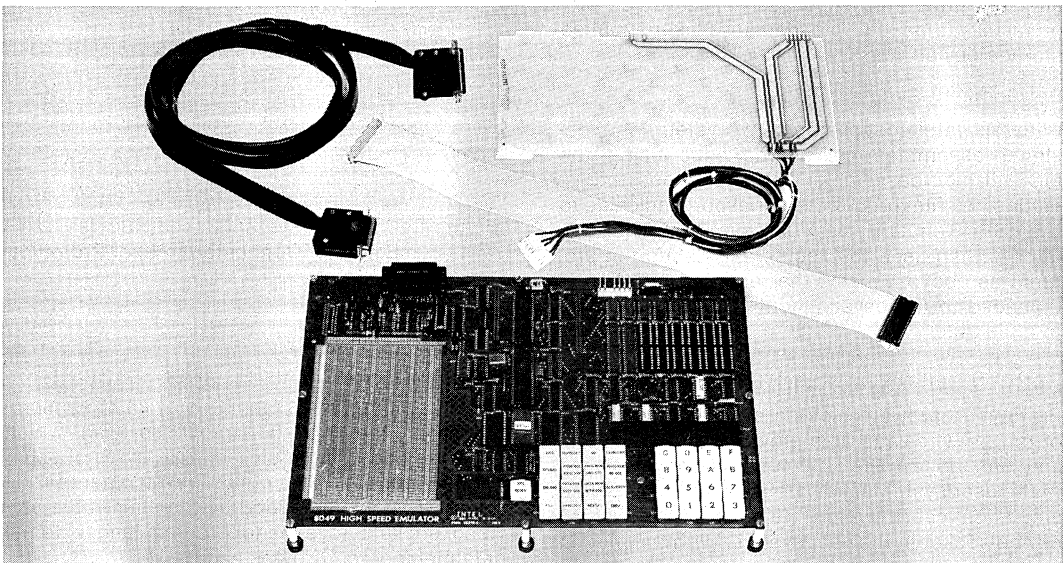
$V_{CC} = +5V \pm 5\%$   
 $I_{CC} = 10A \text{ max}; 7.0A \text{ typ}$   
 $V_{DD} = +12V \pm 5\%$   
 $I_{DD} = 79 \text{ mA max}; 45 \text{ mA typ}$   
 $V_{BB} = -10V \pm 5\%$   
 $I_{BB} = 20 \text{ mA max}$



## 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
- Intellect<sup>®</sup>-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<sup>®</sup>-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 Intellect<sup>®</sup> 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).

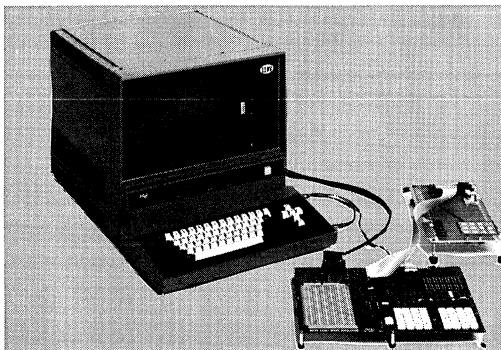


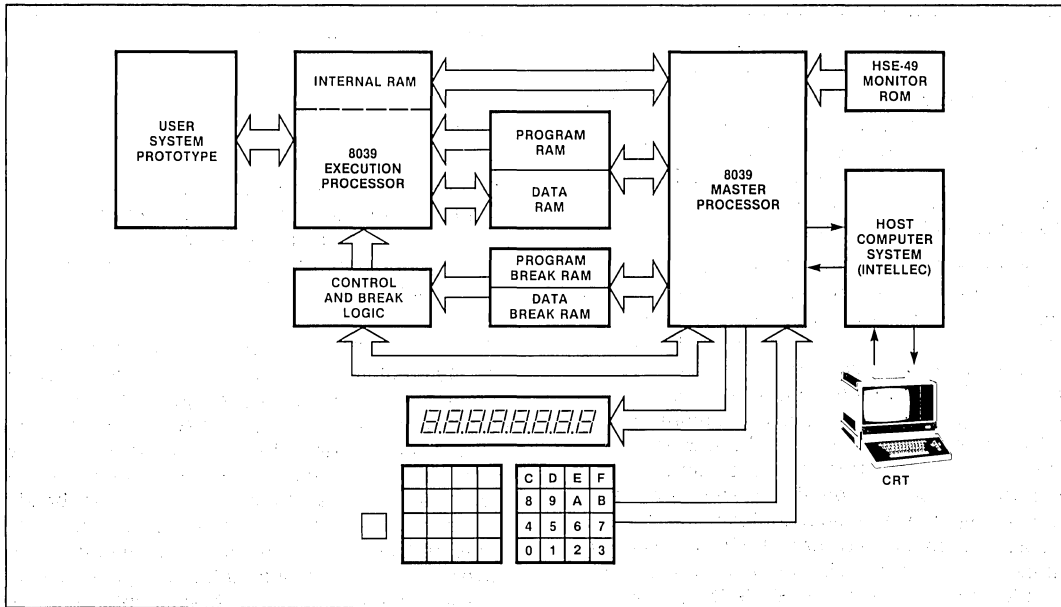
Figure 1. Intellec® System Based HSE-49 Emulator Connected to User Prototype System

## Emulator Overview

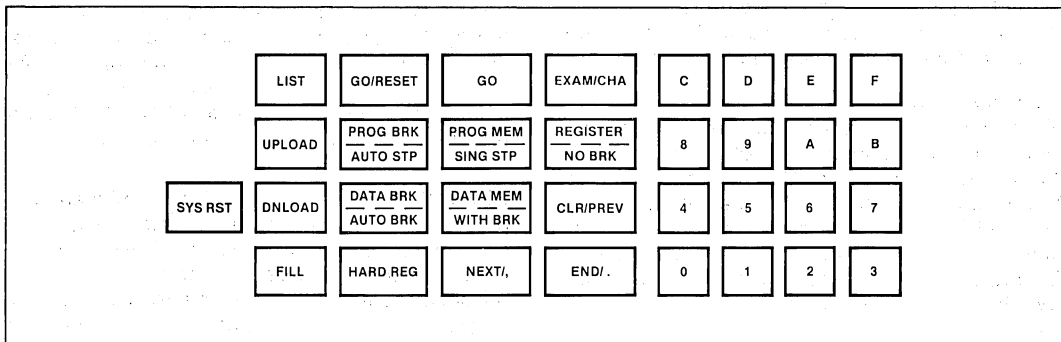
### EMULATION AND MASTER PROCESSORS

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 on-board 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" overlaid 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 overlaid. 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 exe-

cution 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.

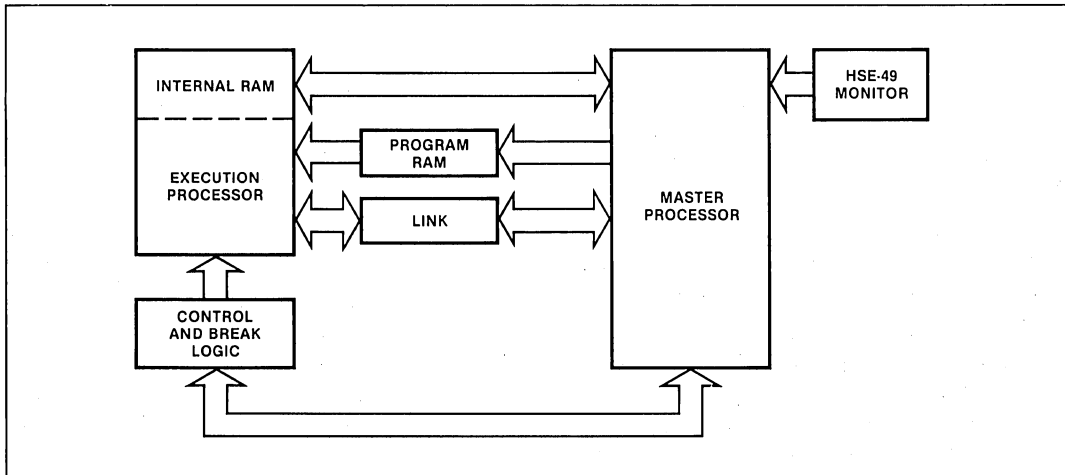


Figure 4. Communication between EP and MP

Table 1. HSE-49 Emulator Command Description

Command	Command Modifier	Description
GO	NO BRK W/BRK SING STP AUTO STP AUTO BRK	Begins emulation: Real-time breakpoints not enabled Real-time breakpoints enabled Steps program one instruction Automatically steps/pauses/steps/... 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 Intel 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 ad-

dress 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<sub>2</sub>-P2<sub>3</sub> 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 inputs, latched outputs or bi-directional 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<sub>CC</sub> = +5V ± 5%

I<sub>CC</sub> = 2.0A max; 1.5A typical

V<sub>RS232</sub> = +12V ± 5%; -12V ± 5%

I<sub>RS232</sub> = 0.020A max; 0.015A typical

(V<sub>RS232</sub> 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
------------	--



## EM1 8021 EMULATION BOARD

**EPROM functional equivalent of 8021 —  
single component 8-bit microcomputer**

**Connects to prototype system through  
8021 pin compatible plug**

**Based on 8748 — user programmable/  
erasable EPROM 8-bit computer**

**On-card 3.0 MHz or external TTL driven  
clock**

**Operates with ICE-49™ module to provide  
full in-circuit debugging of 8021 prototype  
system**

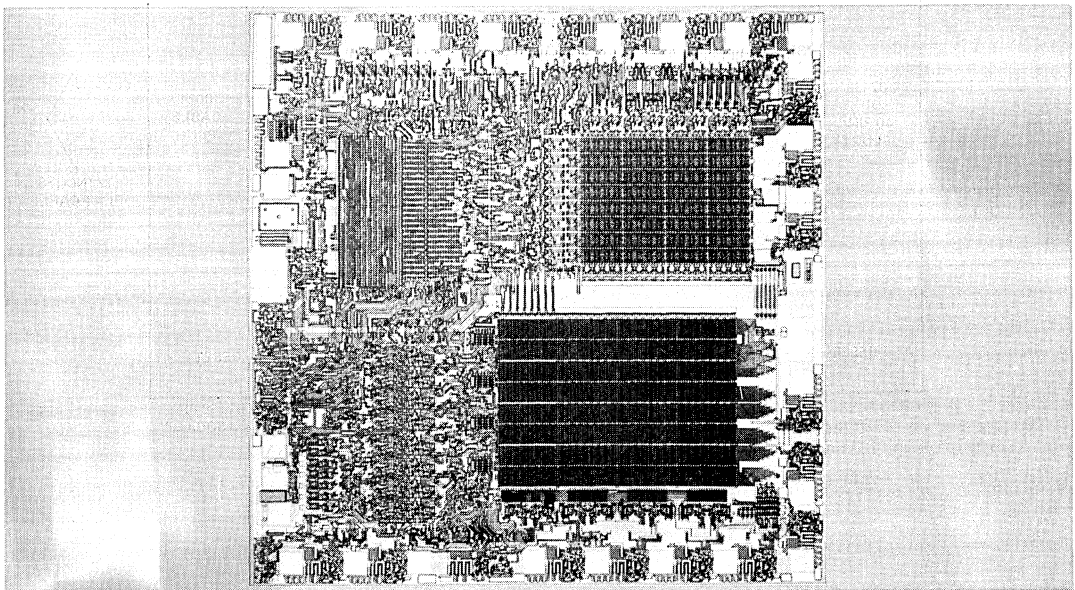
**Portable 4" x 7" microcomputer circuit  
assembly**

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

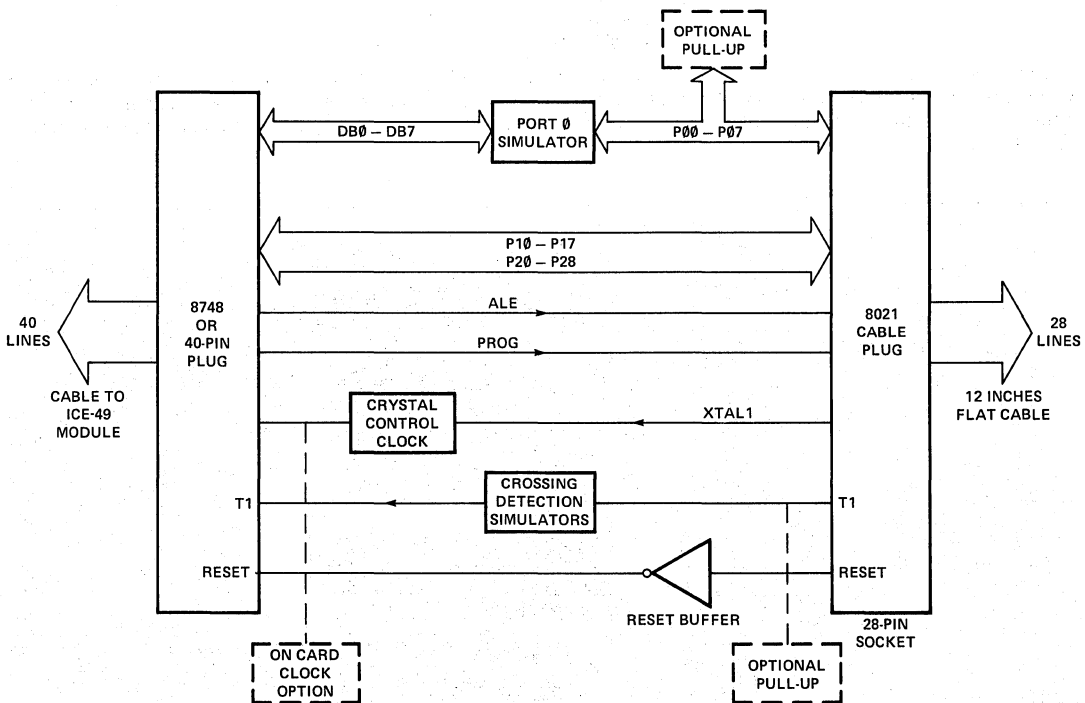
The 8021 resets on a logic HIGH level signal. However, the 8748 resets on a logic LOW level, thus an inverter is provided on the EM1 to make the two chips compatible.

### 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.



# EM1

---

## SPECIFICATIONS

### Operating Environment

#### Stand-Alone

Required Hardware:  
EM1 emulation board

#### In-Circuit Emulation

Required Hardware:  
EM1 emulation board  
Intellec Microcomputer Development System configured 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:  
 $V_{CC} 5V \pm 5\%$   
 $I_{CC} 300 \text{ mA (max.)}$

### 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**

**Provides Intel® 8755A — 2K x 8 EPROM**

**Connects directly into prototype system through Intel® 8022\* pin compatible socket**

**EPROM functional and electrical equivalent of Intel® 8022 — single component 8-bit computer**

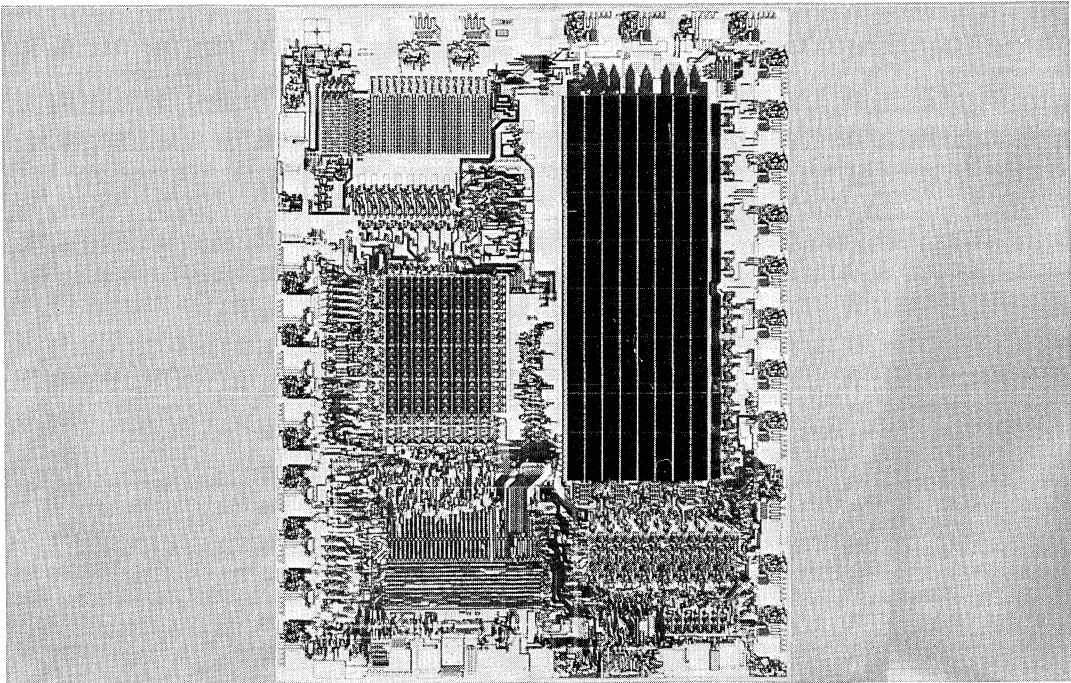
The EM2 emulator board is a ready-to-use 4.25" x 2.75" microcomputer circuit assembly that emulates the Intel® 8022 single chip microcomputer. The emulator board is designed to plug directly into the 8022 socket. No interfacing and interconnection cables are necessary. Power is obtained from the user's 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.

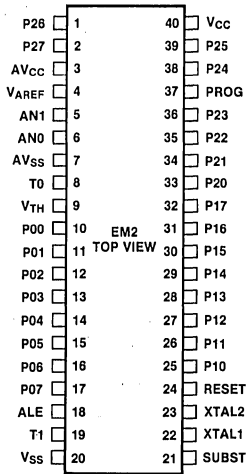
The 8022E emulator chip is a modified version of the 8022 intended for use in design support systems. Instead of using resident ROM memory as the 8022, the 8022E uses an external 2K EPROM 8755A memory for program storage, allowing easy program modification.

\*See Intel® 8022 Data Sheet.



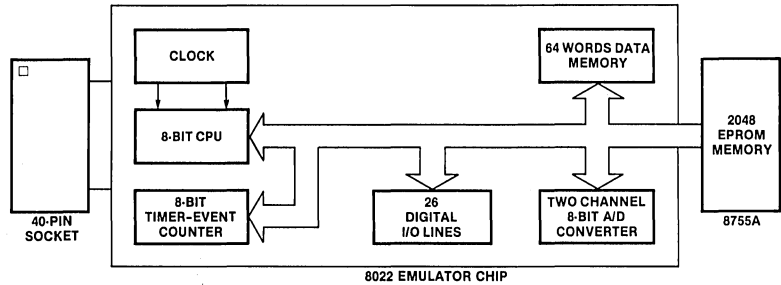
# EM2

## 40-PIN SOCKET CONFIGURATION



□ PIN 1 SQUARE SOLDER PAD

## EM2 BLOCK DIAGRAM



## PIN DESCRIPTION

Designation	Pin #	Function	Designation	Pin #	Function
V <sub>SS</sub>	20	Circuit GND potential.	RESET	24	Input used to initialize the processor by clearing status flip-flops and setting the program counter to zero.
V <sub>CC</sub>	40	+5V circuit power supply.	AV <sub>SS</sub>	7	A/D converter GND potential. Also establishes the lower limit of the conversion range.
PROG	37	Output strobe for Intel® 8243 I/O expander.	AV <sub>CC</sub>	3	A/D +5V power supply.
P00-P07 Port 0	10-17	8-bit open-drain port with comparator inputs. The switching threshold is set externally by V <sub>TH</sub> . Optional pull-up resistors may be added via ROM mask selection. (The emulator board has switch selection of this option.)	SUBST	21	Substrate pin used with a bypass capacitor to stabilize the substrate voltage and improve A/D accuracy.
V <sub>TH</sub>	9	Port 0 threshold reference pin.	VAREF	4	A/D converter reference voltage. Establishes the upper limit of the conversion range.
P10-P17 Port 1	25-32	8-bit quasi-bidirectional port.	AN0, AN1	6,5	Analog inputs to A/D converter. Software selectable on-chip via SEL AN0 and SEL AN1 instructions.
P20-P27 Port 2	33-36 38-39	8-bit quasi-bidirectional port. P20-P23 also serve as a 4-bit I/O expander for Intel® 8243.	ALE	18	Address Latch Enable. Signal occurring once every 30 input input clocks (once every single cycle instruction), used as an output clock.
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.	XTAL1	22	One side of crystal, inductor, or resistor input for internal oscillator. Also input for external frequency source. (Not TTL compatible.)
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.	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 Programming 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 12,000 $\mu$ W/em<sup>2</sup> 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**

$V_{CC} = 5V \pm 5\%$   
 $I_{CC} = 300 \text{ mA (maximum)}$

**Environmental Characteristics**

**Operating Temperature** — 0 to 55°C

**Operating Humidity** — Up to 95% relative humidity without condensation

**ORDERING INFORMATION****Part Number Description**

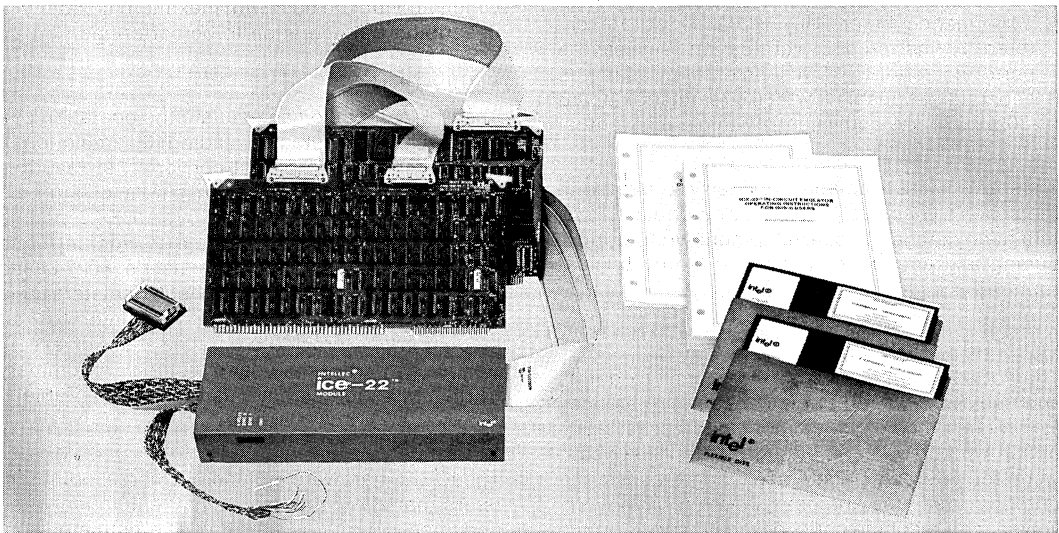
MDS-EM2 8022 Emulation Board



## 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 <, ≤, >, ≥, and ≠ are permitted in addi-

tion to the = condition, 2) more 8022 and ICE variables 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 re-assembly. When assembling new mnemonic instructions into program memory, previously de-

defined 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 Inteltec 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 SY1              ASM REGISTER DEFINE
BR BRO BR1               <MATCH$COND>          CBYTE STACK REMOVE
<BREAK$REG>              <CHANGE>              DBYTE SECONDS GROUP
<MATCH$COND>             Trace Display:        <DISPLAY> <CPUSREF> SYMBOL
                        TRACE                   <ICE$REF>

Step Emulation:          OLDEST
STEP SR                 NEWEST
CR CR0 CR1 CR2 CR3     MOVE
<COMPARISON$REG>       PRINT
<COMPARISON$COND>     DUMP

State/Mode:
BASE ENABLE SYMBOLIC
SUFFIX DISABLE SYNC
LIST RESET

EVALUATE <ADDRESS> <INSTRUCTION> <PRIMARY>
EXIT <ADDRESS$PARTITION> <MASKED$CONSTANT> <PRIMARY$10>
HELP <CHANNEL> <MATCH$GROUP> <STRING>
INERRUPT <CHANNEL$PARTITION> <MATCH$VALUE> <SYMBOLIC$REF>
LOAD <EXPR> <MEMORY$REF> <SYSTEM$GROUP>
SAVE <EXPR$10> <NUMERIC$CONSTANT> <USER$GROUP>
<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> =
    <INSTRUCTION$LIST>
    END
    (Change memory starting at <ADDRESS>)
    Ex: ASM 100 =
        MOV A, @R0
        JNZ 100
        END
(3) ASM <ADDRESS> TO/LENGTH <ADDRESS> =
    <INSTRUCTION$LIST>
    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.)
<INSTRUCTION$LIST> - 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 dou-  
ble 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**

$V_{CC} = +5V, +5\%, -1\%$

$I_{CC} = 13.2A \text{ max}; 11.0A \text{ typical}$

$V_{DD} = +12V, \pm 5\%$

$I_{DD} = 0.1A \text{ max}; 0.05A \text{ typical}$

$V_{BB} = -10V, \pm 5\%$

$I_{BB} = 0.05A \text{ max}; 0.01A \text{ typical}$

**ORDERING INFORMATION**

<b>Part Number</b>	<b>Description</b>
MCI-22-ICE	8022 Microcontroller In-Circuit Emulator, cable assembly and interactive diskette software







## 8051 SOFTWARE DEVELOPMENT PACKAGE

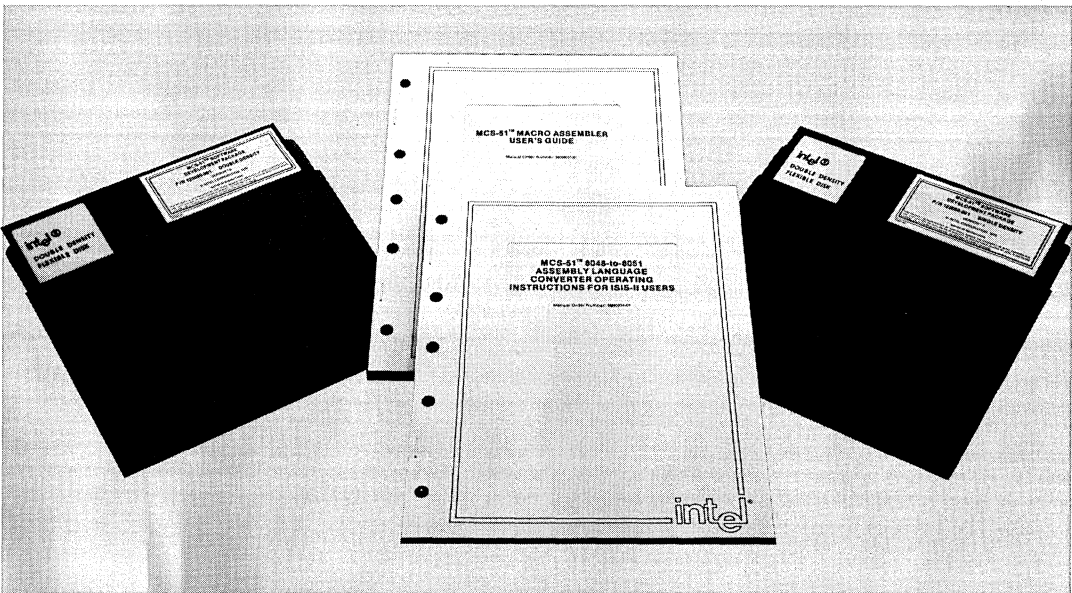
- Symbolic assembly language programming for 8051 microcontrollers
- Extends Intellec® Microcomputer Development System to support 8051 program development
- Provides assembler output in standard Intel hex format
- Macro Assembler features conditional assembly and macro capabilities
- CONV51 Converter for translation of 8048 assembly language source code to 8051 assembly language source code
- Provides upward compability from the MCS-48™ family of single-chip microcontrollers
- Supports conversion of ASM48 source code macro definitions

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 absolute machine code from 8051 macro assembly language instructions. This 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.





# 8051 SOFTWARE DEVELOPMENT PACKAGE

## 8051 MACRO ASSEMBLER

- Supports 8051 family program development on Intellect<sup>®</sup> Microcomputer Development Systems
- Gives symbolic access to powerful 8051 hardware features
- Produces object file, listing file and error diagnostics
- 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 instructions into machine executable object code. These assembly language mnemonics are easier to program and are more readable than binary or hexadecimal machine instructions. Also, 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.

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.

```
HCS-51 MACRO ASSEMBLER                                     PAGE
IS15-11 HCS-51 MACRO ASSEMBLER
OBJECT MODULE PLACED IN :F1:CONVRT.HEX
ASSEMBLER INVOKED BY: :F1:ASM51 :F1:CONVRT.51 SYMBOLS XREF

LOC OBJ          LINE    SOURCE
1
2
3
4 ; This routine converts BCD to binary and binary to BCD.
5 ;
6 ;
7 ;
8
9          ORC 10BH
10         CONVRT: MOV A,R1
11         MOV B,#10      ; Load B for division or multiplication by 10
12         JC BINBCD
13         BCD2BIN: ANL A,#0F0H ; Mask out low order digit
14         SWAP A         ; Move high digit into low order nibble of accumulator
15         ; Multiply digit by 10
16         MUL AB         ; Multiply digit by 10
17         XCH A,R1:      ; Store intermediate result and get new copy of BCD
18         ANL A,#0F0H   ; Mask out high order digit
19         ADD A,R1:      ; Add high order digit and low order digit
20         MOV R1,A      ; Store result
21         RET
22         BINCD: DIV AB  ; Divide by 10
23         SWAP A        ; Put quotient in high order nibble
24         ORL A,B       ; Place remainder in low order nibble
25         MOV R1,A      ; Store result
26         RET
27         END

HCS-51 MACRO ASSEMBLER                                     PAGE 2
XREF SYMBOL TABLE LISTING
-----
NAME    TYPE    VALUE AND REFERENCES
0       M DSEC  00F0H 11 24
BCD2BIN L CSEC  0100H 13#
BINBCD  L CSEC  0100H 12 22#
CONVRT  L CSEC  0100H 10#
ASSEMBLY COMPLETE. NO ERRORS FOUND
```

Sample ASM51 Listing

# 8051 SOFTWARE DEVELOPMENT PACKAGE

---

## 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 designs 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 Macro Assembly Language 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

# THE HISTORY OF THE UNITED STATES

The history of the United States is a story of growth, struggle, and progress. From the first settlers to the present day, the nation has evolved through various challenges and triumphs.

In the early years, the colonies fought for independence from British rule. This led to the American Revolution and the birth of a new nation.

The 19th century was a time of westward expansion and industrialization. The Civil War was a pivotal moment in the nation's history, leading to the abolition of slavery.

The 20th century saw the United States emerge as a global superpower. It played a central role in World War II and the Cold War.

Today, the United States continues to face new challenges and opportunities. It remains a nation of innovation, diversity, and resilience.

The history of the United States is a testament to the power of the American dream and the pursuit of freedom and justice for all.

As we look to the future, we must learn from our past and strive to create a better world for ourselves and for generations to come.

The United States has always been a land of hope and possibility. It is a nation that has the capacity to overcome any obstacle and achieve greatness.

Let us continue to uphold the values of liberty, equality, and justice that are at the heart of our nation's identity.

Together, we can build a brighter future for the United States and for the world.

The history of the United States is a story of courage and determination. It is a story that inspires us to reach for the stars.

Let us embrace our heritage and the challenges ahead with confidence and faith.

The United States is a nation of opportunity. It is a land where anyone can make their dreams a reality.

Let us continue to strive for excellence and progress in all that we do.

The history of the United States is a story of hope and promise. It is a story that gives us the strength to face whatever comes our way.

Let us move forward with courage and conviction, knowing that our future is bright.

The United States is a nation of greatness. It is a land where the impossible becomes possible.

Let us continue to write the story of the United States with pride and honor.

*UPI-41A<sup>TM</sup>*  
*Development Systems*

---

**19**

THE UNIVERSITY OF CHICAGO  
DEPARTMENT OF CHEMISTRY  
5408 S. UNIVERSITY AVENUE  
CHICAGO, ILLINOIS 60637  
TEL: 773-936-3700  
WWW.CHEM.UCHICAGO.EDU

1. Name of the donor: \_\_\_\_\_

2. Address of the donor: \_\_\_\_\_

3. City and State: \_\_\_\_\_

4. Zip Code: \_\_\_\_\_

5. Date of gift: \_\_\_\_\_

6. Description of gift: \_\_\_\_\_

7. Value of gift: \_\_\_\_\_

8. Name of recipient: \_\_\_\_\_

9. Address of recipient: \_\_\_\_\_

10. City and State: \_\_\_\_\_

11. Zip Code: \_\_\_\_\_

12. Date of receipt: \_\_\_\_\_

13. Name of recipient: \_\_\_\_\_

14. Address of recipient: \_\_\_\_\_

15. City and State: \_\_\_\_\_

16. Zip Code: \_\_\_\_\_

17. Date of receipt: \_\_\_\_\_

18. Name of recipient: \_\_\_\_\_

19. Address of recipient: \_\_\_\_\_

20. City and State: \_\_\_\_\_

21. Zip Code: \_\_\_\_\_

22. Date of receipt: \_\_\_\_\_

23. Name of recipient: \_\_\_\_\_

24. Address of recipient: \_\_\_\_\_

25. City and State: \_\_\_\_\_

26. Zip Code: \_\_\_\_\_

27. Date of receipt: \_\_\_\_\_



## 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**

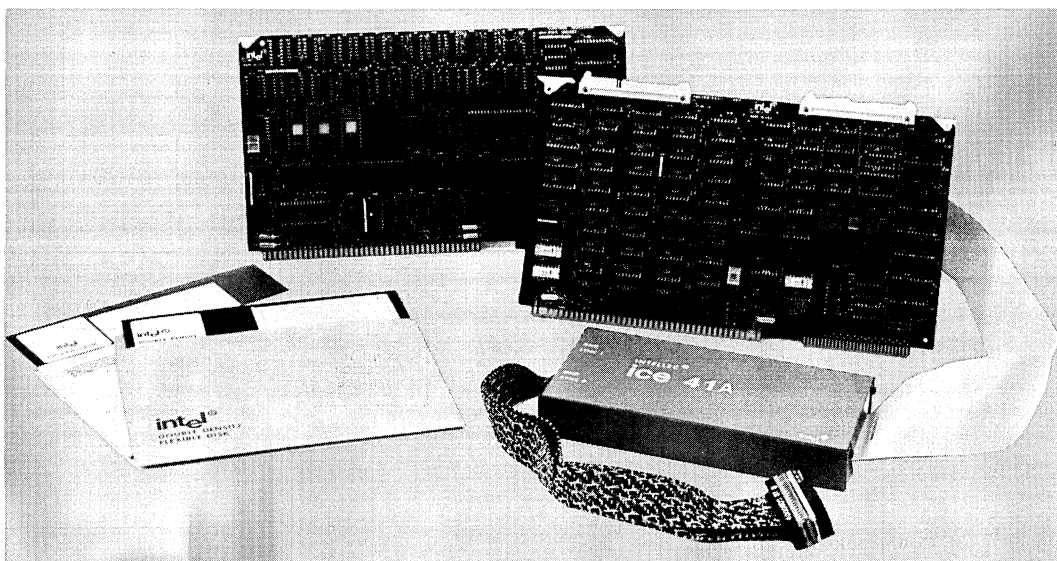
**Eliminates need for extraneous debugging tools residing in user system**

**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**

GO FROM .START TILL CODE. RSLT

**Debug Capability Inside User System**

Inteltec memory is used for the execution of the ICE-41A software. The Inteltec 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 Inteltec 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:

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.

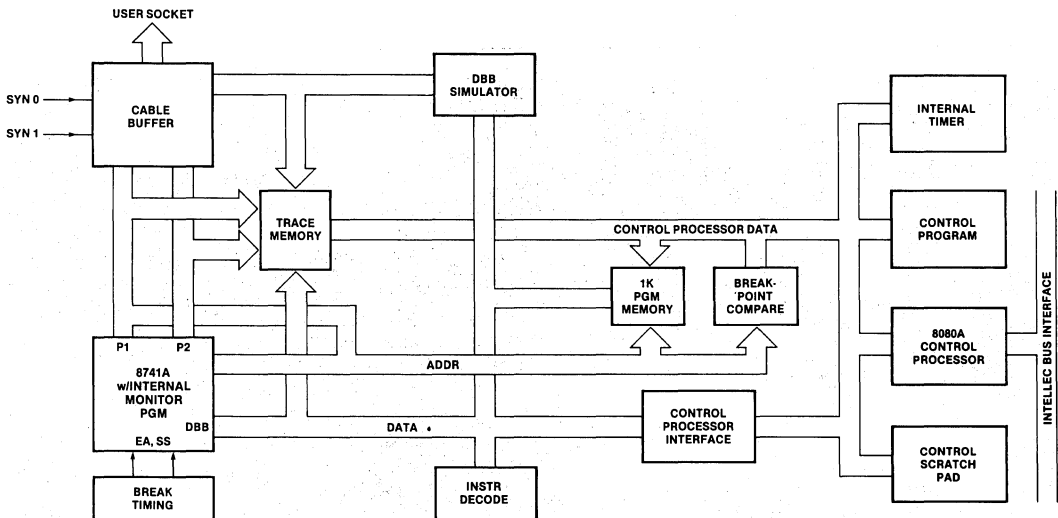


Figure 1. ICE-41A Module Block Diagram

## 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 pro-

vide 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



# ICE-41A™

## SPECIFICATIONS

### ICE-41A Operating Environment

#### Required Hardware

Intel microcomputer development system  
System console  
Intel 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

$V_{CC} = +5V, \pm 5\%$   
 $I_{CC} = 10A \text{ max}; 8A \text{ typ}$

$V_{DD} = +12V, \pm 5\%$   
 $I_{DD} = 100 \text{ mA max}; 60 \text{ mA typ}$   
 $V_{BB} = -10V$   
 $I_{BB} = 30 \text{ mA}$

#### Input Impedance

@ ICE-41A user socket pins:

$V_{IL} = 0.8V \text{ max}; I_{IL} = 1.6 \text{ mA}$   
 $V_{IH} = 2.0V \text{ min}; I_{IH} = 40 \mu A$

@ Bus:

$V_{IL} = 0.8V \text{ max}; I_{IL} = 250 \mu A$   
 $V_{IH} = 2.0V \text{ min}; I_{IH} = 20 \mu A$

#### Output Impedance

@ P1, P2:

$V_{OL} = 0.5V \text{ max}; I_{OL} = 16 \text{ mA}$   
 $V_{OH} = V_{CC} (10K \text{ pullup})$

@ Bus:

$V_{OL} = 0.5V \text{ max}; I_{OL} = 25 \text{ mA}$   
 $V_{OH} = 3.65V \text{ min}; I_{OH} = 1 \text{ mA}$

#### Others

$V_{OL} = 0.5V \text{ max}; I_{OL} = 16 \text{ mA}$   
 $V_{OH} = 2.4V \text{ max}; I_{OH} = 400 \mu A$

### 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



THE UNIVERSITY OF CHICAGO  
DEPARTMENT OF CHEMISTRY  
5301 S. DICKINSON DRIVE  
CHICAGO, ILLINOIS 60637  
TEL: 773-936-3700

NAME: \_\_\_\_\_  
ADDRESS: \_\_\_\_\_  
CITY: \_\_\_\_\_ STATE: \_\_\_\_\_ ZIP: \_\_\_\_\_  
PHONE: \_\_\_\_\_

DATE: \_\_\_\_\_  
COURSE: \_\_\_\_\_  
SECTION: \_\_\_\_\_

PROFESSOR: \_\_\_\_\_  
ASSISTANT: \_\_\_\_\_

STUDENT ID: \_\_\_\_\_

LABORATORY: \_\_\_\_\_

TECHNICIAN: \_\_\_\_\_

ADVISOR: \_\_\_\_\_

RESEARCH ASSISTANT: \_\_\_\_\_

LABORATORY ASSISTANT: \_\_\_\_\_



## 2920 SOFTWARE SUPPORT PACKAGE

- Complete software design development support for the 2920
- Extends Inteltec® Microcomputer Development System to support 2920 software development

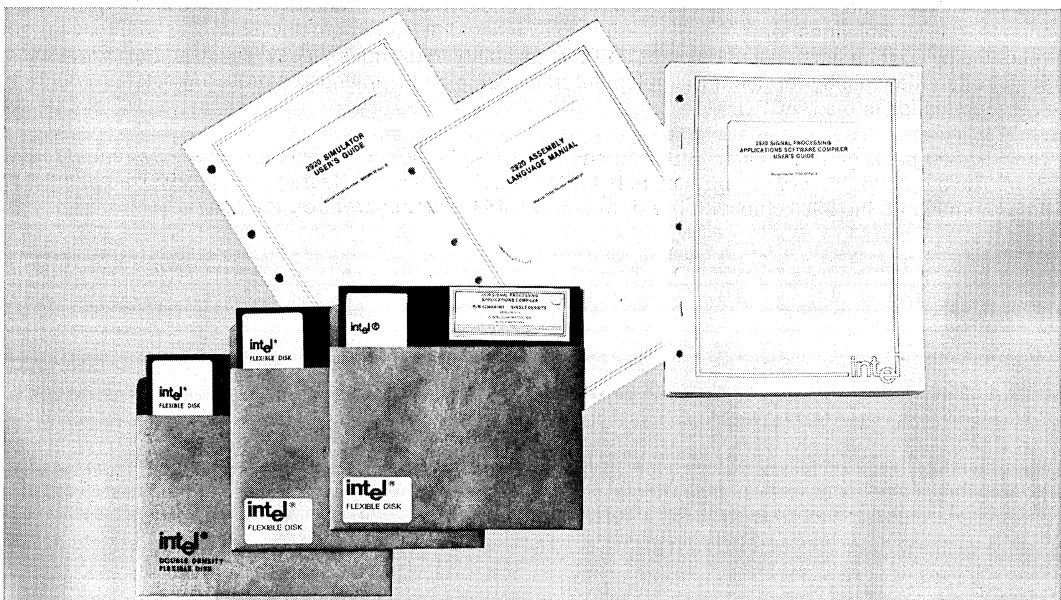
The 2920 Software Support Package furnishes a 2920 Signal Processing Applications Software/Compiler, 2920 Assembler, and 2920 Software Simulator. These three software design and development tools run on the Inteltec® Microcomputer Development System.

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 Inteltec® 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

## 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  $x=TS*FREQ*\pi$ . It corrects group delay by subtracting  $\pi*TS$ .
- 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 algorithms 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 interstitial 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
DB1HZ
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/ZEROS MOVED
*
*P          ; LIST TRANSFORMED POLE
POLE 1 = 0.71092836,0.34118369,2
*
* ; COMPARE RESPONSES OF THE ANALOG AND DIGITAL FILTERS BY GRAPHING THE
* ; NEW RESPONSE OVER THE OLD.
*
*GRAPH GAIN
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
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
PHASE     !.....^.....^.....^.....^.....^.....^.....^.....^.....^.....!
3.14     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
2.84     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
2.54     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
2.24     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
1.94     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
1.65     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
1.35     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
1.05     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
0.75     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
0.45     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
0.15     -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-0.15    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-0.45    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-0.75    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-1.05    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-1.35    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-1.65    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-1.94    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-2.24    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-2.54    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-2.84    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
-3.14    -.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....-.....!
RADIHZ   !.....^.....^.....^.....^.....^.....^.....^.....^.....^.....!
100 150 200 300 400 500 700 1000 1400 2000 3000 5000 10000
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_P1-0.50341796*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 XCUBED(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=4.0000000*XCUBED-0.23437500*TEMP
*
*INST ; THE FUNCTION WAS CODED IN THIS MANY INSTRUCTIONS;
INST = 10,0000000
*

```



# 2920 SOFTWARE SUPPORT PACKAGE

```

*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.
FUNCTION !.....!
0.95
0.91
0.86
0.82
0.77
0.73
0.68
0.64
0.59
0.54
0.50
0.45
0.41
0.36
0.32
0.27
0.23
0.18
0.14
0.09
0.05
0.00
!.....!
*OGRA X**3        ; THE DIFFERENCE BETWEEN THE CODED AND THE ACTUAL APPEARS AS "+".
FUNCTION !.....!
1.00
0.95
0.90
0.86
0.81
0.76
0.71
0.67
0.62
0.57
0.52
0.48
0.43
0.38
0.33
0.29
0.24
0.19
0.14
0.10
0.05
0.00
!.....!
*GRA (X**3)-DATA(X) ; THE ERROR WILL BE GRAPHED.
FUNCTION !.....!
0.047
0.043
0.039
0.036
0.032
0.028
0.025
0.021
0.017
0.014
0.010
0.006
0.003
-0.001
-0.005
-0.008
-0.012
-0.016
-0.020
-0.023
-0.027
-0.031
!.....!
*EXIT      ; THAT'S ALL FOLKS

```



## 2920 ASSEMBLER

- 2920 program development on Intellec® Microcomputer Development Systems
- Produces Assembly Listing, Object Code File, and Error Diagnostics
- Translates symbolic assembly language instructions into 2920 machine code
- 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 7ABAED      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, 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 capacitors 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

Table with 2 columns: Command and Operation. Rows include Display, Change, Base, Suffix, Load, Save, Define, Console, List, Exit, Evaluate, Remove, Help, Graphics On/Off, X Size.

Software Simulator Keyword References

Table with 2 columns: Keyword and Description. Rows include TIME, TQUAL, COUNT, BUFFERSIZE, TINST, SIZE, TPROG, VREF.

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.

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

Table with 2 columns: Keyword and Description. Rows include IN0-OUT7, IN, DAR, PC, CY, OVf, OVE.

Sample 2920 Simulation Session

```
-SM2920.SFT
ISIS-II 2920 SIMULATOR, V1.1
*
*; THIS IS THE SIMULATION OF THE 'SAWTOOTH GENERATOR'
*
*LIST SRG.LOG ; LISTS THE SIMULATION SESSION TO AN ISIS FILE
*LOAD SRG.HEX ; LOAD THE OBJECT CODE INTO THE 2920 SIMULTOR
*ROM 0 TO 5 ; DISPLAY SRG PROGRAM
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,RAH .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
1.00000000000000000000000000000000 0.10100000000000000000000000000000
2.0000000E+0 0.10100001000000000000000000000000
3.0000000E+0 0.00101000010000000000000000000000
SIMULATION TERMINATED
*QUALIFIER=PC=0 ; TRACE EVERY PROGRAM PASS
*TRACE=T,DAR,RAH .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,RAH .OSC,-1,-1,1,1 ; SETS ITEMS TO BE TRACED
*RAH .OSC=ONE      ; INITIALIZE THE RAH LOCATION
*SIMULATE FROM 0
T          0          1          DAR          1          RAH 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

FORTTRAN-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/Simulator Software

# THE HISTORY OF THE

1780

The history of the...  
The first...  
The second...  
The third...  
The fourth...  
The fifth...  
The sixth...  
The seventh...  
The eighth...  
The ninth...  
The tenth...  
The eleventh...  
The twelfth...  
The thirteenth...  
The fourteenth...  
The fifteenth...  
The sixteenth...  
The seventeenth...  
The eighteenth...  
The nineteenth...  
The twentieth...  
The twenty-first...  
The twenty-second...  
The twenty-third...  
The twenty-fourth...  
The twenty-fifth...  
The twenty-sixth...  
The twenty-seventh...  
The twenty-eighth...  
The twenty-ninth...  
The thirtieth...

The thirty-first...  
The thirty-second...  
The thirty-third...  
The thirty-fourth...  
The thirty-fifth...  
The thirty-sixth...  
The thirty-seventh...  
The thirty-eighth...  
The thirty-ninth...  
The fortieth...  
The forty-first...  
The forty-second...  
The forty-third...  
The forty-fourth...  
The forty-fifth...  
The forty-sixth...  
The forty-seventh...  
The forty-eighth...  
The forty-ninth...  
The fiftieth...  
The fifty-first...  
The fifty-second...  
The fifty-third...  
The fifty-fourth...  
The fifty-fifth...  
The fifty-sixth...  
The fifty-seventh...  
The fifty-eighth...  
The fifty-ninth...  
The sixtieth...  
The sixty-first...  
The sixty-second...  
The sixty-third...  
The sixty-fourth...  
The sixty-fifth...  
The sixty-sixth...  
The sixty-seventh...  
The sixty-eighth...  
The sixty-ninth...  
The seventieth...  
The seventy-first...  
The seventy-second...  
The seventy-third...  
The seventy-fourth...  
The seventy-fifth...  
The seventy-sixth...  
The seventy-seventh...  
The seventy-eighth...  
The seventy-ninth...  
The eightieth...  
The eighty-first...  
The eighty-second...  
The eighty-third...  
The eighty-fourth...  
The eighty-fifth...  
The eighty-sixth...  
The eighty-seventh...  
The eighty-eighth...  
The eighty-ninth...  
The ninetieth...  
The ninety-first...  
The ninety-second...  
The ninety-third...  
The ninety-fourth...  
The ninety-fifth...  
The ninety-sixth...  
The ninety-seventh...  
The ninety-eighth...  
The ninety-ninth...  
The hundredth...







# MEMORY SYSTEMS

## INTRODUCTION

Intel's standard line of memory systems includes OEM memory systems for use in general and special purpose applications, and add-on/add-in memory systems and cards for use with DEC\* and DG\*\* minicomputer systems. An overview of Intel's family of general purpose memory systems, the Series 90, is included. A separate Series 90 data catalog is available which provides complete specifications together with ordering information. This section provides information on the in-5770, a video refresh memory system providing 4-bit resolution to the picture elements of video images projected onto raster-scan CRT display terminals, and is designed specifically for image-enhancement applications in conjunction with sophisticated, computer-driven medical, scientific, and laboratory applications such as X-rays. Our standard products cover a wide range of applications, and, because they use proven designs manufactured in volume, provide the most cost-effective solutions for most applications. However, in applications whose specialized requirements cannot be met with standard products, Intel designs and manufactures customized systems in exact conformance with user specifications.

\* DEC and PDP are registered trademarks of Digital Equipment Corporation.

\*\* Nova is a registered trademark of Data General Corporation.



## 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 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 tested to Intel's quality standards for industrial grade products.

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'l, INTEL, INTELLEC, MCS, 'm, ICS, ICE, UPI, BXP, ISBC, ISBX, INSITE, IRMX, CREDIT, RMX/60,  $\mu$ Scope, 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 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 270, 350, and 400 nanosecond cycle times and capacities ranging from 128 kilobytes up to 16 megabytes within a single chassis. The System 91 (advance information) is a lower cost product with a slower cycle time of 650 nanoseconds and capacities up to 16 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 performance ranges of 100, 140, and 250 nanoseconds. Capacities generally are available in one-fourth those of the System 90, but consult the Series 90 Data Catalog for the latest information.

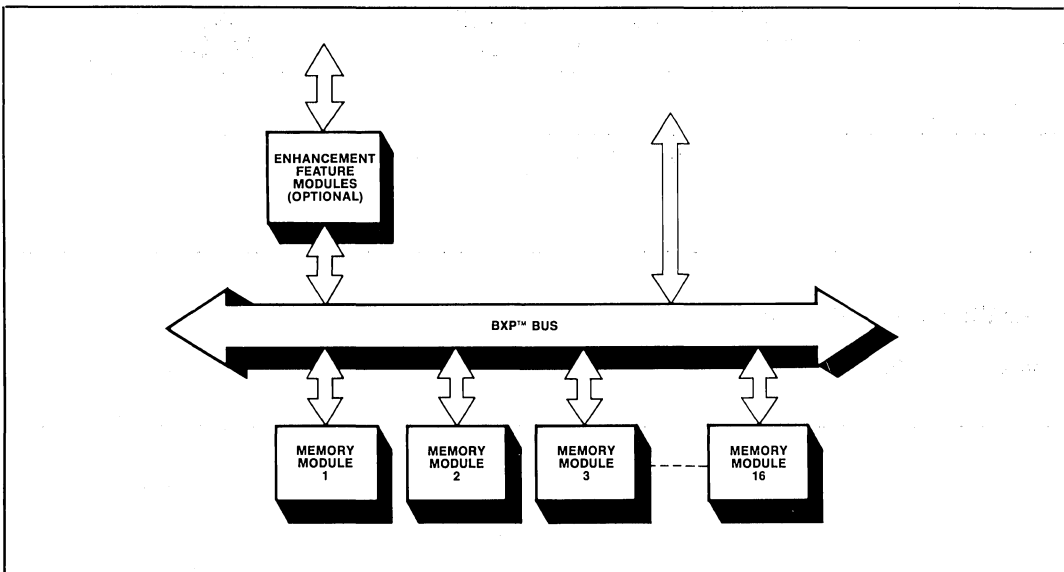
**SYSTEM CONFIGURATION**

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 BXP systems. There are two types of basic features: ECX (error correcting) and IQX (intelligent controller). See Figure 2 for a feature comparison.

**ECX**

This feature (also called Control Interface) 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 (ADVANCED INFORMATION)**

For asynchronous applications that do not require interleaving, this intelligent enhancement offers advanced features such as ECC plus; fault-tolerant (uninterruptible) operation; versatile maintenance tools; self-test and automatic memory diagnostics; remote error reporting and fault-isolation; off-loading of memory management overhead; and simple message-driven interface protocol.

SERIES 90 PRODUCT FAMILY			
		ENHANCEMENT FEATURES	
BXP™ SYSTEMS		ECX	IQX (ADV. INFO.)
SYSTEM 90	275, 350, 400 ns cycle; synchronous or asynchronous; interleaved	275, 350, 400 ns cycle; synchronous or asynchronous; ECC; interleaved	425, 500, or 550 ns cycle; asynchronous; ECC plus; fault-tolerant
SYSTEM 91 (ADVANCED INFO.)	Not Available	500 or 650 ns cycle; synchronous or asynchronous; ECC	650 or 800 ns cycle; asynchronous; ECC plus; fault-tolerant
SYSTEM 92	100, 140, 250 ns cycle; synchronous or asynchronous; interleaved	100, 140, 250 ns cycle; synchronous or asynchronous; ECC; interleaved	250, 290, or 400 ns cycle; asynchronous; ECC plus; fault-tolerant

**Figure 2. Feature Comparison**

**ORDERING INFORMATION**

Consult Series 90 Data Catalog.



## in-5770 VIDEO REFRESH MEMORY SYSTEM

**Single card memory system complete with addressing and timing logic**

**Utilizes MOS dynamic RAMs for maximum bit density at low system cost**

**256K-word  $\times$  4-bit capacity in four image planes, each 256K  $\times$  1 bit wide**

**Operates in both parallel and single bit mode control**

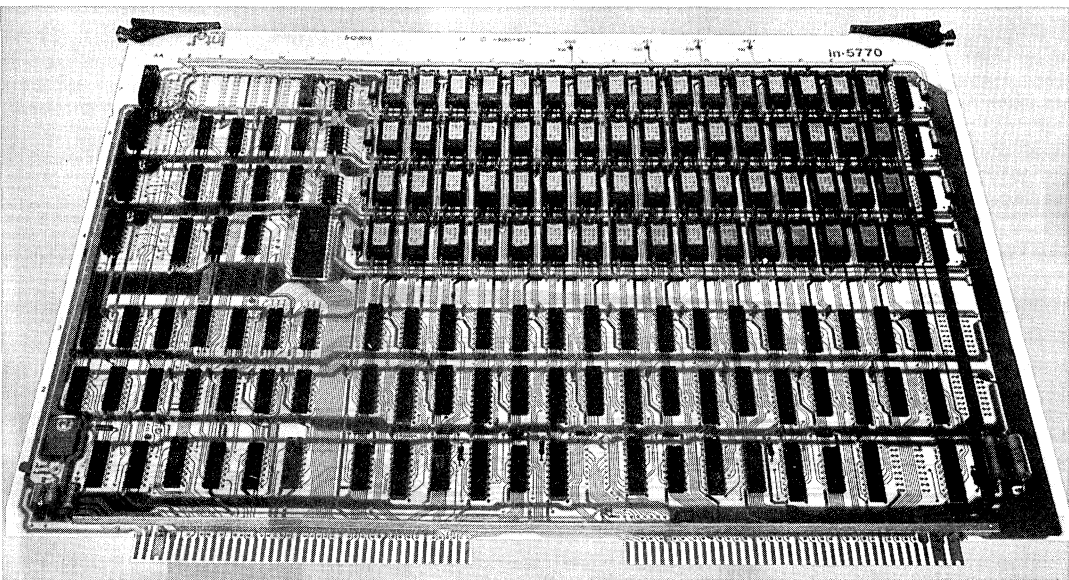
**Four-bit resolution, allowing 16 shades of grey for image enhancement**

**Provides automatic refresh or externally initiated refresh**

**Expandable in multiple card systems for greater resolution in both grey scale and multicolor displays**

**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  $\times$  1 MOS dynamic RAM technology for maximum bit density at low system cost. An optional chassis is available for multiple card housing.



Intel, Inteltec, INSITE, Library Manager, MCS, MEGACHASSIS, MICROAMP, PROMPT,  $\mu$ SCOPE, MULTIBUS, RMX/80, UPI-41, ICE, and iSBC are trademarks of Intel Corporation.

## 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 stor-

age 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 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.

### Chassis Option

To accommodate a wide range of memory capacities, Intel offers a number of chassis options, available as standard equipment. The in-5770 system, whose outer dimensions and card outline are shown in Figure 2, may be mounted in the optional VC-5770 vertical chassis.

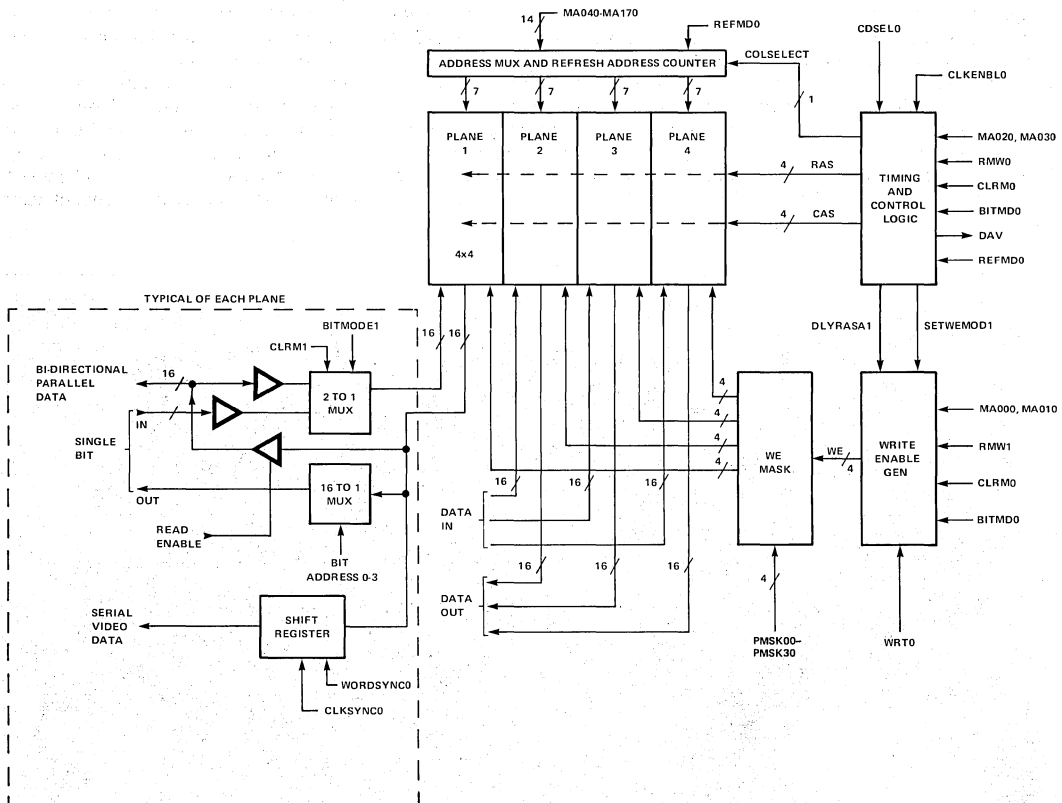


Figure 1. in-5770 Video Refresh Memory System Block Diagram

## in-5770

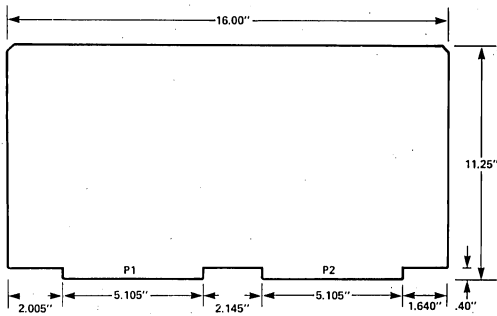


Figure 2. in-5770 Card Dimensions

**Vertical CRT Chassis** — The VC-5770 (as shown in Figure 3) is a 24-slot chassis used for a combination of in-5770 CRT refresh memories and custom control interface card mounted vertically. The UT-5770 utility card may also be used with the in-5770 to provide custom logic. Power supplies and cooling fans are not included.

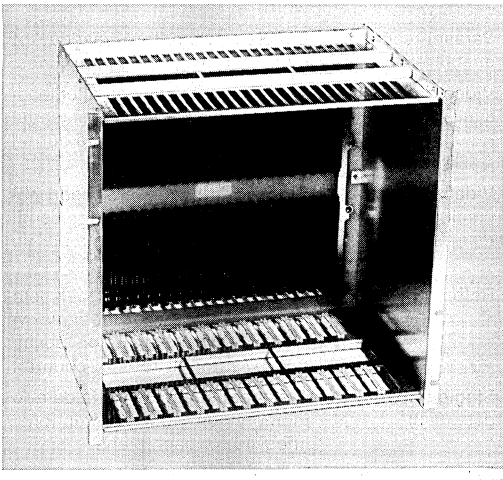


Figure 3. VC-5770 Vertical CRT Chassis Unit

### 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.

**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.

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.

Table 3. in-5770 Output Signal Operations

Signal	Operation	Signal	Operation
Addresses (0-17)	Controls address selection of both single bit data and parallel data on 18 bidirectional lines.	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.
Card select	Enables memory activity on card when low.	Parallel read enable	Enables, when low, readout of data onto 64 bidirectional lines.
Memory cycle start	Initiates memory cycle upon going from high to low.	Video data load sync	Enables, when low, parallel loading of shift registers for serial transfer out of video data during serial read operations.
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.	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.
Write	Initiates write operation when held low at beginning of memory cycle.	Refresh mode	Initiates refresh cycle, when held low at beginning of memory cycle, on address location specified by internal refresh counter.
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.	Single bit write data (0-3)	Transfers single bit write data to memory on four bidirectional lines, 16 for each plane.
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.	Parallel memory data (0-15)	Transfers parallel data to and from memory on 64 bidirectional lines, 16 for each plane.

Table 2. in-5770 Input Signal Operations

**SPECIFICATIONS**

**Storage Capacity**

256K x 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  $\mu$ A

## Output

Low: -0.5V to +0.5V @ 15 mA  
High: +2.4V to +5.2V @ 200  $\mu$ A

## 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)

### Chassis

Width: 19 in. (48.26 cm)  
Height: 17.5 in. (44.45 cm)  
Depth: 11.5 in. (29.21 cm)

## Electrical Characteristics

### DC Power Requirements

Voltage ( $\pm 5\%$ )	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

## Optional Accessories

**IC-10** — 96-pin interface connector

**EX-5770** — Extender card

**UT-5770** — Utility card

**BA-40** — Blower assembly

**VC-5770** — Vertical mount chassis/cardcage with un-wired backplane to accommodate up to 24 CM-5770 cards.

## Reference Manuals

**TM-5770-000** — in-5770 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**

<b>Model</b>	<b>Capacity</b>	<b>Description</b>
CM-5770-512	256K × 4	512 × 512 × 4 video refresh memory system card
CM-5770-256	128K × 4	256 × 512 × 4 video refresh memory system card
VC-5770	—	Chassis with universal backplane to accommodate up to 24 CM-5770 cards
UT-5770	—	Utility card
EX-5770	—	Extender card
BA-40	—	Blower assembly, rack mount, 5.25 in. × 19 in.



## in-1670 PDP\*-11/70 ADD-ON MEMORY SYSTEM

**Total hardware and software compatibility with PDP-11/70**

**128K-bytes storage capacity expandable to 1024K bytes in 128K-byte increments**

**User access to full PDP-11/70 address space at 3,932,160 bytes**

**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**

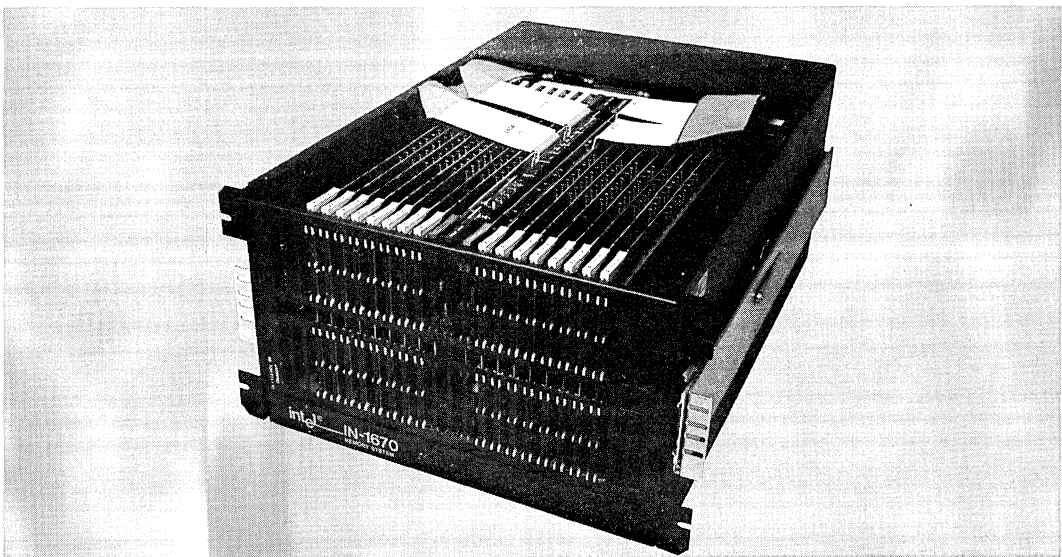
**High speed CPU data transfer rate**

- cycle time: 750 ns
- access time: 555 ns

**Installation and maintenance available from Intel**

**UL recognition**

The Intel in-1670 PDP-11/70 Add-On Memory System is a monolithic memory system offering up to one megabyte 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-1670 system utilizes four types of printed circuit cards: memory unit cards, control unit cards, data cards, and error logging cards. 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-1670 is used in PDP-11/70 computer systems as a direct replacement for the memory module (MK-11) 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 128K-byte increments to 1024K bytes. High speed 750-nanosecond read and write cycle times allow the maximum utilization of Unibus data throughput. The in-1670 system includes error coding and correction (ECC), error monitoring, and error logging.



Intel, Intellec, INSITE, Library Manager, MCS, MEGACHASSIS, MICROAMP, PROMPT,  $\mu$ SCOPE, MULTIBUS, RMX/80, UPI-41, ICE, and iSBC are trademarks of Intel Corporation.

\*PDP and DEC are registered trademarks of the Digital Equipment Corporation.

© Intel Corporation, 1978

## FUNCTIONAL DESCRIPTION

The in-1670 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 four types of printed circuit cards: 1) control cards containing address and control circuitry; 2) data cards to provide interfacing and parity checking for read and write data; 3) error logger cards to monitor and log data error conditions; and 4) memory cards (MU-167As) for storage. 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-1670 memory system is shown in Figure 1.

### Compatibility

The in-1670 is specifically designed for use in PDP-11/70 computer systems as a direct replacement for the DEC-supplied memory module (MK-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-1670 memory system has a capacity of 64K bytes. Two memory cards make up the basic 128K-byte storage in increments of the in-1670 memory system. A fully expanded memory system contains 1024K bytes. Interleaving is possible between two memory systems with the same capacity.

**System Capacity** — The storage area of the in-1670 system consists of 16 Intel MU-167A memory system units. Each MU-167A contains 80 Intel 2109 8K dynamic MOS RAMs to provide a capacity of 32K words by 20 bits per card. The 16 MU-167A's in the system are configured to provide 256K words of storage with a double-word 40-bit interface. Thirty-two of the bits are assigned to data and eight of the bits to error correction and coding (ECC). These systems are housed in a chassis along with power supplies and fan assemblies. Up to four 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.

### System Components

The in-1670 consists of four basic printed circuit cards for memory storage, control, data transfer, and error logging, all housed in a system chassis with power sup-

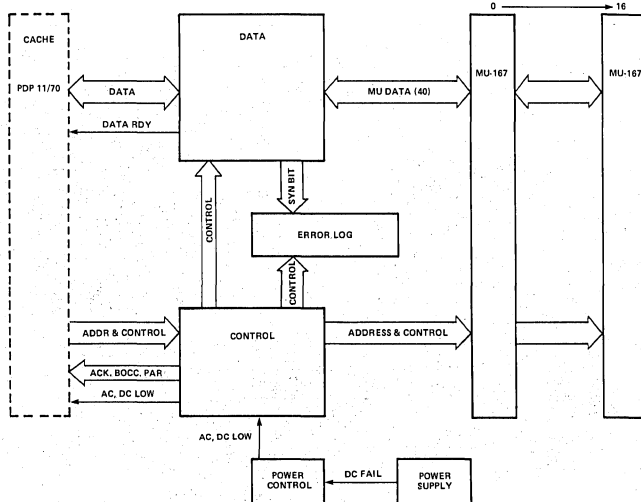


Figure 1. in-1670 Add-On Memory System Block Diagram

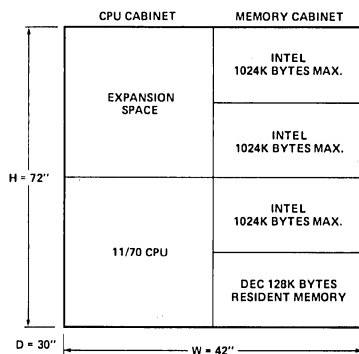


Figure 2. PDP-11/70 CPU Cabinet and In-1670 Add-On Memory Cabinet Dimensions

plies and fan assemblies. Up to four chassis may be mounted in one equipment rack and interfaced in daisy-chain fashion with the processor.

### Maintainability

The in-1670 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

- Memory on-line/off-line switch to disconnect the in-1670 from the PDP-11/70 memory bus
- Reset logic switch
- Error logger on/off switch
- Lamp test
- ECC on/off switch
- Error logger scan switch

**Card Indicators** — The following indicators are provided either on the control card or on the error logger card:

- Address parity error indicator
- Mismatch error indicator
- Write data parity error indicator
- Address display
- Syndrome bits display
- Single bit mode or double bit mode indicator
- On-line/off-line mode indicator

### Reliability

**Error Correction Coding** — The Intel in-1670 memory system includes error correction coding (ECC) as a standard feature. The ECC logic detects single bit memory errors and automatically corrects a single bit failure prior to read operation. Since most memory errors are due to single bit failures, ECC provides a 10 to 25 times improvement in memory system reliability over systems with parity checking only. During write operations, ECC logic removes parity bits from the input data word and generates error correction code bits to be stored with the data word. When write data is received from the processor, it is checked for correct parity before being written into memory. During a read operation, the word is checked for correct status. The ECC circuitry automatically corrects the incorrect bit should a single bit be detected. A double bit error in the read word is not corrected. For double bit errors, all parity bits sent to the CPU are forced to the error state. This allows the CPU to process double bit errors as parity errors. The ECC logger records the faulty location address.

## SPECIFICATIONS

### Storage Capacity

Up to 1024K bytes in 128K-byte increments

### Word Length

16 bits per memory word, plus 8 check bits per double word

### Performance

**Cycle Time** — 750 ns max

**Access Time** — 555 ns max

### Operational Modes

Read  
Write  
Partial 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-1670 memory card

**Notes**

1. Installation can be done by the customer or purchased from Intel.
2. Maintenance contracts are available from Intel.

**Physical Characteristics**

**Width** — 10.5 in. (26.3 cm)  
**Height** — 10.5 in. (26.3 cm)  
**Depth** — 25.0 in. (77.5 cm)  
**Weight** — Less than 70 lb (54 kg)

**Electrical Characteristics**

**AC Power Requirements**

Requirement	115V AC (max)	230V AC (max)
Voltage	90V AC to 140V AC	170V AC to 260V AC
Frequency	45 to 440 Hz single phase	45 to 440 Hz single phase
Input current	6.5A max	3.3A 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 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**

Model	Capacity	Description
SY-1670-128	128K bytes (64K words)	Add-on memory system for PDP-11/70
SY-1670-256	256K bytes (128K words)	Same as above except capacity
SY-1670-384	384K bytes (192K words)	Same as above except capacity
SY-1670-512	512K bytes (256K words)	Same as above except capacity
SY-1670-640	640K bytes (320K words)	Same as above except capacity
SY-1670-768	768K bytes (384 words)	Same as above except capacity
SY-1670-896	896K bytes (448K words)	Same as above except capacity
SY-1670-1MB	1024K bytes (512K words)	Same as above except capacity
XF-1670-128	128K bytes	Expansion for in-1670 PDP-11 Add-On Memory System
EX-1670	—	Card extender
XX-1670-001	—	Memory card
XX-1670-002	—	Data interface card
XX-1670-003	—	Control card
XX-1670-004	—	Error logger card
XX-1670-005	—	Terminator board
XX-1670-006	—	Power control card
XX-1670-007	—	I/O cable assembly
XX-1670-008	—	Power supply
XX-1670-009	—	Diagnostic memory test. Includes magnetic tape.
XX-1670-010	—	Branch spare kit containing one each EX-1670 and XX-1670-001 through XX-1670-009



## in-5150 ECLIPSE\* ADD-IN MEMORY

**High speed — 200 ns from cache memory**

**High reliability**

**Compatible with core memories**

**On-board cache memory**

**ERCC single bit error correction**

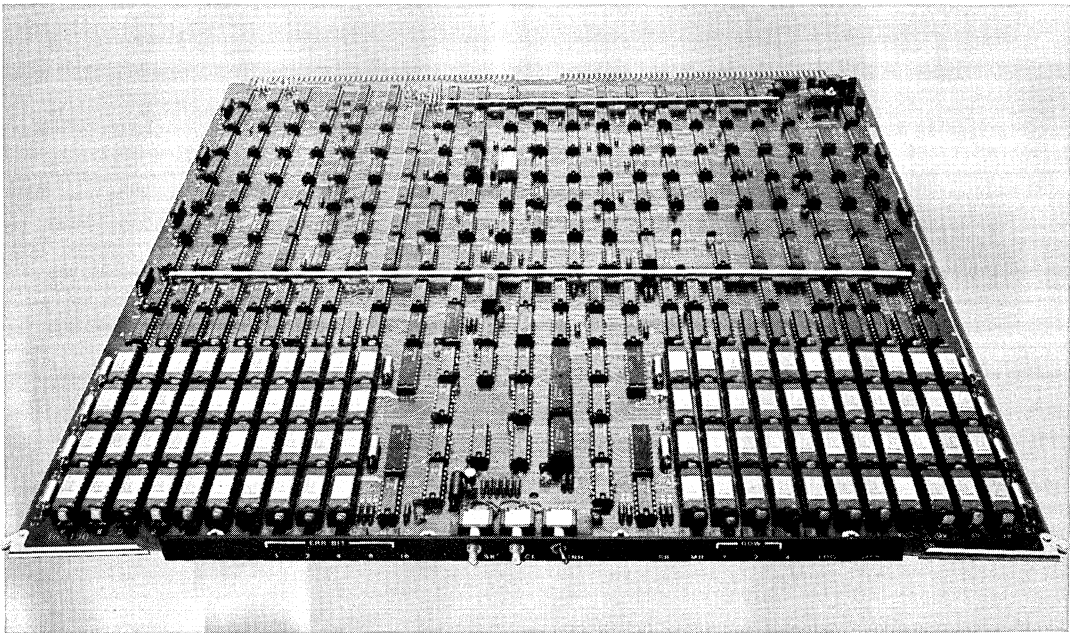
**On-line error logging**

**On-board address select DIP switches**

**Two, four, or eight-way interleaving**

**One year warranty**

The Intel in-5150 is a 32K × 16 or 64K × 16 plug-in memory card, designed for use in any Data General Eclipse\* S200, S230, or S250 series computers. It is totally compatible with all Eclipse models. No modifications to the computer or memory module are required for installation. All in-5150 circuitry, memory, address, control, error correction, error logging, and voltage regulator circuits are contained on one 15-inch × 15 inch printed-circuit card, which can be inserted into one card slot. The in-5150 is designed for low-power dissipation. When operating at full speed in an Eclipse, the in-5150 dissipates 65 watts. In standby mode the power dissipation is 50 watts. The power dissipation is calculated at worst case maximum. These power levels minimize heat dissipation and cooling requirements. In case of power failure, the contents of the memory will be retained, provided there are battery backups on +5V, +15V, and -15V. The in-5150 provides error correction and error logging on the board. Single bit errors are logged and corrected. Most multiple bit errors are detected and flagged as parity errors for the processor. The error correction circuitry can be disabled by a switch on the in-5150's control panel. The logging circuits function with the error correction feature enabled or disabled.





## FUNCTIONAL DESCRIPTION

The in-5150 Eclipse Add-In Memory is a semiconductor memory specifically designed as an add-in memory for Data General Eclipse S200, S230, S250, or CS60 series computers. It is compatible with either Data General core or semiconductor memories. The memory card contains memory storage, memory interface and control, address select cache memory, error correction and logging, and voltage regulator circuits. On-line operating sequences are controlled by processor supplied control signals and by internally generated response signals sent to the processor from the memory. A simplified block diagram of the in-5150 is shown in Figure 1.

### Capacity

The in-5150 is manufactured in two capacities (32K or 64K words) and utilizes either 8K or 16K dynamic RAMs. The 8K and 16K devices must pass the same quality control testing before being used on the in-5150.

### Interleaving

Two, four, or eight-way interleaving is possible. Interleaving increases program execution speed as well as increasing the effective speed of non-Intel memory boards. In addition, cache memory capability is expanded, since interleaving will increase the effective size of cache by the number of in-5150s being interleaved.

### Compatibility

The in-5150 can be inserted into any Eclipse S200, S230, S250, or CS60 series computers, together with any

any appropriate memory card, core, or semiconductor having a capacity of 8K or a multiple of 8K words. Thus, a 32K Intel in-5150 takes up the following addresses:

- 0-32K when used alone
- 8K-40K when used with one 8K memory
- 16K-48K when used with two 8K memories or one 16K memory
- 24K-56K when used with three 8K memories or with one 8K and one 16K memory

The list of combinations shown above is a partial one — there are many more possible combinations.

### Addressability

The in-5150 memory card address circuitry consists of address select logic and the address register. The address range for each memory card is set by dual-in-line package (DIP) switches in the address select logic. If the five highest order bits of the processor supplied address fall within the range set by the switches, the memory card is selected. The address register receives and decodes the 21 processor address bits and generates the row and column address bits for the memory chips in the main memory. Because each in-5150 generates its own control signals, each card operates asynchronously. The processor can thus operate several cards at the same time by interleaving. There are jumper provisions on the card to allow for no interleaving and for two, four, or eight-way interleaving. To achieve interleaving, the memory select logic swaps low-order bits with high-order bits. Swapping the bits does not change the way the bits are produced, only how they are interpreted by the address registers on the individual memory boards.

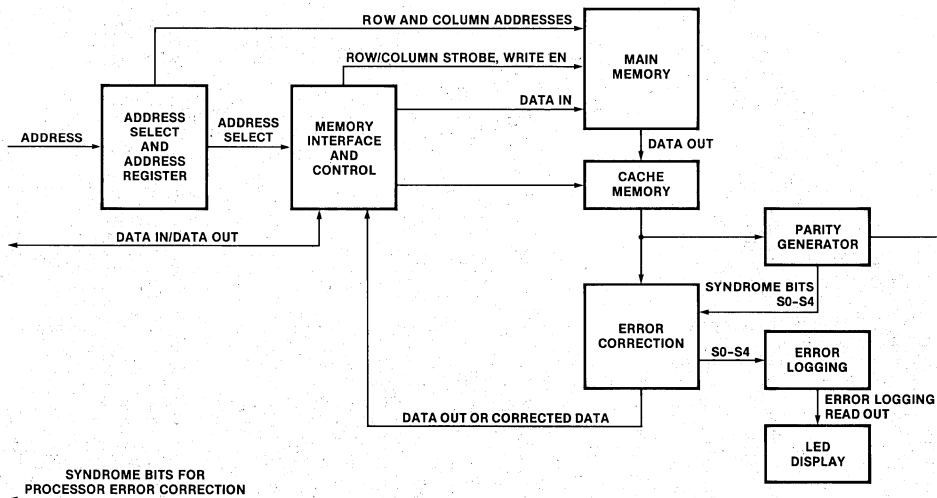


Figure 1. in-5150 Add-In Memory Simplified Block Diagram

## Operating Modes

The in-5150 emulates all of the operating modes supported by Data General on the Eclipse series. The in-5150 has three operating modes: read, write, and read-modify-write. In addition, a fourth mode called refresh is performed every 16 microseconds and lasts for 500 nanoseconds.

**Read cycle** — During a read cycle, data stored in the memory is read out from the memory location specified by the input address and placed on a bidirectional interface data bus. The data is valid 350 nanoseconds after the cycle is initiated and remains stable until it is strobed into the processor.

**Write cycle** — During a write cycle, data sent to the memory is written into the memory location specified by the address inputs. Five extra bits are also written into memory as ECC check bits.

**Read-modify-write cycle** — In a read-modify-write cycle, data is read out of the memory from the location specified by the address inputs. After the read operation is complete, the memory remains busy and writes new data, sent by the processor, into the memory location from which data was read out.

**Refresh cycle** — Every 16 microseconds the in-5150 'steals' a memory cycle to to rewrite (refresh) a portion of the memory cells. In a 2-millisecond period all memory cells are refreshed. The memory is in a busy state during refresh and accepts no access requests until the refresh cycle is finished.

## Cache Memory

A cache memory is provided in the in-5150 to allow faster accessing of data. The cycle time and access time of the cache memory is 200 nanoseconds, provided a 'hit' is made (data is found in cache). The chance that the data being sought by the processor is in cache memory is approximately 60% for a normal application program with the best case hit ratio being 75%. The cache memory is a four-block memory, each block being four words in length. If the memory is divided into a program area, a data area, and two variable areas, the cache memory will have access to all areas at the same time. Every time the data being asked for by the processor is not in cache, the control logic transfers four words from the main memory to the cache memory. If the addressing is sequential, the next three cycles will be accessing the cache memory, which gives the optimum hit ratio of 75%.

## Error Check and Correction (ERCC) and Error Logging

**Error Correction** — The in-5150 error correction circuits provide on-board, single-bit correction capability. This feature is particularly useful when the in-5150 is used with CPU systems that do not include the processor

ERCC option. On-board ERCC greatly increases memory reliability and minimizes non-recoverable system failures. In the case of a data error, the error correcting circuitry corrects the appropriate bit. After correction, the in-5150 sends the data to the processor. An inhibit switch on the in-5150's control panel permits disabling of the ERCC, while the error logging circuits remain functional and parity bits are still generated from the input data.

**Error logging** — On-line error logging is performed with or without ERCC enabled and provides enough storage to log error information for any or all dynamic RAM memory elements on the board. This feature is extremely useful, since it facilitates location of faulty memory chips and enables 'no supervision' testing without software error logging support which improves memory board maintainability. On-line operation also means that the processor does not have to stop while the error logger is being interrogated by the operator. The on-line error logging feature is easy to use (see Figure 2). Pressing the SK (seek) button on the in-5150 control panel causes the data bit error to be shown on an LED display. The LEDs indicate whether the error is single bit or multiple bit. If a single bit error has been detected, the bit that failed is indicated as well as the chip row location. If a multiple bit error has been detected, the row in which the error occurred is indicated. The latest errors are always displayed. The error logging circuit is cleared by pressing the CL (clear) switch on the control panel.

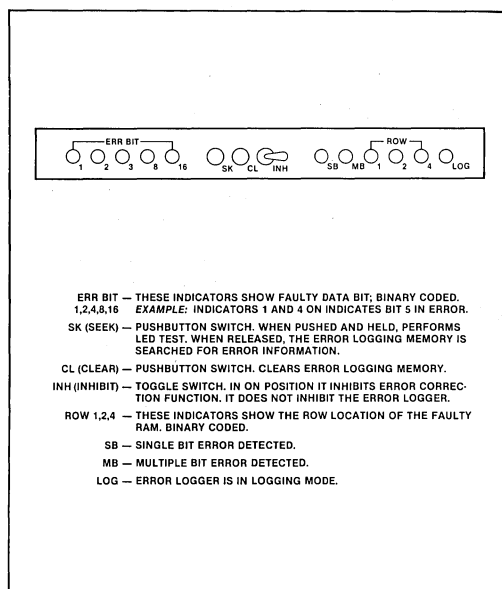


Figure 2. Error Logger Control Panel and Indicators

## SPECIFICATIONS

### Storage Capacity

32K words with cache memory  
64K words with cache memory

### Word Length

16 data bits plus five ERCC check bits

### Performance

#### Cycle Time

Read or write mode — 700 ns  
Read from cache memory — 200 ns

#### Access Time

350 nanoseconds  
200 nanoseconds from cache memory

### Operational Modes

Read  
Write  
Read-modify-write  
Refresh (transparent)

### Refresh Rate

Every 16  $\mu$ s

### Interface Characteristics

TTL compatible  
16 bidirectional data input/output lines  
20 binary address input lines  
22 control and status lines

### Physical Characteristics

Width — 15 in. (38.1 cm)  
Height — 15 in. (38.1 cm)  
Depth — 0.5 in. (1.27 cm) maximum  
Weight — less than 3.1 lb (1.36 kg)

### Electrical Characteristics

#### Power Requirements

Voltage	Standby Current	Operating Current	Standby Power	Operating Power
+5V	9.4A	9.6A	47W	48W
+15V (+12V)	0.2A	1.1A*	3W	16.5W (22.5W)*
-5V (-15V)	5mA	8.5mA*	15mW	42.5mW (75mW)*

\*When cache memory is all miss and no hit, +15V may rise to 1.5A and -5V current may rise to 15mA

### Environmental Characteristics

**Ambient Temperature** — 0°C to +50°C operation,  
-40°C to +120°C non-operating

**Relative Humidity** — Up to 90% with no condensation

**Altitude** — 0 to 10,000 ft (3048 meters) operating; up to 50,000 ft (15,240 meters) non-operating

### Reference Manuals

TM-5150-000 — in-5150 Technical Manual, Number 111767

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-5150-032	32K words, word length 16 bits	Add-in memory card for Data General Eclipse computers with cache memory, ERCC (ECC), and error logger
CM-5150-064	64K words, word length 16 bits	Add-in memory card for Data General Eclipse computers with cache memory, ERCC (ECC), and error logger



## 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**

**32K, 64K, 96K, or 128K word capacity and compatible with resident core memory**

**High circuit density, reducing system cost**

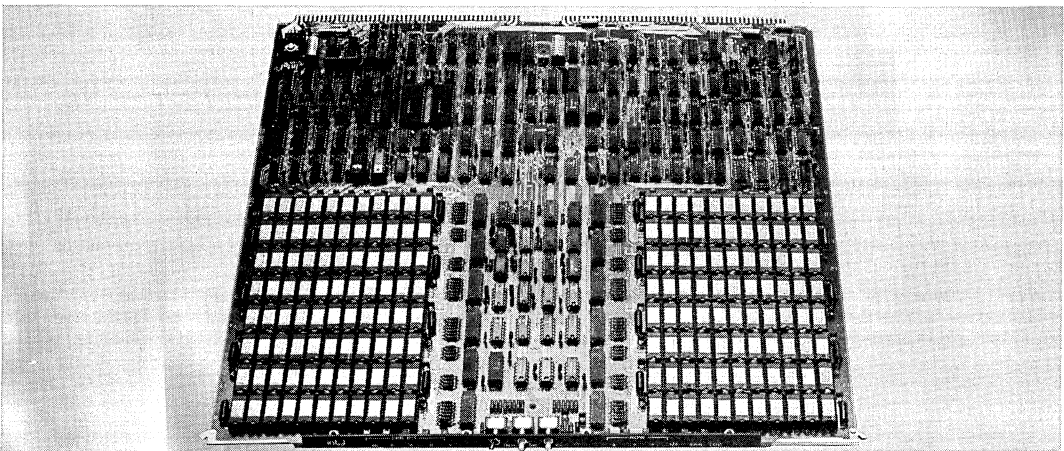
**Low power dissipation**

**Battery backup operation supported**

**Full commercial operating temperature/humidity range**

**One year warranty**

The Intel in-5160 is a 32K, 64K, 96K, 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.



## 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, MPPU, 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.

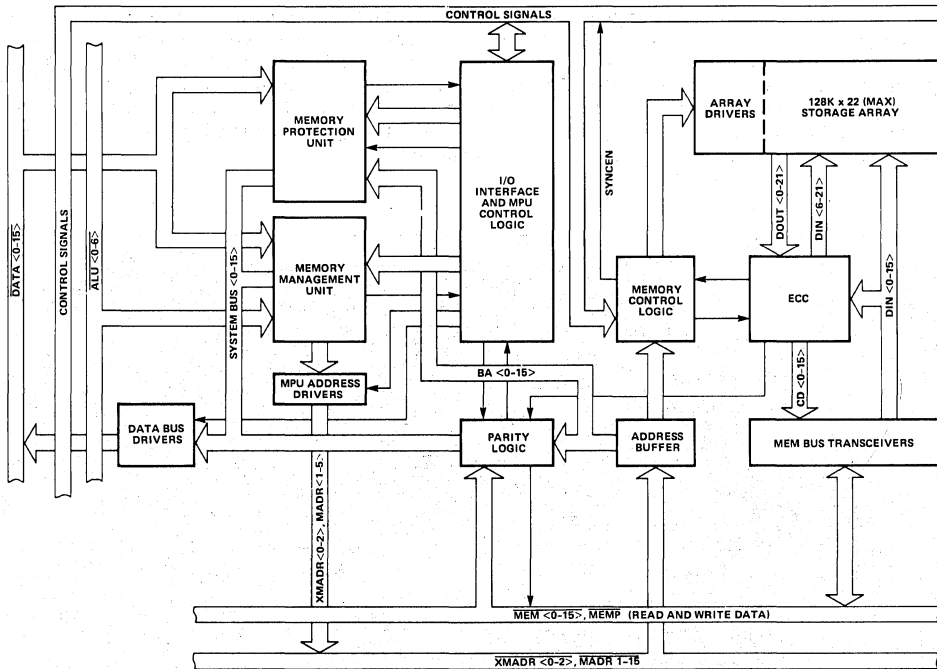


Figure 1. in-5160 NOVA-3 Add-in Memory Simplified Block Diagram

## 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.

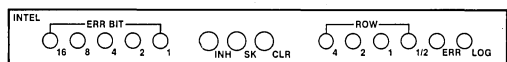
When a violation is detected, the MMPU initiates a map interrupt routine in the processor.

## 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 on-board 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  
4,2,1,1/2 (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 manufactured by Intel are covered by a one year warranty.

## SPECIFICATIONS

### Storage Capacity

32K, 64K, 96K, 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**

<b>Model</b>	<b>Capacity</b>	<b>Description</b>
CM-5160A-032	32K × 17-bit	32K memory with parity
CM-5160A-064	64K × 17-bit	64K memory with parity
CM-5160A-096	96K × 17-bit	96K memory with parity
CM-5160A-128	128K × 17-bit	128K memory with parity
CM-5160B-032	32K × 22-bit	32K memory with ECC
CM-5160B-064	64K × 22-bit	64K memory with ECC
CM-5160B-096	96K × 22-bit	96K memory with ECC
CM-5160B-128	128K × 22-bit	128K memory with ECC
CM-5160C-032	32K × 17-bit	32K memory with parity and MMPU
CM-5160C-064	64K × 17-bit	64K memory with parity and MMPU
CM-5160C-096	96K × 17-bit	96K memory with parity and MMPU
CM-5160C-128	128K × 17-bit	128K memory with parity and MMPU
CM-5160D-032	32K × 22-bit	32K memory with ECC and MMPU
CM-5160D-064	64K × 22-bit	64K memory with ECC and MMPU
CM-5160D-096	96K × 22-bit	96K memory with ECC and MMPU
CM-5160D-128	128K × 22-bit	128K memory with ECC and MMPU



## MU-5780 VAX\*-11/780 ADD-IN MEMORY CARD

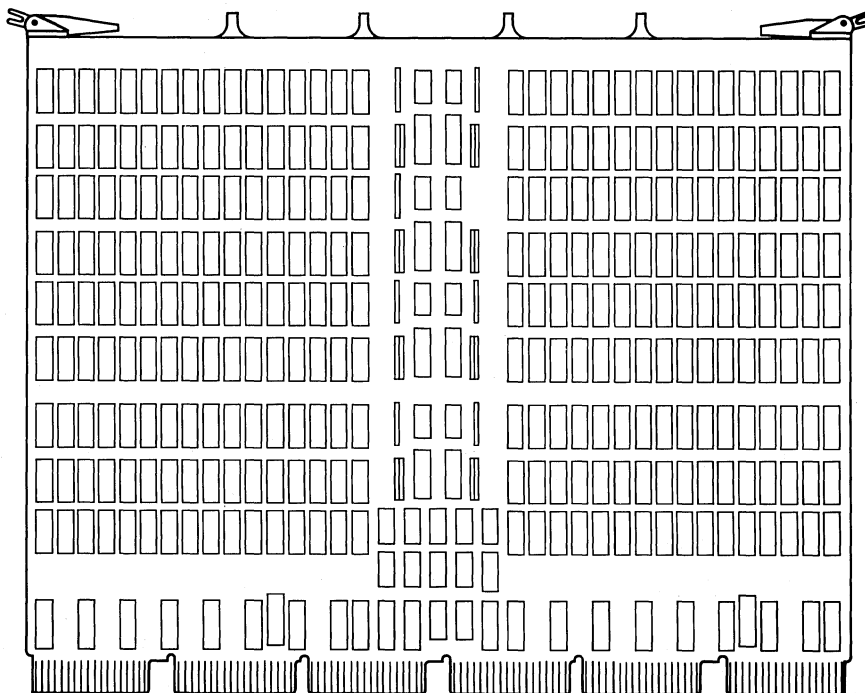
- 512KB or 1024KB Capacity (Equivalent to 2 or 4 DEC M8210 Memory Modules)
  - Spare Memory Devices On-Board
- Replaces DEC\* MS 780-DB or MS 780-DC
  - Complete Hardware and Software Compatibility

The Intel® MU-5780 is a 512KB or 1024KB plug-in memory card designed for use in any DEC VAX-11/780 computer. Organized as 64K or 128K words of 72 bits each, the MU-5780 operates at speeds controlled by the VAX-11/780 memory controller. Typical READ cycle and access times are 530 ns and 250 ns, respectively.


The MU-5780 may be used to expand the MS 780-C or MS 780-CC basic VAX-11/780 memories up to 4096K Bytes (four megabytes). The MU-5780 is available only as double or quadruple the capacity of the DEC M8210 memory module. Because VAX-11/780 memory backplane slots are prewired as 256K Bytes each, the 512KB MU-5780 must be installed in every other backplane slot and the 1024KB MU-5780 must be installed in every fourth slot.

Two pre-tested memory devices are plugged into sockets on the MU-5780 to provide on-board spares. These devices may be used to replace any failing memory devices in the field.

\*VAX, 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<sub>l</sub>, INTEL, INTELLEC, MCS, <sup>™</sup>m, ICS, ICE, UPI, BXP, ISBC, ISBX, INSITE, IRMX, CREDIT, RMX/80, <sub>μ</sub>Scope, 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.



*Insite™ User's Program  
Library*

---

**22**

THE UNIVERSITY OF CHICAGO  
DEPARTMENT OF CHEMISTRY  
5800 S. UNIVERSITY AVENUE  
CHICAGO, ILLINOIS 60637  
TEL: (773) 835-3100  
FAX: (773) 835-3101  
WWW: WWW.CHEM.UCHICAGO.EDU

1. Name of the donor: \_\_\_\_\_

2. Address of the donor: \_\_\_\_\_

3. City and State: \_\_\_\_\_

4. Zip Code: \_\_\_\_\_

5. Date of gift: \_\_\_\_\_

6. Description of gift: \_\_\_\_\_

7. Value of gift: \_\_\_\_\_

8. Name of recipient: \_\_\_\_\_

9. Address of recipient: \_\_\_\_\_

10. City and State: \_\_\_\_\_

11. Zip Code: \_\_\_\_\_

12. Date of receipt: \_\_\_\_\_

13. Name of recipient: \_\_\_\_\_

14. Address of recipient: \_\_\_\_\_

15. City and State: \_\_\_\_\_

16. Zip Code: \_\_\_\_\_

17. Date of receipt: \_\_\_\_\_



## INSITE™ USER'S PROGRAM LIBRARY

- Programs for 8008, 8048, 8080, 8085, 8088 and 8086 Processors
- Updates of New Programs Sent Bi-Monthly
- Diskettes, Paper Tapes, and Listings Available for Library Programs
- Hundreds of Programs
- For Each Accepted Program Submittal, Insite™ Library Will Provide Either a One-Year Free Membership, Five Free Paper Tapes, or Three Programs on Diskette

Insite Intel's Software Index and Technology Exchange, 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. 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.



**LIBRARY PROGRAMS AVAILABLE ON PAPER TAPE AND SINGLE OR DOUBLE DENSITY DISKETTE**

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, Intelligence, Micromap,  $\mu$ Scope and the combination of ICE, iCS, ISBC, ISBX, MCS, or RMX and a numerical suffix.



## INSITE™ USER'S PROGRAM LIBRARY

### 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 on the back of this data sheet.

### 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 not required to purchase or submit 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.

User's Library members will be updated quarterly with abstracts of new programs submitted to Insite 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.

### PROGRAM SUBMITTAL

Programs submitted for our review must follow the guidelines listed below:

1. Programs must be written in a standard Intel Assembly Language or PL/M 80. These languages are documented in the following manuals:
  - a. *8008/MCS-8 Assembly Programming Manual #98-019B*
  - b. *8080/8085 Assembly Language Programming Manual #9800301C*
  - c. *8080/8085 Floating-Point Arithmetic Library User's Manual #9800452B*
  - d. *PL/M-80 Programming Manual #98-268B*
  - e. *MCS-48 and UPI-41 Assembly Language Manual #9800255C*
  - f. *FORTRAN-80 Programming Manual #9800481A*
  - g. *8086 Assembly Language Reference Manual #9800640A*
  - h. *PL/M-86 Programming Manual #9800466A*
2. A source listing of the program must be included. This must be the output listing of a compile or assembly. All accepted programs must be capable of compilation or assembly by Intel standard compilers or assemblers. No consideration will be given to partial programs or duplication of existing programs.
3. A test program which assures the validity of the contributed program must be included. This must show the correct operation of the program.
4. A source paper tape or diskette of the contributed program is required. This will be used for the reproduction of tapes for other members.

Complete the Submittal Form as follows: (please type or print)

1. **Processor** (check appropriate box).
2. **Program Title:** Name and brief description of program function.
3. **Function:** Detailed description of operations performed by the program. Attach additional pages if necessary.
4. **Required Hardware:**  
*For example:* TTY or Ports 0 and 1  
Interrupt Circuitry  
I/O Interface  
Machine line and configuration  
for cross products
5. **Required Software:**  
*For example:* TTY Drive  
Floating Point Package  
Support software required for  
cross products
6. **Input Parameters:** Description of register values, memory areas or values accepted from input ports.
7. **Output Results:** Values to be expected in registers, memory areas or on output ports.
8. **Program Details:** (for resident products only)
  - a. Register modified
  - b. RAM required (bytes)
  - c. ROM required (bytes)
  - d. Maximum subroutine nesting level
9. **Assembler/Compiler Used:**  
*For example:* PL/M 80  
Intel Macro Assembler  
FORTRAN 80
10. **Programmer, Company and Address.**

# insite<sup>T.M.</sup> INTEL<sup>®</sup> USER'S LIBRARY SUBMITTAL FORM

8008    8048    8080/8085    8086/8088    Other \_\_\_\_\_ (Use additional sheets if necessary)

Program Title	
Function	
Required Hardware	
Required Software	
Input Parameters	
Output Results	

Registers Modified:	Programmer:
RAM Required:	Company:
ROM Required:	Address:
Maximum Subroutine Nesting Level:	City:
	State:
Assembler/Compiler Used:	Phone:

---

**insite** <sup>T.M.</sup> **USER'S LIBRARY MEMBERSHIP FORM**

I am interested in becoming a member of Insite™

Enclosed is  Check  Money Order  Program Submittal

NAME: \_\_\_\_\_

COMPANY: \_\_\_\_\_

SHIP TO: \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_

---

**SEND TO NEAREST LOCATION:**

**North America**

Intel Corporation  
User's Library 6-5000  
Microcomputer Systems  
3065 Bowers Avenue  
Santa Clara, California  
95051  
Ph. 408-987-8080

**Europe**

Intel International Corp. S.A.  
User's Library  
Rue du Moulin a Papier 51  
Boite 1  
B-1160 Brussels, Belgium  
Ph. 02-660-3010

**Orient**

Intel Japan K.K.  
User's 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.)

Please refer to Intel OEM Price List for membership fee.

---

*Microcomputer  
Workshops*

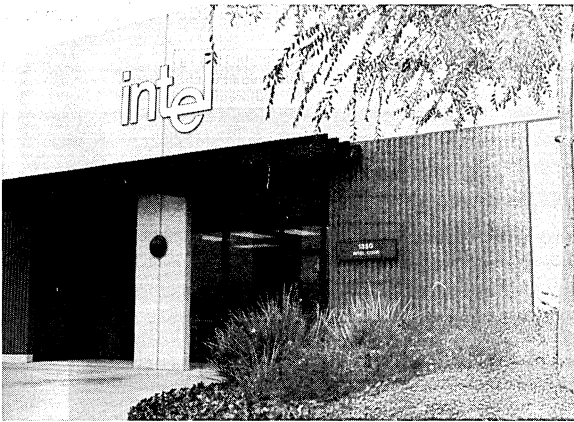
**23**

---





# Intel Microcomputer Workshops



## Training in Microcomputers

Whether your present involvement with microcomputers is a result of long-term planning or simply an exploratory project undertaken by your company in response to external circumstances, there exists an obvious and urgent need for you to familiarize yourself with this exciting new technology as quickly and painlessly as possible. If the microcomputer is, or is destined to become, a part of your working scene (and there is no field of manufacturing or data processing it will not touch sooner or later) then the importance of carefully planned training cannot be over-emphasized. A modest outlay in time and money now can save many weeks of self-study and could well prevent some very expensive mistakes during the initial development of your systems.

## Why Intel Training?

### EXPERIENCE

Intel has been training engineers in the application of microprocessors and the development of microcomputer systems since the early 70's, and there are now many thousands of engineers creating the most advanced microcomputer systems as a direct result of successful training with us.

### VARIETY OF COURSES

Intel Microcomputer Workshops offer training at three levels — introductory, intermediate and advanced. Your particular training requirement may involve just one or several courses, so we have taken care to ensure that each Workshop is a high quality training module that can either stand independently or integrate with other modules to form a complete coverage of our subject. The Workshops are constantly being updated to include the latest developments in devices and support software, and course materials are similarly reviewed on a regular basis — both for clarity and content.

### PRODUCT KNOWLEDGE

As the designers and manufacturers of the most widely accepted microcomputer products in the world, our knowledge is both comprehensive and topical. Remember the saying about 'the horse's mouth'!

### EXTENSIVE MATERIAL

Teaching aids include slide and video tape equipment, student notebooks and a wide range of printed materials which are designed to provide post-training assimilation and act as practical reference manuals in your own laboratory.

### "HANDS-ON" EXPERIENCE

We believe that students learn better by doing than by listening, so a sizeable proportion of course time is devoted to dynamic training via the Intellec MDS development system, appropriate single board computers, In-Circuit Emulators (ICE), I/O units for programming exercises, and computer kits for design and debugging sessions. Each student therefore receives valuable 'hands-on' experience of the principles and techniques featured in the lecture sessions.

## Where is Intel Training?

### TRAINING CENTERS

Workshops are scheduled nearly every week of the year in our three regional training centers:

**BOSTON AREA**  
Intel Customer Training  
27 Industrial Ave.  
Chelmsford, MA 01824  
(617) 256-1374

**CHICAGO AREA**  
Intel Customer Training  
Gould Center, East Tower  
2550 Golf Rd., Suite 815  
Rolling Meadows, IL 60008  
(312) 981-7250

**SAN FRANCISCO AREA**  
Intel Customer Training  
1350 Bordeaux Dr., SV3-1  
Sunnyvale, CA 94086  
(408) 734-8102

**ADDITIONAL LOCATIONS**

Courses have been scheduled in 35 additional locations across the U.S., Canada, and Mexico. These cities include Los Angeles, Washington, Dallas, Minneapolis, New York, Atlanta, and Ottawa. Intel Workshops are also available in major European cities and Japan. (See local workshop brochure)

**CUSTOMER-SITE WORKSHOPS**

All workshops detailed in the following pages can, by special arrangement, be presented with full laboratory equipment by Intel instructors at your facility. Considering the transportation and lodging savings, the breakeven point is typically 5-7 attendees. For further information about on-site training, please contact your nearest regional training center.

**EVENING WORKSHOPS**

If you cannot find time to attend our day-time workshops, check the schedule for our evening workshops.

**Which Workshops Should You Attend?**

As a DESIGN ENGINEER or SYSTEM ENGINEER with a good understanding of digital electronics, you might start with the MCS-48/49, MCS-51, MCS-80/85, or iSBC Workshops. If you lack the digital electronics background, start with the Introduction to Microcomputers Workshop. Unless you have working experience with microprocessor-based design and assembly language programming, you should attend the MCS-80/85 Workshop prior to attending the iAPX86,88 Workshops.

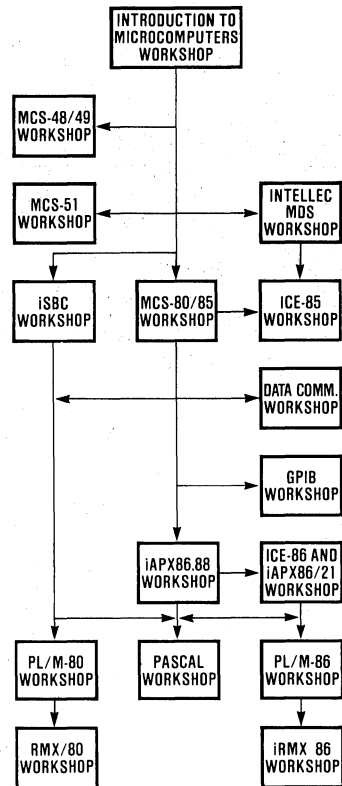
As a PROGRAMMER, you may be ready to start with the iSBC, PASCAL, or PL/M Workshops; however, if you intend to use assembly language the MCS-48/49, MCS-51, MCS-80/85, or iAPX86,88 Workshops should be attended first. We suggest that you also consider the Introduction to Microcomputers Workshop to gain basic knowledge of digital electronics and machine language, since these topics are usually much more important to the microcomputer programmer than to the minicomputer or large computer programmer.

As a MANAGER, ADMINISTRATOR, or PROJECT MANAGER wishing an overview of this new field, the Microcomputer Concepts for Managers Workshop would be suitable; or if you want more details, the Introduction to Microcomputers Workshop is recommended.

If you have sufficient working experience with microcom-

**Which Workshops Should You Attend?**

**Recommended Sequence of Workshops**



puter hardware and software you might attend any of the Intel Workshops without first attending a prerequisite workshop; however, since microcomputers are new to many people, we recommend the sequence indicated in the chart for those who want to advance their microcomputer education.

## Announcing . . .

---

### **New MCS-51™ Microcontroller Workshop**

This new 4-day workshop will give a boost to your first 8051 based project or allow you to evaluate the 8051. The workshop covers the architecture, instruction set, and software development package of this new single chip microcontroller. (See page 18)

### **New Data Communication Chips Design Workshop**

This new workshop focuses on bit-serial synchronous communication modes and devices, and provides both an overview and hands-on lab for data communications. (See page 19)

### **New IEEE-488 GPIB Workshop**

This course introduces the concepts and specifications of the IEEE-488 General Purpose Interface Bus by design and implementation with the Intel IEEE-488 chip set. (See page 19)

### **New ICE-85™ User's Workshop**

This 3-day workshop provides intensive and comprehensive training on the ICE-85 In-Circuit Emulator hardware/software debug tool through lecture and hands-on laboratory. (See page 21)

### **New Training Center Facilities and Labs**

Responding to your request, we have significantly increased the number of Intellec Microcomputer Development Systems and other lab

equipment in our training centers. We had to move to new facilities to accomplish this, but our goal is to deliver the best training you can get your hands-on.

### **New Workshop Vans for More Hands-On at Your Site**

We have outfitted three vans with new Intellec Microcomputer Development Systems and ICE In-Circuit Emulators, so we can easily bring our extensive workshop lab equipment to your facility.

### **New Video Tape Library**

#### **ICE CONCEPTS**

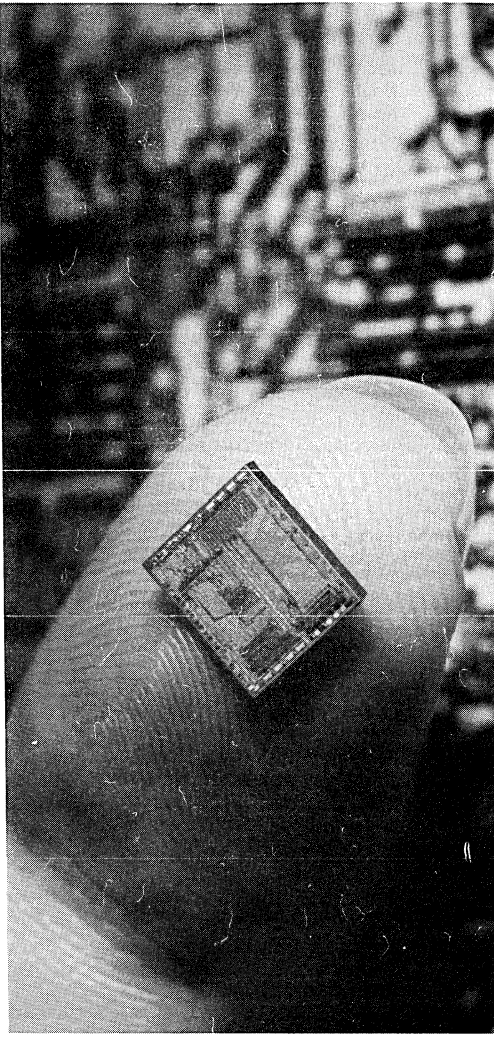
Intended for users of an In-Circuit Emulator, this 22-minute color video tape demonstrates the unique capabilities of ICE debug tool, including the debugging of an actual traffic control application.

#### **INTELLEC® MICROCOMPUTER DEVELOPMENT SYSTEM OPERATION**

Intended for users of an Intellec 220, 225, 230, 235, 240, or 245 system. This 20-minute color video tape shows how to power-on the system, handle the floppy disk, edit a file with the CRT editor CREDIT, and use simple ISIS-II commands like COPY, DELETE, and DIRECTORY.

Both are available in 3/4" and 1/2" video tape cassette formats at \$250.00 per copy. Order from the San Francisco Area Training Center, (408) 734-8102.

## Microcomputer Concepts for Managers Workshop



### Introductory

#### Course Description

- Microcomputer terminology, buzzwords, and fundamental programming concepts explained
- Hardware/software development process outlined
- Microcomputer applications reviewed
- Lab sessions on SDK-85 and in programming in BASIC on the Intellec Series II Microcomputer Development System

#### Attendees

- Project leader, manager, or administrative person who needs overview of microprocessor and microcomputer concepts

**Length:** 2 Days

**Tuition:** \$395  
\$325 (Group rate)

#### Course Outline

##### DAY 1

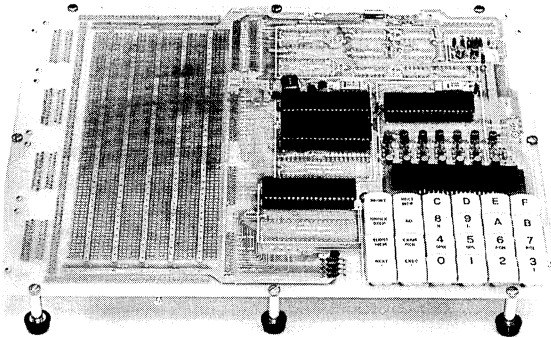
Terminology  
Microcomputer System  
Organization  
Memory  
Computer Selection  
Lab: Using SDK-85

##### DAY 2

Introduction to Software  
Development System Overview  
Single Board Computer  
Overview  
Reliability, Testing, Timing  
Project Chronology  
Application Topics  
Lab: Programming in BASIC

# Introduction to Microcomputers Workshop

---



Included in the price of the course is an SDK-85 kit which includes:

- 3 MHz 8085 CPU (enhanced 8080)
- Keyboard — 24 keys
- Display — 6 digits
- Monitor ROM 2048 bytes
- RAM Memory 256 bytes
- 38 I/O lines
- Teletype interface
- Complete documentation

## Course Description

- Fundamental computer concepts and terminology introduced
- Operation of the Intel 8085 microprocessor explained
- 8085 assembly language programming
- Stacks, subroutines, interrupts and I/O interfacing introduced
- Lab sessions on SDK-85 System Design Kit
- An SDK-85 Kit (valued at \$300) is yours to keep

## Attendees

Engineers, scientists, or other technical people with limited computer or digital electronics background

**Length:** 4 Days

**Tuition:** \$895 (includes SDK-85 Kit)  
\$750 (Group rate)

## Course Outline

### DAY 1

Introduction to Microelectronics  
Computer Concepts  
Computer Languages  
Using the SDK-85  
Lab: Kit Operation and Programming  
Moving Data

---

### DAY 2

Delay Loops  
Lab: Audio Oscillator using Digital Techniques  
Subroutines  
Stack Operation  
Lab: Using Subroutines

---

### DAY 3

Logic Instructions  
Lab: Handshaking Techniques  
Addition  
Lab: Multi-Function Programs  
Microprocessor Operation

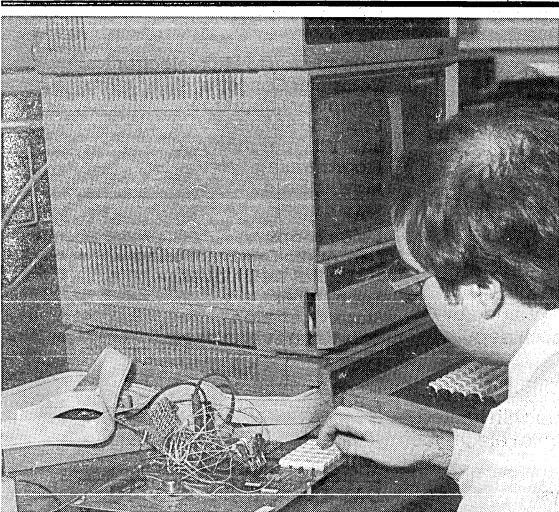
---

### DAY 4

Interrupts  
Memory Operation and Address Decoding  
Programmable Peripheral Chips  
Lab: Chip Programming Exercises  
Introduction to the Development System, ICE, Assembler, and Higher Level Languages  
Course Summary and Review

---

## MCS<sup>®</sup>-80/85 System Design Workshop



### Course Description

- 8085 architecture explained in detail
- Assembly language programming for 8080/8085
- Design and development of systems using Intel 8080, 8085 chips
- Interfacing and programming techniques
- Lab sessions "hands-on" using the Intellec Series II microcomputer development system
- ICE-85 In-Circuit Emulator used to debug programs

### Attendees

- Design engineer or programmer who is familiar with binary numbers and logic functions
- Prior attendance at Introduction to Microcomputers Workshop or equivalent knowledge is recommended

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

### Course Outline

#### DAY 1

Introduction to Microprocessors  
Assembly Language Instructions  
Programmed Input and Output  
Microcomputer Development  
System Monitor

Lab: Using System Monitor

#### DAY 2

Microcomputer Development  
System Disk Operating System  
CREDIT Text Editor and Macro  
Assembler  
The Processors

Lab: Using Text Editor and  
Assembler

#### DAY 3

Stacks and Subroutines  
Interrupts  
Input and Output Techniques  
Programming Techniques  
Lab: In-Circuit Emulator Introduction

#### DAY 4

In-Circuit Emulator  
8155, 8355, 8185 Peripherals  
Memory Interfacing  
Lab: Using 8085 In-Circuit  
Emulator

#### DAY 5

8255 and 8251A Peripherals  
Tools for Modular Program  
Development  
Chips vs. Boards  
Preview of 8088

## **iSBC™ Single Board Computer Workshop**

### **Course Description**

- Covers state of the art microprocessors and single board computer systems
- Explains operation and programming of selected SBC products including: iSBC80/30, iSBC80/24, iSBC86/12A
- Describes design, configuration and implementation of single board computer systems for industrial applications
- Intel Development System used in laboratory sessions
- Laboratory sessions also implement a variety of iSBC system configurations
- ICE In-Circuit Emulator used to aid system debugging
- Includes field service, fault location, and maintenance topics

### **Attendees**

- System engineer or programmer who will be using iSBC Products
- Some programming experience is required, preferably 8085 assembly language; PL/M workshop is helpful, but not required

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

### **Course Outline**

#### **DAY 1**

Survey of Intel Single Board Computers, Languages, and Operating Systems  
Overview of the Intel iSBC 80/30 Processor Board and Associated Peripheral Interface Chips  
The 8085A Microprocessor Fundamentals and Assembly Language Programming Techniques  
Designing with the Intel 8255 Programmable Peripheral Interface Chip  
Lab: iSBC Monitor and Parallel I/O Capability of iSBC 80/30 Processor Board

#### **DAY 2**

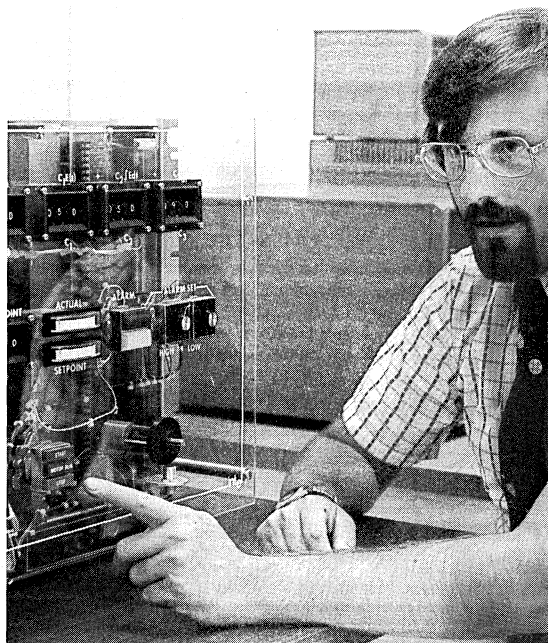
Intel Development System Credit — CRT Based Text Editor  
ASM 80 Assembler/PLM-80 Programming  
Tools for Modular Programming — (link, locate, include and submit)  
ICE-85, In-Circuit Emulator  
Lab: Development of Programs for iSBC Processor Boards via Development System

#### **DAY 3**

Design and implementation of a Serial Communication System  
iSBC 544 Intelligent Communication Controller  
iSBC 80/24 Single Board Computer and Multimodule Expansion Boards  
Multibus Specifications and Architecture  
Lab: Development Programs to operate on iSBC Processor Boards, Multimodule I/O Expansion and Memory Boards

#### **DAY 4**

iSBC 86/12A Single Board Computer  
iSBC Memory Boards, RAM, PROM, EPROM, and Bubble Memory Boards



iSBC Parallel I/O Expansion Boards, iSBC 519, iSBC 569  
iSBC Analog I/O Boards, iSBC 711, 724, and 732  
iCS Industrial Control System and Signal Conditioning/Terminator Panels  
Lab: iSBC 544 Communication Lab

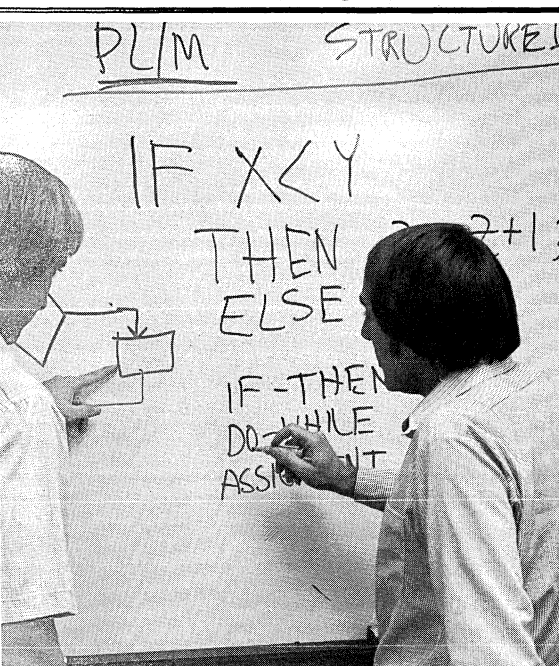
#### **DAY 5**

FSP and Model 810 — Software Development Modules  
RMX/80 Real-Time Multitasking Executive Service and Maintenance of iSBC, iCS Systems  
Future iSBC Products  
Lab: iSBC 544, iSBC 80/30 Multitasking Lab

## **8080/8085, Peripheral Chips**



# PL/M-80 Language/ Software Design Workshop



## Course Description

- Learn PL/M programming for 8080 and 8085 microprocessors
- Design, implement, and debug PL/M-80 modular software
- LINK multi-modules using libraries
- Develop software for disk file handling
- "Hands-on" laboratory sessions using the Intellec Series II microcomputer development system
- In-Circuit Emulator used for system debugging

## Attendees

- Design engineer or programmer
- Prior programming experience will assist in obtaining maximum benefit from the course
- Introduction to Microcomputers or MCS-80/85 System Workshops recommended

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

## Course Outline

### DAY 1

PL/M Overview  
Declaration Statements  
Sequential Statements  
Conditional Statements  
Loop Statements  
Program Preparation  
Lab: Microcomputer Development Systems Introduction

### DAY 2

Intellec Series II Development System  
ISIS-II Commands  
System Monitor  
CREDIT Editor  
Lab: Implement Program for Oven Control

### DAY 3

Procedures  
Compiler Controls  
Linking Modules  
Locating  
Submit Command  
Lab: Design and Implement On-Line Editor

### DAY 4

In-Circuit Emulation  
Libraries  
Lab: Use ICE-85 Module on SDK-85 Kit

### DAY 5

System Calls  
Interrupts  
Reentrancy  
Lab: Design and Implement File Handling Program

## RMX/80™ Operating System Workshop

---

### Course Description

- Operation and implementation of Intel RMX/80 Real-Time Multi-Tasking Executive
- Basic concepts of multi-tasking are explained
- Concurrency of tasks, asynchronous events, priorities, scheduling, resource sharing, interrupts, and inter-task communication discussed
- Lab sessions on task writing and system generation using iSBC Processor Boards
- In-Circuit Emulator used to aid system debugging

### Attendees

- System engineer or programmer who will be using RMX/80 in iSBC system application
- PL/M-80 Language/Software Design Workshop or equivalent knowledge is required (eg. based variables, structures, and declare literally)
- Familiarity with Intel Microprocessor Development System is required

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

### Course Outline

#### DAY 1

Introduction to Multi-Tasking  
What is RMX/80?  
Review of Development Process  
Lab: RMX/80 Operation

#### DAY 2

RMX/80 Model  
Implementing RMX/80 Tasks  
Configuring a System  
Lab: Writing Tasks and Configuring System

#### DAY 3

RMX/80 Extensions: Terminal Handler  
Task Priorities  
Lab: Implementing a Terminal Handler Task

#### DAY 4

RMX/80 Extensions:  
The Debugger and  
The Free Space Manager  
Lab: Implementing RMX Extensions

#### DAY 5

RMX/80 Extensions:  
The Disk System  
Lab: Implementing RMX Extensions



---

**8080/8085**

## IAPX 86,88 System Design Workshop



### Course Description

- Design systems for 8086/8088
- Develop programs using 8086 assembly language
- Lectures cover CPU, addressing modes, interrupt system, timing, I/O
- Laboratory "hands-on" uses Intellec Series II Microcomputer Development System and SDK-86 Kit
- ICE-86 In-Circuit Emulator introduced as software debug tool

### Attendees

- Workshop for design engineer or programmer who will use or evaluate 8086, 8088, or iSBC 86/12A
- Prior experience with microcomputers and a working knowledge of assembly language is required, and the MCS-85 workshop is recommended

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

### Course Outline

#### DAY 1

Introduction to Assembly  
Language Instruction

8086/8088 CPU's

I/O Operations

System Design Kit (SDK-86)

Lab: Using the SDK-86 and the  
Serial Monitor

#### DAY 2

Development System

Addressing Modes

Programming Exercises

Lab: Using the 8086 Assembler  
and the Development System

#### DAY 3

Procedures

String Operators

Interrupt System

Lab: Programming with Procedures  
and the String Operators

#### DAY 4

Programming for Large Systems

CPU Timing

8086/8088 Support Chips

Lab: Programming with Multiple  
Segments

#### DAY 5

Programming with Multiple  
Modules

Introduction to ICE-86

Introduction to Multiprocessing  
and the Multibus

### Course Description

- Detailed discussions on assembler directives, segmentation, library
- The design and programming of large systems
- 8089 I/O Controller explained
- "Hands-on" lab sessions use Intellec Series II Development System and iSBC 86/12A Single Board Computers
- ICE-86 In-Circuit Emulator used to debug programs

### Attendees

- Programmer or system designer who needs complete understanding of 8086 assembly language programming and ICE-86 debugging techniques
- Knowledge of Intellec Development System assumed
- Prior attendance at MCS-86/88 Workshop required

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

### Course Outline

#### DAY 1

Review of the 8086/8088  
Advanced Programming Topics  
Linkage  
The Software Development Process  
Lab: PL/M-86 Linkage

#### DAY 2

Execution Vehicle Considerations  
iSBC 86/12A Board  
8259A Peripheral Chip Programming  
Introduction to ICE-86 Emulator Module  
Lab: Interrupt Handling

#### DAY 3

Advanced ICE-86 Module Features  
Interrupt Structures  
Multiprocessor Topics  
Lab: ICE-86 Introduction

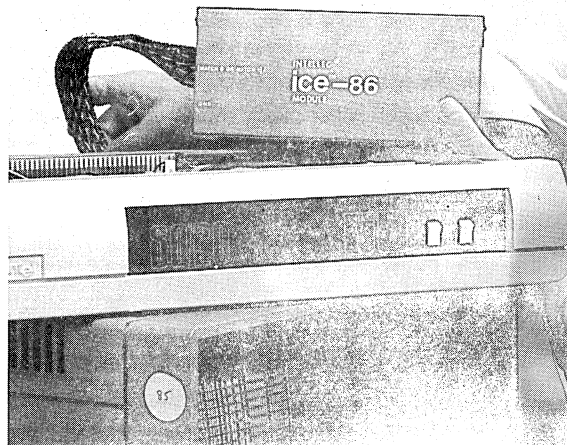
#### DAY 4

8086 Component Family Review  
Multibus Arbitration Logic  
Introduction to Input/Output Processors  
Lab: ICE-86

#### DAY 5

Advanced 8089 Topics  
Application Topics

## ICE-86/88™ and IAPX 86/21 Workshop



**8086/8088**

## PL/M-86 Language/Software Design Workshop



### Course Description

- Learn PL/M programming for 8086 microprocessor
- Design, implement, and debug PL/M-86 modular software
- LINK multi-modules using libraries
- "Hands-on" laboratory sessions using Intellec Series II Microcomputer Development System and SDK-86 Design Kit
- ICE-86 In-Circuit Emulator is introduced as software debugging tool

### Attendees

- Design engineer or programmer who is ready to use high level language with Intel 8086 or iSBC 86/12A
- MCS-86/88 workshop recommended as prerequisite for understanding the underlying structure of 8086 microprocessor

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

### Course Outline

#### DAY 1

Introduction  
Declaration Statements  
Data Elements  
Data Types  
Operators, Operations, and Priorities  
Execution Statements  
Lab: Microcomputer Development System and SDK-86

#### DAY 2

Procedures  
ISIS-II Disk Operating System  
CREDIT Editor  
PL/M-86 Compiler Options  
Lab: Implement Console I/O Routines

#### DAY 3

Data References  
Blocks  
Link and Locate Programs  
Built-In Facilities  
Lab: Design and Implement Modular Line Editor

#### DAY 4

Using Libraries  
Character String Handling  
Interrupt Procedures  
Reentrant Procedures  
Lab: Design and Implement Command Line Interpreter

#### DAY 5

ICE-86 Module  
Discussion of Selected Programs

## IRMX 86™ Operating System Workshop

### Course Description

- Introduction to iRMX 86 Real-Time Multi-Tasking Executive principles and implementation
- Basic concepts of multi-tasking are explained
- Concurrency of tasks, asynchronous events, priorities, scheduling, resource sharing, interrupts, and inter-task communication discussed
- Lab sessions on task writing and system generation using iSBC Processor Boards
- In-Circuit Emulator used to aid system debugging

### Attendees

- System engineer or programmer who will be using iRMX 86 on an iSBC 86/12A Single Board Computer
- PL/M 86 Language/Software Design Workshop or equivalent knowledge is required (eg. based variables, structures, and declare literally)
- Familiarity with Intellec Series II Microcomputer Development System is required

**Length:** 5 Days

**Tuition:** \$895

\$725 (Group rate)

### Course Outline

#### DAY 1

Real-Time Concepts  
The iRMX 86 Model  
Overview of iRMX 86 Nucleus  
The Software Development Process

Lab: Write a single task for a system

#### DAY 2

The iRMX 86 Nucleus  
The Terminal Handler  
Lab: Write multiple tasks for a system which uses the Terminal Handler

#### DAY 3

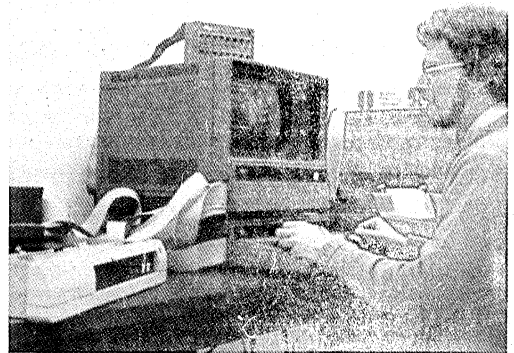
Interrupts  
The Debugger  
Lab: Write interrupt task

#### DAY 4

The Configuration Process  
Installing iRMX 86  
Lab: Write configuration modules

#### DAY 5

Overview of I/O System  
Future iRMX 86 Features  
Lab: Write task for file I/O



8086/8088

# PASCAL Programming Workshop



## Course Description

- Learn PASCAL programming on Inteltec Development Systems
- Design, implement, and debug PASCAL modular software
- Learn Structured Programming Concepts
- "Hands-on" laboratory using the Inteltec Series II microcomputer development system and PASCAL-80 or PASCAL-86 software

## Attendees

- Design engineer, scientist, or programmer
- No prior programming experience necessary

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

## Course Outline

### DAY 1

PASCAL Overview  
Block Structuring  
Lab: Microcomputer Development System Introduction  
Data Storage  
PASCAL Standard Data Types  
Program Construction in PASCAL  
Assignment Statements

---

### DAY 2

ISIS-II Disk Operating System  
Credit Editor  
PASCAL-80 Run Time System  
Lab: Design and implement PASCAL Program  
Conditional Statements  
Iteration-Loops  
Procedures and Functions  
Lab: PASCAL Program using Procedures

---

### DAY 3

Scalar Data Types  
Structured Data Types:  
1. Sets  
2. Arrays  
3. Strings  
4. Records  
Lab: PASCAL Program using Structured Data Types

---

### DAY 4

File Type  
Built-in Procedures and Functions to Manipulate Files  
Textfiles  
Interactive Files  
Untyped Files  
Lab: PASCAL program using File Data Structures

---

### DAY 5

Dynamic Data Structures  
Advanced Programming Topics  
Advanced Design Topics  
Lab: PASCAL Program using Dynamic Data Structures  
PASCAL-86 Compiler

---

## 2920 Signal Processor Workshop

### Course Description

- Design systems using 2920
- Develop programs using 2920 assembly language
- Use Microcomputer Development Systems software support tools for program assembly and simulation
- Investigate applications for 2920 signal processor

### Attendees

- Design engineer who is familiar with analog filter design, frequency domain analysis and S-Plane representation
- Prior digital design or digital filter experience recommended

**Length:** 4 Days

**Tuition:** \$795  
\$625 (Group rate)

### Course Outline

#### DAY 1

Overview of 2920  
Instruction Set  
Binary Arithmetic  
Simulator

Lab: Sawtooth Generator VCO

#### DAY 2

Constant Scaling  
2920 Assembler  
Constant Multiplication  
Simulator Output

Lab: Low Pass Filter and Two Pole Filter

#### DAY 3

Variable Multiplication  
Variable Division  
Signal Processing Application  
Compiler

Lab: Using the Compiler

#### DAY 4

Review Sample Techniques  
Noise  
Aliasing  
Application Discussion

Lab: Open





# MCS<sup>®</sup>-48/49 System Design Workshop



## Course Description

- Design and develop a system using Intel MCS-48/49 single chip microcomputers
- Programming in 8048 assembly language
- Explanation of Intel 8048 architecture, system timing and input/output design
- Lab sessions provide "hands-on" experience using PROMPT-48 and Intellec Series II micro-computer development system to develop 8048 programs
- ICE-49 In-Circuit Emulator used for system debugging
- Intel 8021, 8022, and 8041 chips are discussed

## Attendees

- Design engineer or programmer who is familiar with binary numbers and digital logic
- Prior attendance at Introduction to Microcomputers Workshop or equivalent knowledge recommended

**Length:** 5 Days

**Tuition:** \$895  
\$725 (Group rate)

## Course Outline

### DAY 1

Microprocessor System  
8048 Overview  
Assembly Language Instructions  
PROMPT-48 Design Aid  
Lab: Software Development  
Using PROMPT-48 Design Aid

### DAY 2

Assembly Language Instructions  
Subroutines  
Development System  
Text Editor and Macro  
Assembler  
Lab: Development System  
Introduction

### DAY 3

System Timing  
Interrupts  
Expanding I/O  
Lab: Programming Using Timer  
and Interrupts

### DAY 4

Expanding Memory  
Peripheral Interfacing Using  
8279, 8251, 8255, 8355, 8155  
In-Circuit Emulator  
Lab: Debugging Using ICE-49  
Module

### DAY 5

8021, 8041A, 8022  
Analog Interfacing

## MCS<sup>®</sup>-51 Microcontroller Workshop

### Course Description

- Program in 8051 Assembly Language and discuss system design with 8051 components
- Explanation of MCS-51 Architecture including:
  - Boolean Processor
  - I/O Port Structure
  - Timers
  - Serial Port
  - Interrupts
  - Memory Types and Memory Spaces
- "Hands-on" Lab sessions covering the MDS Development System, the ISIS Operating System, the CREDIT text editor, the MCS-51 Assembler, and execution vehicle.
- Review of some 8048 concepts.

### Attendees

This course is designed for the design engineer who wants to become familiar with the 8051 family of components. Prior knowledge of Intel components is not required. The attendee should be familiar with digital logic functions and have some programming experience. The Introduction to Microcomputer Workshop is recommended.

**Length:** 4 Days

**Tuition:** \$795  
\$625 (Group rate)

### Course Outline

#### DAY 1

Introduction  
Programming and Instruction Set Representation  
Internal Data Memory Lab  
Program Sequencing  
Immediate Data  
Lab: Traffic Light Control using Delays

#### DAY 2

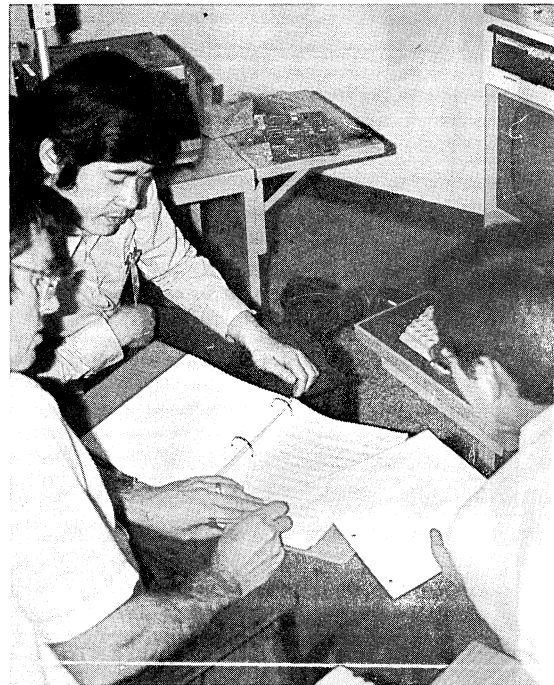
ISIS-II  
CREDIT  
MCS-51 Assembler  
Lab: 7-segment display Subroutines  
Lab: Traffic Light Controller Revisited using Subroutines

#### DAY 3

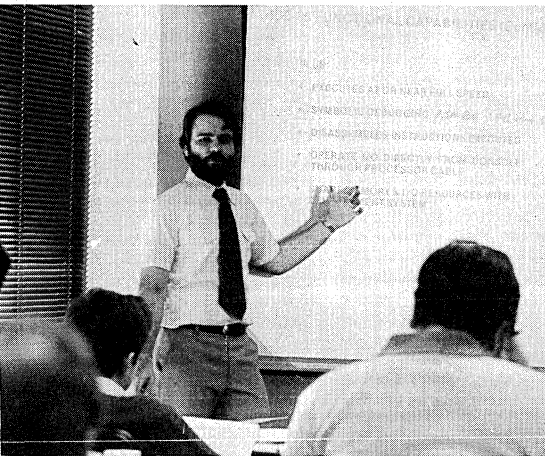
Port Structure  
Register Addressing  
Boolean Processor  
Lab: Multiplexing 7-segment display  
Indirect Addressing  
Lab: Using Table Look-up Techniques

#### DAY 4

Interrupts  
8051 Timers  
Serial Port  
Lab: Setting up and using the Serial Port to communicate with a CRT  
Arithmetic Instructions  
Accessing External Memory  
Application Topics



## Data Communication Chips Workshop



### Course Description

- Reviews data communication fundamentals
- Describes Asynchronous and Synchronous bit serial communication protocols: Bi-Synch, HDLC/SDLC, X.25
- Utilize Intel chips in laboratory:  
8251 USART  
8273 HDLC/SDLC Controller
- Explain new Intel chips for data communication: 8256 and 8274
- Discusses RS232, Local network and ether net

### Attendees

- Intended for the design engineer, service engineer, or programmer who needs to understand how serial data communication is used in a distributed system.
- The Introduction to Micro-computer Workshop or equivalent knowledge is required.

**Length:** 4 Days

**Tuition:** \$795  
\$625 (Group rate)

### Course Outline

#### DAY 1

History of Data Comm.  
Communication Hierarchy  
Non-standards  
Asynch  
**Lab: Using Intellec**

#### DAY 2

Byte Synch. Comm.  
Bi-synch  
RS232  
**Lab: Using SBC 544**

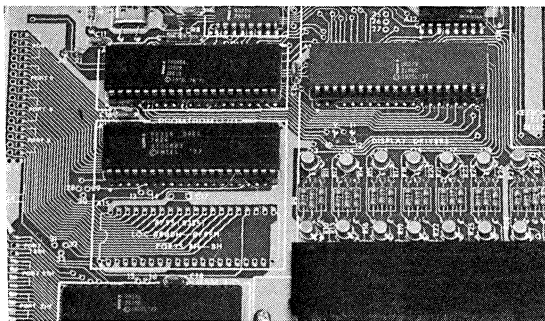
#### DAY 3

Bit Synchronous  
HDLC/SDLC  
Loop mode  
X.25 Protocol  
**Laboratory Demonstration**

#### DAY 4

Local networks  
Ether-net  
Star Configurations  
**Application discussion**

## IEEE-488 GPIB Chips Workshop



### Course Description

- Describes the General Purpose Interface Bus characteristics including protocol, electrical, and mechanical specifications
- Explains systems application examples and implementation guidelines
- Includes programming and use of:  
8291 GPIB Talker/Listener  
8292 GPIB Controller  
8293, 8296, 8297 Transceiver and Interface  
iSBC 80/24 and IEEE 488 Multi-module

### Attendees

- Design engineer or programmer who has prior assembly language or PL/M programming experience

- The Introduction to Micro-computer Workshop or equivalent knowledge is required. The Intel Development System Operation Workshop is recommended.

**Length:** 4 Days

**Tuition:** \$795  
\$625 (Group rate)

### Topics

IEEE-488/GPIB Overview  
Intel Chip Set Overview  
Hardware Considerations  
Programming Examples  
Bus Functions  
Message Protocols  
Interrupt and DMA Operation  
Servicing Requests  
iSBC Interface Considerations

## Peripheral Chips

## Intellec® Development System Workshop

### Course Description

- Comprehensive coverage of ROM based system monitor, ISIS-II Diskette Operating System, and CREDIT text editor
- Introduction to FORTRAN-80, LINK/LOCATE, system monitor debugging techniques, In-Circuit Emulation (ICE-85 used as an example), and Universal PROM Programmer
- System installation and checkout

### Attendees

- Programmer or design engineer who will be using Intellec Series II Development System to develop, debug, and integrate hardware/software
- Manager who wants to understand how this tool can help get a project completed faster

**Length:** 3 Days

**Tuition:** \$595  
\$475 (Group rate)

### Course Outline

**DAY 1**  
Introduction  
System Overview  
System Installation  
Introduction to ISIS-II  
Using CREDIT Text Editor

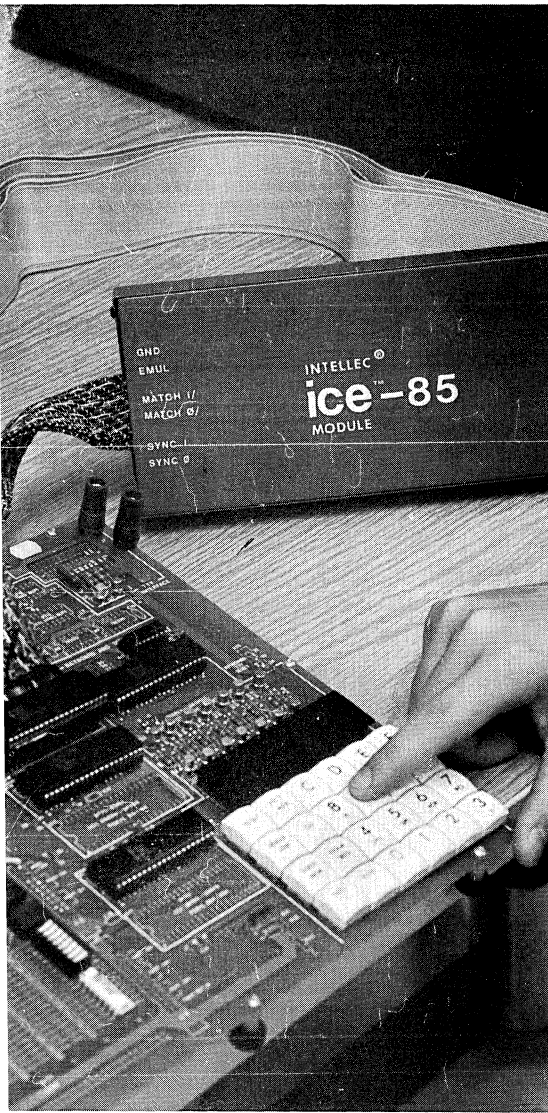
**DAY 2**  
Introduction to FORTRAN-80  
Compiling a Program  
Introduction to LINK/LOCATE  
System Monitor  
ICE-85

**DAY 3**  
ICE-85 (Continued)  
User Maintenance  
Programming Aids  
Universal PROM Programmer  
Course Review



**Development System**

## ICE-85™ Users Workshop



**Development System**

### Course Description

- Comprehensive coverage of In-Circuit Emulation using ICE-85
- Debugging of Assembly and PL/M Programs
- Extensive Hands-on Laboratory practice with instructor's guidance.

### Attendees

- Programmer or design engineer who will be using an ICE Product (ICE-48, ICE-41, or ICE-85) and who needs to develop expertise
- Prior programming experience with 8085, and use of a development system is assumed. Introduction to Microcomputer Workshop or equivalent is required

**Length:** 3 Days

**Tuition:** \$595  
\$475 (Group rate)

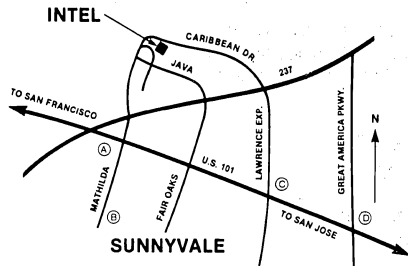
### Topics

ICE Concept  
Installation  
Use of ICE Manuals Documentation  
Command Language  
Memory Mapping  
Emulation Control with Breakpoints  
Trace Data Collection Modes  
Data Display Formats  
Debugging Strategies  
Multi-ICE  
Comparison of ICE-48, ICE-41, and ICE-85

## Training Center Information

### SAN FRANCISCO AREA

1350 Bordeaux Drive, Sunnyvale, CA 94086 408-734-8102



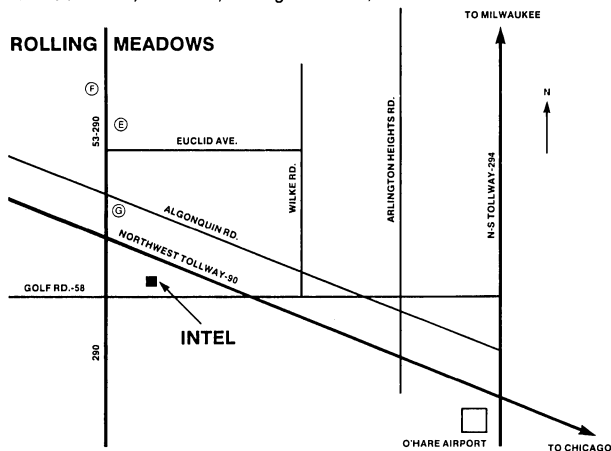
#### HOTELS near training center:

- Ⓐ Sundowner Sunnyvale, CA 408-734-9900
- Ⓑ Captains Cove Motel Sunnyvale, CA 408-735-7800
- Ⓒ Holiday Inn Sunnyvale, CA 408-245-5330 800-238-8000
- Ⓓ Marriott Hotel Santa Clara, CA 408-988-1500 800-228-9290



### CHICAGO AREA

Gould Center, East Tower  
2550 Golf Road, Suite 815, Rolling Meadows, IL 60008 312-981-7250

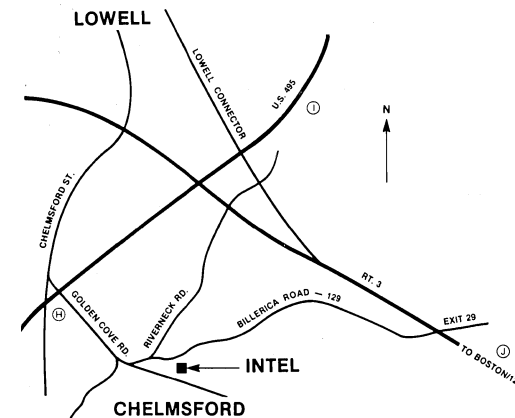


#### HOTELS near training center:

- Ⓔ Arlington Park Hilton Arlington Heights, IL 312-394-2000
- Ⓕ Howard Johnson's Palatine, IL 312-359-6900 800-654-2000
- Ⓖ Holiday Inn Rolling Meadows, IL 312-259-5000 800-238-8000

### BOSTON AREA

27 Industrial Avenue, Chelmsford, MA 01824 617-256-1374



#### HOTELS near training center:

- Ⓙ Howard Johnson's Chelmsford, MA 617-256-7511 800-654-2000
- Ⓚ Town House Motor Inn Lowell, MA 617-455-5606
- Ⓛ Howard Johnson Burlington, MA 617-272-6550 800-654-2000

### **Registration**

To enroll in a scheduled workshop, call the appropriate Intel Training Center between the

hours of 8-12 and 1-5, and ask for Customer Training.

Since enrollment is limited, registration 4-8 weeks in advance is recommended.

- (S) San Francisco (408) 734-8102 (also for other West Coast locations)
- (C) Chicago (312) 981-7250 (also for other Mid West locations)
- (B) Boston (617) 256-1374 (also for other East Coast locations)

### **Confirmation**

A confirmation letter will be sent to you specifying details of workshop time, location, directions, and lodging.

### **Payment**

**DUE BEFORE THE WORKSHOP BEGINS.** Tuition and schedule subject to change.

### **Group Rate Tuition Policy**

Tuition is at the group rate when an organization enrolls 3 or more people in the same course, one person in 3 or more courses, or some combination thereof. Groups are eligible for this discount only when enrollment and pre-payment are made together.







## INTEL PRODUCT SERVICE

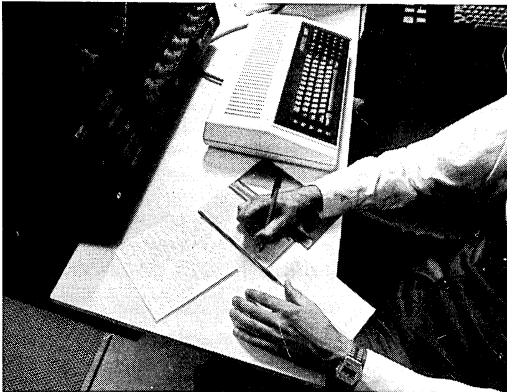
Today, it's essential to have dependable data processing and information storage equipment.

To insure trouble-free performance, every Intel product is engineered and manufactured to exacting standards. But sometimes, even the finest components may malfunction.

It's then that Intel Product Service delivers fast, economical, quality maintenance and service support to keep vital projects on schedule.

Intel is committed to providing a total service support package through a wide variety of service offerings available from Intel Product Service.

Intel Product Service takes pride in its capabilities and in the entire Intel systems product line, and is dedicated to total customer satisfaction.



### Customer Is Assured Maximum Equipment Performance

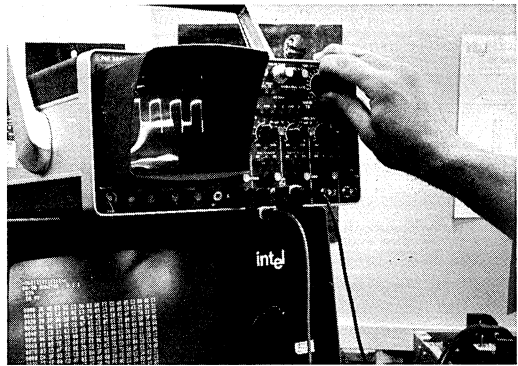
The Intel Product Service team is a skilled group capable of solving any problem, either by phone or on-site. All service needs (including emergency service and preventive maintenance) are handled quickly and efficiently by Intel's Customer Engineers, supported by district, regional and factory specialists.

This assures the customer of minimum downtime and maximum equipment performance.

### Immediate Availability of Service and Parts

Working from field service offices throughout the United States, Canada and abroad, Intel's Customer Engineers provide maintenance agreement customers with equipment installation and regularly scheduled preventive maintenance, including automatic installation of engineering changes as they occur.

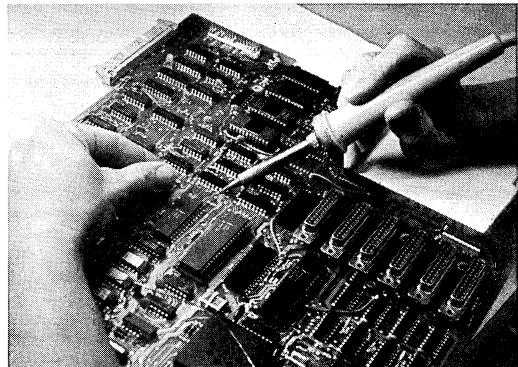
If there should be an equipment problem, customers contact the nearest Intel service location. If the customer location is within Intel's service area, an Intel Customer Engineer is dispatched to the site as quickly as possible. Once there, using advanced diagnostic tools and sophisticated test equipment, the Customer Engineer will determine the trouble and the replacement parts needed. With Intel's vast inventory of replacement parts, the right part is always available on the shortest notice, and the Customer Engineer is able to make repairs quickly and verify performance.



### Product Service Alternatives

Intel offers customers a number of service alternatives... all at reasonable prices and featuring the quality expected of the industry leader.

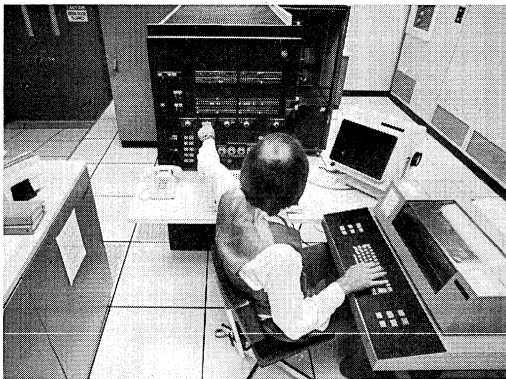
First, there's the Maintenance Agreement, which guarantees maintenance at the customer's site plus parts replacement and labor at no extra charge beyond an easy-to-budget, fixed monthly rate--all billed against one purchase order. The Maintenance Agreement features preventive maintenance which helps keep emergencies at a minimum. But should an emergency arise, a Customer Engineer is sent to the site immediately.



---

Of course Intel provides service for every system it sells. Thus, site service, parts replacement and repair are also available without a Maintenance Agreement. However, the site must be within a reasonable distance of an Intel field service location, and all charges are paid as they are incurred.

For customers not located in an Intel service area, or for those who prefer direct factory service, an Intel Direct Return Authorization enables the customer to return parts direct to Intel for repair, refurbishing and upgrade services by its factory experts. This service is offered with or without contract. (With contract, one purchase order covers all returns. Without contract, a separate purchase order is needed for each individual return.)



When distance from an Intel field location is too great, the Before Return Replacement contract is the logical solution for customers. To allow them to save valuable time and make repairs as soon as possible should trouble strike, a call via Intel's Hotline puts the customer in direct contact with support specialists who can determine which part is needed to repair the system. The part is then sent immediately--before the return of the old part to Intel. This service is also available with or without contract.

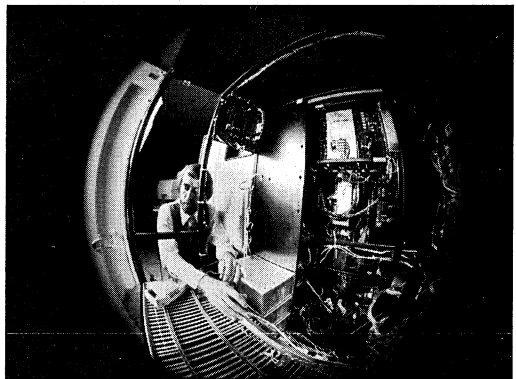


### Toll-Free Service Hotline

In the United States, Intel's toll-free service Hotline is a direct connection to service support specialists who can help immediately.

If on-site service is available, they direct the customer to the nearest Intel field service location. If not, Hotline specialists can usually help locate the trouble...and indicate how to make the necessary repairs.

Whichever service alternative is chosen, the customer is assured fast, affordable service. Intel offers customers the total service they need--that is one of the most important benefits of buying Intel products.





## U.S. AND CANADIAN SERVICE OFFICES

3065 Bowers Avenue  
Santa Clara, California 95051  
Tel: (408) 987-8080  
TWX: 910-338-0026  
TELEX: 34-6372

# 800-528-0595

## TOLL-FREE U.S. HOT-LINE NUMBER

For service in Arizona and Canada call (602)869-4600

### CALIFORNIA

Intel Corp.  
1601 Old Bayshore Hwy.  
Suite 345  
Burlingame 94010  
Tel: (415) 692-4762  
TWX: 910-375-3310

Intel Corp.  
2000 E. 4th Street  
Suite 110  
Santa Ana 92705  
Tel: (714) 835-2670  
TWX: 910-595-2475

Intel Corp.  
7670 Opportunity Road  
San Diego 92111  
Tel: (714) 268-3563

Intel Corp.  
3375 Scott Blvd.  
Santa Clara 95051  
Tel: (408) 987-8068

Intel Corp.  
5530 N. Corbin  
Tarzana 91356  
Tel: (213) 708-0333

### COLORADO

Intel Corp.  
650 South Cherry  
Suite 720  
Denver 80222  
Tel: (303) 321-8086  
TWX: 910-931-2289

### FLORIDA

Intel Corp.  
1001 N.W. 62nd Street  
Suite 406  
Ft. Lauderdale 33309  
Tel: (305) 771-0600  
TWX: 510-956-9407  
Intel Corp.  
5151 Adamson St.  
Orlando 32804  
Tel: (305) 628-2393

### GEORGIA

Intel Corp.  
3300 Holcombe Bridge Rd.  
Suite 225  
Norcross 30092  
Tel: (404) 449-0541

### ILLINOIS

Intel Corp.  
2550 Golf Road  
Suite 815  
Rolling Meadows 60008  
Tel: (312) 981-7230  
TWX: 910-253-1825

### KANSAS

Intel Corp.  
9393 W. 110th Street  
Suite 265  
Overland Park 66210  
Tel: (913) 642-8080

### MARYLAND

Intel Corp.  
7257 Parkway Drive  
Hanover 21076  
Tel: (301) 796-7500  
TWX: 710-862-1944

### MASSACHUSETTS

Intel Corp.  
27 Industrial Avenue  
Chelmsford 01824  
Tel: (617) 667-8126  
TWX: 710-343-6333

### MICHIGAN

Intel Corp.  
26500 Northwestern Hwy.  
Suite 401  
Southfield 48075  
Tel: (313) 353-0920  
TWX: 810-224-4915

### MINNESOTA

Intel Corp.  
7401 Metro Blvd.  
Suite 355  
Edina 55435  
Tel: (612) 835-6722  
TWX: 910-576-2867

### MISSOURI

Intel Corp.  
502 Earth City Plaza  
Suite 121  
Earth City 63045  
Tel: (314) 291-1990

### NEW JERSEY

Intel Corp.  
2460 Lemoine Avenue  
Ft. Lee 07024  
Tel: (201) 947-6267  
TWX: 710-991-8593

### OHIO

Intel Corp.  
Chagrin-Brainard Bldg. 300  
28001 Chagrin Blvd.  
Cleveland 44122  
Tel: (216) 464-2736  
TWX: 810-427-9298

Intel Corp.  
6500 Poe Avenue  
Dayton 45414  
Tel: (513) 890-5350  
TWX: 810-450-2528

### OREGON

Intel Corp.  
10700 S.W. Beaverton-Hillsdale Hwy.  
Suite 22  
Beaverton 97005  
Tel: (503) 641-8086  
TWX: 910-467-8741

### PENNSYLVANIA

Intel Corp.  
510 Pennsylvania Ave.  
3rd Floor  
Fort Washington 19034  
Tel: (215) 641-1000  
TWX: 510-661-2077

### TEXAS

Intel Corp.  
313 E. Anderson Lane  
Suite 314  
Austin 78752  
Tel: (512) 454-3628  
TWX: 910-874-1347

Intel Corp.  
2925 L.B.J. Freeway  
Suite 175  
Dallas 75234  
Tel: (214) 241-2820  
TWX: 910-860-5617

Intel Corp.  
6420 Richmond Avenue  
Suite 280  
Houston 77057  
Tel: (713) 784-1300  
TWX: 910-881-2490

### VIRGINIA

Intel Corp.  
7700 Leesburg Pike  
Suite 412  
Falls Church 22043  
Tel: (703) 734-9707  
TWX: 710-931-0625

### WASHINGTON

Intel Corp.  
1603 116th Ave. N.E.  
Suite 114  
Bellevue 98005  
Tel: (206) 232-7823  
TWX: 910-443-3002

### WISCONSIN

Intel Corp.  
150 S. Sunnyslope Road  
Suite 148  
Brookfield 53005  
Tel: (414) 784-9060

### CANADA

Intel Corp.  
50 Galaxy Blvd.  
Unit 12  
Rexdale, Ontario  
M9W4Y5  
Tel: (416) 675-2105  
Telex: 069-89278

Intel Corp.  
39 Hwy. 7, Bell Corners  
Ottawa, Ontario  
K2H 8R2  
Tel: (613) 829-9714  
Telex: 053-4115

This list of service offices is accurate as of the date of printing. However, new offices are being added as required.



## U.S. AND CANADIAN SALES OFFICES

3065 Bowers Avenue  
Santa Clara, California 95051  
Tel: (408) 987-8080  
TWX: 910-338-0026  
TELEX: 34-6372

### ALABAMA

Intel Corp.  
303 Williams Avenue, S.W.  
Suite 1422  
Huntsville 35801  
Tel: (205) 533-9353

### ARIZONA

Intel Corp.  
10210 N. 25th Avenue, Suite 11  
Phoenix 85021  
Tel: (602) 997-9695

### BFA

4426 North Saddle Bag Trail  
Scottsdale 85251  
Tel: (602) 994-5400

### CALIFORNIA

Intel Corp.  
7670 Opportunity Rd.  
Suite 135  
San Diego 92111  
Tel: (714) 268-3583

### Intel Corp.\*

2000 East 4th Street  
Suite 100  
Santa Ana 92705  
Tel: (714) 835-9642

### Intel Corp.\*

5530 Corbin Ave.  
Suite 120  
Tarzana 91356  
Tel: (213) 708-0333

### Intel Corp.\*

3375 Scott Blvd.  
Santa Clara 95051  
Tel: (408) 987-8086  
TWX: 910-339-9279  
910-338-0255

### Intel Corp.\*

Earle Associates, Inc.  
4617 Ruffner Street  
Suite 202  
San Diego 92111  
Tel: (714) 278-5441

### Mac-I

P.O. Box 1420  
Cupertino 95014  
Tel: (408) 257-9880

### Mac-I

558 Valley Way  
Calaveras Business Park  
Milpitas 95035  
Tel: (408) 946-8885

### Mac-I

P.O. Box 8763  
Fountain Valley 92708  
Tel: (714) 839-3341

### Mac-I

1321 Centinela Avenue  
Suite 1  
Santa Monica 90404  
Tel: (213) 829-4797

### Mac-I

29121 Ventura Blvd., Suite 240E  
Woodland Hills 91364  
Tel: (213) 347-5900

### COLORADO

Intel Corp.\*  
650 S. Cherry Street  
Suite 720  
Denver 80222  
Tel: (303) 331-8086  
TWX: 910-931-2289

### CONNECTICUT

Intel Corp.  
Pascok Alley  
36 Padanaram Road  
Danbury 06810  
Tel: (203) 792-8366  
TWX: 710-456-1199

### EMC Corp.

48 Purnell Place  
Manchester 06040  
Tel: (203) 646-8085

### FLORIDA

Intel Corp.  
1001 N.W. 62nd Street, Suite 406  
Ft. Lauderdale 33309  
Tel: (305) 771-0600  
TWX: 510-956-9407

### FLORIDA (cont.)

Intel Corp.  
5151 Adanson Street, Suite 203  
Orlando 32804  
Tel: (305) 628-2393  
TWX: 810-853-9219

### GEORGIA

Intel Corp.  
3300 Holcomb Bridge Rd.  
Norcross 30092  
Tel: (404) 449-0541

### ILLINOIS

Intel Corp.  
2550 Golf Road, Suite 815  
Rolling Meadows 60008  
Tel: (312) 981-7200  
TWX: 910-651-5881

### Technical Representatives

1502 North Linde Street  
Bloomington 81701  
Tel: (309) 829-8080

### INDIANA

Intel Corp.  
9101 Weesleyan Road  
Suite 204  
Indianapolis 46268  
Tel: (317) 299-0623

### IOWA

Technical Representatives, Inc.  
St. Andrews Building  
1930 St. Andrews Drive N.E.  
Cedar Rapids 52405  
Tel: (319) 393-5510

### KANSAS

Intel Corp.  
9393 W. 110th St., Ste. 265  
Overland Park 66210  
Tel: (913) 642-8080

### Technical Representatives, Inc.

8245 Nieman Road, Suite 100  
Lenexa 66214  
Tel: (913) 888-0212, 3 & 4  
TWX: 910-749-6412

### Technical Representatives, Inc.

360 N. Rock Road  
Suite 4  
Wichita 67206  
Tel: (316) 681-0242

### MARYLAND

Intel Corp.\*  
7257 Parkway Drive  
Hanover 21076  
Tel: (301) 796-7500  
TWX: 710-862-1944

### Mesa Inc.

16201 Industrial Dr.  
Gaithersburg 20760  
Tel: (301) 948-4350

### MASSACHUSETTS

Intel Corp.\*  
27 Industrial Ave.  
Chelmsford 01824  
Tel: (617) 867-8126  
TWX: 710-343-6333

### EMC Corp.

381 Elliot Street  
Newton 02459  
Tel: (617) 244-4740  
TWX: 922531

### MICHIGAN

Intel Corp.\*  
26500 Northwestern Hwy.  
Suite 401  
Southfield 48075  
Tel: (313) 353-0920  
TWX: 810-244-4915

### MINNESOTA

Intel Corp.  
7401 Metro Blvd.  
Suite 355  
Edina 55435  
Tel: (612) 835-6722  
TWX: 910-576-2867

### MISSOURI

Intel Corp.  
506 Earth City Plaza  
Suite 121  
Earth City 60345  
Tel: (314) 291-1990

### Technical Representatives, Inc.\*

502 Earth City Plaza  
Suite 201  
Earth City 63045  
Tel: (314) 291-0001

### Technical Representatives, Inc.\*

VSW Bldg., Suite 560  
408 W. 31st Street  
Kansas City 64111  
Tel: (816) 756-3575  
TWX: 910-771-0025

### NEW JERSEY

Intel Corp.\*  
Raritan Plaza  
2nd Floor  
Raritan Center  
Edison 08817  
Tel: (201) 225-3000  
TWX: 710-480-6238

### NEW MEXICO

BFA Corporation  
1704 Moon N.E., Suite 7  
Las Cruces 87112  
Tel: (505) 523-0601  
TWX: 910-983-0543

### BFA Corporation

3705 Westfield, N.E.  
Albuquerque 87111  
Tel: (505) 292-1212  
TWX: 910-989-1157

### NEW YORK

Intel Corp.\*  
300 Motor Pkwy.  
Hauppauge 11787  
Tel: (516) 231-3300  
TWX: 510-227-6236

### Intel Corp.

80 Washington St.  
Poughkeepsie 12601  
Tel: (914) 473-2303  
TWX: 510-248-0080

### Intel Corp.\*

2255 Lyell Avenue  
Lower Floor East Suite  
Rochester 14606  
Tel: (716) 254-9120  
TWX: 510-253-7391

### Measurement Technology, Inc.

159 Northern Boulevard  
Great Neck 11021  
Tel: (516) 482-3500

### T-Squared

4054 Newcourt Avenue  
Syracuse 13206  
Tel: (315) 463-8592  
TWX: 710-541-0554

### T-Squared

2 E. Main  
Victor 14564  
Tel: (716) 924-9101  
TWX: 510-254-8542

### NORTH CAROLINA

Intel Corp.  
154 Hulfman Mill Rd.  
Burlington 27215  
Tel: (919) 584-3631

### OHIO

Intel Corp.\*  
6500 Poe Avenue  
Dayton 45415  
Tel: (513) 890-5350  
TWX: 810-450-2528

### Intel Corp.\*

Chagrin-Brainard Bldg., No. 300  
28001 Chagrin Blvd.  
Cleveland 44122  
Tel: (216) 464-2736  
TWX: 810-427-9298

### OREGON

Intel Corp.  
10700 S.W. Beaverton  
Hillsdale Highway  
Suite 324  
Beaverton 97005  
Tel: (503) 641-8086  
TWX: 910-467-8741

### PENNSYLVANIA

Intel Corp.\*  
510 Pennsylvania Avenue  
Fort Washington 19034  
Tel: (215) 641-1000  
TWX: 510-661-2077

### Intel Corp.\*

201 Penn Center Boulevard  
Suite 301W  
Pittsburgh 15235  
Tel: (412) 823-4970

### O.E.D. Electronics

300 N. York Road  
Halboro 19040  
Tel: (215) 674-9600

### TEXAS

Intel Corp.\*  
2925 L.B.J. Freeway  
Suite 175  
Dallas 75234  
Tel: (214) 241-9521  
TWX: 910-860-5617

### Intel Corp.\*

6420 Richmond Ave.  
Suite 280  
Houston 77057  
Tel: (713) 784-3400  
TWX: 910-881-2490

### Industrial Digital Systems Corp.

5925 Sovereign  
Suite 101  
Houston 77036  
Tel: (713) 988-9421

### Intel Corp.

313 E. Anderson Lane  
Suite 314  
Austin 78752  
Tel: (512) 454-3628

### WASHINGTON

Intel Corp.  
Suite 114, Bldg. 3  
1603 116th Ave. N.E.  
Bellevue 98005  
Tel: (206) 453-8086  
TWX: 910-443-3002

### WISCONSIN

Intel Corp.  
150 S. Sunnyslope Rd.  
Brookfield 53005  
Tel: (414) 784-9060

### CANADA

Intel Semiconductor Corp.\*  
Suite 233, Bell Mews  
39 Highway 7, Bells Corners  
Ottawa, Ontario K2H 6R2  
Tel: (613) 829-9714  
TELEX: 053-4115

### Intel Semiconductor Corp.

50 Galaxy Blvd.  
Unit 12  
Rexdale, Ontario  
M9W 4V5  
Tel: (416) 675-2105  
TELEX: 06983574

### Multitek, Inc.\*

15 Grenfell Crescent  
Ottawa, Ontario K2G 0G3  
Tel: (613) 228-2365  
TELEX: 055-4585

### Multitek, Inc.\*

Toronto  
Tel: (416) 245-4622

### Multitek, Inc.

Montreal  
Tel: (514) 481-1350

\*Field Application Location



## U.S. AND CANADIAN DISTRIBUTORS

3065 Bowers Avenue  
Santa Clara, California 95051  
Tel: (408) 987-8080  
TWX: 910-338-0026  
TELEX: 34-6372

### ALABAMA

†Hamilton/Avnet Electronics  
4812 Commercial Drive N.W.  
Huntsville 35805  
Tel: (205) 837-7210  
†Pioneer/Huntsville  
3207 Putman Drive N.W.  
Huntsville 35805  
Tel: (205) 837-9033  
TWX: 810-726-2197

### ARIZONA

†Hamilton/Avnet Electronics  
2615 S. 21st Street  
Phoenix 85034  
Tel: (602) 275-7851  
†Wyle Distribution Group  
8165 N. 24th Avenue  
Phoenix 85021  
Tel: (602) 995-9185  
TWX: 910-951-4282

### CALIFORNIA

Arrow Electronics, Inc.  
9511 Ridge Haven Court  
San Diego 92123  
Tel: (714) 565-4800  
Arrow Electronics, Inc.  
720 Palomar Avenue  
Sunnyvale California 94086  
Tel: (408) 739-3011  
TWX: 910-339-9371

†Avnet Electronics  
350 McCormick Avenue  
Costa Mesa 92626  
Tel: (714) 754-6051  
TWX: 910-595-1928

Hamilton/Avnet Electronics  
1175 Bordeaux Dr.  
Sunnyvale 94086  
Tel: (408) 743-3300  
TWX: 910-339-9332

†Hamilton/Avnet Electronics  
8917 Complex Drive  
San Diego 92123  
Tel: (714) 571-7923  
TWX: 910-335-1216

†Hamilton/Avnet Electronics  
10912 W. Washington Blvd.  
Culver City 90230  
Tel: (213) 558-2193  
TWX: 910-340-6364 or 7073

†Hamilton Electro Sales  
3170 Pullman Street  
Costa Mesa 92626  
Tel: (714) 641-4100  
TWX: 910-595-2638

†Wyle Distribution Group  
124 Maryland Street  
El Segundo 90245  
Tel: (213) 322-3826  
TWX: 910-348-7140 or 7111

†Wyle Distribution Group  
9525 Chesapeake Dr.  
San Diego 92123  
Tel: (714) 565-9171  
TWX: 910-335-1590

†Wyle Distribution Group  
3000 Bowers Avenue  
Santa Clara 95052  
Tel: (408) 727-2500  
TWX: 910-338-0451 or 0296

Wyle Distribution Group  
17872 Cowan Avenue  
Inyone 92714  
Tel: (714) 641-1611

### COLORADO

†Wyle Distribution Group  
6777 E. 50th Avenue  
Commerce City 80022  
Tel: (303) 287-9611  
TWX: 910-931-0510

### COLORADO (Cont.)

†Hamilton/Avnet Electronics  
8765 E. Orchard Road  
Suite 708  
Englewood 80111  
Tel: (303) 534-1212  
TWX: 910-931-0510  
Wyle Distribution Group  
451 E. 124th Avenue  
Thornton 80241  
Tel: (303) 457-WYLE  
TWX: 910-931-0510

### CONNECTICUT

†Arrow Electronics  
12 Beaumont Road  
Wallingford 06512  
Tel: (203) 265-7741  
TWX: 710-476-0162  
†Hamilton/Avnet Electronics  
Commerce Drive  
Danbury 06810  
Tel: (203) 797-2800  
TWX: 710-456-9974

†Harvey Electronics  
112 Main Street  
Norwalk 06851  
Tel: (203) 853-1515  
TWX: 710-468-3373  
TWX: 710-393-8770

†Arrow Electronics  
1001 N.W. 82nd Street  
Suite 109  
Ft. Lauderdale 33309  
Tel: (305) 776-7790  
TWX: 510-955-9456  
†Arrow Electronics  
115 Palm Bay Road, N.W.  
Suite 10, Bldg. 200  
Palm Bay 32905  
Tel: (305) 725-1480  
TWX: 510-959-6337

### FLORIDA

†Arrow Electronics  
6800 Northwest 20th Ave.  
Ft. Lauderdale 33309  
Tel: (305) 971-2900  
TWX: 510-955-3097

†Hamilton/Avnet Electronics  
3197 Tech. Drive North  
St. Petersburg 33702  
Tel: (813) 576-3930  
TWX: 810-863-0374

†Pioneer/Orlando  
6220 S. Orange Blossom Trail  
Suite 412  
Orlando 32809  
Tel: (305) 859-3600  
TWX: 810-850-0177

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

### INDIANA

†Hamilton/Avnet Electronics  
485 Gradle Drive  
Carmel 46032  
Tel: (317) 844-9333  
Pioneer/Indiana  
8408 Castleplace Drive  
Indianapolis 46250  
Tel: (317) 849-7300  
TWX: 810-260-1794

### KANSAS

†Hamilton/Avnet Electronics  
9219 Quivira Road  
Overland Park 66215  
Tel: (913) 868-8900  
†Component Specialties, Inc.  
8369 Nieman Road  
Lenexa 66214  
Tel: (913) 492-3555

### MARYLAND

Arrow Electronics, Inc.  
4801 Benson Avenue  
Baltimore 21227  
Tel: (301) 247-5200  
†Hamilton/Avnet Electronics  
7235 Standard Drive  
Hanover 21076  
Tel: (301) 796-5684  
TWX: 710-862-1861

†Pioneer/Washington  
9100 Gaither Road  
Gaithersburg 20750  
Tel: (301) 946-0710  
TWX: 710-828-0545

### MASSACHUSETTS

†Hamilton/Avnet Electronics  
50 Tower Office Park  
Woburn 01801  
Tel: (617) 273-7500  
TWX: 710-393-0382

†Arrow Electronics  
96D Commerce Way  
Woburn 01801  
Tel: (617) 933-8130  
Harvey/Boston  
4 Hartwell Ave.  
Lexington 02173  
Tel: (617) 861-9200  
TWX: 710-326-6617

†Hamilton/Avnet Electronics  
3197 Tech. Drive North  
St. Petersburg 33702  
Tel: (813) 576-3930  
TWX: 810-863-0374

†Pioneer/Orlando  
6220 S. Orange Blossom Trail  
Suite 412  
Orlando 32809  
Tel: (305) 859-3600  
TWX: 810-850-0177

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

†Hamilton/Avnet Electronics  
8700 I-85 Access Road, No. 11  
Suite 1E  
Norcross 30071  
Tel: (404) 448-0800

### MISSOURI

†Hamilton/Avnet Electronics  
13743 Shoreline Ct.  
Earth City 63045  
Tel: (314) 344-1200  
TWX: 910-762-0606

### NEW HAMPSHIRE

†Arrow Electronics  
1 Perimeter Drive  
Manchester 03103  
Tel: (603) 668-6968  
TWX: 710-220-1684

### NEW JERSEY

†Arrow Electronics  
Pleasant Valley Avenue  
Moorestown 08057  
Tel: (215) 929-1800  
TWX: 710-897-0829  
†Arrow Electronics  
285 Midland Avenue  
Saddle Brook 07662  
Tel: (201) 797-5800  
TWX: 710-998-2206

†Hamilton/Avnet Electronics  
1 Keystone Ave.  
Bldg. 36  
Cherry Hill 08003  
Tel: (609) 424-0100  
TWX: 710-897-1405

†Hamilton/Avnet Electronics  
10 Industrial Road  
Fairfield 07006  
Tel: (201) 575-3390  
TWX: 710-734-4438

†Arrow Electronics  
45 Route 46  
Pinetree 07058  
Tel: (201) 227-1262  
TWX: 710-734-4382

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500

†Alliance Electronics Inc.  
11030 Cochiti S.E.  
Albuquerque 87123  
Tel: (505) 292-3360  
TWX: 910-989-1151

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500

†Hamilton/Avnet Electronics  
2524 Baylor Drive S.E.  
Albuquerque 87119  
Tel: (505) 765-1500



## U.S. AND CANADIAN DISTRIBUTORS

3065 Bowers Avenue  
Santa Clara, California 95051  
Tel: (408) 987-8080  
TWX: 910-338-0026  
TELEX: 34-6372

### NEW YORK (Cont.)

Harvey Electronics  
P.O. Box 1208  
Binghamton 13902  
Tel: (607) 749-8211  
TWX: 510-252-0893

†Harvey Electronics  
60 Crossways Park West  
Woodbury 11797  
Tel: (516) 921-8700  
TWX: 510-221-2184

Harvey/Rochester  
840 Fairport Park  
Fairport 14450  
Tel: (716) 381-7070  
TWX: 510-253-7001

### NORTH CAROLINA

Arrow Electronics  
938 Burke Street  
Winston-Salem 27102  
Tel: (919) 725-8711  
TWX: 510-922-4765

†Hamilton/Avnet Electronics  
2803 Industrial Drive  
Raleigh 27609  
Tel: (919) 829-8030  
Pioneer/Carolina  
106 Industrial Ave.  
Greensboro 27406  
Tel: (919) 273-4441  
TWX: 510-925-1114

### OHIO

Arrow Electronics  
7620 McEwen Road  
Centerville 45459  
Tel: (513) 435-5563  
TWX: 810-459-1611

Arrow Electronics  
6238 Cochran Rd.  
Solon 44139  
Tel: (216) 248-3990  
TWX: 810-427-9409

Arrow Electronics  
10 Knollcrest Dr.  
Cincinnati 45237  
Tel: (513) 761-5432  
TWX: 810-461-2670

†Hamilton/Avnet Electronics  
954 Senate Drive  
Dayton 45459  
Tel: (513) 433-0610  
TWX: 910-340-2531

†Hamilton/Avnet Electronics  
4588 Emery Industrial Parkway  
Warrensville Heights 44128  
Tel: (216) 831-3500  
TWX: 810-427-9452

†Pioneer/Dayton  
1900 Troy Street  
Dayton 45404  
Tel: (513) 236-9900  
TWX: 810-459-1622

†Pioneer/Cleveland  
4800 E. 131st Street  
Cleveland 44105  
Tel: (216) 587-3600  
TWX: 810-422-2210

### OKLAHOMA

†Components Specialties, Inc.  
7920 E. 40th Street  
Tulsa 74145  
Tel: (918) 664-2820  
TWX: 910-845-2215

### OREGON

†Alamac/Strom Electronics  
8022 S.W. Nimbus Bldg. 7  
Beaverton 97005  
Tel: (503) 641-9070

†Hamilton/Avnet Electronics  
6024 S.W. Jean Rd.  
Bldg. C, Suite 10  
Lake Oswego 97034  
Tel: (503) 635-7848

### PENNSYLVANIA

†Arrow Electronics  
4297 Greensburg Pike  
Suite 3114  
Pittsburgh 15221  
Tel: (412) 351-4000

Pioneer/Pittsburgh  
259 Kappa Drive  
Pittsburgh 15238  
Tel: (412) 782-2300  
TWX: 710-795-3122

Pioneer/Delaware Valley  
261 Gibraltar Road  
Horsham 19044  
Tel: (215) 674-4000  
TWX: 510-665-6778

### TEXAS

Arrow Electronics  
13715 Gamma Road  
Dallas 75234  
Tel: (214) 386-7500  
TWX: 910-861-5495

Arrow Electronics, Inc.  
10700 Corporate Drive, Suite 100  
Stafford 77477  
Tel: (713) 491-4100

Component Specialties, Inc.  
8222 Jamestown Drive  
Suite 115  
Austin 78758  
Tel: (512) 837-8922  
TWX: 910-874-1320

†Component Specialties, Inc.  
10807 Shady Trail, Suite 101  
Dallas 75220  
Tel: (214) 357-6511  
TWX: 910-861-4999

†Component Specialties, Inc.  
8585 Commerce Park Drive, Suite 590  
Houston 77036  
Tel: (713) 771-7237  
TWX: 910-881-2422

Hamilton/Avnet Electronics  
10508A Boyer Blvd.  
Austin 78757  
Tel: (512) 837-8911

†Hamilton/Avnet Electronics  
2111 W. Walnut Hill Lane  
Living 75062  
Tel: (214) 661-4111  
TWX: 910-860-5371

†Hamilton/Avnet Electronics  
3939 Ann Arbor Drive  
Houston 77063  
Tel: (713) 780-1771

### UTAH

†Hamilton/Avnet Electronics  
1585 West 2100 South  
Salt Lake City 84119  
Tel: (801) 972-2600

### WASHINGTON

†Alamac/Strom Electronics  
5811 Sixth Ave. South  
Seattle 98108  
Tel: (206) 763-2300  
TWX: 910-444-2067

Arrow Electronics, Inc.  
Electronics Distribution Division  
1059 Andover Park East  
Tukwila 98188  
Tel: (206) 575-0907

†Hamilton/Avnet Electronics  
14212 N.E. 21st Street  
Bellevue 98005  
Tel: (206) 453-5844

### WASHINGTON (Cont.)

†Wyle Distribution Group  
1750 132nd Avenue N.E.  
Bellevue 98005  
Tel: (206) 453-8300  
TWX: 910-443-2526

### WISCONSIN

†Arrow Electronics  
430 W. Rawson Avenue  
Oak Creek 53154  
Tel: (414) 764-6600  
TWX: 910-338-0026

†Hamilton/Avnet Electronics  
2975 Moorland Road  
New Berlin 53151  
Tel: (414) 784-4510  
TWX: 910-262-1182

### CANADA

#### ALBERTA

L.A. Varah Ltd.  
4742 14th Street N.E.  
Calgary T2D 6L7  
Tel: (403) 230-1235  
TWX: 018-258-97

Zentronics  
9224 27th Avenue  
Edmonton T6N 1B2  
Tel: (403) 463-3014

Zentronics  
3651 21st N.E.  
Calgary T2E 6T5  
Tel: (403) 230-1422

#### BRITISH COLUMBIA

L.A. Varah Ltd.  
2077 Alberta Street  
Vancouver V5Y 1C4  
Tel: (604) 873-3211  
TWX: 610-929-1068

Zentronics  
550 Cambie St.  
Vancouver V6B 2N7  
Tel: (604) 688-2533  
TWX: 04-50777-89

#### MANITOBA

L.A. Varah  
1-1832 King Edward Street  
Winnipeg R2R 0N1  
Tel: (204) 833-6190  
TWX: 07-55-365

Zentronics  
590 Berry St.  
Winnipeg R3H 0S1  
Tel: (204) 775-8661

#### ONTARIO

†Hamilton/Avnet Electronics  
3688 Rexwood Road, Units G & H  
Mississauga L4V 1M5  
Tel: (416) 877-4732  
TWX: 610-492-8860

†Hamilton/Avnet Electronics  
1735 Courtwood Crescent  
Ottawa K2C 3J2  
Tel: (613) 226-1700  
TWX: 053-4971

L.A. Varah, Ltd.  
505 Kenora Avenue  
Hamilton L8E 3P2  
Tel: (416) 561-9311  
TWX: 061-9349

Zentronics  
141 Catherine Street  
Ottawa K2P 1C3  
Tel: (613) 238-6411  
TWX: 053-3636

Zentronics  
1355 Meyerside Drive  
Mississauga, Ontario L5T 1C9  
Tel: (416) 676-9000  
Telex: 06-983-657

### QUEBEC

†Hamilton/Avnet Electronics  
2670 Sabourin Street  
St. Laurent H4S 1M2  
Tel: (514) 331-6443  
TWX: 610-421-3731

Zentronics  
5010 Pars Street  
Montreal H4P 1P3  
Tel: (514) 735-6361  
TWX: 05-827-535



# INTERNATIONAL SALES AND MARKETING OFFICES

3065 Bowers Avenue  
Santa Clara, California 95051  
Tel: (408) 987-8080  
TWX: 910-338-0026  
TELEX: 34-6372

## INTERNATIONAL DISTRIBUTORS/REPRESENTATIVES

### ARGENTINA

Micro Sistemas S. A.  
9 De Julio 561  
Cordoba  
Tel: 54-51-32-880  
TELEX: 51837 BICCO

### AUSTRALIA

A. J. F. Systems & Components Pty. Ltd.  
310 Queen Street  
Melbourne  
Victoria 3000  
Tel: 679-702  
TELEX: AA 31261  
Warburton Franki  
Corporate Headquarters  
372 Eastern Valley Way  
Chatswood, New South Wales 2067  
Tel: 407-3261  
TELEX: AA 21299

### AUSTRIA

Bacher Elektronische Gerate GmbH  
Rotenmugasse 26  
A 1120 Vienna  
Tel: (0222) 83 63 96  
TELEX: (01) 1532  
Rekirach Elektronik Gerate GmbH  
Lichtensteinalstrasse 97  
A 1000 Vienna  
Tel: (222) 347646  
TELEX: 74759

### BELGIUM

Inelco Belgium S.A.  
Ave. des Croix de Guerre 94  
B 1120 Brussels  
Tel: (02) 216 01 60  
TELEX: 25441

### BRAZIL

Icotron S.A.  
0911 Av. Mutinga 3650  
6 Andar  
Pirituba Sao Paulo  
Tel: 261-0211  
TELEX: (011) 222 ICO BR

### CHILE

#### DIN

Av. Vic. Mc Kenna 204  
Casilla 6055  
Santiago  
Tel: 227 564  
TELEX: 3520003

### CHINA

C.M. Technologies  
525 University Avenue  
Suite A-40  
Palo Alto, CA 94301  
Tel: (415) 326-9150

### COLOMBIA

International Computer Machines  
Carrera 7 No. 72-34  
Apdo. Aereo 19403  
Bogota 1  
Tel: 211-7282  
TELEX: 41314 INCO

### CYPRUS

Cyprus Eltrom Electronics  
P.O. Box 5393  
Nicosia  
Tel: 21-27982

### DENMARK

STL-Lyngso Komponent A/S  
Ostmarken 4  
DK-2860 Soborg  
Tel: (01) 87 00 77  
TELEX: 22990  
Scandinavian Semiconductor  
Supply A/S  
Nannasgade 18  
DK-2200 Copenhagen  
Tel: (01) 83 50 90  
TELEX: 19037

### FINLAND

Oy Fintronic AB  
Melkonkatu 24 A  
SF-00210  
Helsinki 21  
Tel: 0-692 6022  
TELEX: 124 224 Ftron SF

### FRANCE

Celdis S.A.\*  
53, Rue Charles Freret  
F-94250 Gentilly  
Tel: (1) 581 00 20  
TELEX: 200 485  
Foutrier  
Rue des Trois Glorieuses  
F-42270 St. Priest-en-Jarez  
Tel: (77) 74 87 33  
TELEX: 300 0 21

### Metrologie\*

La Tour d'Asnieres  
4, Avenue Laurent Cely  
92606-Asnieres  
Tel: 791 44 44  
TELEX: 611 449  
Tekelco Airtronic\*  
Cite des Bruyeres  
Rue Carle Vernet  
F-92310 Sevres  
Tel: (1) 534 75 35  
TELEX: 204552

### GERMANY

Electronic 2000 Vertriebs GmbH  
Neumarkter Strasse 75  
D-8000 Munich 80  
Tel: (089) 434061  
TELEX: 522561  
Jermyn GmbH  
Postfach 1180  
D-60777 Camberg  
Tel: (06434) 231  
TELEX: 484426

### Kontron Elektronik GmbH

Breslauerstrasse 2  
8057 Eching B  
D-8000 Munich  
Tel: (89) 319 011  
TELEX: 522122  
Neye Enatechnik GmbH  
Schillerstrasse 14  
D-2085 Quickborn-Hamburg  
Tel: (04106) 6121  
TELEX: 02-13590

### GREECE

American Technical Enterprises  
P.O. Box 156  
Athens  
Tel: 30-1-8811271  
30-1-8219470

### HONG KONG

Schmidt & Co.  
28/F Wing on Center  
Connaught Road  
Hong Kong  
Tel: 5-455-644  
TELEX: 74766 Schmc Hx

### INDIA

Micronic Devices  
104/109C, Nirmal Industrial Estate  
Sion (E)  
Bombay 400022, India  
Tel: 486-170  
TELEX: 011-5947 MDEV IN

### ISRAEL

Electronics Ltd.\*  
11 Rozanis Street  
P.O. Box 39300  
Tel Aviv 61390  
Tel: 475151  
TELEX: 33638

### ITALY

Eledra 35 S.P.A.\*  
Viale Elvezia, 18  
I 20154 Milan  
Tel: (02) 34.93.041-31.85.441  
TELEX: 332332

### JAPAN

Asahi Electronics Co. Ltd.  
KMM Bldg. Room 407  
2-14-1 Asano, Kokura  
Kita-Ku, Kitakyushu City 802  
Tel: (093) 511-6471  
TELEX: AECKY 7126-16  
Hamilton-Avnet Electronics Japan Ltd.  
YU and YOU Bldg. 1-4 Horidome-Cho  
Ninonbashi  
Tel: (03) 662-9911  
TELEX: 2523774  
Ryoyo Electric Corp.  
Konwa Bldg.  
1-12-22, Tsukiji, 1-Chrome  
Chuo-Ku, Tokyo 104  
Tel: (03) 543-7711  
Tokyo Electron Ltd.  
No. 1 Higashikata-Machi  
Midori-Ku, Yokohama 226  
Tel: (045) 471-8811  
TELEX: 781-4473

### KOREA

Koram Digital  
Room 909 Woonam Bldg.  
7, 1KA Bongre-Dong  
Chung-Ku Seoul  
Tel: 238-123  
TELEX: K23542 HANSINT  
Lewwood International, Inc.  
C.P.O. Box 4046  
112-25 Sokong-Dong  
Chung-Ku, Seoul 100  
Tel: 285-927  
TELEX: MOCNDM K23231  
CABLE: "LEEWOOD" Seoul

### MEXICO

Proveedora Electronica, S.A. (Proesa)  
Frol. Mactezuma Ote. 24  
Col. Romero de Terreros  
Apdo. Postal 21-139  
Mexico 21, D.F.  
Tel: 554-8300  
TELEX: 017-72402 SAULME

### NETHERLANDS

Inelco Nether. Comp. Sys. BV  
Turfstekersstraat 63  
Aalsmeer 1431 D  
Tel: (2977) 26855  
TELEX: 14693  
Koning & Hartman  
Koperwerf 30  
2544 EN Den Haag  
Tel: (70) 210-11  
TELEX: 31528

### NEW ZEALAND

W. K. McLean Ltd.  
P.O. Box 18-065  
Glenn Innes, Auckland, 6  
Tel: 587-037  
TELEX: NZ2763 KOSFY

### NORWAY

Nordisk Elektronik (Norge) A/S  
Postoffice Box 122  
Smedsvingen 4  
1364 Hvalstad  
Tel: 02 78 62 10  
TELEX: 17546

### PORTUGAL

Ditram  
Componentes E Electronica LDA  
Av. Miguel Bombarda, 133  
Lisboa  
Tel: (19) 545313  
TELEX: 14347 GESPIC

### SINGAPORE

General Engineers Associates  
Blk 3, 1003-1008, 10th Floor  
P.S.A. Multi-Storey Complex  
Telok Blangah/Pasir Panjang  
Singapore 5  
Tel: 271-3163  
TELEX: RS23987 GENERCO

### SOUTH AFRICA

Electronic Building Elements  
Pine Square  
18th Street  
Hazelwood, Pretoria 0001  
Tel: 011-27-12-46-9221  
TELEX: 30181SA

### SPAIN

Interface S.A.  
Ronda San Pedro 22, 3\*  
Barcelona 10  
Tel: 301 78 51  
TWX: 51508  
ITT SESA  
Miguel Angel 16  
Madrid 10  
Tel: (1) 4190957  
TELEX: 27707/27461

### SWEDEN

AB Gosta Backstrom  
Box 12009  
10221 Stockholm  
Tel: (08) 541 080  
TELEX: 10135

Nordisk Elektronik AB  
Box 27301  
S-10254 Stockholm  
Tel: (08) 835040  
TELEX: 10547

### SWITZERLAND

Industrade AG  
Gemsenstrasse 2  
10221 Stockholm  
Tel: (08) 541 080  
CH-8021 Zurich  
Tel: (01) 60 22 30  
TELEX: 56788

### TAIWAN

Taiwan Automation Co.\*  
3d Floor #75, Section 4  
Nanking East Road  
Taipei  
Tel: 771-0940  
TELEX: 11942 TAIAUTO

### TURKEY

Turkelek Electronics  
Apapurk Boulevard 169  
Ankara  
Tel: 189483

### UNITED KINGDOM

Comway Microsystems Ltd.  
Market Street  
68-Barknell, Berkshire  
Tel: (344) 51654  
TELEX: 847201  
G.E.C. Semiconductors Ltd.  
East Lane  
North Wembley  
Middlesex HA9 7PP  
Tel: (01) 904-9303/908-4111  
TELEX: 28817

Jermyn Industries  
Vestry Estate  
Sevenoaks, Kent  
Tel: (0732) 501.44  
TELEX: 95142

Rapid Recall, Ltd.  
6 Soho Mills Ind. Park  
Wooburn Green  
Bucks, England  
Tel: (6285) 24961  
TELEX: 849439

Sintron Electronics Ltd.\*  
Arkwright Road 2  
Reading, Berkshire RG2 0LS  
Tel: (0734) 85464  
TELEX: 847395

### VENEZUELA

Componentes y Circuitos  
Electronicos TTLCA C.A.  
Apartado 3223  
Caracas 101  
Tel: 718-100  
TELEX: 21795 TELETIPOS

\*Field Application Location





## INTERNATIONAL SALES AND MARKETING OFFICES

3065 Bowers Avenue  
Santa Clara, California 95051  
Tel: (408) 987-8080  
TWX: 9 10-338-0026  
TELEX: 34-6372

### INTEL® MARKETING OFFICES

#### AUSTRALIA

Intel Semiconductor Pty. Ltd.  
Suite 2, Level 15, North Point  
100 Miller Street  
North Sydney, NSW, 2060  
Tel: 450-847  
TELEX: AA 20097

#### BELGIUM

Intel Corporation S.A.  
Rue du Moulin a Papier 51  
Boite 1  
B-1160 Brussels  
Tel: (02) 660 30 10  
TELEX: 24814

#### DENMARK

Intel Denmark A/S\*  
Lynghbvej 32 2nd Floor  
DK-2100 Copenhagen East  
Tel: (01) 18 20 00  
TELEX: 19567

#### FINLAND

Intel Finland OY  
Sentrarikuja 3  
SF-00400 Helsinki 40  
Tel: (0) 556531  
TELEX: 123 332

#### FRANCE

Intel Corporation, S.A.R.L.\*  
5 Place de la Balance  
Silic 223  
94528 Rungis Cedex  
Tel: (01) 687 22 21  
TELEX: 270475

#### GERMANY

Intel Semiconductor GmbH\*  
Seidstrasse 27  
8000 Muenchen 2  
Tel: (089) 53 891  
TELEX: 523 177

Intel Semiconductor GmbH  
Mainzer Strasse 75  
6230 Wiesbaden 1  
Tel: (06121) 700674  
TELEX: 04186183

Intel Semiconductor GmbH  
Wernerstrasse 67  
P.O. Box 1460  
7012 Fellbach  
Tel: (0711) 580082  
TELEX: 7254826

Intel Semiconductor GmbH  
Hohenzollern Strasse 5  
3000 Hannover 1  
Tel: (0511) 327081  
TELEX: 923625

Intel Semiconductor GmbH  
Oberrathstrasse 2  
4000 Duesseldorf 30  
Tel: (0211) 851054-6  
TELEX: 6596977

#### HONG KONG

Intel Semiconductor Ltd.  
99-105 Des Voeux Rd., Central  
18F, Unit B  
Hong Kong  
Tel: 5-450-847  
TELEX: 63889

#### ISRAEL

Intel Semiconductor Ltd.\*  
P.O. Box 2404  
Haifa  
Tel: 372/452 4261  
TELEX: 92246511

#### ITALY

Intel Corporation Italia, S.p.A.  
Corso Sempione 39  
I-20145 Milano  
Tel: 2/34.93287  
TELEX: 311271

#### JAPAN

Intel Japan K.K.\*  
Flower Hill-Shinmachi East Bldg.  
1-23-9 Shinmachi, Setagaya-ku  
Tokyo 154  
Tel: (03) 426-9261  
TELEX: 781-28426

#### NETHERLANDS

Intel Semiconductor B.V.  
Cometongebouw  
Westblaak 106  
3012 Km Rotterdam  
Tel: (10) 149122  
TELEX: 22283

#### NORWAY

Intel Norway A/S  
P.O. Box 92  
Hvamveien 4  
N-2013  
Skjetten  
Tel: (2) 742 420  
TELEX: 18018

#### SWEDEN

Intel Sweden A.B.\*  
Box 20092  
Alpvagen 17  
S-16120 Bromma  
Tel: (08) 98 53 90  
TELEX: 12261

#### SWITZERLAND

Intel Semiconductor A.G.  
Forchstrasse 95  
CH 8032 Zurich  
Tel: 1-55 45 02  
TELEX: 557 89 ich ch

#### UNITED KINGDOM

Intel Corporation (U.K.) Ltd.\*  
5 Hospital Street  
Nantwich, Cheshire CW5 5RE  
Tel: (0270) 62 65 60  
TELEX: 36620

Intel Corporation (U.K.) Ltd.  
Dorcan House  
Eldine Drive  
Swindon, Wiltshire SN3 3TU  
Tel: (0793) 26101  
TELEX: 444447 INT SWN

\*Field Application Location



*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*