

**iSBC 86/12  
SINGLE BOARD COMPUTER  
HARDWARE  
REFERENCE MANUAL**

Manual Order Number: 9800645A



**iSBC 86/12  
SINGLE BOARD COMPUTER  
HARDWARE  
REFERENCE MANUAL**

Manual Order Number: 9800645A

Copyright © 1978 Intel Corporation

Intel Corporation, 3065 Bowers Avenue, Santa Clara, California 95051

The information in this manual is subject to change without notice. Intel Corporation makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Intel Corporation assumes no responsibility for any errors that may appear in this manual. Intel Corporation makes no commitment to update nor to keep current the information contained in this manual.

No part of this manual 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 be used only to describe Intel products:

ICE-30  
ICE-80  
INSITE  
INTEL  
INTELLEC

iSBC  
LIBRARY MANAGER  
MCS  
MEGACHASSIS  
MICROMAP

MULTIBUS  
PROMPT  
UPI  
RMX



## PREFACE

This manual provides general information, installation, programming information, principles of operation, and service information for the Intel iSBC 86/12 Single Board Computer. Additional information is available in the following documents:

- *8086 Assembly Language Reference Manual*, Order No. 9800640
- *Intel MCS-85 User's Manual*, Order No. 98-366
- *Intel 8255A Programmable Peripheral Interface*, Application Note AP-15
- *Intel 8251 Universal Synchronous/Asynchronous Receiver/Transmitter*, Application Note AP-16
- *Intel MULTIBUS Interfacing*, Application Note AP-28
- *Intel 8259 Programmable Interrupt Controller*, Application Note AP-31



## CHAPTER 1

### GENERAL INFORMATION

	PAGE
Introduction .....	1-1
Description .....	1-1
System Software Development .....	1-3
Equipment Supplied .....	1-3
Equipment Required .....	1-3
Specifications .....	1-3

## CHAPTER 2

### PREPARATION FOR USE

Introduction .....	2-1
Unpacking and Inspection .....	2-1
Installation Considerations .....	2-1
User-Furnished Components .....	2-1
Power Requirement .....	2-1
Cooling Requirement .....	2-1
Physical Dimensions .....	2-1
Component Installation .....	2-1
ROM/EPROM Chips .....	2-1
Line Drivers and I/O Terminators .....	2-4
Jumper/Switch Configuration .....	2-4
RAM Addresses (Multibus Access) .....	2-4
Priority Interrupts .....	2-6
Serial I/O Port Configuration .....	2-9
Parallel I/O Port Configuration .....	2-9
Multibus Configuration .....	2-9
Signal Characteristics .....	2-13
Serial Priority Resolution .....	2-13
Parallel Priority Resolution .....	2-13
Power Fail/Memory Protect Configuration .....	2-13
Parallel I/O Cabling .....	2-23
Serial I/O Cabling .....	2-23
Board Installation .....	2-23

## CHAPTER 3

### PROGRAMMING INFORMATION

Introduction .....	3-1
Failsafe Timer .....	3-1
Memory Addressing .....	3-1
CPU Access .....	3-1
Multibus Access .....	3-2
I/O Addressing .....	3-3
System Initialization .....	3-3
8251A USART Programming .....	3-4
Mode Instruction Format .....	3-4
Sync Characters .....	3-5
Command Instruction Format .....	3-5
Reset .....	3-5
Addressing .....	3-5
Initialization .....	3-6
Operation .....	3-7
Data Input/Output .....	3-7
Status Read .....	3-7

	PAGE
8253 PIT Programming .....	3-8
Mode Control Word and Count .....	3-8
Addressing .....	3-12
Initialization .....	3-12
Operation .....	3-13
Counter Read .....	3-13
Clock Frequency/Divide Ratio Selection .....	3-13
Rate Generator/Interval Timer .....	3-14
Interrupt Timer .....	3-14
8255A PPI Programming .....	3-14
Control Word Format .....	3-15
Addressing .....	3-15
Initialization .....	3-16
Operation .....	3-16
Read Operation .....	3-16
Write Operation .....	3-16
8259A PIC Programming .....	3-17
Interrupt Priority Modes .....	3-17
Nested Mode .....	3-17
Fully Nested Mode .....	3-17
Automatic Rotating Mode .....	3-17
Specific Rotating Mode .....	3-17
Special Mask Mode .....	3-18
Poll Mode .....	3-18
Status Read .....	3-18
Initialization Command Words .....	3-18
Operation Command Words .....	3-19
Addressing .....	3-19
Initialization .....	3-19
Operation .....	3-19
Hardware Interrupts .....	3-25
Non-Maskable Interrupt (NMI) .....	3-25
Maskable Interrupt (INTR) .....	3-25
Master PIC Byte Identifier .....	3-25
Slave PIC Byte Identifier .....	3-25

## CHAPTER 4

### PRINCIPLES OF OPERATION

Introduction .....	4-1
Functional Description .....	4-1
Clock Circuits .....	4-1
Central Processor Unit .....	4-1
Interval Timer .....	4-1
Serial I/O .....	4-1
Parallel I/O .....	4-1
Interrupt Controller .....	4-2
ROM/EPROM Configuration .....	4-2
RAM Configuration .....	4-2
Bus Structure .....	4-2
Multibus Interface .....	4-3



## CONTENTS (Continued)

	<b>PAGE</b>
Circuit Analysis .....	4-3
Initialization .....	4-4
Clock Circuits .....	4-4
Central Processor Unit .....	4-4
Basic Timing .....	4-4
Bus Timing .....	4-4
Address Bus .....	4-6
Data Bus .....	4-6
Bus Time Out .....	4-6
Internal Control Signals .....	4-8
Dual Port Control Logic .....	4-8
Multibus Access Timing .....	4-8
CPU Access Timing .....	4-8
Multibus Interface .....	4-11
I/O Operation .....	4-11
On-Board I/O Operation .....	4-11
System I/O Operation .....	4-12
ROM/EPROM Operation .....	4-12

	<b>PAGE</b>
RAM Operation .....	4-12
RAM Controller .....	4-12
RAM Chips .....	4-13
On-Board Read/Write Operation .....	4-13
Bus Read/Write Operation .....	4-13
Byte Operation .....	4-13
Interrupt Operation .....	4-14
NBV Interrupt .....	4-14
BV Interrupt .....	4-14

### **CHAPTER 5 SERVICE INFORMATION**

Introduction .....	5-1
Replaceable Parts .....	5-1
Service Diagrams .....	5-1
Service and Repair Assistance .....	5-1

### **APPENDIX A TELETYPEWRITER MODIFICATIONS**



# TABLES

TABLE	TITLE	PAGE	TABLE	TITLE	PAGE
1-1	Specifications	1-4	3-6	Typical USART Status Read Subroutine	3-9
2-1	User-Furnished and Installed Components	2-2	3-7	PIT Counter Operation Vs Gate Inputs	3-12
2-2	User-Furnished Connector Details	2-3	3-8	Typical PIT Control Word Subroutine	3-12
2-3	Line Driver and I/O Terminator Locations	2-4	3-9	Typical PIT Count Value Load Subroutine	3-12
2-4	Jumper and Switch Selectable Options	2-5	3-10	Typical PIT Counter Read Subroutine	3-13
2-5	Priority Interrupt Jumper Matrix	2-8	3-11	PIT Count Value Vs Rate Multiplier for Each Baud Rate	3-14
2-6	Serial I/O Connector J2 Pin Assignments Vs Configuration Jumpers	2-9	3-12	PIT Rate Generator Frequencies and Timer Intervals	3-15
2-7	Parallel I/O Port Configuration Jumpers	2-10	3-13	PIT Time Intervals Vs Timer Counts	3-15
2-8	Multibus Connector P1 Pin Assignments	2-14	3-14	Typical PPI Initialization Subroutine	3-16
2-9	Multibus Signal Functions	2-15	3-15	Typical PPI Port Read Subroutine	3-16
2-10	iSBC 86/12 DC Characteristics	2-16	3-16	Typical PPI Port Write Subroutine	3-16
2-11	iSBC 86/12 AC Characteristics (Master Mode)	2-18	3-17	Typical PIC Initialization Subroutine (NBV Mode)	3-21
2-12	iSBC 86/12 AC Characteristics (Slave Mode)	2-18	3-18	Typical Master PIC Initialization Subroutine (BV Mode)	3-21
2-13	Auxiliary Connector P2 Pin Assignments	2-22	3-19	Typical Slave PIC Initialization Subroutine (BV Mode)	3-22
2-14	Auxiliary Signal (Connector P2) DC Characteristics	2-22	3-20	PIC Operation Procedures	3-22
2-15	Parallel I/O Connector J1 Pin Assignments	2-23	3-21	Typical PIC Interrupt Request Register Read Subroutine	3-24
2-16	Parallel I/O Signal (Connector J1) DC Characteristics	2-24	3-22	Typical PIC In-Service Register Read Subroutine	3-24
2-17	Connector J2 Vs RS232C Pin Correspondence	2-24	3-23	Typical PIC Set Mask Register Subroutine	3-24
3-1	On-Board Memory Addresses (CPU Access)	3-2	3-24	Typical PIC Mask Register Read Subroutine	3-24
3-2	I/O Address Assignments	3-3	3-25	Typical PIC End-of-Interrupt Command Subroutine	3-25
3-3	Typical USART Mode or Command Instruction Subroutine	3-7	5-1	Replaceable Parts	5-1
3-4	Typical USART Data Character Read Subroutine	3-8	5-2	List of Manufacturers' Codes	5-3
3-5	Typical USART Data Character Write Subroutine	3-8			



# ILLUSTRATIONS

FIGURE	TITLE	PAGE	FIGURE	TITLE	PAGE
1-1	iSBC 86/12 Single Board Computer .....	1-1	3-11	PIT Counter Register Latch Control Word Format .....	3-13
2-1	Dual Port RAM Address Configuration (Multibus Access) .....	2-7	3-12	PPI Control Word Format .....	3-15
2-2	Simplified Master/Slave PIC Interconnect Example .....	2-8	3-13	PPI Port C Bit Set/Reset Control Word Format .....	3-17
2-3	Bus Exchange Timing (Master Mode) .....	2-19	3-14	PIC Initialization Command Word Formats .....	3-18
2-4	Bus Exchange Timing (Slave Mode) .....	2-20	3-15	PIC Operation Control Word Formats .....	3-20
2-5	Serial Priority Resolution Scheme .....	2-21	4-1	iSBC 86/12 Input/Output and Interrupt Simplified Logic Diagram .....	4-15
2-6	Parallel Priority Resolution Scheme .....	2-21	4-2	iSBC 86/12 ROM/EPROM and Dual Port RAM Simplified Logic Diagram .....	4-17
3-1	Dual Port RAM Addressing (Multibus Access) .....	3-2	4-3	Internal Bus Structure .....	4-3
3-2	USART Synchronous Mode Instruction Word Format .....	3-4	4-4	CPU Read Timing .....	4-5
3-3	USART Synchronous Mode Transmission Format .....	3-4	4-5	CPU Write Timing .....	4-6
3-4	USART Asynchronous Mode Instruction Word Format .....	3-5	4-6	CPU Interrupt Acknowledge Cycle Timing .....	4-7
3-5	USART Asynchronous Mode Transmission Format .....	3-5	4-7	Dual Port Control Multibus Access Timing With CPU Lockout .....	4-9
3-6	USART Command Instruction Word Format .....	3-6	4-8	Dual Port Control CPU Access Timing With Multibus Lockout .....	4-10
3-7	Typical USART Initialization and I/O Data Sequence .....	3-6	5-1	iSBC 86/12 Parts Location Diagram .....	5-6
3-8	USART Status Read Format .....	3-9	5-2	iSBC 86/12 Schematic Diagram .....	5-7
3-9	PIT Mode Control Word Format .....	3-10	5-3	iSBC 604 Schematic Diagram .....	5-29
3-10	PIT Programming Sequence Examples .....	3-11	5-4	iSBC 614 Schematic Diagram .....	5-31



## 1-1. INTRODUCTION

The iSBC 86/12 Single Board Computer, which is a member of Intel's complete line of iSBC 80/86 computer products, is a complete computer system on a single printed-circuit assembly. The iSBC 86/12 includes a 16-bit central processing unit (CPU), 32K bytes of dynamic RAM, a serial communications interface, three programmable parallel I/O ports, programmable timers, priority interrupt control, Multibus control logic, and bus expansion drivers for interface with other Multibus-compatible expansion boards. Also included is dual port control logic to allow the iSBC 86/12 to act as a slave RAM device to other Multibus masters in the system. Provision is made for user installation of up to 16K bytes of read only memory.

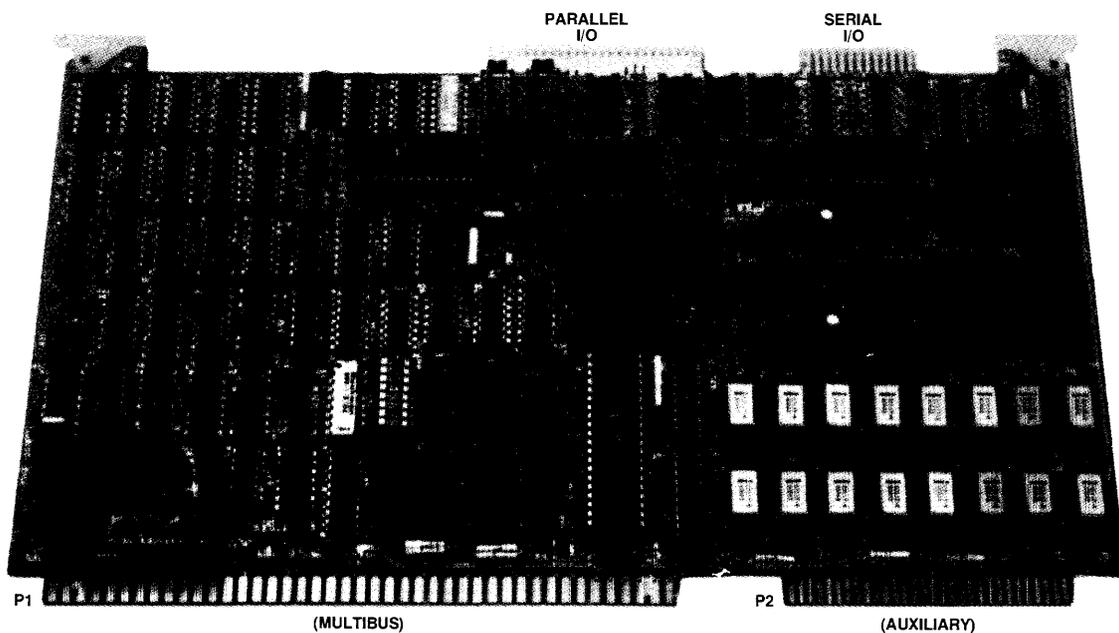
## 1-2. DESCRIPTION

The iSBC 86/12 Single Board Computer (figure 1-1) is controlled by an Intel 8086 16-Bit Microprocessor (CPU). The 8086 CPU includes four 16-bit general purpose registers that may also be addressed as eight 8-bit registers. In

addition, the CPU contains two 16-bit pointer registers and two 16-bit index registers. Four 16-bit segment registers allow extended addressing to a full megabyte of memory. The CPU instruction set supports a wide range of addressing modes and data transfer operations, signed and unsigned 8-bit and 16-bit arithmetic including hardware multiply and divide, and logical and string operations. The CPU architecture features dynamic code relocation, reentrant code, and instruction lookahead.

The iSBC 86/12 has an internal bus for all on-board memory and I/O operations and accesses the system bus (Multibus) for all external memory and I/O operations. Hence, local (on-board) operations do not involve the Multibus, making the Multibus available for true parallel processing when several bus masters (e.g., DMA devices and other single board computers) are used in a multimaster scheme.

Dual port control logic is included to interface the dynamic RAM with the Multibus so that the iSBC 86/12 can function as a slave RAM device when not in control of the Multibus. The CPU has priority when accessing on-board RAM. After the CPU completes its read or write



operation, the controlling bus master is allowed to access RAM and complete its operation. Where both the CPU and the controlling bus master have the need to write or read several bytes or words to or from on-board RAM, their operations are interleaved. For CPU access, the on-board RAM addresses are assigned from the bottom up of the 1-megabyte address space; i.e., 00000-07FFF<sub>H</sub>. The slave RAM address decode logic includes jumpers and switchers to allow partitioning the on-board RAM into any 128K segment of the 1-megabyte system address space.

The slave RAM can be configured to allow either 8K, 16K, 24K, or 32K access by another bus master. Thus, the RAM can be configured to allow other bus masters to access a segment of the on-board RAM and still reserve another segment strictly for on-board use. The addressing scheme accommodates both 16-bit and 20-bit addressing.

Four IC sockets are included to accommodate up to 16K bytes of user-installed read only memory. Configuration jumpers allow read only memory to be installed in 2K, 4K, or 8K increments.

The iSBC 86/12 includes 24 programmable parallel I/O lines implemented by means of an Intel 8255A Programmable Peripheral Interface (PPI). The system software is used to configure the I/O lines in any combination of unidirectional input/output and bidirectional ports. The I/O interface may be customized to meet specific peripheral requirements and, in order to take full advantage of the large number of possible I/O configurations, IC sockets are provided for interchangeable I/O line drivers and terminators. Hence, the flexibility of the parallel 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 (J1) that mates with flat, woven, or round cable.

The RS232C compatible serial I/O port is controlled and interfaced by an Intel 8251A USART (Universal Synchronous/Asynchronous Receiver/Transmitter) chip. The USART is individually programmable for operation in most synchronous or asynchronous serial data transmission formats (including IBM Bi-Sync).

In the synchronous mode the following are programmable:

- a. Character length,
- b. Sync character (or characters), and
- c. Parity.

In the asynchronous mode the following are programmable:

- a. Character length,
- b. Baud rate factor (clock divide ratios of 1, 16, or 64),
- c. Stop bits, and
- d. Parity.

In both the synchronous and asynchronous modes, the serial I/O port features half- or full-duplex, double buffered transmit and receive capability. In addition, USART error detection circuits can check for parity, overrun, and framing errors. The USART transmit and receive clock rates are supplied by a programmable baud rate/time generator. These clocks may optionally be supplied from an external source. The RS232C command lines, serial data lines, and signal ground lines are brought out to a 50-pin edge connector (J2) that mates with flat or round cable.

Three independent, fully programmable 16-bit interval timer/event counters are provided by an Intel 8253 Programmable Interval Timer (PIT). Each counter is capable of operating in either BCD or binary modes; two of these 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 may be independently routed to the 8259A Programmable Interrupt Controller (PIC). The gate/trigger inputs of the two counters may be routed to I/O terminators associated with the 8255A PPI or as input connections from the 8255A PPI. The third counter is used as a programmable baud rate generator for the serial I/O port. In utilizing the iSBC 86/12, the systems designer simply configures, via software, each counter independently to meet system requirements. Whenever a given time delay or count is needed, software commands to the 8253 PIT select the desired function. The contents of each counter may be read at any time during system operation with simple operations for event counting applications, and special commands are included so that the contents of each counter can be read "on the fly".

The iSBC 86/12 provides vectoring for bus vectored (BV) and non-bus vectored (NBV) interrupts. An on-board Intel 8259A Programmable Interrupt Controller (PIC) handles up to eight NBV interrupts. By using external PIC's slaved to the on-board PIC (master), the interrupt structure can be expanded to handle and resolve the priority of up to 64 BV sources.

The PIC, which can be programmed to respond to edge-sensitive or level-sensitive inputs, treats each true input signal condition as an interrupt request. After resolving the interrupt priority, the PIC issues a single interrupt request to the CPU. Interrupt priorities are independently programmable under software control. The programmable interrupt priority modes are:

- a. Fully Nested Priority. Each interrupt request has a fixed priority: input 0 is highest, input 7 is lowest.
- b. Auto-Rotating Priority. Each interrupt request has equal priority. Each level, after receiving service, becomes the lowest priority level until the next interrupt occurs.
- c. Specific priority. Software assigns lowest priority. Priority of all other levels is in numerical sequence based on lowest priority.

The CPU includes a non-maskable interrupt (NMI) and a maskable interrupt (INTR). The NMI interrupt is intended to be used for catastrophic events such as power outages that require immediate action of the CPU. The INTR interrupt is driven by the 8259A PIC which, on demand, provides an 8-bit identifier of the interrupting source. The CPU multiplies the 8-bit identifier by four to derive a pointer to the service routine for the interrupting device.

Interrupt requests may originate from 18 sources without the necessity of external hardware. Two jumper-selectable interrupt requests can be automatically generated by the Programmable Peripheral Interface (PPI) when a byte of information is ready to be transferred to the 8086 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 8086 CPU (i.e., receive channel buffer is full) or when a character is ready to be transmitted (i.e., transmit channel data buffer is empty.) A jumper-selectable interrupt request can be generated by two of the programmable counters and eight additional interrupt request lines are available to the user for direct interfaces to user-designated peripheral devices via the Multibus. One interrupt request line may be jumper routed directly from a peripheral via the parallel I/O driver/terminator section and one power fail interrupt may be input via auxiliary connector P2.

The iSBC 86/12 includes the resources for supporting a variety of OEM system requirements. For those applications requiring additional processing capacity and the benefits of multiprocessing (i.e., several CPU's and/or controllers logically sharing systems tasks with communication over the Multibus), the iSBC 86/12 provides full bus arbitration control logic. This control logic allows up to three bus masters (e.g., combination of iSBC 86/12 DMA controller, diskette controller, etc.) to share the Multibus in serial (daisy-chain) fashion or up to 16 bus masters to share the Multibus using an external parallel priority resolving network.

The Multibus arbitration logic operates synchronously with the bus clock, which is derived either from the iSBC 86/12 or can be optionally generated by some other bus master. Data, however, is transferred via a handshake between the controlling master and the addressed slave module. This arrangement allows different speed control-

lers to share resources on the same bus, and transfers via the bus proceed asynchronously. Thus, the transfer speed is dependent on transmitting and receiving devices only. This design prevents slower 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.

### 1-3. SYSTEM SOFTWARE DEVELOPMENT

The development cycle of iSBC 86/12 based products may be significantly reduced using an Intel Intellec Microcomputer Development System. The resident text editor and system monitor greatly simplify the design, development, and debug of iSBC system software. An optional diskette operating system provides a relocating loader and linkage editor, and a library manager.

Intel's high level programming language, PL/M 86, is also available as a resident 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.

### 1-4. EQUIPMENT SUPPLIED

The following are supplied with the iSBC 86/12 Single Board Computer:

- a. Schematic diagram, dwg no. 2002259
- b. Assembly drawing, dwg no. 1001801

### 1-5. EQUIPMENT REQUIRED

Because the iSBC 86/12 is designed to satisfy a variety of applications, the user must purchase and install only those components required to satisfy his particular needs. A list of components required to configure all the intended applications of the iSBC 86/12 is provided in table 2-1.

### 1-6. SPECIFICATIONS

Specifications of the iSBC 86/12 Single Board Computer are listed in table 1-1.

**Table 1-1. Specifications**

<b>WORD SIZE</b>																																	
Instruction:	8, 16, 24, or 32 bits.																																
Data	8/16 bits.																																
<b>CYCLE TIME:</b>																																	
	800 nanosecond for fastest executable instruction (assumes instruction is in the queue). 1.2 microseconds for fastest executable instruction (assumes instruction is not in the queue).																																
<b>MEMORY CAPACITY</b>																																	
On-Board ROM/EPROM:	Up to 16K bytes; user installed in 1K, 2K, or 4K byte increments.																																
On-Board Dynamic RAM:	32K bytes. Integrity maintained during power failure with user-furnished batteries.																																
Off-Board Expansion:	Up to 1 megabyte of user-specified combination of RAM, ROM, and EPROM.																																
<b>MEMORY ADDRESSING</b>																																	
On-Board ROM/EPROM:	FF000-FFFFFH (using 2758 EPROM's), FE000-FFFFFH (using 2316E ROM's or 2716 EPROM's), and FC000-FFFFFH (using 2332 ROM's).																																
On-Board RAM: (CPU Access)	00000-07FFFH.																																
On-Board RAM: (Multibus Access)	Jumpers and switches allow board to act as slave RAM device for access by another bus master. Addresses may be set within any 8K boundary of any 128K segment of the 1-megabyte system address space. Access is selectable for 8K, 16K, 24K, or 32K bytes.																																
<b>SERIAL COMMUNICATIONS</b>																																	
Synchronous:	5-, 6-, 7-, or 8-bit characters. Internal; 1 or 2 sync characters. Automatic sync insertion.																																
Asynchronous:	5-, 6-, 7-, or 8-bit characters. Break character generation. 1, 1½, or 2 stop bits. False start bit detection.																																
Sample Baud Rate:	<table border="1"> <thead> <tr> <th rowspan="2">Frequency<sup>1</sup> (kHz, Software Selectable)</th> <th colspan="2">Baud Rate (Hz)<sup>2</sup></th> </tr> <tr> <th>Synchronous</th> <th>Asynchronous</th> </tr> </thead> <tbody> <tr> <td></td> <td></td> <td>÷ 16    ÷ 64</td> </tr> <tr> <td>153.6</td> <td>—</td> <td>9600    2400</td> </tr> <tr> <td>76.8</td> <td>—</td> <td>4800    1200</td> </tr> <tr> <td>38.4</td> <td>38400</td> <td>2400    600</td> </tr> <tr> <td>19.2</td> <td>19200</td> <td>1200    300</td> </tr> <tr> <td>9.6</td> <td>9600</td> <td>600    150</td> </tr> <tr> <td>4.8</td> <td>4800</td> <td>300    75</td> </tr> <tr> <td>2.4</td> <td>2400</td> <td>150    —</td> </tr> <tr> <td>1.76</td> <td>1760</td> <td>110    —</td> </tr> </tbody> </table>	Frequency <sup>1</sup> (kHz, Software Selectable)	Baud Rate (Hz) <sup>2</sup>		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    —
Frequency <sup>1</sup> (kHz, Software Selectable)	Baud Rate (Hz) <sup>2</sup>																																
	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    —																															
<p>Notes: 1. Frequency selected by I/O writes of appropriate 16-bit frequency factor to Baud Rate Register.</p> <p>2. Baud rates shown here are only a sample subset of possible software-programmable rates available. Any frequency from 18.75 Hz to 613.5 kHz may be generated utilizing on-board crystal oscillator and 16-bit Programmable Interval Timer (used here as frequency divider).</p>																																	

**Table 1-1. Specifications (Continued)**

<b>INTERVAL TIMER AND BAUD RATE GENERATOR</b>																							
Input Frequency (selectable):	2.46 MHz $\pm$ 0.1% (0.41 $\mu$ sec period nominal), 1.23 MHz $\pm$ 0.1% (0.82 $\mu$ sec period nominal), and 153.6 kHz $\pm$ 0.1% (6.5 $\mu$ sec period nominal).																						
Output Frequencies:	<table border="1"> <thead> <tr> <th rowspan="2">Function</th> <th colspan="2">Single Timer</th> <th colspan="2">Dual Timers (Two Timers Cascaded)</th> </tr> <tr> <th>Min.</th> <th>Max.</th> <th>Min.</th> <th>Max.</th> </tr> </thead> <tbody> <tr> <td>Real-Time Interrupt Interval</td> <td>1.63 <math>\mu</math>sec</td> <td>427.1 msec</td> <td>3.26 <math>\mu</math>sec</td> <td>466.5 minutes</td> </tr> <tr> <td>Rate Generator (Frequency)</td> <td>2.342 Hz</td> <td>613.5 kHz</td> <td>0.000036 Hz</td> <td>306.8 kHz</td> </tr> </tbody> </table>				Function	Single Timer		Dual Timers (Two Timers Cascaded)		Min.	Max.	Min.	Max.	Real-Time Interrupt Interval	1.63 $\mu$ sec	427.1 msec	3.26 $\mu$ sec	466.5 minutes	Rate Generator (Frequency)	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz
Function	Single Timer		Dual Timers (Two Timers Cascaded)																				
	Min.	Max.	Min.	Max.																			
Real-Time Interrupt Interval	1.63 $\mu$ sec	427.1 msec	3.26 $\mu$ sec	466.5 minutes																			
Rate Generator (Frequency)	2.342 Hz	613.5 kHz	0.000036 Hz	306.8 kHz																			
SYSTEM CLOCK (8086 CPU):	5.0 MHz $\pm$ 0.1%.																						
I/O ADDRESSING:	All communication to Parallel I/O and Serial I/O Ports, Timer, and Interrupt Controller is via read and write commands from on-board 8086 CPU. Refer to table 3-2.																						
INTERFACE COMPATIBILITY Serial I/O:	EIA Standard RS232C signals provided and supported: Clear to Send                      Receive Data Data Set Ready                      Secondary Receive Data* Data Terminal Ready                Secondary CTS* Request to Send                      Transmit Clock* Receive Clock                        Transmit Data *Can support only one.																						
Parallel I/O:	24 programmable lines (8 lines per port); one port includes bidirectional bus driver. IC sockets included for user installation of line drivers and/or I/O terminators as required for interface ports. Refer to table 2-1.																						
INTERRUPTS:	8086 CPU includes non-maskable interrupt (NMI) and maskable interrupt (INTR). NMI interrupt is provided for catastrophic event 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.																						
COMPATIBLE CONNECTORS/CABLES:	Refer to table 2-2 for compatible connector details. Refer to paragraphs 2-21 and 2-22 for recommended types and lengths of I/O cables.																						
<b>ENVIRONMENTAL REQUIREMENTS</b>																							
Operating Temperature:	0° to 55°C (32° to 131°F).																						
Relative Humidity:	To 90% without condensation.																						
<b>PHYSICAL CHARACTERISTICS</b>																							
Width:	30.48 cm (12.00 inches).																						
Height:	17.15 cm (6.75 inches).																						
Thickness:	1.78 cm (0.7 inch).																						
Weight:	539 gm (19 ounces).																						

**Table 1-1. Specifications (Continued)**

POWER REQUIREMENTS:

CONFIGURATION	$V_{CC} = +5V \pm 5\%$	$V_{DD} = +12V \pm 5\%$	$V_{BB} = -5V \pm 5\%$	$V_{AA} = -12V \pm 5\%$
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	450 mA	—	140 mA
With 8K ROM <sup>5</sup> (Using 2316E)	6.1A	450 mA	—	140 mA
With 8K EPROM <sup>5</sup> (Using 2716)	5.5A	450 mA	—	140 mA
With 16K ROM <sup>5</sup> (Using 2332)	5.4A	450 mA	—	140 mA

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



## CHAPTER 2 PREPARATION FOR USE

### 2-1. INTRODUCTION

This chapter provides instructions for the iSBC 86/12 Single Board Computer in the user-defined environment. It is advisable that the contents of Chapters 1 and 3 be fully understood before beginning the configuration and installation procedures provided in this chapter.

### 2-2. UNPACKING AND INSPECTION

Inspect the shipping carton immediately upon receipt for evidence of mishandling during transit. If the shipping carton is severely damaged or waterstained, request that the carrier's agent be present when the carton is opened. If the carrier's agent is not present when the carton is opened and the contents of the carton are damaged, keep the carton and packing material for the agent's inspection.

For repairs to a product damaged in shipment, contact the Intel Technical Support Center (see paragraph 5-3) to obtain a Return Authorization Number and further instructions. A purchase order will be required to complete the repair. A copy of the purchase order should be submitted to the carrier with your claim.

It is suggested that salvageable shipping cartons and packing material be saved for future use in the event the product must be reshipped.

### 2-3. INSTALLATION CONSIDERATIONS

The iSBC 86/12 is designed for use in one of the following configurations:

- a. Standalone (single-board) system.
- b. Bus master in a single bus master system.
- c. Bus master in a multiple bus master system.

Important criteria for installing and interfacing the iSBC 86/12 in these configurations are presented in following paragraphs.

### 2-4. USER-FURNISHED COMPONENTS

The user-furnished components required to configure the iSBC 86/12 for a particular application are listed in table 2-1. Various types and vendors of the connectors specified in table 2-1 are listed in table 2-2.

### 2-5. POWER REQUIREMENT

The iSBC 86/12 requires +5V, -5V, +12V, and -12V power. The -5V power, which is required only for the dual port RAM, can be supplied by the system -5V supply, an auxiliary battery, or by the on-board -5V regulator. (The -5V regulator operates from the system -12V supply.)

### 2-6. COOLING REQUIREMENT

The iSBC 86/12 dissipates 451 gram-calories/minute (1.83 Btu/minute) and adequate circulation of air must be provided to prevent a temperature rise above 55°C (131°F). The System 80 enclosures and the Intellec System include fans to provide adequate intake and exhaust of ventilating air.

### 2-7. PHYSICAL DIMENSIONS

Physical dimensions of the iSBC 86/12 are as follows:

- a. Width: 30.48 cm (12.00 inches).
- b. Height: 17.15 cm (6.75 inches).
- c. Thickness: 1.78 cm (0.70 inch).

### 2-8. COMPONENT INSTALLATION

Instructions for installing optional ROM/EPROM and parallel I/O port line drivers and/or line terminators are given in following paragraphs. When installing these chip components, be sure to orient pin 1 of the chip adjacent to the white dot located near pin 1 of the associated IC socket. The grid zone location on figure 5-1 (parts location diagram) is specified for each component chip to be installed.

### 2-9. ROM/EPROM CHIPS

IC sockets A28, A29, A46, and A47 (figure 5-1 zone C3) accommodate 24-pin ROM/EPROM chips. Because the CPU jumps to location FFFF0 on a power up or reset, the ROM/EPROM address space resides in the topmost portion of the 1-megabyte address space and must be loaded from the top down. IC sockets A29 and A47 accommodate the top of the ROM/EPROM address space and must always be loaded; IC sockets A28 and A46 accommodate the ROM/EPROM space directly below that installed in A29 and A47.

**Table 2-1. User-Furnished and Installed Components**

Item No.	Item	Description	Use												
1	iSBC 604	Modular Backplane and Cardcage. Includes four slots with bus terminators. (See figure 5-3.)	Provides power input pins and Multibus signal interface between iSBC 86/12 and three additional boards in a multiple board system.												
2	iSBC 614	Modular Backplane and Cardcage. Includes four slots without bus terminators. (See figure 5-4.)	Provides four-slot extension of iSBC 604.												
3	Connector (mates with P1)	See Multibus Connector details in table 2-2.	Power inputs and Multibus signal interface. Not required if iSBC 86/12 is installed in an iSBC 604/614.												
4	Connector (mates with P2)	See Auxiliary Connector details in table 2-2.	Auxiliary backup battery and associated memory protect functions.												
5	Connector (mates with J1)	See Parallel I/O Connector details in table 2-2.	Interfaces parallel I/O port with Intel 8255A PPI.												
6	Connector (mates with J2)	See Serial I/O connector details in table 2-2.	Interfaces serial I/O port with Intel 8251A USART.												
7	ROM/EPROM Chips	Two or four each of the following types:  <table style="margin-left: 40px; border: none;"> <tr> <td style="text-align: center;"><u>ROM</u></td> <td style="text-align: center;">or</td> <td style="text-align: center;"><u>EPROM</u></td> </tr> <tr> <td style="text-align: center;">—</td> <td></td> <td style="text-align: center;">2758</td> </tr> <tr> <td style="text-align: center;">2316E</td> <td></td> <td style="text-align: center;">2716</td> </tr> <tr> <td style="text-align: center;">2332</td> <td></td> <td style="text-align: center;">—</td> </tr> </table>	<u>ROM</u>	or	<u>EPROM</u>	—		2758	2316E		2716	2332		—	Ultraviolet Erasable PROM (EPROM) for development. Masked ROM for dedicated program.
<u>ROM</u>	or	<u>EPROM</u>													
—		2758													
2316E		2716													
2332		—													
8	Line Drivers	Type	Current												
		SN7403 I, OC SN7400 I SN7408 NI SN7409 NI, OC	16 mA 16 mA 16 mA 16 mA												
		Types selected as typical; I = inverting, NI = noninverting, and OC = open collector.													
9	Line Terminators	Intel iSBC 901 Divider or iSBC 902 Pull-Up:	Interface parallel I/O ports CA and CC with Intel 8255A PPI. Requires two line driver IC's for each 8-bit parallel output port.												

Table 2-2. User-Furnished Connector Details

Function	No. Of Pairs/Pins	Centers (inches)	Connector Type	Vendor	Vendor Part No.	Intel Part No.
Parallel I/O Connector	25/50	0.1	Flat Crimp	3M 3M AMP ANSLEY SAE	3415-0000 WITH EARS 3415-0001 W/O EARS 88083-1 609-5015 SD6750 SERIES	iSBC 956 Cable Set
Parallel I/O Connector	25/50	0.1	Soldered	AMP VIKING TI	2-583485-6 3VH25/1JV5 H312125	N/A
Parallel I/O Connector	24/50	0.1	Wirewrap <sup>1</sup>	TI VIKING CDC <sup>3</sup> ITT CANNON	H311125 3VH25/1JND5 VPB01B25D00A1 EC4A050A1A	N/A
Serial I/O Connector	13/26	0.1	Flat Crimp	3M AMP ANSLEY SAE	3462-0001 88106-1 609-2615 SD6726 SERIES	iSBC 955 Cable Set
Serial I/O Connector	13/26	0.1	Soldered	TI AMP	H312113 1-583485-5	N/A
Serial I/O Connector	13/26	0.1	Wirewrap <sup>1</sup>	TI	H311113	N/A
Multibus Connector	43/86	0.156	Soldered <sup>1</sup>	CDC <sup>3</sup> MICRO PLASTICS ARCO VIKING	VPB01E43D00A1 MP-0156-43-BW-4 AE443WP1 LESS EARS 2VH43/1AV5	N/A
Multibus Connector	43/86	0.156	Wirewrap <sup>1,2</sup>	CDC <sup>3</sup> CDC <sup>3</sup> VIKING	VFB01E43D00A1 or VPB01E43A00A1 2VH43/1AV5	MDS 985
Auxiliary Connector	30/60	0.1	Soldered <sup>1</sup>	TI VIKING	H312130 3VH30/1JN5	N/A
Auxiliary Connector	30/60	0.1	Wirewrap <sup>1,2</sup>	CDC <sup>3</sup> TI	VPB01B30A00A2 H311130	N/A

## NOTES:

1. Connector heights are not guaranteed to conform to OEM packaging equipment.
2. Wirewrap pin lengths are not guaranteed to conform to OEM packaging equipment.
3. CDC VPB01 ..., VPB02 ..., VPB04 ..., etc. are identical connectors with different electroplating thicknesses or metal surfaces.

The low-order byte (bits 0-7) of ROM/EPROM must be installed in sockets A29 and A28; the high-order byte (bits 8-15) must be installed in sockets A47 and A46. Assuming that 2K bytes of EPROM are to be installed using two Intel 2758 chips, the chip containing the low-order byte must be installed in IC socket A29 and the chip containing the high-order byte must be installed in IC socket A47. In this configuration, the usable ROM/EPROM address space is FF800-FFFFFF. Two additional Intel 2758 chips may be installed later in IC sockets A28 and A46 and occupy the address space FF000-FF7FF. (Even addresses read the low-order bytes and odd addresses read the high-order bytes.)

The default (factory connected) jumpers and switch S1 are configured for 2K by 8-bit ROM/EPROM chips (e.g., two or four Intel 2716's). If different type chips are installed, reconfigure the jumpers and switch S1 as listed in table 2-4.

## 2-10. LINE DRIVERS AND I/O TERMINATORS

Table 2-3 lists the I/O ports and the location of associated 14-pin IC sockets for installing either line drivers or I/O terminators. (Refer to table 2-1 items 8 and 9.) Port C8 is factory equipped with Intel 8226 Bidirectional Bus Drivers and requires no additional components.

## 2-11. JUMPER/SWITCH CONFIGURATION

The iSBC 86/12 includes a variety of jumper- and switch-selectable options to allow the user to configure the board for his particular application. Table 2-4 summarizes these options and lists the grid reference locations of the jumpers and switches as shown in figure 5-1 (parts location diagram) and figure 5-2 (schematic diagram). Because the schematic diagram consists of 11 sheets, grid

references to figure 5-2 may be either four or five alphanumeric characters. For example, grid reference 3ZB7 signifies sheet 3 Zone B7.

Study table 2-4 carefully while making reference to figures 5-1 and 5-2. If the default (factory configured) jumpers and switch settings are appropriate for a particular function, no further action is required for that function. If, however, a different configuration is required, reconfigure the switch settings and/or remove the default jumper(s) and install an optional jumper(s) as specified. For most options, the information in table 2-4 is sufficient for proper configuration. Additional information, where necessary for clarity, is described in subsequent paragraphs.

## 2-12. RAM ADDRESSES (MULTIBUS ACCESS)

The dual port RAM can be shared with other bus masters via the Multibus. One jumper wire connected between a selected pair of jumper posts (113 through 128) places the dual port RAM in one of eight 128K byte segments of the 1-megabyte address space. Switch S1 is a dual-inline package (DIP) composed of eight individual single-pole, single-throw switches. (Two of these individual switches are used for ROM/EPROM configuration.) Two switches (6-11 and 5-12) are configured to allow 8K, 16K, 24K, or 32K bytes of dual port RAM to be accessed. Four switches (1-16, 2-15, 3-14, and 4-13) are configured to displace the addresses from the top of the selected 128K byte segment of memory.

Figure 2-1 provides an example of 8K bytes of dual port RAM being made accessible from the Multibus and how the addresses are established. Note in figure 2-1 that the Multibus accesses the dual port RAM from the top down. Thus, as shown for 8K byte access via the Multibus, the bottom 24K bytes of the iSBC 86/12 on-board RAM is reserved strictly for on-board CPU access.

**Table 2-3. Line Driver and I/O Terminator Locations**

	I/O Port	Bits	Driver/Terminator	Fig. 5-1 Grid Ref.	Fig. 5-2* Grid Ref.
8255A PPI Interface	C8	0-7	None Required	—	—
	CA	0-3	A12	ZD4	9ZA3
		4-7	A13	ZD4	9ZA3
CC	0-3	A11	ZD5	9ZC3	
	4-7	A10	ZD5	9ZB3	

\*Figure 5-2 is the schematic diagram. Grid reference 9ZA3, for example, denotes sheet 9 Zone A3.

Table 2-4. Jumper and Switch Selectable Options

Function	Fig. 5-1 Grid Ref.	Fig. 5-2 Grid Ref.	Description																						
ROM/EPROM Configuration	ZC3, ZB6, ZD7	6ZB3, 6ZC7, 2ZB6	<p>Jumpers 94 through 99 and switch S1 may be configured to accommodate four types of ROM/EPROM chips:</p> <table border="1" style="margin-left: 20px;"> <thead> <tr> <th rowspan="2">ROM/EPROM Type</th> <th rowspan="2">Jumpers</th> <th colspan="2">Switch S1</th> </tr> <tr> <th>8-9</th> <th>7-10</th> </tr> </thead> <tbody> <tr> <td>2758</td> <td>94-95, 97-98</td> <td>C</td> <td>C</td> </tr> <tr> <td>2316E/2716</td> <td>*94-96, *97-98</td> <td>*C</td> <td>*O</td> </tr> <tr> <td>2332</td> <td>94-96, 97-99</td> <td>O</td> <td>C</td> </tr> <tr> <td>Reserved</td> <td>—</td> <td>O</td> <td>O</td> </tr> </tbody> </table> <p>C = closed switch position. O = open switch position.</p> <p>Default jumpers and switch settings accommodate Intel 2316E/2716 chips. Disconnect existing configuration jumpers (if necessary) and reset switch S1 if reconfiguration is required.</p>	ROM/EPROM Type	Jumpers	Switch S1		8-9	7-10	2758	94-95, 97-98	C	C	2316E/2716	*94-96, *97-98	*C	*O	2332	94-96, 97-99	O	C	Reserved	—	O	O
ROM/EPROM Type	Jumpers	Switch S1																							
		8-9	7-10																						
2758	94-95, 97-98	C	C																						
2316E/2716	*94-96, *97-98	*C	*O																						
2332	94-96, 97-99	O	C																						
Reserved	—	O	O																						
Dual Port RAM (Multibus Access)	ZB7, ZB6	3ZB6, 3ZB7	The dual port RAM permits access by the local (on-board) CPU and any system bus master via the Multibus. For local CPU access, the dual port RAM address space is fixed beginning at location 00000. For access via the Multibus, one jumper and one switch can configure the dual port RAM on any 8K boundary within the 1-megabyte address space. Refer to paragraph 2-12 for configuration details.																						
Bus Clock	ZB7	10ZA2	Default jumper *105-106 routes Bus Clock signal BCLK/ to the Multibus. (Refer to table 2-9.) Remove this jumper <i>only</i> if another bus master supplies this signal.																						
Constant Clock	ZB7	10ZA2	Default jumper *103-104 routes Constant Clock signal CCLK/ to the Multibus. (Refer to table 2-9.) Remove this jumper <i>only</i> if another bus master supplies this signal.																						
Bus Priority Out	ZB7	3ZD2	Default jumper *151-152 routes Bus Priority Out signal BPRO/ to the Multibus. (Refer to table 2-9.) Remove this jumper <i>only</i> in those systems employing a parallel priority bus resolution scheme. (Refer to paragraph 2-19.)																						
Bus Arbitration	ZB8, ZD7	3ZD2, 3ZC3	The Common Bus Request signal (CBRQ) from the Multibus and the ANYRQST input to the Bus Arbiter chip are not presently used.																						
Auxiliary Backup Batteries	ZD3, ZB6, ZB5	1ZC7, 1ZC6	If auxiliary backup batteries are used to sustain the dual port RAM contents during ac power outages, remove default jumpers *W4(A-B), *W5(A-B), and *W6(A-B).																						
On-Board -5V Regulator	ZB6	1ZC6	The dual port RAM requires a -5V AUX input, which can be supplied by the system -5V supply, an auxiliary backup battery, or by the on-board -5V regulator. (The -5V regulator operates from the system -12V supply.) If a system -5V supply is available and auxiliary backup batteries are not used, disconnect default jumper *W5(A-B) and connect jumper W5(B-C). If auxiliary backup batteries are used, disconnect default jumper *W5(A-B); do not connect W5(B-C).																						
Failsafe Timer	ZD7	2ZB6	If the on-board CPU addresses either a system or an on-board memory or I/O device and that device does not return an acknowledge signal, the CPU will hang up in a wait state. A failsafe timer is triggered during T1 of every machine cycle and, if not retriggered within 6.2 milliseconds, the resultant time-out pulse can be used to allow the CPU to exit the wait state. If this feature is desired, connect jumper 5-6.																						
*Default jumper connected at the factory.																									

Table 2-4. Jumper and Switch Selectable Options (Continued)

Function	Fig. 5-1 Grid Ref.	Fig. 5-2 Grid Ref.	Description
Timer Input Frequency	ZD3	7ZB5	Input frequencies to the 8253 Programmable Interval Timer are jumper selectable as follows: <u>Counter 0 (TMR0 INTR)</u> 57-58: 153.6 kHz. *57-56: 1.23 MHz. 57-53: 2.46 MHz. 57-62: External Clock to/from Port CC terminator/driver.
	ZD3	7ZA5	<u>Counter 1 (TMR1 INTR)</u> *59-60: 153.6 kHz. 59-56: 1.23 MHz. *59-53: 2.46 MHz. 59-62: External Clock to/from Port CC terminator/driver. 59-61: Counter 0 output.  Jumper 59-61 effectively connects Counter 0 and Counter 1 in series in which the output of Counter 0 serves as the input clock to Counter 1. This permits programming the clock rates to Counter 1 and thus provide longer TMR1 INTR intervals.
	ZD3	7ZB5	<u>Counter 2 (8251 Baud Rate Clock)</u> 55-58: 153.6 kHz. *55-54: 1.23 MHz. 55-53: 2.46 MHz. 55-62: External Clock to/from Port CC terminator/driver.
Priority Interrupts	—	Sheet 8	A jumper matrix provides a wide selection of interrupts to be interfaced to the 8086 CPU and the Multibus. Refer to paragraph 2-13 for configuration.
Serial I/O Port Configuration	—	Sheet 7	Jumpers posts 38 through 52 are used to configure the 8251A USART as described in paragraph 2-14.
Parallel I/O Port Configuration	—	Sheet 9	Jumper posts 7 through 37 are used to configure the 8255A PPI as described in paragraph 2-15.
*Default jumper connected at the factory.			

The configuration for 16K, 24K, or 32K access is done in a similar manner. Always observe the IMPORTANT note in figure 2-1 in that the address space intended for Multibus access of the dual port RAM must not cross a 128K boundary.

If it is desired to reserve all the dual port RAM strictly for local CPU access, connect jumper 112-114.

## 2-13. PRIORITY INTERRUPTS

Table 2-5 lists the source (from) and destination (to) of the priority interrupt jumper matrix shown in figure 5-2 sheet 8. The INTR output of the on-board Intel 8259A Programmable Interrupt Controller (PIC) is applied directly to the INTR input of the 8086 CPU. The on-board PIC,

which handles up to eight vectored priority interrupts, provides the capability to expand the number of priority interrupts by cascading each interrupt line with another 8259A PIC. Figure 2-2 shows as an example the on-board PIC (master) with two slave PIC's interfaced by the Multibus. This arrangement leaves the master PIC with six inputs (IR2 through IR7) that can be used to handle the various on-board interrupt functions.

The master/slave PIC arrangement illustrated in figure 2-2 is implemented by programming the master PIC to handle IR0 and IR1 as bus vectored interrupt inputs. For example, if the Multibus INT3/line is driven low by slave PIC 1, the master PIC will let slave PIC 1 send the restart address to the 8086 CPU.

Each interrupt input (IR0 through IR7) to the master PIC can be individually programmed to be a non-bus vectored

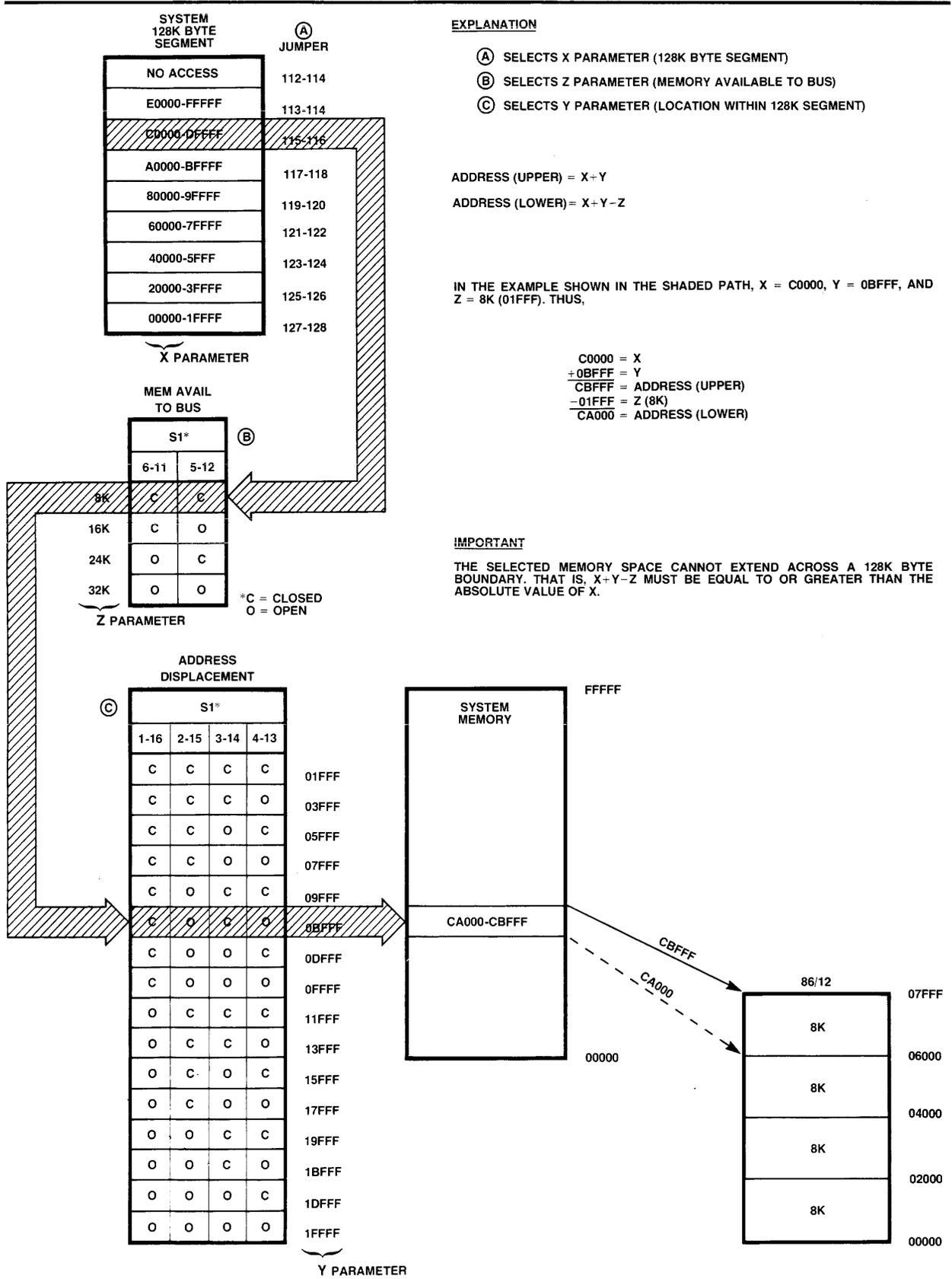
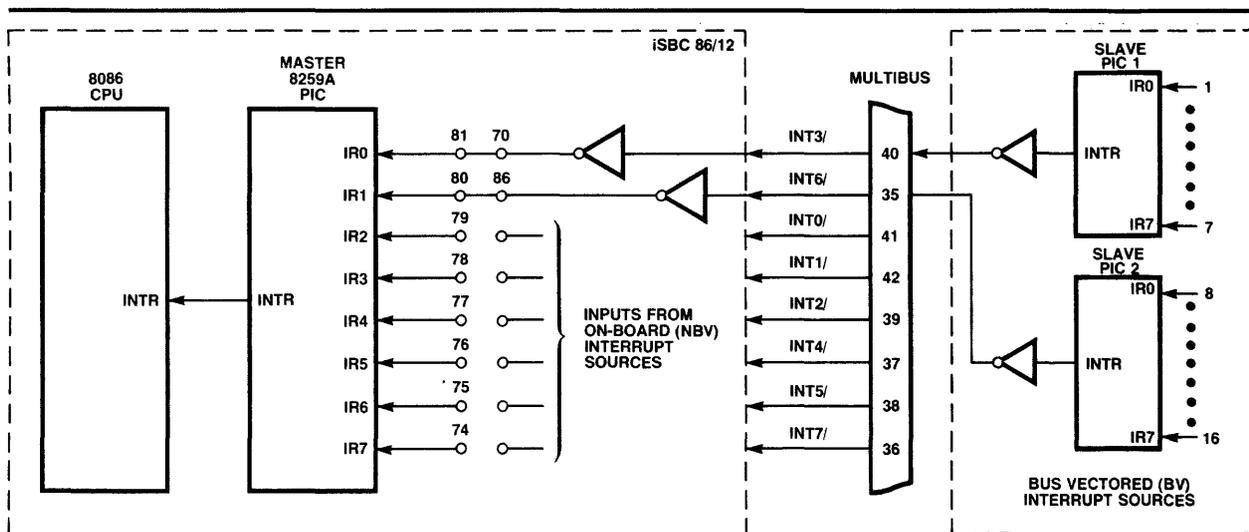


Figure 2-1. Dual Port RAM Address Configuration (Multibus Access)

Table 2-5. Priority Interrupt Jumper Matrix

Interrupt Request From			Interrupt Request To			
Source	Signal	Post	Device	Signal	Post	
Multibus (2)	INT0/ (1)	73	Multibus (2)	INT0/ (4)	141	
	INT1/ (1)	72		INT1/ (4)	140	
	INT2/ (1)	71		INT2/ (4)	139	
	INT3/ (1)	70		INT3/ (4)	138	
	INT4/ (1)	69		INT4/ (4)	137	
	INT5/ (1)	68		INT5/ (4)	136	
	INT6/ (1)	66		INT6/ (4)	135	
	INT7/ (1)	65		INT7/ (4)	134	
External Via J1-50 Power Fail Logic Via P2-19 Failsafe Timer	EXT INT0/ (1)	67	8259A PIC (6)	IR0 (5)	81	
	PFI/ (1)	86		IR1 (5)	80	
TIME OUT INTR (1)	88	IR2 (5)		79		
8255A PPI Port A (Port C8) Port B (Port CA) Any Unused Bit	PA INTR (1)	84		IR3 (5)	78	
	PB INTR (1)	85		IR4 (5)	77	
	BUS INTR OUT (3)(9)			142	IR5 (5)	76
					IR6 (5)	75
8251A USART Trans Buffer Empty Rec Buffer Empty	51TX INTR (1)	90	8086 CPU	NMI (7)	89	
	51RX INTR (1)	82		INTR (8)	—	
8253 PIT Timer 0 Out Timer 1 Out	TMR0 INTR (1)	83				
	TMR1 INTR (1)	91				

NOTES:  
 (1) Signal is positive-true at associated jumper post.  
 (2) INT0/ is highest priority; INT7/ is lowest priority.  
 (3) Signal is ground-true at associated jumper post.  
 (4) Requires ground-true signal at associated jumper post.  
 (5) Requires positive-true signal at associated jumper post.  
 (6) IR0 is highest priority; IR7 is lowest priority.  
 (7) Default jumper 87-89 disables (grounds) input. The NMI input is highest priority, non-maskable, and is both level and edge sensitive.  
 (8) INTR is connected directly to output of 8259A PIC.  
 (9) Used to generate an interrupt on Multibus.



645-3

Figure 2-2. Simplified Master/Slave PIC Interconnect Example

(NBV) interrupt (the master PIC generates the restart address) or bus vectored (BV) interrupt (the slave PIC generates the restart address). Thus, the master PIC can handle eight on-board or single Multibus interrupt lines (an interrupt line that is not driven by a slave PIC) or up to 64 interrupts with the implementation of slave PIC's.

The iSBC 86/12 can also generate an interrupt to another interrupt handler via the Multibus. This is accomplished by using one of the bits of the 8255A PPI to drive the BUS INTR OUT signal. (The BUS INTR OUT signal is ground-true at jumper post 142 as footnoted in table 2-5.)

Default jumper 87-89 grounds the NMI (nonmaskable interrupt) input to the CPU to prevent the possibility of false interrupts being generated by noise spikes. Since the NMI is not maskable, cannot be disabled by the program, and has the highest priority, it should only be used to detect a power failure. For this purpose, disconnect default jumper 87-89 and connect 86-89. The Power Fail Interrupt (PFI/) is an externally generated signal that is input via auxiliary connector P2. (Refer to paragraph 2-20.)

## 2-14. SERIAL I/O PORT CONFIGURATION

Table 2-6 lists the signals, signal functions, and the jumpers required (if necessary) to input or output a particular signal to or from the serial I/O port (Intel 8251A USART).

## 2-15. PARALLEL I/O PORT CONFIGURATION

Table 2-7 lists the jumper configuration for three parallel I/O ports. Note that each of the three ports (C8, CA, and CC) can be configured in a variety of ways to suit the individual requirement.

## 2-16. MULTIBUS CONFIGURATION

For systems applications, the iSBC 86/12 is designed for installation in a standard Intel iSBC 604/614 Modular Backplane and Cardcage. (Refer to table 2-1 items 1 and 2.) Alternatively, the iSBC 86/12 can be interfaced to a user-designed system backplane by means of an

**Table 2-6. Serial I/O Connector J2 Pin Assignments Vs Configuration Jumpers**

Pin <sup>1</sup>	Signal	Function	Jumper In	Jumper Out
2	CHASSIS GND	Protective ground	63-64	—
4	TRANSMITTER DATA	8251A RXD in	—	—
5	SEC REC SIG <sup>2</sup>	Same as 8261A TXC in or 8255A STXD out (Note 3)	48-49, 45-46 49-50, 45-46	— —
6	RECEIVER DATA	8251A TXD out	—	—
7	REC SIG ELE TIMING	8251A RXC in (Note 4) 8251A TXC in (Note 4)	38-39 41-42	*39-40 *42-43
8	RQT TO SEND	8251A CTS in (Note 5)	—	—
10	CLEAR TO SEND	8251A RTS out (Note 5)	—	—
12	DATA SET RDY	8251A DTR out	—	—
13	DATA TERMINAL RDY	8251A DSR in	—	—
14	GND	Ground	—	—
19	-12V	-12V out	*W3A-B	—
21	TRANS SIG ELE TIMING <sup>2</sup>	Same as 8251A TXC in or 8255A STXD out (Note 3)	48-49, 44-45 49-50, 44-45	— —
22	+12V	+12V out	*W2A-B	—
23	+5V	+5V out	*W1A-B	—
25	GND	Ground	—	—
26	SEC CTS <sup>2</sup>	Same as 8251A TXC in or 8255A STXD out (Note 3)	48-49, 45-47 49-50, 45-47	— —

### NOTES:

- All odd-numbered pins (1,3,5, . . . 25) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side of the board with the extractors at the top.
- Only one of these signal outputs (pin 5, 21, or 26) may be selected.
- Optional jumper selected output of 8255A PPI. Refer to figure 5-2 sheet 9.
- Default jumpers \*39-40 and \*42-43 connect 8253 CTR2 output to 8251A RXC and TXC inputs, respectively. See Timer Input Frequency (Counter 2) in table 2-4.
- For those applications without CTS capability, connect jumper 51-52. This routes 8251A RTS output to 8251A CTS input.

\*Default jumpers connected at the factory.

**Table 2-7. Parallel I/O Port Configuration Jumpers**

Port	Mode	Driver (D)/ Terminator (T)	Jumper Configuration			Restrictions	
			Delete	Add	Effect	Port	
C8	0 Input	8226: A8, A9	*21-25	24-25	8226 = input enabled.	CA	None; can be in mode 0 or 1, input or output.
						CC	None; can be in Mode 0, input or output, unless Port CA is in Mode 1.
C8	0 Output (latched)	8226: A8, A9		*21-25	8226 = output enabled.	CA	None; can be in Mode 0 or 1, input or output.
						CC	None; can be in Mode 0, input or output, unless Port CA is in Mode 1.
C8	1 Input (strobed)	8226: A8, A9 T: A10 D: A11	*21-25  *19-20 and *32-33	24-25	8226 = input enabled.	CA	None; can be in Mode 0 or 1, input or output.
				*15-16 19-33 22-32	Connects J1-26 to STB <sub>A</sub> / input. Connects IBF <sub>A</sub> output to J1-18. Connects INT <sub>A</sub> output to interrupt matrix.	CC	Port CC bits perform the following: <ul style="list-style-type: none"> <li>• Bits 0, 1, 2 — Control for Port CA if Port CA is in Mode 1.</li> <li>• Bit 3 — Port C8 Interrupt (PA INTR) to interrupt jumper matrix</li> <li>• Bit 4 — Port C8 Strobe (STB/) input.</li> <li>• Bit 5 — Port C8 Input Buffer Full (IBF) output.</li> <li>• Bits 6, 7 — Port CC input or output (both, must be in same direction).</li> </ul>
C8	1 Output (latched)	8226: A8, A9 T: A10 D: A11	*32-33 and *13-14	*21-25	8226 = output enabled.	CA	None; can be in Mode 0 or 1, input or output.
				*17-18 13-33 22-32	Connects J1-30 to ACK <sub>A</sub> / input. Connects OBF <sub>A</sub> output to J1-18. Connects INT <sub>A</sub> output to interrupt matrix.	CC	Port EA bits perform the following: <ul style="list-style-type: none"> <li>• Bits 0, 1, 2 — Control for Port CA if Port CA is in Mode 1.</li> <li>• Bit 3 — Port C8 Interrupt (PA INTR) to interrupt jumper matrix.</li> <li>• Bits 4, 5 — Port CC input or output (both must be in same direction).</li> <li>• Bit 6 — Port C8 Acknowledge (ACK/) input.</li> <li>• Bit 7 — Port C8 Output Buffer Full (OBF/) output.</li> </ul>

\*Default jumper connected at the factory.

Table 2-7. Parallel I/O Configuration Jumpers (Continued)

Port	Mode	Driver (D)/ Terminator (T)	Jumper Configuration			Restrictions	
			Delete	Add	Effect		
C8	2 (bidirectional)	8226: A8, A9 T: A10 D: A11	*21-25	17-25	Allows ACK <sub>A</sub> / input to control 8226 in/out direction.	CA	None; can be in Mode 0 or 1, input or output.
				*15-16	Connects J1-26 to STB <sub>A</sub> / input.	CC	Port CC bits perform the following: <ul style="list-style-type: none"> <li>• Bit 0 — Can only be used for jumper option (see figure 5-2 zone 9ZC6).</li> <li>• Bits 1, 2— Can be used for input or output if Port CC is in Mode 0.</li> <li>• Bit 3 — Port C8 Interrupt (PA INTR) to interrupt jumper matrix.</li> <li>• Bit 4 — Port C8 Strobe (STB/) input.</li> <li>• Bit 5 — Port C8 Input Buffer Full (IBF) output.</li> <li>• Bit 6 — Port C8 Acknowledge (ACK/) input.</li> <li>• Bit 7 — Port C8 Output Buffer Full (OBF/) output.</li> </ul>
CA	0 Input	T: A12, A13	None	None		C8	None.
						CC	None; Port CC can be in Mode 0, input or output, if Port C8 is also in Mode 0.
CA	0 Output (latched)	D: A12, A13	None	None		C8	None
						CC	None; Port CC can be in Mode 0, input or output, if Port C8 is also in Mode 0.
CA	1 Input (strobed)	T: A10, A12, A13 D: A11	*13-14 *30-31	*28-29	Connects IBF <sub>B</sub> output to J1-22.	C8	None.
				14-30	Connects J1-32 to STB <sub>B</sub> / input.	CC	Port CC bits perform the following: <ul style="list-style-type: none"> <li>• Bit 0 — Port CA Interrupt (PB INTR) to interrupt jumper matrix.</li> <li>• Bit 1 — Port CA Input Buffer Full (IBF) output.</li> <li>• Bit 2 — Port CA Strobe (STB/) input.</li> </ul>
				26-34	Connects INT <sub>B</sub> output interrupt matrix.		
*Default jumper connected at the factory.							

**Table 2-7. Parallel I/O Port Configuration Jumpers (Continued)**

Port	Mode	Driver (D) Terminator (T)	Jumper Configuration			Restrictions	
			Delete	Add	Effect		
						<ul style="list-style-type: none"> <li>● Bit 3 — If Port C8 is in Mode 0, bit 3 can be input or output. Otherwise, bit 3 is reserved.</li> <li>● Bits 4, 5 — Depends on Port C8 mode.</li> <li>● Bits 6, 7 — Input or output (both must be in same direction).</li> </ul>	
CA	1 Output (latched)	T: A10 D: A11, A12, A13		*28-29	Connects OBF <sub>B</sub> / output output J1-22.	C8	None.
			*13-14 and *30-31	14-30	Connects J1-32 to ACK <sub>B</sub> / input.	CC	Port CC bits perform the following: <ul style="list-style-type: none"> <li>● Bit 0 — Port CA interrupt (PB INTR) to interrupt jumper matrix.</li> <li>● Bit 1 — Port CA Output Buffer Full (OBF/) output.</li> <li>● Bit 2 — Port CA Acknowledge (ACK/) input.</li> <li>● Bit 3 — If Port C8 is in Mode 0, bit 3 can be input or output. Otherwise, bit 3 is reserved.</li> <li>● Bits 4, 5 — Input or output (both must be in same direction).</li> <li>● Bit 6, 7 — Depends on Port C8 mode.</li> </ul>
			*26-27	26-34	Connects INT <sub>B</sub> output to interrupt matrix.		
CC (upper)	0 Input	T: A10	None	*15-16 *19-20 *17-18 *13-14	Connects bit 4 to J1-26. Connects bit 5 to J1-28. Connects bit 6 to J1-30. Connects bit 7 to J1-32.	C8	Port C8 must be in Mode 0 for all four bits to be available.
						CA	Port CA must be in Mode 0 for all four bits to be available.
CC (lower)	0 Input	T: A11	None	*26-27 *28-29 *30-31 *32-33	Connects bit 0 to J1-24. Connects bit 1 to J1-22. Connects bit 2 to J1-20. Connects bit 3 to J1-18.	C8	Port C8 must be in Mode 0 for all four bits to be available.
						CA	Port CA must be in Mode 0 for all four bits to be available.
CA (upper)	0 Output (latched)	D: A10	None	Same as for Port CC (upper) mode 0 Input.		C8	Same as for Port CC (upper) Mode 0 Input.
CA (lower)	0 Output (latched)	D: A11	None	Same as for Port CC (lower) Mode 0 Input.		CC	Same as for Port CC (lower) Mode 0 Input.

\*Default jumper connected at the factory.

86-pin connector. (Refer to table 2-1 item 3.) Multibus signal characteristics and methods of implementing a serial or parallel priority resolution scheme for resolving bus contention in a multiple bus master system are described in following paragraphs.



Always turn off the system power supply before installing or removing any board from the backplane. Failure to observe this precaution can cause damage to the board.

### 2-17. SIGNAL CHARACTERISTICS

As shown in figure 1-1, connector P1 interfaces the iSBC 86/12 to the Multibus. Connector P1 pin assignments are listed in table 2-8 and descriptions of the signal functions are provided in table 2-9.

The dc characteristics of the iSBC 86/12 bus interface signals are provided in table 2-10. The ac characteristics of the iSBC 86/12 when operating in the master mode and slave mode are provided in tables 2-11 and 2-12, respectively. Bus exchange timing diagrams are provided in figures 2-3 and 2-4.

### 2-18. SERIAL PRIORITY RESOLUTION

In a multiple bus master system, bus contention can be resolved in an iSBC 604 Modular Backplane and Cardcage by implementing a serial priority resolution scheme as shown in figure 2-5. Due to the propagation delay of the BPRO/ signal path, this scheme is limited to a maximum of three bus masters capable of acquiring and controlling the Multibus. In the configuration shown in figure 2-5, the bus master installed in slot J2 has the highest priority and is able to acquire control of the Multibus at any time because its BPRN/ input is always enabled (tied to ground) through jumpers B and N on the backplane. (See figure 5-3.)

If the bus master in slot J2 desires control of the Multibus, it drives its BPRO/ output high and inhibits the BPRN/ input to all lower-priority bus masters. When finished using the Multibus, the J2 bus master pulls its BPRO/ output low and gives the J3 bus master the opportunity to take control of the Multibus. If the J3 bus master does not desire to control the Multibus at this time, it pulls its BPRO/ output low and gives the lowest priority bus master in slot J4 the opportunity to assume control of the Multibus.

The serial priority scheme can be implemented in a user-designed system bus if the chaining of BPRO/ and BPRN/ signals are wired as shown in figure 5-3.

### 2-19. PARALLEL PRIORITY RESOLUTION

A parallel priority resolution scheme allows up to 16 bus masters to acquire and control the Multibus. Figure 2-6 illustrates one method of implementing such a scheme for resolving bus contention in a system containing eight bus masters installed in an iSBC 604/614. Notice that the two highest and two lowest priority bus masters are shown installed in the iSBC 604.

In the scheme shown in figure 2-6, the priority encoder is a 74148 and the priority decoder is an Intel 8205. Input connections to the priority encoder determine the bus priority, with input 7 having the highest priority and input 0 having the lowest priority. Here, the J3 bus master has the highest priority and the J5 bus master has the lowest priority.

**IMPORTANT:** In a parallel priority resolution scheme, the BPRO/ output must be disabled on all bus masters. On the iSBC 86/12 disable the BPRO/ output signal by removing jumper 151-152. If a similar jumper cannot be removed on the other bus masters, either clip the IC pin that supplies the BPRO/ output signal to the Multibus or cut the signal trace.

### 2-20. POWER FAIL/MEMORY PROTECT CONFIGURATION

A mating connector must be installed in the iSBC 604/604 Modular Cardcage and Backplane to accommodate auxiliary connector P2. (Refer to figure 1-1.) Table 2-2 lists some 60-pin connectors that can be used for this purpose; flat crimp, solder, and wirewrap connector types are listed. Table 2-13 correlates the signals and pin numbers on the connector.

Procure the appropriate mating connector for P2 and secure it in place as follows:

- a. Position holes in P2 mating connector over mounting holes that are in line with corresponding P1 mating connector.
- b. From top of connector, insert two 0.5-inch #4-40 pan head screws down through connector and mounting holes.
- c. Install a flat washer, lock washer, and star-type nut on each screw; then tighten the nuts.

When the mating connector for P2 is in place, wire the power fail signals to the appropriate pins of the connector as listed in table 2-13. The dc characteristics of the signals interfaced via P2 are given in table 2-14. In a typical system, these signals would be wired as follows:

- a. Connect auxiliary signal common and returns for +5V, -5V, and +12V backup batteries to P2 pins 1, 2, 21, and 22.

**Table 2-8. Multibus Connector P1 Pin Assignments**

Pin*	Signal	Function	Pin*	Signal	Function
1	GND	} Ground	44	ADRF/	} Address bus
2	GND		45	ADRC/	
3	+5V	} Power input	46	ADRD/	
4	+5V		47	ADRA/	
5	+5V		48	ADRB/	
6	+5V		49	ADR8/	
7	+12V		50	ADR9/	
8	+12V		51	ADR6/	
9	-5V	} Ground	52	ADR7/	
10	-5V		53	ADR4/	
11	GND	} Data bus	54	ADR5/	
12	GND		55	ADR2/	
13	BCLK/		56	ADR3/	
14	INIT/		57	ADR0/	
15	BPRN/		58	ADR1/	
16	BPRO/		59	DATE/	
17	BUSY/		60	DATF/	
18	BREQ/		61	DATC/	
19	MRDC/		62	DATD/	
20	MWTC/		63	DATA/	
21	IORC/		64	DATB/	
22	IOWC/		65	DAT8/	
23	XACK/		66	DAT9/	
24	INH1/		67	DAT6/	
25			68	DAT7/	
26		69	DAT4/		
27	BHEN/	70	DAT5/		
28	ADR10/	71	DAT2/		
29	CBRQ/	72	DAT3/		
30	ADR11/	73	DAT0/		
31	CCLK/	74	DAT1/		
32	ADR12/	75	GND		
33	INTA/	76	GND		
34	ADR13/	77			
35	INT6/	78			
36	INT7/	79	+12V		
37	INT4/	80	-12V		
38	INT5/	81	+5V		
39	INT2/	82	+5V		
40	INT3/	83	+5V		
41	INT0/	84	+5V		
42	INT1/	85	GND		
43	ADRE/	86	GND		

\*All odd-numbered pins (1,3,5 . . . 85) are on component side of the board. Pin 1 is the left-most pin when viewed from the component side of the board with the extractors at the top. All unassigned pins are reserved.

Table 2-9. Multibus Signal Functions

Signal	Functional Description
ADR0/ADRF/ ADR10/-ADR13/	<i>Address.</i> These 20 lines transmit the address of the memory location or I/O port to be accessed. For memory access, ADR0/ (when active low) enables the even byte bank (DAT0/-DAT7/) on the Multibus; i.e., ADR0/ is active low for all even addresses. ADR13/ is the most significant address bit.
BCLK/	<i>Bus Clock.</i> Used to synchronize the bus contention logic on all bus masters. When generated by the iSBC 86/12, BCLK/ has a period of 108.5 nanoseconds (9.22 MHz) with a 35-65 percent duty cycle.
BHEN/	<i>Byte High Enable.</i> When active low, enables the odd byte bank (DAT8/-DATF/) onto the Multibus.
BPRN/	<i>Bus Priority In.</i> Indicates to a particular bus master that no higher priority bus master is requesting use of the bus. BPRN/ is synchronized with BCLK/.
BPRO/	<i>Bus Priority Out.</i> In serial (daisy chain) priority resolution schemes, BPRO/ must be connected to the BPRN/ input of the bus master with the next lower bus priority.
BREQ/	<i>Bus Request.</i> In parallel priority resolution schemes, BREQ/ indicates that a particular bus master requires control of the bus for one or more data transfers. BREQ/ is synchronized with BCLK/.
BUSY/	<i>Bus Busy.</i> Indicates that the bus is in use and prevents all other bus masters from gaining control of the bus. BUSY/ is synchronized with BCLK/.
CBRQ/	<i>Common Bus Request.</i> Indicates that a bus master wishes control of the bus but does not presently have control. As soon as control of the bus is obtained, the requesting bus controller raises the CBRQ/ signal.
CCLK/	<i>Constant Clock.</i> Provides a clock signal of constant frequency for use by other system modules. When generated by the iSBC 86/12, CCLK/ has a period of 108.5 nanoseconds (9.22 MHz) with a 35-65 percent duty cycle.
DAT0/-DATF/	<i>Data.</i> These 16 bidirectional data lines transmit and receive data to and from the addressed memory location or I/O port. DATF/ is the most-significant bit. For data byte operations, DAT0/-DAT7/ is the even byte and DAT8/-DATF/ is the odd byte.
INH1/	<i>Inhibit RAM.</i> For system applications, allows iSBC 86/12 dual port RAM addresses to be overlaid by ROM/PROM or memory mapped I/O devices. This signal has no effect of local CPU access of its dual port RAM.
INIT/	<i>Initialize.</i> Resets the entire system to a known internal state.
INTA/	<i>Interrupt Acknowledge.</i> This signal is issued in response to an interrupt request.
INT0/-INT7/	<i>Interrupt Request.</i> These eight lines transmit Interrupt Requests to the appropriate interrupt handler. INTO has the highest priority.
IORC/	<i>I/O Read Command.</i> Indicates that the address of an I/O port is on the Multibus address lines and that the output of that port is to be read (placed) onto the Multibus data lines.
IOWC/	<i>I/O Write Command.</i> Indicates that the address of an I/O port is on the Multibus address lines and that the contents on the Multibus data lines are to be accepted by the addressed port.
MRDC/	<i>Memory Read Command.</i> Indicates that the address of a memory location is on the Multibus address lines and that the contents of that location are to be read (placed) on the Multibus data lines.
MWTC/	<i>Memory Write Command.</i> Indicates that the address of a memory location is on the Multibus address lines and that the contents on the Multibus data lines are to be written into that location.
XACK/	<i>Transfer Acknowledge.</i> Indicates that the address memory location has completed the specified read or write operation. That is, data has been placed onto or accepted from the Multibus data lines.

Table 2-10. iSBC 86/12 DC Characteristics

Signals	Symbol	Parameter Description	Test Conditions	Min.	Max.	Units
AACK/, XACK/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 16 mA		.04	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -3 mA	2.0		V
	V <sub>IL</sub>	Input Low Voltage			0.8	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.4V		-2.2	mA
	I <sub>IH</sub>	Input Current at High V	V <sub>IN</sub> = 2.4V		-1.4	mA
	*C <sub>L</sub>	Capacitive Load			15	pF
ADR0/-ADRF/ ADR10/-ADR13/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 32 mA		0.55	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = 3 mA	2.4		V
	V <sub>IL</sub>	Input Low Voltage			0.8	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.45V		-0.25	mA
	I <sub>IH</sub>	Input Current at High V	V <sub>IN</sub> = 5.25V		50	μA
	I <sub>LH</sub>	Output Leakage High	V <sub>O</sub> = 5.25V		-0.25	mA
	I <sub>LL</sub>	Output Leakage Low	V <sub>O</sub> = 0.45V		-0.25	mA
*C <sub>L</sub>	Capacitive Load			18	pF	
BCLK/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 59.5 mA		0.5	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -3 mA	2.7		V
	V <sub>IL</sub>	Input Low Voltage			0.8	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.45V		-0.5	mA
	I <sub>IH</sub>	Input Current at High V	V <sub>IN</sub> = 5.25V		40	μA
	*C <sub>L</sub>	Capacitive Load			15	pF
BHEN/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 16 mA		0.4	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -2.0 mA	2.4		V
	V <sub>IL</sub>	Input Low Voltage			0.8	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.4V		1.6	mA
	I <sub>IH</sub>	Input Current at High V	V <sub>IN</sub> = 2.4V		40	μA
	*C <sub>L</sub>	Capacitive Load			15	pF
BPRN/	V <sub>IL</sub>	Input Low Voltage		2.0	0.8	V
	V <sub>IH</sub>	Input High Voltage				V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.4V		-0.5	mA
	I <sub>IH</sub>	Input Current at High V	V <sub>IN</sub> = 5.25V		50	μA
	*C <sub>L</sub>	Capacitive Load			18	pF
BPRO/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 3.2 mA		0.45	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -0.4 mA	2.4		V
	*C <sub>L</sub>	Capacitive Load			15	pF
BREQ/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 20 mA		0.45	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -0.4 mA	2.4		V
	*C <sub>L</sub>	Capacitive Load			10	pF
BUSY/, CBRQ/, INTROUT/ (OPEN COLLECTOR)	V <sub>OL</sub> *C <sub>L</sub>	Output Low Voltage Capacitive Load	I <sub>OL</sub> = 20 mA		0.4 20	V pF

\*Capacitive load values are approximations.

Table 2-10. iSBC 86/12 DC Characteristics (Continued)

Signals	Symbol	Parameter Description	Test Conditions	Min.	Max.	Units
CCLK/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 60 mA		0.5	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -3 mA	2.7		V
	*C <sub>L</sub>	Capacitive Load			15	pF
DAT0/-DATF/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 32 mA		0.45	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -5 mA	2.4		V
	V <sub>IL</sub>	Input Low Voltage			0.80	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.45V		-0.20	mA
	I <sub>LH</sub>	Output Leakage High	V <sub>O</sub> = 5.25V		100	μA
	*C <sub>L</sub>	Capacitive Load			18	pF
INH1/	V <sub>IL</sub>	Input Low Voltage			0.8	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low	V <sub>IN</sub> = 0.5V		-2.0	mA
	I <sub>IH</sub>	Input Current at High	V <sub>IN</sub> = 2.7V		50	μA
	*C <sub>L</sub>	Capacitive Load			18	pF
INIT/ (SYSTEM RESET)	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 44 mA		0.4	V
	V <sub>OH</sub>	Output High Voltage	OPEN COLLECTOR			
	V <sub>IL</sub>	Input Low Voltage			0.8	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.4V		-4.2	mA
	I <sub>IH</sub>	Input Current at High V	V <sub>IN</sub> = 2.4V		-1.4	mA
	*C <sub>L</sub>	Capacitive Load			15	pF
INT0/-INT7	V <sub>IL</sub>	Input Low Voltage			0.8	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.4V		-1.6	mA
	I <sub>IH</sub>	Input Current at High V	V <sub>IN</sub> = 2.4V		40	μA
	*C <sub>L</sub>	Capacitive Load			18	pF
IORC/, IOWC/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 32 mA		0.45	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -5 mA	2.4		V
	I <sub>LH</sub>	Output Leakage High	V <sub>O</sub> = 5.25V		100	μA
	I <sub>LL</sub>	Output Leakage Low	V <sub>O</sub> = 0.45V		-100	μA
	*C <sub>L</sub>	Capacitive Load			15	pF
INTA/, MRDC/, MWTC/	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 30 mA		0.45	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -5 mA	2.4		V
	V <sub>IL</sub>	Input Low Voltage			0.95	V
	V <sub>IH</sub>	Input High Voltage		2.0		V
	I <sub>IL</sub>	Input Current at Low V	V <sub>IN</sub> = 0.45V		-2.0	mA
	I <sub>IH</sub>	Input Current at High V	V <sub>IN</sub> = 5.25		1000	μA
	*C <sub>L</sub>	Capacitive Load			25	pF

\*Capacitive load values are approximations.

**Table 2-11. iSBC 86/12 AC Characteristics (Master Mode)**

Parameter	Minimum (ns)	Maximum (ns)	Description	Remarks
t <sub>AS</sub>	50		Address setup time to command	
t <sub>AH</sub>	50		Address hold time from command	
t <sub>DS</sub>	50		Data setup to write CMD	
t <sub>DHW</sub>	50		Data hold time from write CMD	
t <sub>CY</sub>	198	202	CPU cycle time	
t <sub>CMDR</sub>	430		Read command width	No wait states
t <sub>CMDW</sub>	430		Write command width	With 1 wait state
t <sub>CSWR</sub>	380		Read-to-write command separation	In override mode
t <sub>CSRR</sub>	380		Read-to-read command separation	In override mode
t <sub>CSWW</sub>	580		Write-to-write command separation	In override mode
t <sub>CSRW</sub>	580		Write-to-read command separation	In override mode
t <sub>XACK1</sub>	-55		Command to XACK sample point	In override mode
t <sub>SAM</sub>	202	210	Time between XACK samples	In override mode
t <sub>ACKRD</sub>	115		AACK to valid read data	When AACK is used
t <sub>ACKWT</sub>	205		AACK or write command inactive	When AACK is used
t <sub>DHR</sub>	0		Read data hold time	
t <sub>DXL</sub>	-115		Read data setup to XACK	
t <sub>XKH</sub>	0		XACK hold time	
t <sub>DXL</sub>	0		AACK to XACK turn off delay	
t <sub>BWS</sub>	35	∞	Bus clock low or high intervals	Supplied by system
t <sub>BS</sub>	23		BPRN to BCLK setup time	
t <sub>DBY</sub>		55	BCLK to BUSY delay	
t <sub>NOD</sub>		30	BPRN to BPRO delay	
t <sub>DBO</sub>	35		BCLK/ to bus request	
t <sub>DBO</sub>	40		BCLK/ to bus priority out	
t <sub>BCY</sub>	108	109	Bus clock period (BCLK)	From iSBC 86/12 when terminated
t <sub>BW</sub>	35	74	Bus clock low or high interval	From iSBC 86/12 when terminated
t <sub>INIT</sub>	3000		Initialization width	After all voltages have stabilized

**Table 2-12. iSBC 86/12 AC Characteristics (Slave Mode)**

Parameter	Minimum (ns)	Maximum (ns)	Description	Remarks
t <sub>AS</sub>	50		Address setup to command	From address to command
t <sub>DS</sub>	-200		Write data setup to command	Note 1
t <sub>OBD</sub>		980	On-board memory cycle delay	No refresh
t <sub>ACK</sub>		720	Command to XACK	Notes 1 and 2
t <sub>CMD</sub>	720		Command width	Note 1
t <sub>AH</sub>	0		Address hold time	
t <sub>DHW</sub>	0		Write data hold time	
t <sub>DHR</sub>	0		Read data hold time	
t <sub>XH</sub>	0	65	Acknowledge hold time	Acknowledge turnoff delay
t <sub>ACC</sub>		640	Read to data valid	Note 3
t <sub>IH</sub>	50		Inhibit hold time	Blocks AACK if t <sub>IS</sub> > t <sub>IS</sub> min.
t <sub>IPW</sub>	100		Inhibit pulse width	
t <sub>CY</sub>		920	Cycle time of board	
t <sub>RD</sub>		555	Refresh delay time	
t <sub>DXL</sub>	30		Read data setup to XACK	
t <sub>CS</sub>	200		Command separation	
t <sub>IS</sub>		50	Inhibit setup time	

**NOTES:**

1. No refresh, dual port RAM not busy.
2. Maximum = t<sub>RD</sub> + t<sub>OBD</sub> + t<sub>ACK</sub>.
3. Maximum access = t<sub>ACC</sub> + t<sub>OBD</sub> + t<sub>RD</sub>.

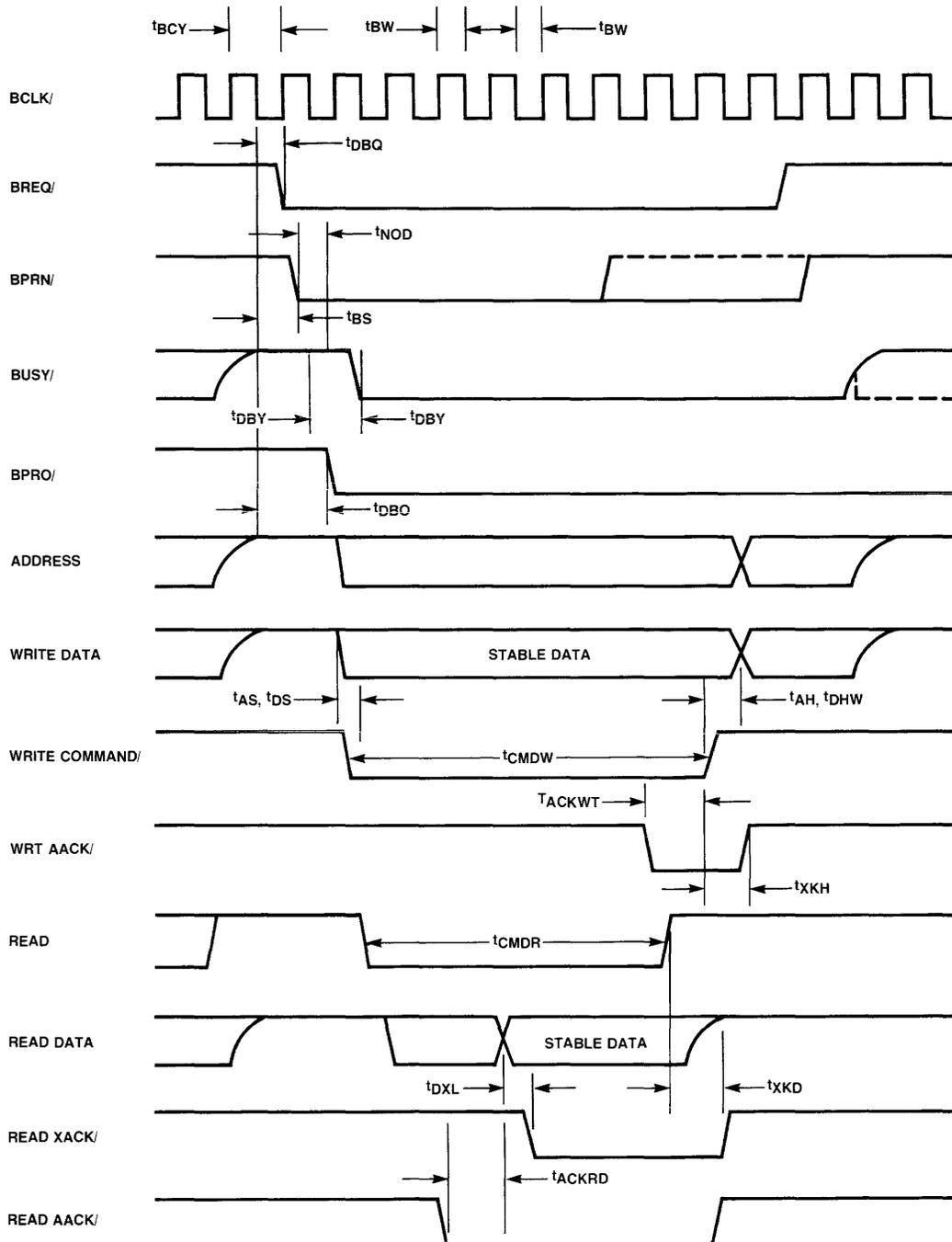
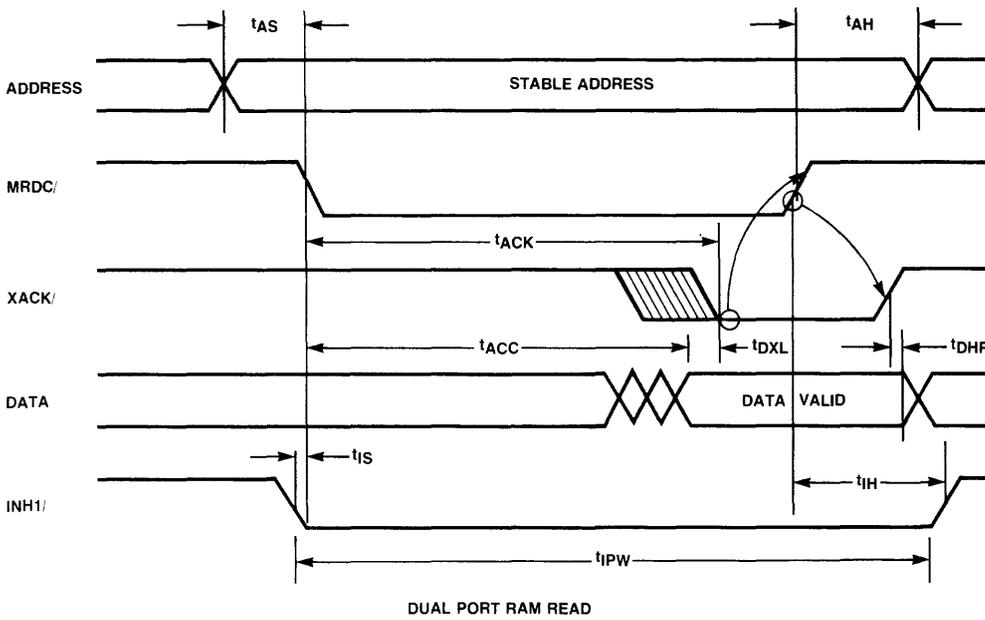
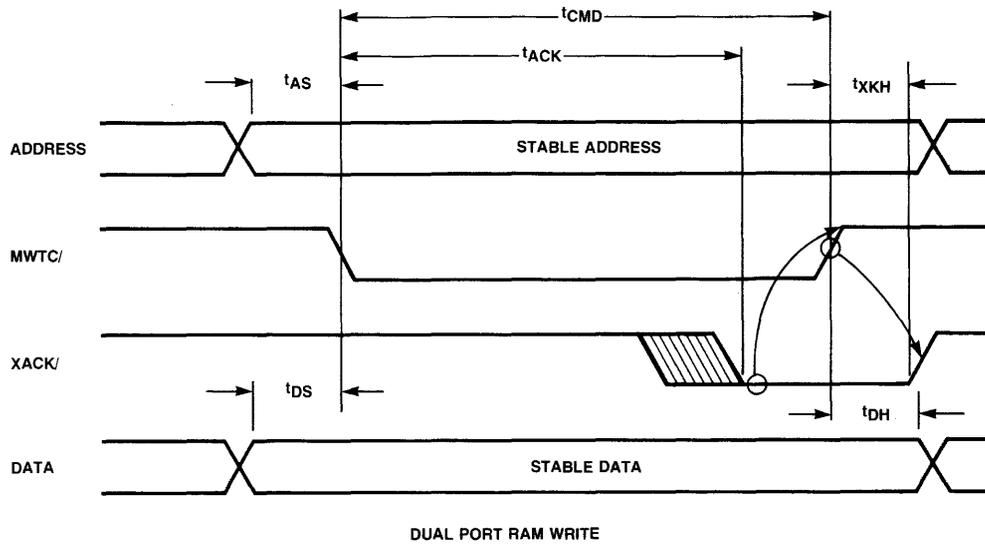
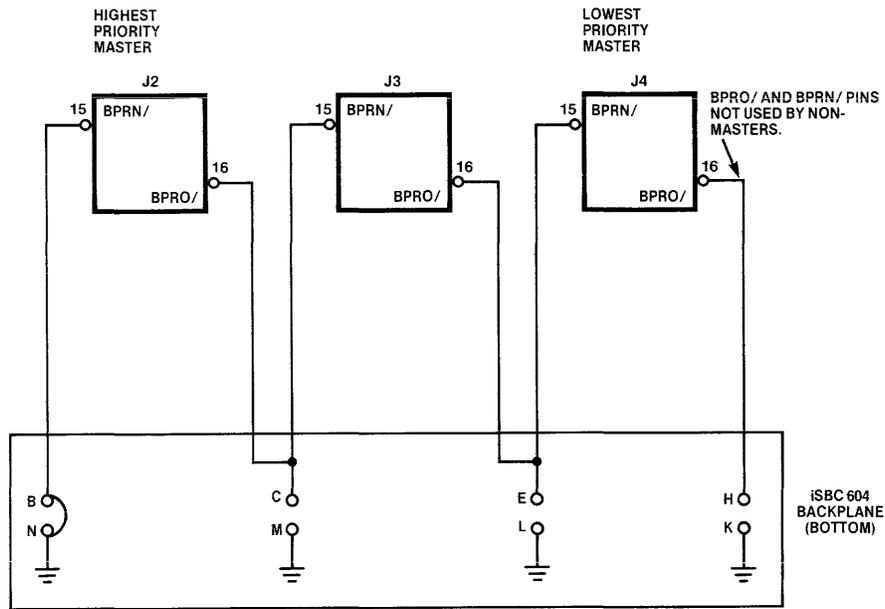


Figure 2-3. Bus Exchange Timing (Master Mode)



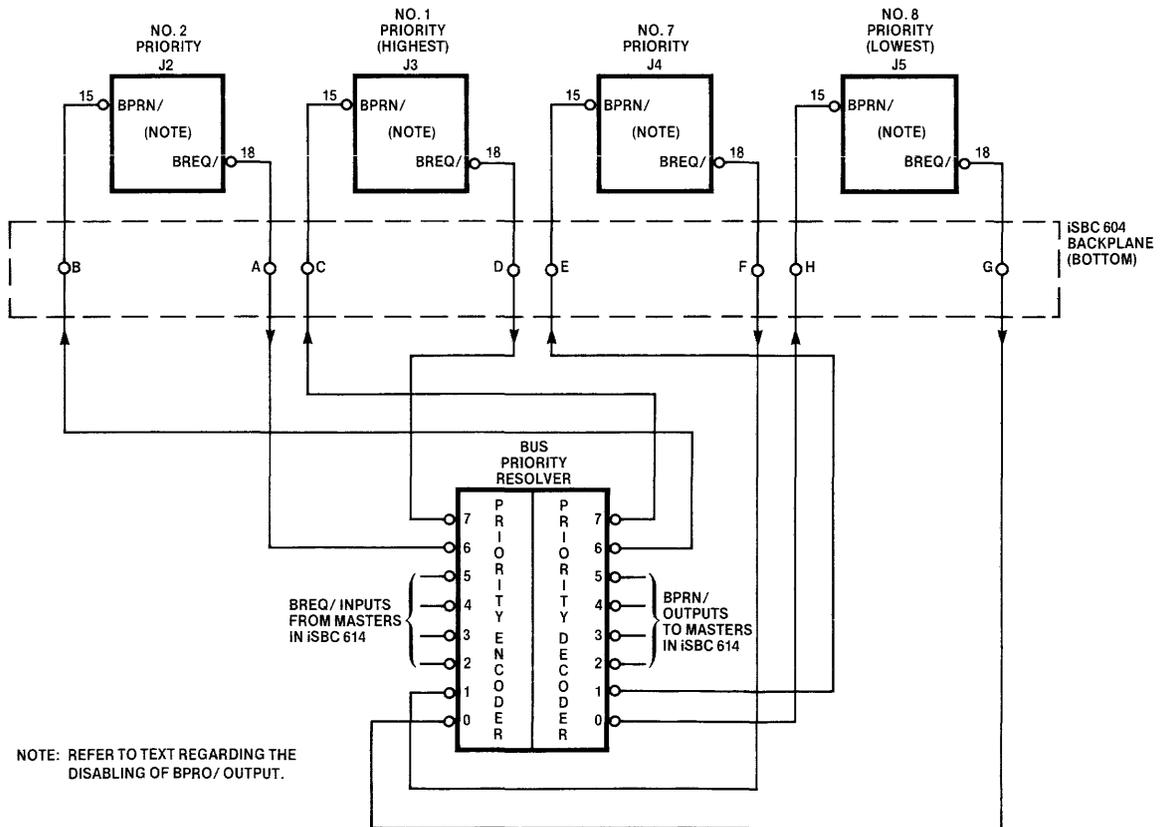
611-5

Figure 2-4. Bus Exchange Timing (Slave Mode)



484-2

Figure 2-5. Serial Priority Resolution Scheme



484-1

Figure 2-6. Parallel Priority Resolution Scheme

**Table 2-13. Auxiliary Connector P2 Pin Assignments**

Pin*	Signal	Definition	
1 2	GND GND	} Auxiliary common	
3 4 7 8	+5V AUX +5V AUX -5V AUX -5V AUX		} Auxiliary backup battery supply
11 12	+12V AUX +12V AUX		
19	PFI/	<i>Power Fail Interrupt.</i> This externally generated signal, which is input to the priority interrupt jumper matrix, should normally be connected to the 8086 CPU NMI input.	
20	MEM PROT/	<i>Memory Protect.</i> This externally generated signal prevents access to the dual port RAM during backup battery operation.	
21 22	GND GND	} Auxiliary common.	
32	ALE		<i>Address Latch Enable.</i> The iSBC 86/12 activates ALE during T1 of every CPU/ machine cycle. This signal may be used as an auxiliary address latch.
38	AUX RESET/	<i>Reset.</i> The externally generated signal initiates a power-up sequence; i.e., initializes the iSBC 86/12 and resets the entire system to a known internal state.	

\*All odd-numbered pins (1,3,5 . . . 59) are on component side of the board. Pin 1 is the left-most pin when viewed from the component side of the board with the extractors at the top.

**Table 2-14. Auxiliary Signal (Connector P2) DC Characteristics**

Signals	Symbol	Parameter Description	Test Conditions	Min.	Max.	Units
ALE	V <sub>OL</sub>	Output Low Voltage	I <sub>OL</sub> = 8 mA		0.45	V
	V <sub>OH</sub>	Output High Voltage	I <sub>OH</sub> = -1.0 mA	2.4		V
	*C <sub>L</sub>	Capacitive Load			20	pF
PFI	V <sub>IL</sub>	Input Low Voltage	V <sub>IN</sub> = 0.4V V <sub>IN</sub> = 2.4V	2.4	0.8	V
	V <sub>IH</sub>	Input High Voltage			V	
	I <sub>IL</sub>	Input Current at Low V			-0.4	mA
	I <sub>IH</sub>	Input Current at High V			20	μ
	*C <sub>L</sub>	Capacitive Load			20	pF
MEM PROT/	V <sub>IL</sub>	Input Low Voltage	V <sub>IN</sub> = 0.45V V <sub>IN</sub> = 5.25V	2.0	0.80	V
	V <sub>IH</sub>	Input High Voltage			V	
	I <sub>IL</sub>	Input Current at Low V			-6.0	mA
	I <sub>IH</sub>	Input Current at High V			250	μA
	*C <sub>L</sub>	Capacitive Load			15	pF
RESET/	V <sub>IL</sub>	Input Low Voltage	V <sub>IN</sub> = 0.45V V <sub>IN</sub> = 5.25V	2.6	0.8	V
	V <sub>IH</sub>	Input High Voltage			V	
	I <sub>IL</sub>	Input Current at Low V			-0.25	mA
	I <sub>IH</sub>	Input Current at High V			10	μA
	*C <sub>L</sub>	Capacitive Load			10	μF

\*Capacitance load values are approximations.

- b. Connect +5V battery input to P2 pins 3 and 4, -5V battery input to P2 pins 7 and 8, and +12V battery input to P2 pins 11 and 12. Remove jumpers W4, W5, and W6.
- c. Connect MEM PROT/ input to P2 pin 20.
- d. Connect PFI/ input to P2 pin 19; this signal is inverted and applied to the priority jumper matrix. To assign the PFI/ input the highest priority (8086 NMI input), remove jumper 87-89 and connect jumper 86-89.
- e. Connect RESET/ input to P2 pin 38. This signal is usually supplied by a momentary-closure switch mounted on the system enclosure.
- f. Connect ALE output signal to P2 pin 32.

## 2-21. PARALLEL I/O CABLING

Parallel I/O ports C8, CA, and CC, controlled by the Intel 8255 Programmable Peripheral Interface (PPI), are interfaced via edge connector J1. (Refer to figure 1-1.) Pin assignments for connector J1 are listed in table 2-15; dc characteristics of the parallel I/O signals are given in table 2-16. Table 2-2 lists some 50-pin edge connectors that can be used for interface to J1 and J2; flat crimp, solder, and wirewrap connector types are listed.

The transmission path from the I/O source to the iSBC 86/12 should be limited to 3 meters (10 feet) maximum. The following bulk cable types (or equivalent) are recommended for interfacing with the parallel I/O ports:

- a. Cable, flat, 50-conductor, 3M 3306-50.
- b. Cable, flat, 50-conductor (with ground plane), 3M 3380-50.
- c. Cable, woven, 25-pair, 3M 3321-25.

An Intel iSBC 956 Cable Set, consisting of two cable assemblies, is recommended for parallel I/O interfacing. Both cable assemblies consist of a 50-conductor flat cable with a 50-pin PC connector at one end. When attaching the cable to J1, be sure that the connector is oriented properly with respect to pin 1 on the edge connector. (Refer to the footnote in table 2-15.)

## 2-22. SERIAL I/O CABLING

Pin assignments and signal definitions for RS232C serial I/O interface are listed in table 2-6. An Intel iSBC 955 Cable Set is recommended for RS232C interfacing. One cable assembly consists of a 25-conductor flat cable with a 26-pin PC connector at one end and an RS232C interface connector at the other end. The second cable assembly includes an RS232C connector at one end and has spade lugs at the other end; the spade lugs are used to interface to a teletypewriter. (See Appendix A for ASR33 TTY interface instructions.)

**Table 2-15. Parallel I/O Connector J1 Pin Assignments**

Pin*	Function	Pin*	Function	
1	↑ Ground ↓	2	Port CA bit 7	
3		4	Port CA bit 6	
5		6	Port CA bit 5	
7		8	Port CA bit 4	
9		10	Port CA bit 3	
11		12	Port CA bit 2	
13		14	Port CA bit 1	
15		16	Port CA bit 0	
17		↑ Ground ↓	18	Port CC bit 3
19			20	Port CC bit 2
21			22	Port CC bit 1
23			24	Port CC bit 0
25			26	Port CC bit 4
27			28	Port CC bit 5
29	30		Port CC bit 6	
31	32		Port CC bit 7	
33	↑ Ground ↓		34	Port C8 bit 7
35			36	Port C8 bit 6
37			38	Port C8 bit 5
39			40	Port C8 bit 4
41			42	Port C8 bit 3
43			44	Port C8 bit 2
45		46	Port C8 bit 1	
47		48	Port C8 bit 0	
49		Ground	50	EXT INTRO/

\*All odd-numbered pins (1,3,5, . . . 49) are on component side of the board. Pin 1 is the right-most pin when viewed from the component side of the board with the extractors at the top.

For OEM applications where cables will be made for the iSBC 86/12, it is important to note that the mating connector for J2 has 26 pins whereas the RS232C connector has 25 pins. Consequently, when connecting the 26-pin mating connector to 25-conductor flat cable, be sure that the cable makes contact with pins 1 and 2 of the mating connector and not with pin 26. Table 2-17 provides pin correspondence between connector J2 and an RS232C connector. When attaching the cable to J2, be sure that the PC connector is oriented properly with respect to pin 1 on the edge connector. (Refer to the footnote in table 2-6.)

## 2-23. BOARD INSTALLATION



Always turn off the computer system power supply before installing or removing the iSBC 86/12 board and before installing or

**Table 2-16. Parallel I/O Signal (Connector J1) DC Characteristics**

Signals	Symbol	Parameter Description	Test Conditions	Min.	Max.	Units
Port C8 Bidirectional Drivers	$V_{OL}$	Output Low Voltage	$I_{OL} = 20 \text{ mA}$	2.4	0.45	V
	$V_{OH}$	Output High Voltage	$I_{OH} = -12 \text{ mA}$		V	
	$V_{IL}$	Input Low Voltage	$V_{IN} = 0.45\text{V}$	2.0	0.95	V
	$V_{IH}$	Input High Voltage			V	
	$I_{IL}$	Input Current at Low V			-5.25	mA
	$*C_L$	Capacitive Load			18	pF
8255A Driver/Receiver	$V_{OL}$	Output Low Voltage	$I_{OL} = 1.7 \text{ mA}$	2.4	0.45	V
	$V_{OH}$	Output High Voltage	$I_{OH} = -200 \mu\text{A}$		V	
	$V_{IL}$	Input Low Voltage	$V_{IN} = 0.45$	2.0	0.8	V
	$V_{IH}$	Input High Voltage			V	
	$I_{IL}$	Input Current at Low V			10	$\mu\text{A}$
	$I_{IH}$	Input Current at High V	$V_{IN} = 5.0$	10	$\mu\text{A}$	
	$*C_L$	Capacitive Load			18	pF
EXT INTRO/	$V_{IL}$	Input Low Voltage	$V_{IN} = 0.4\text{V}$ $V_{IN} = 2.4\text{V}$	2.0	0.8	V
	$V_{IH}$	Input High Voltage			V	
	$I_{IL}$	Input Current at Low V			-1.0	mA
	$I_{IH}$	Input Current at High V			-0.8	mA
	$*C_L$	Capacitive Load			30	pF

\*Capacitive load values are approximations.

removing device interface cables. Failure to take these precautions can result in damage to the board.

**Table 2-17. Connector J2 Vs RS232C Pin Correspondence**

PC Conn. J3	RS232C Conn.	PC Conn. J3	RS232C Conn.
1	14	14	7
2	1	15	21
3	15	16	8
4	2	17	22
5	16	18	9
6	3	19	23
7	17	20	10
8	4	21	24
9	18	22	11
10	5	23	25
11	19	24	12
12	6	25	N/C
13	20	26	13

**NOTE**

Inspect the modular backplane and cardcage and ensure that pull-up resistors have been included for pins 27, 28, 30, 32, 33, and 34. Earlier backplanes did not include pull-ups on these pins.

In an iSBC 80 Single Board Computer based system, install the iSBC 86/12 in any slot that has not been wired for a dedicated function. In an Intellec System, install the iSBC 86/12 in any odd-numbered slot except slot 1. If another module in the Intellec System is to supply the BCLK/ and CCLK/ signals, disconnect 105-106 and 103-104 jumpers on the iSBC 86/12. Make sure that auxiliary connector P2 (if used) mates with the user-installed mating connector. Attach the appropriate cable assemblies to connectors J1 and J2.



## 3-1. INTRODUCTION

This chapter lists the dual port RAM, ROM/EPROM, and I/O address assignments, describes the effects of a hardware initialization (power-up and reset), and provides programming information for the following programmable chips:

- a. Intel 8251A USART (Universal Synchronous/Asynchronous Receiver/Transmitter) that controls the serial I/O port.
- b. Intel 8253 PIT (Programmable Interval Timer) that controls various frequency and timing functions.
- c. Intel 8255A PPI (Programmable Peripheral Interface) that controls the three parallel I/O ports.
- d. Intel 8259A PIC (Programmable Interrupt Controller) that can handle up to 64 vectored priority interrupts for the on-board microprocessor.

This chapter also discusses the Intel 8086 Microprocessor (CPU) interrupt capability. A complete description of programming with Intel's assembly language is given in the *8086 Assembly Language Reference Manual*, Manual Order No. 9800640.

## 3-2. FAILSAFE TIMER

The 8086 CPU expects an acknowledge signal to be returned from the addressed I/O or memory device in response to each Read or Write Command. The iSBC 86/12 includes a Failsafe Timer that is triggered during T1 of every machine cycle. If the Failsafe Timer is enabled by hardware jumper as described in table 2-4, and no acknowledge signal is received within approximately 6 milliseconds after the command is issued, the Failsafe Timer will time out and allow the CPU to exit the wait state. As described in Chapter 2, provision is made so that the Failsafe Timer output (TIME OUT/) can optionally be used to interrupt the CPU.

If the Failsafe Timer is not enabled by hardware jumper and an acknowledge signal is not returned for any reason, the CPU will hang up in a wait state. In this situation, the only way to free the CPU is to initialize the system as described in paragraph 3-7.

## 3-3. MEMORY ADDRESSING

The iSBC 86/12 includes 32K bytes of dynamic random access memory (RAM) and four IC sockets to accom-

modate up to 16K bytes of user-installed read-only memory (ROM or EPROM). The iSBC 86/12 features a dual port RAM access arrangement in which the on-board RAM can be accessed by the on-board 8086 microprocessor (CPU) or by another bus master via the Multibus. The ROM/EPROM can be accessed only by the CPU.

The dual port RAM can be accessed by another bus master that currently has control of the Multibus. It should be noted that, even though another bus master may be continually accessing the dual port RAM, this does not prevent the CPU from also accessing the dual port RAM. When this situation occurs, memory accesses by the CPU and controlling bus master are interleaved. Such interleaved access will, of course, impose a longer wait state both for the CPU and for the controlling bus master. Dual-port RAM access by another bus master does not interfere with the CPU while it is accessing the on-board ROM/EPROM and I/O devices.

## 3-4. CPU ACCESS

Addresses for CPU access of ROM/EPROM and on-board RAM are provided in table 3-1. Note that the ROM/EPROM addresses are assigned from the top down of the 1-megabyte address space with the bottom address being determined by the user ROM/EPROM configuration. The on-board RAM addresses are assigned from the bottom up of the 1-megabyte address space.

When the CPU is addressing *on-board* memory (RAM, ROM, or EPROM), an internal acknowledge signal is automatically generated and imposes one wait state for each CPU operation. When the CPU is addressing *system* memory via the Multibus, the CPU must first gain control of the Multibus and, after the Memory Read or Memory Write Command is given, must wait for a Transfer Acknowledge (XACK/) to be received from the addressed memory device. The Failsafe Timer, if enabled, will prevent a CPU hang-up in the event of a memory device equipment failure or a bus failure.

It should be noted in table 3-1 that it is possible to configure ROM/EPROM such as to create *illegal* addresses. If an illegal address is used in conjunction with a Memory Write Command to ROM/EPROM, an internal acknowledge signal is generated as though the address was legal and the CPU will continue executing the program. However, in this case, erroneous data will be returned.

Table 3-1. On-Board Memory Addresses (CPU Access)

Type	Configuration	Legal Addresses	Illegal Addresses
EPROM	Two 2758 chips Four 2758 chips	FF800-FFFF FF000-FFFF	FF000-FF7FF —
	Two 2716 chips Four 2716 chips	FF000-FFFF FE000-FFFF	FE000-FEFFF —
ROM	Two 2316E chips Four 2316E chips	FF000-FFFF FE000-FFFF	FE000-FEFFF —
	Two 2332 chips Four 2332 chips	FE000-FFFF FC000-FFFF	FC000-FDFFF —
RAM	Sixteen 2117 chips	0000-07FFF	—

### 3-5. MULTIBUS ACCESS

As described in paragraph 2-12, the iSBC 86/12 can be configured to permit Multibus access of 8K, 16K, 24K, or 32K bytes of on-board RAM. The Multibus allows both 8-bit and 16-bit masters to reside in the same system and, to accomplish this, the memory is divided into two 8-bit data banks to form one 16-bit word. The banks are organized such that all even bytes are in one bank (DAT0-DAT7) and all odd bytes are in the other bank (DAT8-DATF).

The Byte High Enable (BHEN/) signal controls the odd data byte and, when active, enables the high byte (DAT8/-DATF/) onto the Multibus. Address bit ADR0/ controls the even data byte and, when active, enables the low byte (DAT0/-DAT7/) onto the Multibus. For maximum efficiency, 16-bit word operations must occur on an even byte boundary with BHEN/ active. Address bit ADR0/ is active for all even byte addresses. Odd byte addressing requires two operations to form a 16-bit word.

Byte operations can occur in two ways. The even byte can be accessed by controlling ADR0/, which places the data on the DAT0/-DAT7/ lines. (See figure 3-1A.) To access the odd data bank, which normally is placed on the DAT8/-DATF/ lines, a new data path is defined. The inactive state of ADR0/ and BHEN/ enable a swap byte buffer that places the odd data bank on DAT0/-DAT7/. (See figure 3-1B.) This permits an 8-bit bus master to access both bytes of a data word by controlling only ADR0/.

Figure 3-1C illustrates how a 16-bit bus master obtains a 16-bit word by a single address on an even byte boundary. Figure 3-1A illustrates how a 16-bit bus master may selectively address an even (low) data byte.

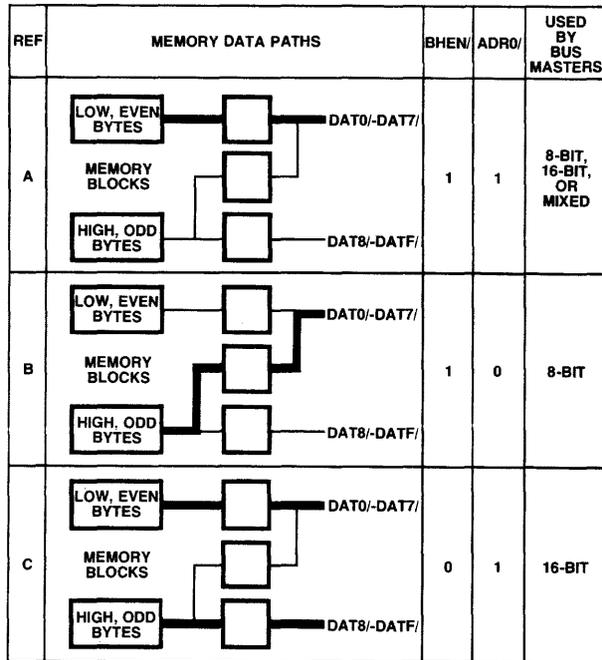


Figure 3-1. Dual Port RAM Addressing (Multibus Access)

645-4

### 3-6. I/O ADDRESSING

The CPU communicates with the on-board programmable chips through a sequence of I/O Read and I/O Write Commands. As shown in table 3-2, each of these chips recognizes four separate hexadecimal I/O addresses that are used to control the various programmable functions. (The I/O address decoder operates on the lower eight bits and all addresses must be on an even byte boundary.) Where two hexadecimal addresses are listed for a single function, either address may be used. For example, an I/O Read Command to 000DA or 000DE will read the status of the 8251A USART.

### 3-7. SYSTEM INITIALIZATION

When power is initially applied to the system, a reset signal is automatically generated that performs the following:

- a. The 8086 CPU internal registers are set as follows:
    - PSW = 0000
    - IP = 0000
    - DS = 0000
    - ES = 0000
    - Code Relocation Register = FFFF
- This effectively causes a long JMP to FFFF0.

Table 3-2. I/O Address Assignments

I/O Address*	Chip Select	Function
000C0 or 000C4	8259A PIC	Write: ICW1, OCW2, and OCW3 Read: Status and Poll
000C2 or 000C6		Write: ICW2, ICW3, ICW4, OCW1 (Mask) Read: OCW1 (Mask)
000C8	8255A PPI	Write: Port A (J1) Read: Port A (J1)
000CA		Write: Port B (J1) Read: Port B (J1)
000CC		Write: Port C (J1) Read: Port C Status
000CE		Write: Control Read: None
000D0	8253 PIT	Write: Counter 0 (Load Count ÷ N) Read: Counter 0
000D2		Write: Counter 1 (Load Count ÷ N) Read: Counter 1
000D4		Write: Counter 2 (Load Count ÷ N) Read: Counter 2
000D6		Write: Control Read: None
000D8 or 000DC	8251A USART	Write: Data (J2) Read: Data (J2)
000DA or 000DE		Write: Mode or Command Read: Status
*Odd addresses (i.e., 000C1, 000C3, . . . . 000DD) are illegal.		

- b. The 8251A USART serial I/O port is set to the "idle" mode, waiting for a set of Command Words to program the desired function.
- c. The 8255A PPI parallel I/O ports are set to the input mode.

The 8253 PIT and the 8259 PIC are not affected by the power-up sequence.

The reset signal is also gated onto the Multibus to initialize the remainder of the system components to a known internal state.

The reset signal can also be generated by an auxiliary RESET switch. Pressing and releasing the RESET switch produces the same effect as the power-up reset described above.

### 3-8. 8251A USART PROGRAMMING

The USART converts parallel output data into virtually any serial output data format (including IBM Bi-Sync) for half- or full-duplex operation. The USART also converts serial input data into parallel data format.

Prior to starting transmitting or receiving data, the USART must be loaded with a set of control words. These control words, which define the complete functional operation of the USART, must immediately follow a reset (internal or external). The control words are either a Mode instruction or a Command instruction.

### 3-9. MODE INSTRUCTION FORMAT

The Mode instruction word defines the general characteristics of the USART and must follow a reset operation. Once the Mode instruction word has been written into the USART, sync characters or command instructions may be inserted. The Mode instruction word defines the following:

- a. For Sync Mode:
  - (1) Character length
  - (2) Parity enable
  - (3) Even/odd parity generation and check
  - (4) External sync detect (not supported by 86/1X)
  - (5) Single or double character sync
- b. For Async Mode:
  - (1) Baud rate factor (X1, X16, or X64)
  - (2) Character length
  - (3) Parity enable
  - (4) Even/odd parity generation and check
  - (5) Number of stop bits

Instruction word and data transmission formats for synchronous and asynchronous modes are shown in figures 3-2 through 3-5.

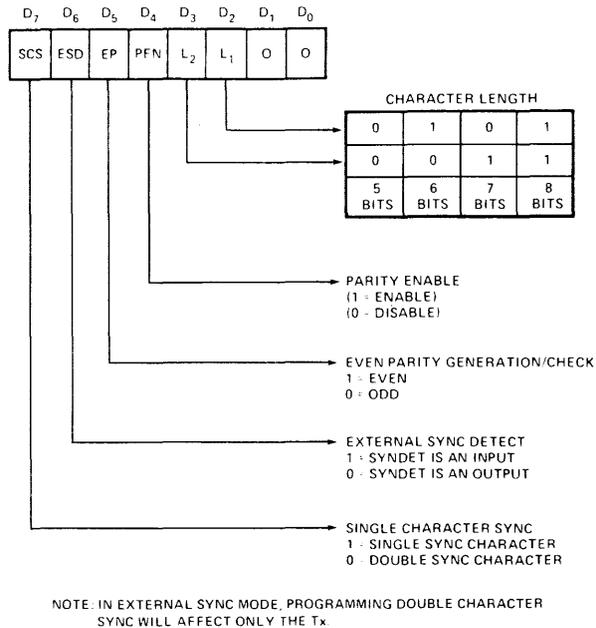


Figure 3-2. USART Synchronous Mode Instruction Word Format

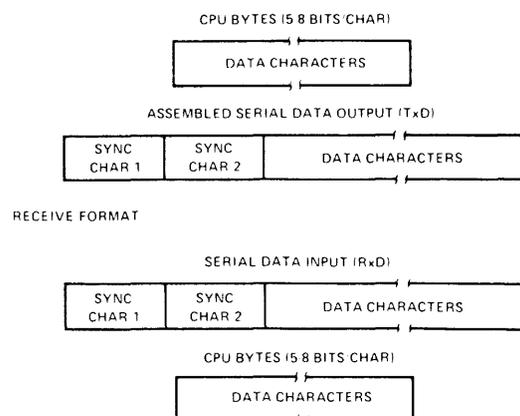


Figure 3-3. USART Synchronous Mode Transmission Format

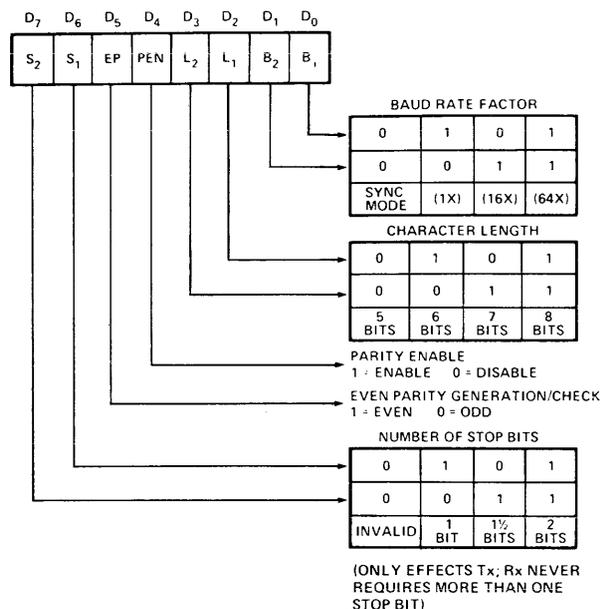
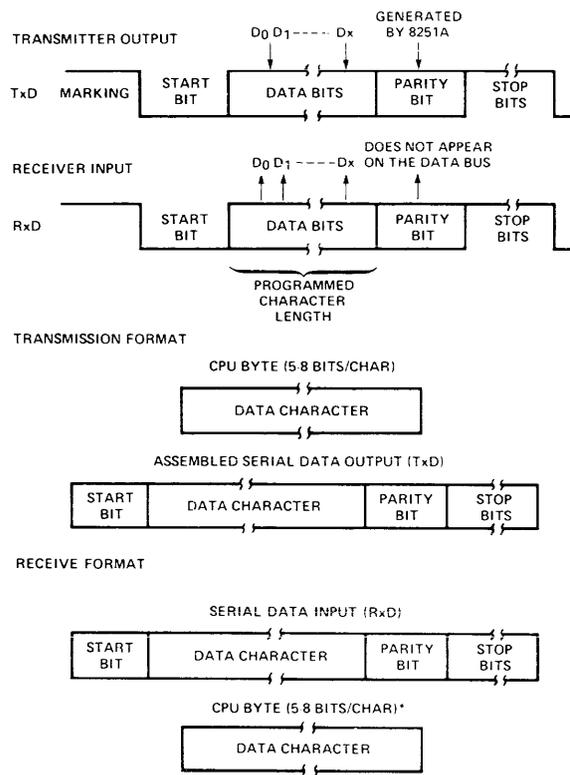


Figure 3-4. USART Asynchronous Mode Instruction Word Format



\*NOTE: IF CHARACTER LENGTH IS DEFINED AS 5, 6 OR 7 BITS THE UNUSED BITS ARE SET TO "ZERO"

Figure 3-5. USART Asynchronous Mode Transmission Format

3-10. SYNC CHARACTERS

Sync characters are written to the USART in the synchronous mode only. The USART can be programmed to either one or two sync characters; the format of the sync characters is at the option of the programmer.

3-11. COMMAND INSTRUCTION FORMAT

The Command instruction word shown in figure 3-6 controls the operation of the addressed USART. A Command instruction must follow the mode and/or sync words. Once the Command instruction is written, data can be transmitted or received by the USART.

It is not necessary for a Command instruction to precede all data transactions; only those transmissions that require a change in the Command instruction. An example is a change in the enable transmit or enable receive bus. Command instructions can be written to the USART at any time after one or more data operations.

After initialization, always read the chip status and check for the TXRDY bit prior to writing either data or command words to the USART. This ensures that any prior input is not overwritten and lost. Note that issuing a

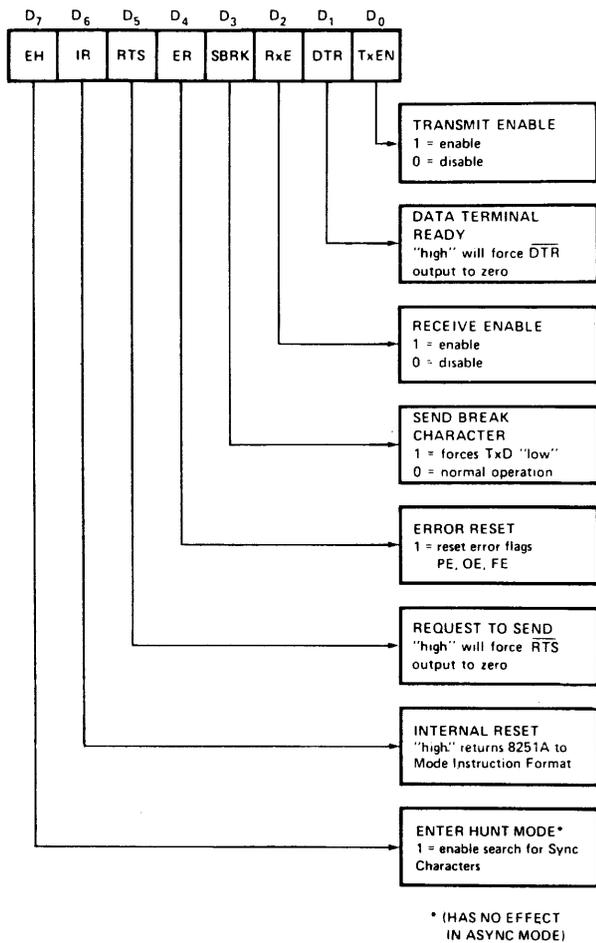
Command instruction with bit 6 (IR) set will return the USART to the Mode instruction format.

3-12. RESET

To change the Mode instruction word, the USART must receive a Reset command. The next word written to the USART after a Reset command is assumed to be a Mode instruction. Similarly, for sync mode, the next word after a Mode instruction is assumed to be one or more sync characters. All control words written into the USART after the Mode instruction (and/or the sync character) are assumed to be Command instructions.

3-13. ADDRESSING

The USART chip uses address 000D8 or 000DC to read and write I/O data; address 000DA or 000DE is used to write mode and command words and read the USART status. (Refer to table 3-2.)



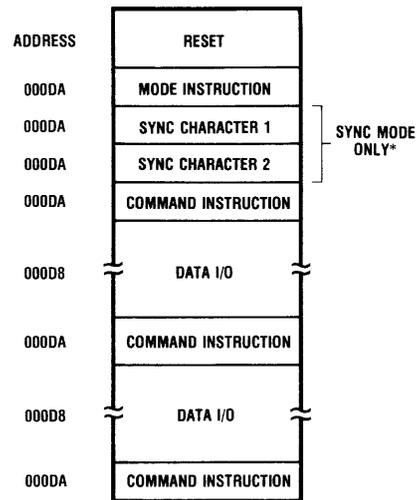
**Note:** Error Reset must be performed whenever RxEnable and Enter Hunt are programmed.

**Figure 3-6. USART Command Instruction Word Format**

### 3-14. INITIALIZATION

A typical USART initialization and I/O data sequence is presented in figure 3-7. The USART chip is initialized in four steps:

- a. Reset USART to Mode instruction format.
- b. Write Mode instruction word. One function of mode word is to specify synchronous or asynchronous operation.
- c. If synchronous mode is selected, write one or two sync characters as required.
- d. Write Command instruction word.



\*The second sync character is skipped if Mode instruction has programmed USART to single character internal sync mode. Both sync characters are skipped if Mode instruction has programmed USART to async mode.

**Figure 3-7. Typical USART Initialization and Data I/O Sequence**

645-5

To avoid spurious interrupts during USART initialization, disable the USART interrupt. This can be done by either masking the appropriate interrupt request input at the 8259A PIC or by disabling the 8086 microprocessor interrupts by executing a DI instruction.

First, reset the USART chip by writing a Command instruction to location 000DA (or 000DE). The Command instruction must have bit 6 set (IR = 1); all other bits are immaterial.

#### NOTE

This reset procedure should be used only if the USART has been completely initialized, or the initialization procedure has reached the point that the USART is ready to receive a Command word. For example, if the reset command is written when the initialization sequence calls for a sync character, then subsequent programming will be in error.

Next write a Mode instruction word to the USART. (See figures 3-2 through 3-5.) A typical subroutine for writing both Mode and Command instructions is given in table 3-3.

If the USART is programmed for the synchronous mode, write one or two sync characters depending on the transmission format.

**Table 3-3. Typical USART Mode or Command Instruction Subroutine**

;CMD2 OUTPUTS CONTROL WORD TO USART.			
;USES-A, STAT2; DESTROYS-NOTHING.			
	PUBLIC	CMD2	
	EXTRN	STAT2	
CMD2:	LAHF		
	PUSH	AX	
LP:	CALL	STAT2	
	AND	AL,1	;CHECK TXRDY
	JZ	LP	;TXRDY MUST BE TRUE
	POP	AX	
	SAHF		
51INT:	OUT	0DAH	;ENTER HERE FOR INITIALIZATION
	RET		
	END		

Finally, write a Command instruction word to the USART. Refer to figure 3-6 and table 3-3.

**IMPORTANT:** During initialization, the 8251A USART requires a minimum recovery time of 3.2 microseconds (16 clock cycles) between back-to-back writes in order to set up its internal registers. This recovery time can be satisfied by the CPU performing two byte reads and a NOP between the back-to-back writes to the 8251A USART as follows:

```

OUT  0DAH  ;FIRST USART WRITE
SAHF          ;TWO-BYTE READ
NOP          ;ADDED WAIT
OUT  0DAH  ;SECOND USART WRITE

```

This precaution applies only to the USART initialization and does not apply otherwise.

### 3-15. OPERATION

Normal operating procedures use data I/O read and write, status read, and Command instruction write operations. Programming and addressing procedures for the above are summarized in following paragraphs.

#### NOTE

After the USART has been initialized, always check the status of the TXRDY bit *prior* to writing data or writing a new command word to the USART. The TXRDY bit *must* be *true* to prevent overwriting and subsequent loss of command or data words. The TXRDY bit is inactive until initialization has been completed; do not check TXRDY until after the command word, which concludes the initialization procedure, has been written.

Prior to any operating change, a new command word must be written with command bits changed as appropriate. (Refer to figure 3-6 and table 3-3.)

**3-16. DATA INPUT/OUTPUT.** For data receive or transmit operations, perform a read or write, respectively, to the USART. Table 3-4 and 3-5 provide examples of typical character read and write subroutines.

During normal transmit operation, the USART generates a Transmit Ready (TXRDY) signal that indicates that the USART is ready to accept a data character for transmission. TXRDY is automatically reset when the CPU loads a character into the USART.

Similarly, during normal receive operation, the USART generates a Receive Ready (RXRDY) signal that indicates that a character has been received and is ready for input to the CPU. RXRDY is automatically reset when a character is read by the CPU.

The TXRDY and RXRDY outputs of the USART are available at the priority interrupt jumper matrix. If, for instance, TXRDY and RXRDY are input to the 8259A PIC, the PIC resolves the priority and interrupts the CPU. TXRDY and RXRDY are also available in the status word. (Refer to paragraph 3-16.)

**3-17. STATUS READ.** The CPU can determine the status of a serial I/O port by issuing an I/O Read Command to the upper address (000DA or 000DE) of the USART chip. The format of the status word is shown in figure 3-8. A typical status read subroutine is given in table 3-6.

**Table 3-4. Typical USART Data Character Read Subroutine**

```

;RX1 READS DATA CHARACTER FROM USART.
;USES-STAT0; DESTROYS-A,FLAGS.

                PUBLIC   RX1,RXA1
                EXTRN    STAT0

RX1:            CALL     STAT0
                AND      AL,2                ;CHECK FOR RXRDY TRUE
                JZ       RX1
RXA1:           IN       ODCH                ;ENTER HERE IF RXRDY IS TRUE
                RET

                END
    
```

**Table 3-5. Typical USART Data Character Write Subroutine**

```

;TX1 WRITES DATA CHARACTER FROM REG A TO USART.
;USES-STAT0; DESTROYS-FLAGS.

                PUBLIC   TX1,TXA1
                EXTRN    STAT0

TX1:            PUSH     AX
TX11:           CALL     STAT0
                AND      AL,1                ;CHECK FOR TXRDY TRUE
                JZ       TX11
                POP      AX
TXA1:           OUT      OD8H                ;ENTER HERE IF TXRDY IS TRUE
                RET

                END
    
```

**3-18. 8253 PIT PROGRAMMING**

A 22.1184-MHz crystal oscillator supplies the basic clock frequency for the programmable chips. This clock frequency is divided by 9, 18, and 144 to produce three jumper-selectable clocks: 2.46 MHz, 1.23 MHz, and 153.6 kHz. These clocks are available for input to Counter 0, Counter 1, and Counter 2 of the 8253 PIT. The default (factory connected) and optional jumpers for selecting the clock inputs to the three counters are listed in table 2-4.

Default jumpers connect the output of Counter 2 to the TXC and RXC inputs of the 8251A USART. Jumpers are included so that Counters 0 and 1 can provide real-time interrupts to the 8259A PIC.

Before programming the 8253 PIT, ascertain the input clock frequency and the output function of each of the three counters. These factors are determined and established by the user during the installation.

**3-19. MODE CONTROL WORD AND COUNT**

All three counters must be initialized prior to their use. The initialization for each counter consists of two steps:

- a. A mode control word (figure 3-9) is written to the control register for each individual counter.
- b. A down-count number is loaded into each counter; the down-count number is in one or two 8-bit bytes as determined by mode control word.

The mode control word (figure 3-9) does the following:

- a. Selects counter to be loaded.
- b. Selects counter operating mode.
- c. Selects one of the following four counter read/load functions:
  - (1) Counter latch (for stable read operation).
  - (2) Read or load most-significant byte only.

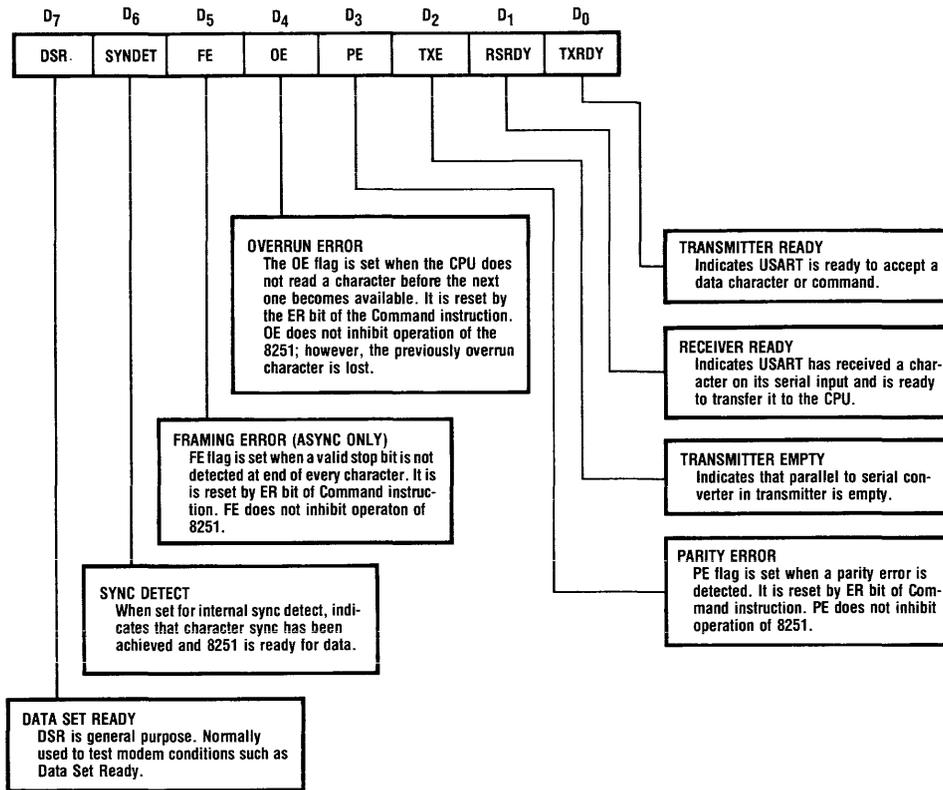


Figure 3-8. USART Status Read Format

450-14

- (3) Read or load least-significant byte only.
- (4) Read or load least-significant byte first, then most-significant byte.
- d. Sets counter for either binary or BCD count.

The mode control word and the count register bytes for any given counter must be entered in the following sequence:

- a. Mode control word.
- b. Least-significant count register byte.
- c. Most-significant count register byte.

As long as the above procedure is followed for each counter, the chip can be programmed in any convenient sequence. For example, mode control words can be loaded first into each of three counters per chip, followed by the least-significant byte, etc. Figure 3-10 shows the two programming sequences described above.

Since all counters in the PIT chip are downcounters, the value loaded in the count registers is decremented. Loading all zeroes into a count register results in a maximum count of  $2^{16}$  for binary numbers of  $10^4$  for BCD numbers.

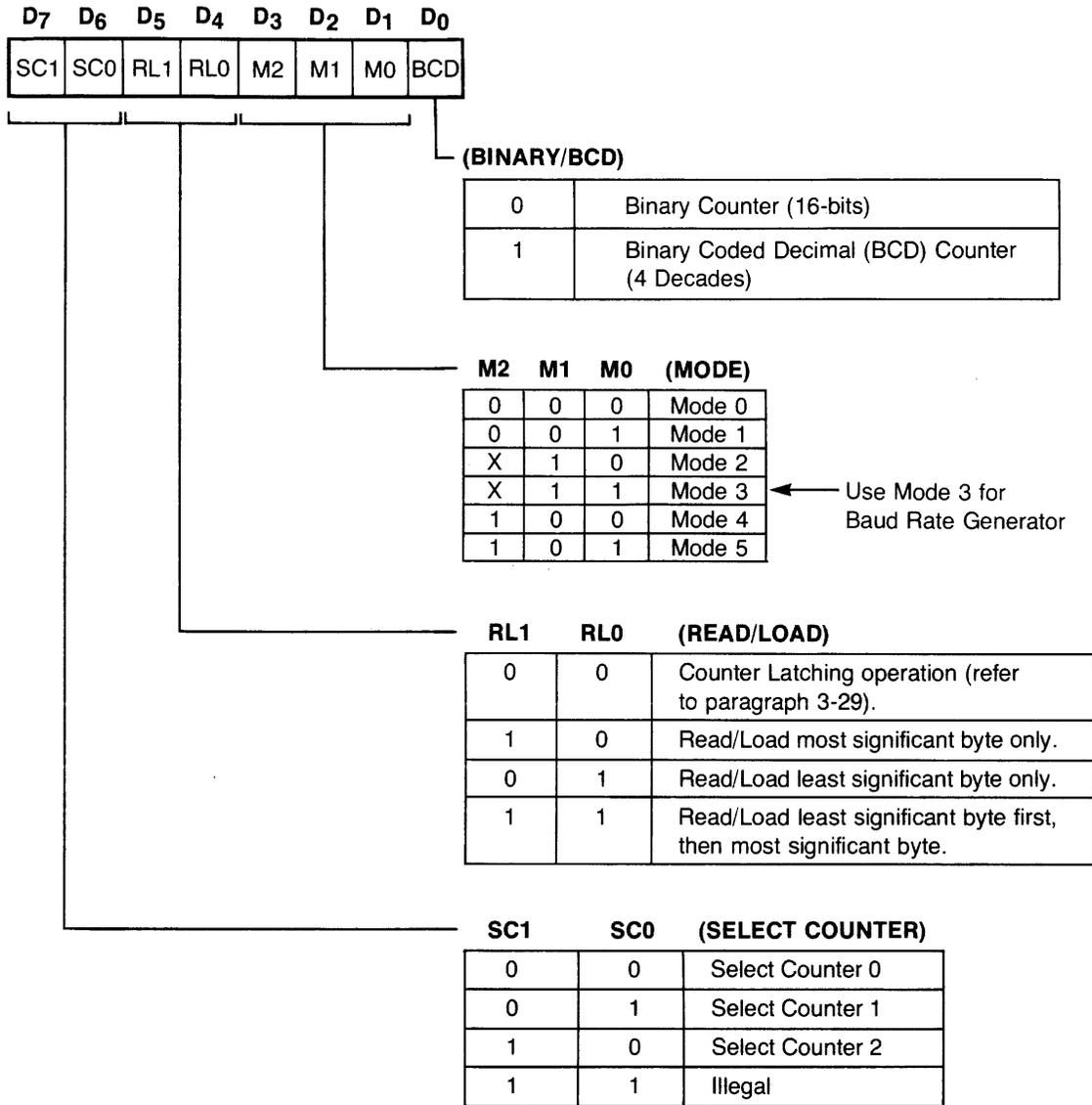
Table 3-6. Typical USART Status Read Subroutine

```

;STAT0 READS STATUS FROM USART.
;DESTROYS-A.

PUBLIC STAT0

STAT0: IN 0DEH ;GET STATUS
RET
END
    
```



611-7

Figure 3-9. PIT Mode Control Word Format

When a selected count register is to be loaded, it *must* be loaded with the number of bytes programmed in the mode control word. One or two bytes can be loaded, depending on the appropriate down count. These two bytes can be programmed at any time following the mode control word, as long as the correct number of bytes is loaded in order.

The count mode selected in the control word controls the counter output. As shown in figure 3-9, the PIT chip can operate in any of six modes:

- a. Mode 0: Interrupt on terminal count. In this mode, Counters 1 and 2 can be used for auxiliary functions, such as generating real-time interrupt intervals. After

the count value is loaded into the count register, the counter output goes low and remains low until the terminal count is reached. The output then goes high until either the count register or the mode control register is reloaded.

- b. Mode 1: Programmable one-shot. In this mode, the output of Counter 1 and/or Counter 2 will go low on the count following the rising edge of the GATE input from Port CC (assuming Port CC jumpers are so configured). The output will go high on the terminal count. If a new count value is loaded while the output is low, it will not affect the duration of the one-shot pulse until the succeeding trigger. The current count can be read at any time without affecting the

**PROGRAMMING FORMAT**

Step		
1		Mode Control Word Counter n
2	LSB	Count Register Byte Counter n
3	MSB	Count Register Byte Counter n

**ALTERNATE PROGRAMMING FORMAT**

Step		
1		Mode Control Word Counter 0
2		Mode Control Word Counter 1
3		Mode Control Word Counter 2
4	LSB	Counter Register Byte Counter 1
5	MSB	Count Register Byte Counter 1
6	LSB	Count Register Byte Counter 2
7	MSB	Count Register Byte Counter 2
8	LSB	Count Register Byte Counter 0
9	MSB	Count Register Byte Counter 0

450-18

**Figure 3-10. PIT Programming Sequence Examples**

- one-shot pulse. The one-shot is retriggerable, hence the output will remain low for the full count after any rising edge of the gate input.
- c. Mode 2: Rate generator. In this mode, the output of Counter 1 and/or Counter 2 will be low for one period of the clock input. The period from one output pulse to the next equals the number of input counts in the count register. If the count register is reloaded between output pulses, the present period will not be affected but the subsequent period will reflect the new value. The gate input, when low, will force the output high. When the gate input goes high, the counter will start from the initial count. Thus, the gate input can be used to synchronize the counter. When Mode 2 is set, the output will remain high until after the counter register is loaded; thus, the count can be synchronized by software.
- d. Mode 3: Square wave generator. Mode 3, which is the primary operating mode for Counter 2, is used for generating Baud rate clock signals. In this mode, the counter output remains high until one-half of the count value in the count register has been decremented (for even numbers). The output then goes low for the other half of the count. If the value in the count register is odd, the counter output is high for  $(N + 1)/2$  counts, and low for  $(N - 1)/2$  counts.
- e. Mode 4: Software triggered strobe. After this mode is set, the output will be high. When the count is loaded, the counter begins counting. On terminal count, the output will go low for one input clock period and then go high again. If the count register is reloaded between output pulses, the present count will not be affected, but the subsequent period will reflect the new value. The count will be inhibited while the gate input is low. Reloading the count register will restart the counting for the new value.
- f. Mode 5: Hardware triggered strobe. Counter 0 and/or Counter 1 will start counting on the rising edge of the gate input and the output will go low for one clock period when the terminal count is reached. The counter is retriggerable. The output will not go low until the full count after the rising edge of the gate input.
- Table 3-7 provides a summary of the counter operation versus the gate inputs. The gate inputs to Counters 0 and 1 are tied high by default jumpers; these gates may optionally be controlled by Port CC. The gate input to Counter 2 is not optionally controlled.

**Table 3-7. PIT Counter Operation Vs Gate Inputs**

Signal Status / Modes	Low Or Going Low	Rising	High
0	Disables counting	—	Enables counting
1	—	1) Initiates counting 2) Resets output after next clock	—
2	1) Disables counting 2) Sets output immediately high	Initiates counting	Enables counting
3	1) Disables counting 2) Sets output immediately high	Initiates counting	Enables counting
4	Disables counting	—	Enables counting
5	—	Initiates counting	—

**3-20. ADDRESSING**

As listed in table 3-2, the PIT uses four I/O addresses. Addresses 000D0, 000D2, and 000D4, respectively, are used in loading and reading the count in Counters 0, 1, and 2. Address 000D6 is used in writing the mode control word to the desired counter.

**3-21. INITIALIZATION**

To initialize the PIT chip, perform the following:

- a. Write mode control word for Counter 0 to 000D6. Note that *all* mode control words are written to 000D6, since mode control word must specify which counter is being programmed. (Refer to figure 3-9.) Table 3-8 provides a sample subroutine for writing mode control words to all three counters.
- b. Assuming mode control word has selected a 2-byte load, load least-significant byte of count into Counter 0 at 000D0. (Count value to be loaded is described in paragraphs 3-23 through 3-25.) Table 3-9 provides a sample subroutine for loading 2-byte count value.
- c. Load most-significant byte of count into Counter 0 at 000D0.

**Table 3-8. Typical PIT Control Word Subroutine**

```

;INTTMR INITIALIZES COUNTERS 0,1,2.
;COUNTERS 0 AND 1 ARE INITIALIZED AS INTERRUPT TIMERS.
;COUNTER 2 IS INITIALIZED AS BAUD RATE GENERATOR.
;ALL THREE COUNTERS ARE SET UP FOR 16-BIT OPERATION.
;DESTROYS-A.

PUBLIC INTTMR

INTTMR: MOV AL,30H ;MODE CONTROL WORD FOR COUNTER 0
        OUT 0D6H
        MOV AL,70H ;MODE CONTROL WORD FOR COUNTER 1
        OUT 0D6H
        MOV AL,B6H ;MODE CONTROL WORD FOR COUNTER 2
        OUT 0D6H
        RET

END
    
```

**Table 3-9. Typical PIT Count Value Load Subroutine**

```

;LOAD0 LOADS COUNTER 0 FROM D&E. D IS MSB, E IS LSB.
;USES-D,E; DESTROYS-A.

PUBLIC LOAD0

LOAD0: MOV AL,EL ;GET LSB
        OUT 0D0H
        MOV AL,DL ;GET MSB
        OUT 0D0H
        RET

END
    
```

**NOTE**

Be sure to enter the downcount in two bytes if the counter was programmed for a two-byte entry in the mode control word. Similarly, enter the downcount value in BCD if the counter was so programmed.

- d. Repeat steps b, c, and d for Counters 1 and 2.

**3-22. OPERATION**

The following paragraphs describe operating procedures for a counter read, clock frequency divide/ratio selection, and interrupt timer counter selection.

**3-23. COUNTER READ.** There are two methods that can be used to read the contents of a particular counter. The first method involves a simple read of the desired counter. The only requirement with this method is that, in order to ensure stable count reading, the desired counter must be *inhibited* by controlling its gate input. Only Counter 0 and Counter 1 can be read using this method because the gate input to Counter 2 is not controllable.

The second method allows the counter to be read “on-the-fly.” The recommended procedure is to use a mode control word to latch the contents of the count register; this ensures that the count reading is accurate and stable. The latched value of the count can then be read.

**NOTE**

If a counter is read during the down count, it is mandatory to complete the read procedure; that is, if two bytes were programmed to the counter, then two bytes *must* be read before any other operations are performed with that counter.

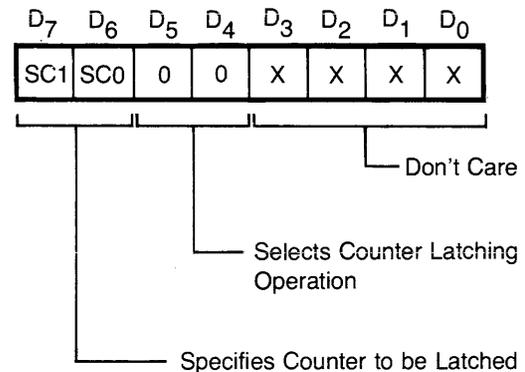
To read the count of a particular counter, proceed as follows (a typical counter read subroutine is given in table 3-10):

- a. Write counter register latch control word (figure 3-11) to 000D6. Control word specifies desired counter and selects counter latching operation.
- b. Perform a read operation of desired counter; refer to table 3-2 for counter addresses.

**NOTE**

Be sure to read one or two bytes, whichever was specified in the initialization mode control word. For two bytes, read in the order specified.

**3-24. CLOCK FREQUENCY/DIVIDE RATIO SELECTION.** Table 2-4 lists the default and optional timer input frequencies to Counters 0 through 3. The timer input frequencies are divided by the counters to generate TMR0 INTR OUT (Counter 0), TMR1 INTR OUT (Counter 1), and the 8251A Baud Rate Clock (Counter 2).



**Figure 3-11. PIT Counter Register Latch Control Word Format**

450-19A

**Table 3-10. Typical PIT Counter Read Subroutine**

```

;READ1 READS COUNTER 1 ON-THE-FLY INTO D&E. MSB IN D, LSB IN E.
;DESTROYS-A,D,E.

PUBLIC READ1
READ1:  MOV     AL,40H           ;MODE WORD FOR LATCHING COUNTER 1 VALUE
        OUT     0D6H
        IN      0D2H           ;LSB OF COUNTER
        MOV     E,A
        IN      0D2H           ;MSB OF COUNTER
        MOV     D,A
        RET
END
    
```

Each counter must be programmed with a down-count number, or count value N. When count value N is loaded into a counter, it becomes the clock divisor. To derive N for either synchronous or asynchronous RS232C operation, use the procedures described in following paragraphs.

3-25. *Synchronous Mode.* In the synchronous mode, the TXC and/or RXC rates equal the Baud rate. Therefore, the count value is determined by:

$$N = C/B$$

where N is the count value,  
B is the desired Baud rate, and  
C is 1.23 MHz, the input clock frequency.

Thus, for a 4800 Baud rate, the required count value (N) is:

$$N = \frac{1.23 \times 10^6}{4800} = \underline{256}.$$

If the binary equivalent of count value N = 256 is loaded into Counter 2, then the output frequency is 4800 Hz, which is the desired clock rate for synchronous mode operation.

3-26. *Asynchronous Mode.* In the asynchronous mode, the TXC and/or RXC rates equal the Baud rate times one of the following multipliers: X1, X16, or X64. Therefore, the count value is determined by:

$$N = C/BM$$

where N is the count value,  
B is the desired Baud rate,  
M is the Baud rate multiplier (1, 16, or 64), and  
C is 1.23 MHz, the input clock frequency.

Thus, for a 4800 Baud rate, the required count value (N) is:

$$N = \frac{1.23 \times 10^6}{4800 \times 16} = \underline{16}.$$

If the binary equivalent of count value N = 16 is loaded into Counter 2, then the output frequency is 4800 × 16 Hz, which is the desired clock rate for asynchronous mode operation. Count values (N) versus rate multiplier (M) for each Baud rate are listed in table 3-11.

**NOTE**

During initialization, be sure to load the count value (N) into the appropriate counter and the Baud rate multiplier (M) into the 8251A USART.

**Table 3-11. PIT Count Value Vs Rate Multiplier for Each Baud Rate**

Baud Rate: (B)	*Count Value (N) For		
	M = 1	M = 16	M = 64
75	16384	1024	256
110	11171	698	175
150	8192	512	128
300	4096	256	64
600	2048	128	32
1200	1024	64	16
2400	512	32	8
4800	256	16	4
9600	128	8	2
19200	64	4	
38400	32	2	
76800	16		

\*Count Values (N) assume clock is 1.23 MHz. Double Count Values (N) for 2.46 MHz clock. Count Values (N) and Rate Multipliers (M) are in decimal.

3-27. **RATE GENERATOR/INTERVAL TIMER.** Table 3-12 shows the maximum and minimum rate generator frequencies and timer intervals for Counters 0 and 1 when these counters, respectively, have 1.23-MHz and 153.6-kHz clock inputs. The table also provides the maximum and minimum generator frequencies and time intervals that may be obtained by connecting Counters 0 and 1 in series.

3-28. **INTERRUPT TIMER.** To program an interval timer for an interruption terminal count, program the appropriate timer for the correct operating mode (Mode 0) in the control word. Then load the count value (N), which is derived by

$$N = TC$$

where

N is the count value for Counter 2,  
T is the desired interrupt time interval in seconds, and  
C is the internal clock frequency (Hz).

Table 3-13 shows the count value (N) required for several time intervals (T) that can be generated for Counters 0 and 1.

**3-29. 8255A PPI PROGRAMMING**

The three parallel I/O ports interfaced to connector J1 are controlled by an Intel 8255A Programmable Peripheral Interface. Port A includes bidirectional data buffers and Ports B and C include IC sockets for installation of either input terminators or output drivers depending on the user's application.

**Table 3-12. PIT Rate Generator Frequencies and Timer Intervals**

	Single Timer <sup>1</sup> (Counter 0)		Single Timer <sup>2</sup> (Counter 1)		Dual Timer <sup>3</sup> (0 and 1 in Series)	
	Minimum	Maximum	Minimum	Maximum	Minimum	Maximum
Rate Generator (frequency)	18.75 Hz	614.4 kHz	2.344 Hz	76.8 kHz	0.00029 Hz	307.2 kHz
Real-Time Interrupt (interval)	1.63 $\mu$ sec	53.3 msec	13 $\mu$ sec	426.67 msec	3.26 $\mu$ sec	58.25 minutes

NOTES:  
 1. Assuming a 1.23-MHz clock input.  
 2. Assuming a 153.6-kHz clock input.  
 3. Assuming Counter 0 has 1.23-MHz clock input.

**Table 3-13. PIT Time Intervals Vs Timer Counts**

T	N*
10 $\mu$ sec	12
100 $\mu$ sec	123
1 msec	1229
10 msec	12288
50 msec	61440

\*Count Values (N) assume clock is 1.23 MHz. Count Values (N) are in decimal.

Default jumpers set the Port A bidirectional data buffers to the input mode. Optional jumpers allow the bidirectional data buffers to be set to the output mode or allow any one of the eight Port C bits to selective set the Port A bidirectional data buffers to the input or output mode.

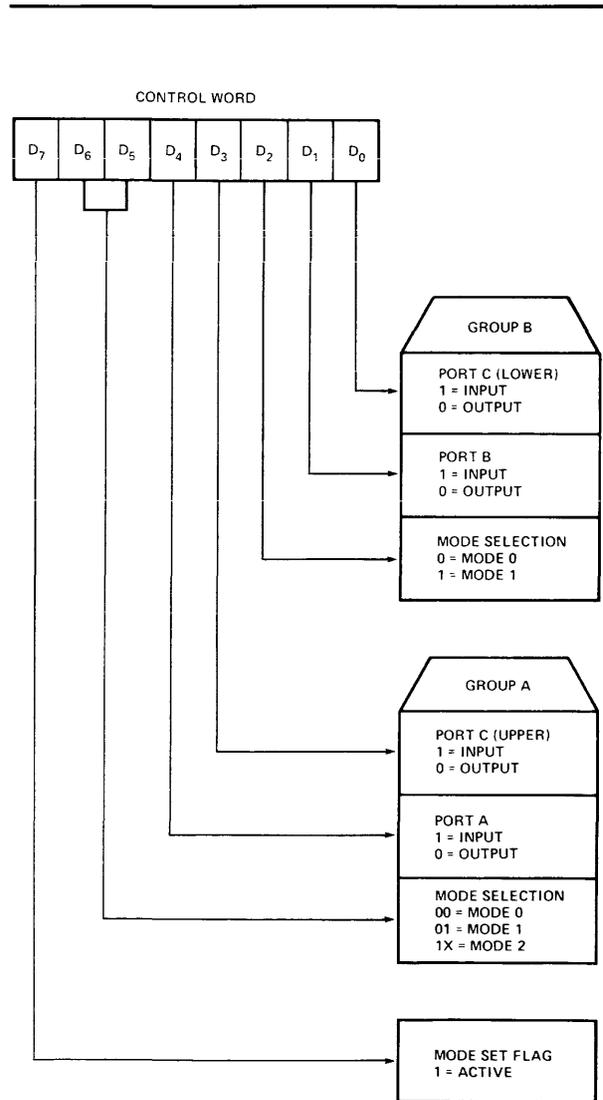
Table 2-11 lists the various operating modes for the three PPI parallel I/O ports. Note that Port A (C8) can be operated in Modes 0, 1, or 2; Port B (CA) and Port C (CC) can be operated in Mode 0 or 1.

**3-30. CONTROL WORD FORMAT**

The control word format shown in figure 3-12 is used to initialize the PPI to define the operating mode of the three ports. Note that the ports are separated into two groups. Group A (control word bits 3 through 6) defines the operating mode for Port A (C8) and the upper four bits of Port C (CC). Group B (control word bits 0 through 2) defines the operating mode for Port B (CA) and the lower four bits of Port C (CC). Bit 7 of the control word controls the mode set flag.

**3-31. ADDRESSING**

The PPI uses four consecutive even addresses (000C8 through 000CE) for data transfer, obtaining the status of Port C (CC), and for port control. (Refer to table 3-2.)



**Figure 3-12. PPI Control Word Format**

**3-32. INITIALIZATION**

To initialize the PPI, write a control word to 000CE. Refer to figure 3-12 and table 3-14 and assume that the control word is 92 (hexadecimal). This initializes the PPI as follows:

- a. Mode Set Flag active
- b. Port A (C8) set to Mode 0 Input
- c. Port C (CC) upper set to Mode 0 Output
- d. Port B (CA) set to Mode 0 Input
- e. Port C (CC) lower set to Mode 0 Output

**3-33. OPERATION**

After the PPI has been initialized, the operation is simply performing a read or a write to the appropriate port.

**3-34. READ OPERATION.** A typical read subroutine for Port A is given in table 3-15.

**3-35. WRITE OPERATION.** A typical write subroutine for Port C is given in table 3-16. As shown in figure 3-13, any of the Port C bits can be selectively set or cleared by writing a control word to 000CE.

**Table 3-14. Typical PPI Initialization Subroutine**

```

;INTPAR INITIALIZES PARALLEL PORTS.
;DESTROYS-A.

                PUBLIC   INTPAR

INTPAR:        MOV     A,92H                ;MODE WORD TO PPI PORT A&B IN,C OUT
                OUT     0CEH
                RET

                END
    
```

**Table 3-15. Typical PPI Port Read Subroutine**

```

;AREAD READS A BYTE FROM PORT A INTO REG A.
;DESTROYS-A.

                AREAD

AREAD:         IN     0C8H                ;GET BYTE
                RET

                END
    
```

**Table 3-16. Typical PPI Port Write Subroutine**

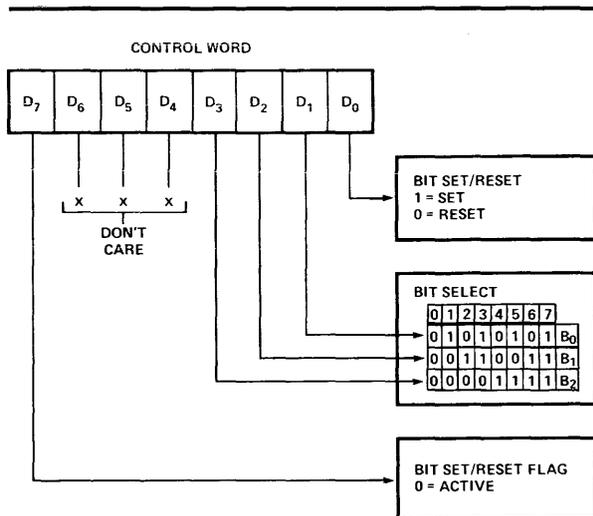
```

;COUT OUTPUTS A BYTE FROM REG A TO PORT C.
;USES-A; DESTROYS-NOTHING.

                PUBLIC   COUT

COUT:         OUT     0CCH                ;OUTPUT BYTE
                RET

                END
    
```



**Figure 3-13. PPI Port C Bit Set/Reset Control Word Format**

### 3-36. 8259A PIC PROGRAMMING

The on-board master 8259A PIC handles up to eight vectored priority interrupts and has the capability of expanding the number priority interrupts by cascading one or more of its interrupt input lines with slave 8259A PIC's. (Refer to paragraph 2-13.)

The basic functions of the PIC are to (1) resolve the priority of interrupt requests, (2) issue a single interrupt request to the CPU based on that priority, and (3) send the CPU a vectored restart address for servicing the interrupting device.

### 3-37. INTERRUPT PRIORITY MODES

The PIC can be programmed to operate in one of the following modes:

- a. Nested Mode
- b. Fully Nested Mode
- c. Automatic Rotating Mode
- d. Specific Rotating Mode
- e. Special Mask Mode
- f. Poll Mode

**3-38. NESTED MODE.** In this mode, the PIC input signals are assigned a priority from 0 through 7. The PIC operates in this mode unless specifically programmed otherwise. Interrupt IR<sub>0</sub> has the highest priority and IR<sub>7</sub> has the lowest priority. When an interrupt is acknowledged, the highest priority request is available to the

CPU. Lower priority interrupts are inhibited; higher priority interrupts will be able to generate an interrupt that will be acknowledged if the CPU has enabled its own interrupt input through software. The End-Of-Interrupt (EOI) command from the CPU is required to reset the PIC for the next interrupt.

**3-39. FULLY NESTED MODE.** This mode is used only when one or more PIC's are slaved to the master PIC, in which case the priority is conserved within the slave PIC's.

The operation in the fully nested mode is the same as the nested mode except as follows:

- a. When an interrupt from a slave PIC is being serviced, that particular PIC is not locked out from the master PIC priority logic. That is, further interrupts of higher priority within this slave PIC will be recognized and the master PIC will initiate an interrupt to the CPU.
- b. When exiting the interrupt service routine, the software must check to determine if another interrupt is pending from the same slave PIC. This is done by sending an End-of-Interrupt (EOI) command to the slave PIC and then reading its In-Service (IS) register. If the IS register is clear (empty), an EOI command is sent to the master PIC. If the IS register is not clear (interrupt pending), no EOI command should be sent to the master PIC.

**3-40. AUTOMATIC ROTATING MODE.** In this mode the interrupt priority rotates. Once an interrupt on a given input is serviced, that interrupt assumes the lowest priority. Thus, if there are a number of simultaneous interrupts, the priority will rotate among the interrupts in numerical order. For example, if interrupts IR<sub>4</sub> and IR<sub>6</sub> request service simultaneously, IR<sub>4</sub> will receive the highest priority. After service, the priority level rotates so that IR<sub>4</sub> has the lowest priority and IR<sub>5</sub> assumes the highest priority. In the worst case, seven other interrupts are serviced before IR<sub>4</sub> again has the highest priority. Of course, if IR<sub>4</sub> is the only request, it is serviced promptly. The priority shifts when the PIC receives an End-of-Interrupt (EOI) command.

**3-41. SPECIFIC ROTATING MODE.** In this mode, the software can change interrupt priority by specifying the bottom priority, which automatically sets the highest priority. For example, if IR<sub>5</sub> is assigned the bottom priority, IR<sub>6</sub> assumes the highest priority. In specific rotating mode, the priority can be rotated by writing a Specific Rotate at EOI (SEOI) command to the PIC. This command contains the BCD code of the interrupt being serviced; that interrupt is reset as the bottom priority. In addition, the bottom priority interrupt can be fixed at any time by writing a command word to the appropriate PIC.

**3-42. SPECIAL MASK MODE.** One or more of the eight interrupt request inputs can be individually masked during the PIC initialization or at any subsequent time. If an interrupt is masked while it is being serviced, lower priority interrupts are inhibited. There are two ways to enable the lower priority interrupts:

- a. Write an End-of-Interrupt (EOI) command.
- b. Set the Special Mask Mode.

The Special Mask Mode is useful when one or more interrupts are masked. If for any reason an input is masked while it is being serviced, the lower priority interrupts are disabled. However, it is possible to enable the lower priority interrupt with the Special Mask Mode. In this mode, the lower priority lines are enabled until the Special Mask Mode is reset. Higher priorities are not affected.

**3-43. POLL MODE.** In this mode the CPU internal Interrupt Enable flip-flop is clear (interrupts disabled) and a software subroutine is used to initiate a Poll command. In the Poll Mode, the addressed PIC treats an I/O Read Command as an interrupt acknowledge, sets its In-Service flip-flop if there is a pending interrupt request, and reads the priority level. This mode is useful if there is a common service routine for several devices.

**3-44. STATUS READ**

Interrupt request inputs is handled by the following two internal PIC registers:

- a. Interrupt Request Register (IRR), which stores all interrupt levels that are requesting service.
- b. In-Service Register (ISR), which stores all interrupt levels that are being serviced.

Either register can be read by writing a suitable command word and then performing a read operation.

**3-45. INITIALIZATION COMMAND WORDS**

The on-board master PIC and each slave PIC requires a separate initialization sequence to work in a particular mode. The initialization sequence, depending on the hardware configuration, requires either three or four of the Initialization Command Words (ICW's) shown in figure 3-14.

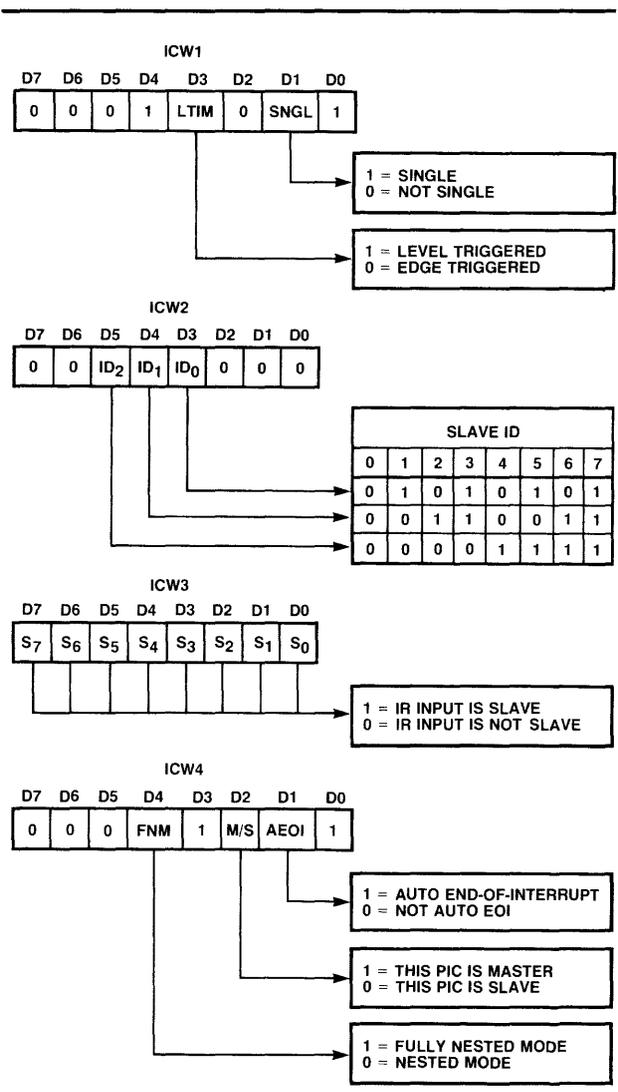
The first Initialization Command Word (ICW1), which is required in all modes of operation, consists of the following:

- a. Bits 0 and 4 are both 1's and identify the word is ICW1 for an 8086 CPU operation.
- b. Bit 1 denotes whether or not the PIC is employed in a multiple PIC configuration. In other words, code bit 1 = 1 if no slave PIC(s) is interfaced to the master PIC via the Multibus.

- c. Bits 2, 5, 6, and 7 are don't care and are normally coded as 0's.
- d. Bit 3 establishes whether the interrupts are requested by a positive-true level input or requested by a low-to-high transition input. This applies to all input requests handled by the PIC. In other words, if bit 3 = 1, a low-to-high transition is required to request an interrupt on any of the eight levels handled by the PIC.

The second Initialization Command Word (ICW2), which is also required in all modes of operation, consists of the following:

- a. For programming the master PIC, write 00H in ICW2. Although in this case ICW2 conveys no information, it is required to prepare the master PIC for either ICW3 or ICW4 (or both) to follow.



645-6 **Figure 3-14. PIC Initialization Command Word Formats**

- b. For programming a slave PIC, code bits 3-5 with a slave identification (ID) number. *Do not use 000 unless there are eight PIC's slaved to the on-board master PIC.* (These ID bits are retained and returned by the slave PIC in response to a CPU interrupt acknowledge.)

The third Initialization Control Word (ICW3) is required only if bit 1 = 0 in ICW1, specifying that multiple PIC's are used; i.e., one or more PIC's are slaved to the on-board master PIC. The S<sub>0</sub>-S<sub>7</sub> bits correspond to the IR0-IR7 bits of the master PIC. For example, if a slave PIC is connected to the master PIC IR3 input, code bit 3 = 1.

The fourth Initialization Control Word (ICW4), which is required for all modes of operation, consists of the following:

- Bits 0 and 3 are both 1's to identify that the word is ICW4 for an 8086 CPU and that the hardware is configured for buffered operation.
- Bit 1 programs the End-of-Interrupt (EOI) function. Code bit 1 = 1 if an EOI is to be automatically executed (hardware). Code bit 1 = 0 if an EOI command is to be generated by software before returning from the service routine.
- Bit 2 specifies if ICW4 is addressed to a master PIC or a slave PIC. For example, code bit 2 = 1 in ICW4 for the master PIC.
- Bit 4 programs the nested or fully nested mode. (Refer to paragraphs 3-38 and 3-39.)

In summary, three or four ICW's are required to initialize the master and each slave PIC. Specifically:

- Master PIC — No Slaves
  - ICW1
  - ICW2
  - ICW4
- Master PIC — With Slave(s)
  - ICW1
  - ICW2
  - ICW3
  - ICW4
- Each Slave PIC
  - ICW1
  - ICW2
  - ICW4

### 3-46. OPERATION COMMAND WORDS

After being initialized, the master and slave PIC's can be programmed at any time for various operating modes. The Operation Command Word (OCW) formats are shown in figure 3-15 and discussed in paragraph 3-49.

### 3-47. ADDRESSING

The master PIC uses addresses 000C0 or 000C2 to write initialization and operation command words and addresses 000C4 or 000C6 to read status, poll, and mask bytes. Addresses for the specific functions are provided in table 3-2.

Slave PIC's, if employed, are accessed via the Multibus and their addresses are determined by the hardware designer.

### 3-48. INITIALIZATION

To initialize the PIC's (master and slaves), proceed as follows (table 3-17 provides a typical PIC initialization subroutine for a PIC operated in the non-bus vectored mode; tables 3-18 and 3-19 are typical master PIC and slave PIC initialization subroutines for the bus vectored mode):

- Disable system interrupts by executing a CLI (Clear Interrupt Flag) instruction.
- Initialize master PIC by writing ICW's in the following sequence:
  - Write ICW1 to 000C0 and ICW2 to 000C2.
  - If slave PIC's are used, write ICW3 and ICW4 to 000C2. If no slave PIC's are used, omit ICW3 and write ICW4 only to 000C2.
- Initialize *each* slave PIC by writing ICW's in the following sequence: ICW1, ICW2, and ICW4.
- Enable system interrupts by executing an STI (Set Interrupt Flag) instruction.

#### NOTE

Each PIC independently operates in the nested mode (paragraph 3-38) after initialization and before an Operation Control Word (OCW) programs it otherwise.

### 3-49. OPERATION

After initialization, the master PIC and slave PIC's can independently be programmed at any time by an Operation Command Word (OCW) for the following operations:

- Auto-rotating priority.
- Specific rotating priority.
- Status read of Interrupt Request Register (IRR).
- Status read of In-Service Register (ISR).
- Interrupt mask bits are set, reset, or read.
- Special mask mode set or reset.

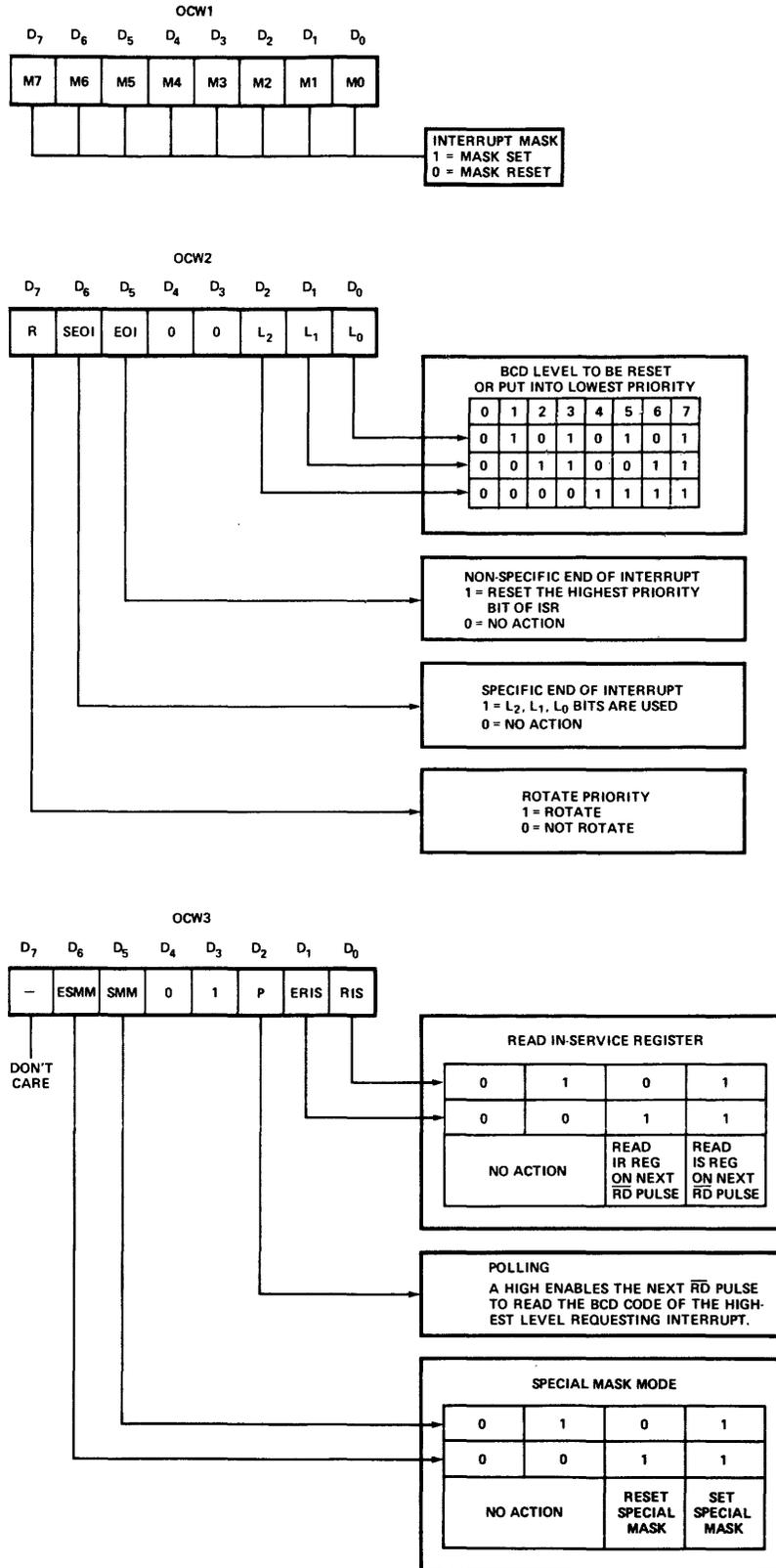


Figure 3-15. PIC Operation Control Word Formats

**Table 3-17. Typical PIC Initialization Subroutine (NBV Mode)**

```

;INT59 INITIALIZES THE PIC. A 64-BYTE ADDRESS BLOCK BEGINNING WITH
;00000H IS SET UP FOR INTERRUPT SERVICE ROUTINES.
;PIC MASK IS SET, DISABLING ALL PIC INTERRUPTS.
;PIC IS IN FULLY NESTED MODE, NON-AUTO EOI.
;USES-SETI, SMASK; DESTROYS-A.

                PUBLIC   INT59
                EXTRN    SETI, SMASK

INT59:          CALL     SETI
                MOV      AL,13H
                OUT      0C0H           ;ICW1 TO PIC
                MOV      AL,00H
                OUT      0C6H           ;ICW2 TO PIC
                MOV      AL,1DH
                OUT      0C2H           ;ICW4 TO PIC
                MOV      AL,0FFH
                CALL     SMASK
                RET

                END
    
```

**Table 3-18. Typical Master PIC Initialization Subroutine (BV Mode)**

```

;INTMA INITIALIZES MASTER PIC WITH A SINGE SLAVE ATTACHED
;IN THE 0 LEVEL INTERRUPT.
;PIC MASK IS SET WITH ALL PIC INTERRUPTS DISABLED.
;PIC IS FULLY NESTED, NON-AUTO EOI.
;USES-SETI, SMASK

                PUBLIC   INTMA
                EXTRN    SETI, SMASK

INTMA:          CALL     SETI
                MOV      AL,11H           ;ICW1
                OUT      0C0H
                MOV      AL,00H           ;ICW2
                OUT      0C2H
                MOV      AL,01H           ;ICW3
                OUT      0C2H
                MOV      AL,1DH           ;ICW4
                OUT      0C2H
                MOV      AL,0FFH
                CALL     SMASK
                RET

                END
    
```

Table 3-20 lists details of the above operations. Note that an End-Of-Interrupt (EOI) or a Special End-Of-Interrupt (SEOI) command is required at the end of each interrupt service routine to reset the ISR. The EOI command is used in the fully nested and auto-rotating priority modes and the SEOI command, which specifies the bit to be reset, is used in the specific rotating priority mode. Tables 3-21 through 3-25 provide typical subroutines for the following:

- a. Read IRR (table 3-21).
- b. Read ISR (table 3-22).
- c. Set mask register (table 3-23).
- d. Read mask register (table 3-24).
- e. Issue EOI command table (3-25).

**Table 3-19. Typical Slave PIC Initialization Subroutine (BV Mode)**

```

;INTSL INITIALIZES A SLAVE PIC LOCATED AT ADDRESS BLOCK
;BEGINNING WITH 0200H.
;PIC IS FULLY NESTED, NON-AUTO EOI.
;USES-SETI, DESTROYS-A.

                PUBLIC   INTSL
                EXTRN    SETI

INTSL:          CALL     SETI
                MOV      AL,11H           ;ICW1
                OUT      0C0H
                MOV      AL,08H         ;ICW2
                OUT      0C2H
                MOV      AL,19H         ;ICW4
                OUT      0C2H
                RET

                END
    
```

**Table 3-20. PIC Operation Procedures**

Operation	Procedure																																																
Auto-Rotating Priority Mode	<p>To set: In OCW2, write a Rotate Priority at EOI command (A0H) to 000C0.</p> <p>Terminate interrupt and rotate priority: In OCW2, write EOI command (20H) to 000C0.</p>																																																
Specific Rotating Priority Mode	<p>To set: In OCW2, write a Rotate Priority at SEOI command in the following format to 000C0:</p> <table border="1" data-bbox="836 1213 1339 1312"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> <tr> <td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>L2</td><td>L1</td><td>L0</td> </tr> </table> <p style="text-align: center;">BCD of IR line to be reset and/or put into lowest priority.</p> <p>To terminate interrupt and rotate priority: In OCW2, write an SEOI command in the following format to 000C0.</p> <table border="1" data-bbox="836 1453 1339 1551"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> <tr> <td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>L2</td><td>L1</td><td>L0</td> </tr> </table> <p style="text-align: center;">BCD of ISR flip-flop to be reset.</p> <p>To rotate priority without EOI: In OCW2, write a command word in the following format to 000C0:</p> <table border="1" data-bbox="836 1713 1339 1812"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> <tr> <td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>L2</td><td>L1</td><td>L0</td> </tr> </table> <p style="text-align: center;">BCD of bottom priority IR line.</p>	D7	D6	D5	D4	D3	D2	D1	D0	1	1	1	0	0	L2	L1	L0	D7	D6	D5	D4	D3	D2	D1	D0	0	1	1	0	0	L2	L1	L0	D7	D6	D5	D4	D3	D2	D1	D0	1	1	0	0	0	L2	L1	L0
D7	D6	D5	D4	D3	D2	D1	D0																																										
1	1	1	0	0	L2	L1	L0																																										
D7	D6	D5	D4	D3	D2	D1	D0																																										
0	1	1	0	0	L2	L1	L0																																										
D7	D6	D5	D4	D3	D2	D1	D0																																										
1	1	0	0	0	L2	L1	L0																																										

**Table 3-20. PIC Operation Procedures (Continued)**

Operation	Procedure																								
<p><b>Interrupt Request Register (IRR) Status</b></p>	<p>The IRR stores a "1" in the associated bit for each IR input line that is requesting an interrupt. To read the IRR (refer to footnote):</p> <p>(1) Write 0AH to 000C0.                      (2) Read 000C0. Status is as follows:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table> <p style="text-align: center;">IR Line:     7    6    5    4    3    2    1    0</p>	D7	D6	D5	D4	D3	D2	D1	D0																
D7	D6	D5	D4	D3	D2	D1	D0																		
<p><b>In-Service Register (ISR) Status</b></p>	<p>The ISR stores a "1" in the associated bit for priority inputs that are being serviced. The ISR is updated when an EOI command is issued. To read the ISR (refer to footnote):</p> <p>(1) Write 0BH to 000C0.                      (2) Read 000C0. Status is as follows:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table> <p style="text-align: center;">IR Line:     7    6    5    4    3    2    1    0</p> <p>Be sure to reset ISR bit at end-of-interrupt when in the following modes:                      Auto-Rotating (both types) and Special Mask. To reset ISR in OCW2, write:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> <tr> <td style="text-align: center;">0</td><td style="text-align: center;">1</td><td style="text-align: center;">1</td><td style="text-align: center;">0</td><td style="text-align: center;">0</td><td style="text-align: center;">L2</td><td style="text-align: center;">L1</td><td style="text-align: center;">L0</td> </tr> </table> <p style="text-align: center;">BCD identifies bit to be reset.</p>	D7	D6	D5	D4	D3	D2	D1	D0	D7	D6	D5	D4	D3	D2	D1	D0	0	1	1	0	0	L2	L1	L0
D7	D6	D5	D4	D3	D2	D1	D0																		
D7	D6	D5	D4	D3	D2	D1	D0																		
0	1	1	0	0	L2	L1	L0																		
<p><b>Interrupt Mask Register</b></p>	<p>To set mask bits in OCW1, write the following mask byte to 000C2:</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>D7</td><td>D6</td><td>D5</td><td>D4</td><td>D3</td><td>D2</td><td>D1</td><td>D0</td> </tr> </table> <p style="text-align: center;">IR Bit Mask:     M7   M6   M5   M4   M3   M2   M1   M0                      1 = Mask Set, 0 = Mask Reset</p> <p>To read mask bits, read 000C2.</p>	D7	D6	D5	D4	D3	D2	D1	D0																
D7	D6	D5	D4	D3	D2	D1	D0																		
<p><b>Special Mask Mode</b></p>	<p>The Special Mask Mode enables desired bits that have been previously masked; lower priority bits are also enabled.</p> <p>To set, write 68H to 000C0.                      To reset, write 48H to 000C0.</p>																								
<p><b>NOTE:</b>                      If previous operation was addressed to same register, it is not necessary to rewrite the OCW.</p>																									

**Table 3-21. Typical PIC Interrupt Request Register Read Subroutine**

```

;RR0 READS PIC INTERRUPT REQUEST REG.
;USES-SETI; DESTROYS-A.

                PUBLIC   RR0
                EXTRN   SETI

RR0:           CALL     SETI
                MOV     AL,0AH           ;OCW3 RR INSTRUCTION TO PIC
                OUT    0C0H
                IN     0C0H
                RET

                END
    
```

**Table 3-22. Typical PIC In-Service Register Read Subroutine**

```

;RIS0 READS PIC IN-SERVICE REGISTER.
;USES-SETI; DESTROYS-A.

                PUBLIC   RIS0
                EXTRN   SETI

RIS0:          CALL     SETI
                MOV     AL,08H           ;OCW3 RIS INSTRUCTION TO PIC
                OUT    0C4H
                IN     0C4H
                RET

                END
    
```

**Table 3-23. Typical PIC Set Mask Register Subroutine**

```

;SMASK STORES A REG INTO PIC MASK REG.
;A ONE MASKS OUT AN INTERRUPT, A ZERO ENABLES IT.
;USES-A,SETI; DESTROYS-NOTHING.

                PUBLIC   SMASK
                EXTRN   SETI

SMASK:         CALL     SETI
                OUT    0C2H
                RET

                END
    
```

**Table 3-24. Typical PIC Mask Register Read Subroutine**

```

;RMASK READS PIC MASK REG INTO A REG.
;USES-SETI; DESTROYS-A.

                PUBLIC   RMASK
                EXTRN   SETI

RMASK:         CALL     SETI
                IN     0C2H
                RET

                END
    
```

**Table 3-25. Typical PIC End-of-Interrupt Command Subroutine**

```

;EOI ISSUES END-OF-INTERRUPT TO PIC.
;USES-SETI; DESTROYS-A.

                PUBLIC   EOI
                EXTRN   SETI

EOI:            CALL    SETI
                MOV     A,20H
                OUT    0C0H
                RET

                END
;NON-SPECIFIC EOI
    
```

**3-50. HARDWARE INTERRUPTS**

The 8086 CPU includes two hardware interrupts inputs, NMI and INTR, classified as non-maskable and maskable, respectively.

**3-51. NON-MASKABLE INTERRUPT (NMI)**

The NMI input has the higher priority of the two interrupt inputs. A low-to-high transition on the NMI input will be serviced at the end of the current instruction or between whole moves of a block-type instruction. Worst-case response to NMI is during a multiply, divide, or variable shift instruction.

When the NMI input goes active, the CPU performs the following:

- a. Pushes the Flag registers onto the stack (same as a PUSHF instruction).
- b. If not already clear, clears the Interrupt Flag (same as a CLI instruction); this disables maskable interrupt.
- c. Transfers control with an indirect call through 00008.

The NMI input is intended only for catastrophic error handling such as a system power failure. Upon completion of the service routine, the CPU automatically restores the flags and returns to the main program.

**3-52. MASKABLE INTERRUPT (INTR)**

The INTR input has the lower priority of the two interrupt inputs. A high level on the INTR input will be serviced at the end of the current instruction or at the end of a whole move for a block-type instruction.

When INTR goes active, the CPU performs the following (assuming the Interrupt Flat is set):

- a. Issues two acknowledge signals; upon receipt of the second acknowledge signal, the interrupting device (master or slave PIC) will respond with a one-byte interrupt identifier.
- b. Pushes the Flag registers onto the stack (same as a PUSHF instruction).
- c. Clears the Interrupt Flag, thereby disabling further maskable interrupts.
- d. Multiplies by four (4) the binary value (X) contained in the one-byte identifier from the interrupting device.
- e. Transfers control with an indirect call through 4X.

Upon completion of the service routine, the CPU automatically restores its flags and returns to the main program.

**3-53. MASTER PIC BYTE IDENTIFIER.** The master (on-board) PIC responds to the second acknowledge signal from the CPU only if the interrupt request is from a non-slaved device; i.e., a device that is connected directly to one of the master PIC IR inputs. The master PIC has eight IR inputs numbered IR0 through IR7, which are identified by a 3-bit binary number. Thus, if an interrupt request occurs on IR5, the master PIC responds to the second acknowledge signal from the CPU by outputting the byte 00000101<sub>2</sub> (05<sub>H</sub>). The CPU multiplies this value by four and transfers control with an indirect call through 00010100<sub>2</sub> (14<sub>H</sub>).

**3-54. SLAVE PIC BYTE IDENTIFIER.** Each slave PIC is initialized with a 3-bit identifier (ID) in ICW2. These three bits will form a part of the byte identifier transferred to the CPU in response to the second acknowledge signal.

The slave PIC requests an interrupt by driving the associated master PIC IR line. The master PIC, in turn, drives the CPU INTR input high and the CPU outputs the first of two acknowledge signals. In response to the first acknowledge signal, the master PIC outputs a 3-bit binary code to slaved PIC's; this 3-bit code allows the appropriate slave PIC to respond to the second acknowledge signal from the CPU.

Assume that the slave PIC has the ID code  $111_2$  assigned in ICW2, and that the device requesting service is driving the IR2 line (010). Thus, in response to the second acknowledge signal, the slave PIC outputs  $00111010_2$  ( $3A_H$ ). The CPU multiplies this value by four and transfers control with an indirect call through  $11101000_2$  ( $E8_H$ ).



## 4-1. INTRODUCTION

This chapter provides a functional description and a circuit analysis of the iSBC 86/12 Single Board Computer. Figures 4-1 and 4-2, located at the end of this chapter, are simplified foldout logic diagrams that illustrate the functional interface between the 8086 microprocessor (CPU) and the on-board facilities and between the CPU and the system facilities via the Multibus. Also shown in figure 4-2 is the Dual Port Control Logic that allows the iSBC 86/12 to function in a master/slave relationship with the Multibus to allow another bus master to access the on-board dual port RAM.

## 4-2. FUNCTIONAL DESCRIPTION

A brief description of the functional blocks of logic comprising the iSBC 86/12 is given in following paragraphs. A operational circuit analysis is given beginning with paragraph 4-13.

## 4-3. CLOCK CIRCUITS

The clock circuit composed of A16, A17, and A18 is stabilized by a 22.1184-MHz crystal. This circuit provides nominal 153.7-kHz, 1.23-MHz, and 2.46-MHz optional clock frequencies to the 8253 Programmable Interval Timer (PIT); 2.46-MHz Baud rate clock to the 8251A Universal Synchronous/Asynchronous Receiver/Transmitter (USART); and a 22.12-MHz clock frequency to the Dual Port Control Logic and RAM Controller.

The clock circuit composed of A80 and A63 is stabilized by an 18.432-MHz crystal. This circuit divides the crystal frequency by two to provide the nominal 9.22-MHz Bus Clock (BCLK/) and Constant Clock (CCLK/) signals to the Multibus. (The BCLK/ signal is also used by the Bus Arbiter Assembly.) Removeable jumpers are provided to allow this clock circuit to be disabled if some other source supplies BCLK/ and CCLK/ to the Multibus.

Clock A38 is stabilized by a 15-MHz crystal and provides a nominal 5-MHz clock to CPU A39, Status Decoder A81, the Bus Arbiter Assembly, and Bus Command Decoder A83. Clock A38 also provides a reset signal on power-up and when commanded to do so by an optional signal supplied via auxiliary connector P2. The RESET signal initializes the system as well as certain iSBC 86/12 components to a known internal state.

## 4-4. CENTRAL PROCESSOR UNIT

The 8086 Microprocessor (CPU A39), which is the heart of the single board computer, performs the system processing functions and generates the address and control signals required to access memory and I/O devices. Control signals S0, S1, and S2 are driven by the CPU and decoded by Status Decoder A81 to develop the various signals required to control the board. The CPU AD0-AD15 pins are used to multiplex the 16-bit input/output data and the lower 16-bits of the address. During the first part of a machine cycle, for example, the lower 16-bits (AD0-AD15) and the upper 4-bits (AD16-AD19) are strobed into Address Latch A40/41/57 by the Address Latch Enable (ALE) signal. (The ALE signal is derived by decoding S0, S1, and S2.) The Address Latch outputs form the 20-bit address bus AB0-AB13; i.e., AB0-ABF and AB10-AB13. During the remainder of the machine cycle, the AD0-AD15 pins of the CPU are used to form the 16-bit data bus AD0-ADF.

## 4-5. INTERVAL TIMER

The 8253 PIT provides three independently controlled counters that derive their optional basic timing inputs from the clock circuit composed of A16/17/18.

Counter 2 provides timing for the serial I/O port (8251A USART). This counter, in conjunction with the USART, can provide programmable Baud rates from 110 to 9600. Counter 0 can be used in one of two ways: (1) as a clock generator it can be buffered to provide an external user-defined clock or (2) as an interval timer to generate a CPU interrupt. Counter 1, which is the system interval timer and can also generate an interrupt, has a range of 1.6 microseconds to 853.3 milliseconds. If longer times are needed, Counters 0 and 1 can be cascaded to provide a single timer with a maximum delay of over 50 hours.

## 4-6. SERIAL I/O

The 8251A USART provides RS232C compatibility and is configured as a data terminal. Synchronous or asynchronous mode, character size, parity bits, stop bits, and Baud rates are all programmable. Data, clocks and control lines to and from connector J2 are buffered.

## 4-7. PARALLEL I/O

The 8255A Programmable Peripheral Interface provides 24 programmable I/O lines. Two IC sockets are provided so that, depending on the application, TTL drivers or I/O

terminators may be installed to complete the interface to connector J1. The 24 lines are grouped into three ports of eight lines each; these ports can be programmed to be simple I/O ports, strobed I/O ports with handshaking, or one port can be programmed as a bidirectional port with control lines. The iSBC 86/12 includes various optional functions controlled by the parallel I/O lines such as an RS232C interface line, timer gate control lines, bus override, strobed I/O port interrupts, and one Multibus interrupt.

#### 4-8. INTERRUPT CONTROLLER

The 8259A Programmable Interrupt Controller (PIC) handles up to eight vectored priority interrupts. The 8259A PIC provides the capability to expand the number of priority interrupts by cascading each interrupt line with another 8259A PIC. (Refer to figure 2-2.) This is done by programming the master PIC (the one on the iSBC 86/12) that an interrupt line (e.g., IR3) is connected to a slave PIC (the one interfaced to the master PIC via the Multibus). If an IR3 interrupt is sensed by the master PIC, it will allow the slave PIC to send the restart vector address to the CPU. Each interrupt line into the master PIC can be individually programmed to be a non-bus vectored (NBV) interrupt line (master PIC generates the restart address) or a bus vectored (BV) interrupt (cascaded to a slave PIC which generates the restart address). The iSBC 86/12 can handle eight on-board or single Multibus interrupt lines (an interrupt line which does not have a slave PIC connected to it) or, with the aid of eight slave PIC's, expand the number of interrupts to 64. All 64 interrupts must be processed through the slave PIC's and must therefore be external to the iSBC 86/12.

There are nine jumper-selectable interrupt sources: serial I/O port (2), parallel I/O interface (2), timers (2), external via J1 (1), power fail (1), and Multibus time out (1). The eight Multibus interrupt lines (INT0/-INT7) can be connected to the master PIC to provide 8 to 64 bus interrupt levels. The user can map interrupt sources into interrupt levels by hardware jumpers. The iSBC 86/12 can also generate one Multibus interrupt that is controlled by an 8255A PPI output bit.

#### 4-9. ROM/EPROM CONFIGURATION

IC sockets A28, A29, A46, and A47 are provided for user installation of ROM or EPROM chips; jumpers are provided to accommodate either 2K, 4K, or 8K chips. The ROM/EPROM address space is located at the top of the 1-megabyte memory space because the 8086 CPU branches to FFFF0 after a reset. Starting addresses for the different ROM/EPROM configurations are FF000 (using 2K chips), FE000 (using 4K chips), and FC000 (using 8K chips).

#### 4-10. RAM CONFIGURATION

The iSBC 86/12 includes 32K bytes of read/write memory composed of sixteen 2117 Dynamic RAM chips and an 8202 RAM Controller.

The Dual Port Control Logic interfaces the RAM with the Multibus so that the iSBC 86/12 can perform as a slave RAM device when not acting as a bus master. This dual port is designed to maximize the CPU throughput by defaulting control to the CPU when not in demand. Each time a bus master generates a memory request to the dual port RAM via the Multibus, the RAM must be taken away from the CPU (when the CPU is not using it). When the slave request is completed, the control of the RAM returns to the CPU.

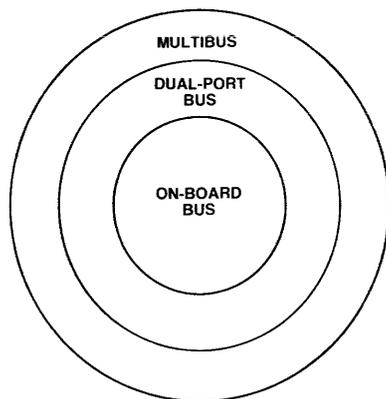
The dual port consists of CPU address and data buffers and decoder; bidirectional address and data bus (Multibus) drivers; slave RAM address decoder/translator; control logic; and the RAM and RAM controller.

The CPU address and data buffers separate the on-board bus (I/O and ROM/EPROM) from the dual port bus. On-board RAM addresses (as seen by the CPU) are (assigned from the bottom up) 00000-07FFF.

The address bus drivers and data bus drivers separate the dual port bus from the Multibus. The slave RAM address decoder is separate from the CPU RAM address decoder to provide independent Multibus address selection that can be located throughout the 1-megabyte address space. The slave RAM address is selected by specifying the base address and memory size. The base address can be on any 8K boundary with the exception that the memory space cannot extend across a 128K boundary. The memory size specifies the amount of Dual Port RAM accessible by the Multibus and is switch selectable in 8K increments. This provides the capability to reserve sections of the dual port RAM for use only by the CPU and frees up the address space. Regardless of what base address is selected, the slave RAM address is mapped into an on-board RAM address (as seen by the CPU). (Refer to figure 2-1.)

#### 4-11. BUS STRUCTURE

The iSBC 86/12 architecture is organized around a three-bus hierarchy: the on-board bus, the dual port bus, and the Multibus. (Refer to figure 4-3.) Each bus can communicate only within itself and an adjacent bus, and each bus can operate independently of each other. The performance of the iSBC 86/12 is directly related to which bus it must go to perform an operation; that is, the closer the bus to the on-board bus, the better the performance.



645-9 **Figure 4-3. Internal Bus Structure**

The iSBC 86/12 operates at a 5-MHz CPU cycle and requires one wait state for all on-board system accesses. (Exception: a RAM write requires two wait states.) However, the pipeline effect of the 8086 CPU effectively “hides” these wait states.

The core of the iSBC 86/12 series bus architecture is the on-board bus, which connects the CPU to all on-board I/O devices, ROM/EPROM, and the dual port RAM bus. Activity on this bus does not require control of the outer buses, thus permitting independent execution of on-board activities. Activities at this level require no bus overhead and operate at maximum board performance.

The next bus in the hierarchy is the dual port bus. This bus controls the dynamic RAM and communicates with the on-board bus and the Multibus. The dual port bus can be in one of three states:

- a. State 1 — On-board bus is controlling it but not using it (not busy).
- b. State 2 — On-board bus is controlling it and using it (busy).
- c. State 3 — Multibus is controlling it and using it (busy).

State 1 is the idle state of the dual port bus and is left in control of the on-board bus to minimize delays when the CPU needs it. When the on-board bus requires the dual port bus to access RAM, the dual port bus control logic will go from State 1 to State 2. (If the dual port bus is busy, it will wait until it is not busy). Activity at this level requires a minimum of bus overhead and the

RAM performance is designed to equal that of on-board activity (if the dual port bus is not busy when the on-board bus requests it). The dual port bus control logic returns to State 1 when the CPU completes its operation. This level of bus activity operates independently of Multibus activity (if the Multibus does not need the dual port bus).

When the Multibus requests the dual port bus, the control logic goes from State 1 to 3 (it will wait if busy) in about 150 nanoseconds and, upon completion, returns to State 1. The Multibus use of the dual port bus is independent of the on-board activity.

When the on-board bus needs the Multibus, it must go through the dual port bus to the Multibus. The on-board bus uses the dual port bus only to communicate with the Multibus and leaves the dual port bus in State 1. Activity at this level requires a minimum 200-nanosecond overhead for Multibus exchange.

#### 4-12. MULTIBUS INTERFACE

The iSBC 86/12 is completely Multibus compatible and supports both 8-bit and 16-bit operations. The Multibus interface includes the Bus Arbiter Assembly, Bus Command Decoder A83, bidirectional address bus and data bus drivers, and interrupt drivers and receivers. The Bus Arbiter allows the iSBC 86/12 to operate as a bus masters in the system in which the 8086 CPU can request the Multibus when a bus resource is needed.

The Bus Arbiter Assembly mounts on the iSBC 86/12 and is electrically interfaced to the board via connector J12.

#### 4-13. CIRCUIT ANALYSIS

The schematic diagram for the iSBC 86/12 is given in figure 5-2. The schematic diagram consists of 11 sheets, each of which includes grid coordinates. Signals that traverse from one sheet to another are assigned grid coordinates at both the signal source and signal destination. For example, the grid coordinates 2ZB1 locate a signal source (or signal destination as the case may be) on sheet 2 Zone B1.

Both active-high and active-low signals are used. A signal mnemonic that ends with a virgule (e.g., DAT7/) denotes that the signal is active low ( $\leq 0.4V$ ). Conversely, a signal mnemonic without a virgule (e.g., ALE) denotes that the signal is active high ( $\geq 2.0V$ ).

Figures 4-1 and 4-2 at the end of this chapter are simplified logic diagrams of the input/output, interrupt, and memory sections. These diagrams will be helpful in understanding both the addressing scheme and the internal bus structure of the board.

#### 4-14. INITIALIZATION

When power is applied in a start-up sequence, the contents of the 8086 CPU program counter, program status word, interrupt enable flip-flop, etc., are subject to random factors and cannot be predicted. For this reason, a power-up sequence is used to set the CPU, Bus Arbiter, and I/O ports to a known internal state.

When power is initially applied to the iSBC 86/1X, capacitor C26 (2ZD6) begins to charge through resistor R9. The charge developed across C26 is sensed by a Schmitt trigger, which is internal to Clock Generator A38. The Schmitt trigger converts the slow transition appearing at pin 12 into a clean, fast-rising synchronized RESET signal at pin 11. The RESET signal is inverted by A48-6 to develop RESET/ and INIT/. The RESET/ signal automatically sets the 8086 CPU program counter to FFFF0 and clears the interrupt enable flip-flop; resets the parallel I/O ports to the input mode; resets the serial I/O port to the "idle" mode; and resets the Bus Arbiter (outputs are tristated). The INIT/ signal is transmitted over the Multibus to set the entire system to a known internal state.

The initialization described above can be performed at any time by inputting a RESET/ signal via auxiliary connector P2.

#### 4-15. CLOCK CIRCUITS

The 5-MHz CLK is developed by Clock Generator A38 (2ZC6) in conjunction with crystal Y2. This clock is the time base for CPU A39, Status Decoder A81, the Bus Arbiter Assembly and Bus Command Decoder A83.

The time base for Bus Clock BCLK/ and Constant Clock CCLK/ is provided by Clock Generator A80 (10ZA5) and crystal Y3. The 18.432-MHz crystal frequency is divided by A63 and driven onto the Multibus through jumpers 105-106 and 103-104. The BCLK/ signal is also used as a clock input to the Bus Arbiter Assembly.

The time base for the remaining functions on the board is provided by clock Generator A17 (7ZA7) and crystal Y1. The nominal 22.12-MHz crystal frequency appearing at the OSC output of A17 is buffered and supplied to the Dual Port Control Logic and to RAM Controller A70. Clock Generator A17 also divides the crystal frequency by nine to develop a 2.46-MHz clock at its  $\Phi$ 2TTL output. The 2.46-MHz clock is applied directly to the clock input of the 8251A USART and applied through A18 to provide a selectable clock for the 8253 PIT. Divider A16 also divides the 2.46-MHz clock by two and by nine, respectively, to produce 1.23-MHz and 153.6-kHz selectable clocks for the 8253 PIT.

#### 4-16. CENTRAL PROCESSOR UNIT

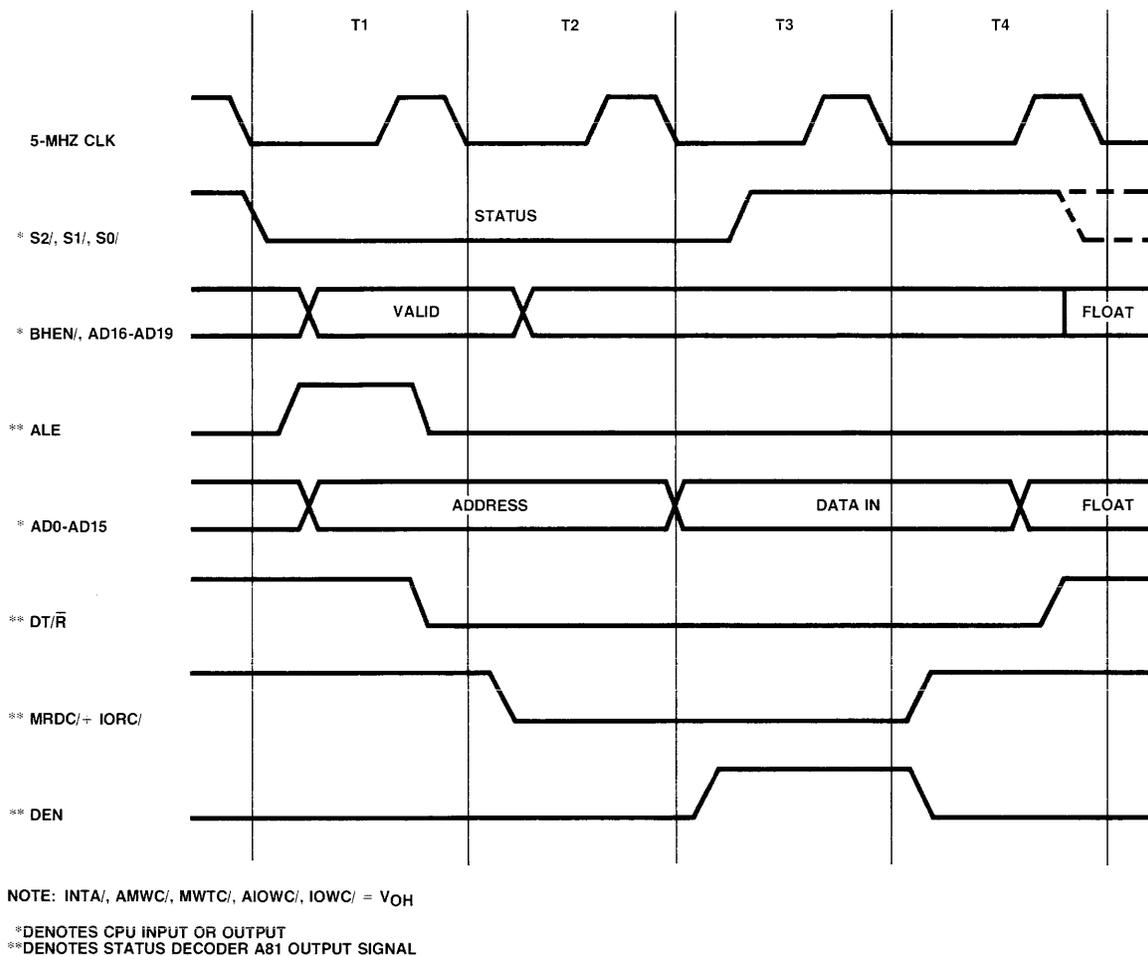
The 8086 CPU uses the 5-MHz clock input to develop the timing requirements for various time-dependent functions described in following paragraphs.

**4-17. BASIC TIMING.** Each CPU bus cycle consists of at least four clock (CLK) cycles referred to as T1, T2, T3 and T4. The address is emitted from the CPU during T1 and data transfer occurs on the bus during T3 and T4; T2 is used primarily for changing the direction of the bus during read operations. In the event that a "not ready" indication is given by the addressed device, "wait" states (TW) are inserted between T3 and T4. Each inserted TW state is of the same duration as a CLK cycle. Periods can occur between CPU-driven bus cycles; these periods are referred to as "idle" states (TI) or inactive CLK cycles. The processor uses TI states for internal housekeeping.

**4-18. BUS TIMING.** The CPU generates status signals S0, S1, and S2 during T1 of every machine cycle. These status signals are used by Status Decoder A81, Bus Arbiter Assembly, and Bus Command Decoder A83 to identify the following types of machine cycles.

S2	S1	S0	CPU Machine Cycle
0	0	0	Interrupt Acknowledge
0	0	1	I/O Read
0	1	0	I/O Write
0	1	1	Halt
1	0	0	Code Access
1	0	1	Memory Read
1	1	0	Memory Write
1	1	1	Passive

A read cycle begins in T1 with the assertion of the Address Latch Enable (ALE) signal and the emission of the address. (Refer to figure 4-4.) The trailing edge of ALE signal latches the address into Address Latch A40/41/57 (2ZB2). (The BHEN/ signal and address bit AD0 address the low byte, high byte, or both bytes.) The Data Transmit/Receive (DT/ $\bar{R}$ ) signal, which is asserted at the end of T1, is used to set up the various data buffers and data bus drivers for a CPU read operation. The Memory Read Command (MRDC/) or I/O Read Command (IORC/) is asserted from the beginning of T2 to the beginning of T4. At the beginning of T3, the AD0-AD15 lines of the local bus are switched to the "data" mode and the Data Enable (DEN) signal is asserted. (The DEN signal enables the data buffers.) The CPU examines the state of its READY input during the last half of T3. If its READY input is high (signifying that the addressed device has placed data on the data lines), the CPU proceeds into T4; if its READY input is low, the CPU enters a wait (TW) state and stays there until READY



645-10

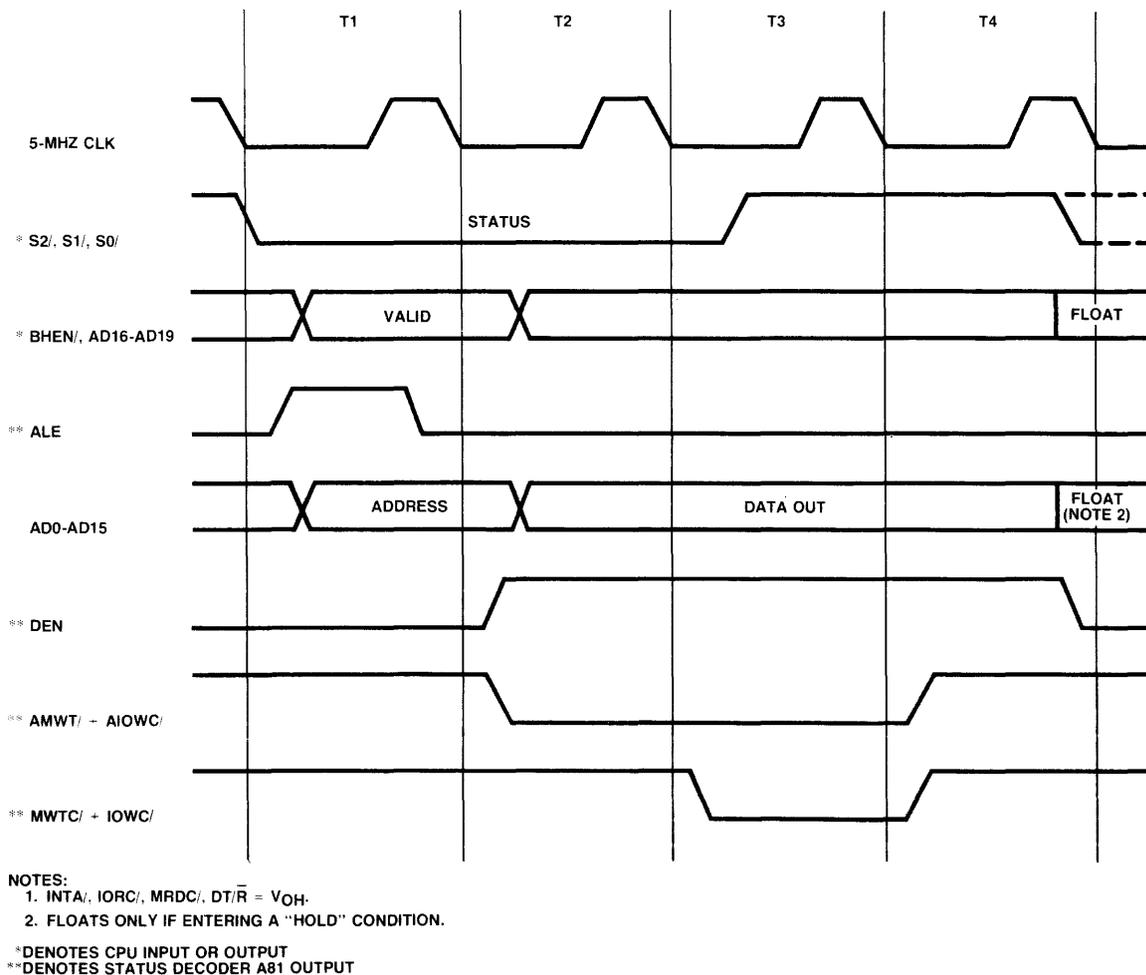
Figure 4-4. CPU Read Timing

goes high. The external effect of using the READY input is to preserve the exact state of the CPU at the end of T3 for an integral number of clock periods before finishing the machine cycle. This 'stretching' of the system timing, in effect, increases the allowable access time for memory or I/O devices. By inserting TW states, the CPU can accommodate slower memory or slower I/O devices. The CPU accepts the data and terminates the command in T4; the DEN signal then goes false and the data buffers are tristated.

A write cycle begins in T1 with the assertion of the ALE signal and the emission of the address. (Refer to figure 4-5.) The trailing edge of ALE latches the address into the address latch as described for a write cycle. The DT/ $\bar{R}$  signal remains high throughout the entire read cycle to set up the data buffers and data bus buffers for a CPU write operation. Status Decoder A81 provides two types of write strobe signals: advanced (AMWT/ and AIOWC/) and normal (MWTC/ and IOWC/). As shown in figure 4-5, the advanced memory and I/O write strobes are

issued one clock cycle earlier than the normal memory and I/O write strobes. (The iSBC 86/12 doesn't use advanced I/O write strobe AIOWC/.) At the beginning of T2, the advance write and DEN signals are asserted and the AD0-AD15 lines of the local bus are switched to the "data" mode. (The DEN signal enables the data buffers.) The CPU then places the data on the AD0-AD15 lines and, at the beginning of T3, the normal write strobe is issued. The CPU examines the state of its READY input during the last half of T3. When READY goes high (signifying that the addressed device has accepted the data), the CPU enters T4 and terminates the write strobe. DEN then goes false and the data buffers are tristated.

The CPU interrupt acknowledge (INTA) cycle timing is shown in figure 4-6. Two back-to-back INTA cycles are required for each interrupt initiated by the 8259A PIC or by a slave 8259A PIC cascaded to the master PIC. The INTA cycle is similar to a read cycle. The basic difference is that an INTA/ signal is asserted instead of an MRDC/ or IORC/ signal and the address bus is floated. In the second



645-11

Figure 4-5. CPU Write Timing

INTA cycle, a byte of information (supplied by the 8259A PIC) is read from "data" lines AD0-AD7. This byte, which identifies the interrupting source, is multiplied by four by the CPU and used as a pointer into an interrupt vector look-up table.

#### 4-19. ADDRESS BUS

The address bus is shown in weighted lines in figures 4-1 and 4-2. The 20-bit address (AD0-AD19) is output by CPU A39 during the first clock cycle (T1) of the memory or I/O instruction. The trailing edge of the Address Latch Enable (ALE) signal, output by Status Decoder A81 during T1, strobes and latches the address into Latch A40/41/57. The latched address is distributed as follows:

- a. AB3-ABF to I/O Address Decoder A54/55/56 (6ZA7).
- b. ABB-AB13 to PROM Address Decode Logic A18/68 (6ZB6).

- c. AB1-ABC to PROM A28/29/46/47 (6ZC3).
- d. AB13 to on-board RAM address recognition gate A53-6 (6ZD6).

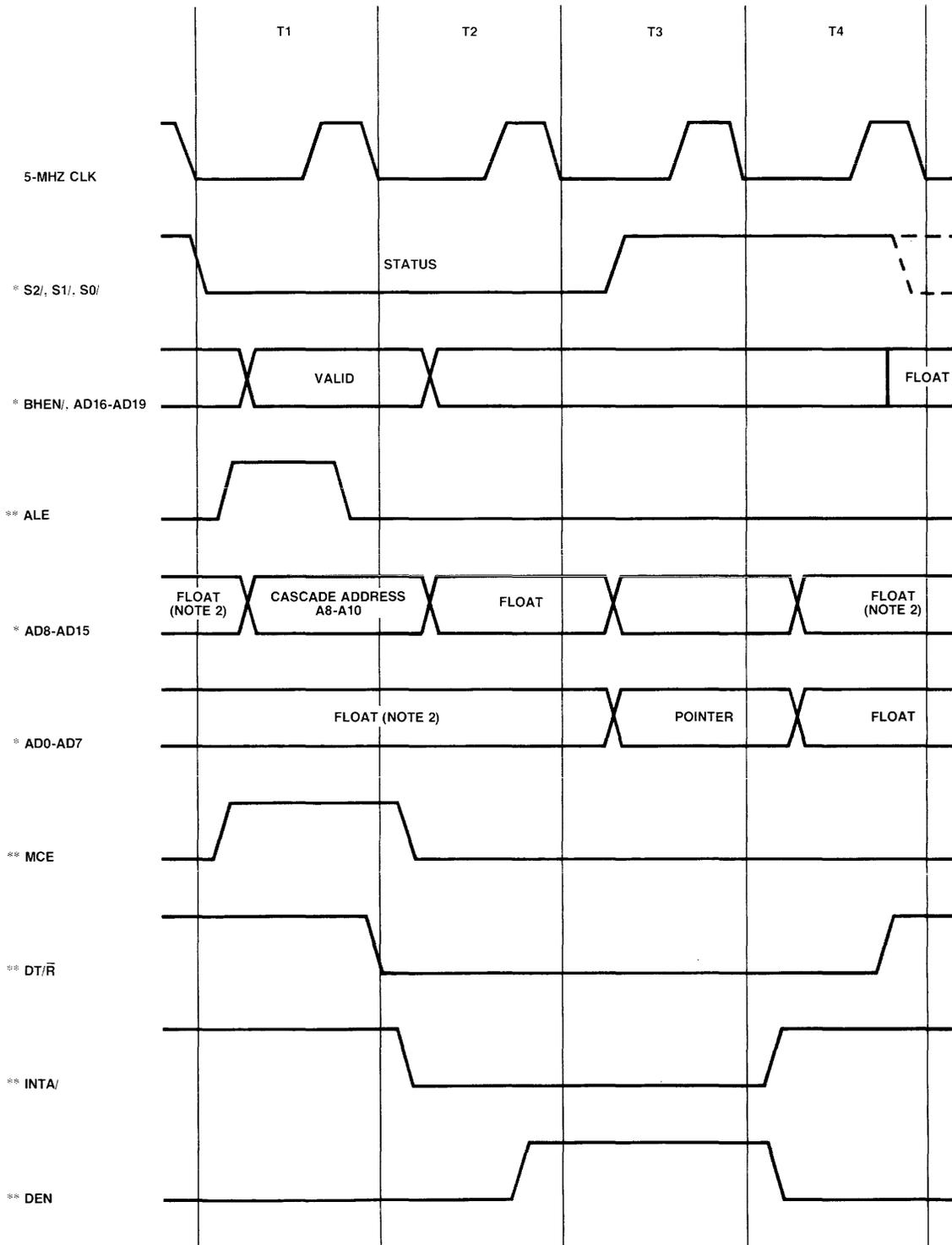
#### 4-20. DATA BUS

At the beginning of clock cycle T2, the CPU AD0-AD15 pins become the source or destination of data bus AD0-ADF. Data can be sourced to or input from the following:

- a. Data Buffer A44/45 (4ZD4).
- b. Data Buffer A60/61 (4ZD5).

#### 4-21. BUS TIME OUT

Bus Time Out one-shot A5 (10ZA6) is triggered by the leading edge of the ALE signal. If the CPU halts, or is hung up in a wait state for approximately 6.2 ( $\pm 15\%$ ) nanoseconds, A5 times out and asserts the TIMEOUT/ signal. If jumper 5-6 is installed, the TIMEOUT/ signal



NOTES:  
 1. MRDC/, IORC/, AMWC/, MWTC/, AIOWC/, IOWC/ = V<sub>OH</sub>; BHEN/ = V<sub>OL</sub>  
 2. THE TWO INTA CYCLES RUN BACK-TO-BACK. THUS, THE LOCAL BUS IS FLOATING WHEN THE SECOND INTA CYCLE IS ENTERED.  
 \*DENOTES CPU INPUT OR OUTPUT  
 \*\*DENOTES STATUS DECODER A81 OUTPUT

Figure 4-6. CPU Interrupt Acknowledge Cycle Timing

drives the CPU READY line high through A7-12 and A38-5 to allow the CPU to exit the wait state. The TIMEOUT/ signal is also routed as a TIMEOUT INTR signal to the interrupt jumper matrix (8ZD1).

#### 4-22. INTERNAL CONTROL SIGNALS

Status Decoder A81 (3ZB3) receives the 5-MHz CLK signal from Clock Generator A38 and status signals S0-S2 from CPU A39. The CLK signal establishes when the command signals are generated as a result of decoding S0-S2. The following signals are output from Status Decoder A81:

Signal	Definition
ALE	<i>Address Latch Enable.</i> Strobes address into Address Latch A40/41/57.
AIOWC/	<i>Advanced I/O Write.</i> An I/O Write Command that is issued earlier than IOWC/ in an attempt to avoid imposing a CPU wait state.
AMWC/	<i>Advanced Memory Write Command.</i> A Memory Write Command that is issued earlier than MWTC/ in an attempt to avoid imposing a CPU wait state.
DEN	<i>Data Enable.</i> Enables Data Buffers A44 and A60/61.
DT/ $\bar{R}$	<i>Data Transmit/Receive.</i> Establishes direction of data transfer through Data Buffers A44/45 and A60/61 and Data Bus Buffers A69/89/90.
IORC/	<i>I/O Read Command</i> to on-board PPI, USART, PIT, and PIC.
IOWC/	<i>I/O Write Command</i> to on-board PPI, USART, PIT, and PIC.
INTA/	<i>Interrupt Acknowledge.</i> Provides on-board control during INTA cycle.
MCE	<i>Master Cascade Enable.</i> Enable cascade address from master 8259A PIC onto local bus so that slave PIC address can be latched.
MRDC/	<i>Memory Read Command.</i>
MWTC/	<i>Memory Write Command.</i>

#### 4-23. DUAL PORT CONTROL LOGIC

The Dual Port Control Logic (figure 5-2 sheet 11) allows the dual port RAM facilities to be shared by the on-board CPU or by another bus master via the Multibus. When not acting as a bus master or when not accessing the dual port RAM, the iSBC 86/12 can act as a "slave" RAM device in a multiple bus master system. When accessing the dual port RAM, the on-board CPU has priority over any attempt to access the dual port RAM via the Multibus. In this situation, the bus access is held off until the CPU has completed its particular read or write operation. When a bus access is in progress, the Dual Port Control Logic enters the "slave" mode and any subsequent CPU request will be held off until the slave mode is terminated. Figures 4-7 and 4-8 are timing diagrams for the Dual Port Control Logic.

**4-24. MULTIBUS ACCESS TIMING.** Figure 4-7 illustrates the Dual Port Control Logic timing for dual port RAM access via the Multibus. (P-periods P0 through P17 are used only for descriptive purposes and have no relationship to the 22.12-MHz clock signal.) When the OFF BD RAM CMD signal goes high, A49-10 goes high and A49-7 goes low on the next rising edge of the clock at the end of P0 (assuming that ON BD RAM RQT/ and RAM XACK/ are both high).

At the end of P1, A50-5 goes high and A50-6 goes low; A50-6 asserts the SLAVE MODE/ signal. The outputs of A50-6 and A49-7 are ANDed to hold A50-5 in the preset (high) state. At the end of P2, A49-14 goes low and asserts the SLAVE CMD EN/ signal, which gates DP RD/ or DP WRT/ to RAM Controller A70 (10ZB6); SLAVE CMD EN/ also gates the subsequently generated RAM XACK/ to the CPU READY input. (RAM XACK/ is generated by the RAM Controller when data has been read from or written into RAM.)

The RAM Controller asserts RAM XACK/ during P13 and A49-10 goes low on the next rising edge of the clock. The bus master then terminates the DP RD/ or DP WRT/ signal and the OFF BD CMD signal. The RAM controller next terminates RAM XACK/ and then A49-7 goes high on the next rising edge of the clock. At the end of P16, A50-5 goes low and A50-6 goes high (terminating the SLAVE MODE/ signal). At the end of P17, A49-14 goes high and terminates the SLAVE CMD EN/ signal.

The foregoing discussion pertains only to the operation of the Dual Port Control Logic for Multibus access of the dual port RAM. The actual addressing and transfer of data are discussed in paragraph 4-35.

**4-25. CPU ACCESS TIMING.** Figure 4-8 illustrates the Dual Port Control Logic timing for dual port RAM access by the on-board 8086 CPU. (P- periods P0 through P13 are used only for descriptive purposes and have no relationship to the 22.12-MHz clock signal.) To demonstrate that the CPU has priority in the access of the dual port RAM, figure 4-8 shows the OFF BD RAM CMD signal active when the CPU access is initiated by the ON BD RAM RQT/ signal. The timing has progressed through P0, during which time A49-10 has been clocked high and A49-7 has been clocked low.

Flip-Flop A50-9 is preset (high) when the Status Decoder asserts the ALE/ signal at the beginning of T1 in the CPU instruction cycle. When the ON BD RAM RQT/ signal is asserted, the EXT ALE/ signal goes low and, since A51-6 is now low, A49-10 goes low on the next rising edge of the clock. Flip-flop A50-5 is thus prevented from being clocked high and therefore keeps the DP ON BD ADR/ signal asserted; A50-6 remains high and suppresses the SLAVE MODE/ signal.

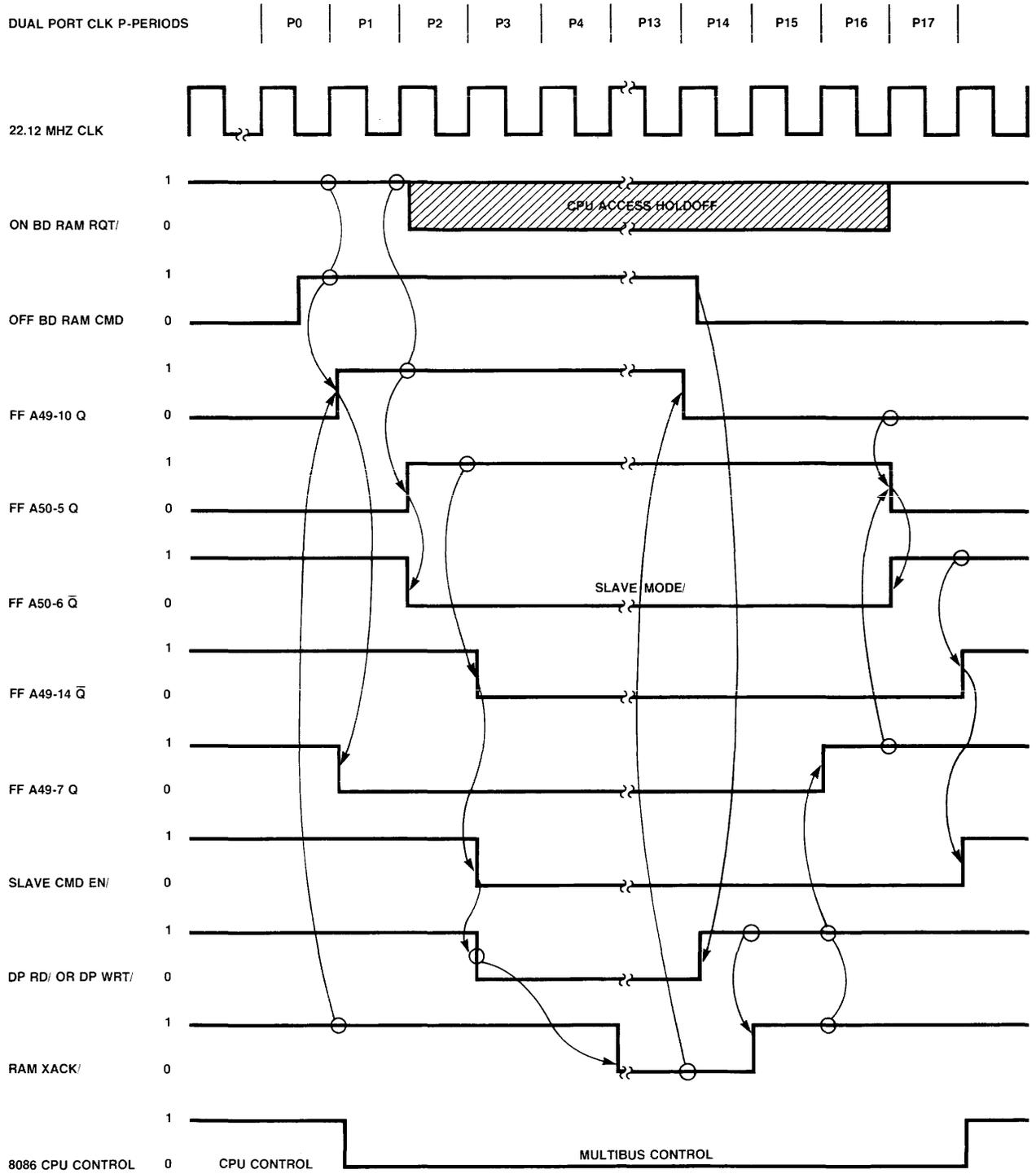
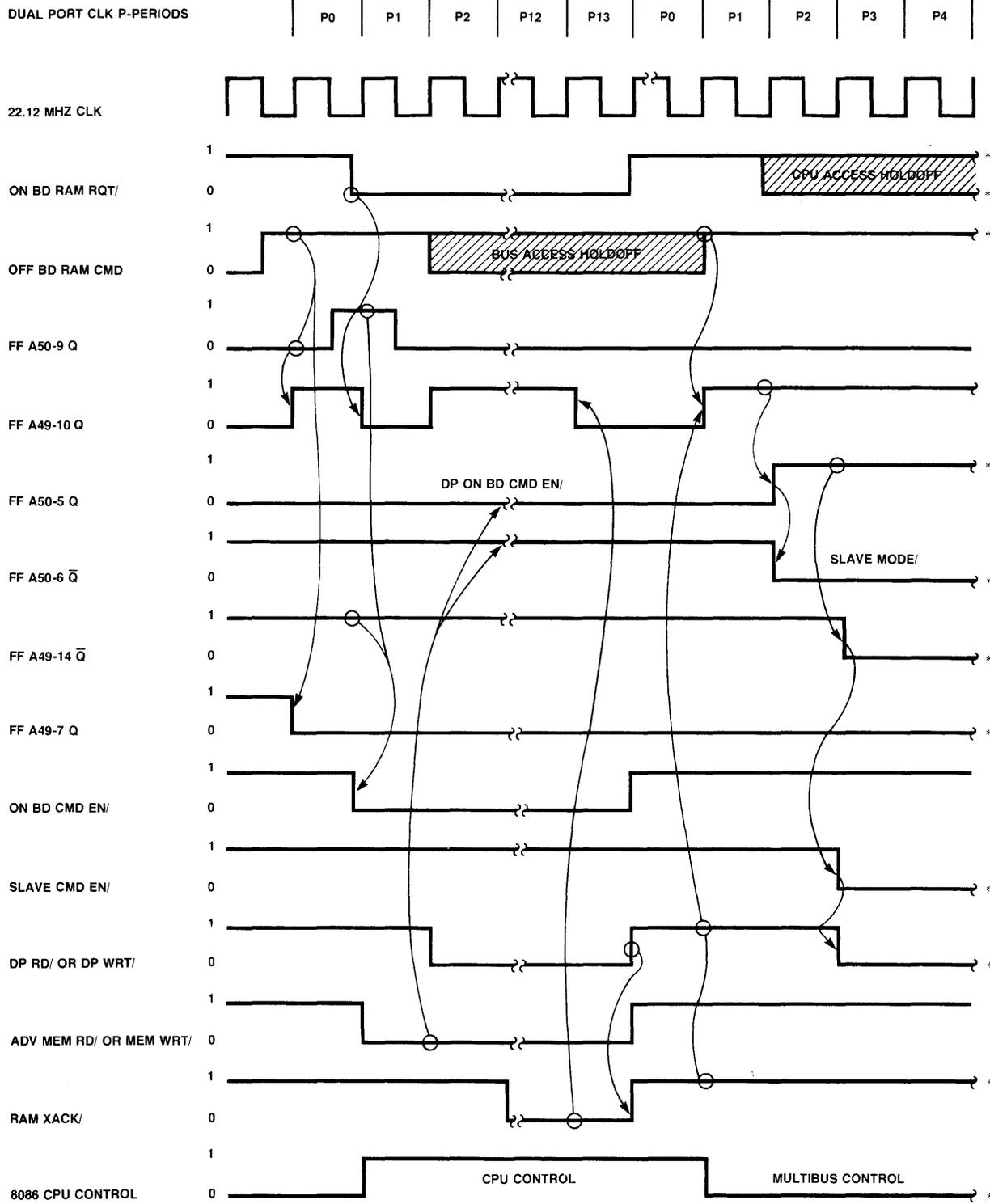


Figure 4-7. Dual Port Control Multibus Access Timing With CPU Lockout



\*FOR REMAINDER OF MULTIBUS ACCESS TIMING. SEE FIG. 4-7 BEGINNING WITH P3.

Figure 4-8. Dual Port Control CPU Access Timing With Multibus Lockout

The ON BD CMD EN/ signal is asserted at the same time as the ON BD RAM RQT/ signal since A49-14 is high. The ADV MEM RD/ or MEM WRT/ signal from the Status Decoder is ORed with the ON BD RAM RQT/ signal to prevent A50-5 and A50-6 from changing states when ALE/ goes false at the end of T1 in the instruction. (A49-10 is allowed to go high on the next rising edge of the clock after ALE/ goes false.)

The subsequently generated DP RD/ or DP WRT/ signal, gated by the asserted ON BD CMD EN/ signal, is transmitted to RAM Controller A70 (10ZB6). When the read or write is completed, the RAM Controller asserts RAM XACK/ and A49-10 goes low at the end of P12. At the end of P13, the CPU terminates the instruction and the ON BD RAM RQT/, DP RD/ or DP WRT/, and ADV MEM/ or MEM WRT/ signals go false. The RAM XACK/ signal is then terminated and A49-10 goes high at the end of P0. At the end of P1, the SLAVE MODE/ is entered when A50-5 goes high and A50-6 goes low.

The foregoing discussion pertains only to the operation of the Dual Port Control Logic for CPU access of on-board RAM. The actual addressing and transfer of data are discussed in paragraph 4-34.

#### 4-26. MULTIBUS INTERFACE

The Multibus interface consists of the Bus Arbiter Assembly (3ZD3), Bus Command Decoder A83 (3ZC3), bidirectional Address Bus Driver A87/88 (5ZC3), bidirectional Data Bus Driver A69/89/90 (4ZB3), and the Slave RAM Decode Logic (figure 5-2 sheet 3).

The falling edge of BCLK/ provides the bus timing reference for the Bus Arbiter, which allows the iSBC 86/1X to assume the role of a bus master. When the ON BD ADR/ signal is false (high) and the S0-S2 status signals indicate either a read or write operation, the Bus Arbiter drives BREQ/ low and BRPO/ high. The BREQ/ output from each bus master in the system is used by the Multibus when the bus priority is resolved by a parallel priority scheme as described in paragraph 2-19. The BPRO/ output is used by the Multibus when the bus priority is resolved by a serial priority scheme as described in paragraph 2-18.

The iSBC 86/12 gains control of the Multibus when the BPRN/ input to the Bus Arbiter is driven low. On the next falling edge of BCLK/, the Bus Arbiter drives BUSY/ and BUS ADEN/ low. The BUSY/ output indicates that the bus is in use and that the current bus master in control will not relinquish control until it raises its BUSY/ signal.

The BUS ADEN/ output, which can be thought of as a "master bus control" signal, is applied to the AEN2/ input of Clock Generator A38 (2ZC6), the Bus Address Driver (sheet 5), and the input of gate A2-11 (3ZC4).

With AEN2/ enabled, the Clock Generator is prepared to recognize the ensuing acknowledge signal (AACK/ or XACK/) transmitted by the addressed system device. To ensure adequate setup for the address and data, counter A4 (2ZB5) is held in the clear state as long as ALE/ is asserted. When ALE/ goes false, A4-3 is clocked low by the 5-MHz clock to generate T21/. This signal (T21/) is driven through gate A2-11 to enable the Bus Command Decoder.

The false ON BD ADR/ signal also enables the Bus Command Decoder, which decodes S0-S2 and drives the appropriate command low on the Multibus when T21/ occurs. The Bus Command Decoder also drives BUS DEN high to enable Data Bus Driver A69/89. The Data Bus Driver is switched to the appropriate "transmit" or "receive" mode depending on the state of the DT/ $\bar{R}$  output of Status Decoder A81.

After the command is acknowledged (signified by the addressed device driving the Multibus XACK/ line low), the CPU terminates the appropriate command. The Bus Arbiter and Bus Command Decoder, respectively, terminate BUS ADEN/ and BUS DEN; the Bus Arbiter also relinquishes control of the Multibus by driving BREQ/ high and BPRO/ low and then raising BUSY/.

It should be noted that, after gaining control of the Multibus, the iSBC 86/12 can invoke a "bus lock" condition to prevent losing control at a critical time. (For instance, it may be desired to execute several consecutive commands without having to contend for the bus after each command is executed.) The "bus lock" condition is invoked by driving the Bus Arbiter LOCK input low in one of two ways:

- By executing a software LOCK XCNG command.
- By clearing an option bit via I/O Port CC.

During an interrupt from the 8259A PIC, the LOCK input is automatically driven low by the first of two INTA/ signals issued by Status Decoder A81. (Refer to paragraphs 4-37 through 4-39.)

#### 4-27. I/O OPERATION

The following paragraphs describe on-board and system I/O operations. The actual functions performed by specific read and write commands to on-board I/O devices are described in Chapter 3.

**4-28. ON-BOARD I/O OPERATION.** Address bits AB3-ABF are applied to the I/O Address Decoder composed of A54/55/56 (6ZA7). The ADV I/O ADR signal is developed by flip-flop A63-5 (2ZA2) when the ALE signal latches the CPU inverted S2 signal. When ADV I/O ADR is true, the I/O Address Decoder develops IO AACK/ when AB8-ABF are false, AB6-AB7 are true,

and AB5 is either true or false. The I/O AACK/ signal enables decoder A54, which then decodes AB3-AB4. (The I/O AACK signal also drives the CPU READY input high.) Assuming AB8-ABF are false, AB3-AB7 are decoded to generate the following chip select signals:

Bits 7 6 5 4 3	Addresses*	Chip Select Signal
1 1 0 0 0	C0, C2	8259CS/
1 1 0 0 1	C8, CA, CC, CE	8255CS/
1 1 0 1 0	D0, D2, D4, D6	8253CS/
1 1 0 1 1	D8, DA, DC, DE	8251CS/
*Odd address (i.e., C1, C3, . . . DD) are invalid.)		

The IO AACK/ signal is driven through A32-8 and A6-8, respectively, to develop PROM IO EN/ and ON BD ADR/. PROM IO EN/ enables Data Buffer A44/45 (4ZD4) and ON BD ADR/ inhibits the Bus Arbiter and Bus Command Decoder. The DT/R output of Status Decoder A81 is inverted to select the proper direction of data transfer through the Data Buffer.

After the proper I/O device is enabled, the specific function for the device is selected by address bits AB0-AB1 and the IORC/ or IOWC/ output of Status Decoder A81.

**4-29. SYSTEM I/O OPERATION.** Address bits AB3-ABF are decoded by the I/O Address Decoder as described in paragraph 4-27. If the address is not for an on-board I/O device, the ON BD ADR/ signal is false (high) and enables the Bus Arbiter Assembly and Bus Command Decoder A53. (Refer to figure 5-2 sheet 3.) The Bus Arbiter and Bus Command Decoder, which are clocked by the 5-MHz clock to latch in and decode status signals S0-S2, then acquire control of the Multibus as described in paragraph 4-26.

#### 4-30. ROM/EPROM OPERATION

The four ROM/EPROM chips are installed by the user in IC sockets A28/29/46/47. (Refer to figure 5-2 sheet 6.) The ROM/EPROM addresses are assigned from the top down in the 1-megabyte address space; the bottom address is determined by the user configuration of chips as follows:

ROM	EPROM	Address Block
—	2758	FF000-FFFFF
2316E	2716	FE000-FFFFF
2332	—	FC000-FFFFF

Jumper posts 94 through 99 and switch S1 must be properly configured to accommodate the type of ROM/EPROM installed. (Refer to table 2-4.)

IC sockets A29 and A47 accommodate the top of ROM/EPROM; IC sockets A28 and A46 accommodate the ROM/EPROM space directly below that installed in A29 and A47. The low-order bytes (bits DB0-DB7) are installed in A29 and A28; the high-order bytes (bits DB8-DBF) are installed in A47 and A46.

When ADV IO ADR is false, a custom ROM (A68) (6ZB6) decodes address bits ABB-AB12. If the address is within the limit specified above, the O4 and O3 output pins will be low and the O2 and O1 output pins will depend on whether the address is in the upper half or lower half of the address block. For instance, if 2758 EPROM chips are installed and the address is in the range FF000-FF7FF, the O2 and O1 pins will be high and low, respectively; if the address is in the range FF800-FFFFF, the O2 and O1 pins will both be high. The O4 and O3 output pins are compared with address bit AB13. If AB13 is high, the PROM AACK/ signal is asserted; if AB13 is low, the ON BD RAM RQT/ signal is asserted.

When ALE goes false, Decoder A18 (6ZC4) is enabled and decodes the inputs presented by the O2 and O1 output of A68. If O2/O1 = 10, PCS2/ is asserted and enables A28 and A46; if O2 and O1 = 11, PS3/ is asserted and enables A29 and A47. Each chip of the selected pair of chips are individually addressed by AB1-ABA. Thus, when the associated enable signal (PCS2/ or PCS3/) is asserted, the contents of the address specified by AB1-ABA are transferred to the CPU via Data Buffer A44/45.

#### 4-31. RAM OPERATION

As described in paragraph 4-22, the Dual Port Control logic allows the on-board RAM facilities to be shared by the 8086 CPU and another bus master via the Multibus. The following paragraphs describe the RAM Controller, RAM chip arrays, and the overall operation of how the RAM is addressed for read/write operation.

**4-32. RAM CONTROLLER.** All address and control inputs to the on-board RAM is supplied by RAM Controller A70 (10ZB6). The RAM Controller automatically provides a 64-cycle RAS/CAS refresh timing cycle to the dynamic RAM composed of RAM chips A72-79 and A92-99.

The RAM Controller, when enabled by a low input to its PCS/ pin, multiplexes the address to the RAM chips. Low-order address bits A0-A6 are presented at the RAM address lines and RAS/ is driven low at the beginning of the first memory clock cycle. High-order address bits A7-A13 are presented at the RAM address lines and CAS/ is driven low during the second memory clock cycle. The RAM Controller drives its WE/ output pin according to whether the CPU instruction is a read or write. For a write operation, the WT/ input is low to the RAM Controller, in

which case the WE/ output is driven low. For a write operation, the WR/ input is low and the WE/ output remains high. When the memory cycle (read or write) starts, the RAM Controller drives its SACK/ output low; when the memory cycle is complete, it drives its XACK/ output low. The SACK/ and XACK/ go high when the RD/ or WR/ input goes high.

**4-33. RAM CHIPS.** Even bytes of data are stored in A72-A79 and odd bytes of data are stored in A92-A99. The WE/ input pin to A72-A79 is controlled by ANDing the RAM Controller WE/ output and memory address bit AM0. The WE/ input pin to A92-A99 is controlled by ANDing the RAM Controller WE/ output, AM0, and MBHEN/ (Memory Byte High Enable).

**4-34. ON BOARD READ/WRITE OPERATION.** When the O4 output of A68 (6ZB6) and address bit AB13 are both low, the output of A53-6 goes low and asserts the ON BD RAM RQT/ signal. When ON BD RAM RQT/ goes low, A52-3 (11ZA3) is enabled and generates ON BDCMD EN/ to generate RAMCS via A52-11 and to gate DPRD/ or DRWT/ to the RAM Controller. (See Figure 4-8.) The RAM Controller then multiplexes the address to RAM and, depending on which input command it true (DPRD/ or DPWT/), drives its WE/ output high or low. (The WE/ output is driven low for a write; it remains high for a read.) The SACK/ and XACK/ signals are generated by the RAM Controller as described in paragraph 4-33. The CPU completes the read or write operation when XACK/ is asserted.

During the CPU access of on-board RAM, the Address Bus Drivers and Data Bus Drivers are disabled and the Address Buffer and Data Buffer are enabled.

**4-35. BUS READ/WRITE OPERATION.** When another bus master has control of the Multibus, that bus master can address the iSBC 86/12 as a slave RAM device. The bus master first places the address on the Multibus and then asserts MRDC/ or MWTC/. Address bits ADRD/-ADR10/ and switch S1 present a 10-bit address to a special ROM (A67) (3ZB6); address bits ADRD/-ADR13/ are decoded by A66. The switch settings of S1 represent the base address and memory bus size; the O1-O3 outputs of A67 are ATRD/-ATRF/, which are multiplexed by A86 (5ZC4) into memory address bits AMC-AMF when the SLAVE MODE/ signal is subsequently activated by the Dual Port Control Logic. The O4 output of A67 is driven through A23-4 (when the 128K byte matches) to develop the OFF BD RAM ADR RQT signal, which is applied to the Dual Port Control Logic. If no CPU access is in progress, the Dual Port Control Logic then enters the slave mode and, when

A49-10 goes low, develops the RAMCS and SLAVE CMD EN/ signals. RAMCS enables RAM Controller A70 and SLAVE CMD EN/ gates DPRD/ or DPWT/ to the RAM Controller. The RAM Controller then multiplexes the address to RAM and, depending on which input command is true (DPRD/ or DPWT/), drives its WE/ output high or low. (The WE/ output is driven low for a write; it remains high for a read.) The SACK/ and XACK/ signals are generated by the RAM Controller as described in paragraph 4-34. The CPU completes the read or write operation when XACK/ is asserted.

During the Multibus access of on-board RAM, the SLAVE MODE/ signal enables the Address Bus Drivers (A86/87/88); the ON BD ADR/ signal is false and enables the Data Bus Drivers (A69/89).

**4-36. BYTE OPERATION.** For Multibus operation, the on-board RAM is organized as two 8-bit data banks; all even byte data is in one bank (DAT0/-DAT7/) and all odd byte data is in the other bank (DAT8/-DATF/). Refer to figure 3-1 which shows the data path for Multibus operation by 8-bit and 16-bit bus masters.

The Byte High Enable (BHEN/) signal, when asserted, access the high (odd) byte; address bit ADR0/, when low, access the low (even) byte. All word operations must occur on an even byte address boundary with BHEN/ asserted. Byte operations can occur in one of two ways:

- a. The even bank can be accessed by controlling ADR0/, which places the data on the DAT0/-DAT7/ lines. (Refer to figure 3-1A.)
- b. To access the odd bank, which is normally placed on DAT8/-DATF/, the data path shown in figure 3-1B is implemented. This requires that BHEN/ be false and ADR0/ to be low.

These operations permit the access of both bytes of the 16-bit data word by controlling ADR0/. In other words, ADR0/ therefore specifies a unique byte and is not a part of a 16-bit word operation.

Shown below are the states of BHEN/ and ADR0/ for 8-bit and 16-bit operations and the effects on transceiver control and memory block chip select.

Bus Control Lines		Data Bus Driver Chip Select			Memory Block Chip Select	
BHEN/	ADR0/	A69	A89	A90	A72-A79	A92-A99
1	1	On	On	Off	Yes	No
1	0	Off	Off	On	No	Yes
0	1	On	On	Off	Yes	Yes
0	0	Off	Off	On	No	Yes

#### 4-37. INTERRUPT OPERATION

The 8259A PIC can support both bus vectored (BV) and non-bus vectored (NBV) interrupts. For both BV and NBV interrupts, the on-board PIC (A24) (8ZB6) serves as the master PIC. (Refer to paragraph 2-13:) The master PIC drives the CPU INTR input high to initiate an interrupt request and the CPU then enters the interrupt timing cycle in which two INTA cycles occur back-to-back. The NBV and BV interrupts are described in following paragraphs.

**4-38. NBV INTERRUPT.** Assume that a NBV interrupt is initiated by an on-board function driving the IR5 line high to the on-board PIC; if no higher interrupt is in progress, the PIC then drives the CPU INTR input high. Assuming that the NMI interrupt is inactive and that the CPU interrupt enable flip-flop is set, the CPU suspends the current operation and proceeds with the first of two back-to-back INTA cycles. (Refer to figure 4-6 for signals activated during the first and subsequent INTA cycle.)

The Bus Arbiter acquires control of the Multibus and the MCE signal drives the LOCK/ signal low to ensure Multibus control until the second INTA cycle is complete. The Bus Command Decoder drives the INTA/ signal low. On receipt of the first INTA/ signal, the master PIC freezes the internal state of its priority resolution logic. The first INTA/ signal also sets flip-flop A63-5 (8ZA2), which generates the 1st ACK/ signal to drive the CPU READY input high.

The CPU then proceeds with the second INTA cycle. On receipt of the second INTA/ signal, the master PIC places an 8-bit identifier for IR5 on the data bus, and drives its DEN/ output low. The resultant LOCAL INTA DEN/ signal enables Data Buffer A44 and drives the CPU READY input high. (The second INTA/ signal clears

flip-flop A63-5.) The CPU then inputs the 8-bit identifier and terminates the interrupt timing cycle.

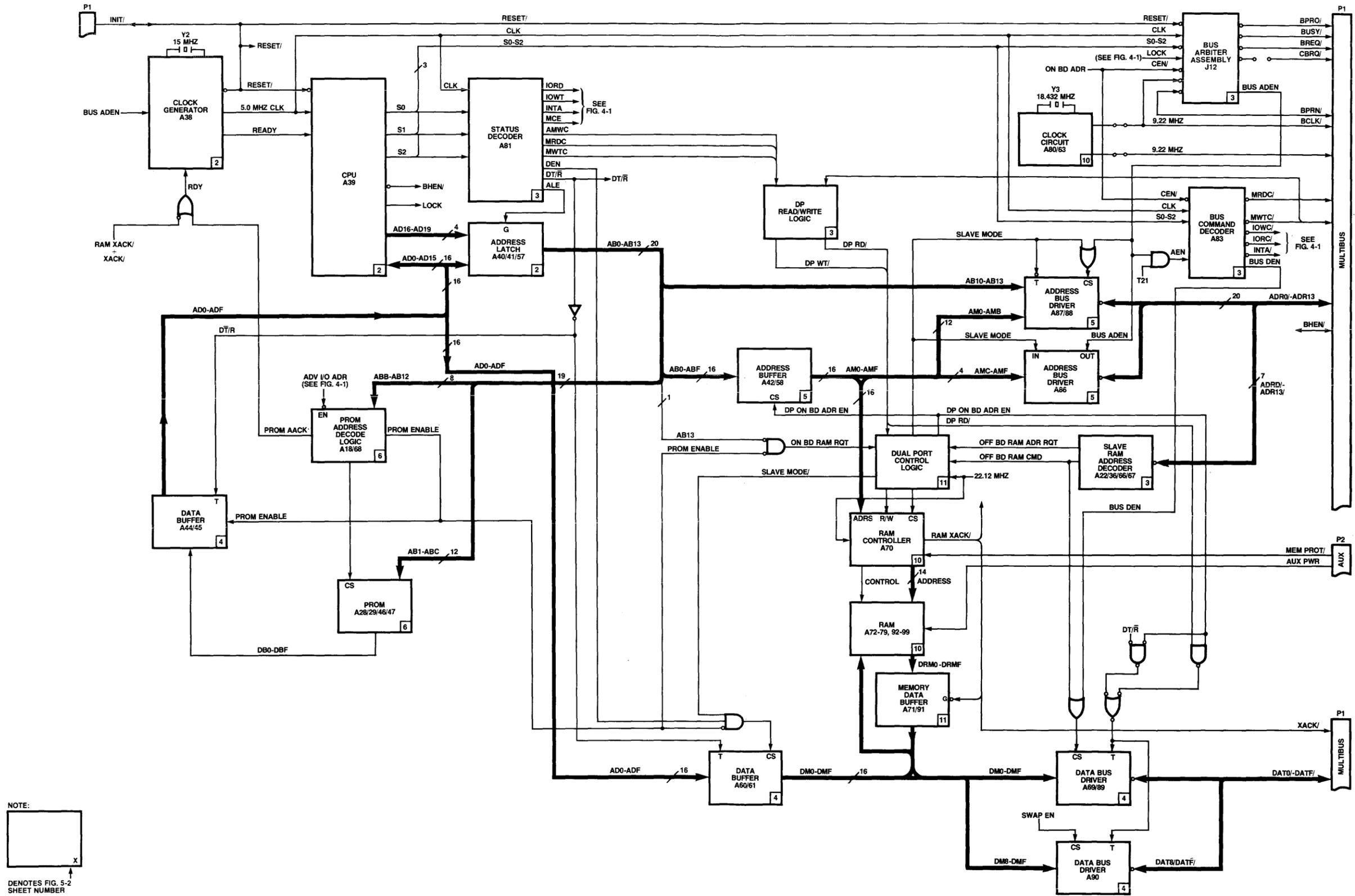
The CPU multiplies the 8-bit identifier by four to derive the restart address of the interrupting device. After the service routine is completed, the CPU automatically resets all its affected flags and returns to the main program.

**4-39. BV INTERRUPT.** As far as the CPU is concerned, BV interrupts are handled exactly the same as NBV interrupts. Assume that the IR6 line to the master PIC is driven by a slave PIC on the Multibus. When IR6 goes high, the master PIC drives the CPU INTR input high as previously described. On receipt of the first INTA/ signal, the master PIC generates BUS INTA DEN/ via its DEN/ output and places the interrupt address code for IR6 on its C0-C2 pins; since QMCE/ is enabled by the MCE output of the Status Decoder, the C0-C2 is transferred to the Address Latch via address lines AD8-ADA. (These bits are latched when the ALE signal goes false.) The BUS INTA DEN/ signal enables the Data Bus Driver in preparation to receive the 8-bit identifier from the slave PIC. (The interrupt address code is now on Multibus address lines ADR8/-ADRA/.)

The first INTA/ signal sets flip-flop A63-5 to drive the CPU READY input high. The CPU then proceeds with the second INTA cycle. When the second INTA/ signal is driven onto the Multibus and the slave PIC recognizes its address, it outputs an 8-bit identifier onto the DAT0/-DAT7/ lines and drives the Multibus XACK/ line low. (The second INTA/ also toggles and clears flip-flop A63-5.) The CPU then inputs the 8-bit identifier and terminates the interrupt timing cycle.

The CPU multiplies the 8-bit identifier by four to derive the restart address of the interrupting device. After the service routine is completed, the CPU automatically resets all its affected flags and returns to the main program.





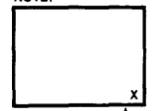
NOTE:  
  
 DENOTES FIG. 5-2 SHEET NUMBER

Figure 4-2. iSBC 86/1X ROM/EPROM and Dual Port RAM Simplified Logic Diagram



# CHAPTER 5

## SERVICE INFORMATION

### 5-1. INTRODUCTION

This chapter provides a list of replaceable parts, service diagrams, and service and repair assistance instructions for the iSBC 86/12 Single Board Computer.

Telephone:

From Alaska or Hawaii call —  
(408) 987-8080

From locations within California call toll free —  
(800) 672-3507

From all other U.S. locations call toll free —  
(800) 538-8014

TWX: 910-338-0026

TELEX: 34-6372

### 5-2. REPLACEABLE PARTS

Table 5-1 provides a list of replaceable parts for the iSBC 86/12. Table 5-2 identifies and locates the manufacturers specified in the MFR CODE column in table 5-1. Intel parts that are available on the open market are listed in the MFR CODE column as "COML"; every effort should be made to procure these parts from a local (commercial) distributor.

Always contact the MCD Technical Support Center before returning a product to Intel for service or repair. You will be given a "Repair Authorization Number", shipping instructions, and other important information which will help Intel provide you with fast, efficient service. If the product is being returned because of damage sustained during shipment from Intel, or if the product is out of warranty, a purchase order is necessary in order for the MCD Technical Support Center to initiate the repair.

### 5-3. SERVICE DIAGRAMS

The iSBC 86/12 parts location diagram and schematic diagram are provided in figures 5-1 and 5-2, respectively. On the schematic diagram, a signal mnemonic that ends with a slash (e.g., IOWC/) is active low. Conversely, a signal mnemonic without a slash (e.g., INTR) is active high.

In preparing the product for shipment to the MCD Technical Support Center, use the original factory packaging material, if available. If the original packaging is not available, wrap the product in a cushioning material such as Air Cap TH-240 (or equivalent) manufactured by the Sealed Air Corporation, Hawthorne, N.J., and enclose in a heavy-duty corrugated shipping carton. Seal the carton securely, mark it "FRAGILE" to ensure careful handling, and ship it to the address specified by MCD Technical Support Center personnel.

### 5-4. SERVICE AND REPAIR ASSISTANCE

United States customers can obtain service and repair assistance from Intel by contacting the MCD Technical Support Center in Santa Clara, California, at one of the following numbers:

#### NOTE

Customers outside of the United States should contact their sales source (Intel Sales Office or Authorized Intel Distributor) for directions on obtaining service or repair assistance.

**Table 5-1. Replaceable Parts**

Reference Designation	Description	Mfr. Part No.	Mfr. Code	Qty.
A1,37,62	IC, 74125, Quad Bus Buffer (3-state)	SN74125	TI	3
A2,21,53	IC, 74S32, Quad 2-Input Positive-OR Gate	SN74S32	TI	3
A3,7	IC, 74S10, Triple 3-Input Positive-NAND Gate	SN74S10	TI	2
A4,49	IC, 74S175, Hex Quad D-Type Flip-Flop	SN74S175	TI	2
A5	IC, 9602, Dual One-Shot Multivibrator	9602PC	FAIR	1
A6,51	IC, 74S11, Triple 3-Input Positive-AND Gate	SN74S11	TI	2
A8,9	IC, Intel 8226, 4-Bit Bidirectional Bus Driver	8226	COML	2
A14	IC, 75189, Quad Line Receivers	SN75189	TI	1
A15	IC, 75188, Quad Line Drivers	SN75188	TI	1
A16	IC, 74163, Sync 4-Bit Counter	SN74163	TI	1
A17,80	IC, Intel 8224, Clock Generator and Driver	8224	COML	2
A18,54	IC, 74S139, Decoder/Multiplexer	SN74S139	TI	2
A19,32	IC, 74S08, Quad 2-Input Positive-AND Gate	SN74S08	TI	2
A20,34	IC, 74S04, Hex Inverters	SN74S04	TI	2
A22,59	IC, 7432, Quad 2-Input Positive-OR Gate	SN7432	TI	2
A23	IC, 74S02, Quad 2-Input Positive-NOR Gate	SN74S02	TI	1
A24	IC, Intel 8259A, Programmable Interrupt Controller	8259A	COML	1

Table 5-1. Replaceable Parts (Continued)

Reference Designation	Description	Mfr. Part No.	Mfr. Code	Qty.
A25	IC, Intel 8255A, Programmable Peripheral Interface	8255A	COML	1
A26	IC, Intel 8253, Programmable Interval Timer	8253	COML	1
A27	IC, Intel 8251A, Programmable Comm. Interface	8251A	COML	1
A30, 57	IC, 74LS75, 4-Bit Bistable Latch	SN74LS75	TI	2
A31,33,43,52	IC, 74S00, Quad 2-Input Positive-NAND Gate	SN74S00	TI	4
A35,84,85	IC, 74LS04, Hex Inverters	SN74LS04	TI	3
A36	IC, 7400, Quad 2-Input Positive-NAND Gate	SN7400	TI	1
A38	IC, Intel 8284, 18-Pin Clock Generator	8284L	COML	1
A39	IC, Intel 8086, 16-Bit Microprocessor	8086	COML	1
A40,41,71,91	IC, 74S373, Octal D-Type Latches	SN74S373	TI	4
A42,44,45,58,60,61	IC, Intel 8286, 8-Bit Non-Inverting Transceiver	8286	COML	6
A48	IC, 7438, Quad 2-Input Positive-NAND Gate	SN7438	TI	1
A50,63	IC, 74S74, Dual D-type Edge-Triggered Flip Flop	SN74S74	TI	2
A55	IC, 74S30, 8-Input Positive-NAND Gate	SN74S30	TI	1
A56	IC, 7425, Dual 4-Input Positive-NOR Gate w/Strobe	SN7425	TI	1
A64	IC, 74S140, Dual 4-Input Positive-NAND Gate	SN74S140	TI	1
A65	IC, 8097, 3-State Hex Buffers	DM8097	NAT	1
A66	IC, Intel 8205, 1-of-8 Decoder	8205	COML	1
A67	IC, PROM, Address Decoder	INTEL	9100134	1
A68	IC, PROM, Address Decoder	INTEL	9100129	1
A69,87-90	IC, Intel 8287, 8-Bit Inverting Transceiver	8287	COML	5
A70	IC, Intel 8202, Dynamic RAM Controller	8202	COML	1
A72-79,92-99	IC, Intel 2117-4, Dynamic RAM	2117-4	COML	16
A81,83	IC, Intel 8288, Bus Controller for 8086	8288	COML	2
A86	IC, 74S240, Octal Buffer:Line Driver:Line Receiver	SN74S240	TI	1
CR1,2	Diode, 1N9148	OBD	COML	2
C1,2,4-11,13,15-19, 21-25,28-51,65-75, 91 C3	Cap., mono. 0.1 $\mu$ F. +80 -20 $\%$ . 50V	OBD	COML	57
C12,27,64	Cap., mica, 10pF, $\pm$ 5 $\%$ . 500V	OBD	COML	3
C20,98	Cap., mono, 0.001 $\mu$ F. +20 $\%$ . 50V	OBD	COML	2
C26	Cap., tant, 10 $\mu$ F, $\pm$ 10 $\%$ . 20V	OBD	COML	1
C52,54,55,57,58,60,61 63,76,78,79,81,82,84, 85,87,92	Cap., mono, 0.33 $\mu$ F. +80 -20 $\%$ . 50V	OBD	COML	17
C53,56,59,62,77,80, 83,86	Cap., mono. 0.01 $\mu$ F. +80 -20 $\%$ . 50V	OBD	COML	8
C88-90,93-97	Cap., tant. 22 $\mu$ F, $\pm$ 10 $\%$ . 15V	OBD	COML	8
J12	Assembly, Bus Arbiter	1001794	INTEL	1
—	*IC, Bus Controller		INTEL	1
—	*IC, 74S00, Quad 2-Input Positive-NAND Gate	SN74S00	TI	1
—	*Resistor, fxd, comp. 270 ohm, $\pm$ 5 $\%$ . 1/4W	OBD	COML	1
—	*Resistor, fxd, comp. 2.2K, $\pm$ 5 $\%$ . 1/4W	OBD	COML	1
—	*Capacitor, mono. 0.1 $\mu$ F, +80 -20 $\%$ . 50V	OBD	COML	1
—	*Capacitor, mono, 220pF, $\pm$ 5 $\%$ . 500V	OBD	COML	1
RP1	Res., pack, 8-pin, 1K, $\pm$ 5 $\%$ . 2W PP	OBD	COML	1
RP2	Res., pack, 14-pin, 1K, $\pm$ 2 $\%$ . 1.5W PP	OBD	COML	1
RP3	Res., pack, 16-pin, 10K, $\pm$ 5 $\%$ . 2W PP	OBD	COML	1
RP4	Res., pack, 6-pin, 2.2K, $\pm$ 5 $\%$ . 1W PP	OBD	COML	1
R1,11,16,17	Res., fxd, comp, 10K, $\pm$ 5 $\%$ . 1/4W	OBD	COML	4
R2,22	Res., fxd, comp, 20K, $\pm$ 5 $\%$ . 1/4W	OBD	COML	2
R3-5,13,20	Res., fxd, comp, 5.1K, $\pm$ 5 $\%$ . 1/4W	OBD	COML	5
R7,8,10,14,18,19,21,23	Res., fxd, comp, 1K, 5 $\%$ . 1/4W	OBD	COML	8
R9	Res., fxd, comp, 100K, 5 $\%$ . 1/4W	OBD	COML	1
R12	Res., fxd, comp, 330 ohm, $\pm$ 5 $\%$ . 1/4W	OBD	COML	1
R15	Res., fxd, comp, 270 ohm, $\pm$ 5 $\%$ . 1/4W	OBD	COML	1
S1	Switch, 8-position, DIP	206-8	CTS	1
VR1	Voltage regulator	MC79L05AC	MOT	1

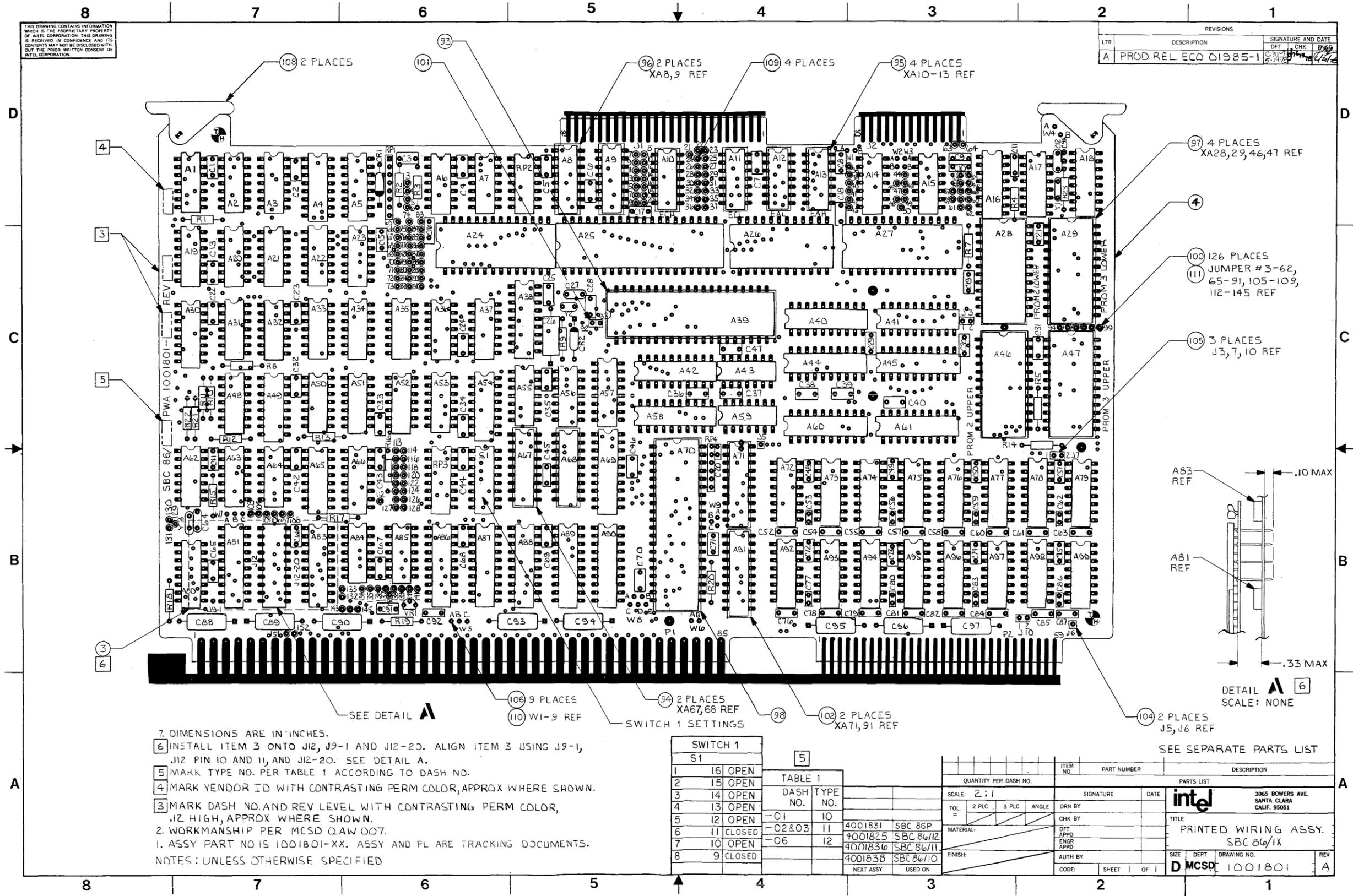
**Table 5-1. Replaceable Parts (Continued)**

Reference Designation	Description	Mfr. Part No.	Mfr. Code	Qty.
XA8,9	Socket, 16-pin, DIP	C-93-16-02	TI	2
XA10-13	Socket, 14-pin, DIP	C-93-14-02	TI	4
XA27	Socket, 28-pin, DIP	C-93-28-02	TI	1
XA28,29,46,47	Socket, 24-pin, DIP	C-93-24-02	TI	4
XA39,70	Socket, 40-pin, DIP	540-A37D	AUG	2
XA67,68	Socket, 18-pin, DIP	C-93-18-02	TI	2
XA71,91	Socket, 20-pin, DIP	C-93-20-02	TI	2
Y1	Crystal, 22.1184-MHz, fundamental	OBD	CTS	1
Y2	Crystal, 15-MHz, fundamental	OBD	CTS	1
Y3	Crystal, 18.432-MHz, fundamental	OBD	CTS	1
	Extractor, Card	S-203	SCA	2
	Post, Wire Wrap	89531-6	AMP	126
	Plug, Shorting, 2-position	530153-1	AMP	1

**Table 5-2. List of Manufacturers' Codes**

MFR. CODE	MANUFACTURER	ADDRESS	MFR. CODE	MANUFACTURER	ADDRESS
AMP	AMP, Inc.	Harrisburg, PA	NAT	National Semiconductor	Santa Clara, CA
AUG	Augat, Inc.	Attleboro, MA	SCA	Scanbe, Inc.	El Monte, CA
CTS	CTS Corp.	Elkhart, IN	TI	Texas Instruments	Dallas, TX
FAIR	Fairchild Semiconductor	Mt. View, CA	OBD	Order by Description; available from any commercial (COML) source.	
MOT	Motorola Semiconductor	Phoenix, AZ			





THIS DRAWING CONTAINS INFORMATION WHICH IS THE PROPRIETARY PROPERTY OF INTEL CORPORATION. THIS DRAWING IS RECEIVED IN CONFIDENCE AND ITS CONTENTS MAY NOT BE DISCLOSED WITHOUT THE PRIOR WRITTEN CONSENT OF INTEL CORPORATION.

REVISIONS			
LTR	DESCRIPTION	SIGNATURE AND DATE	DATE
A	PROD REL. ECO D1985-1	[Signature]	8-17-85

7. DIMENSIONS ARE IN INCHES.
- INSTALL ITEM 3 ONTO J12, J9-1 AND J12-20. ALIGN ITEM 3 USING J9-1, J12 PIN 10 AND 11, AND J12-20. SEE DETAIL A.
  - MARK TYPE NO. PER TABLE 1 ACCORDING TO DASH NO.
  - MARK VENDOR ID WITH CONTRASTING PERM COLOR, APPROX WHERE SHOWN.
  - MARK DASH NO. AND REV LEVEL WITH CONTRASTING PERM COLOR, .12 HIGH, APPROX WHERE SHOWN.
  - WORKMANSHIP PER MCSD QAW 007.
  - ASSY PART NO IS 1001801-XX. ASSY AND PL ARE TRACKING DOCUMENTS.
- NOTES: UNLESS OTHERWISE SPECIFIED

SWITCH 1	
S1	
1	16 OPEN
2	15 OPEN
3	14 OPEN
4	13 OPEN
5	12 OPEN
6	11 CLOSED
7	10 OPEN
8	9 CLOSED

TABLE 1	
DASH NO.	TYPE NO.
-01	10
-02&03	11
-06	12

ITEM NO.	PART NUMBER	DESCRIPTION
1	4001831	SBC 86P
2	4001825	SBC 86/12
3	4001836	SBC 86/11
4	4001838	SBC 86/10

PARTS LIST		SIGNATURE		DATE	
SCALE: 2:1	TOL: 2 PLC 3 PLC ANGLE	DRN BY			
MATERIAL:	FINISH:	CHK BY			
		DFT APPD			
		ENGR APPD			
		AUTH BY			
		CODE:	SHEET	OF	

Figure 5-1. iSBC 86/12 Parts Location Diagram







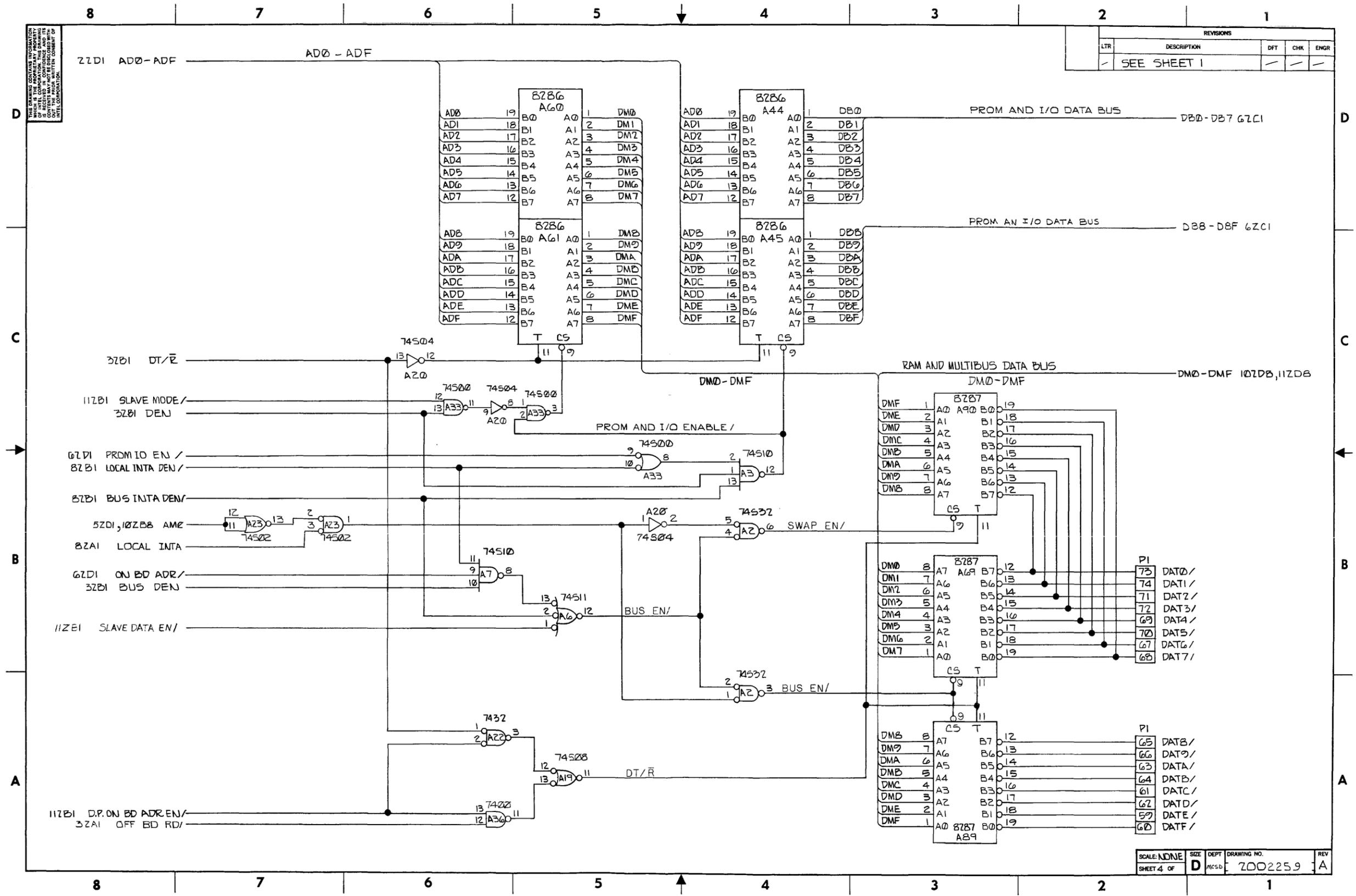


Figure 5-2. iSBC 86/12 Schematic Diagram (Sheet 4 of 11)

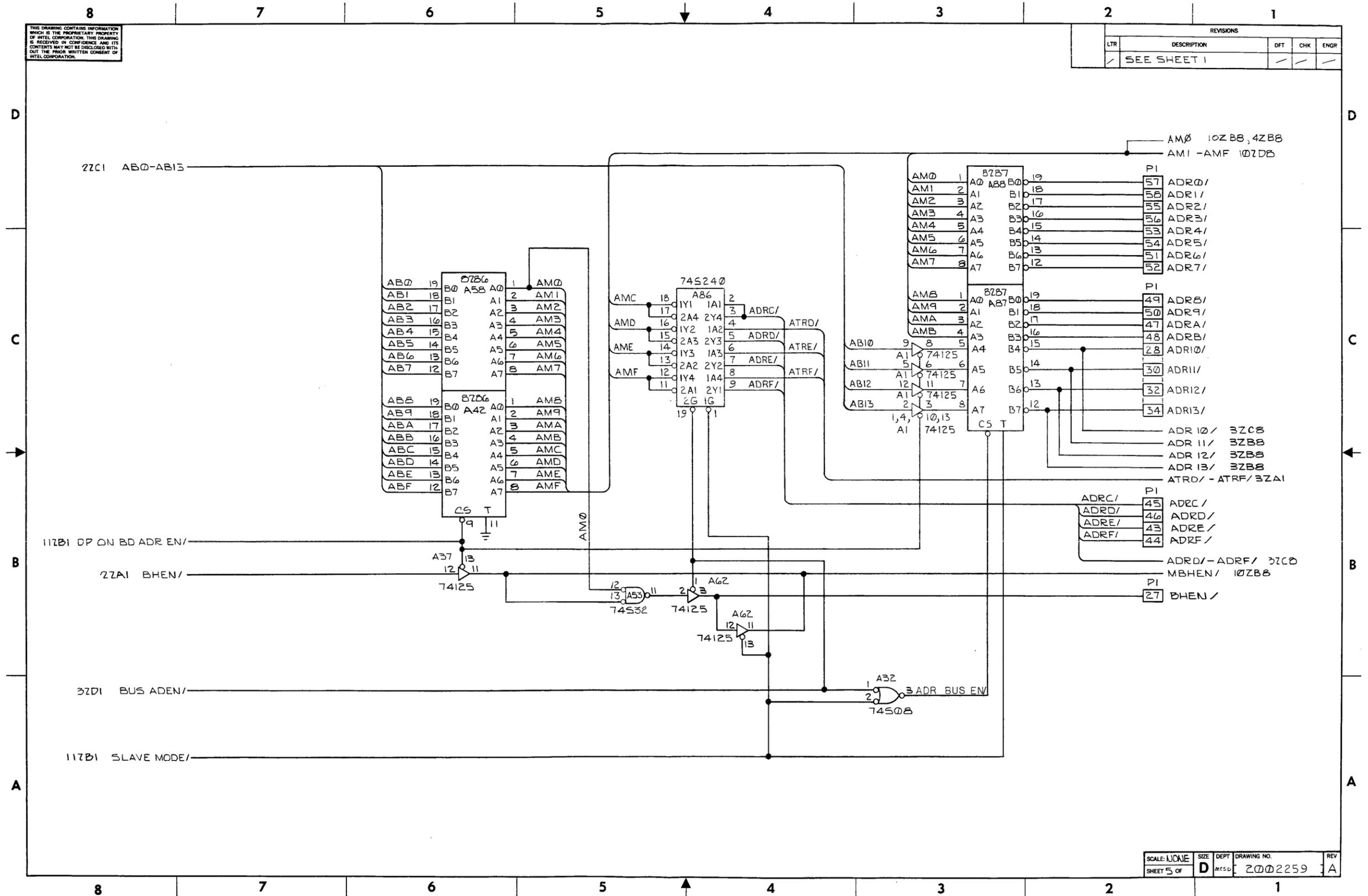


Figure 5-2. iSBC 86/12 Schematic Diagram (Sheet 5 of 11)



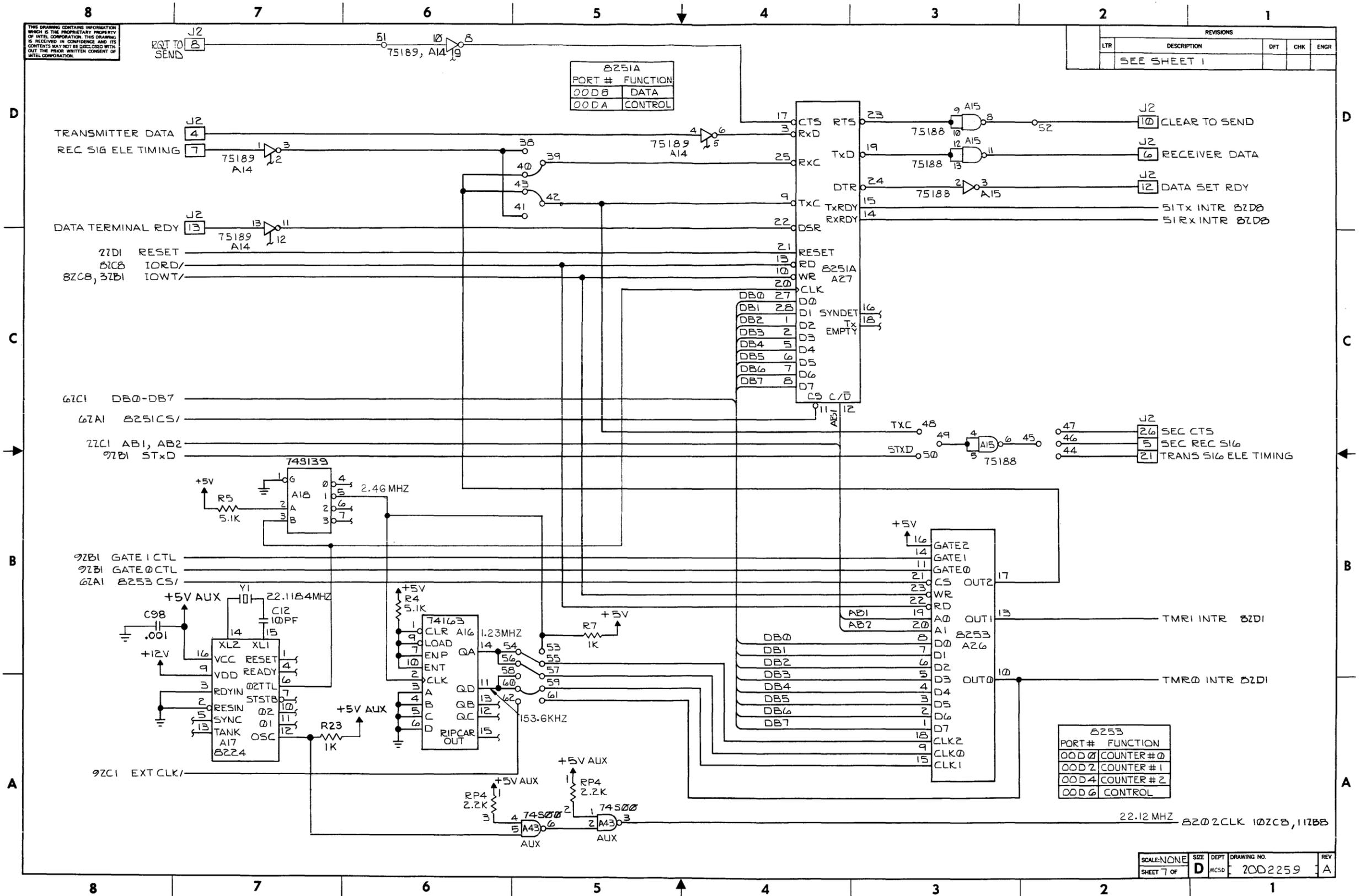


Figure 5-2. iSBC 86/12 Schematic Diagram (Sheet 7 of 11)







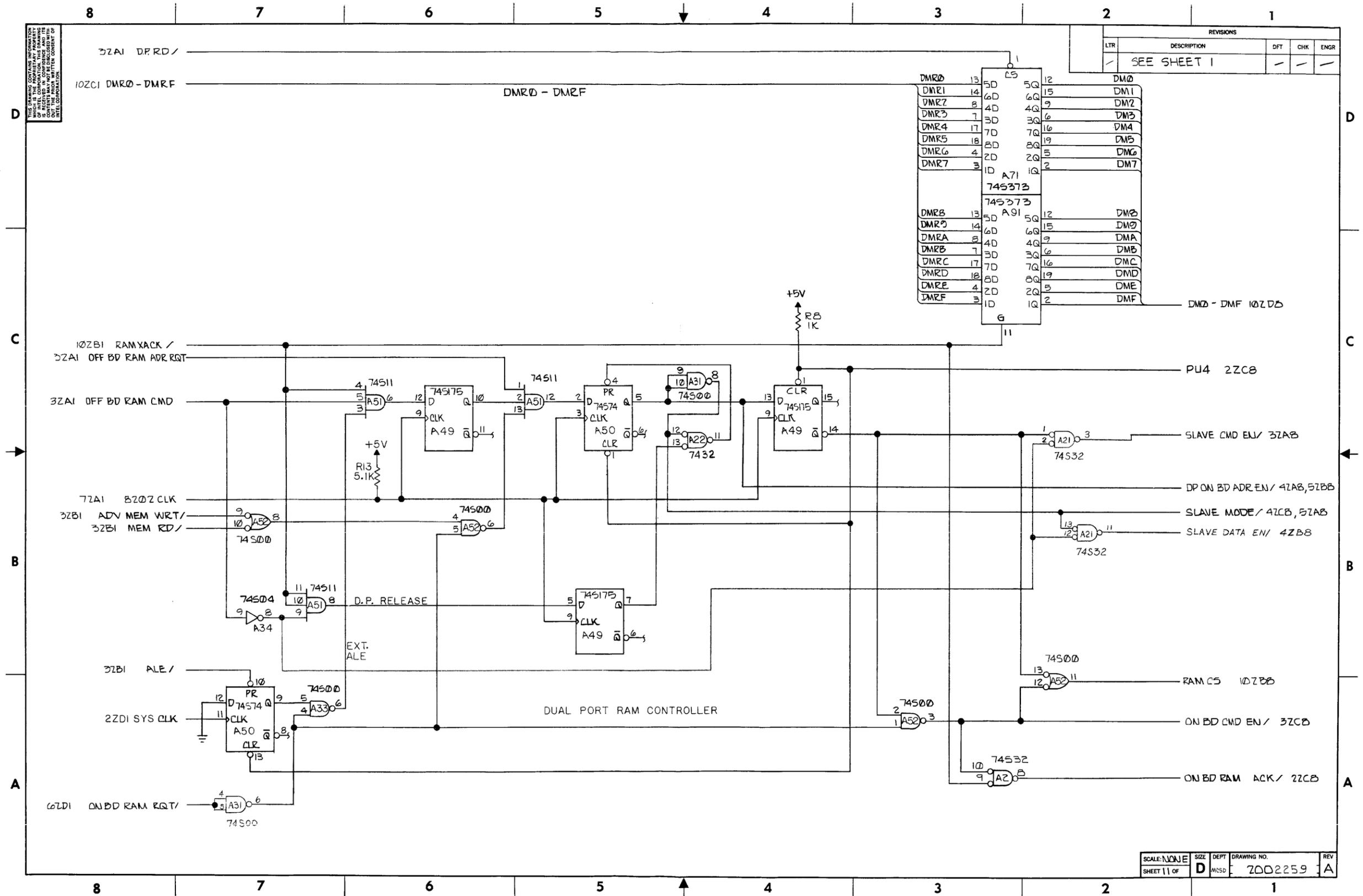
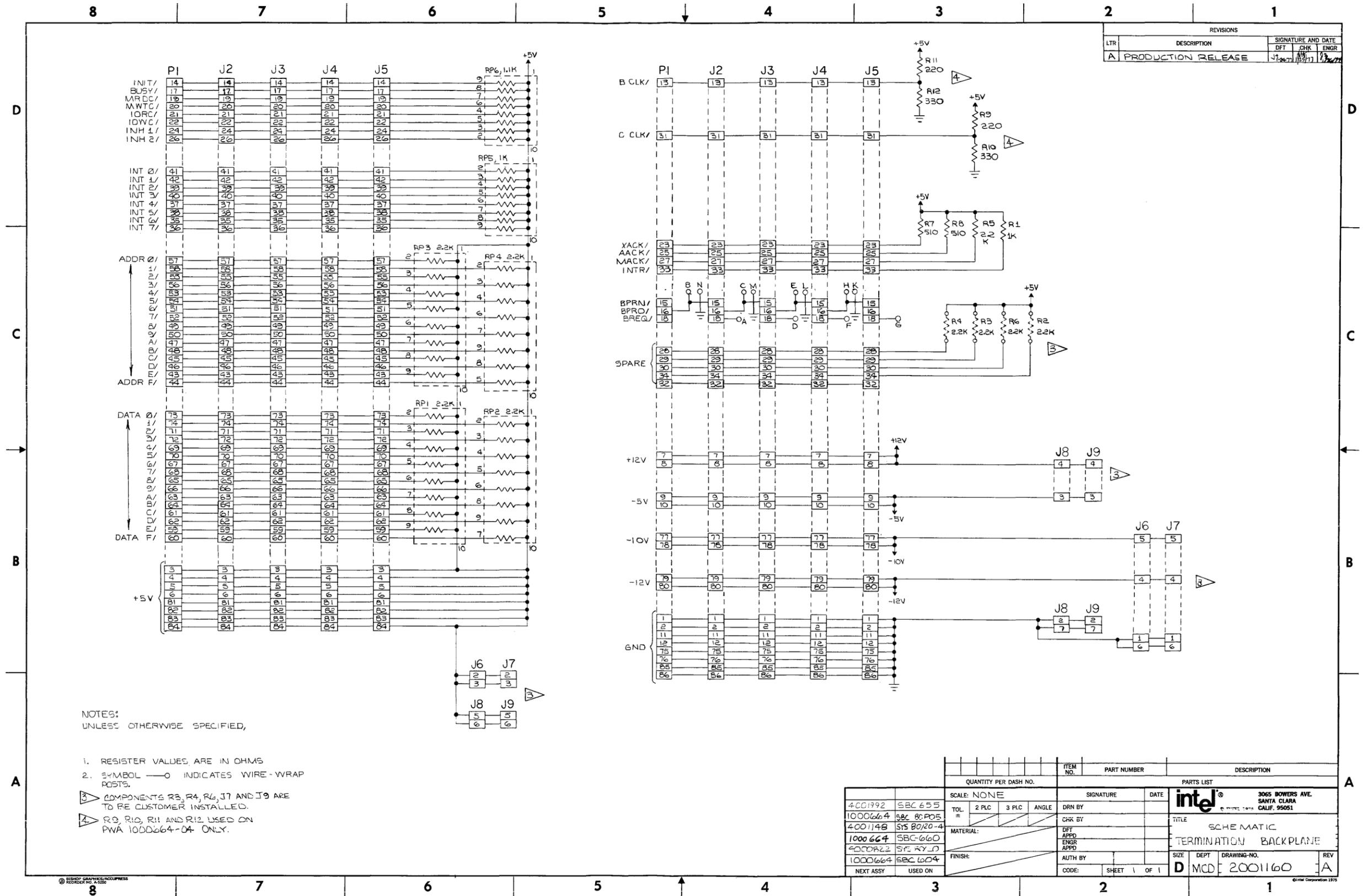


Figure 5-2. iSBC 86/12 Schematic Diagram (Sheet 11 of 11)



REVISIONS			
LTR	DESCRIPTION	SIGNATURE AND DATE	
		DFT	ENGR
A	PRODUCTION RELEASE		

- NOTES:  
UNLESS OTHERWISE SPECIFIED,
- RESISTOR VALUES ARE IN OHMS
  - SYMBOL INDICATES WIRE-WRAP POSTS.
  - COMPONENTS R3, R4, R6, J7 AND J9 ARE TO BE CUSTOMER INSTALLED.
  - R9, R10, R11 AND R12 USED ON PWA 100D664-04 ONLY.

QUANTITY PER DASH NO.		ITEM NO.	PART NUMBER	DESCRIPTION
400192	SBC 655			
1000664	SBC 80P05			
4001148	SYS 80/20-4			
1000664	SBC 660			
4000822	SYS XY 10			
1000664	SBC 604			
	NEXT ASSY		USED ON	

SCALE: NONE	SIGNATURE	DATE	 3065 BOWERS AVE. SANTA CLARA CALIF. 95051
TOL: 2 PLC 3 PLC ANGLE	DRN BY		
MATERIAL:	CHK BY		TITLE
FINISH:	DFT APPD		TERMINATION BACKPLANE
	ENGR APPD		
	AUTH BY		SIZE DEPT DRAWING-NO.
	CODE:	SHEET 1 OF 1	D MCD 2001160
			REV
			A

Figure 5-3. iSBC 604 Schematic Diagram

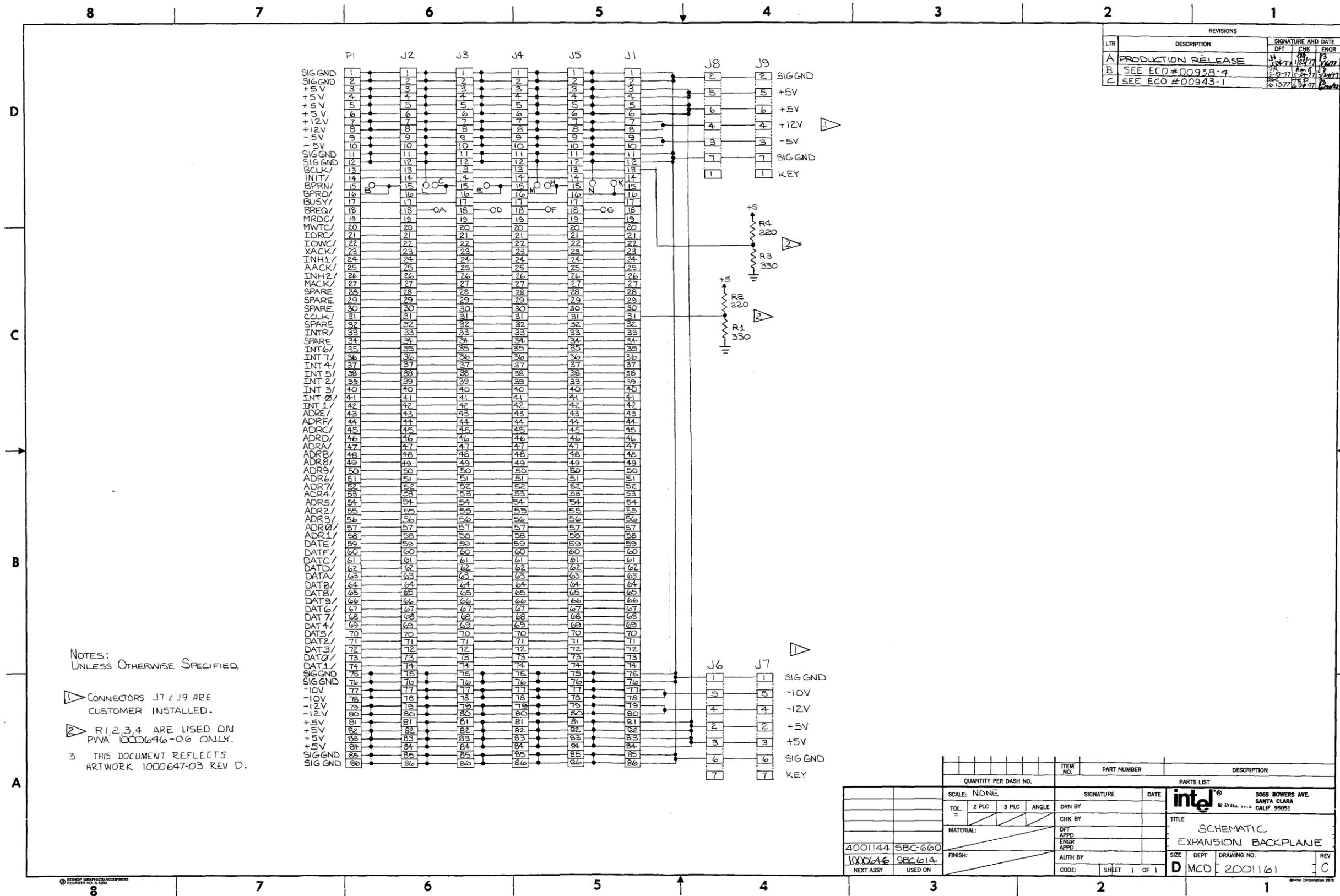


Figure 5-4. iSBC 614 Schematic Diagram

## A-1. INTRODUCTION

This appendix provides information required to modify a Model ASR-33 Teletypewriter for use with certain Intel iSBC 80 computer systems.

## A-2. INTERNAL MODIFICATIONS

### WARNING

Hazardous voltages are exposed when the top cover of the teletypewriter is removed. To prevent accidental shock, disconnect the teleprinter power cord before proceeding beyond this point.

Remove the top cover and modify the teletypewriter as follows:

- a. Remove blue lead from 750-ohm tap on current source register; reconnect this lead to 1450-ohm tap. (Refer to figures A-1 and A-2.)
- b. On terminal block, change two wires as follows to create an internal full-duplex loop (refer to figures A-1 and A-3):
  1. Remove brown/yellow lead from terminal 3; reconnect this lead to terminal 5.
  2. Remove white/blue lead from terminal 4; reconnect this lead to terminal 5.
- c. On terminal block, remove violet lead from terminal 8; reconnect this lead to terminal 9. This changes the receiver current level from 60 mA to 20 mA.

A relay circuit card must be fabricated and connected to the paper tape reader driver circuit. The relay circuit card to be fabricated requires a relay, a diode, a thyractor, a small 'vector' board for mounting the components, and suitable hardware for mounting the assembled relay card.

A circuit diagram of the relay circuit card is included in figure A-4; this diagram also includes the part numbers of the relay, diode, and thyractor. (Note that a 470-ohm resistor and a 0.1  $\mu$ F capacitor may be

substituted for the thyractor.) After the relay circuit card has been assembled, mount it in position as shown in figure A-5. Secure the card to the base plate using two self-tapping screws. Connect the relay circuit to the distributor trip magnet and mode switch as follows:

- a. Refer to figure A-4 and connect a wire (Wire 'A') from relay circuit card to terminal L2 on mode switch. (See figure A-6.)
- b. Disconnect brown wire shown in figure A-7 from plastic connector. Connect this brown wire to terminal L2 on mode switch. (Brown wire will have to be extended.)
- c. Refer to figure A-4 and connect a wire (Wire 'B') from relay circuit board to terminal L1 on mode switch.

## A-3. EXTERNAL CONNECTIONS

Connect a two-wire receive loop, a two-wire send loop, and a two-wire tape reader control loop to the external device as shown in figure A-4. The external connector pin numbers shown in figure A-4 are for interface with an RS232C device.

## A-4. iSBC 530 TTY ADAPTER

The iSBC 530, which converts RS232C signal levels to an optically isolated 20 mA current loop interface, provides signal translation for transmitted data, received data, and a paper tape reader relay. The iSBC 530 interfaces an Intel iSBC 80 computer system to a teletypewriter as shown in figure A-8.

The iSBC 530 requires +12V at 98 mA and -12V at 98 mA. An auxiliary supply must be used if the iSBC 80 system does not supply this power. A schematic diagram of the iSBC 530 is supplied with the unit. The following auxiliary power connector (or equivalent) must be procured by the user:

Connector, Molex 09-50-7071  
Pins, Molex 08-50-0106  
Polarizing Key, Molex 15-04-0219

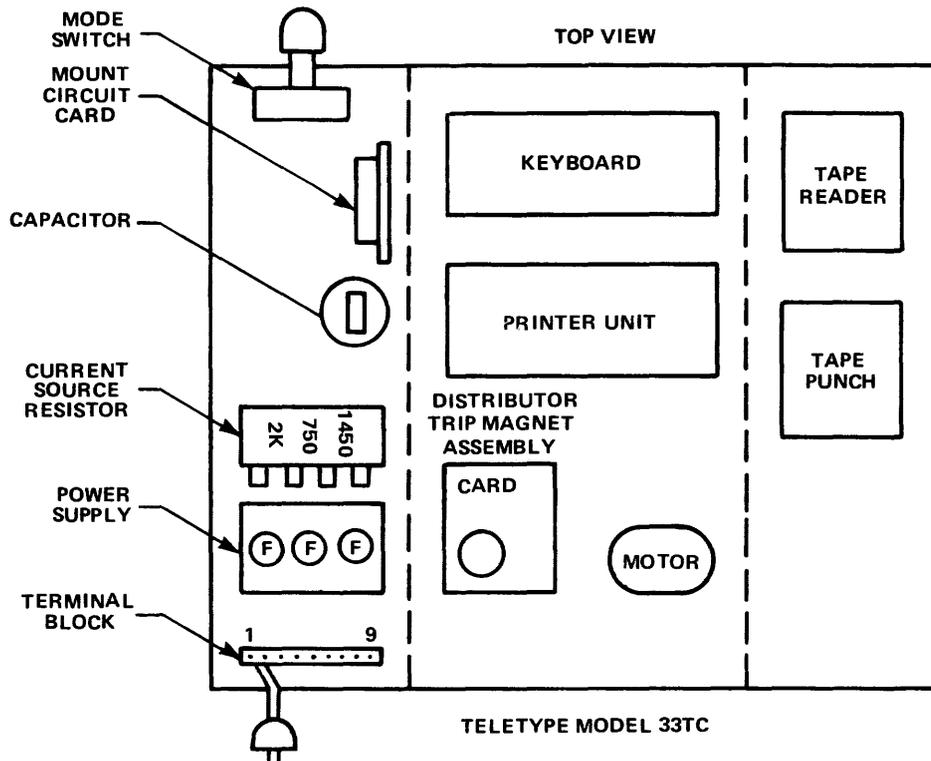


Figure A-1. Teletype Component Layout

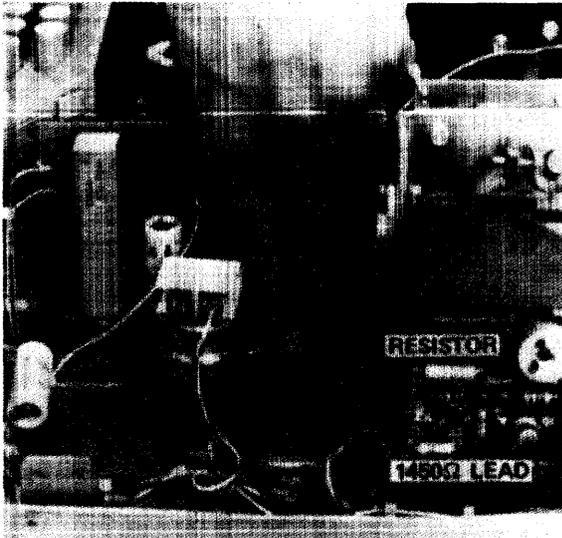


Figure A-2. Current Source Resistor



Figure A-3. Terminal Block

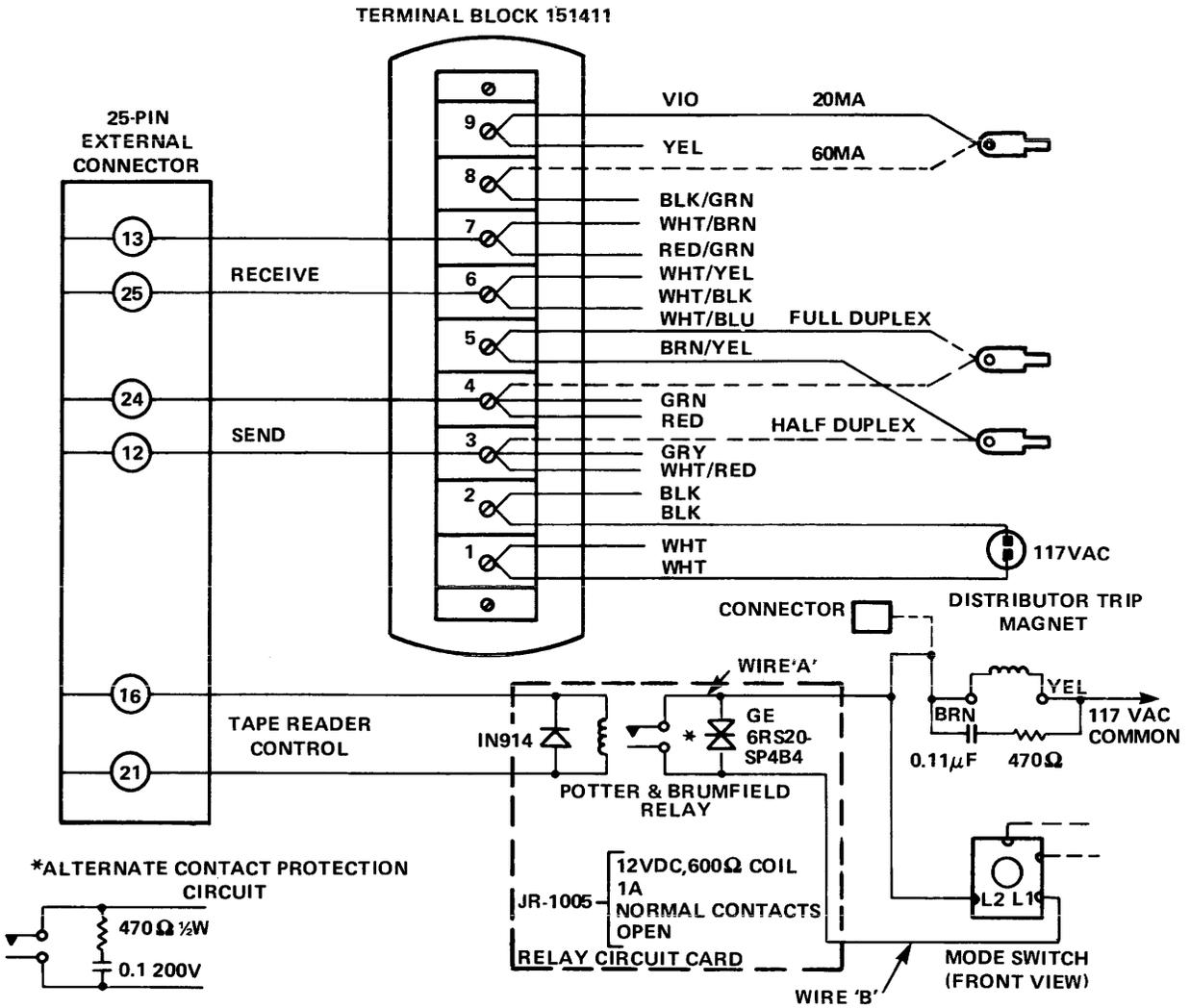


Figure A-4. Teletypewriter Modifications

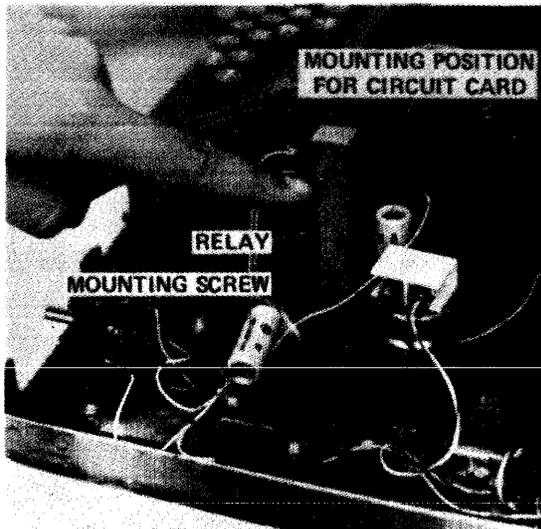


Figure A-5. Relay Circuit

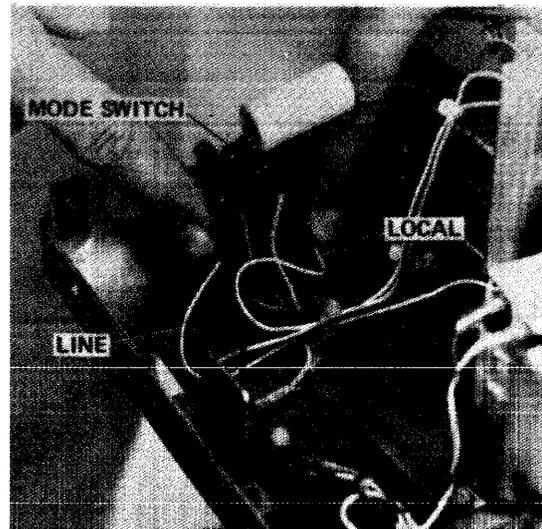


Figure A-6. Mode Switch

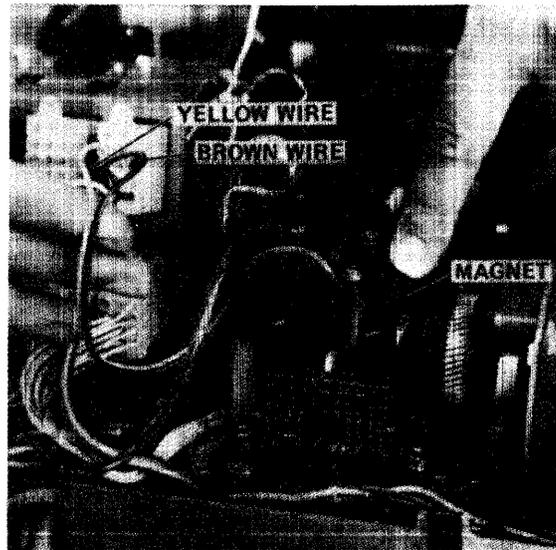


Figure A-7. Distributor Trip Magnet

---

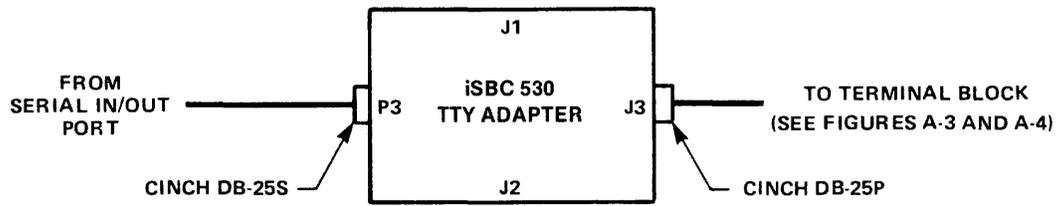


Figure A-8. TTY Adapter Cabling

---



## iSBC 86/12™ SINGLE BOARD COMPUTER

**8086 16 bit H-MOS microprocessor  
central processor unit**

**32K bytes of dual-port read/write memory  
with on-board refresh**

**Sockets for up to 16K-bytes of read only  
memory**

**System memory expandable to  
1 megabyte**

**24 programmable parallel I/O lines with  
sockets for interchangeable line drivers  
and terminators**

**Programmable synchronous/  
asynchronous RS232C compatible serial  
interface with software selectable baud  
rates**

**Two programmable 16-bit BCD or binary  
counters**

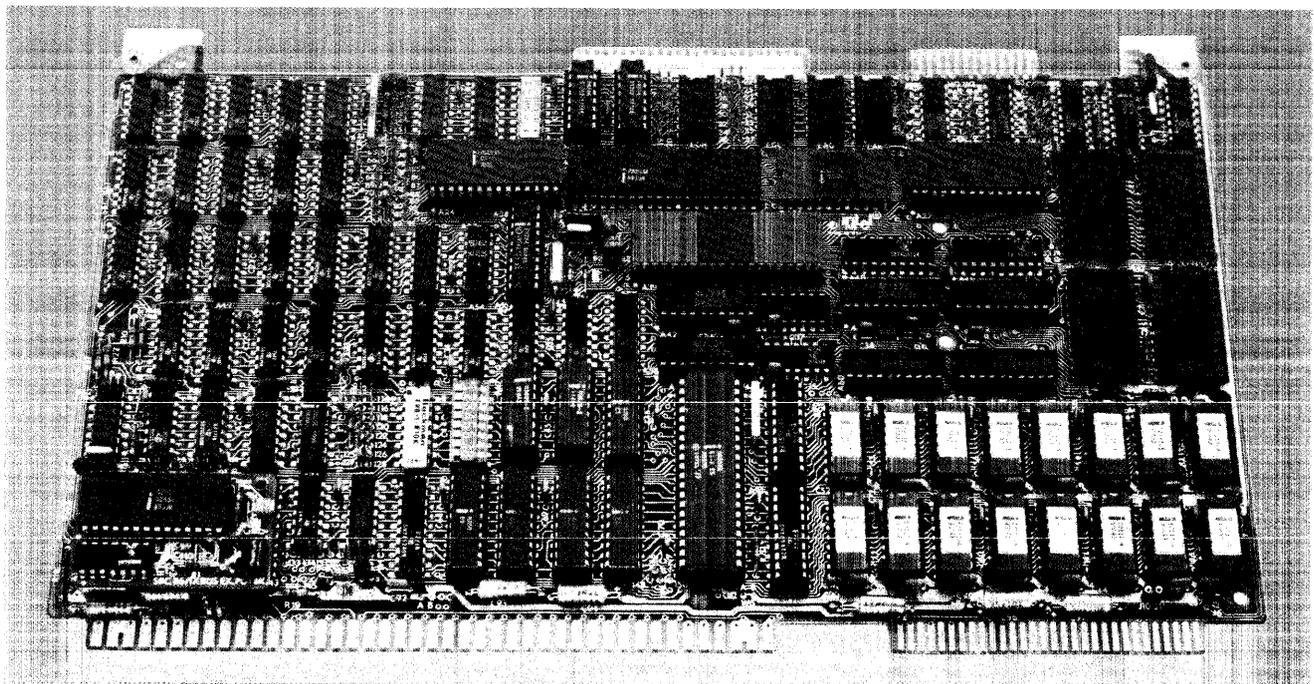
**9 levels of vectored interrupt control,  
expandable to 65 levels**

**Auxiliary power bus, memory protect and  
power fail interrupt control logic for  
read/write memory battery backup**

**MULTIBUS interface for multimaster  
configurations and system expansion**

**Compatible with iSBC 80 family single  
board computers, memory, digital and  
analog I/O, and peripheral controller  
boards**

The iSBC 86/12 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 86/12 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 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 86/12 is Intel's 8086, a powerful 16-bit H-MOS 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. The instruction set of the 8086 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 8086 processor.

**Architectural Features** — A 6-byte instruction queue provides pre-fetching of sequential instructions and can reduce the 1.2 $\mu$  sec minimum instruction cycle to 400 nsec by having the instruction already in the queue.

The stack oriented architecture facilitates nested sub-routines 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 86/12 has an internal bus for communicating with on-board memory and I/O options, a system bus (the MULTIBUS) 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. 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

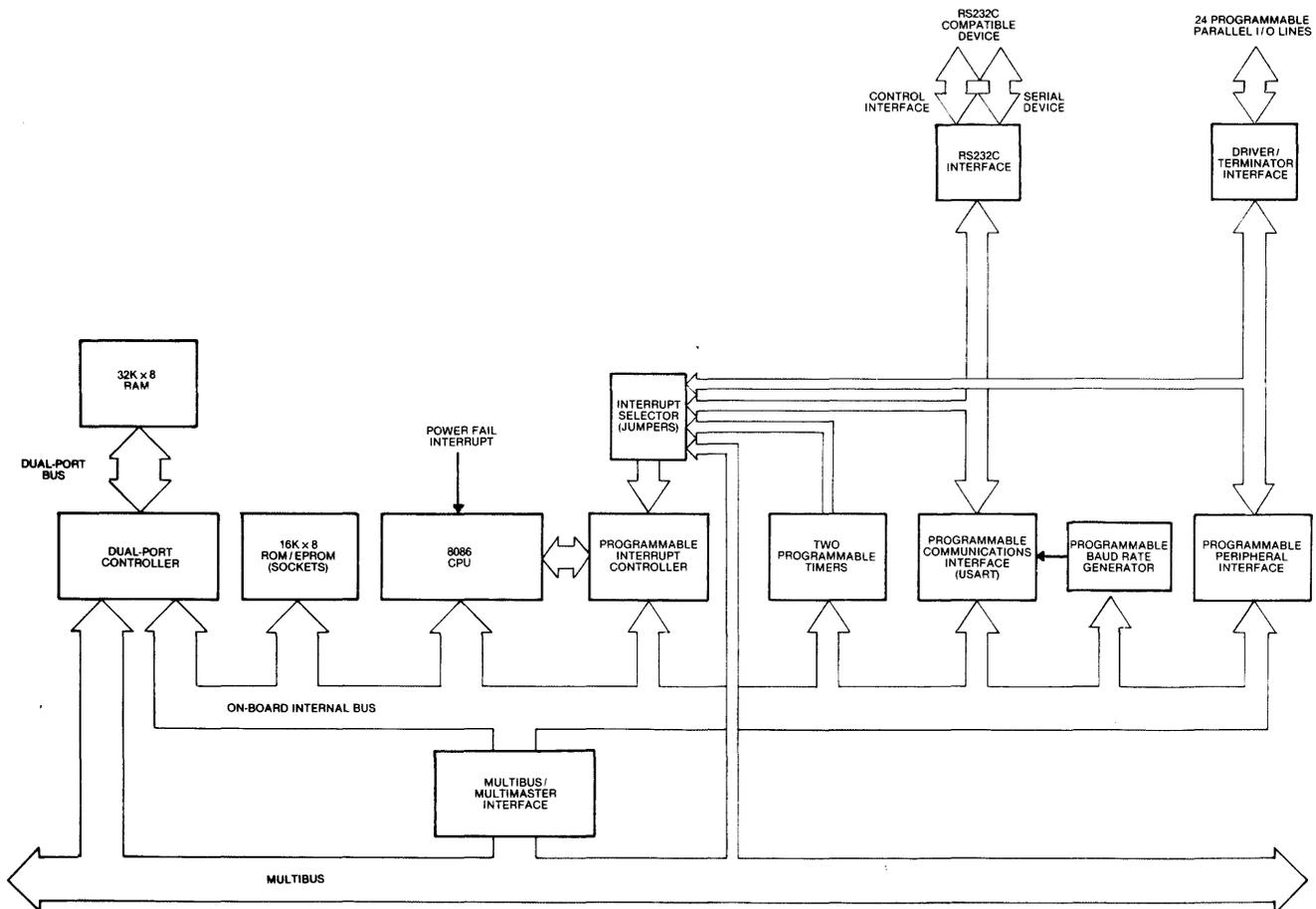


Figure 1. iSBC 86/12 Single Board Computer Block Design

expansion through the use of other 8- and 16-bit iSBC computers, memory and I/O expansion boards.

### RAM Capabilities

The iSBC 86/12 contains 32K-bytes of dynamic read/write memory using Intel® 2117 RAMs. 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 86/12 contains a dual port controller which allows access to the on-board RAM from the iSBC 86/12's CPU and from any other MULTIBUS master via the system bus. The dual port controller allows 8- and 16-bit accesses from the MULTIBUS 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 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. 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. In conjunction with this feature, the iSBC 86/12 has the ability to protect on-board memory from MULTIBUS access to any contiguous 8K-byte segments. These features allow multi-processor 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/ROM Capabilities

Four sockets are provided for up to 16K-bytes of non-volatile read only memory on the iSBC 86/12. ROM 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 Intel 2316E masked ROMs; or in 8K-byte increments up to 16K-

bytes by using Intel 2332 ROMs. On-board ROM is accessed via 16 bit data paths. System memory size is easily expanded by the addition of MULTIBUS compatible memory boards available in the iSBC 80/86 family.

### Parallel I/O Interface

The iSBC 86/12 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 86/12. 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 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

Port	Lines (qty)	Mode of Operation				Control	
		Unidirectional					Bidirectional
		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**

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.

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

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 86/12 to teletypewriters or other 20 mA current loop equipment.

### Programmable Timers

The iSBC 86/12 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

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.

Table 2. Programmable Timer Functions

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 86/12 RS232C USART serial port. In utilizing the iSBC 86/12, 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 and Multimaster Capabilities

The MULTIBUS 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 86/12 which can be used to generate an interrupt if an addressed device does not respond within 6 msec.

**Multimaster Capabilities** — The iSBC 86/12 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 86/12 provides full MULTIBUS arbitration control logic. This control logic allows up to three iSBC 86/12's 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 with the addition of an external priority network. The MULTIBUS arbitration logic operates synchronously with a MULTIBUS clock (provided by the iSBC 86/12 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 direct memory access (DMA) operations, and high speed peripheral control, but are by no means limited to these three.

## Interrupt Capability

The iSBC 86/12 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, to generate additional vector addresses, yielding a total of 65 unique interrupt levels.

**Interrupt Request Generation** — Interrupt requests may originate from 16 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 also be generated by each of the programmable timers. Eight additional

**\*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.

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.

**Table 3. Programmable Interrupt Modes**

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 1 Megabyte 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. 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 86/12 system. Consult the Intel OEM Microcomputer System Configuration Guide for specific data.

## System Development Capabilities

The development cycle of iSBC 86/12 products can be significantly reduced by using the floppy disk based Intellec® series microcomputer development systems. The Assembler, Locating Linker, Library Manager, Text Editor and system monitor are all supported by the ISIS-II disk based operating system. A minimum of 64K-bytes of RAM is needed in the Intellec system to support program development for the iSBC 86/12. To facilitate conversion of 8080A/8085A assembly language programs to run on the iSBC 86/12, CONV-86 is available under the ISIS-II operating system.

**Interface and Execution Package** — The iSBC 957 Interface and Execution Package allows the Intellec user to interface an iSBC 86/12 system to the development system. Included with the package are the necessary cables and software to allow transfer of files between the Intellec system and the iSBC 86/12. Additionally, the Intellec user can access a system monitor program (supplied on ROMs) resident in the iSBC 86/12 which allows access to programs loaded into the iSBC 86/12. The system monitor includes commands to examine and modify to memory, registers and I/O ports. Additionally, breakpoints, searches, and other useful operations are included to simplify software debug. Used in conjunction with the iSBC 957 Package, iSBC 86/12 execution packages are offered

which include additional hardware such as the iSBC 660 system chassis to mount and power the iSBC 86/12 for program development.

**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  $\mu$ sec  
 — 400 nsec (assumes instruction in the queue)

**Note:**

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

### Memory Addressing

**On Board ROM/EPROM** — FF000–FFFFFH (using 2758 EPROM's); FE000–FFFFFH (using 2316E ROM's or 2716 EPROM's); and FC000–FFFFFH (using 2332 ROM's).

**On-board RAM** — 32k bytes of dual port RAM. CPU Access: 00000–07FFFH. MULTIBUS Access is jumper selectable for any 8K-byte boundary, but not crossing a 128K byte boundary. Access for 8K, 16K, 24K, or 32K bytes may be selected for CPU use only.

### Memory Capacity

**On-Board Read Only Memory** — 16K bytes (sockets only)

**On-Board RAM** — 32K bytes

**Off-Board Expansion** — Up to 1 megabyte in user specified combinations of RAM, ROM, and EPROM.

**Note:**

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

### I/O Addressing

**On-Board Programmable I/O**

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

### I/O Capacity

**Parallel** — 24 programmable lines using one 8255A.

**Serial** — 1 programmable line using one 8251A.

## 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: 1CW2, 1CW3, 1CW4, 0CW1 (Mask Register)

Read: 0CW1 (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 inter-

rupt 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)

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

**Note:**

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

**Input Frequencies**

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

**Note:**

Above frequencies are user selectable.  
Event Rate: 2.46 MHz max

**Output Frequencies/Timing Intervals**

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

**Interfaces**

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

**System Clock (8086 CPU)**

5.00 MHz ± 0.1%

**Auxiliary Power**

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

**Connectors**

Interface	Pins (qty)	Centers (in.)	Mating Connectors
Bus	86	0.156	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

**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 86/12.

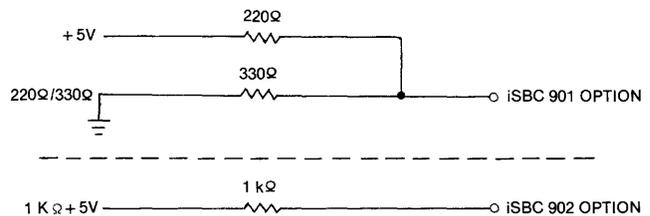
Driver	Characteristic	Sink Current (mA)
7438	I,OC	48
7437	I	48
7432	NI	16
7426	I,OC	16
7409	NI,OC	16
7408	NI	16
7403	I,OC	16
7400	I	16

**Note:**

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

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

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



**Bus Drivers**

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

**Physical Characteristics**

- Width** — 12.00 in. (30.48 cm)
- Height** — 6.75 in. (17.15 cm)
- Depth** — 0.70 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	450 mA	—	140 mA
With 8K ROM <sup>5</sup> (using 2316E)	6.1A	450 mA	—	140 mA
With 8K EPROM <sup>5</sup> (using 2716)	5.5A	450 mA	—	140 mA
With 16K ROM <sup>5</sup> (using 2332)	5.4A	450 mA	—	140 mA

**Notes:**

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

**Environmental Characteristics**

Operating Temperature — 0°C to 55°C

Relative Humidity — to 90% (without condensation)

**Reference Manual****9800645A** — iSBC 86/12 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 86/12	Single Board Computer with 32K bytes RAM

**Intel Corporation**  
3065 Bowers Avenue  
Santa Clara, California 95051  
Tel: (408) 987-8088\*\*  
TWX: 910-338-0026  
TELEX: 34-6372

**U.S.  
REGIONAL SALES OFFICES**

**WESTERN**

**CALIFORNIA**  
Intel Corp.\*  
1651 East 4th Street  
Suite 150  
Santa Ana 92701  
Tel: (714) 835-9642  
TWX: 910-595-1114

**MID-WESTERN**

**ILLINOIS**  
Intel Corp.\*  
900 Jorie Boulevard  
Suite 220  
Oakbrook 60521  
Tel: (312) 325-9510  
TWX: 910-651-5881

**EASTERN**

**OHIO**  
Intel Corp.\*  
8312 North Main Street  
Dayton 45415  
Tel: (513) 890-5350  
TELEX: 288-004

**ATLANTIC**

**MASSACHUSETTS**  
Intel Corp.\*  
187 Billerica Road, Suite 14A  
Chelmsford 01824  
Tel: (617) 256-6567  
TWX: 710-343-6333

\*\*Note New Telephone Number

**OVERSEAS  
MARKETING OFFICES**

**ORIENT**

**JAPAN**  
Intel Japan Corporation\*  
Flower Hill-Shinmachi East Bdg.  
1-23-9, Shinmachi, Setagaya-ku  
Tokyo 154  
Tel: (03) 426-9261  
TELEX: 781-28426

**EUROPE**

**BELGIUM**  
Intel International\*  
Rue du Moulin à Papier  
51-Boite 1  
B-1160 Brussels  
Tel: (02) 660 30 10  
TELEX: 24814

\*Field Application Location

### REQUEST FOR READER'S COMMENTS

The Microcomputer Division Technical Publications Department attempts to provide documents that meet the needs Intel product users. This form lets you participate directly in the documentation process.

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

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

---

---

---

---

---

---

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

---

---

---

---

---

---

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

---

---

---

---

---

---

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

---

---

---

---

---

---

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

NAME \_\_\_\_\_ DATE \_\_\_\_\_  
TITLE \_\_\_\_\_  
COMPANY NAME/DEPARTMENT \_\_\_\_\_  
ADDRESS \_\_\_\_\_  
CITY \_\_\_\_\_ STATE \_\_\_\_\_ ZIP CODE \_\_\_\_\_

Please check here if you require a written reply.





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

Printed in U.S.A.