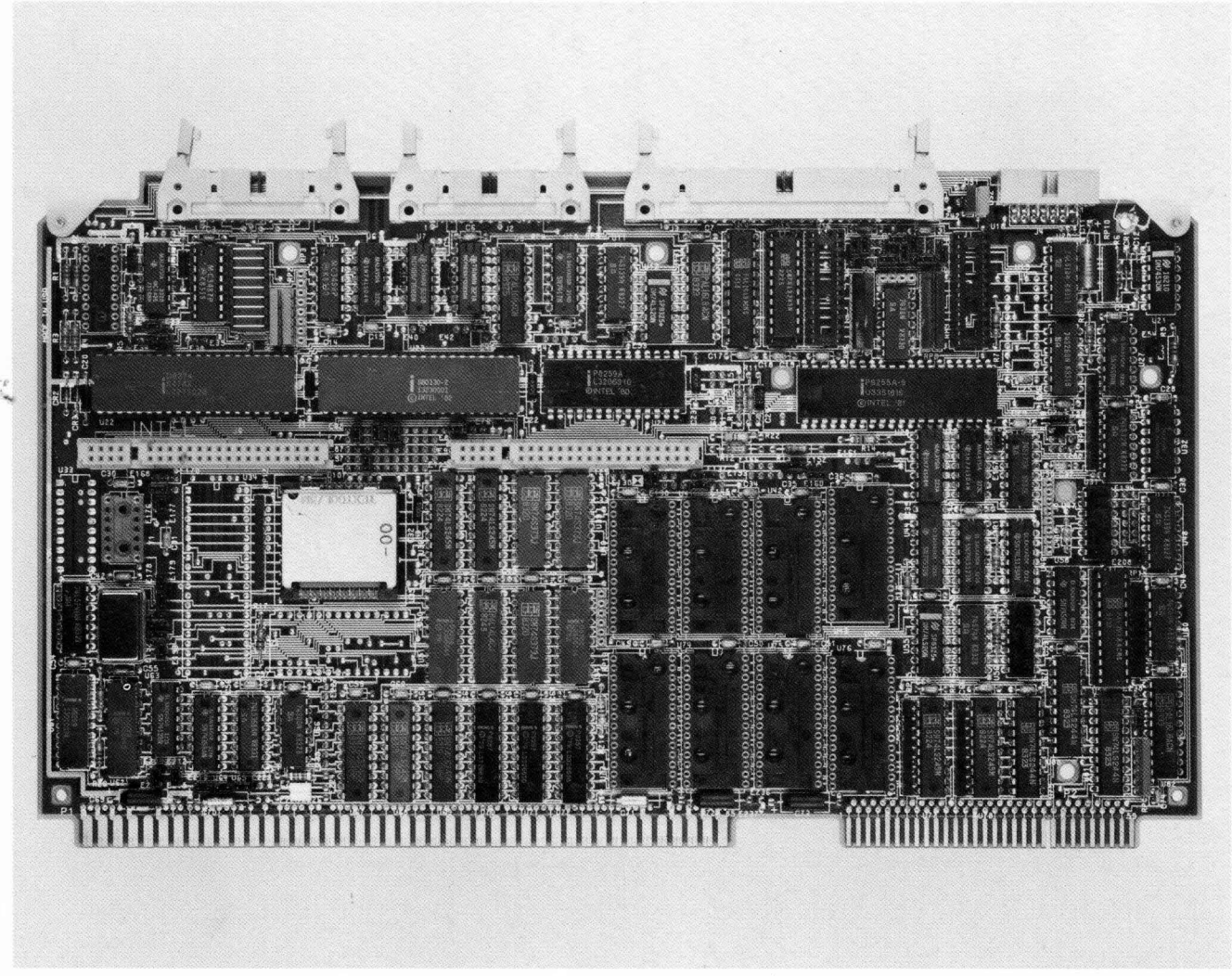
intel®

iSBC® 186/03
SINGLE BOARD COMPUTER
HARDWARE REFERENCE MANUAL



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

Order Number: 146414-001

iSBC® 186/03
SINGLE BOARD COMPUTER
HARDWARE REFERENCE MANUAL

Order Number: 146414-001

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

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

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

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

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

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

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

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

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

Intel retains the right to make changes to these specifications at any time, without notice.

Contact your local sales office to obtain the latest specifications before placing your order.

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

BITBUS	iLBX	iPDS	Plug-A-Bubble
COMMputer	i_m	iRMX	PROMPT
CREDIT	iMMX	iSBC	Promware
Data Pipeline	Insite	iSBX	QUEX
GENIUS	Int$_e$l	iSDM	QUEST
$\overset{\triangle}{i}$	int$_e$l	iSXM	Ripplemode
i	int$_e$lBOS	Library Manager	RMX/80
I^2ICE	Intelevision	MCS	RUPI
ICE	int$_e$ligent Identifier	Megachassis	Seamless
iCS	int$_e$ligent Programming	MICROMAINFRAME	SOLO
iDBP	Intellec	MULTIBUS	SYSTEM 2000
iDIS	Intellink	MULTICHANNEL	UPI
	iOSP	MULTIMODULE	

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

*MULTIBUS is a patented Intel bus.

REV.	REVISION HISTORY	DATE
-001	Original Issue.	

PREFACE

This manual describes the use of the iSBC 186/03 Single Board Computer. By reading this manual you will acquire sufficient knowledge of the board to enable you to successfully:

- Configure the jumpers and interfaces on the board to suit your application.
- Begin programming the board for operation in your system environment.

This manual references the information contained in other Intel technical publications, including:

- Introduction to the 80186 Microprocessor (AP-186), Order Number: 210973.
- Intel iAPX 186 Data Sheet, Order Number: 210451.
- Intel Microprocessor and Peripheral Handbook, Order Number: 210844.
- Intel Memory Components Handbook, Order Number: 210830.
- Intel iLBX™ Bus Specification, Order Number: 145695.
- Intel MULTIBUS® Specification, Order Number: 9800683.
- Intel iSBX™ Bus Specification, Order Number: 142686.
- EIA Standard for RS232C Interfacing, EIA-RS-232C.
- EIA Standard for RS422A Interfacing, EIA-RS-422A.
- EIA Standard for RS449 Interfacing, EIA-RS-449.
- Intel Application Note AP-134, Asynchronous Communications With The 8274 Multiple Protocol Serial Controller.
- Intel Application Note AP-145, Synchronous Communications With The 8274 Multiple Protocol Serial Controller.

This text uses two special characters, the asterisk (*) and the § symbol. Their interpretation is as follows:

* Used after a signal mnemonic to indicate that the signal is an active-low signal. A signal mnemonic without a trailing asterisk as an active-high signal. This notation replaces the use of the slash (/) that was previously used to indicate the active state of a signal.

§ Used after a jumper connection to indicate that the jumper is installed when the board is shipped from the factory.

CONTENTS

CONTENTS (continued)

CONTENTS (continued)

CONTENTS (continued)

TABLES

TABLES (continued)

FIGURES

FIGURES (continued)

***

CHAPTER 1. GENERAL INFORMATION

1.1 INTRODUCTION

The iSBC 186/03 Single Board Computer is a general purpose, 16-bit computer system on a MULTIBUS-compatible printed circuit board. The board supports the high-speed memory execution bus (iLBX bus) for local memory expansion, and the iSBX MULTIMODULE Bus for low-cost I/O expansion. The board is designed to interface to the Small Computer System Interface (SCSI) with some reconfiguration and optional components.

The purpose of this chapter is to introduce you to this state-of-the-art product in a quick and concise manner. This chapter gives a list of key features of the board, provides a brief description of the board, and lists the product specifications.

1.2 KEY FEATURES

This section provides a brief list of key features on the iSBC 186/03 board. More detailed descriptions of the features are located in the "Description" section of this chapter.

- iAPX 186 (80186) high-integration microprocessor.
- iAPX 86/30 (80130) operating system processor.
- Eight byte-wide memory sites for EPROM, EEPROM, SRAM, iRAM, or NVRAM. Configurable memory capacity: up to 32k bytes of RAM and up to 256k bytes of ROM memory on the board; or up to 512k bytes of ROM if on-board RAM is not used.
- iSBC 341 Memory Expansion MULTIMODULE board-compatible, expanding RAM to 64K-bytes.
- iLBX local memory expansion bus for high-speed transfers to/from iLBX memory expansion boards.
- 27 interrupt sources on-board using the 80186, 80130, and 8259A interrupt controllers, and the 8274 serial controller.
- Two serial I/O channels controlled by an 8274 multi-protocol serial controller (MPSC).
- General purpose parallel interface; can easily be reconfigured to the SCSI or Centronics interface by adding custom PALs (programmable array logic devices) as defined in Appendix C of this manual.
- Two iSBX bus interface connectors for low-cost I/O expansion.

- Master capability on the MULTIBUS interface.
- Dedicated front panel interface connector.

1.3 DESCRIPTION

This section provides more detail on each of the board features previously mentioned.

iAPX 186 (80186) Microprocessor

The iSBC 186/03 board is controlled by an Intel 80186 microprocessor operating at 6 MHz. The processor integrates two DMA channels, three timers and an interrupt controller on a single chip. The 80186 is object code compatible with existing 8086 and 8088 software; however, the 80186 instruction set has been expanded to include 10 new instructions. The built-in, on-chip functions are controlled by register programming, similar to other Intel peripheral components.

80130 Operating System Processor

The on-board 80130 component adds several functions to the iSBC 186/03 board: a subset of the iRMX 86 Operating System Nucleus in on-chip memory, three additional programmable timers, and an additional programmable interrupt controller. This subset of the iRMX 86 Nucleus provides operating system primitives for those applications performing functions such as real-time control. If the application requires more of the iRMX 86 Nucleus primitives than are provided in the 80130, they can be added by using the iRMX 86 Interactive Configuration Utility (ICU).

Intel offers run-time foundation software to support applications which range from general purpose to high-performance solutions. The iRMX 86 Nucleus provides an event-driven multitasking structure which includes task scheduling, task management, intertask communications and interrupt servicing for high-performance applications.

The entire iRMX 86 Operating System consists of the Nucleus and other pieces or "sub-systems". If the application tasks require features such as asychronous I/O control, priority-based resource allocation, file support for peripheral controllers, or a human interface/command line interpreter, such functions can be provided by one or more of the iRMX 86 Operating System subsystems. For execution, the iRMX 86 Operating System can be loaded into RAM or programmed into EPROM.

The iRMX 86 Operating System provides a rich set of features and options to support sophisticated applications solutions. In addition to supporting real-time requirements, the iRMX 86 Operating System has a powerful but easy-to-use human interface.

Services provided by the iRMX 86 Operating System include facilities for executing programs concurrently, sharing resources and information, servicing asynchronous events and interactively controlling system resources and utilities. The iRMX 86 Operating System can be readily extended to support assembler, PL/M, PASCAL, and FORTRAN software development environments. The modular building block software lends itself well to customized application solutions.

Local Memory

Up to 16 Megabytes of total system memory may be addressed by the iSBC 186/03 board, using a paging technique. Of this amount, a maximum of 256K-bytes of EPROM type local memory (using four 64K x 8 EPROM devices), and a maximum of 32K-bytes of RAM type local memory (using four 8K x 8 RAM devices) may reside on-board. If the RAM required for your application resides off-board, up to 512K-bytes of EPROM may reside on board (using eight 64K x 8 EPROM devices).

The on-board RAM can be increased to 64K-bytes by adding the optional iSBC 341 expansion board and four additional 8k x 8 devices. Local memory can operate at zero, one, or two wait-states depending on memory type.

The board will accept a wide variety of memory devices, including PROM, EPROM, EEPROM, SRAM, iRAM, and NVRAM devices. Either 24 or 28-pin devices may be used. Device types are specified by configuring two universal site jumper matrices.

iLBX™ Bus Local Memory Expansion

Local memory resource may be expanded via the Local Bus Extension (iLBX) interface which appears to the 80186 as local memory. The iSBC 186/03 board implements the iLBX bus as a Primary Master, using the optimized mode. Up to 896K-bytes of iLBX bus memory (RAM or PROM) can be added, with zero wait states (using an iSBC 428 board) or one wait-state (using an iSBC 012CX board) performance.

Interrupts

Interrupts may originate from numerous on-board or off-board sources. All interrupts, except the 80186 non-maskable interrupt (NMI), are handled by the 80130, configured as the master interrupt controller, and three other slave interrupt controllers. External system interrupts can be connected via the MULTIBUS lines to the 80130 or the 8259A. The 8259A, 8274, and the interrupt controller portion of the 80186 are configured as slaves to the 80130 device. An interrupt jumper matrix allows interrupt configuration flexibility and provides priority selection. Many internal system interrupts are handled by the 80186.

Serial I/O

Serial I/O operation is handled by an Intel 8274 Multi-Protocol Serial Communications (MPSC) device. This device supports two serial I/O channels. One channel may be configured for either RS232C or RS422/449 applications. The other channel handles only the RS232C application. The MPSC interface device transmits data at a rate that is programmable at 1, 1/16, 1/32, or 1/64 of the clock rate using the output from one of the on-board counters.

Parallel I/O

The iSBC 186/03 board uses one Intel 8255A-5 Programmable Peripheral Interface device to control three, 8-bit, parallel I/O ports. The parallel interface may be configured for a variety of dedicated or general purpose applications. By purchasing and programming Programmable Array Logic (PAL) devices as described later in this manual, you can implement a high-performance SCSI (Small Computer System Interface, ANSI-x379.2/82-2) Disk interface or a Centronics Type line printer interface, each using one DMA channel from the 80186 processor. The parallel port may also be used to implement many custom parallel interfaces by designing custom PALs to implement the unique handshake requirements of that interface.

iSBX™ Bus I/O Expansion

Two iSBX bus connectors (J6 & J7) are provided on the iSBC 186/03 board. These connectors are designed to expand the board's I/O functions, using iSBX MULTIMODULE boards, such as the iSBX 350 Parallel I/O MULTIMODULE Board. The MULTIMODULE boards reside directly on the iSBC 186/03 board. The iSBC 186/03 board supports either 8-bit or 16-bit iSBX boards. The iSBC 186/03 board accommodates two single-wide iSBX MULTIMODULE boards or one single-wide and one double-wide iSBX MULTIMODULE boards.

MULTIBUS® Interface

Off-board system access is provided by the MULTIBUS connector (P1) and a portion of the iLBX bus connector (P2). The iSBC 186/03 board is designed to operate as a full master in any Intel MULTIBUS compatible chassis or backplane. The board can also reside in a custom chassis.

1.4 DOCUMENTATION SUPPLIED

Each iSBC 186/03 board is shipped with a schematic diagram. Because the schematic diagram in this manual may not be the latest version, the drawings shipped with the board should be saved for future reference.

1.5 ADDITIONAL EQUIPMENT REQUIRED

The iSBC 186/03 board requires a few optional components for basic operation. Depending on your application, you may need to purchase mating connectors for the parallel I/O connector, serial I/O connectors, iLBX bus connector, and an auxiliary connector. Any on-board memory devices must also be purchased separately. Additional devices may be required for the parallel port interface as described in Appendix C and an optional oscillator may be required when using certain iSBX boards. Chapter 3 provides information for selecting these items.

1.6 COMPLIANCE LEVEL: 796 BUS SPECIFICATION (IEEE STANDARD)

All Intel MULTIBUS-compatible boards are designed around guidelines set forth in the 796 BUS SPECIFICATION (IEEE STANDARD - formerly the "Intel MULTIBUS Specification"). The standard requires that certain board operating characteristics, such as data bus width and memory addressing paths, be clearly stated in the board's printed specifications (i.e., reference manual). Used properly, this information quickly summarizes the level of compliance the board bears to the published 796 BUS SPECIFICATION. It clearly states the board's level of compatibility to the MULTIBUS structure. Refer to the 796 BUS SPECIFICATION or the INTEL MULTIBUS SPECIFICATION for additional information.

The following notation states the iSBC 186/03 board's level of compliance to the 796 BUS SPECIFICATION:

D16 M24 I16 V0 EL

This notation is decoded as follows:

D16 = Data path is 8 and/or 16 bits
M24 = Memory address path is up to 24 bits
I16 = I/O address path is 8 bits or 16 bits
V0 = Non-Bus-Vectored interrupts are supported; and
EL = Level-triggered and edge-triggered interrupts are supported.

As shipped, the BCLK* and CCLK* frequency on the iSBC 186/03 board is 6 MHz whereas the CCLK* specification is 10 MHz. A socket is provided so that you can purchase and install a 10 MHz oscillator, if desired. See Chapter 3 for additional information.

1.7 COMPLIANCE LEVEL: INTEL iSBX™ BUS SPECIFICATION

All Intel iSBX bus-compatible boards are designed around guidelines set forth in the Intel iSBX BUS SPECIFICATION. The standard requires that certain board operating characteristics, such as data bus width and employment of interlocked operation, be clearly stated in the board's printed specifications. Used properly, this information quickly summarizes the level of compliance the board bears to the published iSBX BUS SPECIFICATION.

It clearly states the board's level of compatibility to the iSBX bus structure. Refer to the iSBX BUS SPECIFICATION for additional information.

The following notation states the iSBC 186/03 board's level of compliance to the iSBX BUS SPECIFICATION:

D16/16 DMA

This notation is decoded as follows:

- The DMA indicates that the board supports DMA operations to an iSBX bus connector.
- The D16/16 means that the 16-bit CPU board can interface to either an 8-bit or a 16-bit expansion module.

1.8 COMPLIANCE LEVEL: INTEL iLBX™ BUS SPECIFICATION

All Intel iLBX bus-compatible boards are designed around guidelines set forth in the Intel iLBX BUS SPECIFICATION. The standard requires that certain board operating characteristics, such as data bus width, address path width, and other characteristics, be clearly stated in the board's printed specifications (i.e., reference manual). Used properly, this information quickly summarizes the level of compliance the board bears to the published iLBX BUS SPECIFICATION. It clearly states the board's level of compatibility to the iLBX bus structure. Refer to the iLBX BUS SPECIFICATION for additional information.

The following notation states the iSBC 186/03 board's level of compliance to the iLBX BUS SPECIFICATION:

PM D16

This notation is decoded as follows:

PM	Primary Master device
D16	A 16-bit data path

For additional iLBX bus timing requirements and specifications, refer to the iLBX bus section in Chapter 2.

1.9 SPECIFICATIONS

Specifications of the iSBC 186/03 board are provided in Table 1-1.

Table 1-1. iSBC® 186/03 Board Specifications

CPU	Intel iAPX 186 CPU (referred to as the 80186)
WORD SIZE	
Instruction:	8, 16, 24, or 32 bits
Data:	8 to 16 bits
Address:	20 bits - 1M byte address mode 24 bits - 16M byte Address mode
SYSTEM CLOCK SPEED	6.0 MHz, 167 nanoseconds period
BASIC INSTRUCTION CYCLE TIME	1 microsecond 333 nanoseconds (assumes instruction in the queue). Note: Basic instruction cycle is defined as the fastest instruction time (i.e., two clock cycles plus instruction fetch). Zero wait-state memory is assumed.
LOCAL MEMORY ARRAY	
On-board Memory:	Eight 28-pin JEDEC-compatible, byte-wide sockets hold user-provided memory devices in 2k x 8, 8k x 8, 16k x 8, 32k x 8, or 64k x 8 capacity. Four sockets are dedicated to EPROM or EEPROM memory; four sockets may be used for EPROM, EEPROM, NVRAM or RAM memory.
Maximum EPROM Memory:	256K-bytes (maximum in four sockets); or 512K-bytes in eight sockets if on-board RAM is not required.
Maximum RAM Memory:	32k bytes (maximum in 4 sockets). Optional iSBC 341 Memory Expansion MULTIMODULE board increases RAM capacity to 64K-bytes.
Maximum System Memory:	16 Megabytes
MEMORY ACCESS TIMES	Refer to Appendix D
DEFAULT MEMORY ADDRESSING	Hexidecimal notation
On-Board RAM On-Board ROM	00000 - 01FFF (four 2K x 8 RAMS) F8000 - FFFFF (four 8K x 8 PROMS)

Table 1-1. iSBC® 186/03 Board Specifications (continued)

ON-BOARD I/O ADDRESSING	Hexidecimal notation
iAPX 186 Internal Register Base Address	FF00
iSBX Connector J7 (8-bit board)	80 - 9E Even Bytes or Word
iSBX Connector J7 (16-bit board)	80 - 8E Even Bytes or Word 81 - 8F Odd Bytes Only
iSBX Connector J6 (8-bit board)	A0 - BE Even Bytes or Word
iSBX Connector J6 (16-bit board)	A0 - AE Even Bytes or Word A1 - AF Odd Bytes Only
SBX Opt MDACK	D2 (Byte or word)
SBX 1 MDACK	D4 (Byte or word)
SBX 2 MDACK	D6 (Byte or word)
Slave Interrupt Controller (8259)	C0 ICW1, OCW2, OCW3, Status, & Poll (Bytes only) C2 ICW2, ICW3, ICW4, OCW1 & Masks (Bytes only)
Parallel Interface (8255)	C8 PPI Port A (Bytes) CA PPI Port B (Bytes) CC PPI Port C (Bytes) CE PPI Control (Bytes)
16MB Window Latch	D0 (Byte, write only)
Serial Interface (8274)	
CH A Data	D8 (Bytes)
CH B Data	DA (Bytes)
CH A Control/Status	DC (Bytes)
CH B Control/Status	DE (Bytes)
80130 I/O Addresses	
Master Interrupt Controller	E0 - E2 (Byte access only)
System Timer	E8 (Byte access only)
Delay Timer	EA (Byte access only)
Baud Rate Timer	EC (Byte access only)
Timer Control	EE (Byte access only)

Table 1-1. iSBC® 186/03 Board Specifications (continued)

Special DMA Registers	
DRQ MUX PAL Sel 0	F0 - F1 (Byte access only)
DRQ MUX PAL Sel 1	F2 - F3 (Byte access only)
DRQ MUX PAL Sel 2	F4 - F5 (Byte access only)
DRQ MUX PAL TDMA	F6 (Byte access only)
Reserved	F7 - FF (Byte access only)
INTERFACES	
MULTIBUS	All signals TTL compatible
Parallel I/O	All signals TTL compatible
Interrupt Requests	All signals TTL compatible
iSBX bus	All signals TTL compatible
Serial I/O	RS 232C compatible, DCE RS422A/449 DCE, DTE
I/O CAPABILITY	
Parallel:	24 programmable I/O lines using one 8255A PPI device; reconfigurable to the new SCSI interface or the Centronics type line printer interface.
Serial:	Two serial interfaces using the 8274 MPSC device. Connector J1 is configurable for either RS232C or RS422A/449 operation, Connector J2 is configured for only RS232C operation.
Expansion:	Two 16-bit iSBX bus connectors (J6 and J7) providing expansion via addition of 8-bit or 16-bit iSBX MULTIMODULE boards, in increments as follows: • one single-wide MULTIMODULE board, or • two single-wide MULTIMODULE boards, or • one double-wide MULTIMODULE board, or • one single- and one double-wide MULTIMODULE board
SERIAL COMMUNICATIONS CHARACTERISTICS	
Protocols:	Bit-synchronous Byte-synchronous Asynchronous

Table 1-1. iSBC® 186/03 Board Specifications (continued)

Synchronous: 5 to 8 bit characters; internal character synchronization; automatic sync bit insertion.

Asynchronous: 5 to 8 bit characters; 1, 1 1/2, or 2 stop bits; false start-bit detection.

8274 Baud Rates:
(Reference 6MHz)
(80130 Clock)

Synchronous X1	Asynchronous X1	Asynchronous X16	Asynchronous X32	Asynchronous X64
750K	750K	---	---	---
600K	600K	---	19.2K	9600
300K	300K	19.2K	9600	4800
153.8K	153.8K	9600	4800	2400
76.9K	76.9K	4800	2400	1200
38.4K	38.4K	2400	1200	600
19.2K	19.2K	1200	600	300
	9600	600	300	150
	4800	300	150	---
	2400	150	75	---
	1200	75	---	---
	600	---	---	---

PHYSICAL CHARACTERISTICS

Width	12.00 in. (30.48 cm)
Length	7.05 in. (17.91 cm)
Thickness	0.50 in. (1.27 cm)
Weight	13.0 oz. (369 grams)

ENVIRONMENTAL CHARACTERISTICS

Operating:	
Temperature	32°F to 131°F (0°C to 55°C)
Air Flow	200 linear ft/min. air velocity (MINIMUM)
Humidity:	0 to 95% non-condensing
Non-Operating:	
Temperature:	-40° C to +75° C
Humidity:	0 to 95% non-condensing

Table 1-1. iSBC® 186/03 Board Specifications (continued)

Power Requirements	57.5 Watts
Maximum Heat Dissipation	818.5 gcal/minute (3.31 Btu/minute)

ELECTRICAL CHARACTERISTICS

DC Power Requirements:

Supply Voltage	Current Required (max.)
+5V ±5% (note 2)	5.0 A
+12V ±5% (note 1)	40 mA
-12V ±5% (note 1)	40 mA

Notes:
1. ±12 volts is required for RS232C interface.
2. Values are for iSBC 186/03 board without memory components or a MULTIMODULE.

***

CHAPTER 2. BOARD OPERATION AND CONFIGURATION

2.1 INTRODUCTION

In order for you to successfully use the iSBC 186/03 Single Board Computer, you must understand how to configure each of the functions on your board. This chapter describes the jumper configuration of each function and any relevant interfacing information.

The functional topics covered in this chapter are presented in this order:

- CPU and Support Circuitry Configuration
- Memory Configuration
- Direct Memory Access (DMA)
- iLBX Bus Interface P2
- Interrupts
- Timers
- Parallel I/O Interface J3
- Serial I/O Interfaces J1 and J2
- iSBX Bus Interfaces J6 and J7
- MULTIBUS Interface P1
- Front Panel Interface J4

Appendix A of this manual provides a jumper location diagram and two jumper lists: a numerical listing of all jumpers on the iSBC 186/03 board, and a list of the default jumper configuration of the board.

Figure 2-1 shows a block diagram of the various functions on the board. The following paragraphs describe the configurations available for each functional area.

2.2 CPU AND SUPPORT CIRCUITRY CONFIGURATIONS

The iSBC 186/03 board provides several configuration options used to configure the operation of the CPU and its support circuits. Those options are as follows:

- Wait-state selection
- CPU timer selection
- CPU DMA operation
- CPU Slave Interrupt Controller
- CPU Clock Speed

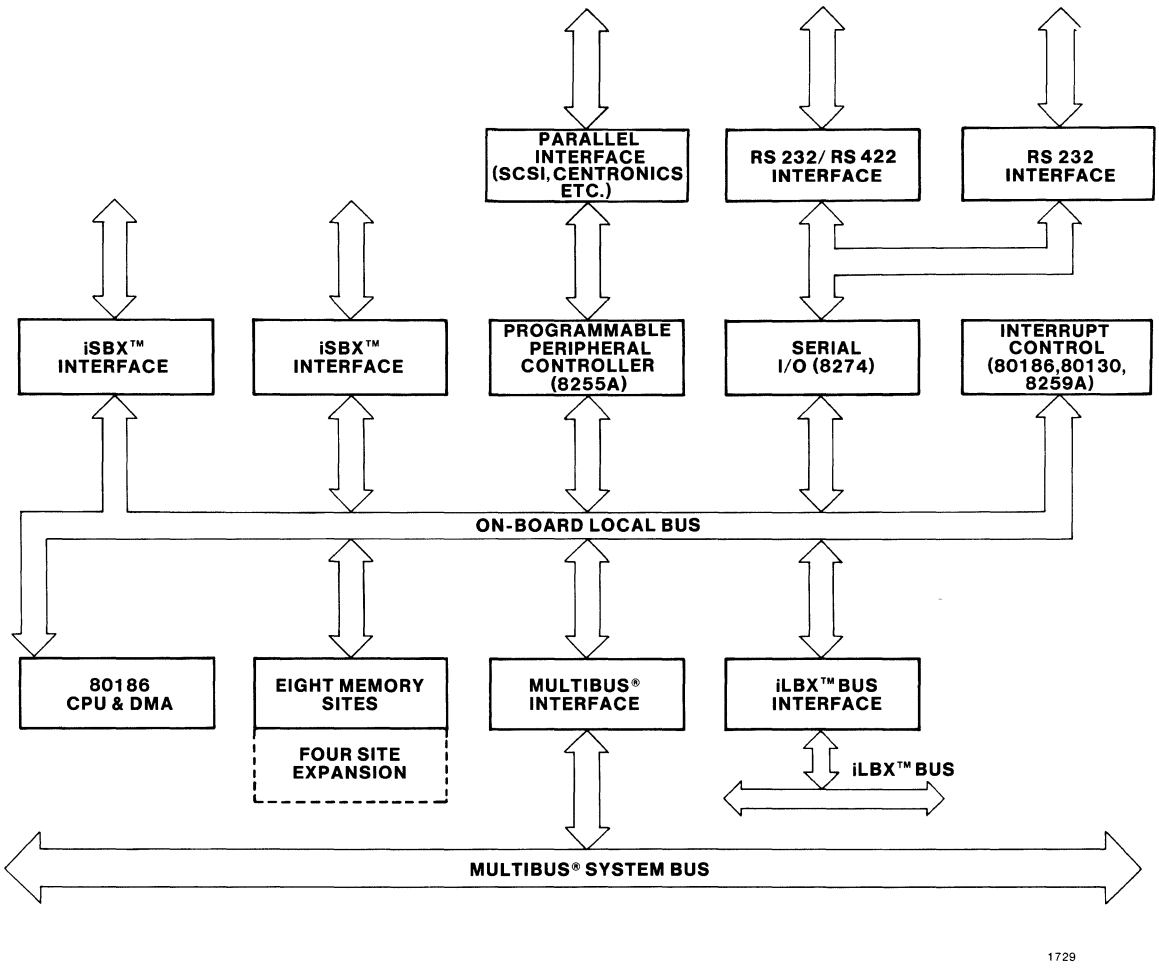


Figure 2-1. Block Diagram

2.2.1 WAIT-STATES FOR ON-BOARD RESOURCES

The iSBC 186/03 board generates wait states to extend CPU operation until the current on-board access (either memory or I/O) is complete. The iSBC 186/03 board allows you to individually configure the number of wait-states (zero, one, or two) for each memory bank (A and B). Refer to Appendix D for timing requirements for wait states of various memory types.

You cannot configure the wait-states for on-board I/O accesses on the iSBC 186/03 board; an on-board I/O operation requires a minimum of one wait-state. However, iSBX bus I/O wait-states can be extended with the iSBX bus MWAIT* signal.

Normally, an iLBX bus access requires a minimum of one wait state, provided the iLBX bus timing response meets the specification outlined in Table 2-9. A special mode allows the iLBX bus interface to run at zero wait-states if an iSBC 428 memory board is used with static RAM.

2.3 MEMORY CONFIGURATION

This section provides the information needed to partition the memory resources addressable by the iSBC 186/03 board. One of the first tasks is to determine the memory requirements needed for your particular application. You need to consider the amount, speed, type, and size of the memory devices required.

The iSBC 186/03 board provides eight 28-pin, byte-wide sockets into which various memory devices can be installed. These eight sockets on the iSBC 186/03 board are partitioned into two banks: Bank A and Bank B. A jumper configurator block (one for each bank) defines the type and size of the memory device installed. Each bank (four sockets) may be configured for the device type and size required by installing the configurator block push-on jumpers. These configuration blocks allow all sockets to be compatible with ROM, PROM, EPROM, SRAM, EEPROM, or NVRAM devices (see the "Allowable Memory Device Types" section, following), and the four Bank A sites are also compatible with iRAMs. However, there are restrictions on which devices can be mixed in a bank and which devices are supported by the decode logic.

The following paragraphs provide configuration information for the memory space on the iSBC 186/03 board. The information is presented in the following sequence:

- MEMORY MAP (DEFAULT CONFIGURATION)
- CONFIGURATION OVERVIEW
- 16M BYTE MODE
- BANK A MEMORY CONFIGURATION
- BANK B MEMORY CONFIGURATION
- 80130 MEMORY CONFIGURATION
- Battery Backup
- Using NVRAMs

2.3.1 MEMORY MAP (DEFAULT CONFIGURATION)

As shipped, the iSBC 186/03 board allows installation of four 2K x 8 SRAM devices in sockets U42/U75 and U43/U76 and installation of four 8K x 8 EPROM devices in sockets U40/U73 and U41/U74. AC and DC memory requirements are listed in Appendix D. The default address range assigned to the 8K-bytes of static random access memory (SRAM) devices is from 00000H to 01FFFH. The default address range assigned to the 32K-bytes of Read Only Memory (ROM) devices is from F8000H up to FFFFFH. (Note that when only two EPROMS are used, they must are installed in sockets U41/U74).

In the default configuration the iSBC 186/03 board assigns the iLBX address range immediately above the Bank A address space from 02000H to 7FFFFH and the MULTIBUS System Bus address resource beginning at 80000H and going up to the bottom of PROM memory. Figure 2-2 shows the default iSBC 186/03 memory map.

Bank A is an on-board memory resource and is typically assigned to the lower portion of memory for RAM devices. Bank B is an on-board memory resource and is assigned to the upper portion of memory for program storage. The 16k bytes of 80130 memory is another on-board resource and is located just below Bank B (PROM) memory, but is disabled in the as-shipped configuration. The MULTIBUS memory is an off-board resource and involves bus arbitration before access is permitted. The iLBX bus memory is an off-board resource but appears to the CPU as local on-board memory.

Default Address Range

Bank A (RAM) Address Space (U42/U75)	000000H-000FFFH
Bank A (RAM) Address Space (U43/U76)	001000H-001FFFH
Bank B (PROM) Memory Address Space At U40/U73:	0F8000H-0FBFFFH
Bank B (PROM) Memory Address Space At U41/U74:	0FC000H-0FFFFFH
iLBX Bus Interface Address Space:	002000H-07FFFFH**

** Bank A RAM address space overlays the iLBX bus address space so that iLBX bus memory starts immediately above the Bank A address space.

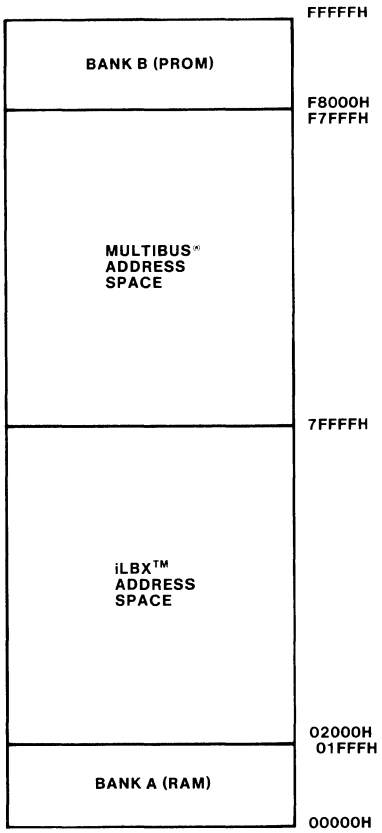


Figure 2-2. Default Memory Map

2.3.2 OVERVIEW OF THE MEMORY CONFIGURATION OPTIONS

The iSBC 186/03 board provides you with configuration control features for the memory space on the board. You can configure the Banks (Bank A and Bank B) as two independent memory address spaces depending on how you configure the jumper matrices for the sockets. Your configuration of the iLBX memory depends on the operation of the iLBX bus memory boards.

The similarities between the Bank A and Bank B memory configuration options are:

1. A standard jumper matrix format for selecting a device type.
2. 28-pin JEDEC sockets for the 24 or 28-pin memory devices.
3. Wait-state selection.

The major differences between the Bank A and Bank B memory configuration options are:

1. Bank A supports iRAM, SRAM, EPROM, EEPROM, or NVRAM devices.
2. One iSBC 341 expansion board for Bank A; none for Bank B.
3. Bank B supports ROM, PROM, EPROM, or EEPROM devices only.
4. Bank B must be assigned to upper memory.
5. Bank A could be assigned to either lower (typical) or upper memory residing below the assigned memory space for Bank B.

2.3.3 16M-BYTE MODE

The iSBC 186/03 board can be configured for either 1 Mbyte or 16 Mbyte address space. The 80186 device on the iSBC 186/03 board is only capable of addressing 1 Mbyte. However, a paging technique is used to access up to 16 Mbytes (using a latch at I/O address 00D0H). To enable the 16 Mbyte addressing, you must install a jumper between E40 and E41. The 16 Mbyte addressing is then accomplished by writing the upper six bits of the 16 Mbyte address to the latch at I/O address 00D0H. The 16 Mbyte address space is then accessed by issuing the corresponding address (excluding the upper two bits in the 1 Mbyte address) within the third 256 Kbyte page of the 80186 address space (80000H through BFFFFH). The iSBC 186/03 board automatically places 24 bits of the address onto the P1 and the P2 connectors. If no other board provides termination for the upper four address lines, then a 2.2K ohm resistor must be installed in socket RP13 on the iSBC 186/03 board. Refer to Chapter 4 for more details on programming the 16M byte address mode.

2.3.4 LOCAL MEMORY CONFIGURATION

This section of the manual describes options that are available to you in configuring the local and iLBX bus memory resources on the iSBC 186/03 board. You need an understanding of this information if you do not wish to use the memory on the board in the as-shipped configuration.

The following paragraphs present information on the local memory configuration in this order:

- Description of the sockets
- Allowable device type mixtures
- Overview of the configuration sequence
- Details of the configuration sequence

2.3.4.1 Local Memory Socket Pair Descriptions

The design of the iSBC 186/03 board is such that it allows you to partition the local memory into two pieces and configure each piece independently.

Figure 2-3 shows the partitioning of the Bank B memory sockets and their relative positions as they appear on the iSBC 186/03 board. Sockets U40 and U73 are a socket set, referred to as pair U40/U73; sockets U41 and U74 are a set, referred to as pair U41/U72.

Bank B memory socket pairs U40/U73 and U41/U74 are always located at upper memory. These sockets are intended for EPROM or EEPROM that contains the program for a power-up sequence, such as the Intel iSDM 86 Monitor.

Bank A memory socket pairs U42/U75 and U43/U76, on the other hand, are moveable within the memory map; you can either assign the memory to the lower portion of memory starting at address 0H, or you can locate it immediately below socket pairs U40/U73 and U41/U74 to provide more EPROM space.

2.3.4.2 Allowable Memory Device Types (Bank A and Bank B)

An array of stake pins is provided for each bank of four sockets to configure the banks for different device types. In the PROM bank (Bank B), all sockets must be populated with the PROM/EPROM/EEPROM devices for program storage. In the RAM bank (Bank A), each pair of sockets must be populated with the same devices, and devices may be mixed between pairs of iRAM, SRAM, EEPROM, and NVRAM devices as represented by an "X" in Table 2-1 (See note below when using EEPROM or NVRAM devices). All address ranges for each socket pair in the PROM bank are the same size, and all address ranges for each socket pair in the RAM bank are the same size.

Table 2-1. Allowable Mixtures In Bank A (RAM Bank)

	SRAM 4K, 8K, 16K	SRAM 32K	iRAM 8K, 16K	EEPROM+ 2K, 4K 8K, 16K	NVRAM 1/2K, 1K, 2K 4K, 8K
SRAM 4K, 8K, 16K	---	X	X	X	X
SRAM 32K	X	---	None	None	None
iRAM 8K, 16K	X	None	---	None++	None
EEPROM+ 2K, 4K, 8K, 16K	X	None	None++	---	None
NVRAM 1/2K, 1K, 2K, 4K, 8K	X	None	None	None	---

Notes: All device sizes are X 8 (byte-wide).
+ = "SMART" 2817A type devices only.
++ = Timing constraint does not allow EEPROM to be mixed with iRAM.
Mixtures are not allowed in Bank B (PROM Bank).

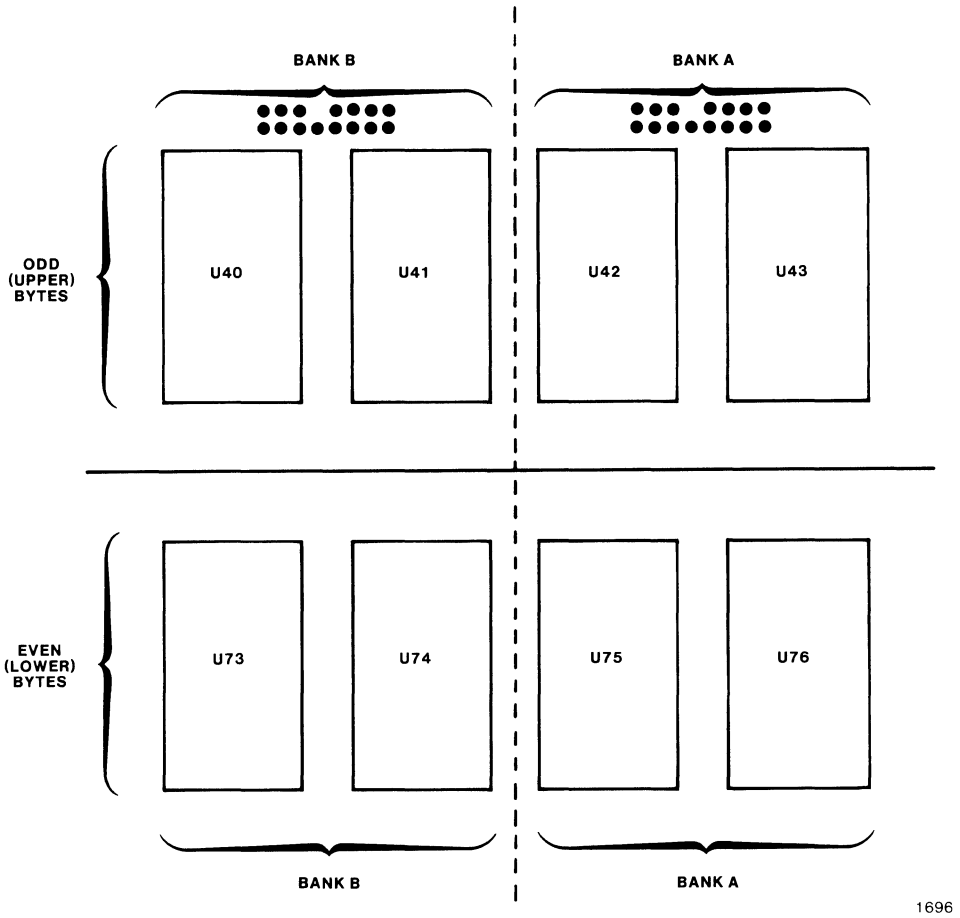


Figure 2-3. Local Memory Socket Pairs and Jumper Matrices

2.3.4.3 Memory Configuration Sequence Overview

You can logically step through the entire local memory configuration sequence by performing a series of 4 operations, as follows:

1. Select a memory device type for each Bank. You configure the board for operation with a specific type of memory device by installing or removing jumpers at jumper matrix E131 through E145 for Bank B socket pairs U40/U73 and U41/U74, and E146 through E160 for Bank A socket pairs U42/U75 and U43/U76. Jumper E238-E239 must be installed when using iRAMs in Bank A. The iRAM devices are not allowed in Bank B.

2. Select the memory address and memory size for Bank B memory. This is a combination selection that is coordinated by PAL U59, the memory decode PAL. Make the selection by configuring jumper inputs E194 through E197 for the PAL. This selects one of four memory map configurations (see Step 2 below, and Table 2-2) for all of the Bank B memory. If one of these four options does not provide a suitable solution, custom-program a replacement PAL for socket U59 to create your own solution. The source equation for the factory installed PAL is provided in Appendix G.

3. Select a memory size and either top or bottom justify Bank A memory. This is another combination selection coordinated by PAL U59. Make the selection by configuring jumper inputs E198 through E203 for the PAL devices (see Step 3 below and Table 2-3). This changes the size of the memory space and determines whether Bank A is bottom-justified (starting at 0H) or top-justified (starting immediately below Bank B). The decode options support 2K x 8, 8K x 8, and 32K x 8 RAMs. If one of these options does not provide a suitable application, a custom PAL may be programmed for socket U59.

4. Select the number of wait-states required for each socket pair. You select the number of CPU wait-states by configuring jumper E123 through E128. The jumpers provide options from zero to two wait-states at each Bank, depending on the requirements of your memory devices. Refer to Step 4 below.

5. Enable or Disable the 80130 memory. To enable the 16K-bytes of memory contained in the 80130, install jumper E204-E205. Refer to table 2-2 and table 2-3 for address ranges. To disable the 80130 memory, remove the jumper.

Subsequent sections provide the details required for each step of the configuration sequence.

STEP 1 - Selecting a Memory Device Type For Each Bank

The jumper matrix for each bank allows all the sockets to be compatible with ROM, PROM, EPROM, EEPROM, static RAM (SRAM), non-volatile RAM (NVRAM), or Intel iRAM devices. However, because the address decoding logic requires Bank B to be used for program storage only, Bank B can only be used for ROM, PROM, EPROM, or EEPROM, unless the decode PROM is actually removed and modified or replaced.

The jumper matrix consists of 15 stake pins arranged in two rows. The pin arrangement is a standard format, as shown in Figure 2-4; the missing pin serves as a key to the orientation of the matrix. Figure 2-4 also lists the signals that are assigned to each pin of the matrix.

There are two of these jumper matrices on the iSBC 186/03 board; one for the Bank A memory socket pairs and one for the Bank B memory socket pairs. Configure each matrix independently for the type of memory device installed.

You can mix types within Bank A, providing you use the guidelines set forth in Table 2-1. Mixtures are not allowed in Bank B.

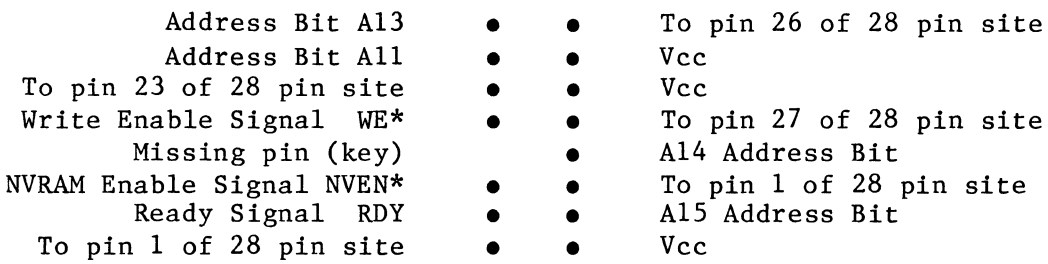


Figure 2-4. Memory Size Configuration Matrix

Jumper Matrix Configuration

Five basic types of memory devices can be installed in the local memory sockets on-board; refer to Figures E-1 through E-6 of Appendix E. Within each type category, there are one or more sizes of device; each is assigned a configuration number. Figures E-1 through E-6 in Appendix E show diagrams of the jumper matrix and the jumpers required to select each memory device (for that socket pair).

The seventeen options for each of the local memory socket pairs are as follows:

Matrix Conf. Numbers	Type Of Memory Device In The Socket Pair	Reference
0 through 3	for applications using Static RAM	Figure E-1
4 through 5	for applications using iRAM devices	Figure E-2
6 through 11	for applications using EPROM devices	Figure E-3
12 through 14	for applications using NVRAM devices	Figure E-4
15 through 16	for applications using EEPROM devices	Figure E-5
17	for applications using mixed devices	Figure E-6

Note that the memory decode PAL (U59) does not support all the possible memory sizes. A new PAL can be programmed if the default decode configuration does not fit the application.

Some of the figures list a specific part number in addition to the more generic part number. As an example, Figure E-1 shows the matrix configuration for 2K x 8 Static RAM devices (configuration 0). You can use any electrical and mechanical equivalent as described in Appendix D.

STEP 2 - Selecting the Address Range for Bank B

In this step of the configuration, you select the memory size of Bank B. Then, you perform the memory size configuration for Bank A in STEP 3. Select the address range for Bank B by configuring jumper inputs E194 through E197 for PAL U59. As you do, you select one of four memory map configurations for PROM. Table 2-2 lists the four address options that are available to you through the configuration of the jumpers.

If one of the four jumper options does not provide a suitable solution, you can custom-program a replacement PAL for socket U59 to create your own solution. Appendix G shows the PAL source code for the factory memory decode which is provided as an example.

Table 2-2. PROM Site Addresses and Jumpers (U40, 41, 73, 74)

Jumpers		Prom Size	Addresses (Hex)	
E195-E197	E194-E196		Prom Sites	80130 ++
OUT	OUT	8K x 8 §	F8000-FFFFF	F4000-F7FFF
OUT	IN	16K x 8	F0000-FFFFF	EC000-EFFFF
IN	OUT	32K x 8	E0000-FFFFF	DC000-DFFFF
IN	IN	64K x 8	C0000-FFFFF	BC000-BFFFF

Notes: § = Default Jumper
++ = 80130 Address is only enabled when Jumper E204-E205 is installed.

STEP 3 - Selecting the Address Range and Placement of Bank A

In this step of the configuration, you select the address size of the Bank A memory area, and whether these sites are placed at lower memory (RAM starting at 0H) or at upper memory (EPROM starting immediately below Bank B.

Select the address range by configuring jumpers E198 through E203 as shown in Table 2-3. When addressed as lower memory there are three possible device sizes allowed: 2K x 8, 8K x 8, or 32K x 8. When addressed as upper memory, the Bank A sites must be addressed as the same size as the devices installed in Bank B. In this case the Bank A sites will be addressed immediately below the Bank B sites as shown in Figure 2-3, and the 80130 (if enabled) will be addressed immediately below the Bank A sites.

When using the iSBC 341 Memory Expansion board to expand RAM, you must remove jumper E207-E208 and install E206-E207. The iSBC 341 board memory sites will then be enabled for addresses immediately above the four Bank A sites, as shown in Table 2-3. In this case, the iLBX bus address space, if enabled, begins immediately above the iSBC 341 board addresses. The iSBC 341 board can not be used to expand the Bank A sites when decoded as upper memory, unless you custom-program and install a new memory decode PAL.

Table 2-3 shows the seven possible options that are available through the device configuration jumpers. One option includes the ability to disable the Bank A sites and the 80130.

NOTE

When Bank A sites are disabled or moved into upper memory, the iLBX bus address range begins at 00000H. (Unless the iLBX bus is disabled.)

If one of the seven jumper options does not provide a suitable solution, you can custom-program a replacement PAL for socket U59 to create your own solution. Appendix G shows the PAL source equations for the factory installed memory decode which is provided as an example.

Table 2-3. Bank A (RAM) Site Addresses and Jumpers (U42, 43, 75, 76)

Jumpers				Address (Hex)		
				Ram Sites		
202-203	200-201	198-199	Ram Size	4 Sites	8 Sites	80130 +
OUT	OUT	OUT	2K x 8 §	0-01FFF	02000-03FFF	Note 1
OUT	OUT	IN	8K x 8	0-07FFF	08000-0FFFF	Note 1
OUT	IN	OUT	32K x 8	0-1FFFF	20000-3FFFF	Note 1
OUT	IN	IN	N/A	N/A	N/A	Note 1
IN	OUT	OUT	Note 2 8K x 8	F0000-F7FFF	N/A	EC000-EFFFF
IN	OUT	IN	Note 2 16K x 8	E0000-EFFFF	N/A	DC000-DFFFF
IN	IN	OUT	Note 2 32K x 8	C0000-DFFFF	N/A	BC000-BFFFF
IN	IN	IN	DISABLED	DISABLED	DISABLED	DIS'ED

Notes: § Default Configuration.
+ The 80130 is enabled only when jumper E204-E205 is installed.
** Eight sites (with the iSBC 341 board) are enabled only when jumper E207-E208 is replaced by E206-E207.
1: When RAM sites are decoded at lower memory, the 80130 address is decoded as shown in Table 2-2.
2: In this configuration, RAM sites are decoded as upper memory addresses immediately below the Bank B sites.

STEP 4 - Select the Number of Wait-States

Because different types of local memory devices output their data at different rates, you must configure (with jumpers) the board for either 0, 1, or 2 wait-states, depending on the devices used. You can independently assign the number of wait-states to Bank A and Bank B. For example, Bank A could be configured for 0 wait-states while Bank B could be configured for 2 wait states.

Table 2-4 shows the jumper configurations for memory devices to meet 0, 1, or 2 wait states. Timing requirements for these wait-states are shown in Appendix D.

NOTE

When Intel iRAM devices are installed, jumper E238-E239 must be installed and the speed must be configured for no less than 1 wait state.

Table 2-4. Wait-State Jumper Options

Wait states	Bank A (RAM)	Bank B (PROM)
0	None	None
1	E124 to E125	E127 to E128
2	E124 to E123	E126 to E127

2.3.4.4 Battery Backup

The Bank A sockets allow for battery backup using SRAM devices only. The +5V battery voltage must be provided through connector J4 as shown in Table 2-24. The Memory Protect (MPRO*) signal must also be provided via J4. When MPRO* becomes active, no further chip select signals are allowed to the memory devices. This protects the data in memory during the power up/down event. MPRO* must be activated after all critical data and the present state of the CPU registers are stored, and before the power drops below 4.75V. Refer to Section 2.12.3 for additional battery backup information.

2.3.4.5 Using NVRAMs

The iSBC 186/03 board provides a non-volatile RAM enable signal (NVEN*). This signal can be driven from the 8255A parallel port or from an off-board source via connector J4. NVEN* must be activated to perform a STORE or RECALL cycle in the NVRAM devices. Refer to the Caution note in the "Allowable Memory Device Types" section.

NOTE

EEPROM and NVRAM specifications require that CE* or WE* for EEPROMs and NVEN* for NVRAMs remain inactive on power up/down while Vcc is above 4.0 volts. Since the digital circuits on the iSBC 186/03 board are not guaranteed to operate below 4.75 volts, in order to guarantee data retention, analog circuitry mounted on an iSBC MULTIMODULE as described in the 2817A data sheet, or battery backup (MPRO*, +5V battery) circuitry should be used to lock out these control signals during power up/down. Refer to Section 2.3.4.4 for battery backup details.

2.3.4.6 Using EEPROMs

The iSBC 186/03 board is designed to use intelligent 2817A-type EEPROM devices. These devices must be the type in which the RDY output from the device is connected to the 80186 Ready circuit via jumpers E147-E146 for Bank A, or E132 to E131 for Bank B.

In this way, when a write cycle is performed to an EEPROM device, the cycle will be completed by the 80186 before the EEPROM RDY signal goes active. However, if a subsequent access is made to the same bank before the internal write has completed, the RDY signal will force the 80186 to execute wait-states until the internal write cycle has completed.

Alternatively, the RDY signal from the EEPROM device could be wired to an interrupt input rather than the 80186 Ready logic. In this case, E147 to E146 and E131 to E132 must be removed. The RDY pin can be connected to the interrupt matrix via the NVEN* signal by connecting E131 to E133 for Bank B or E146 to E148 for Bank A. The NVEN* signal must then be connected to an interrupt input on the 80130 via jumper post E115. The 80130 interrupt input must be programmed to the edge-triggered mode. In this way RDY will go high when the EEPROM device has completed its internal write cycle and is ready to accept another cycle.

2.4 DIRECT MEMORY ACCESS (DMA) CONTROLLER

The iSBC 186/03 board provides a DMA controller (integral to the 80186 processor) with two channels. DMA may be performed I/O to I/O, memory to memory, or I/O to and from memory. A memory to memory DMA transfer is a non-synchronized type and appears like a normal read/write cycle with no DMA request inputs. DMA to or from I/O is source or destination synchronized and requires a DMA request signal from the synchronizing device. The devices that may signal for DMA requests are:

- Parallel Port Interface
- User-installed iSBX boards
- 8274 Multiple Protocol Serial Controller (MPSC) device

There are seven possible sources for the two DMA channels. Therefore, a DMA Multiplexer is provided to allow you to programmatically select a subset of the sources for each DMA channel. Once the DMA controller has been initialized, the DMA operation may be started by a software command to the controller (for non-synchronized transfers) or by the synchronizing device issuing a DMA request signal. Normally the DMA request would be deactivated when a read command or a write command is generated to the synchronizing device. However, if the DMA request signal is held active until the last operation of the DMA transfer, then the DMA controller will maintain control of the bus until the DMA is completed. In this mode the CPU operation is suspended for as long as it takes to complete the DMA operation.

When performing DMA to an iSBX board installed into one of the iSBX bus connectors, the data can be read/written to the iSBX MDACK address rather than the iSBX programmed I/O address. This generates the proper DMA acknowledge signal to the iSBX interface.

When performing a DMA operation to the 8274 MPSC device, the 8274 device must be programmed to place the particular channel(s) into the DMA mode. In the DMA mode, both receive and transmit DMA requests are generated for the channel. If you only want one direction (i.e., TxDRQA) connected to the 80186 DMA controller, then the other direction (RxDRQA) must be connected to an interrupt line. If only one channel is selected for DMA, that channel must be channel A. When either channel is in the interrupt mode, RxDRQB/IPI must be grounded (via jumper E51-E52) to provide the Interrupt Priority Input (IPI) signal.

There are only two DMA channels in the 80186 and there are seven possible sources for DMA requests. A Multiplexer PAL is provided to allow the user to programmatically select a subset of the sources for each DMA channel. Inside the PAL there are three select flip-flops, SEL0*, SEL1*, and SEL2*. SEL0* and SEL1* are used along with one input jumper (IN0) to select one source for DRQ0 DMA request signal. SEL2* and IN1 are used to select another source for DRQ1. In this way, the DMA source may be selected via program control from a subset of possible sources. Upon RESET, all select flops are set (low). Tables 2-5, 2-6, and 2-7 show required programming to modify the select flip-flops and which DMA Request source is selected. Each SEL line powers up in the 0 state represented by RESET in Tables 2-6 and 2-7.

Table 2-5. DMA Request Multiplexer Addressing

Address (write only)	Select Output
F0H	SEL0*=0
F1H	SEL0*=1
F2H	SEL1*=0
F3H	SEL1*=1
F4H	SEL2*=0
F5H	SEL2*=1
F6H	TDMA1 pulsed
F7H	Reserved
Note: 0 = low	

Table 2-6. DRQ0 Source Selection

Jumper	DRQ0 Source	SEL1*	SEL0* (0 = Low)
IN0 = 0 E56 to E58 Installed	SBX 1 MDRQT SBX 2 MDRQT TxDRQB SCSI DRQ	0 0 1 1	0 RESET 1 0 1
IN0 = 1 § E56 to E58 Removed	RxDRQA RxDRQB TxDRQB SCSI DRQ	0 0 1 1	0 RESET 1 0 1
Note: § = Default Jumper.			

Table 2-7. DRQ1 Source Selection

Jumper	DRQ1 Source	SEL2*
IN1 = 0 E57 to E59 Installed	SBX 1 MDRQT TxDRQA	0 RESET 1
IN1 = 1 § E57 to E59 Removed	TxDRQA RxDRQB	0 RESET 1
Note: § = Default Jumper.		

The 80186 can generate a Terminate DMA (TDMA) signal to one of the two iSBX bus connectors by writing any data value to I/O address 00F6H. The TDMA1 signal can be jumper connected either to iSBX bus connector J7 (SBX1) by installing jumper E93-E94 or routed to iSBX bus connector J6 (SBX2) by installing jumper E91-E94.

The iSBX module can terminate the DMA operation by activating the TDMA signal. In this mode, the TDMA signal must be inverted. The TDMA signal is inverted by routing the signal to the input of one of the general purpose inverters in the interrupt matrix and connecting the output from the inverter to an interrupt destination. For example, SBX1 TDMA could be routed to the IR5 input on the 80130 by installing jumpers E93-E96 and E71 to E79. TDMA is configurable either as an input to the expansion module or as an output from the expansion module, but not both.

2.5 iLBX™ BUS INTERFACE DESCRIPTION

The iSBC 186/03 board contains a specialized, high-speed, local memory, expansion interface available at the P2 Connector, called the iLBX bus interface. The iLBX bus interface allows you to expand local memory on the iSBC 186/03 board by linking iLBX memory boards directly to the iSBC 186/03 board via an iLBX cable and connectors. This interface uses the P2 connector in its default condition. The iSBC 186/03 board iLBX bus interface is designed to be a Primary Master Only, as described in the INTEL iLBX BUS SPECIFICATION.

Through the iLBX bus interface, you can expand the local memory resources on the iSBC 186/03 board to a maximum of 896K-bytes. The performance between the 80186 CPU and the iLBX bus memory is comparable to that between the 80186 CPU and the on-board local memory.

Refer to the iLBX BUS SPECIFICATION document for the pin assignments and the AC and DC operating characteristics of the iLBX bus interface.

2.5.1 iLBX™ BUS INTERFACE JUMPER CONFIGURATIONS

In the default configuration, the address of the iLBX bus memory begins immediately above the Bank A (RAM) memory, at address 02000H, and ends at 07FFFFH. If Bank A is disabled via jumpers E198 through E203 or is moved into upper memory, the iLBX bus address starts at address 0H. The size of the iLBX bus memory space can be expanded by modifying the jumpers as shown in Table 2-8.

Table 2-8. iLBX™ Bus Address Ranges

Jumpers		Address Range**
E44-E45	E42-E43	
Out	Out	00000H-DFFFFH (896K)++
Out	In	00000H-BFFFFH (768K)++
In §	Out §	00000H-7FFFFH (512K)
In	In	Not Allowed

Notes:
§ = Default Jumper.
** = For iLBX addressing options that begin at 0H, the decode PAL allows the four RAM BANK sockets to be moved to support PROM address space. See Table 2-1 for allowable PROM/EPROM sizes. If Bank A starts at 0H, then iLBX bus memory begins immediately above Bank A memory.
++ = Selecting iLBX options above 512KB disallows the use of the MULTIBUS 16MB window addressing.

If your application does not need to use the iLBX Bus, you can remove jumper E240-E241 and E163-E164. Doing this reassigns the memory space normally assigned to the iLBX bus memory to the MULTIBUS memory. It also tri-states the iLBX bus address drivers from the P2 connector. The connector can then be used for other purposes, as described in Section 2.13.

The iSBC 186/03 board allows for 0 wait state operation under specific conditions. In the 0 wait state mode, all iLBX bus memory boards must have static memory devices (no refresh allowed) installed and no access delays caused by another request to iLBX bus memory are allowed.

The iSBC 428 board meets these requirements, when static RAMs or EPROMs are installed on it. To allow 0 wait state operation, the iSBC 186/03 board must be configured to ignore the iLBX bus acknowledge signal. This is accomplished by removing jumper E165-E166 and installing jumper E166-E167. All other timing requirements are given in Table 2-9 and Figure 2-6.

As an example, to configure an Intel iSBC 028CX/056CX/012CX memory board for 1 wait state read, install jumpers E38-E68 and E17-E16 on the memory board. Refer to the Hardware Reference Manual for the iSBC 028CX/056CX/012CX memory boards for more information.

2.5.1.1 iLBX™ Bus Timing Specifications

The specific timing requirements that must be met by the iLBX bus memory board when interfacing to the iSBC 186/03 board are given in Table 2-9 and Figure 2-6. The iSBC 186/03 board iLBX bus section can operate in the optimized mode as described in the INTEL iLBX BUS SPECIFICATION. This means that the slave board can respond with an ACK* signal anytime after the iSBC 186/05 board activates ASTB*. See note in Table 2-9 for jumper instructions.

Table 2-9. iLBX™ Bus Timing Specifications

Parameter	Wait States		
	0	1	2
ta,b) ASTB* to ACK* (min-max)			
Read, 6MHz		84-173ns	250-403ns
Write, 6MHz		0-83ns	155-249ns
tc) ASTB* to Read Data valid (max)			
6MHz	266ns	432ns	598ns
td) DSTB* to Read Data valid (max)			
6MHz	241ns	407ns	573ns
te) ACK* to Read Data valid (max)			
6MHz	243ns	243ns	243ns
tf,g) DSTB* to ACK* (min-max)			
Read, 6MHz	---	79-148ns	245-314ns
Write, 6MHz	---	82-7ns	84-173ns
th) ASTB* to Write Data 6MHz (max)	78ns	78ns	78ns
ti) DSTB* to Write Data 6MHz (max)	24ns	24ns	24ns
tj) ASTB* to DSTB*			
6MHz write cycle (min-max)	49-85ns	49-85ns	49-85ns
6MHz read cycle (min-max)	5-25ns	5-25ns	5-25ns

Note: ** For 0 wait states in the optimized mode, remove jumper E165-E166 and install jumper E166-E167. In this mode refresh is not allowed and all iLBX bus memory boards must perform at 0 wait states.

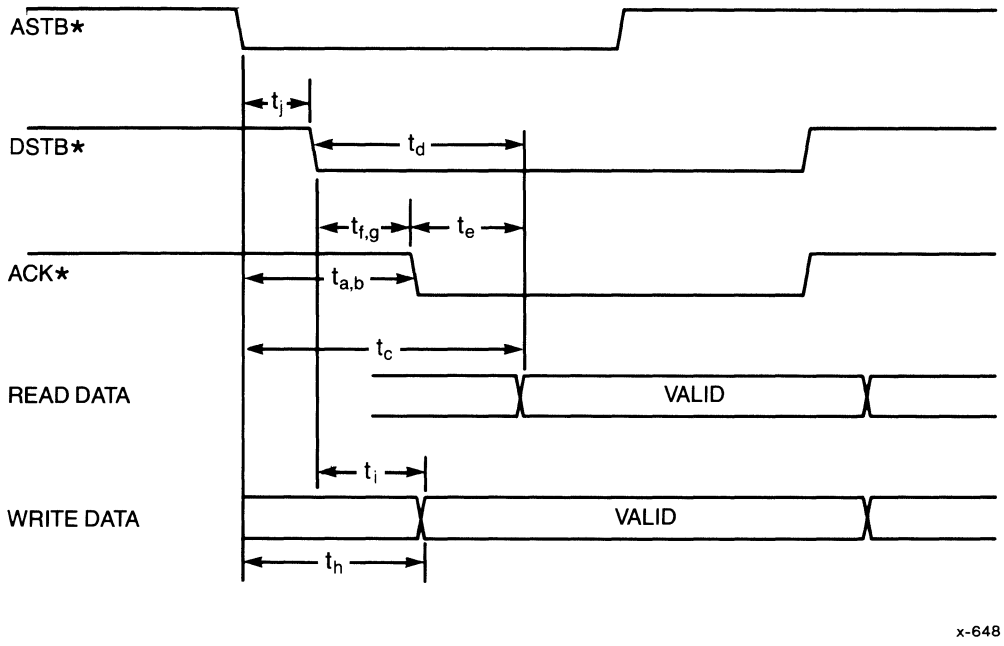


Figure 2-5. iLBX™ Bus Timing Specifications Diagram

2.6 INTERRUPT CONFIGURATION

The iSBC 186/03 board handles both on-board and off-board interrupt requests. On-board interrupts originate from a variety of sources, as listed in Table 2-10. MULTIBUS interrupt requests are routed to the board on MULTIBUS lines INT0* - INT7*. This allows you to reconfigure the sources and destinations of the various requests to suit your application. Table 2-10 lists these sources and Table 2-11 provides the possible destinations. The default connections are listed in Table 2-12.

Interrupt hardware on the iSBC 186/03 board is default-configured in the following manner: the interrupt controller portion of the 80130 component is the master interrupt controller. Connected to the master, as slave interrupt controllers, are the 8259A interrupt controller (U24), the 8274 serial interface controller (U22), and the interrupt controller portion of the 80186 CPU.

Table 2-10. Interrupt Source Jumper Matrix Options

Jumper Post	Interrupt Sources
E31	80186 Timer 0 output (internal)
E172	80186 Timer 1 output (internal)
	80186 Timer 2 output (internal)
	DMA 0 (internal)
	DMA 1 (internal)
E87	SBX 1 INTR0
E88	SBX 1 INTR1
E89	SBX 2 INTR0
E90	SBX 2 INTR1
E93	SBX 1 TDMA
E91	SBX 2 TDMA
E66	8274 INT (connect to IR3 on 80130 if in 8274 vectored mode)
E69	Timeout Interrupt
E65	80186 Slave INT (connect only to IR4 on 80130)
E67	80130 SYSTICK TMR
E46	80130 DELAY TMR
E173	80130 BAUD TMR
E34	80130 BAUD TMR
E122	80130 BAUD TMR
E64	Parallel Port
E68	SCSI INT
E72	EXT INTR (from J4-11)
E62	Not Used - Reserved
E115	NVEN*
E70	8274 TxDRQA
E85	8274 RxDRQA
E95	8274 TxDRQB
E105	8274 RxDRQB
E86	PFIN*

Table 2-10. Interrupt Source Jumper Matrix Options (continued)

Jumper Post	Interrupt Sources
E227	MB INT OUT (connect to E228 through E235 only)
E114	MULTIBUS Interrupt Level 0 (INT0*) Input
E113	MULTIBUS Interrupt Level 1 (INT1*) Input
E112	MULTIBUS Interrupt Level 2 (INT2*) Input
E111	MULTIBUS Interrupt Level 3 (INT3*) Input
E110	MULTIBUS Interrupt Level 4 (INT4*) Input
E109	MULTIBUS Interrupt Level 5 (INT5*) Input
E108	MULTIBUS Interrupt Level 6 (INT6*) Input
E107	MULTIBUS Interrupt Level 7 (INT7*) Input
E71	Floating Inverter

Table 2-11. Interrupt Destination Jumper Matrix Options

Jumper Post	Interrupt Destinations
E84	80130 IR0
E83	80130 IR1
E82	80130 IR2
E81	80130 IR3 (Reserved for 8274 vectored mode interrupts)
E80	80130 IR4
E79	80130 IR5
E78	80130 IR6
E77	80130 IR7
E104	8259A IR0
E103	8259A IR1
E102	8259A IR2
E101	8259A IR3
E100	8259A IR4
E99	8259A IR5
E98	8259A IR6
E97	8259A IR7
E73	NMI (non-maskable interrupt)
E228	MULTIBUS Interrupt Line 0 (INT0*) Output
E229	MULTIBUS Interrupt Line 1 (INT1*) Output
E230	MULTIBUS Interrupt Line 2 (INT2*) Output
E231	MULTIBUS Interrupt Line 3 (INT3*) Output
E232	MULTIBUS Interrupt Line 4 (INT4*) Output
E233	MULTIBUS Interrupt Line 5 (INT5*) Output
E234	MULTIBUS Interrupt Line 6 (INT6*) Output
E235	MULTIBUS Interrupt Line 7 (INT7*) Output

Table 2-12. Default Interrupt Level Assignments

Level	PIC	Jumper	Location	Function
8259A Interrupt Level Assignments				
IR0	U24	E104	Interrupt Matrix	Not connected
IR1	U24	E103	Interrupt Matrix	Not connected
IR2	U24	E102	Interrupt Matrix	Not connected
IR3	U24	E101	Interrupt Matrix	Not connected
IR4	U24	E100	Interrupt Matrix	Not connected
IR5	U24	E99-E109 §	Interrupt Matrix	Interrupt signal from MULTIBUS (INT5*).
IR6	U24	E98	Interrupt Matrix	Not connected
IR7	U24	E97	Interrupt Matrix	Not connected
80130 Interrupt Level Assignments				
IR0	U23	E84	Interrupt Matrix	Not connected
IR1	U23	E83	Interrupt Matrix	Not connected
IR2	U23	E82-E67 §	Interrupt Matrix	Interrupt from 80130 internal timer SYSTICK
IR3	U23	E81-E66 §	Interrupt Matrix	Interrupt from the 8274 MPSC. See Note 1.
IR4	U23	E80-E65 §	Interrupt Matrix	Interrupt from the 80186 slave PIC See Note 2.
IR5	U23	E79	Interrupt Matrix	Not connected
IR6	U23	E78-E63 §	Interrupt Matrix	Interrupt from the 8259 slave PIC
IR7	U23	E77	Interrupt Matrix	Not connected

Notes: § = default jumper installed.

1. IR3 used only for the 8274 controller, and only when it is programmed for the vectored mode. The 8274 can be used in non-vectored (iRMX 86 compatible) mode by connecting its interrupt output (E66) to another 80130 interrupt input. In this configuration IR3 cannot be used for any other function.
2. IR4 used only for the 80186 slave interrupts.

2.6.1 INTERRUPT JUMPER MATRIX CONFIGURATION

The iSBC 186/03 board contains an array of stake pins, used to configure the interrupt functions on the board. Figure 2-7 shows the physical layout of the interrupt matrix and shows the default jumpers installed. The matrix is located between iSBX bus connectors J6 and J7.

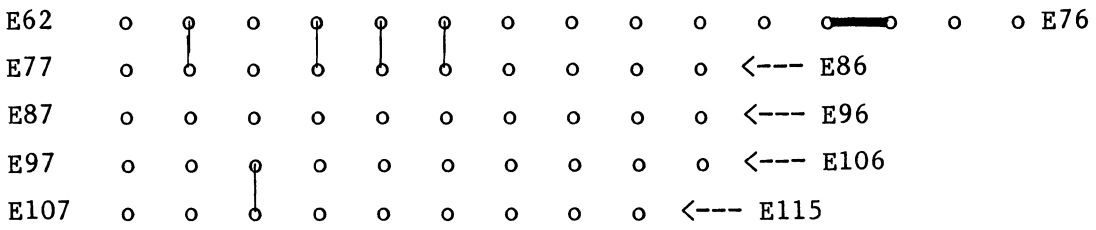


Figure 2-6. Default Configuration Of Interrupt Jumper Matrix

2.6.1.1 On-Board Interrupt Sources

This section describes in more detail each of the interrupt sources on the iSBC 186/03 board.

TIME OUT INTR

Timeout Interrupt. A timeout will occur whenever the processor is held in wait-states longer than the timeout period (10Ms). The timeout will occur for any access (memory or I/O) which is not acknowledged. A timeout interrupt signal is routed to the Interrupt Matrix (post E69). If you want a timeout condition to create an interrupt then you must install a jumper to an unused interrupt input post on the 80130 PIC or the 8259A PIC. The interrupt input must be programmed for the edge mode. Note that a timeout interrupt is not required when using the iRMX 86 Operating System.

During a HALT instruction, the timeout can be disabled by connecting the 80130 BAUD timer output to the timeout circuit via jumper E120-E122. The timer should be initialized to the square wave mode with a period of 1ms. Refer to the 80130 timer section of this chapter for more information.

SCSI INT Signal

Small Computer Systems Interface (SCSI) Interrupt. This signal would only be used in conjunction with a SCSI interrupt or some other custom parallel interface which generates this signal. The parallal port must be configured for SCSI operation as described in Appendix C. To enable the SCSI interrupt you would need to install a jumper from E68 to an unused interrupt input post on the 80130 PIC or 8259A PIC.

NOTE

If using the iRMX 86 Operating System (system clock on IR2) and the SCSI interrupt is connected to IR1, the accuracy of the system clock may decline. To correct this, place the system clock on a higher interrupt level than the SCSI interrupt. Notice also that the operating system will need to be reconfigured.

PPI INT Signal

Parallel Port Interrupt. This is an optional signal which must be implemented at the parallel port jumper matrix.

For example, as described in Appendix C, when using the parallel port in the Centronics interface configuration this line is dedicated to the printer FAULT function.

To connect PPI INT to the 80130 you need to install a jumper from E64 to an usused interrupt input post on the 80130 PIC.

NVEN*

This interrupt jumper post is provided to allow a non-intelligent EEPROM device to interrupt the CPU when the device is ready to accept another write cycle during programming. This can be done by connecting the RDY signal from the EEPROM device to the NVEN* signal line at the memory jumper matrix. Then connect the NVEN* signal line to an 80130 PIC interrupt input. Refer to Section 2.3.4.6 for further details.

8274 TXDRQA, 8274 TXDRQB and 8274 RXDRQA, 8274 RXDRQB/IPI

As previously stated in the DMA section of this chapter, when performing a DMA operation to the 8274, the 8274 must be programmed to place the particular channel(s) into the DMA mode. In the DMA mode, both receive and transmit DMA requests are generated for the channel. If you only want one direction (i.e., TxDRQA) connected to the 80186 DMA controller, then the other direction (RxDRQA) must be connected to an interrupt line. If only one channel is selected for DMA, that channel must be channel A. When either channel is in the interrupt mode, RxDRQB/IPI must be grounded (via jumper E51-E52) to provide the Interrupt Priority Input (IPI) signal to the 8274.

iSBX™ Bus Interrupts

The following interrupt sources (or destinations) are associated with the iSBX bus:

• SBX1 INTR0	Source	(E87)
• SBX1 INTR1	Source	(E88)
• SBX2 INTR0	Source	(E89)
• SBX2 INTR1	Source	(E90)
• SBX2 TDMA	Source/Destination	(E91)
• SBX1 TDMA	Source/Destination	(E93)

The following paragraphs discuss these options.

SBX1 INTR0/INTR1 and SBX2 INTR0/INTR1

These are the interrupt lines reserved for any optional iSBX bus boards. SBX1 signals come from the J7 connector; SBX2 signals come from the J6 connector. The INTR0 and INTR1 interrupt lines correspond to the MINTR0 (pin 14) and MINTR1 (pin 12) lines of the iSBX bus connector. For more information on the use of these lines, refer to the Intel iSBX bus Specification.

SBX1 TDMA and SBX2 TDMA

These two signals can be driven by the iSBC 186/03 board, causing the iSBX board to terminate its DMA request (see section 2.4). These signals can also be driven by the iSBX board, causing the iSBC 186/03 board to terminate DMA. In this mode the SBX1 TDMA or SBX2 TDMA signals must be inverted by connecting to E75 or E96 and the output of the inverter (E76 or E71) would then be connected to an interrupt request input on the 80130 PIC or 8259A PIC. The interrupt input on the 80130 or 8259A must be programmed for the edge-triggered mode. The iSBC 186/03 board would then receive an interrupt when the TDMA pulse arrived from the iSBX board.

MULTIBUS® Interrupts and MB INT OUT*

The iSBC 186/03 board supports eight MULTIBUS input interrupt lines. These eight lines can be connected to the 8259A Slave interrupt controller (U24). In the default configuration, MULTIBUS interrupt INT5* is connected to IR5 on the 8259A.

The MB INT OUT* signal can be used to drive an interrupt onto any of the MULTIBUS interrupt lines INT0* through INT7*. Jumper posts E228 through E235 are used for this purpose. To use the MB INT OUT* signal you must first install a jumper from your source of the interrupt to post E106. This is the input to a driver/inverter circuit. Then install a jumper from the output of this driver/inverter circuit (E227) to the desired MULTIBUS output post (E228 through E235).

J4 Interrupts

Two other interrupt sources are available through connector J4. They are PFIN* and EXTINT. The following paragraphs discuss these options.

PFIN*

Power Fail Interrupt. This active-low signal could be used to interrupt the processor in the event of a power failure. Typically, this signal would come from the power supply. Since it is an active-low signal, it must be inverted before connecting to the 80130 PIC. To accomplish this install jumper E86 to E96. Then install a jumper from E71 (output of inverter) to the desired interrupt input post.

EXTINT

External Interrupt. This interrupt could be used for any general-purpose external event. To use it, install a jumper connector from E72 to the desired interrupt input post. The input must be an active-high TTL level.

2.7 TIMERS

The iSBC 186/03 board contains three timers integrated in the 80186 processor and three timers in the 80130. Timers 0 and 1 of the 80186 are default wired to be the baud rate sources for 8274 serial channels A and B respectively. If using these two channels for baud rate, they should be programmed for square wave operation. Refer to the 80186 data sheet for additional details. The three timers in the 80130 are described in Chapter 4. The BAUD output of the 80130 can be used as a general purpose square wave output.

2.7.1 TIMER JUMPER CONFIGURATIONS

When you receive your iSBC 186/03 board from the factory, the timers are configured (jumpered) as follows:

80186 Timers:

- Timer 0 output (TMR OUT 0) is used as the baud rate source for channel A of the 8274 MPSC device.
- Timer 1 output (TMR OUT 1) is used as the baud rate source for channel B of the 8274 MPSC device.
- Timer 2 output can used internally as a prescaler to Timers 0 and 1.

80130 Timers:

- Timer 0 output (SYSTICK) is connected externally and used as the internal system clock.
- Timer 1 output (DELAY) is not used.
- Timer 2 output (BAUD) is not connected, but can be used as the baud rate source for either serial channel.

The following two sections cover each group of timers and their options in more detail.

2.7.1.1 80186 Timers

Three interval timers are provided on the 80186 component. The timers are sampled every fourth CPU clock cycle, making the base clock rate of each timer 1.5MHz (one-fourth of the 6MHz CPU clock).

Timers 0 and 1 normally supply the baud rates for the 8274 serial channels. Timers 0 and 1 contain two MAX count registers for controlling the duty cycle of the output waveforms on jumper posts E31 and E172. In addition, these two timers may be used for counting external events on input jumper posts E186 and E187. The timers could also be gated by external events using E186 and E187. Timer 2 can be used as a prescale counter to timers 0 and 1 to increase the cycle time of these counters.

The jumpers associated with these timers are summarized as follows:

80186 Timer 0 Input	E186
80186 Timer 0 Output (inverted)	E31
80186 Timer 1 Input	E187
80186 Timer 1 Output	E172
80186 Timer 0 to 8274 Ch. A Clock	E31-E32
80186 Timer 1 to 8274 Ch. B Clock	E171-E172

For additional information on the 80186 timers, refer to the iAPX 186 Data Sheet, Intel Application Note AP-186, and Chapter 4, Programming Information.

2.7.1.2 80130 Timers

The 80130 component contains three timers. Timer 0 is pre-defined as a rate generator and can be used as a system timer (SYSTICK). In the default configuration its output is connected to IR2 of the 80130 PIC. The input clock for timer 1 (DELAY) is derived from the output of timer 0 (SYSTICK).

Timer 2 is predefined as a square wave generator and can be used for baud rate or for general purpose. The BAUD output may also be used to disable a timeout and timeout interrupt during a CPU HALT cycle. In this mode, the BAUD output is connected to the clock input of the timeout generator by installing jumper E120 to E122. Note that this is not required to run the iRMX 86 Operating System.

Certain precautions must be considered when using this mode:

1. If a DMA cycle is allowed to occur during the HALT cycle, a timeout will occur within the predefined period (10ms approximately) of the last DMA cycle.

2. An interrupt indicating the end of a DMA transfer will force the CPU out of the HALT state.

For more information on the iSBC 186/03 board timeout feature, refer to the Timeout paragraph of the Interrupt section in this chapter.

The following list summarizes the jumper posts associated with the 80130 timers:

80130 BAUD Output	E34, E173, E122
80130 SYSTICK Output	E67
80130 DELAY Input	E46
80130 BAUD to 8274 Ch. A	E34-E35
80130 BAUD to 8274 Ch. B	E171-E173

Refer to Chapter 4 for 80130 timer programming information and operation details.

2.8 PARALLEL I/O INTERFACE - CONNECTOR J3

The parallel I/O Interface on the iSBC 186/03 board provides a general purpose parallel interface as shipped from the factory. It utilizes the 8255A parallel peripheral interface (PPI) controller and contains three 8 bit ports: A, B and C. Port A is configured as an output port with a Data Transceiver that can be reconfigured as an input or bidirectional port. The direction of this port can be controlled via jumpers, an external signal, a Port C control signal, or from a control PAL.

Port B is an input or low power output port.

The lower 4 bits of Port C are connected to open - collector drivers as an output port. The upper 4 bits are used for on-board functions and control of the interface. Two sockets are provided for user-installed Programmable Array Logic (PAL) devices for implementing special parallel Interfaces.

Socketed 1K ohm, SIP resistors terminate port A, B and C signals. These may be replaced with different valued 8-pin SIP resistors in either pull-up or pull-up/pull-down configuration for different termination requirements.

The parallel interface has provisions for controlling several on-board signals using Port C or external events. OVERRIDE* is used to control the LOCK function and NVEN* can be driven by a Port C signal to control a non-volatile RAM device (NVRAM) if used.

The interface has two interrupt signals. PPI INT is a general purpose interrupt that could be driven by a Port C signal or by an external signal and SCSI INT is driven by PAL site U18. A DMA request signal SCSI DRQ* is also driven by PAL site U18 for SCSI or Centronics interface use as described in Appendix C. A diagnostic LED can be controlled by an output signal or by an external event. In the as-shipped configuration, this LED should be ON when the board is powered up. The Centronics printer interface configuration is also discussed in Appendix C.

2.8.1 PARALLEL INTERFACE PIN ASSIGNMENTS

Table 2-13 lists the pin assignments for connector J3 interface in the default configuration. A diagram shows the physical locations of the pins in the connector.

Table 2-13. Parallel Interface Default Pin Assignments

J3 Pin number	Signal function	J3 Pin number	Signal function
2	Port A Bit 0	26	Not Used
4	Port A Bit 1	28	Port C Bit 0*
6	Port A Bit 2	30	Port C Bit 1*
8	Port A Bit 3	32	Port C Bit 2*
10	Port A Bit 4	34	Port C Bit 3*
12	Port A Bit 5	36	Port B Bit 0
14	Port A Bit 6	38	Port B Bit 1
16	Port A Bit 7	40	Port B Bit 2
18	Reserved	42	Port B Bit 3
20	Not Used	44	Port B Bit 4
22	Not Used	46	Port B Bit 5
24	Not Used	48	Port B Bit 6
1-49 odd	Ground	50	Port B Bit 7

Pin Numbering Convention for J3 Connector

Front View, Component Side of Board Up

```
49 47 45 . . . . . 7 5 3 1
 o  o  o  o  o  o  o  o  o  o  o  o  o
 o  o  o  o  o  o  o  o  o  o  o  o  o
50 48 46 . . . . . 8 6 4 2
```

2.8.2 PARALLEL INTERFACE DRIVE CHARACTERISTICS

Table 2-14 presents the DC current drive and load specifications for the parallel interface connector J3 in the default configuration.

Table 2-14. Parallel Interface Drive Characteristics

Parameter	Conditions	Min	Max	Units
Port A				
IOL IOH IIL IIH	VOL= 0.5V Maximum VOH= 2 volts Minimum VIL= 0.4V VIH= 2.7V		24 -15 -0.2 20	MA MA MA uA
Port B				
IOL IOH IIL IIH	VOL= 0.45V Maximum VOH= 2.4V Minimum		1.7 -400 $\pm$10 $\pm$10	MA uA uA uA
Port C (Bits 0 - 3)				
IOL IOH(leakage)	VOL=.4v Maximum VOH= 5.5v Maximum		48 250	MA uA

2.8.3 PARALLEL INTERFACE CABLING

The parallel Interface on the iSBC 186/03 provides high-reliability locking type pin connectors. Compatible connectors for the J3 interface are given in Chapter 3. Pin 1 of the connector is located as shown in Figure 2-8.

2.8.4 PARALLEL INTERFACE JUMPER CONFIGURATIONS

The parallel port interface includes a matrix of 30 jumper posts to allow reconfiguration of the default interface to your own interface. Appendix C provides reconfiguration information for two standard interfaces: the Small Computer Systems Interface (SCSI) and the Centronics printer interface.

Port A direction control (input or output) can be controlled by a jumper to ground, by Port C bit 7 or by a PAL installed in socket U16. The output enable signal for Port A can be controlled by a jumper (to GND), by U16 or by Port C bit 4.

All eight bits of Port B are connected to the J3 connector in the default configuration. Bits 1 and 7 can be reconfigured so that they can be driven by Port C bits or by U16. Port B Bits 1, 2, 4 and 7 can be jumpered to ground or allowed to be pulled up and can be read by the CPU as configuration inputs. Port B Bits 1, 5, 6 and 7 can also be monitored by PAL site U18.

Port C Bits 0 through 3 are driven out to J3 via open-collector drivers. These open-collector drivers can also be controlled by the U18 PAL socket and may be reconfigured to other J3 pins via jumpers.

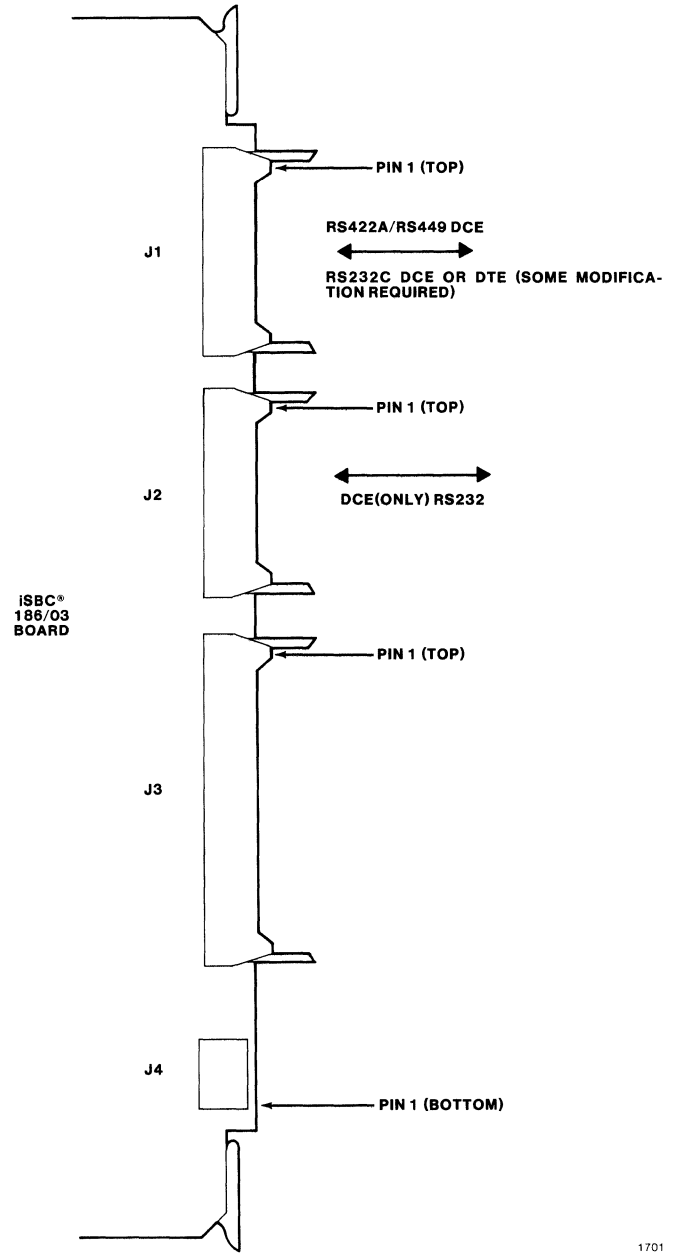


Figure 2-7. Physical Locations of Connectors and Pin 1 Assignments

Table 2-15. Parallel Port Jumper Options

Jumper Number	Description
E1-E11 §	Transceiver at Port A always enabled
E1-E2	Transceiver at Port A enable controlled by PC4
E14	Direction control for Port A transceiver
E14-E15	Port A in input mode (Remove jumper for output mode)
E14-E4	Port A direction controlled by PC7
E14-E24	Port A direction controlled by PAL site U16
E16-E26 §	Port B bit 1 to J3-38
E25-E26	Port B bit 1 to J3-50
E24-E25 §	Port B bit 7 to J3-50
E23-E24	OVERRIDE* signal controlled by PB-7
E22-E23	OVERRIDE* signal controlled by PC-5
E26	PB1 configuration jumper
E10	PB2 configuration jumper
E19	PB4 configuration jumper
E25	PB7 configuration jumper
E20-E30 §	PC0 to J3-28
E28-E29 §	PC1 to J3-30
E5-E6 & E7-E17 §	PC3 to J3-34
E2-E12 §	PC4 to Diagnostic LED
E8	Control of PC0,1 and 3 drivers
E21	NVEN* control for NVRAM
E12	Diagnostic LED control
E23	OVERRIDE* signal
E13	Control of "Test" pin on 80186 (Should be grounded when using iRMX 86)
E27	Parallel Port Interrupt

Notes: § = Default jumper installed.
Appendix C provides additional details on the default jumpering and the parallel port matrix layout.

2.9 SERIAL I/O INTERFACES

The iSBC 186/03 board provides two 26-pin serial I/O interfaces, one at Connector J1 and one at Connector J2. The board provides several configuration options on each interface. Connector J1 is shipped configured as RS422A/RS449 interface; this can be modified to become an RS232C interface. Connector J2 is configured as an RS232C interface; it cannot be changed to another interface. In each interface the electrical specification of the interface matches the EIA RS422A/RS449, or RS232 electrical requirements. Refer to the appropriate EIA interface document for complete details of these interfaces.

2.9.1 CONNECTOR J1 DESCRIPTION

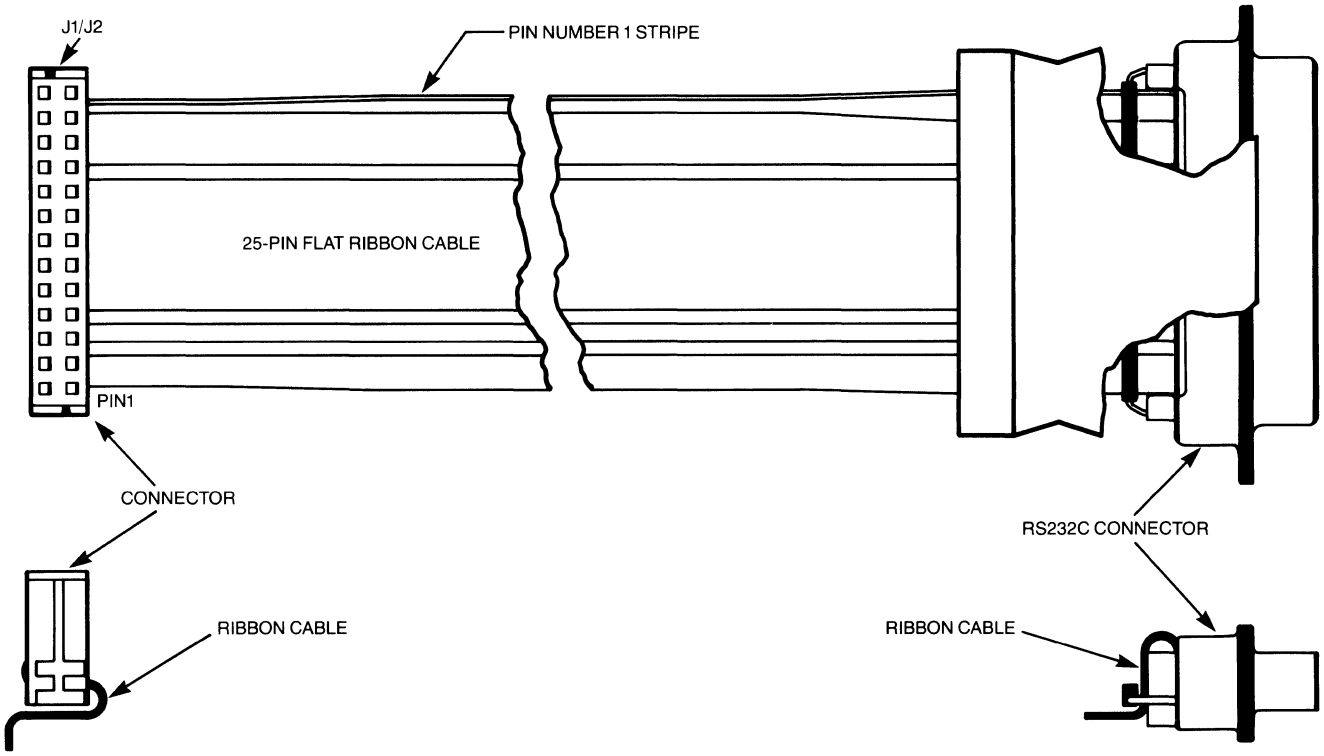
When you receive the iSBC 186/03 board from the factory, Connector J1 operates as a Data Communications Equipment (DCE) device with an RS422A/449 interface.

Table 2-16 provides the pin assignments for Connector J1 and lists the default RS422A/449 interface signals on each pin of the connector. If you change the interface to an RS232C interface, the pin assignment is identical to that listed for connector J2 in Table 2-18. Figure 2-9 shows a cable drawing of an RS232C cable for connector J1. Figure 2-8 shows the location of pin 1 for J2. The following diagram shows the pin numbering convention for both J1 and J2.

Pin Numbering Convention For J1 and J2 Connectors

Front View, Component Side Of Board Up

```
25 23 21 19 . . . . . 7 5 3 1
 o  o  o  o o o o o o o o o o o

 o  o  o  o o o o o o o o o o o
26 24 22 20 . . . . . 8 6 4 2
```



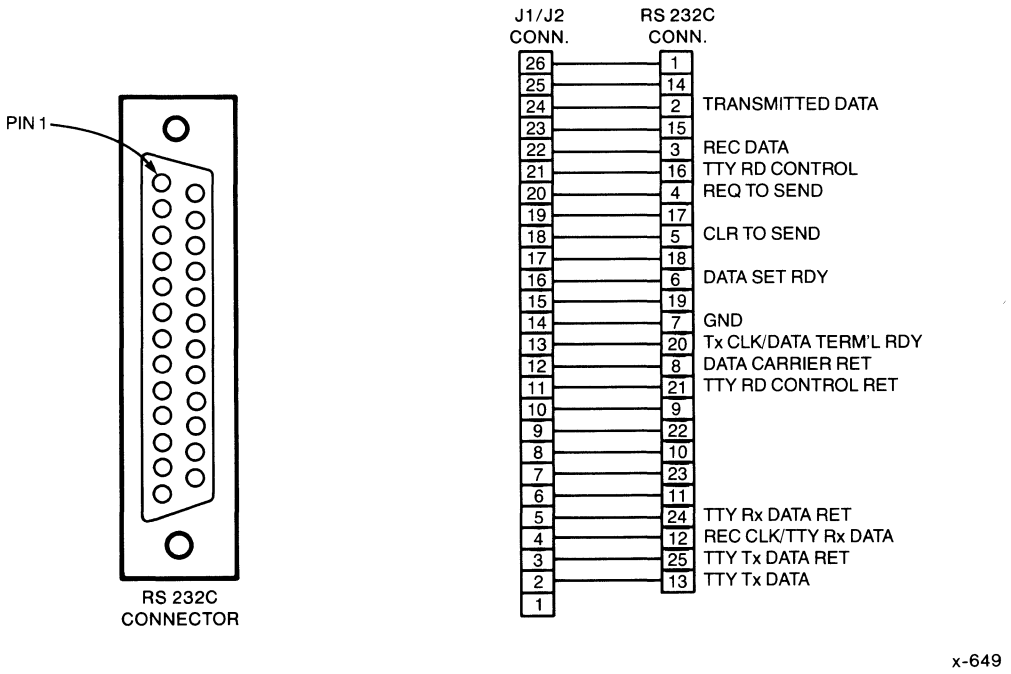


Figure 2-8. RS232C Cable Diagram for Connector J1

Table 2-16. Pin Assignment for Connector J1 - RS422A/449 Interface

J1 Pin Number	RS449 Signal Name	RS449 Signal Function
01	---	---
02	---	---
03	---	---
04	TR (A)	Terminal Ready
05	TR (B)	Terminal Ready
06	DM (A)	Data Mode
07	DM (B)	Data Mode
08	---	---
09	---	---
10	CS (A)	Clear To Send
11	CS (B)	Clear To Send
12	RT (A)	Receive Timing
13	RT (B)	Receive Timing
14	RS (A)	Request To Send
15	RS (B)	Request To Send
16	RD (A)	Receive Data
17	RD (B)	Receive Data
18	---	---
19	---	---
20	SD (A)	Send Data
21	SD (B)	Send Data
22	TT (A)	Terminal Timing (TT)
23	TT (B)	Terminal Timing (TT)
24	---	---
25	RC	Receive Common
26	---	---

Note: All unlisted signals are not supported by the board.

2.9.1.1 Connector J1 Configurations

When you receive the iSBC 186/03 board from the factory, Connector J1 operates as a DCE device with an RS422A/449 interface. As such, socket U4 contains the DIP header (DCE mode configuration), U3 contains a 3487 device, and RP2 is socketed with pin-1 closest to the serial interface connector. The interface can be converted to one of the following operating modes.

- RS232C DCE (Data Communications Equipment) mode operation
- RS232C DTE (Data Terminal Equipment) mode operation
- Multidrop application, RS422A/449 DCE

The following paragraphs provide a configuration sequence for each of these operating modes. In making a choice between the types of operation, you must make some trade-offs either in cost, performance, or ease of use.

Refer to Table 2-17 when choosing between RS232C or RS422A/449 interface operation on Connector J1. The table provides a comparison of the advantages and disadvantages of both the RS232C and the RS422A/449 interfaces.

Consider your application requirements when deciding on either Data Terminal Equipment (DTE) or Data Communications Equipment (DCE) mode operation for Connector J1.

You can select DTE mode if the board is to operate as a terminal device and select DCE mode if the board is to operate as a processor device. The only difference between DTE and DCE operation is the physical location of the transmit signals and the receive signals on the interface connector.

Table 2-17. Serial I/O Interface Configuration Comparison

Feature Description	Comparison	
	RS232C	RS422/449
Driver Receiver Type	Single Ended	Differential
Number of signal lines required to enact the interface protocol	N	2N
Communication cable lengths	50 feet (max.) @ 19.2K baud; longer for slower rates.	Up to 4000 feet
Voltages required to operate the interface	+12 volts -12 volts ground	+5 volts ground

2.9.1.2 RS232C DCE Configuration Sequence

To reconfigure the J1 interface from the as-shipped operation to RS232C DCE operation, perform the following sequence of four steps. With the iSBC 186/03 board in the default configuration, proceed as follows:

1) Move the DIP header from socket U4 to U1. Socket U4 should now be empty.

2) Ensure that the DIP header is configured for DCE operation, as shown in Figure 2-10.

3) Remove the 3487 device from IC socket U3. Socket U3 should be empty.

4) Remove RP2, turn it 180 degrees, and reinstall it in the same SIP socket. At this point, you should have RP2 installed onto the board so that pin-1 is closest to the P1 MULTIBUS connector on the board. The effect of rotating RP2 is to bias the lines to 1.4 V as shown in Figure 2-11.

2.9.1.3 RS232C DTE Configuration Sequence

To reconfigure the J1 interface from the default operation to RS232C DTE operation, perform the following sequence of four steps:

1) Remove the DIP header from socket U4.

2) Provide another DIP header that is configured for DTE operation, as shown in Figure 2-10. Install it into U1.

3) Remove the 3487 device from IC socket U3. Socket U3 should be empty.

4) Remove RP2, turn it 180 degrees, and reinstall it in the same SIP socket. Install RP2 so that pin-1 is closest to the P1 MULTIBUS connector on the board.

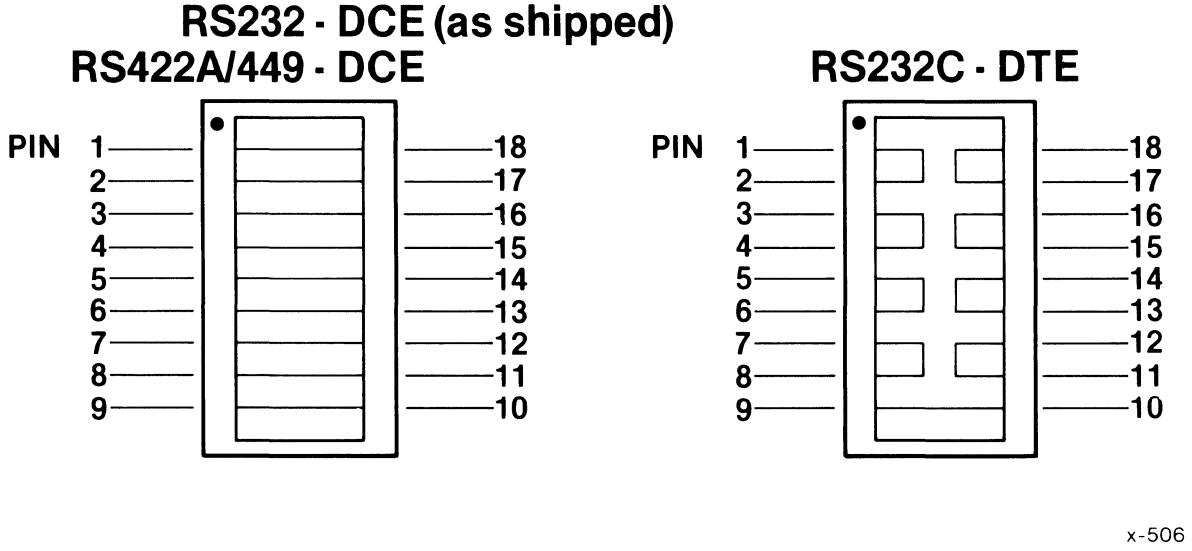


Figure 2-9. DIP Header for DCE and DTE Operation

2.9.1.4 Multidrop Configuration (RS422A/449 Interface Only)

When configured as an RS422A/449 interface, the J1 serial interface can be used in a Multidrop environment. This means that multiple transmitters can co-exist on the same serial lines. To implement this option, only one driver is permitted to drive the lines at any one time, and all other drivers must be tri-stated. The iSBC 186/03 board allows you to enable or disable the RS 422A/449 driver on channel A by controlling it with the channel A DTR signal from the 8274.

Begin the modification with the iSBC 186/03 board in the default configuration. From there, reconfigure the J1 interface for Multidrop operation by adding a jumper at E38-E39. Refer to Appendix B at the end of this manual to figure the amount of termination resistance required in RP1, RP2 and RP3.

2.9.1.5 Bias and Termination Resistor Requirements

The three resistor packs (RP1, RP2, and RP3) provide you with some configuration control for terminating an RS422A/449 interface application. Refer to Appendix B for more information. The options are as follows:

1. RP3 provides signal line termination. For long-line RS422A/449 applications, install RP3, whether the system is a multidrop or a point-to-point network. The total termination resistance value should be as close as possible to the 100 ohm characteristic impedance of the RS422A cable. Refer to Appendix B for instructions on calculating a resistor value.

   Depending on the physical configuration of your network (i.e., stub length, location of receivers/transmitters) termination may be required at both ends of the cable. If so, a 100 ohm terminator at each end of the cable would overload the transmitter. Therefore, split the termination between both end with two 200 ohm resistors.

2. RP1/RP2 provides pull-up/pull-down biasing for the signal lines. You must install RP1/RP2 onto only one receiver on each line in an RS422A/449 interface in a multidrop application. Refer to Section 3.5.3 for details on using RP1/RP2 in RS422/449 and RS232C applications.

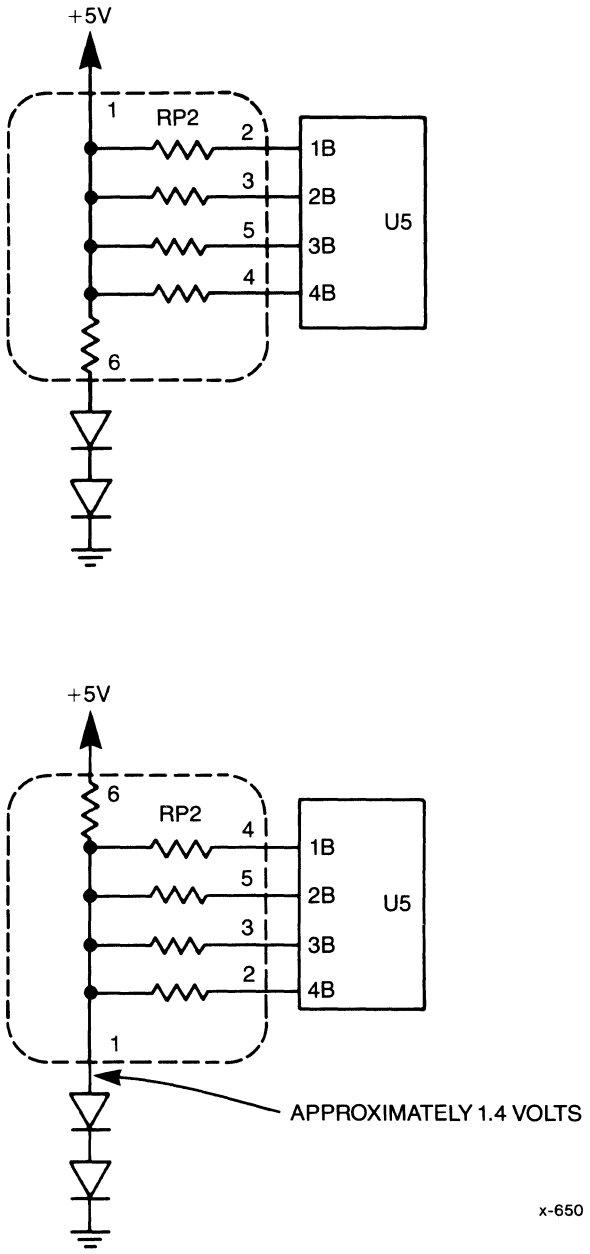


Figure 2-10. RP2 Configurations for RS442A/449 and RS232C

2.9.2 CONNECTOR J2 - RS232C INTERFACE - CHANNEL B

When you receive the iSBC 186/03 board from the factory, Connector J2 operates as a Data Communications Equipment (DCE) device with an RS232C interface.

2.9.2.1 Connector J2 Description

Table 2-18 provides the pin assignments for Connector J2 and lists the the RS232C interface signals on each pin of the connector. Figure 2-9 shows a cable drawing of an RS232C cable for connector J2.

Table 2-18. RS232C Pin Assignment

J1/J2 Pin	RS232C Pin Number	RS232C Signal Name	RS232C Signal Function
1	14	---	---
2	1	---	---
3	15	---	---
5	24	DTE TxC	Transmit Clock
9	18	---	---
11	19	---	---
12 ++	8	DCD	Data Carrier Detect
13	20	DTR	Data Terminal Ready
14	7	SGND	Signal Ground
15	21	---	---
16	6	DSR	Data Set Ready
17	22	---	---
18	5	CTS	Clear To Send
19	17	RxC	Receive Clock
20	4	RTS	Request To Send
22	3	RxD	Receive Data (from iSBC® 186/03)
23	25	---	---
24	2	TxD	Transmit Data (from iSBC® 186/03)
26	13	---	---

Notes: All unlisted signals are not supported by the board.

++ Not supported on J2.

2.9.2.2 Connector J2 Configurations

When you receive the iSBC 186/03 board from the factory, Connector J2 operates as a DCE device with an RS232C interface. This interface does not provide you with configuration options.

2.10 iSBX™ BUS INTERFACE

The iSBC 186/03 board contains two 36/44-pin iSBX bus connectors for either 8- or 16-bit I/O expansion. The connectors are labeled J6 and J7 on the board. The following paragraphs provide a description of the interfaces, and jumper configuring information.

2.10.1 iSBX™ BUS DESCRIPTION

The iSBX bus interfaces on the iSBC 186/03 board provide the following level of compliance with the Intel iSBX Bus Specification:

iSBX Bus Specification Compliance: D16/16 DMA

- The "DMA" means that the board supports DMA operations to iSBX bus connectors.
- The "D16/16" means support for either an 8-bit or a 16-bit expansion module.

Refer to the INTEL iSBX BUS SPECIFICATION for a description of the pin assignments, AC characteristics, and DC characteristics of the J6 and J7 connector interfaces on the iSBC 186/03 board.

2.10.2 iSBX™ BUS INTERFACE CONFIGURATIONS

The iSBX bus interface configuration is controlled by jumper options and programming options. These options are listed below.

- Select between 8 or 16-bit module. Table 2-19 outlines the selections. The I/O addresses used by each interface depends on whether an 8 or 16-bit module is installed. The I/O addresses for each, and the proper iSBX Chip Select lines are shown in Chapter 4. The actual functions of these addresses are described in the particular iSBX board hardware reference manual.

- DMA operations. DMA may be performed to each iSBX bus interface. The iSBX Bus Specification states that a DMA data read operation or write operation must be acknowledged (to the MULTIMODULE board) by generating an MDACK* signal. This is done on the iSBC 186/03 board by accessing the data port of the iSBX board at the MDACK* address shown in Chapter 4, Table 4-4. An optional MDACK* signal is provided to connect to either iSBX bus connector via the iSBX Option line. This allows the iSBC 186/03 board to support two DMA data ports on one iSBX bus module.

- Option Lines. These lines are provided for passing general purpose signals between the iSBX bus board and the iSBC 186/03 board.

2.10.2.1 iSBX™ Bus I/O Addresses

The iSBC 186/03 board reserves all I/O port addresses in the range of 0080H through 00BFH as on-board addresses for the iSBX bus connectors.

Port addresses for the connectors are provided in Chapter 4, along with the other board I/O addresses. In the default configuration the board supports 8-bit iSBX boards.

2.10.2.2 iSBX™ Bus Jumper Configurations

The iSBC 186/03 board contains 4 jumper posts (E129, E130, E168, and E170). These jumper posts provide access to the two option signals (OPT0 and OPT1) for each iSBX bus connector. Table 2-20 lists the jumper connections on the iSBC 186/03 board.

Because the iSBC 186/03 board does not support the MULTIMODULE present signal the iSBX I/O address space is always dedicated to the iSBX interfaces. Jumpers are provided to select an 8 or 16-bit iSBX board size as shown in Table 2-19. The iSBX bus interface supports the TDMA signal (terminate DMA) as a receiver or a driver and supports the iSBX DMA request and acknowledge signals for DMA operation with the iSBX. When TDMA is received from the iSBX board via post E91 or E93, it must be jumpered to the input of one of the inverters in the interrupt matrix (via E75 or E96). The output of the inverter (E71 or E76) can then be connected to an interrupt input. An interrupt will occur on the trailing edge of TDMA with the interrupt controller input in edge-triggered mode. Four option pins are associated with the iSBX interface as shown in Table 2-20. Refer to Section 2.6.1.3 for more information.

Table 2-19. iSBX™ Bus Size Selection

Interface Configuration	Jumpers	
	E48-E47	E49-E50
iSBX 1 8 Bit J7	Out	---
iSBX 1 16 Bit J7	In	---
iSBX 2 8 Bit J6	---	Out
iSBX 2 16 Bit J6	---	In

Table 2-20. iSBX™ Bus Interface Options

Stake Pin	Function
E129	iSBX 1 Option 0
E130	iSBX 1 Option 1
E168	iSBX 2 Option 0
E170	iSBX 2 Option 1
E178-E179	10MHz MCLK (With optional oscillator)
E169	iSBX Optional MDACK*

2.10.2.3 iSBX™ Bus Clock Configurations

As shipped, MCLK frequency is 6MHz. This violates the iSBX Bus Specification. Some iSBX bus boards require the 10MHz MCLK clock for proper operation. A list of Intel iSBX Bus MULTIMODULE boards that require a 10MHz MCLK signal from the baseboard is provided here. If you plan to use one of these MULTIMODULE boards on the iSBC 186/03 board, install the optional 10MHz oscillator at Y1.

iSBX Boards Which Require 10MHz MCLK

iSBX 311 Analog Input Board
iSBX 270 Video Controller Board
iSBX 217 Tape Interface Board

The oscillator should be one of those listed (or an equivalent) in Chapter 3. If you install this optional oscillator, jumper E179-E180 must be removed and jumper E178-179 installed.

NOTE

This modification also changes iSBC 186/03 board BCLK/ & CCLK/ signals to 10MHz.

Another jumper option is provided to allow the 10MHz frequency to be routed to MCLK from CCLK* on the MULTIBUS interface. If another MULTIBUS board in your system generates a 10MHz CCLK* signal you can connect it to MCLK on the iSBC 186/03 board. This is done by removing jumper E220-E221 and installing jumper E221-E222. Table 2-21 shows a list of available clock jumpers on the iSBC 186/03 board.

Table 2-21. iSBX™ Bus Clock Options

Jumper	Description
E179-E180	6MHz clock to MCLK, BCLK and CCLK drivers.
E178-E179	MCLK, CCLK, BLCK, from user-installed oscillator Y1.
E221-E222	MCLK* and BCLK* derived from MULTIBUS generated CCLK*.

2.11 MULTIBUS® INTERFACE

The iSBC 186/03 board contains an 86-pin MULTIBUS connector (P1) for interfacing to other system functions. The MULTIBUS interface on the iSBC 186/03 board is compatible with the MULTIBUS interface standards as presented in the Intel MULTIBUS Specification. The following sections describe the interface jumper options.

2.11.1 MULTIBUS® INTERFACE DESCRIPTION

The MULTIBUS interface on the iSBC 186/03 board provides a level of compliance with the IEEE 796 MICROCOMPUTER SYSTEMS BUS STANDARD as follows:

IEEE 796 Compliance: MASTER D16 M24 I16 EL V0
This translates as follows:

- The "MASTER" means that the board supports master operation on the MULTIBUS interface.
- The "D16" means either an 8- or 16-bit data path.
- The "M24" means a 24-bit memory address path.
- The "I16" means either an 8- or 16-bit I/O address path.
- The "EL" means that the board supports edge triggered or level triggered interrupt sensing.
- The "V0" means that the board supports non-bus vectored interrupt requests and does not support bus vectored interrupt requests.

Refer to the INTEL MULTIBUS SPECIFICATION for a description of the pin assignments, AC characteristics, and DC characteristics of the P1 Connector interface on the iSBC 186/03 board.

2.11.2 MULTIBUS® INTERFACE CONFIGURATIONS

Configuring the MULTIBUS interface on the iSBC 186/03 board consists of changing jumper connections for various bus clock selections and bus arbitration schemes. The following sections describe these options.

2.11.2.1 Jumper Configurations

Installing or removing the jumpers listed in Table 2-22, controls four features of the MULTIBUS interface: the MULTIBUS lock, the bus clock and constant clock (BCLK* and CCLK*), the bus priority resolution, and the bus exchange (arbitration) signals.

Table 2-22. MULTIBUS® Interface Jumpers

Jumper	Signal Name	Description
E218-E219 §	BCLK* Bus clock	Provides a common bus clock from the iSBC 186/03 board to all MULTIBUS boards.
E220-E221 §	CCLK* Constant clock	Provides a common Constant clock from the iSBC 186/03 board to all MULTIBUS boards (CCLK* is 180 degrees out of phase with BCLK*).
E221-E222	CCLK* Constant clock	Allows the BCLK* and iSBX MCLK signals to be derived from the MULTIBUS CCLK* signal. Clocks will be in phase, if installed.
E216-E217 §	LOCK* Bus Lock	Allows the iSBC 186/03 board to generate a LOCK* signal to the MULTIBUS interface. This allows consecutive, unarbitrated access to shared memory.
E210-E214 §	BPRO* Bus Priority Out	Active only when the master is passing control of the MULTIBUS interface to another board. BPRO* indicates to lower priority bus masters that a higher priority device is not requesting the bus (useful only in serial priority resolution schemes and must be removed in parallel priority schemes).
E212-E215 §	CBRQ*	Refer to Table 2-23 and associated explanation.
E209-E213	ARQT	Refer to Table 2-23 and associated explanation.

Note: The § identifies the default configuration.

In the default configuration, the frequency for the BCLK* and CCLK* signals is 6 MHz. This does not conform to the MULTIBUS specification for CCLK (10 MHz). However, you can install a 10 MHz oscillator in socket Y1 if required by your particular application. If required, you must remove jumper between E179 to E180 and install a jumper between stake pins E178 to E179. This jumper allows the iSBC 186/03 board to source the two clocks (BCLK* and CCLK*) that provide the timing for bus arbitration and synchronization logic on other master boards on the MULTIBUS interface. By removing jumpers E218-E219 and E220-E221, you allow another master board to provide the system timing (BCLK* and CCLK*) signals on the MULTIBUS interface. Ensure, however, that you have one and only one device driving each MULTIBUS interface clock signal.

The bus exchange jumpers (E209, E213, E212, E211, and E215) allow you to define the conditions under which the iSBC 186/03 board releases control of the MULTIBUS interface to another bus master.

Table 2-23 lists the conditions for the CBRQ* and ARQT signals in three different interface states. In states 1 and 2, the iSBC 186/03 board receives the CBRQ* signal from the MULTIBUS interface. The CBRQ* signal could be either HIGH or LOW, depending on other MULTIBUS masters. In state 3, on the other hand, the CBRQ* signal on the iSBC 186/03 board is always LOW (jumper E211 - E212 installed).

A socket is provided at RP13 for the installation of a termination SIP resistor pack for MULTIBUS address lines ADR14* through ADR17*. These are used when the 16M byte address mode is enabled. Refer to Section 3.7 for more information.

Table 2-23. MULTIBUS® Interface Arbitration Jumpers

Interface State	Jumper Connect	CBRQ* State	ANYRQST State	Description
1	E212-E215 § E209-E213	LOW	LOW	The Bus Arbiter that has control of the MULTIBUS interface retains control unless a higher priority master deactivates BPRN* or unless the next machine cycle does not require the use of the MULTIBUS interface. It may then relinquish control to a lower priority device.
		HIGH	LOW	The Bus Arbiter that has control of the MULTIBUS interface retains control until another Bus Arbiter pulls CBRQ* low. When CBRQ* goes low, the conditions are as described above.

Table 2-23. MULTIBUS® Interface Arbitration Jumpers (continued)

Interface State	Jumper Connect	CBRQ* State	ANYRQST State	Description
2	E212-E215 § E209-nc §	LOW	HIGH	The Bus Arbiter that controls the MULTIBUS interface surrenders control of the bus, regardless of its priority, upon completion of the current bus cycle.
		HIGH	HIGH	The Bus Arbiter controlling the MULTIBUS interface retains control until another Bus Arbiter pulls CBRQ* low. When CBRQ* goes low, the conditions are as described above.
3	E211-E212 E209-nc §	LOW	HIGH	The Bus Arbiter controlling the MULTIBUS interface surrenders the use of the MULTIBUS interface after each transfer cycle.

Note: The § identifies the as-shipped configuration of each jumper.

2.12 FRONT PANEL INTERFACE - CONNECTOR J4

The front panel interface on the iSBC 186/03 board (Connector J4) provides a 14-pin connector that routes the auxiliary signals required to provide battery backup for memory (MPRO*, PFSN*, PFIN*), the non-volatile enable signal for use with NVRAMs, and key signals necessary for typical front panel control.

The options available for the interface at connector J4 on the iSBC 186/03 board are as follows:

- Use it as it is to provide a front panel interface, and battery backup power and control for the Bank A memory devices on the board.
- Modify jumpers on the board to move certain signals back to the P2 Connector, providing an auxiliary interface at connector P2.

Each of these topics is discussed in greater detail following the description of the interface.

2.12.1 FRONT PANEL INTERFACE DESCRIPTION

The front panel interface is a collection of 7 signals, plus +5 volt battery power, and ground signals on the J4 connector. Table 2-24 lists the pin assignments for the J4 connector. Table 2-25 presents the DC current specifications for the six signals on the connector. Figure 2-13 shows the physical pin-out for connector J4.

Use a 14-pin connector such as the Berg # 65043-030 for interfacing to Connector J4. When installing the connector, ensure that the pins of the mating connectors match the functions shown in Table 2-24 and Figures 2-8 and 2-13. Refer to the INTEL iLBX BUS SPECIFICATION for more information on the front panel interface connector.

Table 2-24. Connector J4 Pin Assignments

Pin Number	Signal	Pin Number	Signal
1	+5 Volt Battery	2	Ground
3	+5 Volt Battery	4	Ground
5	MPRO*	6	NVEN*
7	ALE	8	Ground
9	AUX RESET*	10	Ground
11	EXT INTR*	12	Reserved
13	PFSN*	14	PFIN*

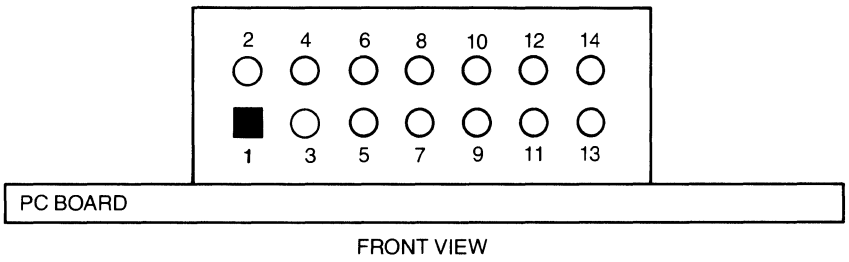


Figure 2-11. Connector J4 Pinout Map

Table 2-25. Connector J4 Interface DC Characteristics

Signal	Parameter/Conditions	Minimum	Maximum	Units
MPRO*	Vih	2.0	---	V
	Vil	---	0.8	V
	Iih @ 2.4V	---	40.0	uA
	Iil @ 0.4V	---	-1.6	mA
PFSN*	Vih	2.0	---	V
	Vil	---	0.8	V
	Iih @ 2.4V	---	40.0	uA
	Iil @ 0.4V	---	-1.6	mA

Table 2-25. Connector J4 Interface DC Characteristics (continued)

Signal	Parameter/Conditions	Minimum	Maximum	Units
PFIN*	Vih	2.0	---	V
	Vil	---	0.8	V
	Iih @ 2.4V	---	40.0	uA
	Iil @ 0.4V	---	-1.6	mA
AUX RESET*	Vih	(note 1)	---	
	Vil	---	0.8	V
	Iih	---	(note 1)	
	Iil @ 0.4V	---	-2.0	mA
EXT INTR*	Vih	2.0	---	V
	Vil	---	0.8	V
	Iih @ 2.4V	---	40.0	uA
	Iil @ 0.4V	---	-1.6	mA
ALE	Voh @ Ioh = -1uA	2.4	---	V
	Vol @ Iol = 16mA	---	0.5	V

Note: 1 The source for AUX RESET* must be an open-collector driver or a front panel switch such that the signal "floats" to the "high" state.

2.12.2 FRONT PANEL CONFIGURATIONS

Use of the front panel interface on connector J4 provides certain signals that are no longer routed to the P2 connector because of the iLBX bus configuration. Alternatively, the P2 interface can be reconfigured to provide the traditional auxiliary interface at Connector P2. Refer to paragraph 2.13.1 for information on reconfiguring the P2 Connector.

2.12.3 BATTERY BACKUP CONFIGURATIONS

Battery backup power and control can be provided to the Bank A memory sockets on by removing jumper E236-E237. This isolates part of the +5 volt power bus that provides power to the Bank A memory devices, and allows the +5 volt battery to power Bank A. Refer to Section 2.3.4.4 for additional battery backup information.

CAUTION

When you provide a source for +5 volt (battery) power on pin-1 and pin-3 of Connector J4, remember to also provide ground on pin-2 and pin-4 from the same battery source. Failure to do so could cause ground-shifts.

2.13 CONNECTOR P2 INTERFACE

The P2 Connector on the iSBC 186/03 board provides you with two basic choices for use: either as an iLBX bus interface (the default configuration) or as an auxiliary connector (requires 3 jumper changes). The iLBX bus interface is discussed in section 2.5. The following section discusses the auxiliary interface.

2.13.1 AUXILIARY INTERFACE DESCRIPTION

As an auxiliary interface, the P2 Connector operates compatibly with the P2 Connector on other boards designed without the iLBX bus interface.

You reconfigure Connector P2 for operation as an auxiliary interface by removing wire jumper E240-E241 and E163-E164. Then install jumpers E161-E162 and E60-E61. These jumper changes disable the iLBX bus transceivers, place the AUX RESET signal from P2-38 on the iSBC 186/03 board and place the ALE signal from the iSBC 186/03 board onto P2-32.

Note that the ALE and AUX RESET signals are still available on J4 after the reconfiguration. When operating as an auxiliary connector, the P2 interface provides DC characteristics for the AUX RESET and ALE signals which are the same as on connector J4. (Refer to Table 2-25). However, the ALE output drive current must now source both connectors J4 and P2.

***

CHAPTER 3. INSTALLATION INFORMATION

3.1 INTRODUCTION

The purpose of this chapter is to help you prepare for installation of the board and to provide you with a preview of equipment and components that you may need to order and use with your board. The chapter includes information on unpacking and inspecting the board when it arrives and on preparing the environmental conditions for the board.

3.2 UNPACKING AND INSPECTING YOUR BOARD

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 are damaged, keep the carton and packing material for the agent's inspection.

You must do two things before Intel can perform any repair of a product damaged in shipment. First, contact the Intel Product Service Marketing Administrator to obtain a return authorization number and further instructions (procedures are listed in Chapter 5). Second, submit to the Intel Product Service Marketing Administrator the number of a purchase order to which the repair can be charged. You should also submit a copy of the purchase order to the carrier with your claim.

3.3 PREPARING YOUR BOARD ENVIRONMENT

The iSBC 186/03 board has several specific requirements regarding the power, cooling, and physical space within the environment of your application. Those requirements are outlined in the following sections.

3.3.1 POWER REQUIREMENTS

The iSBC 186/03 board contains power distribution that satisfies the requirements of the board; the board draws all required power (except +5VB for battery backup which is connected via J4) from the MULTIBUS interface. As a maximum, you must provide four voltage levels (+5 volts, +5 volt battery, -12 volts, +12 volts) and ground, as follows:

- +5 volt power source for all configurations; and battery backup if required.
- Ground for all configurations.
- +12 volt power source for configurations requiring an RS232C interface or possibly a MULTIMODULE board.
- -12 volt power source for configurations requiring an RS232C interface or possibly a MULTIMODULE board.

The iSBC 186/03 board requires a specific minimum amount of current sourcing capability at each power source, depending on the following system configuration factors:

1. The type of user-supplied memory devices on the board.
2. The quantity of user-supplied memory devices on the board, including the iSBC 341 Memory Expansion board.
3. The current required by any installed iSBX bus boards.
4. The type of serial interface at Connector J1 on the board; either RS422A/449 or RS232C.

Table 1-1 lists the current requirements for each voltage required for the board, without memory or optional iSBX bus boards. Add the incremental current for each iSBX bus module and for each memory device installed.

3.3.2 COOLING REQUIREMENTS

The iSBC 186/03 board dissipates 818.5 gram-calories of heat per minute (3.3 BTU per minute). To dissipate this heat and prevent possible heat damage to the board, you must provide adequate air circulation to prevent the ambient air around the board from rising above 55°C (131°F). A minimum air flow of 200 linear feet per minute provides enough air circulation to maintain the air temperature around the iSBC 186/03 board within this limit.

3.3.3 CONNECTOR AND CABLE PARTS

You must provide some connector parts for the iSBC 186/03 board when using it in certain applications. Figure 3-1 shows the approximate location of the connectors on the iSBC 186/03 board, including:

- Two 26-pin pin-and-socket connectors (J1 and J2)
- A 50-pin pin-and-socket connector (J3)
- A 14-pin front panel interface connector (J4)

- Two 36- or 44-pin iSBX Bus connectors (J6 and J7)
- An 86-pin MULTIBUS interface connector (P1)
- A 60-pin iLBX bus interface connector (P2)

Table 3-1 provides you with a list of connector parts that are ready-made to interface to each of these connectors. The table also provides the names of the manufacturers and part numbers by which you may order the connectors. You may substitute any electrically and mechanically equivalent parts.

Table 3-2 provides information on cables that are compatible with the connector parts listed in Table 3-1 for the parallel I/O interface at Connector J3 and for the serial I/O interfaces at Connectors J1 and J2.

Table 3-1. iSBC® 186/03 Board Connector List

Function	# of Pins	Centers Inches	Connector Type	Vendor	Part Number
MULTIBUS Connector (P1)	86	0.156	Soldered[1] PC Board Mount	VIKING ELFAB	2KH43/9AMK12 BS1562D43PBB
			Wirewrap[2] Without Ears	EDAC ELFAB	337086540201 BW1562D43PBB
			Wirewrap[2] With .128 Dia Mounting Holes	EDAC ELFAB	337086540202 BW1562A43PBB
iLBX Bus Connector (P2)	60	0.1	Solder	KELAM	RF30-2853-542
Auxiliary Connector (P2)		0.1	Soldered	ELFAB EDAC	97169001 34060524300
			Wirewrap[2] No Ears	ELFAB EDAC	BW1020D30PBB 345060540201
			Wirewrap[2]	ELFAB EDAC	BS1020A30PBB 345060524802
			Wirewrap[2] (.128 Dia.)	TI Viking	H421121-30 3KH30/9JNK

Table 3-1. iSBC® 186/03 Board Connector List (continued)

Function	# of Pins	Centers Inches	Connector Type	Vendor	Part Number
iSBX Bus Connector 8-bit (J6, J7)	36	0.1	Soldered (male)	Viking	000292-0001
iSBX Bus Connector 16-bit (J6, J7)	44	0.1	Soldered	Viking	000293-0001
Parallel Connector (J3)	50	0.1	Flat Crimp w/strain rel. Flat Crimp	3M 3M T&B Ansley	3425-6000 3425-6050 609-5001M
Serial I/O Connector (J1,J2)	26	0.1	Flat Crimp Flat Crimp	3M AMP	3452-0001 88106-1
Front Panel Connector (J4)	14	0.5	Soldered	BERG	65043-030

Notes: Connector heights are not guaranteed to conform to Intel packaging equipment.

Wirewrap pin lengths are not guaranteed to conform to Intel packaging equipment.

Table 3-2. iSBC® 186/03 Board Connector Cable Information

Interface Type	Mode[2]	Board Connector	Cable	Interface Connector
RS232C	DTE	26-pin[4] 3M 3452-0001 or AMP 88106-1	3M[3]-3349/25	25-pin[5], 3M-3482-1000
RS232C	DCE	26-pin[4] 3M 3452-0001 or AMP 88106-1	3M[3]-3349/25	25-pin[5], 3M-3483-1000
RS449	DCE	26-pin[4] 3M 3452-0001 or AMP 88106-1	3M-3349/25	37-pin[1], 3M-3503-1000

Notes:
1. Cable housing 3M-3485-4000 may be used with the connector.
2. DTE=Data Terminal Equipment mode (male connector). DCE=Data Communication Equipment mode (female connector).
3. Cable is tapered at one end to fit the 3M-3462 connector.
4. Pin 26 of the edge connector is not connected to the flat cable.
5. May be used with the cable housing 3M-3485-1000.

3.3.4 COMPONENTS REQUIRED

Table 3-3 lists all of the optional components that may be required on the iSBC 186/03 board. The list includes memory devices, bias and terminator resistor packs, an oscillator, and PAL devices. Install only those components required to satisfy the application.

When installing the integrated circuit packages into the sockets on the iSBC 186/03 board, ensure pin 1 of the device matches pin 1 of the socket, unless otherwise noted. Figure 3-1 shows the locations of the user-furnished devices; the figure also shows the device orientation and pin 1 location of key components.

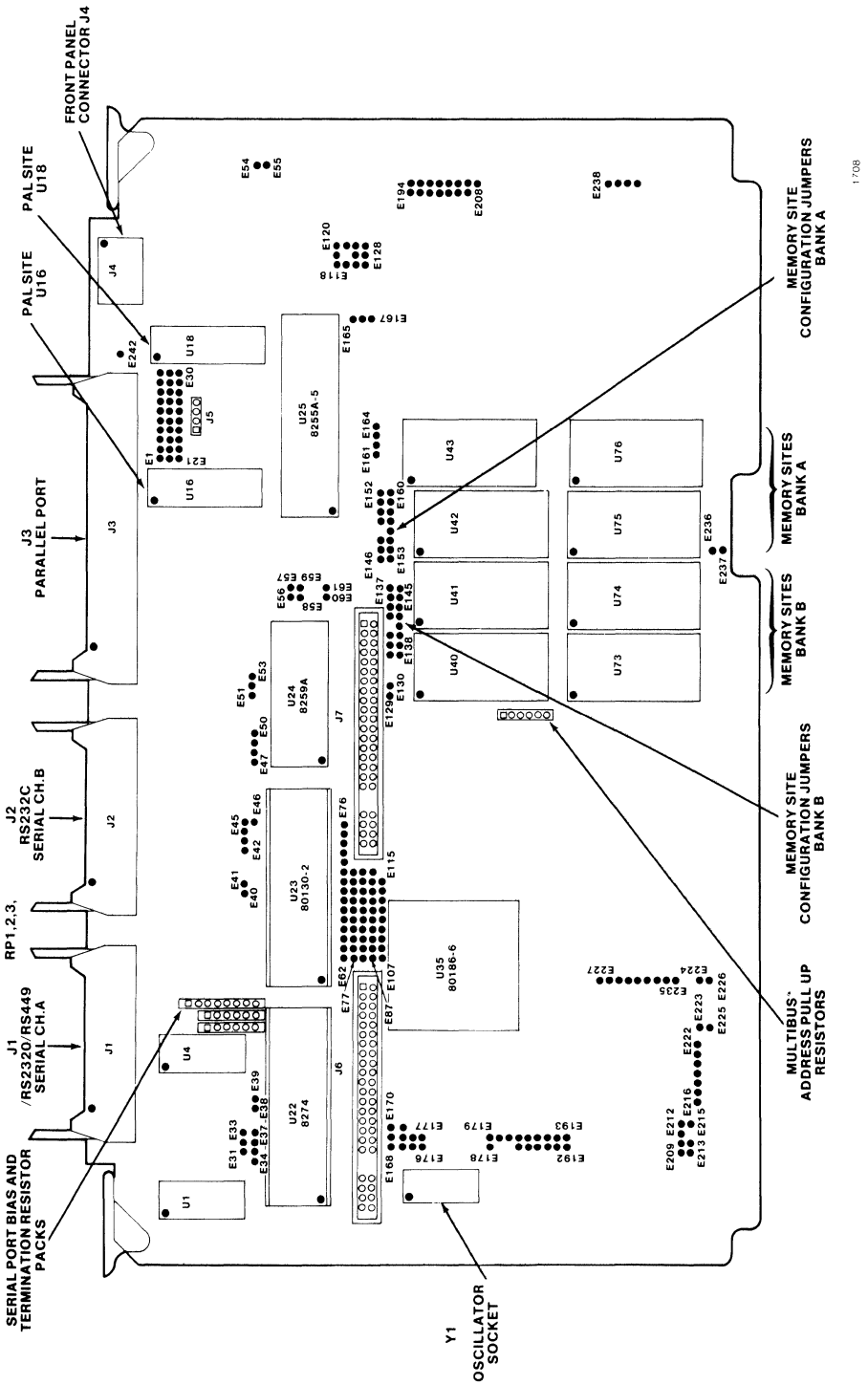


Figure 3-1. User-Furnished Components on the iSBC® 186/03 Board

Table 3-3. User-Furnished Components

	Part	Description
1	Memory	The byte-wide memory sockets accept installation of EPROM, EEPROM, iRAM, NVRAM, and Static RAM devices with restrictions, as identified in Section 2.3.
2	Termination Resistor Pack (RP3)	Allows termination of the differential serial lines when the iSBC 186/03 board is used in an RS422A/449 application. Refer to Chapter 2.
3	Bias Resistor Pack (RP1,RP2)	Change bias resistor packs at RP1 and RP2 whenever using the iSBC 186/03 board in a Multidrop RS422A/449 or RS232C application.
4	Programmable Array Logic Chip (U59)	Change the PAL when you require memory configurations beyond those provided by U59. The PAL is a 20L8A device by Monolithic Memories, Inc., or an equivalent. The default PAL source code equations are provided in Appendix G.
5	Programmable Array Logic Chip (U18)	Install the PAL when configured for SCSI interface, the Centronics printer interface; or use for any custom interface configurations. Refer to Appendix C.
6	Programmable Array Logic Chip (U16)	Install the PAL when using the SCSI interface See Appendix C; also can be used for custom applications.
7	Y1 Oscillator	Some applications require adding a 10MHz oscillator at site Y1 to support certain MCLK, BCLK*, and CCLK* requirements. Refer to Sections 2.10.2.3 and 2.11.2.1.
8	RP13	If using MULTIBUS address lines ADR14-17, terminate these lines by installing a 6-pin, pull-up resistor pack at RP13. See Section 3.7.
9	80130 or 80150	The 80130 component (U23) is installed in a socket, allowing replacement with an 80150 component if using the CP/M-86* Operating System. *CP/M-86 is a trademark of Digital Research Inc.

3.4 COMPATIBLE EQUIPMENT

The iSBC 186/03 board is designed to operate as a MULTIBUS master, with other single board computer products on the MULTIBUS interface. As such, the iSBC 186/03 board provides a MULTIBUS interface that is fully compatible with IEEE MICROCOMPUTER SYSTEM BUS STANDARD 796.

The memory configuration on the iSBC 186/03 board is such that it allows you to add an iSBC 341 Memory Expansion MULTIMODULE Board at chip sockets U43 and U76.

The iSBX bus interfaces on the board are compatible with either 8-bit or 16-bit MULTIMODULE boards. The layout of the iSBC 186/03 board provides compatibility with all iSBX MULTIMODULE boards.

CAUTION

Certain iSBX Boards require a faster MCLK rate than is standard on the iSBC 186/03 board. An oscillator can be installed at Y1 for this purpose. Refer to Chapter 2 for details.

The iLBX bus interface on the iSBC 186/03 board is compatible with the iLBX family of products; operation is specified in the INTEL iLBX BUS SPECIFICATION. Refer to Section 2.5 for details.

3.5 COMPONENT INSTALLATION

The following sections provide instructions for installing the user-provided memory devices and bias/termination resistors. Refer to Figure 3-1 for an approximate location for installation of each component.

3.5.1 MEMORY COMPONENTS

When shipped from the factory, the iSBC 186/03 board contains no memory components. You may install one of several different types of memory component, including EPROM, EEPROM, iRAM, Static RAM, or NVRAM devices onto the memory sockets on the iSBC 186/03 board. Figure 3-2 shows how to install either 24-pin or 28-pin memory chips into the JEDEC-compatible sockets. Table 3-4 lists the types of compatible memory components. Appendix D shows the required electrical and mechanical specifications for these devices. You may substitiute any electrically and mechanically equivalent components. Refer to Chapter 2 for instructions on configuring the memory addresses at each socket.

CAUTION

Never insert components into a board when power is applied. Doing so could cause damage to the components.

All MOS components such as ROM, EPROM, and RAM devices are highly succeptible to damage from static electricity. Use extreme caution when installing MOS components in a low humidity environment. Always ground yourself before handling MOS components. This ensures that a static charge build-up is not dissipated through or around the MOS devices.

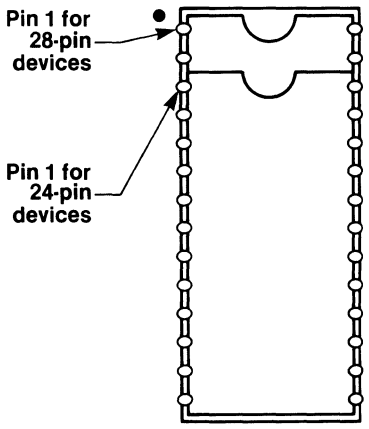


Figure 3-2. Memory Installation

NOTE

The iSBC 186/03 board is designed to accept most byte-wide components. However, typical system operation requires EPROM devices in sockets U41 and U74 to hold the bootstrap program.

Table 3-4. Compatible Memory Devices

Memory Type	Memory Capacity	Allowable Locations		
		Bank A	Bank B	iSBC® 341
Static RAM	2K X 8	yes	no	yes
	8K X 8	yes	no	yes
iRAM	8K X 8	yes	no	yes
EPROM				
	8K X 8	yes	yes [1]	yes
	16K X 8	yes	yes [1]	yes
	32K X 8	yes	yes [1]	yes
EEPROM[3]	2K X 8	yes	yes	yes
NVRAM	512 X 8	yes	no	yes

Notes:
1. Only when Bank A sites are moved to upper memory for use as PROM sites.
2. Allowable mixtures given in Chapter 2.
3. Intel 2817A component only.

NOTE

The iSBC 186/03 board is designed to accommodate 24 or 28-pin memory chips in the same socket. The 24-pin integrated circuits must be installed, as shown in Figure 3-2, with pin 1 of the integrated circuit lined up with pin 3 of the socket.

3.5.2 LINE DRIVER DEVICES

When shipped from the factory, the iSBC 186/03 board contains the line driver and line receiver devices required for an RS422A/449 interface at Connector J1 and an RS232C interface at Connector J2. You can modify the Connector J1 interface for RS232C operation; however, you cannot modify the interface at Connector J2.

If you require an RS422A/449 interface at Connector J1, then use the board in the as-shipped configuration; socket U3 contains a 3487 device, socket U4 contains the header, and socket U1 is empty.

If you require an RS232C interface at Connector J1, then remove the 3487 IC at socket U3 and ensure that the header-chip is moved from socket U4 to U1. The header configuration differs for DTE or DCE mode. See Chapter 2.

The factory installed line driver at location U7 and receiver at U8 on the board configure the serial interface at J2 for operation as an RS232C interface. Modifications are not allowed for this interface.

3.5.3 RESISTOR PACKS

The iSBC 186/03 board contains two socketed resistor packs (RP1 and RP2) that perform biasing and one empty socket for user-installation of another resistor pack (RP3) that performs termination of RS422A/449 differential lines.

The functions of each resistor pack depend on the operating mode of the interface at Connector J1, as follows:

- RP3 — TERMINATION - If using Connector J1 of the iSBC 186/03 board in an RS422A/449 multidrop or point to point network, and if the board is the slave at the end of a line in the network, install a resistor pack at RP3 to properly terminate the signal lines. Refer to Appendix B for details on how to calculate termination requirements.

  If using Connector J1 on the iSBC 186/03 board in an RS232C interface application, do not install a resistor pack into RP3.

- RP2 — BIAS - If using Connector J1 on the iSBC 186/03 board as an RS422A/449 interface in a multidrop network, install the resistor pack at RP2 with pin-1 closest to the J1 edge-connector (the as-shipped configuration). RP2 maintains voltage levels on the signal lines, when the lines are not driven. Refer to Appendix B for details on how to choose the proper resistor value for RP2. Note that RP2 should be installed on only one board in a multidrop network line. RP2 can also be installed in an RS422 point-to-point network, but is not required.

If using Connector J1 on the iSBC 186/03 board as an RS232C interface, then install the 2.2K ohm resistor pack at RP2 with pin-1 closest to the MULTIBUS connector (180 degree rotation from the as-shipped configuration; i.e., pin 1 of R-pack to pin 6 of socket). Refer to Section 2.9.

- RP1 — BIAS - If using Connector J1 on the iSBC 186/03 board in an RS422A/449 multidrop network, install the resistor pack at RP1 (the as-shipped configuration). RP1 maintains voltage levels on the signal lines when the lines are not driven. Note that RP1 should be installed on only one board in a multidrop network line. RP1 can also be installed in an RS422 point-to-point network, but is not required.

  If using Connector J1 on the iSBC 186/03 board in an RS232C application, install the 2.2K ohm resistor pack at RP1 to maintain voltage levels on the signal lines.

The following sections provide more information on the as-shipped configuration of each R-pack and on calculating a resistor value.

3.5.3.1 Bias Resistors

When shipped from the factory, the iSBC 186/03 board contains two 6-pin sockets with 2.2k ohm bias resistor packs (RP1 and RP2). These resistor packs provide bias for the Connector J1 interface. At shipment, the SIP devices are installed so that pin-1 of each SIP is in pin-1 of each SIP socket. This configuration allows RP2 to operate as pull-up resistors and RP1 to operate as pull-down resistors for the RS422A/449 interface at Connector J1.

By removing RP2, rotating it 180 degrees, and reinstalling it so that pin-1 is closest to the MULTIBUS connector on the board (so that pin-1 of the pack is in pin-6 of the socket), you configure the board to bias the four non-inverting inputs to U5 near 1.4 volts. This is done to allow the RS232C interface to use a differential receiver.

By changing the value of the resistor pack, you can change the bias at the serial interface receivers. Appendix B shows how to calculate the bias resistor values for your RS422A/449 application. Figure 3-1 shows the approximate location of the RP1 and RP2 bias resistor packs on the iSBC 186/03 board.

3.5.3.2 Termination Resistors

The iSBC 186/03 board provides you with a place to install the optional termination resistor pack, if you need it. When shipped from the factory, the iSBC 186/03 board provides an empty socket at RP3 for a single in-line resistor pack (SIP).

Termination resistors are required only for RS422A/449 interface applications; do not install a termination resistor at RP3 if operating an RS232C interface. Termination resistors provide you with a means of reducing the amount of signal noise between the lines of a differential pair. If the iSBC 186/03 board is the unit at the end of a long line in a multidrop or point-to-point network, you should install an 8-pin termination resistor SIP into socket RP3.

The value of the termination resistor and the quantity used depends on the configuration of the network. Appendix B provides guidelines for choosing the termination resistor.

3.6 P2 CONNECTOR INSTALLATION

The P2 connector can be used either for the iLBX bus interface connector, or the MULTIBUS Auxiliary interface connector. It cannot be installed for both interfaces at one time. The connectors have differing pin assignments and differing keys installed. Incorrect installation will cause the board and system to malfunction. The following two sections discuss each type of interface connector.

3.6.1 iLBX™ BUS CONNECTOR INSTALLATION

You expand the local memory on the iSBC 186/03 board by cabling it to iLBX memory boards via iLBX bus connectors on P2. Most system backplanes do not include iLBX bus connectors and cables. As a result, before connecting iLBX bus memory to the iSBC 186/03 board, install the iLBX bus connectors and cable into the backplane.

First, build a connector assembly for the iLBX bus interface (refer to the INTEL iLBX BUS SPECIFICATION for cable requirements).

Install a key between pins 41 and 43 on the iLBX bus connector. The iSBC 186/03 board has a key slot between these pins. This allows the board to be installed in an iLBX bus backplane. It also prevents the board from being installed in a non-iLBX bus system containing a key between pins 15 and 17.

Then, install the connectors into the cardcage as shown in the iLBX BUS SPECIFICATION. Pin 1 of the connectors (usually identified by the colored band on the cable), should be closest to the MULTIBUS connector.

Finally, install the iSBC 186/03 board into the cardcage, ensuring that the P1 Connector seats into the MULTIBUS connector and that the P2 Connector seats into an iLBX bus connector.

Install your iLBX bus memory boards into the other cardslots containing iLBX bus connectors.

CAUTION

Do not install any board which is not configured for an iLBX bus interface into an iLBX bus connector.

Do not install or remove iLBX bus interface boards while power is applied to the system.

Failure to adhere to either of these precautions could result in damage to the boards.

3.6.2 AUXILIARY CONNECTOR INSTALLATION

The P2 connector interface can be reconfigured from the iLBX bus interface to an auxiliary interface as described in section 2.13.1. Auxiliary connectors for the iSBC 186/03 board are listed in Table 3-1. These are installed similarly to the iLBX bus connectors, described above. However, because the iSBC 186/03 board is configured for an iLBX bus interface at connector P2, it has a keyslot between pins 41 and 43. Note that this arrangement does not prevent the insertion of a reconfigured (from iLBX bus to Auxiliary) board into an iLBX bus slot.

3.7 MULTIBUS® ADDRESS TERMINATION RESISTORS

A SIP resistor socket is included on the iSBC 186/03 board for terminating MULTIBUS address lines ADR14* through ADR17*. If you implement 24-bit addressing, and no other termination is provided for these address lines, install a SIP resistor pack in RP13 to provide a total of 2.2K ohms of pull-up resistance. For this total, you must include any pull-up resistance already provided by other MULTIBUS memory boards in your system.

3.8 OPTIONAL 10MHz OSCILLATOR

As shipped, the MCLK and CCLK* frequency is 6MHz. This violates the iSBX specification for MCLK and the MULTIBUS specification for CCLK*. Some iSBX Bus boards (see Chapter 2) require a 10MHz clock for proper operation. For these boards a 10MHz oscillator can be installed in location Y1. The oscillator should be one of those listed on the following page. If you install this optional oscillator, jumper E179-E180 must be removed and jumper E178-179 installed.

Oscillator Vendor	Part Number, @ 10MHz
CTS Knights Inc.	MXO-55A series
Monitor Products Co.	970 series
M-TRON	MTO series
MF Electronics	M1115 series

Refer to Figure 3-1 for the pin orientation of Y1.

3.9 PAL DEVICES

Refer to Appendix C for information on PALs associated with the parallel port interface. Appendix G provides the source code equations for the default 20L8A Memory Decode PAL (U59). These equations can be modified for another application and a new PAL programmed with the modified equations.

***

CHAPTER 4. PROGRAMMING INFORMATION

4.1 INTRODUCTION

This chapter provides programming information for specific iSBC 186/03 board functions. Memory addressing locations and I/O addressing information are given in table form, for quick reference. Most of the programmable devices on the iSBC 186/03 board can be programmed as described in their respective data sheets. If the programming techniques differ from the data sheet, if certain programming modes are required, or if a function is board-only, it will be described in this chapter.

4.2 iAPX 186 PROCESSOR INITIALIZATION

After a hardware reset or power-on sequence, the 80186 processor requires several internal registers be programmed for specialized use on the iSBC 186/03 board. The following registers must be programmed:

1. Program the relocation register to set the base address for all internal registers to I/O address FF00H, and for iRMX 86 compatibility mode. The ET (escape trap) bit can be set to cause a trap interrupt on an ESCAPE instruction, or the bit can be cleared.

2. The upper memory chip select (UMCS) register must be programmed to select 0 wait-states using external ready. This allows the selection of wait-states for PROM memory to be determined by the on-board jumpers.

3. The PACS register must be programmed so that PCS0 I/O chip select starts at I/O address 0H, with 0 wait-states, using external ready.

4. The LMCS, MPCS or MMCS registers do not need to be programmed for the iSBC 186/03 board.

Table 4-7 provides the I/O addresses of the 80186 internal I/O registers. The internal peripherals (such as DMA controllers and the interrupt controller) do not require any additional initialization until actual mode programming. However, the 80130 component requires an initialization sequence. Refer to the 80130 Initialization section for more information on this device.

Appendix F provides an example of an initialization program for the iSBC 186/03 board. Refer to the comments in the program for more information.

4.2.1 iRMX™ 86 COMPATIBILITY MODE

The interrupt controller in the 80186 was designed to run in the iRMX 86 compatibility mode. Deviation from this mode is not recommended. This mode assigns master status to the 80130 interrupt controller and slave status to the iAPX 186 interrupt controller. On reset, the iAPX 186 interrupt controller will be in the non-iRMX 86 mode. To set the controller to the iRMX 86 compatibility mode, bit 14 of the Relocation Register (board I/O address FFFEH) should be set. Refer to the iAPX 186 data sheet for additional information on iRMX 86 compatibility mode.

4.2.2 INTERNAL PERIPHERAL INTERFACE

All the iAPX 186 integrated peripherals are controlled via 16-bit registers contained within an internal 256-byte control block. For the iSBC 186/03 board you must specify a base address for the peripheral chip select lines of 0H in the relocation register (FEH). On-board I/O address will then be reserved from 0080H to 00FFH (using the PCS1 chip select signal). Bit 12 of the relocation register must be 0, placing the control block in I/O space. In addition, the upper four bits of the base address must be programmed as 0 since the I/O address is only 16-bits wide. Refer to the "Internal Peripheral Interace" section of the iAPX 186 data sheet for more information.

4.3 MEMORY ADDRESSING

Table 4-1 outlines the memory addressing associated with each memory device socket on the iSBC 186/03 board. Off board local memory (iLBX bus memory addressing is covered in Table 4-2. MULTIBUS (system) memory addressing is shown in Figure 4-1.

Table 4-1. Socket Pair Address Ranges

Socket		BANK B (ROM) (4 sockets)+			
ODD (upper)	EVEN (lower)	Address Size Per Socket (Kbytes)			
		8KB	16KB	32KB	64KB
U41	U74	FC000-FFFFF	F8000-FFFFF	F0000-FFFFF	E0000-FFFFF
U40	U73	F8000-FBFFF	F0000-F7FFF	E0000-EFFFF	C0000-DFFFF
		BANK A (RAM) (4 sockets)++			
Socket		2KB		8KB	
U42	U75	00000-00FFF		00000-03FFF	
U43	U76	01000-01FFF		04000-07FFF	
		BANK A (ROM) (4 sockets)+++			
Socket		8KB	16KB	32KB	64KB
U42	U75	F0000-F3FFF	E0000-E7FFF	C0000-CFFFF	A0000-BFFFF
U43	U76	F4000-F7FFF	E8000-EFFFF	D0000-DFFFF	80000-9FFFF
iSBC 341 Socket		iSBC 341 (4 sockets) if Installed			
		2KB		8KB	
U2	U5	02000-02FFF		08000-0BFFF	
U3	U6	03000-03FFF		0C000-0FFFF	

Notes: If the 80130 address-space is enabled, it resides in the 16K-byte immediately below the PROM bank address range. See table 2-2. If 64K X 8 devices are installed and the 80130 is enabled, the 16K address will overlap the 16M byte window address range. In this case, the 16M byte window cannot be enabled.

++ When Bank A is used for RAM address space starting at 0H.

+++ When the Bank A sites are decoded as upper memory for use as ROM sites. In this mode the four Bank A sites must have the same size devices as Bank B and the 80130 address (if enabled) is addressed as the 16KB immediately below the Bank A address. See Table 2-3.

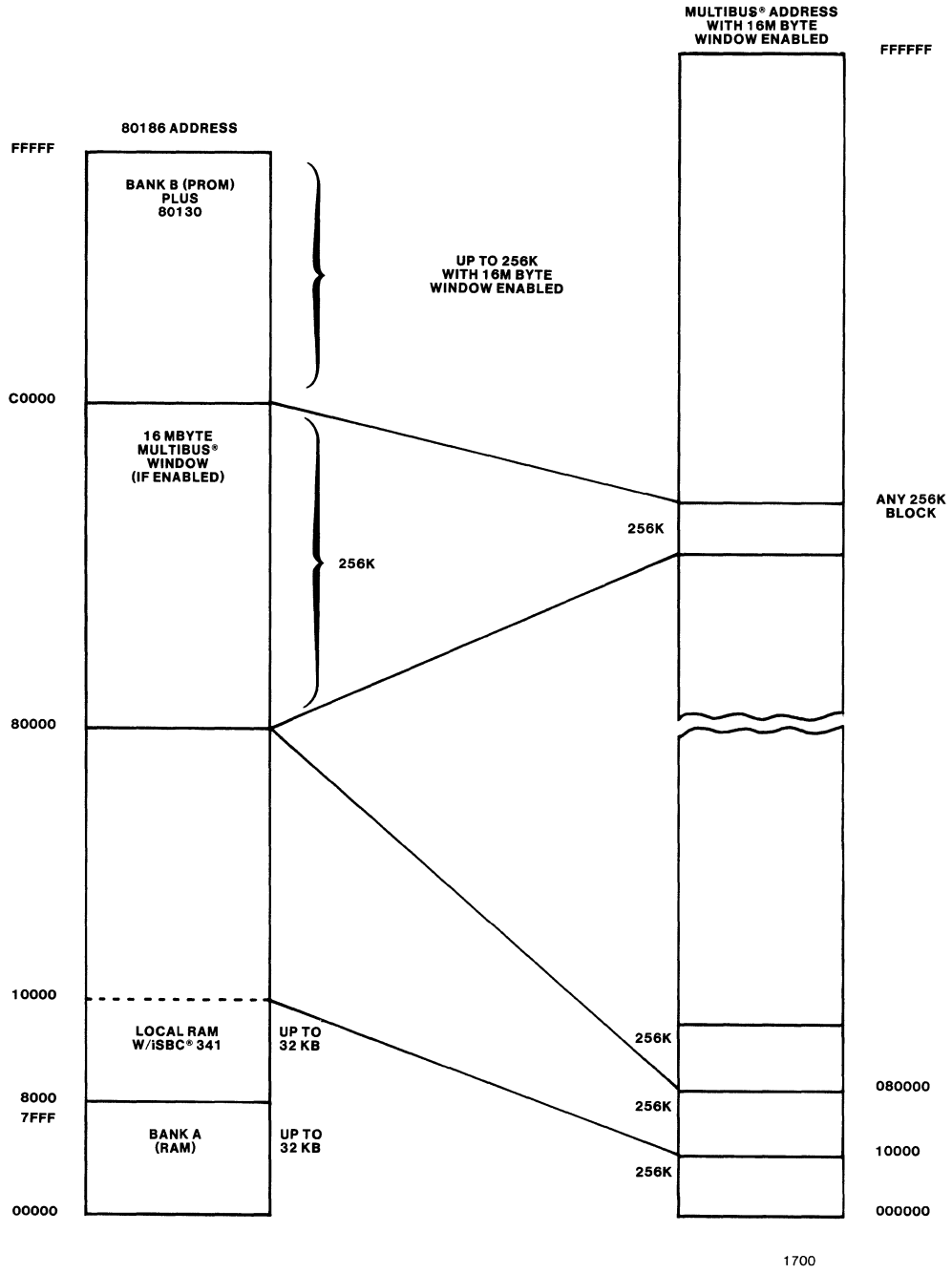


Figure 4-1. iSBC® 186/03 Memory Map

4.3.1 iLBX™ BUS MEMORY

The iLBX bus address space can be expanded to a maximum of 896 K-bytes as shown in Table 4-2. The performance between the 80186 CPU and the iLBX bus memory is comparable to that between the 80186 CPU and the on-board local memory.

The iSBC 186/03 board is configured to use the P2 connector for the iLBX bus interface. In this mode the default address of the iLBX bus memory begins immediately above the Bank A (RAM) memory, at address 02000H, and ends at 07FFFFH. If Bank A is disabled via jumpers E198 through E203 or is moved into upper memory, the iLBX bus address starts at address 0H. The size of the iLBX bus memory space can be expanded by modifying the jumpers as shown in Table 4-2.

Table 4-2. iLBX™ Bus Address Ranges

Jumpers		Address Range +
E44-E45	E42-E43	
Out	Out	00000H-DFFFFH (896KB)++
Out	In	00000H-BFFFFH (768KB)++
In	Out (Default)	00000H-7FFFFH (512KB)
In	In	Not Allowed

Notes: + = In order to address the iLBX bus beginning at 0H, the decode PAL allows the four RAM BANK sockets to be moved to support PROM address space. See Table 2-1 for allowable PROM/EPROM sizes. If Bank A starts at 0H, then iLBX bus memory begins immediately above Bank A memory.

++ = Selecting iLBX options above 512KB disallows the use of the MULTIBUS 16MB window addressing.

If your application does not need to use the iLBX bus, you can remove jumper E240-E241 and E 163-E164. Doing this reassigns the memory space normally assigned to the iLBX bus memory to the MULTIBUS memory. It also tri-states the iLBX bus address, data, and control drivers from the P2 connector to allow the connector to be used for a non-iLBX bus interface.

4.3.2 16M-BYTE MULTIBUS® MEMORY ADDRESSING

The MULTIBUS system memory can be accessed as a 1 M-byte address space or as a 16 M-byte address space. When operating as a 16 M-byte master, the iSBC 186/03 board implements the 16 M-byte address space as sixty-four 256 KByte segments which do not overlap. Six latched data bits select which one of the 64 segments is to be accessed. The 18 lower MULTIBUS address lines determine the particular memory location within the 256 KByte segment which is being accessed. The six latched address lines are appended to the lower 18 lines from the 80186 to create the 24-bit address lines.

To latch the upper six address bits, an I/O write operation to the 16M-byte window latch (I/O address 00D0H) is performed with data bit (D7) in the most significant address location. (Address bit ADR17*) as shown in Figure 4-2.

In this figure, the 16M-byte window latch has been loaded with the value 101001XX binary. When a memory access is made to the 16M-byte window (80000H to BFFFFH) the latched bits are driven to MULTIBUS address lines ADR17* through ADR12*, replacing the upper two address bits of the original 20-bit address generated by the 80186. In this example, given these latched data bits, a memory write to address 83FFFH will actually be routed to MULTIBUS address A43FFH. This is shown in the figure as the upper six bits equal to 101001 binary.

If the 16 MB enable jumper is installed, accesses by the 80186 to the 256 KByte address space of 80000H - BFFFFH will be MULTIBUS memory accesses with the six latched address bits as the upper MULTIBUS addresses.

4.4 I/O ADDRESSING

In general, input/output (I/O) addressing refers to the means by which a programmer must communicate and control the on-board programmable devices and latches. The iSBC 186/03 board is built around a wealth of such devices and features. Therefore, it is crucial that the programmer study each of the I/O functions before attempting to operate this board.

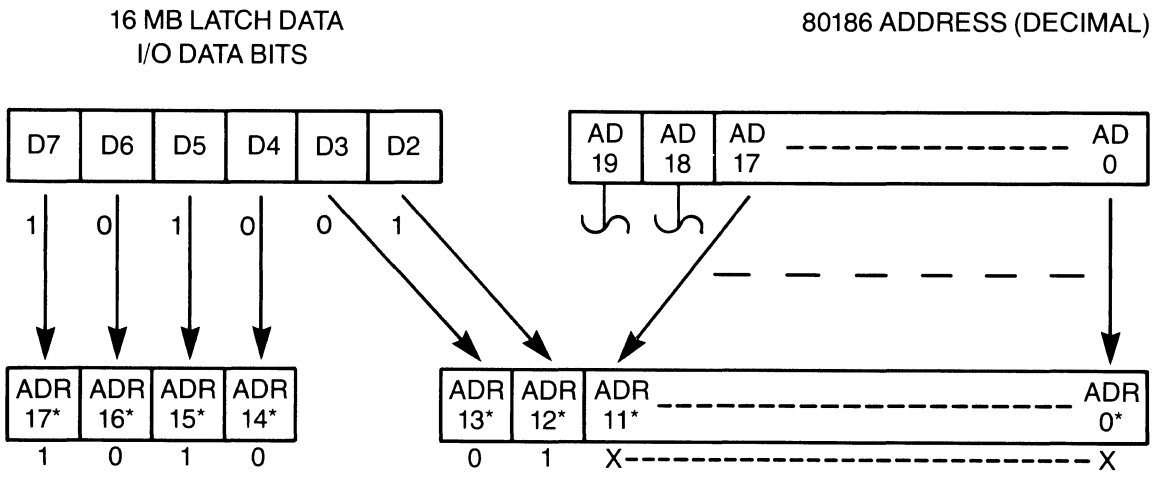
The following sections discuss I/O addressing of the serial channel controllers, parallel I/O functions, timers, DMA functions, interrupt controllers and the iSBX bus interface functions.

Table 4-3 provides I/O port addresses for all on-board I/O functions except the 80186 internal registers. Table 4-7 provides the I/O addresses for all internal 80186 I/O registers.

4.4.1 iSBX™ BUS I/O ADDRESSING

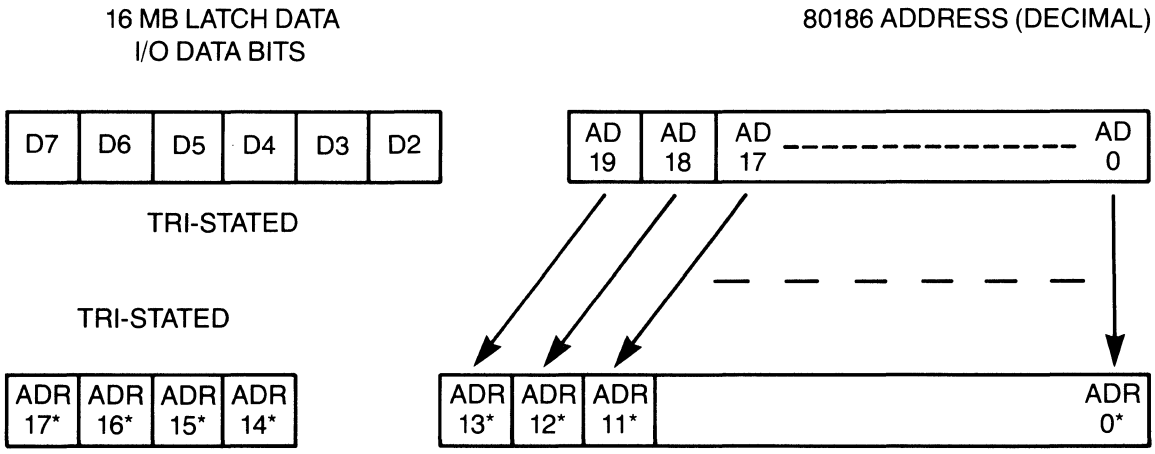
Each iSBX bus interface contains two separate chip select signals for addressing functions on the iSBX board. The address for each chip select on each iSBX connector depends on the bus width (8-bit or 16-bit) of the installed iSBX board. Table 4-4 outlines these signals and their associated I/O addresses for each iSBX connector.

16 MB ADDRESS ENABLED



x-652

1MB ADDRESS ENABLED



x-653

Figure 4-2. 16M-Byte and 1M-Byte MULTIBUS® Addressing

Table 4-3. Local I/O Port Addresses

Hex Address	Device	Note
0080-008E (Even)	SBX 1 (J7)	Byte access: 8 or 16 bit SBX, MCS0 active Word access: 16 bit SBX, MCS0 and MCS1 active
0081-008F (odd)	SBX 1	Byte access: 16 bit SBX, MCS1 active Word access: N/A
0090-009E (Even)	SBX 1	Byte access: 8 bit SBX, MCS1 active
00A0-00AE (Even)	SBX 2 (J6)	Byte access: 8 or 16 bit SBX, MCS0 active Word access: 16 bit SBX, MCS0 and MCS1 active
00A1-00AF (Odd)	SBX 2	Byte access: 16 bit SBX, MCS1 active Word access: N/A
00B0-00BE (Even)	SBX 2	Byte access: 8 bit SBX, MCS1 active Word access: N/A
00C0-00C2	8259A	Byte access only: Slave interrupt controller
00C8	8255A	Byte: Port A Data - Word: N/A
00CA	8255A	Byte: Port B Data - Word: N/A
00CC	8255A	Byte: Port C Data - Word: N/A
00CE	8255A	Byte: Control port - Word: N/A
00D0	16M Window Latch	Byte: Write only - Word: N/A
00D2	SBX Opt MDACK*	Byte or Word: Optional MDACK*
00D4	SBX 1 MDACK*	Byte or Word: MDACK*
00D6	SBX 2 MDACK*	Byte or Word: MDACK*
00D8	8274	Byte: CH A data - Word: N/A
00DA	8274	Byte: CH B data - Word: N/A
00DC	8274	Byte: CH A Control/Status - Word: N/A
00DE	8274	Byte: CH B Control/Status - Word: N/A
00E0-00E2	80130	Byte access only: Master Interrupt Controller
00E8	80130	Byte access only: System Timer
00EA	80130	Byte access only: Delay Counter
00EC	80130	Byte access only: Baud Rate Timer
00EE	80130	Byte access only: Timer Control

Table 4-3. Local I/O Port Addresses (continued)

Hex Address	Device	Note
00F0-F1	DRQ MUX PAL Sel 0	Write Only, Even Byte, Sel 0=0, Word: N/A Odd Byte, Sel 0=1
00F2-F3	DRQ MUX PAL Sel 1	Write Only, Even Byte, Sel 1=0, Word: N/A Odd Byte, Sel 1=1
00F4-F5	DRQ MUX PAL Sel 2	Write Only, Even Byte, Sel 2=0, Word: N/A Odd Byte, Sel 2=1
00F6	DRQ MUX PAL TDMA	Byte, TDMA1=active; Word: N/A
00F7-FF	Reserved	

Table 4-4. iSBX™ Bus Chip Select Signals

I/O Port Address (Hex) (00XX)	Active Chip 16-Bit	Select 8-Bit	iSBX™ Bus Connector
80-8E Even Byte	MCS0*	MCS0*	J7
Word	MCS0* & MCS1*	----	J7
81-8F Odd Byte	MCS1*	----	J7
90-9E Even Byte	----	MCS1*	J7
A0-AE Even Byte	MCS0*	MCS0*	J6
Word	MCS0* & MCS1*	----	J6
A1-AF Odd Byte	MCS1*	----	J6
B0-BE Even Byte	----	MCS1*	J6
D2 Even Byte	Opt. MDACK*	Opt. MDACK*	J6 or J7
Word	Opt. MDACK*	Opt. MDACK*	J6 or J7
D4 Even Byte	SBX1 MDACK*	SBX1 MDACK*	J7
Word	SBX1 MDACK*	SBX1 MDACK*	J7
D6 Even Byte	SBX2 MDACK*	SBX2 MDACK*	J6
Word	SBX2 MDACK*	SBX2 MDACK*	J6
F6	TDMA1	TDMA1	J6 and/or J7 Jumper Selectable

4.5 DMA CONTROLLER

The iSBC 186/03 board provides a DMA controller (integral to the 80186 processor) with two channels. DMA may be performed I/O to I/O, memory to memory, or I/O to and from memory. A memory to memory DMA transfer is a non-synchronized type and appears like a normal read/write cycle with no DMA request inputs. Memory-to-memory DMA transfers are not recommended. This is because a string move is just as fast, and interrupts can be recognized during string moves, while interrupts are not recognized during a DMA cycle. DMA to or from I/O is source or destination synchronized and requires a DMA request signal from the synchronizing device. The devices that may signal for DMA requests are:

User-installed SBX modules
8274 Multiple Protocol Serial Controller (MPSC) device
Parallel Port Interface

There are seven possible sources for the two DMA channels. Therefore, a DMA Multiplexer is provided to allow you to programmatically select a subset of the sources for each DMA channel.

Once the DMA controller is initialized, the DMA operation can be initiated by the synchronizing device activating its DMA request signal (MDRQT, SCSI DRQ, 8274 TxDRQA, etc.).

When performing DMA to an iSBX module installed into one of the iSBX connectors, the data can be read/written to the iSBX MDACK* address (shown in Table 4-4) rather than the iSBX programmed I/O address. This generates the proper DMA acknowledge signal to the iSBX interface.

When performing a DMA operation to the 8274 MPSC device, the 8274 device must be programmed to place the particular channel(s) into the DMA mode. In the DMA mode, both receive and transmit DMA requests are generated for the channel. If you want only one direction connected to the 80186 DMA controller, then the other direction must be connected to an interrupt line. If only one channel is selected for DMA, that channel must be channel A. When either channel is in the interrupt mode, RxDRQB/IPI must be grounded (install a jumper between E51 and E52) to provide the Interrupt Priority Input (IPI) signal.

4.6 DMA PROGRAMMING

Each DMA channel is controlled by a set of programmable registers within the integrated peripheral control block of the 80186 processor. These registers are physically located within the peripheral devices that they control, but are addressed as a single block of registers. These registers include the source and destination pointer registers, the transfer count register and the control register. The programming information is contained in the 80186 data sheet. The DMA multiplexer allows selecting a subset of the seven possible DMA request sources for each DMA channel. Selection is made by jumpers and programming the SELECT bits in the DMA multiplexer.

Tables 4-5, 4-6 and 4-8 provide the jumpers required and the programming address information required to set and clear the three SELECT bits of the multiplexer. Note that the data value written to these addresses is irrelevant.

The DMA channels address the full 1 Mbyte address space with the use of two 16-bit registers (each) for source and destination addresses. When addressing the I/O space the upper four bits of the DMA pointer register should be programmed to zero.

Tables 4-7 provides the I/O port addresses associated with programming the DMA registers. For more information refer to the iAPX 186 data sheet.

Table 4-5. DRQ0 Source Selection

Jumper	DRQU Source	SEL1*	SEL0* (0 = Low)
IN0 = 0 E56 to E58 Installed	SBX 1 MDRQT SBX 2 MDRQT TxDRQB SCSI DRQ	0 0 1 1	0 RESET 1 0 1
IN0 = 1 § E56 to E58 Removed	RxDRQA RxDRQB TxDRQB SCSI DRQ	0 0 1 1	0 RESET 1 0 1
Note: § = Default Jumper.			

Table 4-6. DRQ1 Source Selection

Jumper	DRQ1 Source	SEL2*
IN1 = 0 E57 to E59 installed	SBX 1 MDRQT TxDRQA	0 RESET 1
IN1 = 1 § E57 to E59 removed	TxDRQA RxDRQB	0 RESET 1
Note: All three SELect lines are set to the 0 state upon RESET. § = Default Jumper.		

Table 4-7. Port Addresses for 80186 Internal I/O Registers

I/O Address	Reference	Function Performed
FFCA	DMA 0	DMA0 Control Register
FFC8	DMA 0	DMA0 Count Register
FFC6	DMA 0	DMA0 Destination Register (Upper 4 bits)
FFC4	DMA 0	DMA0 Destination Register (Lower)
FFC2	DMA 0	DMA0 Source Register (Upper 4 bits)
FFC0	DMA 0	DMA0 Source Register (Lower)
FFDA	DMA 1	DMA1 CNTRL
FFD8	DMA 1	DMA1 Count
FFD6	DMA 1	DMA1 Destination Register (Upper 4 bits)
FFD4	DMA 1	DMA1 Destination Register (Lower)
FFD2	DMA 1	DMA1 Source Register (Upper 4 bits)
FFD0	DMA 1	DMA1 Source Register (Lower)
FFFF		Relocation Register
FFA0	UMCS	Upper Memory Chip Select
FFA2	LMCS	Lower Memory Chip Select
FFA8	MPCS	Mid-Range Memory Chip Select
FFA6	MMCS	Mid-Range Memory Chip Select
FFA4	PACS	Peripheral Chip Select
FF56		Timer 0 Control
FF52		Timer 0 Max Count A
FF54		Timer 0 Max Count B
FF50		Timer 0 Count Register
FF5E		Timer 1 Control
FF5C		Timer 1 Max Count A
FF5A		Timer 1 Max Count B
FF58		Timer 1 Count Register
FF66		Timer 2 Control
FF62		Timer 2 Max Count A
FF60		Timer 2 Count Register
Interrupt Control Registers (iRMX 86 Mode)		
FF20		Interrupt Vector Register
FF22		Specific EOI Register
FF28		Mask Register
FF2A		Priority Level Mask Register
FF2C		Interrupt Service Register
FF2E		Interrupt Request Register
FF30		Interrupt Status Register
FF32		Level 0 Control Register (Timer 0)
FF34		Level 2 Control Register (DMA 0)
FF36		Level 3 Control Register (DMA 1)
FF38		Level 4 Control Register (Timer 1)
FF40		Level 5 Control Register (Timer 2)

Table 4-8. DMA Request MUX Addressing

Address (Byte Write Only)	Selected Output
00F0	SEL0* = 0
00F1	SEL0* = 1
00F2	SEL1* = 0
00F3	SEL1* = 1
00F4	SEL2* = 0
00F5	SEL2* = 1
00F6	TDMA 1 pulsed (to iSBX board)
00F7	Reserved

4.7 INTERRUPT CONTROLLER PROGRAMMING

The iSBC 186/03 board provides support for those real-time applications that operate on time of day, external events, or elapsed time with the use of four interrupt controllers. One is integral to the 80186 processor device; another is integral to the 80130 Operating System Firmware (OSF) device; a third is integral to the 8274 Multi-Protocol Serial Controller (MPSC); and the fourth interrupt controller is the 8259A Programmable Interrupt Controller (PIC) device.

The iSBC 186/03 board provides interrupt priority levels through jumper selection and the priority algorithm programmed into the interrupt controller devices. Refer to the data sheets for additional programming information on the individual interrupt controllers.

Each interrupt controller on the iSBC 186/03 board is capable of operating in the vectored interrupt mode. The vectored mode uses the vectoring capabilities of the 8274, 80186, and the 8259A PIC slave devices. In vectored mode, the slave device requesting the interrupt sends an interrupt request signal to the 80130 master interrupt controller, which in turn, passes the interrupt request to the 80186 CPU. The 80130 master PIC does not provide the interrupt vector for a slave request; it allows the slave PIC to give the vector to the 80186 CPU.

The interrupt controllers within the 80186, 8274, and 8259A devices must be initialized as slave devices to the 80130. Table 4-9 lists the port addresses for the 80130 and 8259A controllers and the functions performed by each. Table 4-7 lists the addresses and functions of the 80186 controller. The addresses for the 8274 controller are shown in Table 4-3 and the programming functions are discussed in Section 4.10 and in the 8274 data sheet. Additional information is covered in Intel Application Notes AP-134 and AP-145. The following sections discuss interrupt controller programming.

Table 4-10 and Appendix F lists a programming sequence that initializes the 80130, 8259A, and 80186 interrupt controllers in the iRMX 86 compatibility mode. As such, each interrupt controller is initialized to the vectored mode of operation. For additional details on parameters, refer to the INTEL MICROPROCESSOR AND PERIPHERAL HANDBOOK. Refer to Appendix F for a sample program which initializes the controllers and gives an example of an interrupt service routine when using the master and slave PICs and the 8274.

To avoid unexpected interrupts, you should either mask or ground all unused inputs to the PIC devices.

NOTE

For a slave interrupt, the interrupt service routine must send an EOI command to both the master and the slave controllers.

Table 4-9. Port Addresses for 80130, 8259A, & 8274 Interrupt Functions

I/O Address	Device	Operation	Interrupt Function Performed
00E0H	Master PIC (80130)	Byte Read: Byte Write: Word:	Status and Poll Registers ICW1, OCW2, and OCW3 Parameter N/A
00E2H	Master PIC (80130)	Byte Read: Byte Write: Word:	Mask Register or OCW1 Parameter Mask Register or ICW2, ICW3, ICW4, and OCW1 Parameters N/A
00C0H	Slave PIC (8259A)	Byte Read: Byte Write: Word:	Status and Poll Registers ICW1, OCW2, and OCW3 Parameter N/A
00C2H	Slave PIC (8259A)	Byte Read: Byte Write: Word:	Mask Register or OCW1 Parameter Mask Register or ICW2, ICW3, ICW4, and OCW1 Parameters N/A
00DCH	Slave PIC (8274)	Byte Read: Byte Write: Word:	Channel A Status Channel A WR0, WR1, WR2 (EOI, Enable Interrupts, DMA/Interrupt mode) N/A
00DEH	Slave PIC (8274)	Byte Read: Byte Write: Word:	Channel B Status Channel B WR0, WR1, WR2 (EOI, Enable Interrupts, DMA/Interrupt mode) N/A

Table 4-10. Parameter Sequence For 80130, 8259A, & 8274 Initialization

Sequence	Port/ Data	Functions Performed
1.	00E0H 39H	Port Number of 80130 master PIC. Data; ICW1. This parameter specifies level-triggered operation, cascaded operation, and ICW6 parameter required.
2.	00E2H 20H	Port Number of 80130 master PIC. Data; ICW2. This parameter sets IR0 of the 80130 at Interrupt level type 20H.
3.	00E2H 50H	Port Number of the 80130 master PIC. Data; ICW3 (define slave PICs). On the iSBC 186/03 board, this parameter identifies levels 3, 4 and 6 (IR3, IR4 and IR6) as being connected to slave interrupt controllers supplying a vector.
4.	00E2H 0DH	Port Number for the 80130 master PIC. Data; ICW4. On the iSBC 186/03 board, this parameter selects not-special fully-nested mode, buffered mode, master, normal end of interrupt, and 8086 mode.
5.	00E2H FFH	Port Number for the 80130 master PIC. Data; ICW6. On the iSBC 186/03 board, this parameter must be set to FFH.
6.	00E2H FFH	Port Number for the 80130 master PIC. Data; OCW1. This parameter is the MASK Register format; it masks off all interrupts.
7.	00C2H 19H	Port Number for the 8259A slave PIC. Data; ICW1. This parameter specifies level-triggered operation, cascaded operation, and ICW4 parameter required.
8.	00C2H 28H	Port Number for the 8259A slave PIC. Data; ICW2. This parameter provides T3 through T7 of the interrupt vector (to be multiplied by 4). IR0 of the 8259A is set at interrupt type 28H.
9.	00C2H 06H	Port Number for 8259A slave PIC Data; ICW3 (cascade address). On the iSBC 186/03 board, this parameter provides the slave PIC identification. This will be the same level on the master PIC to which the 8259A is connected.

Table 4-10. Parameter Sequence For 80130, 8259A, & 8274 Initialization (continued)

Sequence	Port/ Data	Functions Performed
10.	00C2H 09H	Port Number for 8259A slave PIC. Data; ICW4. On the iSBC 186/03 board, this parameter selects not-special fully-nested mode, buffered mode, slave, normal end of interrupt, and 8086 mode for the slave PIC.
11.	00C2H FFH	Port Number for 8259A slave PIC. Data; OCW1. This parameter is the MASK Register format; it masks off all interrupts.
12.	FFFEH 40FFH	Port number for relocation register in 80186. Data; Sets iRMX 86 compatibility mode, base register at I/O address FF00H.
13.	FF20H 40H	Port number for Interrupt Vector register in 80186. Data; Sets interrupt type 40H.

4.7.1 80130 MASTER INTERRUPT CONTROLLER

The 80130 device must be programmed as the master interrupt controller. The 80130 must be programmed to indicate the interrupt levels to which the 80186, the 8274, and the 8259A slave controllers are connected (IR4, IR3, IR6 respectively). The 80130 level IR3 is reserved for slave controller connections for the 8274 vectored interrupts only. The 80130 level IR4 is reserved for slave controller connections for the 80186 vectored interrupts only.

If the 8274 is used in non-vectored mode (the 8274 does not provide the interrupt vector), then IR3 to the 80130 must not be used. In this case, the 8274 can be connected to levels IR1, IR2, IR5, or IR6.

The interrupt request connection from the 80186 slave interrupt controller must remain connected to level IR4 of the master interrupt controller; the connection from the slave interrupt controller 8259A may be moved from level IR6 of the master interrupt controller. The 8259A must also be programmed as a slave and in buffered mode. Refer to the 80186, 80130, and 8259A data sheets for programming information.

If the as-shipped jumper configuration is not changed, then the 80130 master PIC should be programmed to support the following levels:

- IR2 is the system time of day timer (SYSTICK) - E67 to E82.

- IR3 is the interrupt request from the serial controller 8274 (E66 to E81) when the 8274 is in the vectored interrupt mode. If a slave controller is not used the interrupt level in the master PIC that is connected to the slave PIC must be masked off.

- IR4 is the interrupt request from the CPU 80186 - E65 to E80.

4.7.2 80186 INTERRUPT CONTROLLER

Initialize the 80186 to operate in the iRMX 86 compatibility mode. This mode is accomplished by setting the iRMX mode bit in the relocation register in the 80186. Refer to the 80186 Data Sheet for interrupt programming information. In the iRMX Compatibility mode, the 80186 interrupt request signal INT3 is sent to the master 80130 PIC (via jumper E65 to E80). The master 80130 PIC, in response, generates the INT0 signal back to the 80186 to interrupt the CPU. Refer to Appendix F for additional programming information.

4.7.3 PROGRAMMING THE 80130 INTERRUPT CONTROLLER

Programming the 80130 interrupt controller is accomplished by accessing the control words in I/O space located at locations E0H and E2H. These registers consist of:

	ADDRESS
Interrupt Request Register	00E0H
In-Service Register	00E0H
Interrupt Mask Register	00E2H
Initialization Control Word 1	00E0H
Initialization Control Word 2	00E2H
Initialization Control Word 3	00E2H
Initialization Control Word 4	00E2H
Initialization Control Word 5	00E2H
Initialization Control Word 6	00E2H
Operation Control Word 1	00E2H
Operation Control Word 2	00E0H
Operation Control Word 3	00E0H

The 80130 accepts two types of command words generated by the CPU:

1) Initialization Command Words (ICW): Before normal operation can begin, the 80130 must be programmed with a sequence of 5 or 6 bytes in succession.

2) Operation Command Words (OCWs): The command words are sent to the 80130 for various forms of operation, such as:

Interrupt Masking
End of Interrupt
Interrupt Status

The OCWs can be written into the 80130 anytime after initialization.

4.7.3.1 80130 Interrupt Initialization (ICWs)

When an I/O write command is issued to address E0H with D4=1 during a write peripheral cycle, the data is interpreted as Initialization Command Word 1 (ICW1). ICW1 starts the initialization sequence during which the following automatically occurs:

a) The edge sense circuits are reset. This means that following initialization, an interrupt request (IR) input must make a low-to-high transition to generate an interrupt unless the IR input is programmed as a level sensitive input in which case a high level will generate the interrupt.

b) The Interrupt mask register is cleared.

c) Status read is set to IRR.

d) The Interrupt Acknowledge cycle is reset and prepared for the first INTA cycle.

e) All Interrupts will be acknowledged with LIR*=1 unless ICW6 is written.

Initialization Command Words 1 and 2

Issuing ICW1, ICW2, ICW3, ICW4, and ICW6 is the minimum amount needed to program the 80130. The remaining control word ICW5 is specified by setting bit 3 in ICW1 if that function is needed. Once ICW1 has been written, the following writes to I/O Address E2H must follow the sequence of ICW2, ICW3, ICW4, ICW5, and ICW6, (except ICW5, if not specified in ICW1). The 80130 is ready to accept Interrupts after the last written ICW in the programming sequence.

Bits 7-3 of ICW2 specifies the vector that is supplied to the CPU upon receipt of the second interrupt acknowledge cycle. The remaining bits, 2-0, are determined by the interrupt level with the 80130. The formats accepted by the 80130 for ICW1 and ICW2 are as follows:

Bits:	76543210	ICW1 Command written to I/O address E0H
	00------	Unused and may be any value
	0---0-	All slaves are non-local; no ICW6 read
	1---0-	ICW6 to specify local slave inputs
	1----	ICW1 indicator
	00--	Edge triggered interrupts, no ICW5
	10--	Level triggered interrupts, no ICW5
	x1--	Level for interrupts to be by ICW5
	1-	No slave units as IR inputs SNGL=1
	0-	One or More slave units as inputs SNGL=0, ICW3 to be read
	1	ICW4 always needed
Bits:	76543210	ICW2 Command written to I/O address E2H
	TTTTT---	Five most significant bits of the vector byte supplied to CPU

x = Unused and may be any value

Initialization Command Word 3

This word is written to indicate that there are slave interrupt controllers on the board and that cascading is used, in which case SNGL=0 in ICW1. This loads the 8-bit slave identification register. A bit is set (1) to indicate the IR level to which a slave interrupt controller is connected. On the iSBC 186/03, the 8274, 8259, and 80186 controllers are slaves and bits 3, 4 and 6 must be set.

Bits:	76543210	ICW3 Command written to I/O address E2H
	1------	IR7 has a slave
	0------	IR7 does not have a slave
	1-----	IR6 has a slave
	0-----	IR6 does not have a slave
	.	
	.	
	.	
	1	IR0 has a slave
	0	IR0 does not have a slave

Intialization Command Word 4

ICW4 is always required by the 80130. The only function this word controls in the 80130 is the selection between Special Fully Nested and Fully Nested modes. The format for ICW4 is as follows:

Bits:	76543210	ICW4 Command written to I/O address E2H
	000-----	Unused
	0----	Not Special Fully Nested Mode
	1----	Special Fully Nested Mode
	1---	Buffered Mode
	1--	Master
	0-	Normal End of Interrupt
	1	8086 Mode

Initialization Command Word 5

This word is written only if ICW5 was specified in ICW1. This word allows the option to individually select edge or level sensitive inputs on the IR inputs. The format is as follows:

Bits:	76543210	ICW5 Command written to I/O address E2H
	1-------	IR7 is level Triggered
	0-------	IR7 is edge Triggered
	.	
	.	
	.	
	1	IR0 is level Triggered
	0	IR0 is edge Triggered

Initialization Command Word 6

This word must be issued to specify that all interrupts are local by setting all the bits in ICW6 to one. An IR input causes the output LIR*=0. If a bit in ICW6 is "0", then an acknowledge of the corresponding IR input will output LIR*=1. The iSBC 186/03 board requires ICW6 to be set to FFH.

Bits:	76543210	ICW6 Command written to I/O address E2H
	1-------	IR7 is a local input
	0-------	IR7 is a non-local input
	.	
	.	
	1	IR0 is a local input
	0	IR0 is a non-local input

4.7.3.2 Operation Command Words (OCWs)

After the Initialization Command Words (ICWs) are programmed into the 80130, the chip is ready to accept interrupt requests at its input lines. However, during the 80130 operation, a selection of algorithms can command the 80130 to operate in various modes through the Operation Command Words (OCWs).

Operation Control Word 1 (OCW1)

OCW1 sets and clears the mask bits in the Interrupt Mask Register (IMR). M7-M0 represent the eight mask bits. M=1 indicates the channel is masked (inhibited), M=0 indicates the channel is enabled.

The format of OCW1 is as follows:

Bits:	76543210	OCW1 Command written to I/O address E2H
	1-------	IR7 is inhibited
	0-------	IR7 is enabled
	1-----	IR6 is inhibited
	0-----	IR6 is enabled
	.	
	.	
	1	IR0 is inhibited
	0	IR0 is enabled

Operation Control Word 2 (OCW2)

This word in the 80130 is used to terminate an interrupt level request. The format for OCW2 is as follows:

Bits:	76543210	OCW2 Command written to I/O address E0H
	011-----	Specific End of Interrupt on EOI
	00---	Select OCW2
	000	End of Interrupt on IR0
	001	End of Interrupt on IR1
	010	End of Interrupt on IR2
	011	End of Interrupt on IR3
	100	End of Interrupt on IR4
	101	End of Interrupt on IR5
	110	End of Interrupt on IR6
	111	End of Interrupt on IR7

4.7.3.3 Interrupt Masks

Each Interrupt Request input can be masked individually by the Interrupt Mask Register (IMR) programmed through OCW1. Each bit in the IMR masks one interrupt channel when set to a one (1). Bit 0 masks IR0, bit 1 masks IR1, and so forth. Masking an IR channel does not affect the other channels operation.

4.7.3.4 End Of Interrupt (EOI)

An In-Service (IS) bit is reset by issuing a command word to the 80130 before returning from a service routine (EOI Command). The EOI command is issued twice, once for the master interrupt controller 80130 and once for the corresponding slave interrupt controller. A Specific End of Interrupt is the only EOI accepted by the 80130 and is also the only designated function of OCW2. Part of this command includes the IS level to be reset.

4.7.3.5 Fully Nested Mode

This mode is entered after initialization unless another mode is programmed. The priority of the interrupt requests range from 0 through 7, where 0 is highest. When an interrupt is acknowledged, the highest priority request is determined and its vector placed on the bus. Additionally, the corresponding bit in the Interrupt Service Register (IS0-7) is set. This bit remains set until an End of Interrupt (EOI) is issued which should occur immediately before returning from the service routine. While the IS bit is set, all further interrupts of the same or lower priority are inhibited. Interrupts request from higher priority levels generate an interrupt.

4.7.3.6 Special Fully Nested Mode

This mode is used when one or more slave interrupt controllers are cascaded to the master interrupt controller (80130) while maintaining priority within each slave. In this case, the master interrupt controller should by programmed for the special fully nested mode using ICW4. This mode is similar to the normal fully nested mode with the following exceptions:

1) When an interrupt request from a slave interrupt controller is being serviced, higher priority interrupt requests from that same slave interrupt controller are recognized by the master interrupt controller (the request is not locked out from the master°s priority logic). Upon receipt of a higher priority interrupt request from the same slave interrupt controller, the master interrupt controller suspends servicing the lower priority interrupt, recognizes the higher priority interrupt request, and initiates an interrupt to the processor.

   In the normal nested mode, a slave is masked out when its request is in service and no higher requests from the same slave can be serviced.

2) When exiting the Interrupt Service routine, the software has to check whether the interrupt serviced was the only one from that slave. This is done by sending a specific End of Interrupt (EOI) command to the slave and then reading its In-Service register and checking for zero. If zero, a specific EOI can be sent to the master too. If not, no EOI should be sent.

4.7.3.7 Edge/Level Triggered Modes

The 80130 has the compatible 8259A group pin selectable Edge or Level triggered mode of operation plus an additional individual pin Edge or Level triggered mode selection. If ICW5 is not specified in ICW1, then all IR inputs are either Edge Triggered or Level Triggered (See ICW1 format). If ICW5 is specified to be read, it contains a mask which determines individual IR input pin mode (See ICW5 format).

In the edge-triggered mode the interrupt controller expects the request input to remain high after the low-to-high edge transition has been observed. Improper vectoring may result unless this level is held until the interrupt has been acknowledged.

4.7.3.8 Local/Non-local Interrupts

The 80130 PIC has a special output, LIR*, which when low, indicates an interrupt acknowledge cycle for an interrupt input with the corresponding bit set in ICW6. LIR* is active for all local slave interrupts. On the iSBC 186/03, all interrupts are local interrupts. The PIC initialization must write ICW6 with the value FFH to activate LIR* for all interrupt levels.

4.7.3.9 Reading The 80130 Interrupt Control Status

You can read the input status of several internal registers to update the user information on the 80130 interrupt system. The following registers are read via OCW3: (with exception of IMR).

Interrupt Request Register (IRR): 8-bit register which contains the levels requesting an interrupt to be acknowledged.

In-Service Register (ISR): 8-bit register which contains the priority levels that are being serviced. The ISR is updated when an End of Interrupt Command is issued.

Interrupt Mask Register (IMR): 8-bit register which contains the interrupt request lines which are masked.

The IRR can be read when, prior to the IORD cycle, a Read Register Command is issued with an OCW3 format:

xxx01xx0B

written to I/O address E0H (Read IRR Register on next read).

The ISR can be read when, prior to the IORD cycle, a READ Register Command is issued with an OCW3 format:

xxx01xx1B

written to I/O address E0H (Read ISR register on next read).

There is no need to write an OCW3 before every status read operation, as long as the status read corresponds with the previous one; such as, the 80130 "remembers" whether the IRR or ISR has been previously selected by the OCW3.

After initialization, the 80130 is set to read the IRR.

For reading the IMR, no OCW3 is needed. The IMR is read when accessing I/O address E2H.

4.7.4 8259A INTERRUPT CONTROLLER

The iSBC 186/03 board operates the 80130 as the master PIC and 80186 and the 8259A as slave PIC. As such, the 8259A must be programmed as a slave PIC, in the buffered, 8086 mode. Any interrupt sensed by the 8259A device (via jumper E63-E78) causes the master PIC to expect an interrupt vector on the bus, from the 8259A.

4.7.5 8274 INTERRUPT CONTROLLER

The interrupt controller in the 8274 is designed to operate in the vectored mode as a slave PIC to the 80130 master controller. On the iSBC 186/03 board, this is also the recommended mode of operation. As such, the 8274 will provide a vector to the 80186 when any of eight internal interrupts are generated. When used in the vectored mode, the 8274 must be connected to the 80130 level IR3 with jumper E66-E81.

If used in the non-vectored mode, the 8274 will provide an interrupt request to the 80130; the 80130 then provides the vector to the 80186 and the CPU must then poll the 8274 status registers to determine the type of interrupt generated by the 8274. In this mode the 8274 must not be connected to 80130 level IR3; instead connect it to IR1, IR2, IR5, or IR6. Level IR7 should not be used for a slave device.

Programming details for the 8274 are provided later in this chapter. In, general, interrupt programming is embedded in the same registers used for programming the serial portion of the 8274.

Write register 0 (WR0) can be programmed to:

1. Reset the external/status interrupts an the Tx Interrupt/DMA pending bit;
2. Enable/disable an interrupt on the next receive character; and
3. Provide an end of interrupt command.

Write register 1 (WR1) can be programmed to:

1. Enable external interrupts and transmit interrupts;
2. Determine the Receive interrupt mode; and
3. Select status affects vector mode (WR1 Channel B only).

Write Register 2 (Channel A) is programmed to:

1. Select between vectored or non-vectored interrupt mode;
2. Select interrupt priorities internal to the 8274; and
3. Select interrupt or DMA mode for each channel (A or B).

Write Register 2 (Channel B) is programmed to provide the interrupt vector base for 8274 interrupt vectors.

Refer to the 8274 programming section later in this chapter, Appendix G, the 8274 data sheet, application note AP-134, and application note AP-145 for additional information.

4.8 TIMERS

There are six timers on the iSBC 186/03 board. Three of the six timers are integral to the 80186 and three are integral to the 80130.

The three 80186 timers can be used to count external events, time external events, or generate non-repetitive waveforms. The third timer is not connected to any external components. This timer can be used as a prescaler to the other two, or as a DMA request source.

Each of the three 80130 timers is restricted to a fixed mode of operation. Timer functions and programming are discussed in subsequent sections.

4.8.1 80186 TIMERS

The clock for the three 80186 timers is a divide-by-four clock from the 6MHz 80186 clock, providing a base clock rate of 1.5MHz.

Timers 0 and 1 can be programmed to provide a square wave output, rate generator output, generate non-repetitive waveforms, interrupt on terminal count, and count external events. These modes are described in the 80186 data sheet and in application note AP-186. A description of the modes and performance is given below.

Timer 2 can be used as a prescaler to timers 0 and 1, a DMA request source or an interrupt source. It cannot be used to count external events.

Timers 0 and 1 are default configured to provide the baud rate clocks for the serial channels. As such, they should be programmed in the square wave mode.

The following table provides a list of 80186 timer operating modes, performance values and required control words.

Table 4-11. 80186 Timer Information

Mode	Mode Word		Min/Max Frequency	
	Single Timer	Prescaled Timer	Single Timer	Prescaled Timer
			Min Max	Min Max
Square Wave	C003H	C00BH	11.49Hz/757KHz	.00018Hz/750KHz
Rate Generator	C001H	C009H	22.99Hz/1.5MHz	.000356Hz/1.5MHz
Interrupt On Terminal Count	E001H or E003H	E009H or E00BH	660ns 43.5us	660ns 48.64min.
Software Triggered Strobe	C002H	C00AH	550us 43.5ms	1.8ms 47.1min.

4.8.2 80130 TIMERS

There are three timers in the 80130. Each of the timers is dedicated to specific function. One is used as the iRMX 86 system clock, one is a delay timer, and the other can be used for baud rate if desired. The latter clock may also be used to disable the Timeout and Timeout Interrupt for a HALT cycle. Refer to Chapter 2 for additional Timeout information.

E32 to E31 routes the output from timer 0 of the 80186 device to the input of Channel A of the 8274 MPSC device.

E171 to E172 routes the output from timer 1 of the 80186 device to the input Channel B of the 8274 MPSC device.

Timer 2 of the 80186 is used internally and is not accessible externally.

4.8.2.1 Description Of 80130 Timer Modes

Each timer can be thought of as consisting of three elements:

1) A count register (CR)
2) A counting element (CE); and
3) An output latch (OL), (only for timers 0 and 1)

The count register (CR) is loaded from the data bus upon the appropriate write cycles and contains the initial value to be used by the counting element (CE) when a counting sequence is initiated.

The CR can be changed while counting is in progress; the effect of writing into the CR while counting is different for the different modes of operation and is explained later.

The output latch (OL) allows for the storing of the contents of the CE for subsequent program interrogation. Storing the CE in the OL simply saves the current value of the CE in the OL; this has no effect on the counting operation currently in progress.

Unless specifically directed by a latch command, the OL follows the CE value. A latch command freezes the contents of the OL. Reading the OL implicitly unlatches the OL, and it once again tracks the CE.

4.8.2.1.1 80130 TIMER 0 OPERATION. Timer 0 is pre-defined to operate in the 8254-compatible Rate Generator mode (Mode 2). In this mode, the output, SYSTICK, will initially be high until the CR is loaded. The first falling edge of the clock after the CR is loaded caused the CR to be transferred to the CE. Subsequent falling edges of the clock cause the CE to count. The output goes low for the clock cycle when CE is equal to 1 and is high for other CE values. The next clock automatically reloads the CE from the CR, and subsequent clocks count.

If the CR is loaded during counting and prior to the time that the CR is automatically transferred to the CE (i.e., at the clock where the CE would normally decrement from 1 to 0), then the new value in the CR will be transferred to the CE at that time. If only one of two bytes is loaded by the time the automatic transfer of the CR to the CE takes place, then the old value is used. In no case does loading the CR effect any count in progress.

Loading the CR with 0 is a special case. As previously mentioned, the first clock transfers the CR to the CE, and subsequent clocks count. The output pulses low when the CE decrements to 0; thus, the output will pulse every 10000H clocks after the CR is loaded with 0.

Loading the CR with 1 is another special case. The first clock transfers the CR to CE and since 1 is the terminating value, the next clock automatically reloads the CE from the CR. Therefore, if the CR value is still 1, the output, SYSTICK, will remain low.

4.8.2.1.2 80130 TIMER 1 OPERATION. Timer 1 is pre-defined to operate in the 8254 compatible Interrupt on Terminal Count mode (Mode 0). The output, DELAY, initially is low and remains low until the CR is loaded. The input clock for timer 1 is the output of timer 0 (SYSTICK). The first falling edge of the clock after the CR is loaded caused the CR to be transferred to the CE. Subsequent falling edges of the clock cause the CE to down count to 0. The output goes high when the content of the CE is 0. The CE continues to count thereafter.

If a new value is loaded in the CR prior to the expiration of the count, the old count is aborted and the first falling edge of the clock after the CR is loaded causes the CR to be transferred to the CE, and subsequent clocks count. The CE is frozen and the output remains low during the time after the first byte is written and before the second byte is written.

Loading the CR with 0 is a special case. As previously mentioned, the first clock transfers the CR to the CE, and subsequent clocks count. The output goes high when the CE decrements to 0; thus, the output will go high FFFFH +2 clocks after the CR is loaded with 0.

4.8.2.1.3 80130 TIMER 2 OPERATION. Timer 2 is pre-defined to have the 8254 compatible Square Wave mode (Mode 3). The output, BAUD, is initially high and remains high until the CR is loaded. The first falling edge of the clock after the CR is loaded causes the CR to be transferred to CE. Subsequent falling edges of the clock cause the CE to count. The output stays high for N/2 ((N+1)/2 if N is odd) counts and then goes low for N/2 ((N-1)/2 if N is odd) counts. On the falling edge of the clock which signifies the final count for the output in the high state, the output goes to the low state and the CR is automatically transferred to the CE. On the falling edge of the clock which signifies the final count for the output in the low state, the output returns to the high state and the CR is automatically transferred to the CE. Then the whole process is repeated. Subsequent falling edges of the clock count N/2 (or (N+1)/2) counts while the output is high, then the output goes low for N/2 (or (N-1)/2) counts, and so on.

Loading the CR while counting works as in timer 0. If the CR is loaded at the time the CR is automatically transferred to the CE, then the new value is used; otherwise the old value is used.

Loading the CR with 0 causes the output to be high for (FFFFH + 1)/2 counts, then low for (FFFFH + 1)/2 counts.

Loading the CR with 1 causes the output to be high for (FFFFH + 1)/2 + 1 counts, then low for (FFFFH + 1)/2 counts.

All CR values that are not 0 or 1 will count as previously described.

4.8.2.2 80130 Timer Programming

The 80130 Timer Control register allows the latching of the count of timers 0, 1, or 2 so their content may be read without interfering with counter operation. None of the timers are gateable.

In reading/writing all timers, the LSB is always accessed first, then the MSB. It is not possible to read/write any one entire timer contents in a single 16-bit operation. Table 4-12 summarizes the I/O port addresses for the 80130 timer functions.

Table 4-12. Port Addresses for the 80130 PIT

I/O Address	Device	Operation	Function Performed
00E8H	PIT Counter 0	Byte Read: Byte Write: Word:	LSB or MSB of count value LSB or MSB of count value N/A
00EAH	PIT Counter 1	Byte Read: Byte Write: Word:	LSB or MSB of count value LSB or MSB of count value N/A
00ECH	PIT Counter 2	Byte Read: Byte Write: Word:	LSB or MSB of count value LSB or MSB of count value N/A
00EEH	PIT Control Word Register Word:	Byte Read: Byte Write: N/A	N/A Control Word Register Parameter

4.8.2.2.1 CONTROL WORD REGISTER. The 80130 has restricted the programmability of each timer so that the control word is a single fixed byte value to initialize the corresponding timer. The formats accepted by the 80130 for mode definition of each timer to I/O Address EEH, is as follows:

```
Timer 0 Initialization word
    00110100    Command byte
    00------    Select timer 0
      11----    Least Significant byte then most
        010-    Mode 2 (Rate Generator)
           0    Binary Counter mode

Timer 1 Initialization word
    01110000    Command byte
    01------    Select timer 1
      11----    Least Significant byte then most
        000-    Mode 0 (Interrupt on Terminal Count)
           0    Binary Counter mode

Timer 2 Initialization word
    10110110    Command word
    10------    Select timer 2
      11----    Least Significant byte then most
        011-    Mode 3 (Square wave generator)
           0    Binary Count mode
```

The Counter latching command freezes the current value of the timer it commands. The values which are allowed and written to I/O Address EEH are:

00000000	Latch count in timer 0
01000000	Latch count in timer 1
10000000	Latch count in timer 2

4.8.2.2.2 80130 TIMER READ OPERATIONS. Reading a timer value always proceeds as follows. First, read the least significant byte and then the most significant byte. The 80130 automatically latches the count of a timer into the Output Latch (OL) whenever the LSB is read unless an earlier latch command was issued for that timer. This insures the MSB is the true extension of the LSB. A latch command transfers the current count for the address timer into its output latch (OL). Reading the MSB unlatches the value for the next operation. If a timer is latched and then, some time later, latched again before it was read, the second latch command is ignored.

4.8.3 80130 PIT PROGRAMMING EXAMPLE

Table 4-13 contains a programming sequence example that issues the required parameters to initialize the 80130 PIT device on the iSBC 186/03 board.

Table 4-13. PIT Programming Example

Sequence	Port/ Data	Functions Performed
1.	0EEH	Port Number for Counter 0
	034H	Command: select Counter 0, Read/Write LSB then MSB, Mode 0, 16-bit binary counter
	0FFH	Least significant byte of count value
	0FFH	Most significant byte of count value
2.	0EEH	Port Number for Counter 1
	070H	Command: select Counter 1, Read/Write LSB then MSB, Mode 3, 16-bit binary counter
	0FFH	Least significant byte of count value
	0FFH	Most significant byte of count value
3.	0EEH	Port Number for Counter 2
	0B6H	Command: select Counter 2, Read/Write LSB then MSB, Mode 3, 16-bit binary counter
	0FFH	Least significant byte of count value
	0FFH	Most significant byte of count value
Note: The count value of FFFF is intended only as an example.		

4.8.4 COUNT VALUES FOR COMMON BAUD RATES

Table 4-14 provides the count values for several common baud rates. These count values need to be programmed into the timers to obtain the desired baud rate. Refer to the previous sections and the data sheets for additional information.

Table 4-14. Timer Count Values for Common Baud Rates

Baud Rate	80186 Timers (1.5MHz)				80130 Timers (6MHz)			
	X1	X16	X32	X64	X1	X16	X32	X64
750K	2	--	--	--	8	--	--	--
500K	--	--	--	--	12	--	--	--
125K	12	--	--	--	48	3	--	--
64K	24	--	--	--	94	6	3	--
48K	31	--	--	--	125	8	6	3
19.2K	78	5	--	--	312	20	10	5
9600	156	10	5	--	625	39	20	10
4800	312	20	10	5	1250	78	39	20
2400	625	39	20	10	2500	156	78	39
1200	1250	78	39	20	5000	312	156	78
600	2500	156	78	39	10000	625	312	156
300	5000	312	156	78	--	1250	625	312
150	10000	625	312	156	--	2500	1250	625
110	--	852	426	213	--	3409	1704	852
75	--	1250	625	312	--	5000	2500	1250

Note: All count values shown in decimal notation.

4.9 PARALLEL PORT PROGRAMMING

When the iSBC 186/03 board is shipped, the parallel port (J3) interface is configured for a generic application. You can use the parallel port for a function other than a generic parallel port interface. Appendix C discusses how to convert the default parallel port configuration to either the SCSI configuration or the Centronics printer interface configuration.

Refer to Chapter 2 and Appendix C for more information.

The parallel port interface bits are read or written by a standard Intel 8255A-5 programmable peripheral interface (PPI) component. The interface allows Port A of the 8255A-5 to be input, output, or bi-directional; port B to be input or output; the lower bits (0 - 3) of port C to be open-collector outputs and the upper bits (4 - 7) of port C to be on-board control only. The 8255A-5 programming addresses are shown in Table 4-3.

These bits are default configured in a specific manner on the iSBC 186/03 board. Refer to chapter 2 for the details. Programming information for the 8255A-5 component is covered in the Intel Microprocessor and Peripheral Handbook and other reference documents.

4.9.1 PPI INITIALIZATION

As shipped, the iSBC 186/03 board requires that you provide a power-up initialization routine that configures the PPI device for Mode 0 operation; as an example if Port A and Port C of the PPI device are output ports and Port B is an input port, then the mode word is 82H.

The direction of data transfer of the transceiver on Port A can be changed by reconfiguring the jumpers. However, remember to re-initialize the PPI device with a different mode word; a mode word definition format of 82H makes port A operate as an output; a mode word of 90H makes port A operate as an input port.

Refer to the Intel Microprocessor and Peripheral Handbook for additional details on the various modes and bit set/reset operations on Port C.

4.10 SERIAL INTERFACE PROGRAMMING INFORMATION

Programming the operation of either the Connector J2 interface or the Connector J3 interface is done by programming channel A and channel B of the 8274 Multiple Protocol Serial Controller (MPSC) device. The following paragraphs present programming information for the 8274 MPSC, including port addresses, and initialization sequences. Appendix F provides a programming example for portions of the 8274. For more information, refer to the Intel Component Data Catalog (1982 version or later) or to the Intel Application Notes entitled: ASYNCHRONOUS COMMUNICATIONS WITH THE 8274 MULTIPLE PROTOCOL SERIAL CONTROLLER, AP-134; and SYNCHRONOUS COMMUNICATIONS WITH THE 8274 MULTIPLE PROTOCOL SERIAL CONTROLLER, AP-145.

4.10.1 8274 MPSC ADDRESSES

The port addresses, listed in Table 4-3, provide access to both channels of the 8274 MPSC device that operate the serial interfaces on the iSBC 186/03 board.

4.10.2 8274 MPSC PROGRAMMING INFORMATION

The 8274 Multiple Protocol Serial Controller (MPSC) is a two channel multi-function peripheral component designed to satisfy a wide variety of serial data communications requirements for microcomputers systems. The basic function performed by the 8274 MPSC is serial-to-parallel and parallel-to-serial data conversion and transfer.

The 8274 MPSC is capable of generating asynchronous and synchronous byte-oriented protocols such as IBM Bisync, and capable of generating synchronous bit-oriented protocols such as High Level Data Link Control (HDLC) or Serial Data Link Control (SDLC).

The 8274 MPSC contains 22 internal 8-bit registers that provide program control for the 2 channels of serial I/O. Each channel within the 8274 MPSC is assigned a set of 8 WRITE REGISTERS (WR0 through WR7) and 3 READ REGISTERS (RR0 through RR2) through which information is transferred. Each of the registers is accessible by means of pointer register WR0, that must be used to gain access to the other registers. Refer to the 8274 MPSC data sheet which shows the complete set of registers available for each serial channel of the 8274 MPSC. The following paragraphs describe the functions performed by each of the registers. Subsequent text provides information on how the registers define the operating mode of the 8274 MPSC.

The 8274 MPSC requires certain parameters to be initialized before beginning operation. The following text provides a quick outline of the parameter requirements for the 8274 MPSC that must be programmed to allow proper operation of the 8274 MPSC with the iSBC 186/03 board. That is, whenever one of the following parameters is written to the 8274 MPSC, ensure that the bits listed in the following text are configured as shown.

Write Register 1 — D7=0; the wait function on the 8274 MPSC is not supported. Always disable the wait enable feature.

Write Register 2 (Channel A) — D7=0; this provides the RTS(B)* signal output on pin-10 of the 8274 MPSC for channel A.

D5=1; the Vectored Interrupt mode of operation of the 8274 MPSC is supported on the iSBC 186/03 board.

D4=1 and D3=0; the combination allows operation in an 8086/8088 interrupt vector generating mode for the CPU.

D0=1 and D1=0; the iSBC 186/03 board (when configured as shipped) dedicates channel A of the 8274 MPSC to DMA operation with the 80186 DMA controller. If only one channel is to be used for DMA, you must use Channel A.

The 8274 MPSC can operate at a speed of 800k baud; this rate of operation requires that the DMA Controller provide service to the 8274 within 10 microseconds for continuous operation. Because of this operating speed, you should not attempt to access slow off-board memory resources during high-speed transmission or reception via the 8274 MPSC.

More information on the functions performed by each of the parameters and operation of the 8274 MPSC is provided in subsequent text.

4.10.2.1 8274 MPSC Register Descriptions

The 8274 MPSC contains two independent channels that each require a separate set of 11 parameter registers as shown in the 8274 MPSC data sheet. The registers are used during initialization and command execution in the 8274 MPSC.

In accessing each register within the 8274 MPSC, you must first load the access code for the register into Write Register 0 to point to the desired register. After doing so, the next operation (either a READ or a WRITE) to the 8274 MPSC is directed by WR0 to the required register; RR0 through RR2 for a READ operation, WR0 through WR7 for a WRITE operation. In some instances, the format of the registers (WR3, WR4, and WR5) changes according to the operating mode selected for the 8274 MPSC. In such cases, the functions of the registers are described for each operating mode. The data sheet lists the functions performed by each register and the pointer values required (WR0) in accessing the registers. A more detailed description of the functions of each register may be found in the following paragraphs.

WRITE REGISTER 0 (WR0). Write Register 0 contains a pointer (the three least significant bits) that selects which of the parameter registers (WR1-WR7 and RR0-RR2) is to be involved in the next sequential read or write operation. It also includes a field of 3 bits for 8 commands to perform channel reset, error and interrupt resets, enable receive interrupt modes and end of interrupt. A field of two bits to perform CRC reset functions are included. Refer to the data sheet for additional information.

WRITE REGISTER 1 (WR1). WR1 provides the control functions that place the 8274 MPSC device into its various interrupt and wait-ready modes, as shown in the 8274 MPSC data sheet. The iSBC 186/03 board does not support the use of the RDY/WAIT pin.

D0 External/Status Interrupt Enable; allows interrupt to occur as the result of transitions on the CD/ or CTS/ inputs. Also allows interrupts as the result of a Break/Abort detection and termination, or at the beginning of CRC, or sync character transmission when the Transmit Underrun/EOM latch is set.

NOTE

The iSBC 186/03 board does not support the use of the SYNDET/ input signal for generation of the external/Status Interrupt Enable bit; refer to the Intel Microprocessor & Peripheral Handbook for more information.

D1 Transmitter Interrupt/DMA Enable; allows the 8274 MPSC to interrupt or request a DMA transfer when the transmitter buffer becomes empty.

D2 Status Affects Vector Enable (WR1, D2 active in channel B only); If this bit is not set, then the fixed vector, programmed in WR2, is returned from a read of RR2 (channel B). If the bit is set, then the vector returned from a read of RR2 (channel B) is variable as shown in the Interrupt Vector Mode Table. The Interrupt Vector Mode table in the 8274 data sheet shows how the 8274 modifies the lower 3 bits of the vector programmed into the 8274 in WR2, channel B.

D4/D3 Receive Interrupt Mode Select

0 0	Receive Interrupts/DMA Disabled
0 1	Receive Interrupt on First Character Only or Special Condition
1 0	Interrupt on All Receive Characters or Special Condition (Parity Error is a Special Receive Condition)
1 1	Interrupt on All Receive Characters or Special Condition (Parity Error is not a Special Receive Condition).

D5 Not used on the iSBC 186/03 board; must be zero.

D6 Not used on the iSBC 186/03 board; must be zero.

D7 Not used on the iSBC 186/03 board; must be zero.

WRITE REGISTER 2 (WR2). WR2 for channel B provides the interrupt vector for either a channel A or a channel B interrupt. Vector bits V1 through V3 are returned as written if the "Status Affects Vector" bit (WR1, bit 2) is LOW. If the "Status Affects Vector" bit is HIGH, vector bits V1 through V3 are modified, as shown in the Vector Mode table of the 8274 data sheet, before being placed into RR2.

WR2 for channel A provides the control functions that select the Interrupt/DMA modes of the 8274, Interrupt Control Priority and Vectored mode.

The bit definitions for WR2 when used with channel A of the 8274 MPSC are as follows.

D1/D0 System Configuration; These specify the type of data transfer control to be used in transferring data from 8274 MPSC to the CPU (either interrupt or DMA control).

0 0	Channel A and Channel B both use interrupts for Data
0 1	Channel A uses DMA, Channel B uses interrupt for Data
1 0	Channel A and Channel B both use DMA for Data
1 1	Illegal Code

Note: When only one channel is used for DMA, it must be Channel A. When either Channel is in the interrupt mode, RxDRQB/IPI must be grounded by installing jumper E51 to E52. Refer to sections 4.5 and 4.7 for additional details on the 8274 DMA/ Interrupt operation.

D2 Priority; this bit specifies the relative priorities of the channel operations internal to the 8274 MPSC.

- 0 Selects priority as follows: RxA (highest), TxA, RxB, TxB, EXTA, EXTB (lowest)
- 1 Selects priority as follows: RxA (highest), RxB, TxA, TxB, EXTA, EXTB (lowest)

D4,D3 Interrupt Code; specifies the behavior of the 8274 MPSC when it receives a command to read the interrupt vector from register RR2.

- 0 0 Not allowed on the iSBC 186/03 board.
- 0 1 Not allowed on the iSBC 186/03 board.
- 1 1 8086/88 Mode.

D5 Vector Mode; specifies whether the 8274 will generate an interrupt vector in response to an interrupt acknowledge sequence or whether it will allow the 80130 master PIC to generate an interrupt vector.

- 0 8274 does not provide a vector
- 1 8274 provides an interrupt vector to 80186.

D6/D7 Must always be zero for the iSBC 186/03 board.

WRITE REGISTER 3 (WR3). WR3 consists of the receiver logic control bits, formatted as shown in the 8274 data sheet. The various bits define for the receiver portion of the 8274 MPSC the type of receive operation that it is to perform. Each bit shown in the figure is described in the following paragraphs.

D0 Receiver Enable; a one enables the receiver to begin. This bit should be set only after the receiver has been initialized. A zero disables the receiver.

D1 Sync Character Load Inhibit; a one prevents the receiver from loading sync characters into the receive buffers. In SDLC, this bit must be zero. The bit should not be used when operating in an asynchronous mode.

D2 Address Search Mode; if the SDLC mode has been selected, the 8274 MPSC will receive all frames unless this bit is a 1. If this bit is a 1, the 8274 MPSC will receive only frames with address bytes that match the global address (0FFH) or the value loaded into WR6. This bit must be zero in all non-SDLC modes.

D3 Receive CRC Enable; a one in this bit enables (or re-enables) CRC calculation by the 8274 MPSC. CRC calculation starts when the last character is placed into the receiver portion of the 8274 MPSC. A zero in this bit disables, but does not reset, the receiver CRC generator.

D4 Enter Hunt Phase; after initialization, the 8274 MPSC automatically enters the Hunt mode. If synchronization is lost, the Hunt phase can be re-entered by writing a one to this bit.

D5 Auto Enables; a one written to this bit causes CD/ to be an automatic enable signal for the receiver and CTS/ to be an automatic enable signal for the transmitter. A zero written to this bit limits the effect of CD/ and CTS/ signals to setting/resetting their corresponding bits in the status register (RR0).

D7/D6 Receive Character length

0 0	Receive 5 data bits per character
0 1	Receive 7 data bits per character
1 0	Receive 6 data bits per character
1 1	Receive 8 data bits per character

WRITE REGISTER 4 (WR4). WR4 contains control bits whose functions affect both the receiver and the transmitter portions of the 8274 MPSC; refer to the 8274 MPSC data sheet. During the initialization of the 8274 MPSC, always configure the WR4 register bits before WR1, WR3, WR5, WR6, and WR7. The function of each bit of WR4 is described in the 8274 MPSC data sheet.

WRITE REGISTER 5 (WR5). WR5 contains control bits that affect the operation of the transmitter, with the exception of bit 2, which affects both the transmitter and the receiver. The 8274 MPSC data sheet shows the format and describes the function of each bit.

When transmitting 5 or less data bits per character, the 8274 MPSC right justifies the data bits and sends the least significant bit first. The transmit character format for an operation containing from one to five bits per character is described in the 8274 data sheet.

WRITE REGISTER 6 (WR6). The format of WR6 is shown in the 8274 MPSC data sheet. The contents of WR6 varies depending on the mode of operation of the 8274 MPSC.

When operating in Monosync Mode, the 8274 MPSC expects to find a transmit sync character in WR6.

In Bisync Mode, the 8274 MPSC expects to find a 16-bit sync character in WR6 and WR7. The WR6 field contains the low order 8 sync bits in Bisync mode.

In SDLC Mode, the 8274 MPSC requires from WR6 a secondary address field that is compared to the address field of the SDLC frame. The Sync/Address field contains the Address byte in SDLC mode.

CAUTION

Be sure to program the receiver and transmitter with its sync character before enabling the receiver or transmitter.

WRITE REGISTER 7 (WR7). The format of WR7 is shown in the 8274 MPSC data sheet. The contents of WR7 varies between sync and flag characters, depending on the mode of operation of the 8274 MPSC. When operating in Monosync mode, the 8274 MPSC expects to find a receive sync character in WR7. In Bisync mode, the 8274 MPSC expects to find the second byte (high order 8-bits) of a 16-bit sync character in WR7. In the SDLC mode, the 8274 MPSC expects to find Flag character (01111110). WR7 is not used in External Sync mode.

READ REGISTER 0 (RR0). RR0 contains the status of the receive and transmit buffers within the 8274 MPSC. Included in the register format are the Data Carrier Detect (DCD/), the Clear-To-Send (CTS), the Transmit Underrun, the Break/Abort, the Sync/Hunt, the Interrupt Pending (Channel A), the Transmit Buffer Empty, and the Receive Character Available status bits. The 8274 MPSC data sheet shows the format and describes the function of each status bit of RR0. There are unique differences when using the 8274 MPSC device on the iSBC 186/03 board. When the 8274 is in the Vectored mode of operation, Channel A of the 8274 MPSC sets bit D1 at the falling edge of the second INTA in an interrupt cycle after WR2 is specifed. Bit D4 (Sync/Hunt bit) is not supported on the iSBC 186/03 board.

READ REGISTER 1 (RR1). RR1 contains the special receive condition status bits and Residue codes describing the I field in the SDLC Receive mode of operation. Included in the framework of the register are status indications for All Bits Sent, Residue Code 0, Residue Code 1, Residue Code 2, Parity Error Detected, Receive Overrun Error Detected, CRC/Framing Error Detected, and End Of Frame Detected (SDLC). Bits D1, D2 and D3 (Residue Codes) synchronous protocols (HDLC and SDLC) allow I-fields that are not an integral number of characters. Since transfers from the 8274 MPSC to the CPU are character oriented, the residue codes provide the capability of receiving leftover bits. Residue bits are right justified in the last two data bytes received. The 8274 MPSC data sheet shows the format of RR1 and describes the function of each bit in the register.

READ REGISTER 2 (RR2). RR2 contains the interrupt vector that was last written into WR2 (channel B format) provided that the Status Affects Vector bit of WR1 is not set. If the bit is set, then RR2 contains the modified vector for the highest priority interrupt pending. If no interrupts are pending, the variable bits in the vector are set to one. The contents of RR2 may be read only through channel B of the 8274 MPSC. Refer to the 8274 MPSC data sheet for the register format. Refer to the description of WR2 for more information on vector modification.

4.10.3 8274 MPSC OPERATING MODES

The following paragraphs describe the protocol generating abilities of the 8274 MPSC in transmitting and receiving data in Asynchronous, Bisynchronous, HDLC, and SDLC communications protocols. Each type of operation requires a particular configuration for WR3, WR4, and WR5. The register settings are used internally in the 8274 MPSC to define the direction of data transfer and the communication protocol.

The description for each operating mode includes details of the requirements for WR3, WR4, and WR5.

Appendix F provides a programming example of the 8274 in the DMA mode.

4.10.3.1 Asynchronous Mode Operation

Figure 4-3 shows the asynchronous mode message format. To receive or transmit data in an Asynchronous mode, the 8274 MPSC must be initialized with the following parameters:

1) Character length,
2) Clock rate,
3) Number of stop bits,
4) Type of parity (whether even, odd, or none),
5) Type of interrupt mode required (whether polled, interrupt driven, or DMA driven), and
6) Transmit and Receive enables.

The system software must load the appropriate parameters into the Write Registers (WR0 through WR5) within the 8274 MPSC. The WR4 parameter must always be issued before the WR1, WR3, and WR5 parameters. The 8274 data sheet shows the format for the WR3, WR4, and WR5 parameters for Asynchronous mode operation; the format of the registers changes for operations other than Asynchronous.

WR2 (channel B only) stores the interrupt vector; WR1 defines the interrupt modes and data transfer modes. WR6 and WR7 are not used in Asynchronous mode operation. The iSBC 186/03 board allows the user to configure the 8274 MPSC in the Vectored Interrupt status affects vector mode. This allows the 8274 to directly generate a unique interrupt vector for all the special conditions described in the Interrupt Vector Mode table of the 8274 data sheet.

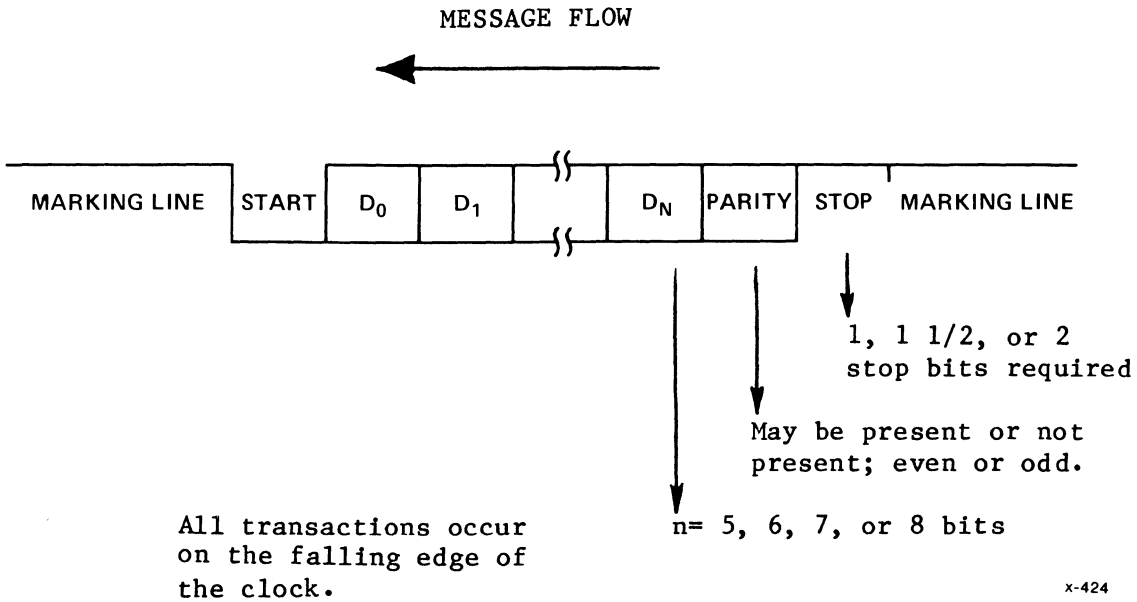


Figure 4-3. Asynchronous Message Format

Table 4-15 shows a list of programming steps required for initializing a full-duplex transmit/receive operation in either channel of the 8274 MPSC operating in Asynchronous mode.

WRITE REGISTER 3 (WR3). When the 8274 MPSC is operating in asynchronous mode, the WR3 register has several functions as shown in the 8274 data sheet. Bit 0 of WR3 controls the Receiver Enable (Rx enable). Bits 1 through 4 are not used and should always be 0. Bit 5 controls the state of the Auto Enable bit. Bits 6 and 7 work together to select the number of data bits per receive character; either 5, 6, 7, or 8 bits per character.

WRITE REGISTER 4 (WR4). When the 8274 MPSC is operating in asynchronous mode, the WR4 register performs several functions as shown in the 8274 data sheet. Bit 0 of WR4 controls the Parity Enable. Bit 1 of the register performs selection of the type of parity: 1 = even, 0 = odd. Bits 2 and 3 allow selecting 0, 1, 1 1/2, or 2 stop bits for each character transfer.

Bits 4 and 5 of WR4 are not used and should remain zeroed. Bits 6 and 7 work together to select the rate at which the 8274 MPSC transmits and receives characters with respect to the 8274 transmit and receive clocks: either divide by 1, 16, 32, or 64 of the clock frequency.

WRITE REGISTER 5 (WR5). When the 8274 MPSC is operating in asynchronous mode, the WR5 register performs several functions as shown in the 8274 data sheet. Bit 0 and bit 2 of WR5 are not used. Bit 1 of the register provides control over the condition of the Request to Send (RTS) signal for an RS232C interface: 1 = active, 0 = inactive. Bit 3 enables or disables the transmitter portion of the 8274 MPSC. Bit 4 of WR5 causes the transmitter to drive the data line low (break condition). Bits 5 and 6 work together to select the number of bits contained in each transmit data field: either 5, 6, 7, or 8 bits per character. Bit 7 provides control of the Data Terminal Ready (DTR) signal for an RS232C interface: 1 = active, 0 = inactive.

Table 4-15. Asynchronous Mode Programming Sequence

Function	Typical Program Steps		Comments
	Register	Information Loaded	
INITIALIZE	WR0	CHANNEL RESET.	Reset 8274 MPSC.
	WR0	LOAD WR2 ADDRESS.	
	WR2	LOAD INTERRUPT VECTOR VALUE.	Channel B only.
	WR0	RESET EXTERNAL/-STATUS INTERRUPT.	Load WR4 address.
	WR4	ISSUE PARAMETERS.	Parameters affected are: Asynchronous Mode, Parity, Stop Bit, Clock Rate.
	WR0	LOAD WR3 ADDRESS.	
	WR3	ISSUE PARAMETERS.	Parameters affected are: Receive Enable, Auto Enable, Receive Character Length.
	WR0	LOAD WR1 ADDRESS, RESET EXTERNAL/-STATUS INTERRUPT.	
	WR1	SET UP INTERRUPT SENSING.	Parameters affected are: Transmit Interrupt Enable, Status Affects Vector, Interrupt On All Receive Characters, Disable Wait, Ready, External Interrupt Enable Transmit/Receive interrupt mode selected.

Table 4-15. Asynchronous Mode Programming Sequence (continued)

Function	Typical Program Steps		Comments
			External Interrupt monitors the status of the CTS/ and CD/ inputs and detects the Break/Abort character. Status Affects Vector for channel B only.
	WR0	LOAD WR2 ADDRESS.	Channel A only
	WR2	SETUP INTERRUPT/DMA MODE.	Channel A only
	WR0	LOAD WR5 ADDRESS.	
	WR5	ISSUE PARAMETERS.	Parameters affected are: Request To Send, Transmit Enable, Transmit Character Length, Data Terminal Ready.
	WR5	ISSUE PARAMETERS.	Receive and Transmit both fully initialized. Auto Enable will enable Transmitter if CTS/ is active and Receiver if CD/ is active.
INITIALIZE	TRANSFER FIRST DATA BYTE TO 8274 MPSC.		
IDLE MODE	EXECUTE HALT INSTRUCTION OR SOME OTHER PROGRAM.		Program is waiting for an interrupt from the 8274 MPSC.
	DATA TRANSFER CONTINUES UNTIL ONE OF 4 EVENTS OCCURS:		When the interrupt occurs, the interrupt vector is modified by: 1) Receive Character Available; 2) Transmit Buffer Empty; 3) External/Status change; and 4) Special Receive condition.
	WHEN A CHARACTER IS RECEIVED IN THE 8274, THE FOLLOWING EVENTS MUST OCCUR: • TRANSFER DATA CHARACTER TO CPU. • UPDATE POINTERS AND PARAMETERS. • RETURN FROM INTERRUPT.		
	IF TRANSMITTER BUFFER IS EMPTY:		TxEmpty interrupt is enabled.

Table 4-15. Asynchronous Mode Programming Sequence (continued)

Function	Typical Program Steps	Comments
	• TRANSFER DATA CHARACTER TO 8274. • UPDATE POINTERS AND PARAMETERS. • RETURN FROM INTERRUPT.	Only when data is to be transmitted. Program control is transferred to an interrupt service routine. If transmit is completed, turn-off the transmitter.
DATA TRANSFER ERROR MONITORING	IF EXTERNAL STATUS CHANGES: • TRANSFER RR0 to CPU. • PERFORM ERROR SERVICE ROUTINES (INCLUDE BREAK DETECTION).	
	• RETURN FROM INTERRUPT.	The modified interrupt vector (RR2) will be returned to the CPU in the Interrupt Acknowledge sequence.
	IF SPECIAL RECEIVE CONDITION OCCURS: • TRANSFER RR1 to CPU. • DO SPECIAL ERROR (E.G., FRAMING ERROR) SERVICE. • ROUTINE RETURN FROM INTERRUPT.	
	REDEFINE RECEIVE/TRANSMIT.	Occurs when transmit or receive data transfer is complete.
TERMINATION	INTERRUPT MODES DISABLE TRANSMIT/RECEIVE MODES.	
	UPDATE MODEM CONTROL OUTPUTS (E.G., TURN-OFF RTS).	

4.10.3.2 Synchronous 8274 MPSC Operation (Monosync, Bisync)

To obtain synchronous operation of the 8274 MPSC (either in a monosynchronous or bisynchronous mode), initialize the 8274 MPSC with the following parameters: either odd or even parity (WR4; D1, D0), X1 clock mode (WR4; D7, D6), either 8-bit or 16-bit sync character (WR4; D5, D4), CRC polynomial (WR5; D2), Transmitter Enable (WR5; D3), interrupt modes (WR1, WR2), transmit character length (WR5; D6, D5) and receive character length (WR3; D7, D6). The WR4 parameters must be written before WR1, WR3, WR5, WR6 and WR7.

During synchronous operation, the 8274 MPSC transmits data on the falling edge of the Transmit Clock (TxC), and receives data on the rising edge of Receive Clock (RxC). The X1 clock is used for both transmit and receive operations for all three sync modes: Monosync, Bisync, and SDLC. Figures 4-4 and 4-5 show the message formats for the synchronous modes of operation.

The 8274 data sheet shows the format of the WR3, WR4 and WR5 registers and the functions performed during synchronous receive and transmit operations. WR0 points to the other registers that are to receive the parameters and commands, WR1 defines the interrupt modes, WR2 stores the interrupt vector, and WR6 and WR7 store sync characters.

Tables 4-16 and 4-17 illustrate the typical program steps that implement a half-duplex Bisync transmit and receive operation.

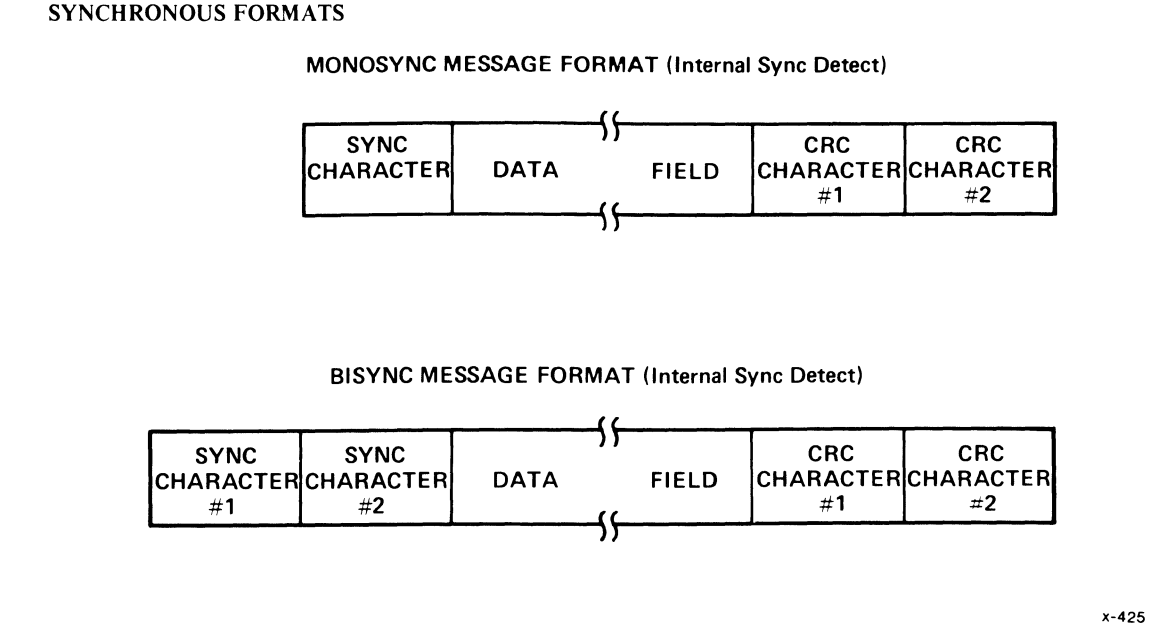


Figure 4-4. Synchronous Message Formats-8274 MPSC

Synchronous Transmit Operation

Data transfers involving the use of the RDY signal output from the 8274 MPSC are not supported. DMA transfers using the TxDRQA and TxDRQB signals indicate to the 80186 CPU that the transmit buffer in the 8274 MPSC is empty, and that the 8274 MPSC is ready to accept the next data character. If the next data character is not loaded into the 8274 MPSC before the transmit shift register empties, the 8274 MPSC indicates a Transmit Underrun condition. The DMA Multiplexer must be programmed on the iSBC 186/03 board to route the proper DMA request to one of the DMA channels. Refer to Section 4.5 for details on the DMA Multiplexer.

SYNCHRONOUS RECEIVE OPERATION. When a channel reset is issued to the 8274 MPSC, the receiver enters Hunt mode, during which the 8274 MPSC looks for sync characters. The Hunt mode begins only when the receiver is enabled and data transfer begins only when sync characters are recognized. Therefore, the proper sync character must be loaded into the receiver and the driver prior to enabling the receiver.

Table 4-16. Bisync Mode Transmit Programming Sequence

Function		Typical Program Steps	Comments
	Register	Information Loaded	
	WR0	CHANNEL RESET, RESET TRANSMIT CRC GENERATOR. LOAD WR2 ADDRESS.	Reset 8274 MPSC, initialize CRC generator.
	WR2	LOAD INTERRUPT VECTOR VALUE.	Channel B only.
	WR0	LOAD WR2 ADDRESS.	
	WR2	SETUP INTERRUPT/DMA MODES.	Channel A only.
	WR0	LOAD WR4 ADDRESS.	
	WR4	ISSUE TRANSMIT PARAMETERS.	Parity Information, Sync modes information, x1 Clock mode.
	WR0	LOAD WR3 ADDRESS.	
	WR3	AUTO ENABLES.	Transmission begins only after CTS* is detected.
	WR0	LOAD WR6 ADDRESS.	

Table 4-16. Bisync Mode Transmit Programming Sequence (continued)

Function	Typical Program Steps		Comments
	Register	Information Loaded	
INITIALIZE	WR6	SEND SYNC CHARACTER 1.	
	WR0	LOAD WR7 ADDRESS, RESET EXTERNAL/STATUS INTERRUPTS.	
	WR7	SEND SYNC CHARACTER 2.	
	WR0	LOAD WR1 ADDRESS, RESET EXTERNAL/STATUS INTERRUPTS.	
	WR1	SET UP INTERRUPT SENSING.	Parameters affected are: Status Affects Vector, Ex- ternal Interrupt Enable, Transmit Interrupt Enable, and Wait/Ready mode Enable. External Interrupt mode monitors the status of CTS* and CD* input pins as well as the status of Tx Underrun/EOM latch. Transmit Interrupt Enable in- terrupts the 80186 when the Transmit buffer becomes empty. The WAIT/Ready mode is not sup- ported on the iSBC 186/03 board.
	WR0 WR5	LOAD WR5 ADDRESS ISSUE PARAMETERS	Parameters affected are: Request to Send, Transmit Enable, Bisync CRC, Transmit Character Length. Transmit CRC Enable should be set when first non-sync and non-control data character is sent to 8274 MPSC.

Table 4-16. Bisync Mode Transmit Programming Sequence (continued)

Function	Typical Program Steps	Comments
	TRANSFER FIRST SYNC BYTE TO 8274 MPSC	Need several sync characters in the beginning of message. Transmitter is fully initialized.
IDLE MODE	EXECUTE HALT INSTRUCTION OR SOME OTHER PROGRAM	Waiting for interrupt to transfer data.
DATA TRANSFER AND STATUS MONITORING	When interrupt occurs: • INCLUDE/EXCLUDE DATA BYTE FROM CRC ACCUMULATION (IN 8274 MPSC). • TRANSFER DATA BYTE FROM CPU (OR MEMORY) TO 8274 MPSC. • DETECT AND SET APPROPRIATE FLAGS FOR CONTROL CHARACTERS (IN CPU). • RESET Tx UNDERRUN/EOM LATCH (WR0) IF LAST CHARACTER OF MESSAGE IS DETECTED. • UPDATE POINTERS AND PARAMETERS (CPU). • RETURN FROM INTERRUPT. IF ERROR CONDITION OR STATUS CHANGE OCCURS: • TRANSFER RR0 TO CPU. • EXECUTE ERROR ROUTINE. • RETURN FROM INTERRUPT.	Interrupt occurs when first data byte is being sent. DMA mode allows DMA transfer from memory to 8274 MPSC. Tx Underrun/EOM indicates either transmit underrun (sync character being sent) or end of message (CRC-16 being sent).
TERMINATION	DISABLE TRANSMIT MODE. REDEFINE RECEIVE/TRANSMIT INTERRUPT MODES. UPDATE MODEM CONTROL OUTPUTS (E.G., TURN OFF RTS).	Occurs when Transmit or Receive Data Transfer is complete. Program should gracefully terminate message.

Table 4-17. Bisync Mode Receive Programming Sequence

Function	Typical Program Steps		Comments
	Register	Information Loaded	
	WR0	CHANNEL RESET, RESET RECEIVE CRC CHECKER.	Reset 8274 MPSC, Initialize Receive CRC checker.
	WR0	LOAD WR2 ADDRESS.	
	WR2	INTERRUPT VECTOR.	Channel B only.
	WR0	LOAD WR4 ADDRESS.	
	WR0	LOAD WR2 ADDRESS.	
	WR2	SETUP INTERRUPT/DMA MODE	Channel A only.
	WR4	ISSUE RECEIVE PARAMETERS.	Parameters affected are: Parity, Sync modes, x1 Clock mode.
	WR0	LOAD WR5 ADDRESS. RESET EXTERNAL STATUS INTERRUPT.	
	WR5	BISYNC CRC-16, DATA TERMINAL READY.	
	WR0	LOAD WR3 ADDRESS.	
INITIALIZE	WR3	ISSUE PARAMETERS.	Parameters affected are: Sync Character Load Inhibit, Receive CRC Enable; Enter Hunt Mode, Auto Enables, Receive Character Length. Sync Character Load Inhibit strips all the leading sync characters at the beginning of the message. Auto Enables enables the receiver to accept data only after the CD* input is active. Receiver is not enabled yet.
	WR0	LOAD WR6 ADDRESS.	
	WR6	SEND SYNC CHARACTER 1.	
	WR0	LOAD WR7 ADDRESS.	
	WR7	SEND SYNC CHARACTER 2.	

Table 4-17. Bisync Mode Receive Programming Sequence (continued)

Function	Typical Program Steps		Comments
	Register	Information Loaded	
INITIALIZE (continued)	WR0	RESET EXTERNAL/STATUS INTERRUPT.	
	WR1	SET UP INTERRUPT SENSING.	Parameters affected are: Status Affects Vector, External Interrupt Enable, Receive Interrupt On First Character Only. In this interrupt mode, only the first non-sync data character causes an interrupt to the CPU. The DMA channel can be enabled by the service routine. All subsequent data is transferred on a DMA basis; however, Special Receive Condition interrupt will interrupt the CPU. Status Affects Vector used in Channel B only.
	WR0	LOAD WR3 ADDRESS. ENABLE INTERRUPT ON NEXT RECEIVE CHARACTER.	Resetting this interrupt provides simple program loopback entry for the next transaction.
	WR3	ISSUE PARAMETERS.	Parameters affected are: Receive Enable, Sync Character Load Inhibit, Enter Hunt mode, Auto Enable, Receive Word Length. WR3 is reissued to enable receiver. Receive CRC Enable must be set after receiving SOH or STX character.
IDLE MODE	EXECUTE HALT INSTRUCTION OR SOME OTHER PROGRAM.		Receive mode is fully initialized and the system is waiting for Interrupt On First Character.

Table 4-17. Bisync Mode Receive Programming Sequence (continued)

Function	Typical Program Steps	Comments
	When interrupt on first character occurs, the CPU does the following: • DETECTS AND SETS APPROPRIATE FLAGS FOR CONTROL CHARACTERS (IN CPU). • INCLUDES/EXCLUDES DATA BYTE IN CRC CHECKER.	Enables Sync Character loading.
	• UPDATES POINTERS AND OTHER PARAMETERS. • ENABLES DMA Multiplexer to provide proper DMA request to the CPU. • ENABLES DMA CONTROLLER. • RETURNS FROM INTERRUPT.	During the Hunt mode, the 8274 MPSC detects two contiguous sync characters to establish synchronization. The CPU establishes the DMA mode and all subsequent data characters are transferred by the 80186 DMA Controller.
DATA TRANSFER AND STATUS MONITORING	When DMA REQUEST (DRQ) becomes active, the DMA Controller does the following: • TRANSFERS DATA BYTE TO MEMORY. • INTERRUPTS THE CPU IF THE LAST CHARACTER OF THE MESSAGE IS DETECTED.	
	For message termination, the CPU does the following: • TRANSFERS RR1 TO THE CPU. • SETS SYNC CHARACTER LOAD INHIBIT. • UPDATES POINTERS AND PARAMETERS. • RETURNS FROM INTERRUPT.	The 8274 MPSC interrupts the CPU for error condition, and the error routine aborts the present message, clears the error condition, and repeats the operation.

Table 4-17. Bisync Mode Receive Programming Sequence (continued)

Function	Typical Program Steps	Comments
TERMINATION	Redefine the interrupt and sync modes. UPDATE MODEM CONTROLS. DISABLES RECEIVE MODE.	 Program should gracefully terminate message handling.

Figure 4-5 illustrates the message format for operation in the SDLC/HDLC mode.

The Interrupt modes for SDLC operation are similar to those discussed previously in the synchronous operations section.

Tables 4-17 and 4-18 show typical programming steps required to implement a half-duplex SDLC transmit and receive operation.

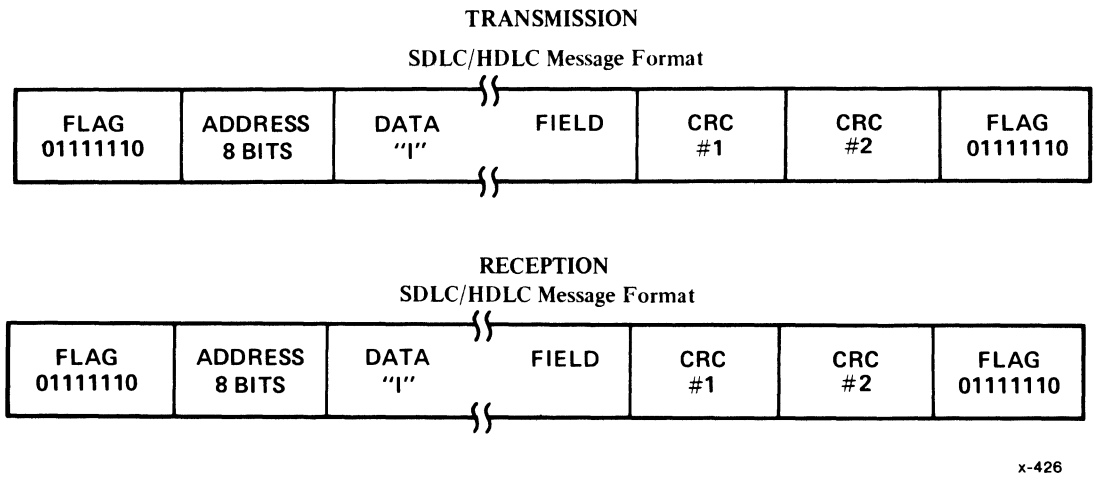


Figure 4-5. Transmit/Receive SDLC/HDLC Message Format

Table 4-18. SDLC Mode Transmit Sequence

Function	Typical Program Steps		Comments
	Register	Information Loaded	
INITIALIZE	WR0	RESET CHANNEL 1.	Reset 8274 MPSC.
	WR0	LOAD WR2 ADDRESS.	
	WR2	INTERRUPT VECTOR.	Channel B only.
	WR0	LOAD WR2 ADDRESS.	Channel A only.
	WR2	SETUP INTERRUPT/DMA MODE	Channel A only.
	WR0	LOAD WR4 ADDRESS. RESET EXTERNAL/STATUS INTERRUPTS.	
	WR4	ISSUE TRANSMIT PARAMETERS. PARITY INFORMATION SDLC MODE, X1 CLOCK MODE.	
	WR0	LOAD WR3 ADDRESS.	
	WR3	ISSUE PARAMETERS.	Auto Enables. Transmitter sends data only after CTS* is detected.
	WR0	LOAD WR1 ADDRESS, RESET EXTERNAL/STATUS INTERRUPTS.	
	WR1	SET UP INTERRUPT SENSING.	External Interrupt Enable, Status Affects Vector, Transmit Interrupt Enable, or DMA Mode Enable. The External interrupt Mode monitors the status of the CTS* and CD* inputs, as well as the status of Transmit underrun EOM latch. Transmit Interrupt interrupts when the Transmit buffer becomes empty. The DMA Mode can be used to transfer data with the 80186 DMA Controller.

Table 4-18. SDLC Mode Transmit Sequence (continued)

Function	Typical Program Steps		Comments
	Register	Information Loaded	
			The first interrupt occurs when CTS* becomes active at which point flags are transmitted by the 8274 MPSC. The first data byte (address field) can be loaded in the 8274 MPSC after this interrupt. Flags cannot be sent to the 8274 MPSC. Status Affects Vector used in Channel B only.
	WR0	LOAD WR5 ADDRESS.	
INITIALIZE	WR5	ISSUE PARAMETERS.	Transmit CRC Enable, Request To Send, SDLC-CRC, Transmit Enable, Transmit Word Length Data Terminal Ready. SDLC-CRC Mode must be defined before initializing transmit CRC generator.
	WR0	RESET TRANSMIT CRC GENERATOR.	Initialize CRC generator to 1's.
IDLE MODE		EXECUTE HALT INSTRUCTION OR SOME OTHER PROGRAM.	Waiting For Interrupt or DMA REQUEST output to transfer data.
When Interrupt occurs, the CPU does the following: • CHANGES TRANSMIT WORD LENGTH (IF NECESSARY). • TRANSFERS DATA BYTE FROM CPU (MEMORY) TO 8274 MPSC. • RESETS Transmit UNDERRUN LATCH (WR0).			

Table 4-18. SDLC Mode Transmit Sequence (continued)

Function	Typical Program Steps	Comments
		Flags are transmitted by the 8274 MPSC as soon as Transmit Enable is set and CTS* becomes active. The CTS* status change is the first interrupt that occurs and is followed by transmit buffer empty for subsequent transfers.
DATA TRANSFER AND STATUS MONITORING	If last character of the I-Field is sent, the 8274 MPSC does the following: • SENDS CRC. • SENDS CLOSING FLAG. • INTERRUPTS CPU WITH BUFFER EMPTY STATUS.	
		Word length can be changed "on the fly" for variable I-field length. The data byte can contain address, control, or I-field information (never a flag). It is good practice to reset Transmit Underrun/EOM latch in the beginning of the message to avoid a false end-of-frame detection at the receiving end. This ensures that when underrun occurs, CRC is transmitted and underrun interrupt (Transmit underrun/EOM latch active) occurs. Note that "Send Abort" can be used in the 8274 MPSC in response to any interrupting continuing to abort the transmission.

Table 4-18. SDLC Mode Transmit Sequence (continued)

Function	Typical Program Steps	Comments
	CPU does the following: • ISSUES RESET TRANSMIT INTERRUPT PENDING COMMAND TO THE 8274 MPSC. • UPDATES POINTERS AND PARAMETERS (CPU). • REPEATS THE PROCESS FOR NEXT MESSAGE, ETC. If the Vector indicates an error, the CPU does the following: • SENDS ABORT. • EXECUTES ERROR ROUTINE. • UPDATES PARAMETERS, MODES, ETC. • RETURNS FROM INTERRUPT.	
TERMINATION	REDEFINE INTERRUPT MODES. UPDATE MODEM CONTROL OUTPUTS. DISABLE TRANSMIT MODE.	Program should terminate message handling gracefully

Table 4-19. SDLC Mode Receive Sequence

Function	Typical Program Steps		Comments
	Register	Information Loaded	
	WR0	RESET CHANNEL 2.	Reset 8274 MPSC.
	WR0	LOAD WR2 ADDRESS.	
	WR2	INTERRUPT VECTOR.	Channel B only.
	WR0	LOAD WR4 ADDRESS.	
	WR4	ISSUE PARAMETERS.	Parity Information, SYNC Mode, SDLC Mode, X1 Clock Mode.
	WR0	LOAD WR5 ADDRESS. RESET EXTERNAL/STATUS INTERRUPTS.	
	WR5	ISSUE PARAMETERS. SDLC-CRC, DATA TERMINAL READY.	
	WR0	LOAD WR3 ADDRESS.	
	WR3	ISSUE RECEIVE PARAMETERS:	Receive CRC Enable, Enter Hunt Mode, Auto Enables, Receive Character Length, Address Search Mode. Auto Enables enables the receiver to accept data only after CD* becomes active. Address Search Mode enables 8274 MPSC to match the message address with the programmed address or the global address.
	WR0	LOAD WR6 ADDRESS.	
INITIALIZE	WR6	SECONDARY ADDRESS FIELD.	This address is matched against the message address in an SDLC poll operation.
	WR0	LOAD WR7 ADDRESS.	

Table 4-19. SDLC Mode Receive Sequence (continued)

Function	Typical Program Steps	Comments
	Register Information Loaded	
INITIALIZE WR7	SDLC FLAG 01111110.	This flag detects the start and end of the frame in an SDLC operation.
WR0	LOAD WR1 ADDRESS. RESET EXTERNAL/STATUS INTERRUPTS.	
WR1	SET UP INTERRUPT SENSING.	Status Affects Vector, External Interrupt Enable, Receive Interrupt On First Character Only. In this interrupt mode, only the Address Field (1 character only) is transferred to the CPU. All subsequent fields (Control, Information, etc.) are transferred on a DMA basis. Status Affects Vector in Channel B only.
WR0	LOAD WR3 ADDRESS. ENABLE INTERRUPT ON NEXT RECEIVE CHARACTER.	Used to provide simple loop- back entry point for next transaction.
WR3	ISSUE PARAMETERS.	Receive Enable, Receive CRC Enable, Enter Hunt Mode, Auto Enables, Receiver Character Length, Address Search Mode.
IDLE MODE	EXECUTE HALT INSTRUCTION OR SOME OTHER PROGRAM.	SDLC Receive Mode is fully initialized and 8274 MPSC is waiting for the opening flag followed by a matching address field to interrupt the CPU.

4.10.4 8274 MPSC INTERRUPTS

The 8274 MPSC offers an elaborate interrupt scheme to provide fast interrupt response in real-time applications. As mentioned earlier, Channel B registers WR2 and RR2 contain the interrupt vector that points to an interrupt service routine in the memory. To service operations in both channels and to eliminate the necessity of writing a status analysis routine, the 8274 MPSC can modify the interrupt vector in RR2 so it points directly to one of eight interrupt service routines. This is done under program control by setting a program bit (WR1, D2) in Channel B called "Status Affects Vector". When this bit is set, the interrupt vector in WR2 is modified according to the assigned priority of the various interrupting conditions.

Transmit interrupts, Receive interrupts, and External/Status interrupts are the main sources of interrupts (refer to Figure 4-6). Each interrupt source is enabled under program control, with Channel A having a higher priority than Channel B, and with Receiver, Transmit and External/Status interrupts prioritized in that order within each channel. When the transmit interrupt is enabled, the CPU is interrupted by the transmit buffer becoming empty. When enabled, the receiver can interrupt the CPU in one of three ways:

- Interrupt on first receive character.
- Interrupt on all receive characters.
- Interrupt on a Special Receive condition.

Interrupt On First Character is typically used with the DMA mode. Interrupt On All Receive Characters has the option of modifying the interrupt vector in the event of a parity error. The Special Receive Condition interrupt can occur on a character or message basis (End Of Frame Interrupt in SDLC, for example). The Special Receive condition can cause an interrupt only if the Interrupt On First Receive Character or Interrupt On All Receive Characters mode is selected. In Interrupt On First Receive Character, an interrupt can occur from Special Receive conditions (except Parity Error) after the first receive character interrupt (example: Receive Overrun interrupt).

The main function of the External/Status interrupt is to monitor the signal transitions of the CTS* and CD* pins; however, an External/Status interrupt is also caused by a Transmit Underrun condition or by the detection of a Break (Asynchronous mode) or Abort (SDLC mode) sequence in the data stream. The interrupt caused by the Break/Abort sequence has a special feature that allows the 8274 MPSC to interrupt when the Break/Abort sequence is detected or terminated. This feature facilitates the proper termination of the current message, correct initialization of the next message, and the accurate timing of the Break/Abort condition in external logic.

To summarize, the 8274 can perform receive operations with or without generating interrupts. Transferring data without interrupts is used for purely polled operation applications and for off-line conditions. There are three interrupt modes available to the 8274 for data transfers: interrupt on first character only; interrupt on every character; and special receive conditions interrupt. The following paragraphs briefly describe the three types of interrupts.

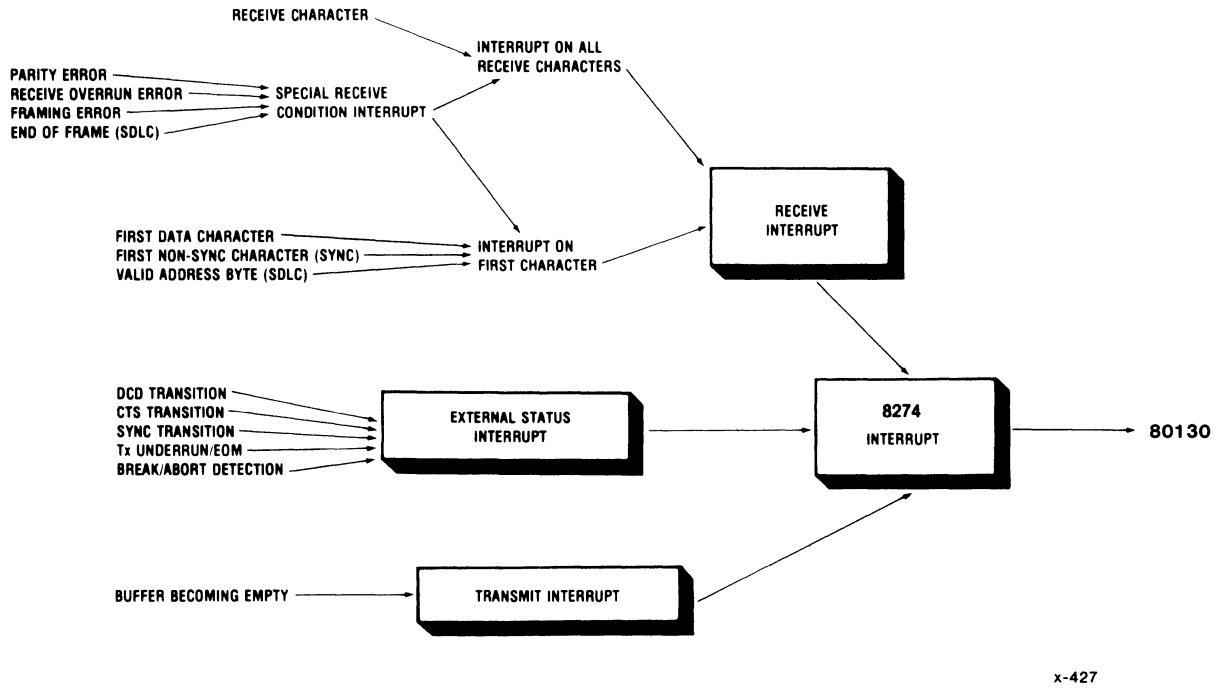


Figure 4-6. Interrupt Structure

Interrupt on First Character Only. Interrupt on First Character Only mode is normally used to start a polling loop or to start a DMA transfer using the RxDRQ signal. The 8274 MPSC generates an interrupt on the first character and thereafter only interrupts after a Special Receive Condition is detected. When using DMA, the DMA channel can be enabled during the interrupt routine (interrupt on first character only) to handle the remaining characters. The Interrupt On First Character mode can be reinitialized in the 8274 MPSC by issuing the Enable Interrupt On Next Receive Character command (WR0; D5, D4, D3) which allows the next character received to generate an interrupt. Parity Error indications from the 8274 MPSC do not cause interrupts, but End of Frame (SDLC operation) and Receive Overrun do cause interrupts in the Interrupt On First Character mode. If the external status interrupts (WR1; D0) are enabled, the 8274 MPSC can generate an interrupt any time the CD* input to the 8274 MPSC changes state.

Interrupt On Every Character. Operation of the 8274 MPSC in an Interrupt On Every Character mode generates an interrupt whenever a character enters the receive buffer. Errors and Special Receive Condition interrupts generate a special vector if the Status Affects Vector (WR1; D2) is selected. Also the Parity Error interrupt may be programmed (WR1; D4, D3) not to generate the special vector while the 8274 MPSC is operating in the Interrupt On Every Character mode.

Special Receive Condition Interrupt. The Special Receive Condition interrupt can only occur while in the Receive Interrupt On First Character Only or the Interrupt On Every Receive Character modes. The Special Receive Condition interrupt is caused by the Receive Overrun (RR1; D5) error condition. The error status reflects an error in the current word in the receive buffer, in addition to any Parity or Overrun errors since the last Error Reset (WR0; D5, D4, D3). The Receive Overrun and Parity Error status bits are latched and can only be reset by the Error Reset command (WR0; D5, D4, D3).

4.10.5 8274 MPSC DMA OPERATION

Modes 1 and 2 programmed in WR2 (CHA) places Channel A or Channels A and B in the DMA mode. In these modes, a DMA REQUEST signal is generated for both receive and transmit data conditions. There are a possible four request signals RxDRQA, TxDRQA, RxDRQB and TxDRQB.

Because there are only two DMA channels, you must multiplex the DMA request signals by programming the DMA Multiplexer under software control or route some of the DMA requests signals to the interrupt stake pins at the 80130. For example, in order to use DMA to transmit data across channels A and B, you need to multiplex TxDRQA to one 80186 DMA channel and TxDRQB to the other. However, because both channels are in the DMA mode, no 8274 interrupt will be generated for a Receive Buffer Full condition. (Although an interrupt will be generated for special conditions.) In this case, RxDRQA and RxDRQB will go active for the corresponding receive data condition. These signals can be connected to interrupt pins on the 80130 to generate an interrupt when the receive characters are available.

4.10.6 8274 MPSC INITIALIZATION

Table 4-19 list a series of parameters that initialize operation of the 8274 MPSC and the interfaces at Connector J1 and Connector J2 for operation in the as-shipped configuration. The iSBC 186/03 board supports all modes of operation of the 8274, however, does not allow the use of the external sync detect features of the chip. Refer to the Intel Microprocessor and Peripheral Handbook (1982 version or later) for more information on the programming the 8274 MPSC device.

Table 4-20. 8274 Parameter Sequence for Initialization

Sequence	Port* Data	Functions Performed In 8274 MPSC
1.	0DEH 004H	Port Number for Channel B Control. Data; point to WR4B.
2.	0DEH 04CH	Port Number for Channel B Control. Data; 16X clock, 8-bit sync, 2 stop bits, odd parity, parity disabled.
3.	0DEH 001H	Port Number for Channel B Control. Data; point to WR1B.
4.	0DEH 01CH	Port Number for Channel B Control. Data; disable wait, wait on transmit, interrupt on all receive, parity does not affect vector, status affects vector, no transmit interrupt, no external interrupt.
5.	0DCH 002H	Port Number for Channel A Control. Data; point to WR2A.
6.	0DCH 034H	Port Number for Channel A Control. Data; pin-10 is RTS, vectored interrupt, 8086 mode, receive priority, both channels interrupt mode.
7.	0DEH 002H	Port Number for Channel B Control. Data; point to WR2B.
8.	0DEH 030H	Port Number for Channel B Control. Data; vector base address.
9.	0DEH 003H	Port Number for Channel B Control. Data; point to WR3B.
10.	0DEH 0C1H	Port Number for Channel B Control. Data; receive 8 bits/char, no auto enables, no halt mode, no receive CRC, no address search, no sync, enable receiver.
11.	0DEH 005H	Port Number for Channel B Control. data; point to WR5B.
12.	0DEH 0FAH	Port Number for Channel B Control. data; DTR on, transmit 8 bits/char, no break, enable transmitter, no SDLC CRC, RTS on, no transmit CRC.

***

CHAPTER 5. SERVICE INFORMATION

5.1 INTRODUCTION

This chapter provides a list of service diagrams and service and repair assistance instructions for the iSBC 186/03 Single Board Computer.

5.2 SERVICE DIAGRAMS

The parts location diagram and schematic diagram for the iSBC 186/03 board are provided in Figures 5-1 and 5-2, respectively. On the schematic diagram, a signal mnemonic that ends with an asterisk (e.g., ALE*) is active low. Conversely, a signal mnemonic without an asterisk (e.g., ALE) is active high.

5.3 SERVICE AND REPAIR ASSISTANCE

United States customers can obtain service and repair assistance by contacting the Intel Product Service Marketing Administration in Phoenix, Arizona. Customers outside the United States should contact their sales source (Intel Sales Office or Authorized Distributor) for service information and repair assistance.

Before calling Product Service Marketing Administration, you should have the following information available:

a. The date on which you received the product.

b. The complete model number (including dash number) and serial number for the product. These numbers are stamped onto the printed circuit boards.

c. Your shipping and billing addresses.

e. A purchase order number for billing purposes if your Intel product warranty has expired.

f. Any extended warranty agreement information, if applicable.

Use the following numbers for contacting the Intel Product Service Marketing Administration group:

Regional Telephone Numbers:

Western Region: 602-869-4862
Midwestern Region: 602-869-4392
Eastern Region: 602-869-4045
International: 602-869-4391

TWX Number:
910 - 951 - 1330
910 - 951 - 0687

Always contact the Product Service Marketing Administration group before returning a product to Intel for 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 you are returning the product because of damage sustained during shipment or if the product is out of warranty, a purchase order is required before Intel can initiate the repair.

In preparing the product for shipment to Intel, use the original factory packing material, if possible. If this material is not available, wrap the product in cushioning material such as Air Cap TH-240, manufactured by the Sealed Air Corporation, Hawthorne, N.J. Then enclose in a heavy duty corrugated shipping carton, and label "FRAGILE" to ensure careful handling. Ship only to the address specified by the Intel Product Service Marketing Administration personnel.

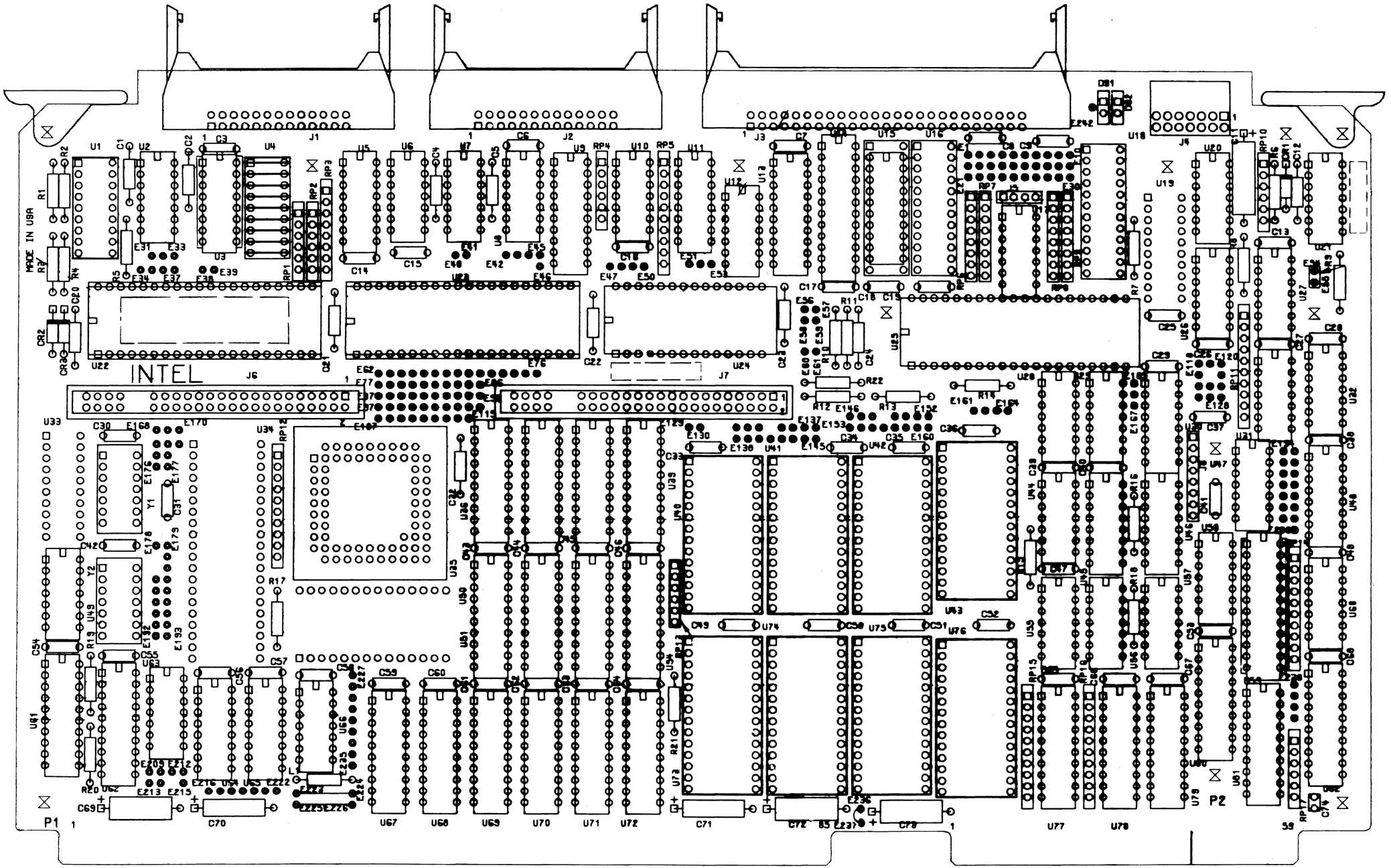
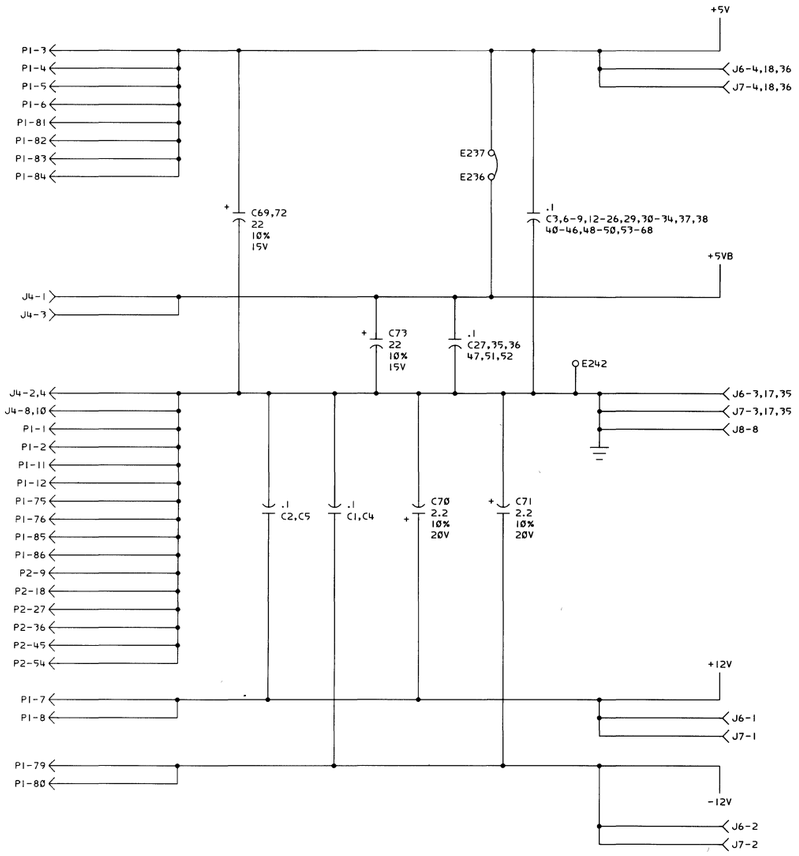


Figure 5-1. Parts Location Diagram

REVISIONS							
REV	DESCRIPTION	DFT	DATE	CHK	DATE	APVD	DATE
A	ECO 40-4132	B.TRong	12/6/83	KA	12/7/83	[illegible]	1/3/84



NOTE; UNLESS OTHERWISE SPECIFIED:

1. CAPACITANCE VALUES ARE IN MICROFARADS, +80%, -20%, 50V.
2. RESISTANCE VALUES ARE IN OHMS, 1/4W, 5%.
3. AN ASTERISK FOLLOWING A SIGNAL NAME DENOTES NEGATION.

[4] PROGRAMMED DEVICE.
[5] CUSTOMER SUPPLIED AND INSTALLED PART.
[6] COMPONENTS NOT SUPPLIED.

DEVICE TABLE

REFERENCE DESIGNATOR	DEVICE TYPE	POWER PINS GND	+5V	+5VB	+12V	-12V	UNUSED LOGIC ELEMENT OUTPUT PINS
U5	7486	8	16				
U3	7487	8	16				
U66,60	74S00	7	14				U60-3,6
U27,58	74S04	7	14				U27-12, U58-8,12
U29,6	74LS04	7	14				U29-8
U28	74S08	7	14				
U20	74LS10	7	14				
U57	74LS11	7	14				
U32	74LS14	7	14				U32-6,8,10,12
U10	7417	7	14				
U12	74LS20	7	14				
U55	74LS20	7		14			
U21	74S30	7	14				
U44	74S32	7	14				
U47	74F32	7	14				
U31	74LS32	7		14			
U17	7438	7	14				
U56	74S74	7	14				U56-8,9
U49	74LS74	7	14				
U45	74LS123	8	16				U45-5,12
U63	74128	7	14				
U48,11	74S139	8	16				
U46	74LS139	8		16			
U30	74S175	8	16				
U69,68,67	74LS240	10	20				
U79,80,81	74LS244	10	20				
U15,36,37,77,78	74LS245	10	20				
U26	74S260	7	14				
U38,39,53	74S373	10	20				
U64	74368	8	16				
U65	74LS368	8	16				
U2,7	75188	7			14	1	
U8	75189A	7	14				
U23	80130	20,1	40				
U35	80186	26,60	9,43				
U25	8255A-5	7	26				
U24	8259A	14	28				
U22	8274	20	40				
U54	8283	10	20				
U61	8289	10	20				
U51,62	8288	10	20				
U70,72,71	8287	10	20				
U52	PAL14L4	10	20				
U9	PAL16R4	10	20				
U13	PAL16L8A	10	20				
U82	PAL16L8A	10	20				
U59	PAL20L8	12	24				
U14	PAL20L8	12	24				

[4]

[4]

iLBX ADDRESS OPTIONS

ADDRESS (HEX)	JUMPERS E42-E43	E44-E45
0-7FFFF	OUT	IN
0-BFFFF	IN	OUT
0-DFFFF	OUT	OUT

iSBX DECODE OPTIONS

iSBX CONNECTOR	DATA BUS SIZE	JUMPERS E47-E48	E49-E50
iSBX1 (J7)	8 BIT	OUT	OUT
	16 BIT	IN	OUT
iSBX2 (J6)	8 BIT	OUT	OUT
	16 BIT	OUT	IN

PROM JUMPER TABLE

DEVICE TYPE	START ADDRESS (HEX)	JUMPERS E195-E197	E194-E196
2764	F8000	OUT	OUT
27128	F0000	OUT	IN
27256	E0000	IN	OUT
27512	C0000	IN	IN

REF DESIGNATIONS

LAST USED	NOT USED
U82 DS2	U19,33
C74	
R22	
RP17	
J8	
P2	
E242	
Y2	
CR3	

QUANTITY PER DASH NO.

UNLESS OTHERWISE SPECIFIED:
1. DIMENSIONS ARE IN INCHES.
2. BREAK ALL SHARP EDGES.
3. DO NOT SCALE DRAWING.
4. TOLERANCES: ANGLES ±, XX±, XXX±

ITEM NO. | DESCRIPTION

PARTS LIST

SIGNATURE	DATE
DRN BY R.TRong	12/6/83
CHK BY [illegible]	12/7/83
ENGR APVD [illegible]	12/8/83
APVD	
APVD	

intel® 3065 BOWERS AVE SANTA CLARA CALIF. 95051

TITLE: SCHEMATIC DIAGRAM iSBC 186/03

SIZE	DCLC	GL CODE	RLSE LVL	DOCUMENT NUMBER	REV
D	40	0100	—	146418	A

SCALE NONE | SHEET 1 OF 13

109515

Figure 5-2. Schematic Diagram (Sheet 1 of 13)

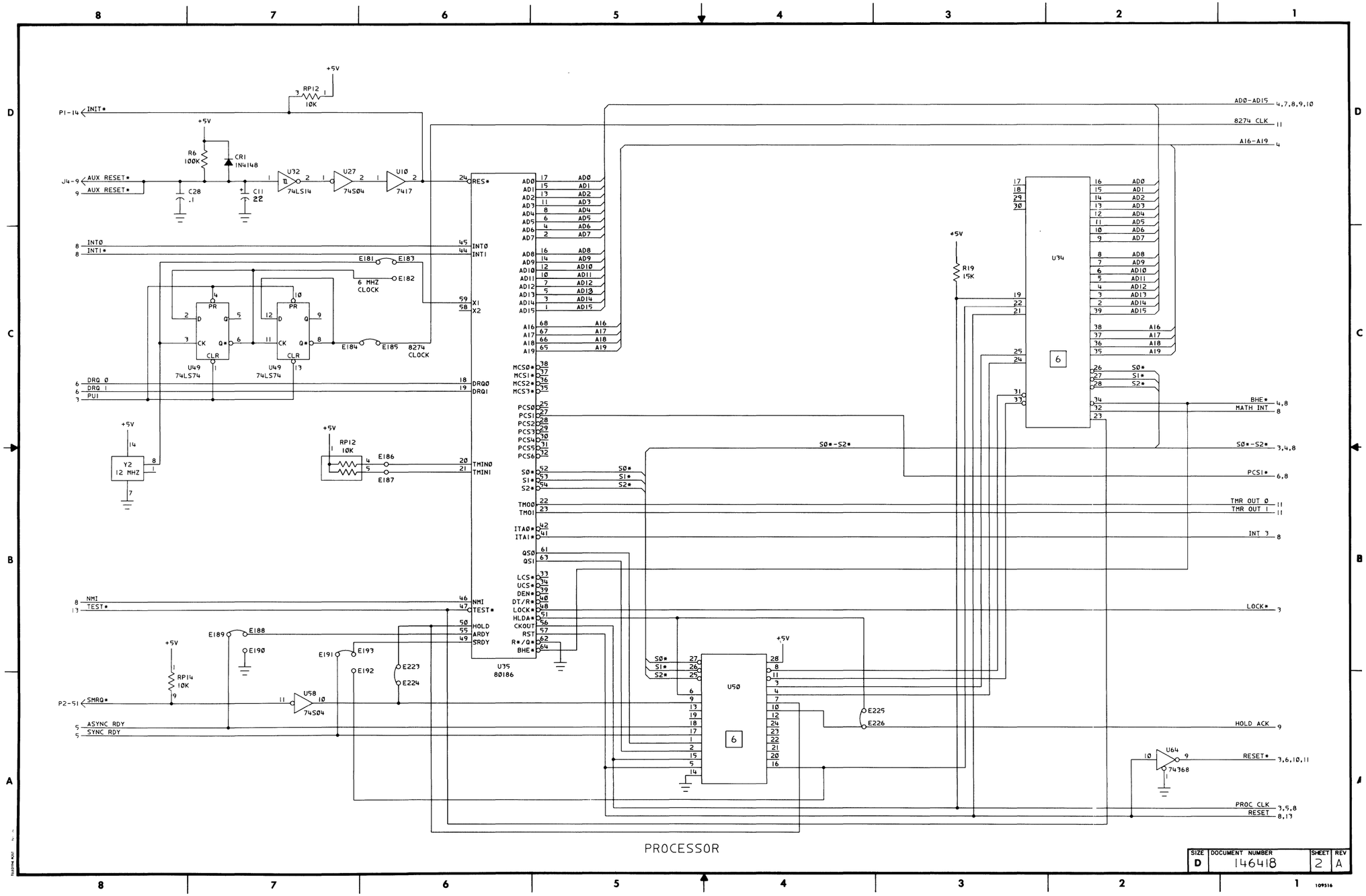


Figure 5-2. Schematic Diagram (Sheet 2 of 13)

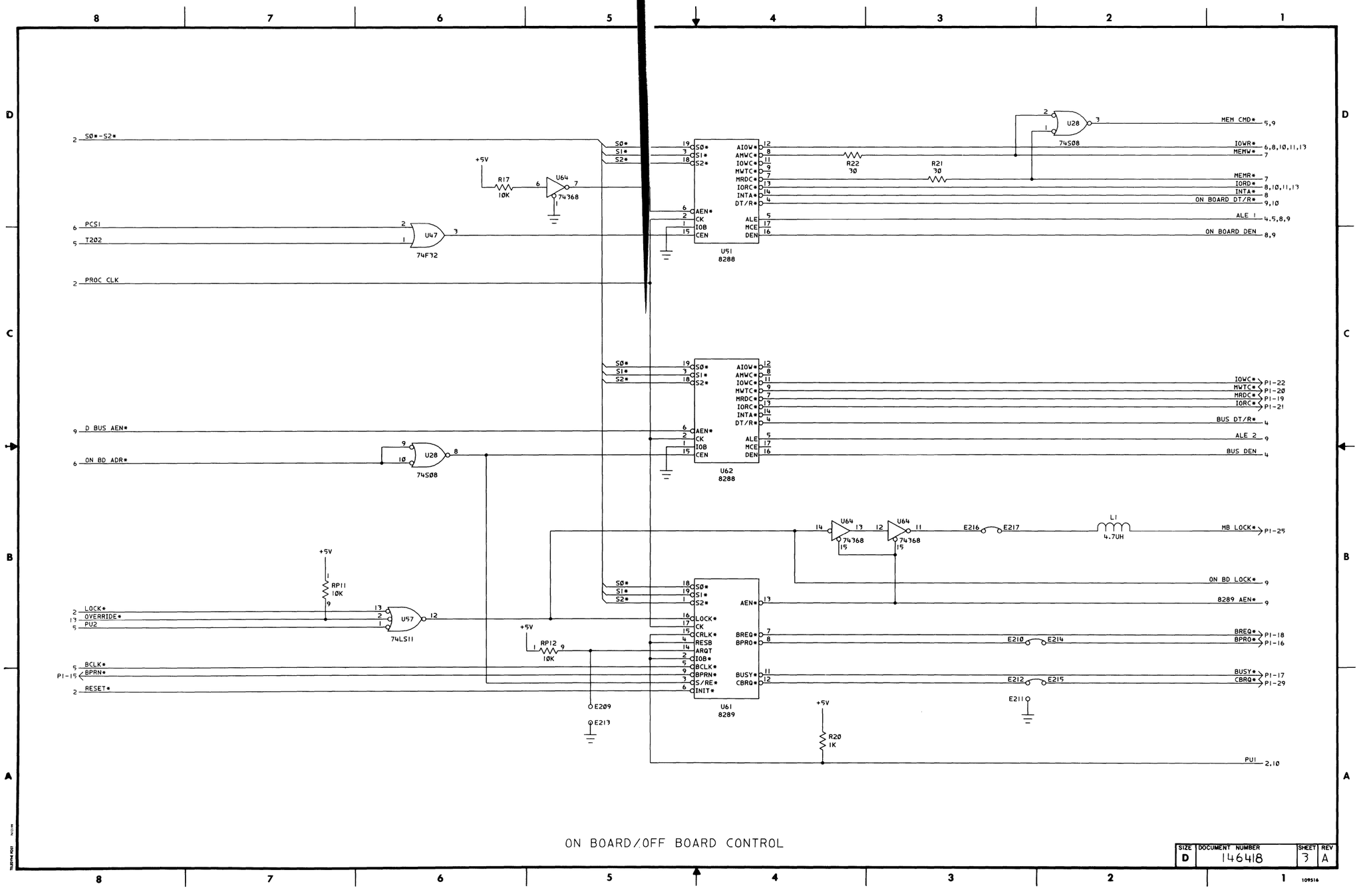


Figure 5-2. Schematic Diagram (Sheet 3 of 13)

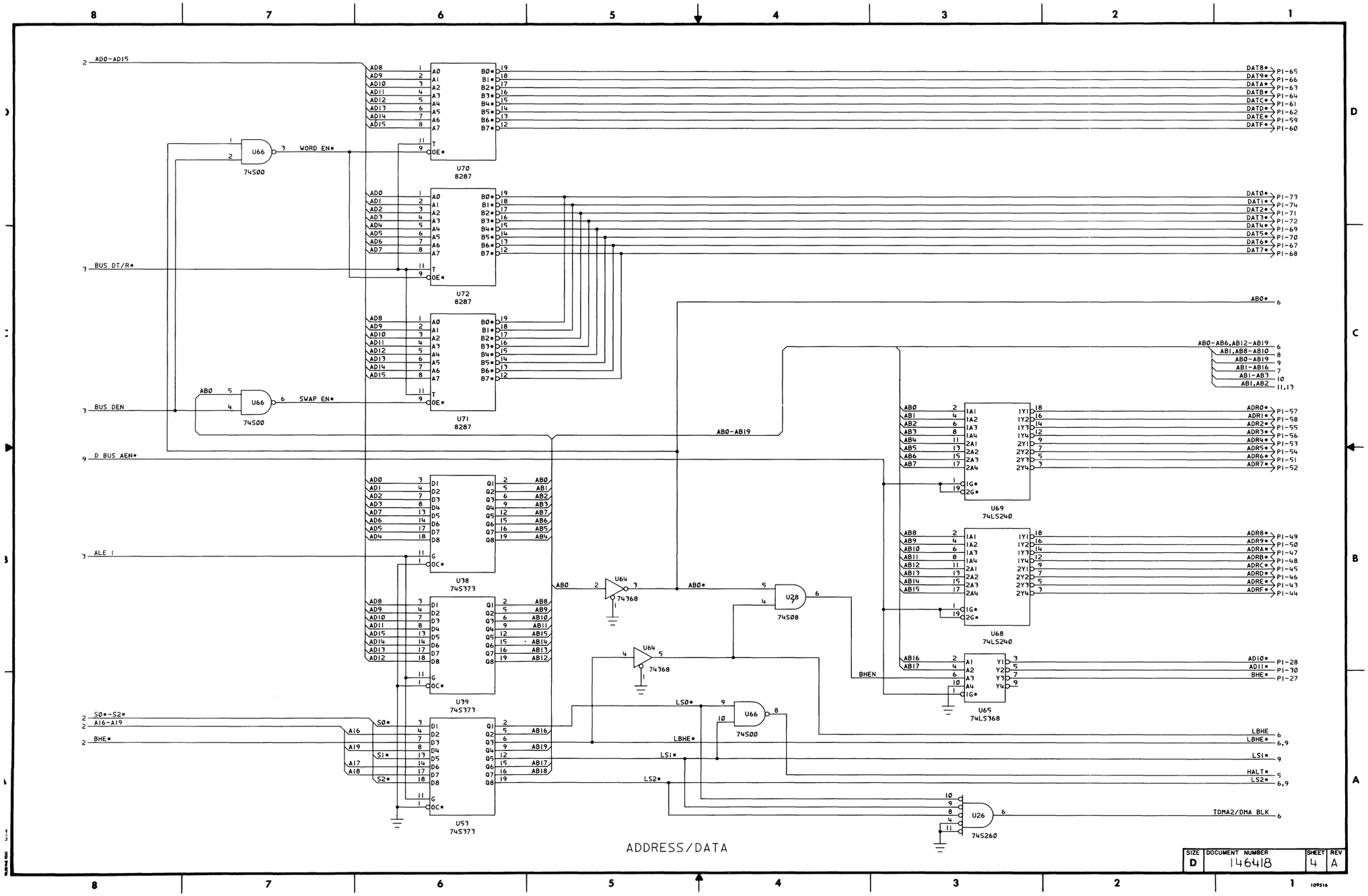


Figure 5-2. Schematic Diagram (Sheet 4 of 13)

WAIT STATES	RAM	PROM
0	NONE	NONE
1	124-125	127-128
2	124-123	126-127

READY LOGIC/MB CLOCK

SIZE	DOCUMENT NUMBER	SHEET	REV
D	146418	5	A

Figure 5-2. Schematic Diagram (Sheet 5 of 13)

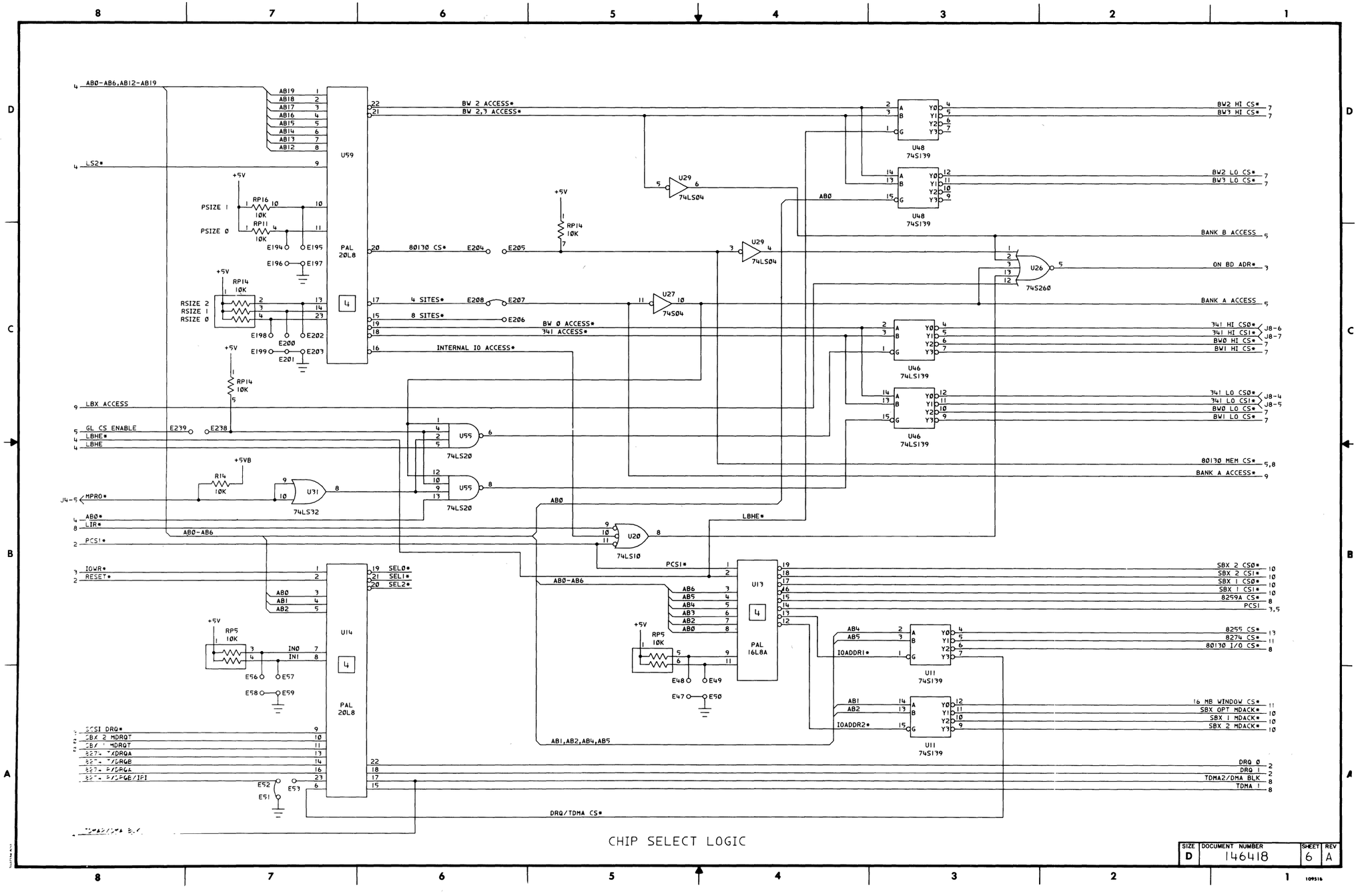


Figure 5-2. Schematic Diagram (Sheet 6 of 13)

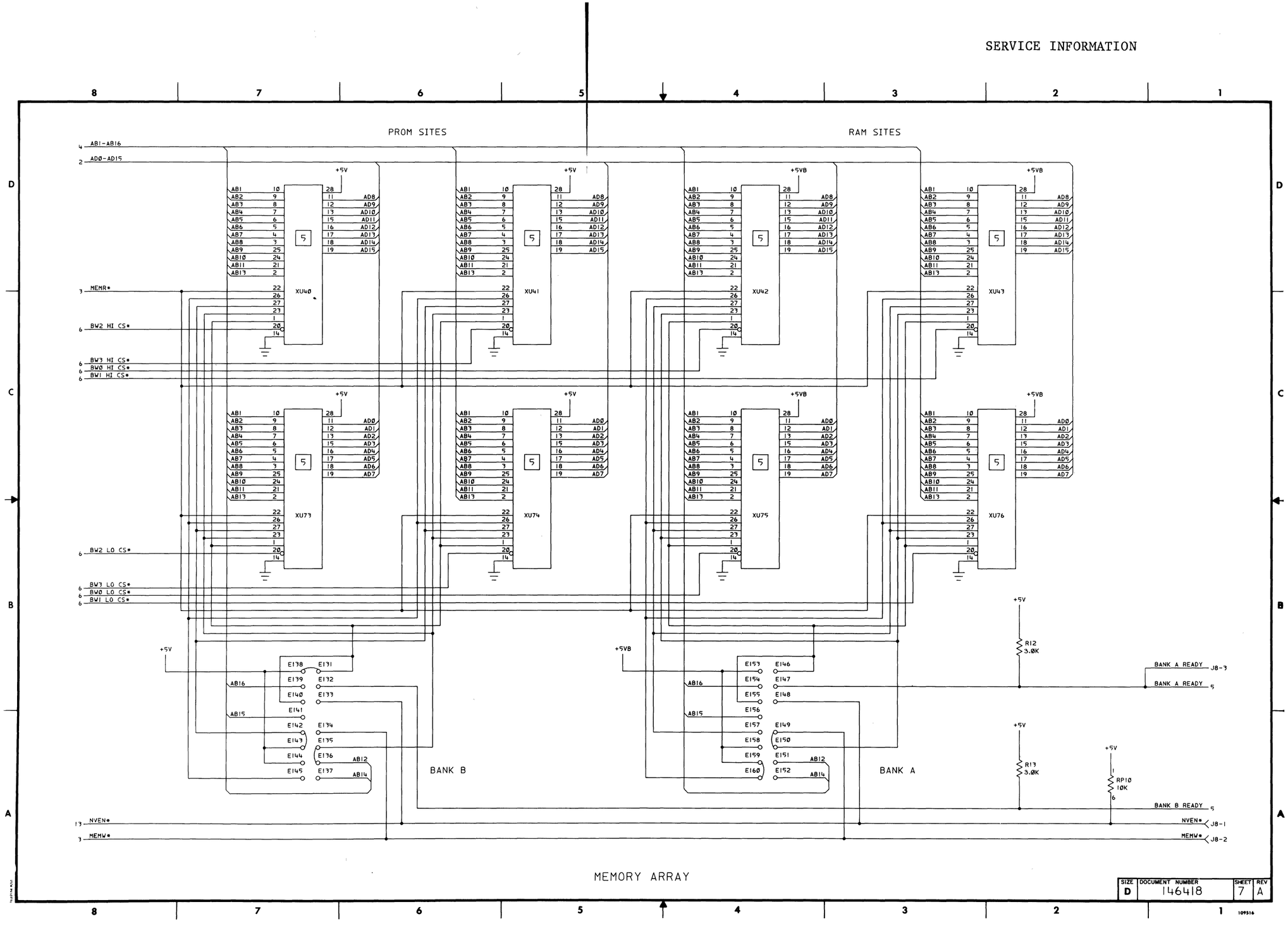


Figure 5-2. Schematic Diagram (Sheet 7 of 13)

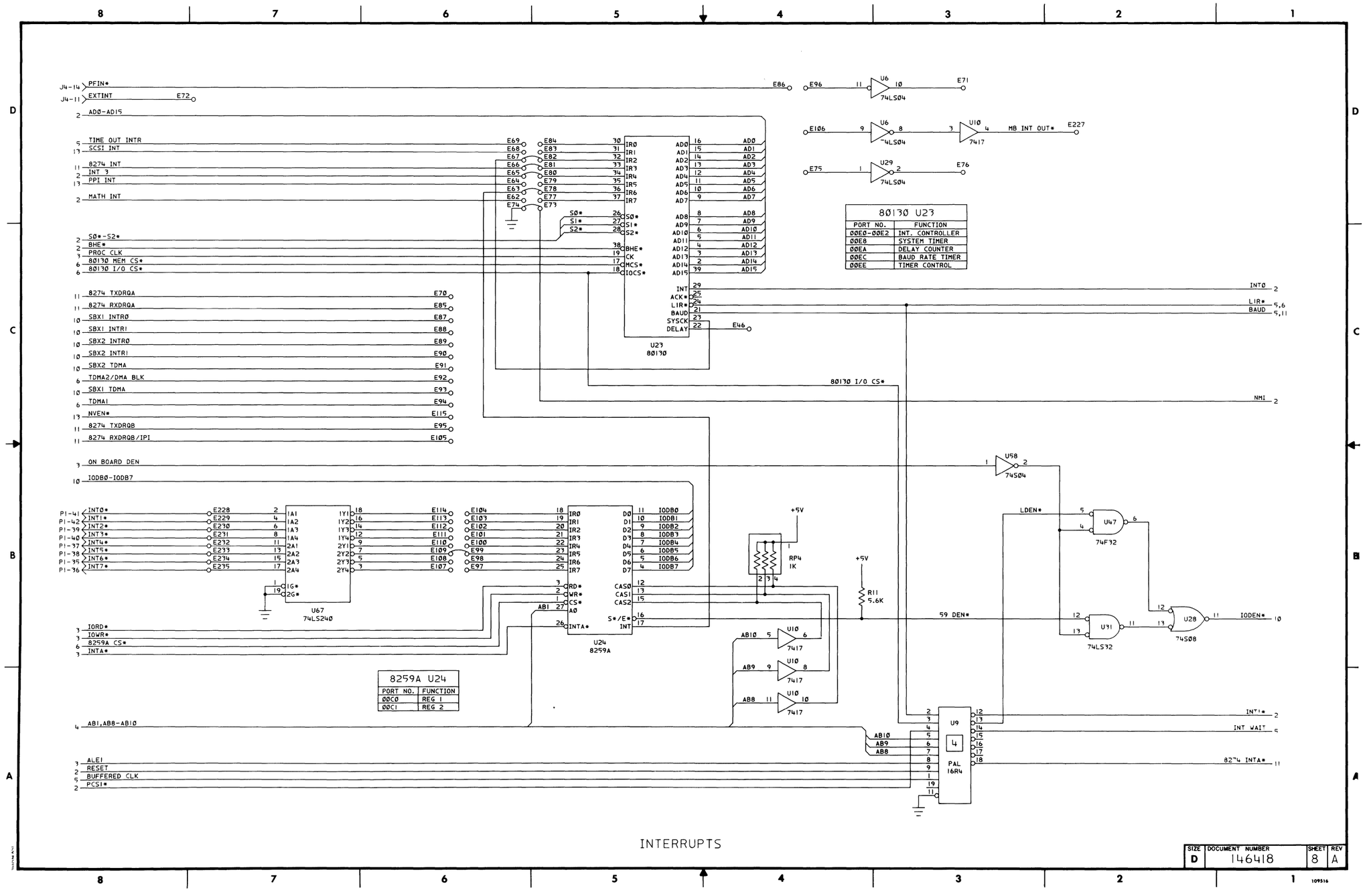


Figure 5-2. Schematic Diagram (Sheet 8 of 13)

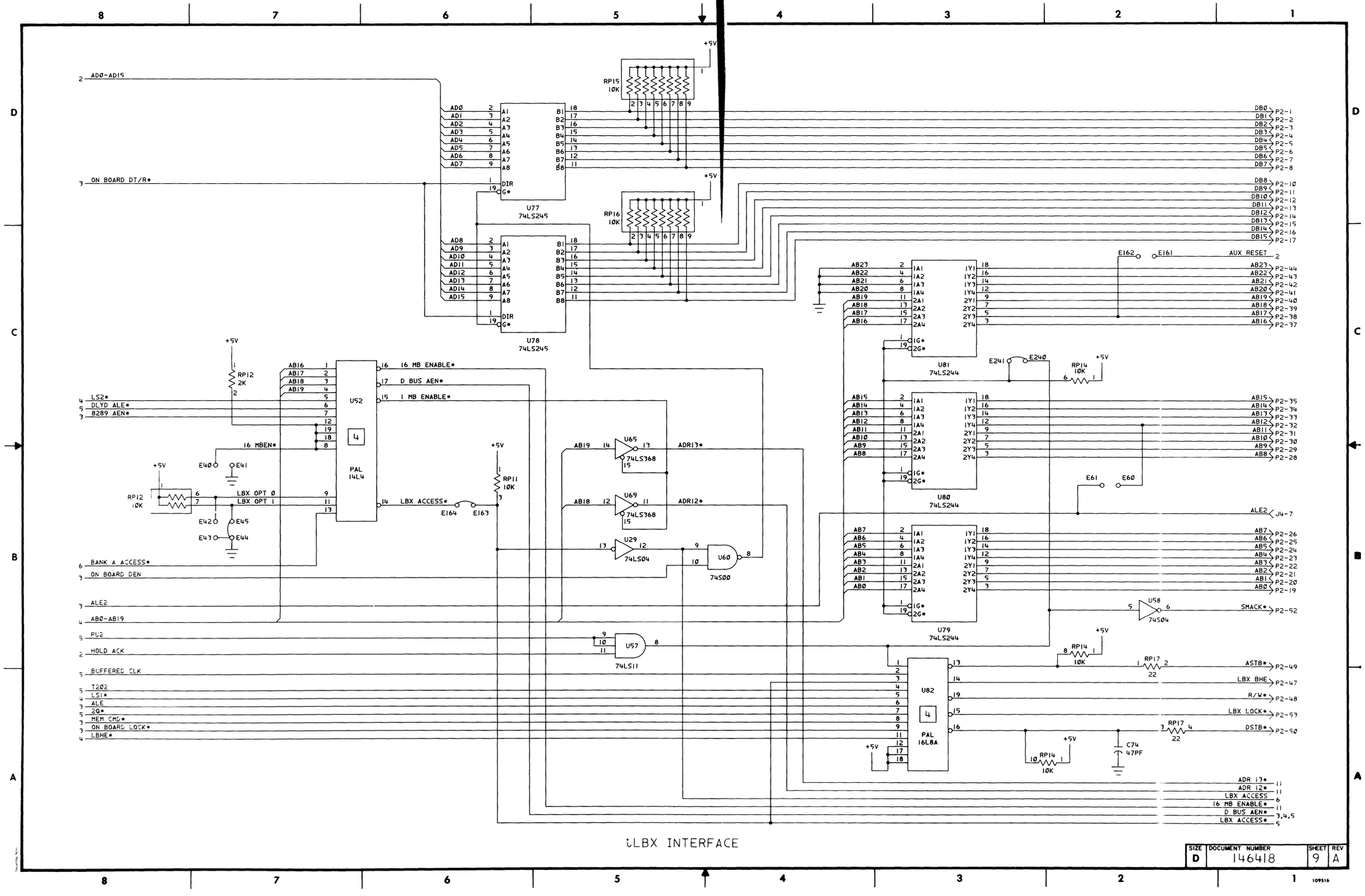


Figure 5-2. Schematic Diagram (Sheet 9 of 13)

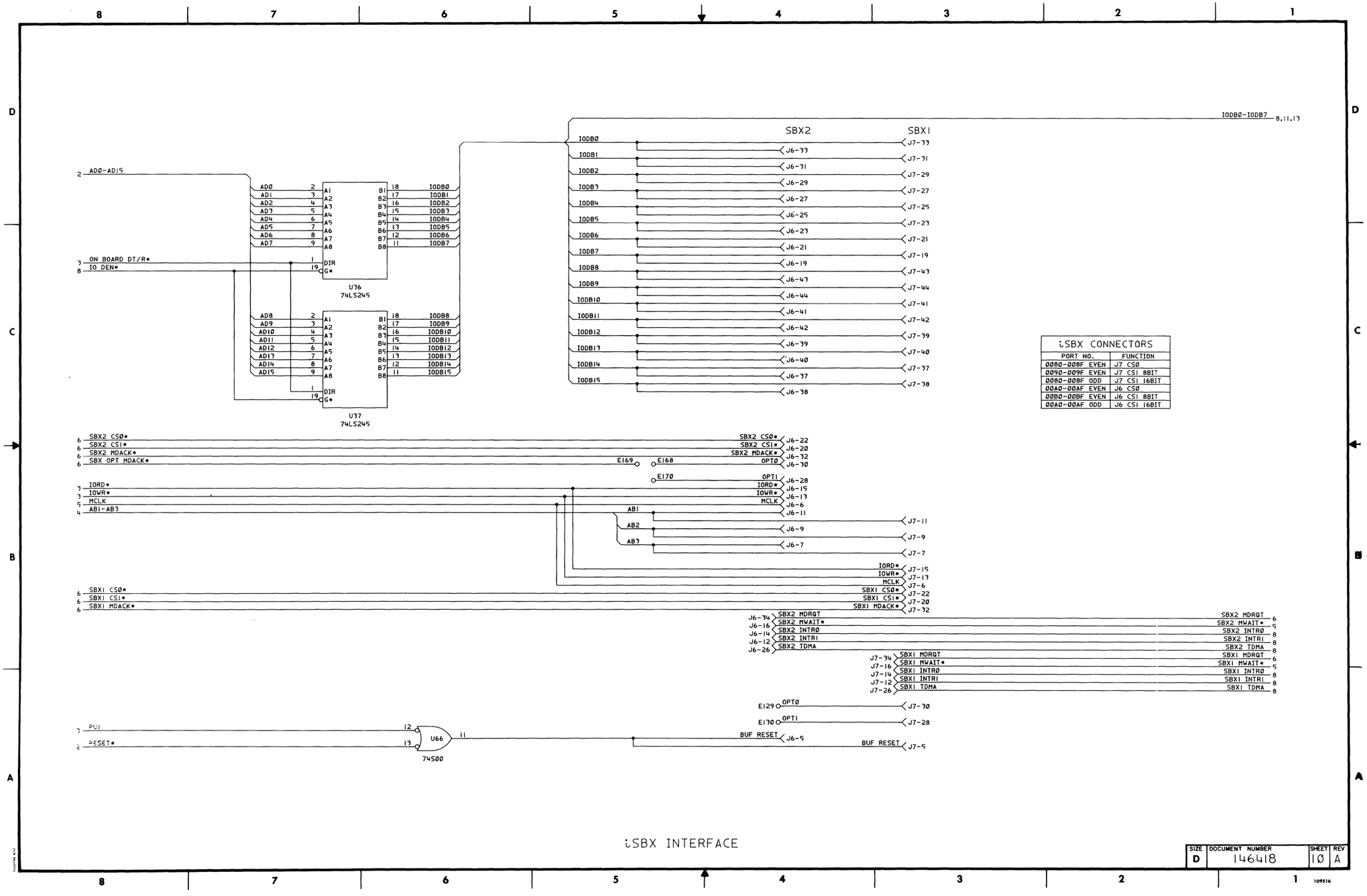


Figure 5-2. Schematic Diagram (Sheet 10 of 13)

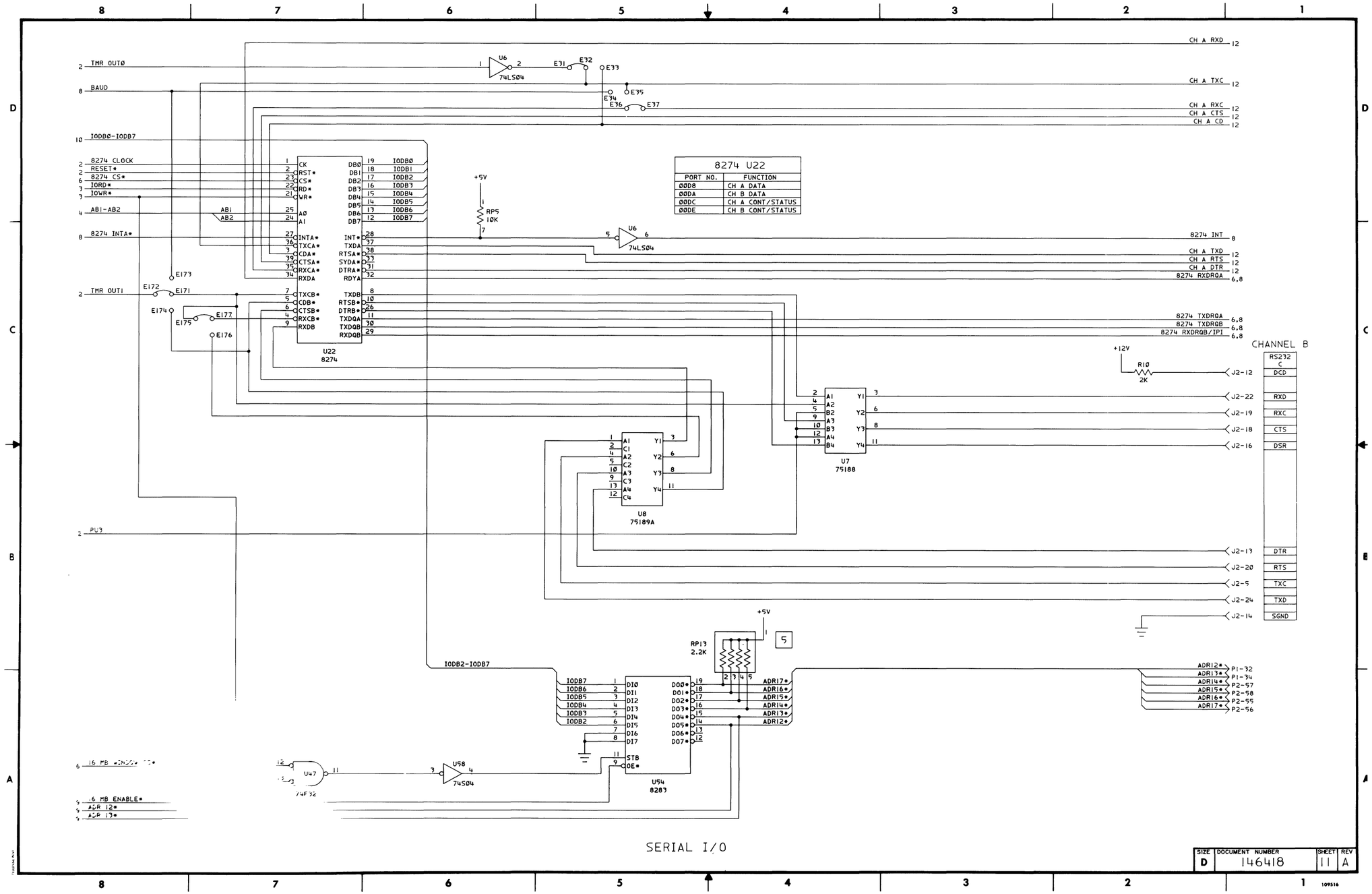


Figure 5-2. Schematic Diagram (Sheet 11 of 13)

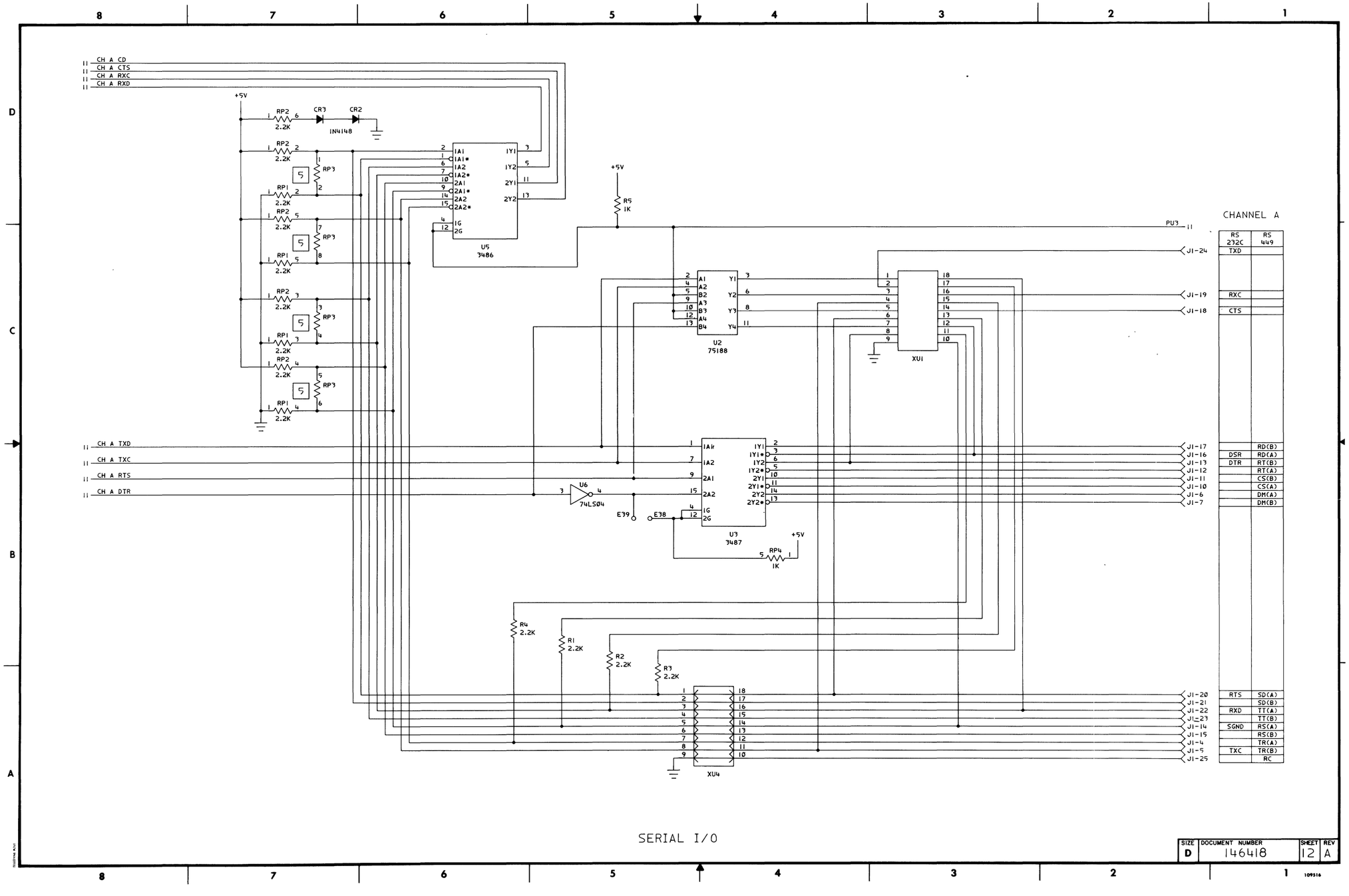


Figure 5-2. Schematic Diagram (Sheet 12 of 13)

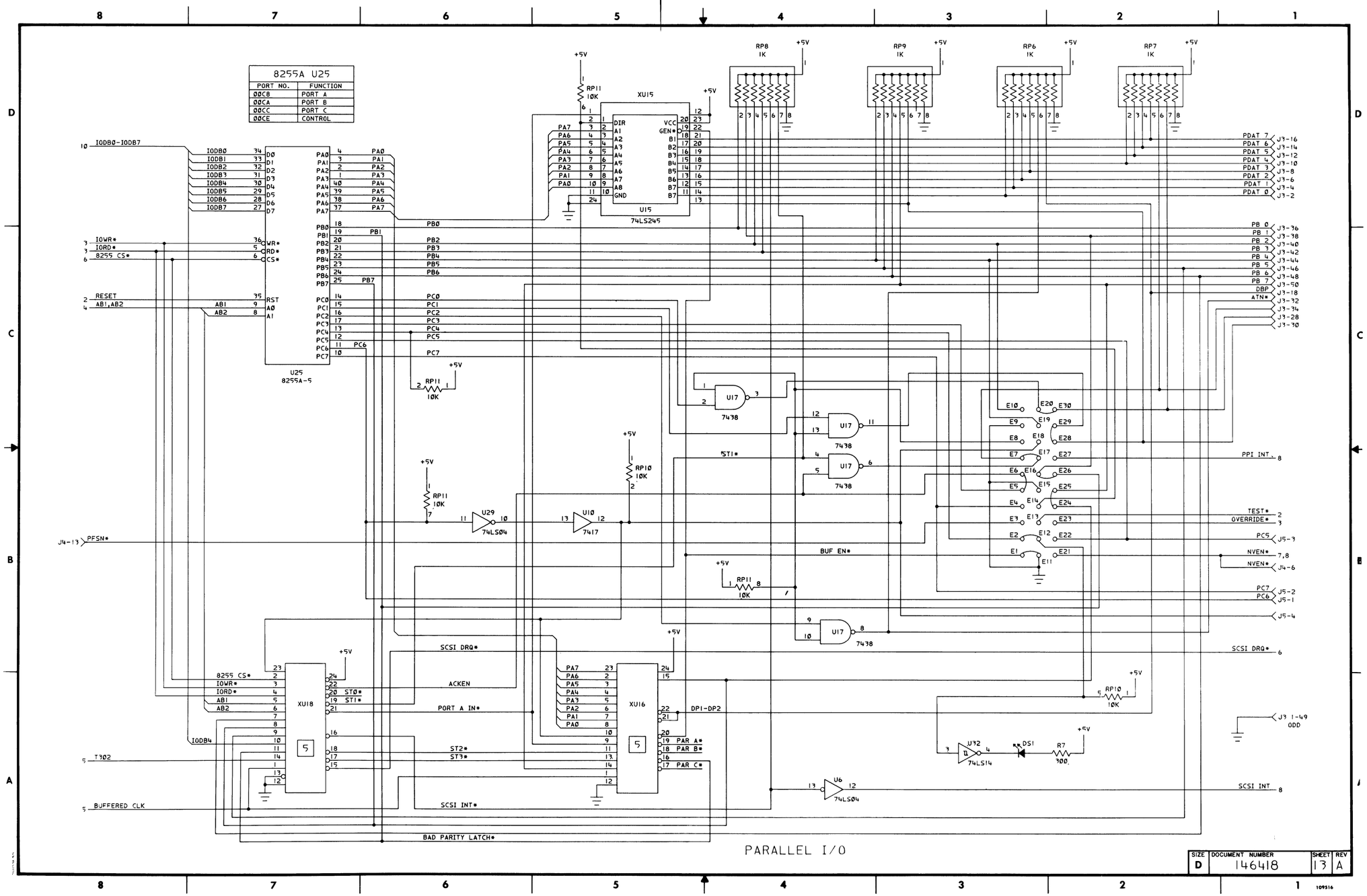


Figure 5-2. Schematic Diagram (Sheet 13 of 13)

***

APPENDIX A. JUMPER LISTS FOR THE iSBC® 186/03 BOARD

This appendix provides an overview of the jumpers on the iSBC 186/03 board. Table A-1 lists the factory installed, default jumper connections. Table A-2 lists the jumper posts and gives a brief description of their functions. Refer to Chapter 2 for additional information on the jumper functions. Figure A-1 shows the physical location of the jumper posts on the board.

Table A-1. Installed Jumpers (As-Shipped Configuration)

E1-E11	E2-E12	E5-E6	E7-E17
E16-E26	E20-E30	E24-E25	E28-E29
E31-E32	E36-E37	E44-E45	E51-E52
E54-E55	E63-E78	E65-E80	E66-E81
E67-E82	E73-E74	E99-E109	E118-E119
E131-E138	E135-E136	E142-E143	E149-E150
E159-E160	E163-E164	E171-E172	E175-E177
E165-E166	E179-E180	E181-E183	E184-E185
E188-E189	E191-E193	E207-E208	E210-E214
E212-E215	E216-E217	E218-E219	E220-E221
E223-E224	E225-E226	E236-E237	E240-E241

Table A-2. Numerical List of Jumpers and their Functions

Jumper Number	Function
E1 to E11	Enables the Port A transceiver.
E2 to E12	Provides connection for bit 4 of Port C to drive the diagnostic LED.
E3	Provides connection for the PFSN* signal from J4 pin 13 to the parallel port jumper matrix.
E4	Provides connection for bit 7 of Port C to the parallel port jumper matrix.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E5	Provides connection for bit 3 of Port C to the parallel port jumper matrix.
E5 to E6 and E7 to E17	Routes bit 3 of Port C to connector J3 pin 34 via U17, a NAND (7438) gate.
E6	Provides connection to input of 7438 gate from the parallel port jumper matrix.
E7	Provides connection to J3 pin 34.
E8	Provides connection to enable or disable three of the four 7438 NAND gates from the parallel port jumper matrix.
E9	Provides connection to Ground for the parallel port jumper matrix. May be used to ground Port B input lines for use as configuration inputs.
E10	Provides connection to bit 2 of Port B and connector J3 pin 40 from the parallel port jumper matrix.
E11	Provides connection to Ground for the parallel port jumper matrix. May be used to ground Port B input lines for use as configuration inputs.
E12	Provides connection to Diagnostic LED.
E13	Provides connection to TEST input of the 80186. Grounded to pin E15 for iRMX Operating System applications without the 8087 math coprocessor.
E14	Provides connection to the direction input of the Port A transceiver.
E15	Provides connection to Ground for the parallel port jumper matrix. May be used to ground Port B input lines for use as configuration inputs.
E16 to E26	Provides connection from bit 1 of Port B to connector J3 pin 38.
E17	Provides connection from output of 7438 gate.
E18	Provides connection from the output of Port C bit 6 and PAL sockets to the parallel port jumper matrix.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E19	Provides connection to Port B bit 4 and connector J3 pin 44.
E20	Provides connection from the output of the 7438 NAND gate connected to the output of PC0.
E20 to E30	Provides connection from bit 0 of Port C to connector J3 pin 28.
E21	Provides connection to the NVEN* signal in the Parallel Port matrix.
E22	Provides connection to bit 5 of Port C.
E23	Provides connection for the OVERRIDE* signal to generate the LOCK* signal.
E24	Provides connection for bit 7 of Port B and for the PAL sockets.
E24 to E25	Provides connection from connection J3 pin 50 to bit 7 of Port B.
E25	Provides connection to connector J3 pin 50.
E26	Provides connection to bit 1 of Port B.
E27	Provides connection for the PPI INT signal to the interrupt jumper matrix at E64.
E28	Provides connection to connector J3 pin 30.
E28 to E29	Provides connection of output from bit 1 of Port C to J3 pin 30.
E29	Provides connection to output from bit 1 of Port C.
E30	Provides connection to connector J3 pin 28.
E31 to E32	Routes TMR OUT 0 signal from the 80186 as TxC to Channel A of the 8274 MULTI PROTOCOL SERIAL CONTROLLER (MPSC).
E32 to E33	Allows the 8274 channel A transmit clock to be derived from the J1 connector.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E34	Routes the output from the 80130 as a clock input to Channel A of the 8274.
E35 to E36	Connects Channel A receive clock to the transmit clock.
E36 to E37	Routes Receive clock (Channel A) to the 8274 MPSC from the TxCA line on connector J1.
E38 to E39	Enables Multidrop capability by allowing Channel A DTR to disable/enable the 3487 line driver for Channel A.
E40 to E41	Enables 16 Mbyte addressing capability when installed.
E42 to E43	Defines the iLBX address space in conjunction with jumper E44 and E45.
E44 to E45	Defines the iLBX address space in conjunction with jumper E42 and E43.
E46	Provides connection from the DELAY output of the 80130 TIMER 1.
E47 to E48	This jumper defines the size of the iSBX 1 local bus. When the jumper is removed, the 8-bit address mode is supported. When the jumper is installed, the 16-bit address mode is supported.
E49 to E50	This jumper defines the size of the iSBX 2 local bus. When the jumper is removed, the 8-bit address mode is supported. When the jumper is installed, the 16-bit address mode is supported.
E51 to E52 §	Provides ground to RxDRQB/IPI for the interrupt mode of the 8274.
E52 to E53	Routes 8274 RxDRQB signal (DMA Request) to the input of the DMA Multiplexer PAL for 8274 Channel B receive character DMA request.
E54 to E55	Routes the output from the fail-safe timer to the ASYNC READY gate. It enables the 80186 to resume processing after 10 ms have elapsed without a READY response.
E56 to E58	Enables the DMA Multiplexer PAL to select a subset of DMA requests as the DRQ0 input to the 80186.
E57 to E59	Enables the DMA Multiplexer PAL to select a subset of DMA requests as the DRQ1 input to the 80186.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E60 to E61	Routes ALE signal out to MULTIBUS connector P2. This jumper must not be installed when using the iSBC 186/03 board in an iLBX interface.
E62	Not Used.
E63 to E78 §	Routes interrupt from the 8259A slave interrupt controller to IR6 of the 80130 master interrupt controller.
E64	Routes PPI (parallel Port Interface) interrupt to the interrupt jumper matrix.
E65 to E80 §	Routes the 80186 slave PIC interrupt request INT 3 to IR4 of the 80130 master interrupt controller.
E66 to E81 §	Routes the interrupt request from the slave PIC internal to the 8274 Serial I/O controller to IR3 of the 80130 master interrupt controller. Reserved for 8274 vectored interrupts only.
E67 to E82 §	Routes the SYSTICK timer 0 output from the 80130 to IR2 of the 80130 master interrupt controller.
E68	Routes SCSI interrupt request signal from the parallel port to the interrupt jumper matrix.
E69	Routes the TIME OUT interrupt request to the interrupt jumper matrix.
E70	Routes the 8274 TxDRQA signal to the interrupt jumper matrix.
E71	Provides connection from the output from a spare inverter to interrupt jumper matrix.
E72	Routes the external interrupt request from connector J4 to the interrupt jumper matrix.
E73 to E74	Inhibits the Non-maskable interrupt (NMI) to the 80186 processor.
E75	Provides connection to the input of a spare inverter from the interrupt jumper matrix.
E76	Provides connection to the output of a spare inverter in the interrupt jumper matrix.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E77 thru E84	Provides connections to the inputs of the interrupt controller 80130 levels IR7 through IR0, respectively.
E85	Routes 8274 RxDRQA to the interrupt jumper matrix.
E86	Routes PFIN* signal from the external source via connector J4.
E87	Routes SBX1 INTR0 to the interrupt jumper matrix.
E88	Routes SBX1 INTR1 to the interrupt jumper matrix.
E89	Routes SBX2 INTR0 to the interrupt jumper matrix.
E90	Routes SBX2 INTR1 to the interrupt jumper matrix.
E91	Routes SBX2 TDMA to the interrupt jumper matrix.
E92	Not Used.
E93	Routes SBX1 TDMA to the interrupt jumper matrix.
E94	Routes TDMA1 to the interrupt jumper matrix for connection to E91 or E93.
E95	Routes 8274 TxDRQB to the interrupt jumper matrix.
E96	Input to spare inverter from jumper matrix.
E97 thru E104	Inputs to the 8259 slave interrupt controller levels IR7 through IR0, respectively.
E99-E109 §	Routes the MULTIBUS Interrupt INT5* to the 8259A level IR5.
E105	Routes 8274 RxDRQB/IPI to the interrupt jumper matrix.
E106	Input to spare buffer inverter from jumper matrix for driving MULTIBUS System bus interrupt lines.
E107 thru E114	Routes interrupt levels from the MULTIBUS System Bus connector P1 levels IR7 thru IR0, respectively.
E115	Routes NVEN* to the interrupt jumper matrix.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E118 to E119 §	Routes the local bus interrupt request from the 80130 to the ASYNC READY gate. This provides a Ready signal to the 80186 for interrupt acknowledge cycles.
E118 to E121	Not Used.
E120 to E122	Enables the output from the 80130 timer to continually retrigger the fail-safe timer when in a HALT state. This prevents a timeout from occuring when the CPU is in a HALT state.
E121	Not Used.
E123 to E124	Inserts 2 wait states for Bank A (RAM) accesses.
E124 to E125	Inserts 1 wait state for Bank A (RAM) accesses.
E126 to E127	Inserts 2 wait states for Bank B (EPROM) accesses.
E127 to E128	Inserts 1 wait state for Bank B (EPROM) accesses.
E129	Provides input or output for SBX 1 Option 0.
E130	Provides input or output for SBX 1 Option 1.
E131 thru E145	Configurator block for PROM (Bank B).
E146 thru E160	Configurator block for RAM (Bank A).
E161 to E162	Routes AUXILIARY RESET signal to MULTIBUS connector P2.
E163 to E164 §	Enables the iLBX bus interface when installed.
E165 to E166 §	Connects the iLBX interface Acknowledge signal to the 80186 ready logic for normal iLBX operation.
E166 to E167	Forces 0 wait state iLBX bus interface operation for use with some iLBX bus interface memory boards.
E168	Provides input or output for SBX2 Option 0.
E169	Provides a third SBX MDACK* signal for connection to one of the SBX Option lines.
E170	Provides input or output for SBX2 Option 1.
E171 to E172 §	Routes TMR OUT 1 from the 80186 as the transmitter clock for Channel B of the 8274 and out as the receive clock for the external terminal via J2 pin 19.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E171 to E174	Allows a clock signal to be routed to the 8274 MPSC via the DTR line.
E173	Provides connection from the 80130 timer to Channel B of the 8274.
E175 to E177 §	Connects the 8274 Channel B Transmit clock to the Channel B receive clock.
E176 to E177	Provides connection from the external transmit clock as the receive clock for the 8274.
E178	Provides connection to a user-installed 10 MHz oscillator for subsequent routing to the MULTIBUS System Bus as BCLK* and CCLK* or to the iSBX bus interface for MCLK.
E179 to E180	Routes the 6 MHz clock to CCLK*, BCLK* and MCLK for the MULTIBUS System Bus and the iSBX expansion bus.
E181 to E183	Routes the output from the 12 MHz oscillator to the clock input of the 80186 processor. The clock output from the 80186 to the rest of the circuitry is 6 MHz.
E182 to E183	Not Used.
E184 to E185	Routes 3 MHz to the clock input of the 8274.
E186	Provides connection to the 80186 timer 0 input.
E187	Provides connection to the 80186 timer 1 input.
E188 to E189	Routes ASYNC READY signal to ARDY input of the 80186 processor. This jumper must not be removed.
E188 to E190	Not used.
E191 to E193	Routes SYNC READY signal to SRDY input of the 80186. Do not remove this jumper.
E192 to E193	Not Used.
E194 thru E197	Selects the PROM address range. Refer Chapter 2 for additional information.
E198 thru E203	Selects the RAM address range. Refer to Chapter 2 for additional information.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E204 to E205	Enables selection of 16 K bytes of the internal OSP of the 80130 Operating System Firmware (OSF) device.
E206 to E207	Enables the iSBC 341 Module RAM address range. Install when using the iSBC 341 board only.
E207 to E208 §	Selects four byte-wide memory sites for RAM memory. If installed, the iSBC 341 Module cannot be used.
E209 to E213	Provides ground to ARQT input of the 8289 Bus Arbiter. Refer to Chapter 2 for operation.
E210 to E214 §	Routes BPRO* to the MULTIBUS System Bus Interface connector P1 pin 16 for serial priority resolution.
E211 to E212	Provides ground to CBRQ* signal input to the Bus Arbiter (8289). Refer to Chapter 2 for 8289 operation.
E212 to E215 §	Routes CBRQ* to the MULTIBUS System Bus interface connector P1 pin 29. Refer to Chapter 2 for 8289 operation.
E216 to E217	Routes the MULTIBUS lock signal MB LOCK* to the MULTIBUS System Bus interface connector P1 pin 25. Remove in AACK* system.
E218 to E219	Routes BCLK* output signal to MULTIBUS connector P1 pin 12.
E220 to E221	Routes CCLK* output signal to MULTIBUS connector P1 pin 31.
E221 to E222	Receives CCLK* from MULTIBUS connector P1 pin 31 to generate on-board BCLK* and MCLK* signals.
E223 to E224 §	Routes SMRQ from the iLBX interface slave to the HOLD input of the 80186 processor. Do not remove.
E225 to E226 §	Routes HLDA from the 80186 as hold acknowledge. Do not remove.
E227	Provides an active low, open-collector driver for driving interrupt signals onto the MULTIBUS System bus interface.
E228 thru E235	Provides connection to MULTIBUS System Bus interrupt lines INT 0 through INT7, respectively, to allow the iSBC 186/03 board to drive these signals.

Table A-2. Numerical List of Jumpers and their Functions (continued)

Jumper Number	Functions
E236 to E237 §	Connects +5 volts bus from +5 volt battery bus. Remove for battery backup applications.
E238 to E239	Install when iRAM devices are installed onto the iSBC 186/03 board.
E240 to E241 §	Enables the iLBX bus address drivers when the iLBX bus is accessed by the iSBC 186/03 board.
Note: The § identifies the as-shipped configuration.	

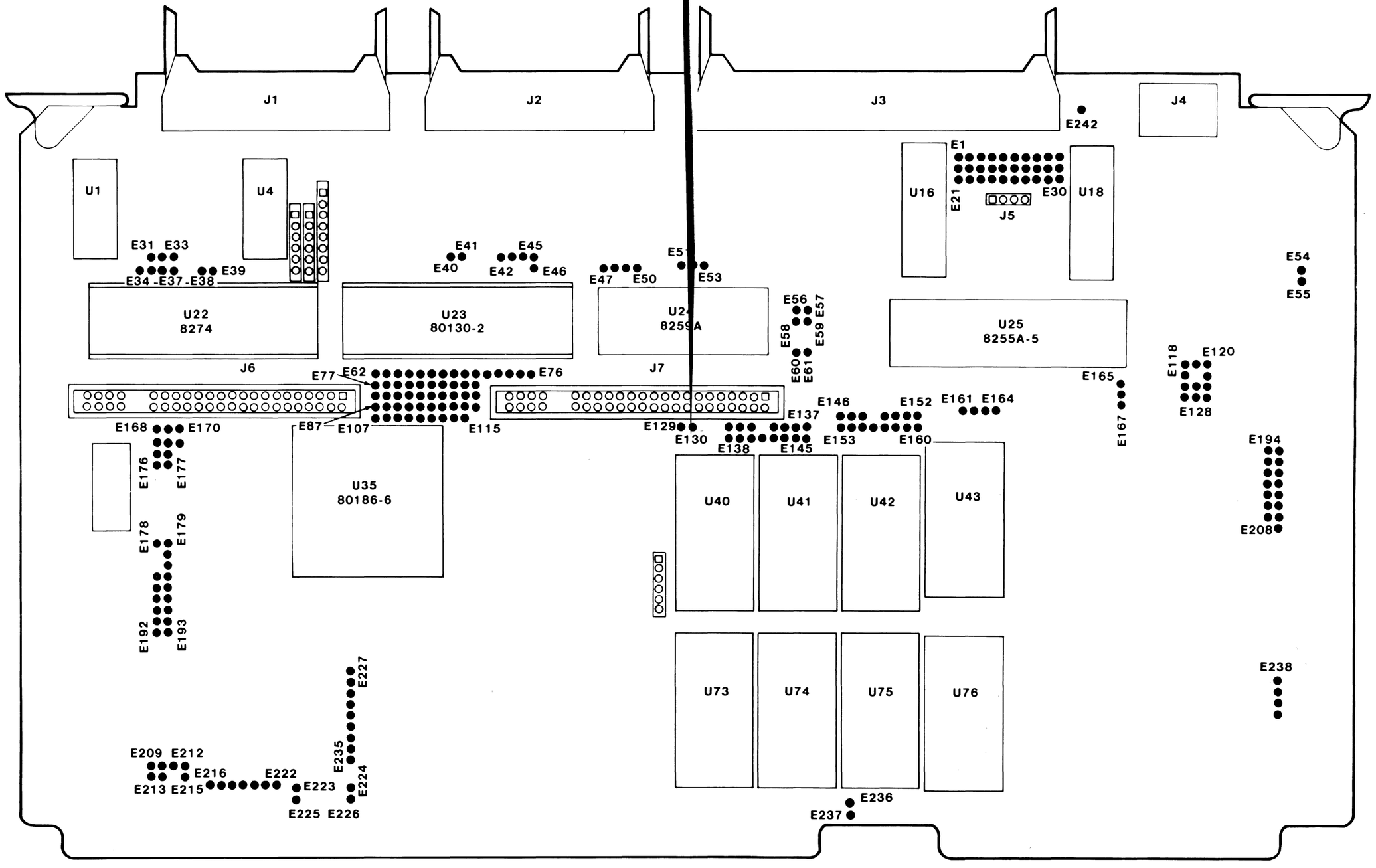


Figure A-1. Jumper Post Location Diagram

***

APPENDIX B. MULTIDROP CONSIDERATIONS

B.1 INTRODUCTION

This appendix describes how to calculate the bias resistance necessary for multidrop installations. An example of a typical system is given.

B.2 MULTIDROP DESCRIPTION

The iSBC 186/03 board allows the driver for the RS422A serial interface on J1 to be enabled or disabled by the 8274 DTR signal. This allows multiple boards to be connected on the same serial cable in a multidrop implementation with only one driver driving the lines at anytime.

When all drivers on the line are disabled, the serial lines are floating. The bias resistors RP1 and RP2 must be installed so that the serial lines are in a 'known state' during this time. The value of these resistors depends on the number of drivers and receivers connected to the lines and the existance of any termination resistors. Three socket SIP resistor locations are provided for bias resistors (RP1 and RP2) and termination resistors (RP3).

For applications with long transmission lines, a termination resistor should be added at the receiver farthest from the driver. This termination resistance value should be as close as possible to the characteristic impedance of the serial cable, approximately 100 ohms, but not less than 90 ohms.

In some of implementations with long cable stubs or drivers driving in multiple directions on the cable, the termination may need to be placed at several end-point locations on the cable. Because the RS422 drivers do not have enough output current to drive multiple 100 ohm terminators, the resistor value for each terminator must be increased such that the total load resistance is no less than 90 ohms between the differential lines (refer to the EIA RS422 specifications).

Because there are many possible driver/receiver configurations, we cannot address all of these resistor calculations in this manual. A single example will be given with multiple drivers, a single receiver, and a single termination resistor on the line. This can easily be expanded to additional receivers or drivers. Note that regardless of the configuration of the termination resistors (single or multiple resistors) the total termination resistance must still be greater than 90 ohms).

Using the configuration shown in Figure B-1, the value of the bias resistors must be calculated for two conditions:

1. When the lines are tri-stated, and

2. When the lines are driven to the marking (off) state.

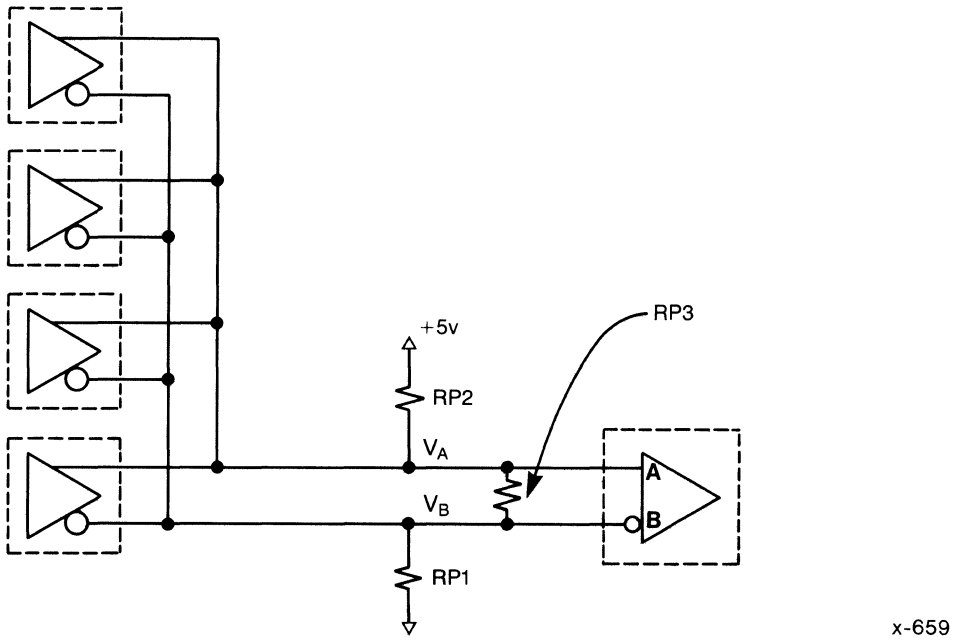


Figure B-1. Example Configuration

When the lines are tri-stated, assume the following conditions:

1. $V_A - V_B \geq 0.3V$ to guarantee differential voltage for a 'spacing' (on) condition.
2. All drivers are tri-stated.
3. Driver leakage current is $\pm 100\mu A$ for each driver (reference RS422 Specification.)

4. Receiver input current is:

$$\text{for Iin=positive (into device)} \qquad I = \frac{V+3\text{volts}}{4K\ \text{ohms}}$$

$$\text{for Iin=negative} \qquad I = \frac{V-3\text{volts}}{4K\ \text{ohms}}$$

(reference RS422 Specification)

5. No Common mode voltage between drivers and receivers

6. Assume $V_A \geq 2.6V$ $\qquad V_B \leq 2.3V$

B.3 CASE 1: LINES FLOATING

To calculate the maximum value of RP1 and RP2, we must calculate the resistance required to guarantee at least a 0.3 volt differential between the lines when they are tri-stated.

Referring to Figure B-2,

$$I_{in(A)} = \frac{V_A+3V}{4K\Omega} = \frac{2.6+3V}{4K\Omega} = +1.4mA$$

$$I_{in(B)} = \frac{V_B-3V}{4K\Omega} = \frac{2.2-3V}{4K\Omega} = -0.20mA$$

$$I_T = \frac{V_A-V_B}{R_{RP3}} = \frac{0.3V}{R_{RP3}}$$

with the termination resistance RP3 = 100 ohms, I_T = 3mA

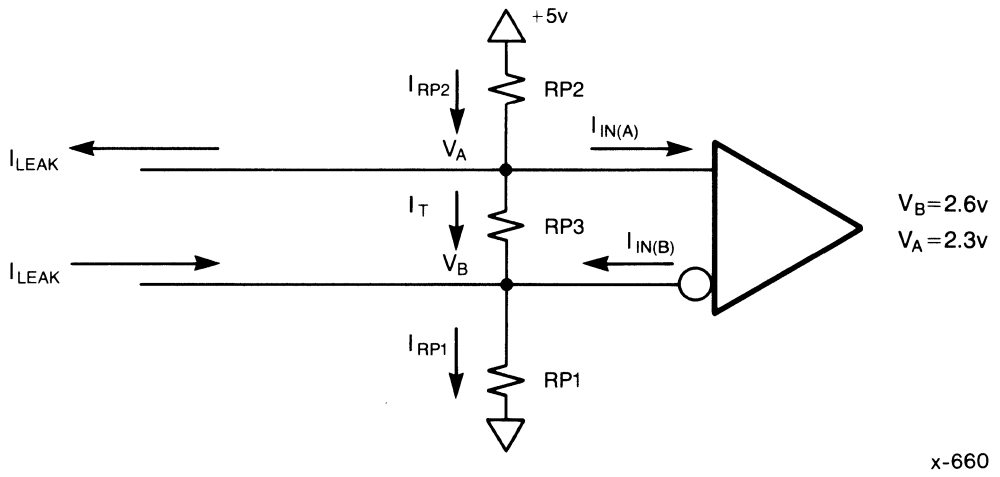


Figure B-2. Case 1 Example Configuration

Writing the node equation for node A:

$$I_{RP2} = I_{in}A + I_{LEAK} + I_T$$

$$= 1.4mA + (100\mu A \times 4 \text{ devices}) + 3mA$$

$$I_{RP2} = 4.8mA$$

To keep $V_A \geq 2.6V$, $R_{RP2} \leq \frac{5V - 2.6V}{4.8mA}$, $R_{RP2} \leq \underline{500\Omega}$

Writing the node equation for Node B:

$$I_{RP1} = I_{in}(B) + I_{LEAK} + I_T$$

$$= 0.2mA + 3mA + 400\mu A = 3.6mA$$

$$\text{To keep } V_A \geq 2.3V,\ R_{RP1} \leq \frac{2.3V}{3.6mA},\quad R_{RP1} \leq \underline{638\Omega}$$

B.4 CASE 2: LINES DRIVEN

When a single driver is driving the lines to a marking (off) state, we must guarantee that $V_B - V_A \geq 0.3$ volts with the following assumptions:

1. Receiver input current equations are same as Case 1.
2. Driver leakage currents are negligible in relation to the drive current of the enabled driver.
3. No common mode voltage between the driver and receiver.
4. Assume $V_A \leq 2.3$ volts, $V_B \geq 2.6$ volts.
5. Driver output current is +20mA (I_{OL}) and -20mA (I_{OH}).

Writing the node equation for Node A and referring to Figure B-3 we have:

$$I_T = \frac{0.3V}{100\Omega} = 3mA$$

$$I_{RP2} + I_{IN}(A) + I_T \leq I_{OL}$$

$$I_{RP2} \leq 20mA - I_T - I_{IN}(A)$$

$$I_{RP2} \leq 16.8mA$$

and

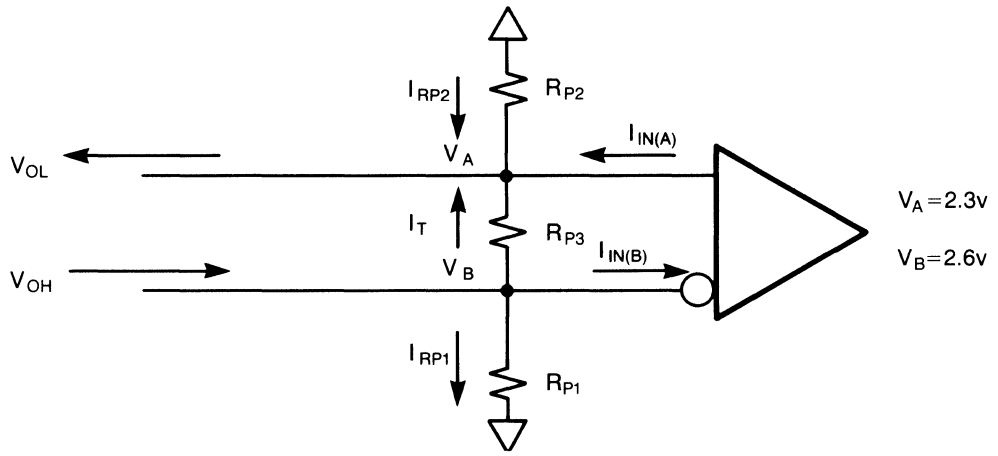


Figure B-3. Case 2 Configuration Example

$$\frac{+5V - V_A}{R_{RP2}} \leq 16.8mA \quad \text{so that } R_{RP2} \geq \frac{+5V - V_A}{16.8mA} \geq \underline{166\Omega}$$

Writing the node equation for Node B:

$$I_{RP1} + I_T + I_{IN}(B) \leq 20mA$$

$$I_{RP1} = \frac{V_B}{R_{RP1}} \leq 20mA - 3mA - 1.32mA$$

$$R_{RP1} \geq \frac{2.3V}{15.7mA} \geq \underline{146\Omega}$$

Combining these results we find:

$$146\Omega \leq R_{RP1} \leq 638\Omega$$

$$166\Omega \leq R_{RP2} \leq 500\Omega$$

The values for RP1 and RP2 should be equal and near the top of the resistor range to reduce the current through the driver. Choosing a resistor value of $450\Omega \pm 10\%$ will satisfy this requirement.

***

APPENDIX C. PARALLEL PORT CONFIGURATIONS

C.1 INTRODUCTION

The iSBC 186/03 board is shipped with the parallel port configured for general purpose I/O operations. Specifically, port A is an 8-bit output port, port B is an 8-bit input port, and port C is configured for various bit level functions. Table C-1 summarizes the default parallel port bit assignments. Figure C-1 shows the physical layout of the parallel port jumper matrix in the default configuration. Refer to Chapter 2 for pin locations of connector J3.

This appendix contains the information needed to convert the parallel interface from the default general purpose parallel port configuration to either the Small Computer Systems Interface (SCSI) configuration, or the Centronics printer interface configuration. In general, conversion consists of programming PAL devices and installing them, and re-configuring the jumpers.

The parallel port can also be converted to any custom interface which your application requires. If you are planning to use a customized interface, you must program your own PALs to provide your own connector interface. Refer to the 8255A Programmable Peripheral Interface data sheet for related programming information.

The following sections cover the two supported configurations:

Section C.2 - SCSI Configuration

Section C.4 - Centronics Printer Configuration

Table C-1. Parallel Port Bit Assignments (Default Configuration)

Port	Direction	Bit	Assigned Function	J3 Pin
Port A	Output	0	Data Bit 0	J3-2
		1	Data Bit 1	J3-4
		2	Data Bit 2	J3-6
		3	Data Bit 3	J3-8
		4	Data Bit 4	J3-10
		5	Data Bit 5	J3-12
		6	Data Bit 6	J3-14
		7	Data Bit 7	J3-16
Note: Port A can be configured as input or output by installing jumper E4-14 and programming Port C, Bit 7.				
Port B	Input	0	Data Bit 0	J3-36
		1	Data Bit 1	J3-38
		2	Data Bit 2	J3-40
		3	Data Bit 3	J3-42
		4	Data Bit 4	J3-44
		5	Data Bit 5	J3-46
		6	Data Bit 6	J3-48
		7	Data Bit 7	J3-50
Port C	Input/ Output	0	PC0; To E20	J3-28
		1	PC1; To E29	J3-30
		2	PC2	J3-32
		3	PC3; To E5 and U17	J3-34
		4	PC4; To E2 and DS1	N/C
		5	PC5; to E22 and J5-3	N/C
		6	PC6; to J5-1	N/C
		7	PC7; to E4 and J5-2	N/C

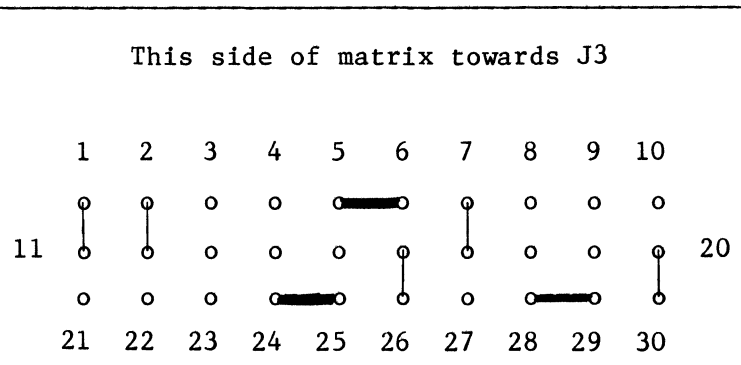


Figure C-1. Parallel Port Matrix Configuration For Default Jumpers

C.2 SCSI INTERFACE CONFIGURATION

The parallel port can be reconfigured for SCSI-compatible mass storage devices via the SCSI parallel port interface. This section provides the following information:

- iSBC 186/03 board conversion information.
- SCSI cabling information.
- SCSI operation overview.
- SCSI software requirements.

Currently, the iSBC 186/03 board has been successfully interfaced to the following SCSI controllers:

- Data Technology Corporation DTC-510A
- Shugart 1601-2
- Shugart 1610
- IOMEGA Alpha 10
- Adaptec ABC 4000
- Xebec S1410
- Vermont Research Corporation 8010/8520
- Fujitsu M2312 with SCSI option.

The iSBC 186/03 board can be interfaced to many other SCSI devices. Refer to the vendor hardware reference manual for interfacing details on a particular product.

The following procedure outlines the required steps for converting the parallel port on the iSBC 186/03 board to the SCSI configuration:

1. Program a 20R4 PAL (U18) and a 20L8A PAL (U16) with the equations given in Tables C-2 and C-3. These devices will assemble using a Data I/O Palasm design adapter and PAL programmer. Use a Monolithic Memories Inc. PAL or equivalent. After programming the PALs, install them in the appropriate sockets on the iSBC 186/03 board.

2. Remove all jumpers from E1 through E29 (these jumpers were installed at the factory).

3. Refer to Figure C-2 and install the following jumpers:

E8 to E18
E10 to E20
E14 to E24
E16 to E17
E19 to E29

E68 to E83 enables the SCSI interrupt to level IR1 of the 80130, if required. This is discussed later in this section.

4. Fabricate the cable assembly in accordance with the data given in Figure C-3.

5. Remove the 74LS245 device from location U15 and install a 74LS640-1 device in its place. Center the 20-pin device in the 24-pin socket.

6. Insert the iSBC 186/03 board back into your system chassis.

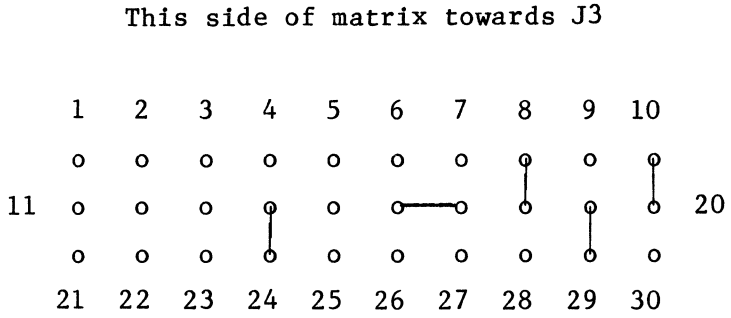


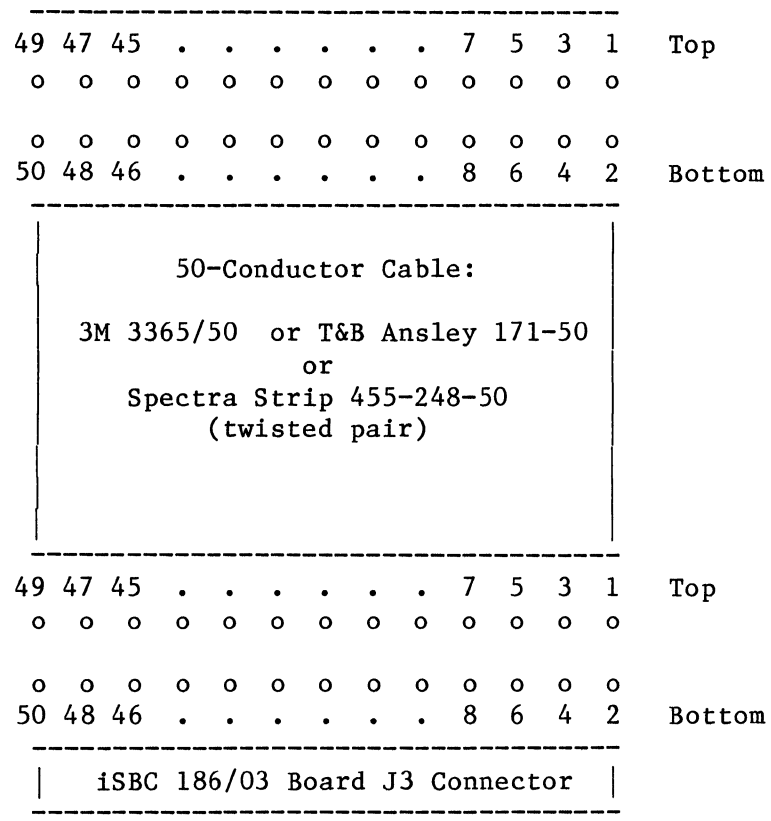
Figure C-2. Parallel Port Matrix Configuration for SCSI Jumpers

Table C-2. XU18 (Handshake Circuitry) PAL Equations

```
PAL20R4
PAT0001
SASI HANDSHAKE PAL FOR PP2    LAST UPDATE 08/25/83

BCLK N55CS NIOWT NIORD A1 A2 NREQ OI DC IOD4 NBPL GND
GNDB T302 NDMAREQ NSCSIINT NST3 NST2 NST1 NST0 NPTAIN ACKEN EN VCC

/NST0 := NST3 * NST2 *  NST1 *  NST0 * EN * T302 * /N55CS * /A1 * /A2 * /NIORD * /OI          ;  I2 Port A Read
       +  NST3 * NST2 *  NST1 *  NST0 * EN * T302 * /N55CS * /A1 * /A2 * /NIOWT * /NREQ * OI   ;  I1 Port A Write
       + /NST3 * NST2 *  NST1 * /NST0 * EN                                                    ;  1001 TO 0001
       +  NST3 * NST2 *  NST1 * /NST0 * EN                                                    ;  0001 TO 0011
       +  NST3 * NST2 * /NST1 * /NST0 * EN                                                    ;  0011 TO 0111

/NST1 := NST3 *  NST2 *  NST1 *  NST0 * EN * T302 * /N55CS * /A1 * /A2 * /NIORD * /OI          ;  I2 Port A Read
       +  NST3 *  NST2 *  NST1 * /NST0 * EN                                                   ;  0001 TO 0011
       +  NST3 *  NST2 * /NST1 * /NST0 * EN                                                   ;  0011 TO 0111
       +  NST3 * /NST2 * /NST1 * /NST0 * EN * OI                                              ;  0111 TO 0110 on Write only
       +  NST3 * /NST2 * /NST1 *  NST0 * EN                                                   ;  0110 TO 0010
       + /NST3 *  NST2 * /NST1 *  NST0 * EN                                                   ;  1010 TO 1110
       +  NST3 *  NST2 *  NST1 *  NST0 * EN * T302 * /N55CS * A1 * /A2 * /NIOWT * /IOD4       ;  I3 Port B Write

/NST2 := NST3 * NST2 *  NST1 *  NST0 * EN * T302 * /N55CS * /A1 * /A2 * /NIORD * /OI          ;  I2 Port A Read
       +  NST3 *  NST2 * /NST1 * /NST0 * EN                                                   ;  0011 TO 0111
       +  NST3 * /NST2 * /NST1 * /NST0 * EN * OI                                              ;  0111 TO 0110 on Write only
       + /NST3 * /NST2 * /NST1 *  NST0 * EN                                                   ;  1110 TO 1100
       +  NST3 * /NST2 *  NST1 *  NST0 * EN                                                   ;  0100 TO 1100
       + /NST3 *  NST2 * /NST1 *  NST0 * EN * T302 * /N55CS * A1 * /A2 * /NIORD               ;  I4 Port B Read
       +  NST3 *  NST2 *  NST1 *  NST0 * EN * T302 * /N55CS * A1 * /A2 * /NIOWT * IOD4       ;  I5 Port B Write

/NST3 := NST3 *  NST2 *  NST1 *  NST0 * EN * T302 * /N55CS * /A1 * /A2 * /NIOWT * /NREQ * OI ;  I1 Port A Write
       +  NST3 *  NST2 *  NST1 *  NST0 * EN * T302 * /N55CS *  A1 * /A2 * /NIOWT * /IOD4      ;  I3 Port B Write
       + /NST3 *  NST2 * /NST1 *  NST0 * EN                                                   ;  1010 TO 1110
       + /NST3 * /NST2 * /NST1 *  NST0 * EN                                                   ;  1110 TO 1100
       + /NST3 * /NST2 *  NST1 *  NST0 * EN                                                   ;  1100 TO 1000
       +  NST3 * /NST2 *  NST1 *  NST0 * EN                                                   ;  0100 TO 1100

IF (VCC) /NDMAREQ = /ACKEN * NST0 * NST1 * NST2 * NST3 * DC * /NREQ * EN * NIOWT
                  + /ACKEN * NST0 * NST1 * NST2 * NST3 * DC * /NREQ * EN * N55CS
                  + /ACKEN * NST0 * NST1 * NST2 * NST3 * DC * /NREQ * EN * A1
                  + /ACKEN * NST0 * NST1 * NST2 * NST3 * DC * /NREQ * EN * A2

IF (EN) /NSCSIINT = /NREQ * /DC                                                  ; Int on any cmd any time
                  + /NST3 * NST2 * /NST1 * NST0 * /NREQ                          ; Int on data @1010

IF (VCC) /NPTAIN = /NIOWT * /N55CS * A1 * A2 * T302 *IOD4
                   + N55CS * /NPTAIN
                   + NIOWT * /NPTAIN
                   + /A1 * /NPTAIN
                   + /A2 * /NPTAIN
                   + IOD4 * /NPTAIN
                   + T302 * /NPTAIN

IF (VCC) /ACKEN = NREQ
                + /EN
                + NST0 * /ACKEN
                + NST1 * /ACKEN
                + NST2 * /ACKEN
                + /NST3 * /ACKEN
```

Table C-3. XU16 (Parity circuitry) PAL Equations

```
PAL20L8
PAT0002
PARITY GENERATION AND CHECKING PAL   LAST UPDATE - 8/8/83

BCLK D6 D5 D4 D3 D2 D1 D0 NPTAIN EN NST2 GND
NST3 OI BUFOI NBPL NPARC NPARB NPARA NBUFEN NDP1 NDP2 D7 VCC


IF (VCC) /NPARA = /D0 * /D1 *  D2
                +  D0 * /D1 * /D2
                + /D0 *  D1 * /D2
                +  D0 *  D1 *  D2

IF (VCC) /NPARB = /D3 * /D4 *  D5
                +  D3 * /D4 * /D5
                + /D3 *  D4 * /D5
                +  D3 *  D4 *  D5

IF (VCC) /NPARC = /D6 *  D7 *  NDP1 * /OI
                +  D6 * /D7 *  NDP1 * /OI
                +  D6 *  D7 * /NDP1 * /OI
                + /D6 * /D7 * /NDP1 * /OI
                +  D6 * /D7 * OI
                + /D6 *  D7 * OI

IF (VCC) /NBPL =  NST3 * /NST2 * /OI * EN * /NPARA * /NPARB *  NPARC
               +  NST3 * /NST2 * /OI * EN *  NPARA * /NPARB * /NPARC
               +  NST3 * /NST2 * /OI * EN * /NPARA *  NPARB * /NPARC
               +  NST3 * /NST2 * /OI * EN *  NPARA *  NPARB *  NPARC
               +  NST2 * /NBPL * EN
               +  NST3 * /NBPL * EN

IF (VCC) /NBUFEN = /NPTAIN * EN
                 + OI * EN

IF (EN * OI * /NBUFEN * NPTAIN) /NDP1
               = /NPARA * /NPARB *  NPARC
               +  NPARA * /NPARB * /NPARC
               + /NPARA *  NPARB * /NPARC
               +  NPARA *  NPARB *  NPARC

IF (EN * OI * /NBUFEN * NPTAIN) /NDP2
               = /NPARA * /NPARB *  NPARC
               +  NPARA * /NPARB * /NPARC
               + /NPARA *  NPARB * /NPARC
               +  NPARA *  NPARB *  NPARC


IF (VCC) /BUFOI = /OI
```



J3 Mating Connector: 3M 3425-7050 or T&B Ansley 609-5001M

Figure C-3. Cable Assembly Data for SCSI Interface

C.3 SCSI OPERATION WITH THE iSBC® 186/03 BOARD

This section provides a general discussion of how the SCSI interface operates on the iSBC 186/03 board. For a more complete description of all features, phases and definitions of the SCSI bus, refer to the SCSI bus specification ANSI X3T9.2/82-2 Rev.7 (25 April 83).

INITIALIZATION - After Reset, by writing the control word to the 8255 control port (I/O addr 00CEH = 082H), the 8255A will have been initialized to Mode 0 with ports A and C as 8 bit output ports and port B as an 8 bit input port. Since PC6 has a 10K pull-up, this line becomes high at inverter U29P11 causing the output at U29P10 to go low which will reset the state machine of U18 to state 0000 (see state machine diagram Figure C-4).

SELECTION - This operation consists of detecting a BUS FREE condition and then asserting the target's ID on the data bus followed by asserting SEL*. BUS FREE is defined as SEL* and BSY* not asserted while RST* is inactive. This condition can be detected by reading from port B the value XXX1X1X1B or 015H (assuming 0's as X's). SEL* is asserted by writing to port C the value X0XXXX1X or 002H (assuming 0's as X's). PC6 = low enables U17 thru the inverter U29 and buffer U10 so that PC1 = high will cause a low on U17P11 to activate SEL* on the SCSI bus at J3P44. Once the target has recognized its ID, it will respond by asserting BSY*. The CPU should be polling port B for BSY* to be asserted. When it is, SEL* can be released. SEL* is released by writing to port C bit 1 of the 8255A the value 0. This will leave ENABLE active. Selection is accomplished without ever leaving state 0000 (see state machine diagram).

The following phases can be grouped together as the information transfer phases because they are all used to transfer data or control information via the data bus. The D/C*, O/I* and MSG* signals are used to distinguish between the different information transfer phases, as laid out in the following table.

Table C-4. Transfer Phase Signals

Signal				
MSG*	D/C*	O/I*	Phase	Direction
1	1	1	DATA WRITE	INITIATOR to TARGET
1	1	0	DATA READ	INITIATOR from TARGET
1	0	1	COMMAND	INITIATOR to TARGET
1	0	0	STATUS	INITIATOR from TARGET
0	1	1	Not Used	
0	1	0	Not Used	
0	0	1	MESSAGE WRITE	INITIATOR to TARGET
0	0	0	MESSAGE READ	INITIATOR from TARGET

Note: 1 = not asserted or high, 0 = asserted or low

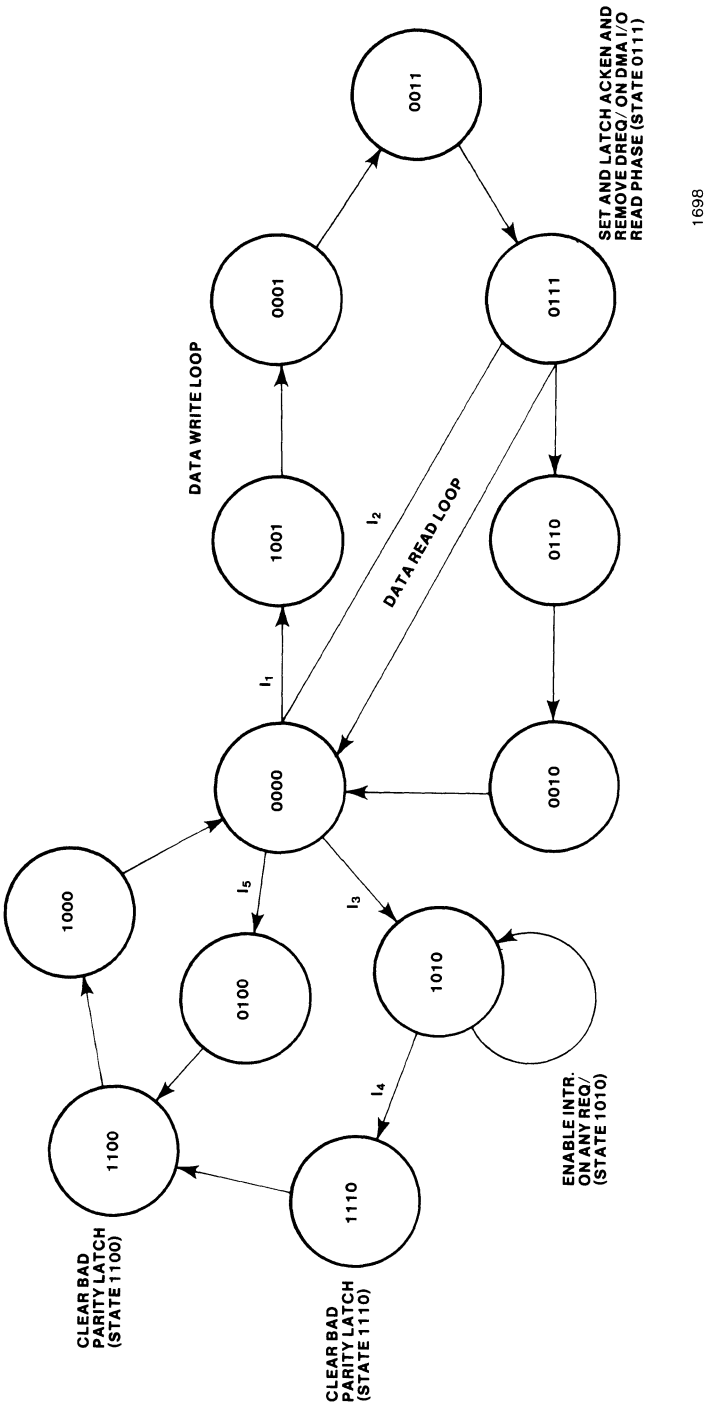


Figure C-4. SCSI State Machine Diagram

The information transfer phases use one or more REQ/ACK handshakes to control the information transfer. Each REQ/ACK allows the transfer of one byte of information. During each information transfer phase the BSY* line shall remain asserted and the SEL* line shall remain released. The target shall also continuously envelope the REQ/ACK handshake(s) with the D/C*, O/I* and MSG* signals in such a manner that these signals are valid before the REQ* of the first handshake and after ACK* of the last handshake.

REQ/ACK HANDSHAKE - This handshake shall start with the target asserting REQ* only when ACK* is not asserted. The initiator responds by asserting ACK*. The target will then respond by removing REQ*. The initiator will then respond by removing ACK*. During the time that REQ* is asserted and while ACK* is asserted, information will be transferred. Therefore data must be valid before ACK* is asserted and held after ACK* is released. ACK* is released when the initiator (iSBC 186/03) releases ACKEN which is released when the target releases REQ*.

CONTROLLING THE BUFFER - The 74LS640-1 bidirectional bus driver/receiver that is on the SCSI data bus is enabled by the signal BUFEN* (U16P20). BUFEN* is active whenever the signal PORTAIN* (U18P21) is active, or the SCSI bus signal O/I*, is in the output state and ENABLE (U10P12) is active. The signal PORTAIN* is generated by writing the mode word to the 8255A. U18 detects an I/O write to the control port of the 8255A (I/O addr 0CEH) with IODB4 = 1. When IODB4 = 1 during a mode word write then port A of the 8255A is being set in the input mode. The direction that the driver is driving is determined by the signal BUFO/I*. This is merely the SCSI bus signal O/I* buffered through U16P15.

COMMAND PHASE - The command phase is where the target requests command information from the initiator. The target shall assert the D/C* signal and deassert the O/I* and MSG* signals during the REQ/ACK handshake(s) of this phase. At the start of this phase the state machine will be in state 0000 and will take the route of a data write (I1). The state machine will go through 3 states before entering into state 0111 where ACKEN will be asserted. The state machine will then go through two more states before returning to state 0000 where ACK* will be asserted as a result of ST1* going inactive. Since the state machine is synchronous to the processor clock there will be a period of 7 clocks from the time the data is written into the 8255A to the time that ACK* is asserted. This allows more than enough time for the 8255A propagation delay and some settling time to ensure valid SCSI bus data at the falling edge of ACK*. (see state machine and timing diagrams)

DATA WRITE - The data write phase allows the target to request data from the initiator. The target shall deassert D/C*, O/I* and MSG* signals during the REQ/ACK handshake(s) of this phase. A DMA request was generated by U18P15 when the target asserted REQ* with D/C* deasserted (data read/write only). Referring to the state machine diagram, this action is indicated by I1. I1 will be initiated when port A of the 8255A (I/O addr 0C8H) is written during phase two of CPU machine cycle T3 (T302). Typically the deposit cycle of a DMA transfer.

On a DMA I/O write the DMA request will be removed in T2 when IOWT* goes low. The state machine will then continue, independent of the CPU, to go through three states into state 0111 where ACKEN will be driven. The state machine will continue through two more states into state 0000 where ACK* will be asserted as a result of ST1* going inactive. Once I1 is started each successive state will occur on the rising edge of the processor clock. The number of states executed allows sufficient propagation time for the 8255A and the 74LS640-1 to ensure that data is valid when ACK* is asserted.

DATA READ - The data read phase allows the target to request that data be output to the initiator. The target shall assert the O/I* signal and deassert the D/C* and MSG* signals during the REQ/ACK handshake(s) of this phase. A DMA request is generated by U18 whenever the target asserts REQ* and D/C* is not asserted (data mode). The state machine functions for the data read phase differ drastically from those of a data write. The target is required to have data set up and stable before it asserts REQ*. Therefore, all that the initiator is required to do is to read the data into the 8255A and assert ACK*. Referring to the state machine diagram, a data read takes the I2 path. This path initiates with a read to port A of the 8255A (I/O addr 0C8H) with O/I* asserted. It consists of jumping into state 0111 and setting ACKEN and then jumping back to state 0000 and driving ACK* when ST1* goes inactive at state 0000. The DMA request for a read is removed when the state machine leaves state 0000. The extra states present in a data write phase have been eliminated in a data read phase because the data is already set up when REQ* is asserted. The data read phase has been optimized in this way since the highest percentage of an operating system's accesses are data reads and this will then increase an operating system's throughput.

STATUS PHASE - The status phase allows the target to request that status be sent to the initiator. The target shall assert the D/C* and O/I* signals and deassert the MSG* signal during the REQ/ACK handshake(s) of this phase. A status phase will follow the same state machine path as that of a data read phase (see above), except that there will be no DMA request because D/C* is asserted.

MESSAGE READ PHASE - The message read phase allows the target to request that a message, or messages, be sent to the initiator. The target shall assert D/C*, O/I* and MSG* during the REQ/ACK handshake(s) of this phase. A message read phase will follow the same state machine path as that of a data read phase (see above), except that there will be no DMA request because D/C* is asserted. Multiple byte messages will be wholly contained within a single message read phase.

MESSAGE WRITE PHASE - The message write phase allows the target to request a message from the initiator. The target may invoke this phase at its convenience only in response to the attention condition created by the initiator (see below). In response to the attention condition the target shall assert the D/C* and MSG* signals and deassert the O/I* signal during the REQ/ACK handshake(s) of this phase. A message write phase will follow the same state machine path as that of a data write. Multiple byte messages will be wholly contained within a single message read phase.

ATTENTION CONDITION - The attention condition allows the initiator to inform the target that it has a message ready. The initiator creates the attention condition by asserting the ATN* signal at any time except during an arbitration or a bus free phase. ATN* is asserted by writing the value X0XXX1XXB or 004H (assuming 0's as X's) to port C of the 8255A (I/O addr 0CCH).

INTERRUPT DRIVEN OPERATION - Through all states, an SCSI interrupt is generated whenever REQ* and D/C* are asserted for a command, status or message phase but normally not for a data phase. Each REQ/ACK handshake during a data phase generates a DMA request instead of an interrupt. However, the DMA request could be routed to an interrupt controller for those designs without DMA.

The path shown in the state machine diagram as I3 is initiated by a dummy write to port B of the 8255A with IODB4 = 0. This action puts the state machine into its only holding condition. The state machine will remain in state 1010 until a read of port B is executed. Upon entering state 1010 a special interrupt capability is armed (U18P16) for any phase including data. In this condition when a REQ* is asserted by the target an interrupt to the CPU will be generated. If the interrupt routine then reads the SCSI status to determine the cause of the interrupt, the state machine will continue around the loop clearing the bad parity latch on its way to state 0000. This function could be utilized as follows : The processor sends a command, such as a data read command, to the SCSI bus.

If the user does not wish to waste CPU time polling the SCSI bus for the first data request, the special interrupt could be armed so that an interrupt would be generated when the first data request occurs or the target responds with some type of status or command rather than the data request. At this time the processor would read the SCSI status (port B of the 8255A - I/O addr 0CAH) to determine the nature of the interrupt. The state machine would then return to state 0000. If the interrupt was for a data request, the user would then enable the DMA controller to handle the required number of data requests. The choice of polling or interrupting on the first data request would be dependent upon the response time of the target. A fast response time from the target would be a reason to choose polling. A slow response time from the target would make interrupting more efficient.

BAD PARITY LATCH - The bad parity latch located in the parity PAL U16, is latched whenever incorrect odd parity is detected by that PAL for any read phase. This signal can be read via port B, bit 7 of the 8255A. Once bad parity has been detected and processed, the latch can be cleared by writing to port B with IODB4 = 1 from state 0000. The bad parity latch should normally be sampled at the end of a read data phase or for status and message reads.

DP1, DP2 - These two signals from U16P21 & P22 are identical and are tied together so as to have sufficient drive capability for the SCSI bus signal DBP (J3P18).

Odd parity is generated by the parity PAL and driven onto the SCSI bus (as DBP) for any write cycle (data, selection, command and message). The generation of parity does not affect the bad parity latch.The SCSI interface implemented on the iSBC 186/03 board does not support multiple host arbitration or re-selection phases.

HARDWARE SECTION GLOSSARY

ASSERT — to drive a SCSI bus signal active (low for signals with * or high for signals without *).

DEASSERT — to drive a SCSI bus signal inactive (high for signals with * or low for signals without *); or allow it to tri-state.

INITIATOR — a bus device which initiates an operation on the SCSI bus. In this case the iSBC 186/03 board.

TARGET — a bus device that services an operation initiated on the SCSI bus (i.e. - Xebec S1410). The target requests commands or data from the initiator once the target has been selected.

Software Requirements

For a more complete description of all features, phases and definitions of the SCSI bus, please refer to the SCSI bus specification ANSI X3T9.2/82-2 Rev.7 (25Apri183). Also refer to the "SCSI Buyers Guide", issue #1, November 1982, by Adaptive Data and Energy Systems, 2627 Pomona Blvd, Pomona, California 91768.

The main purpose of the SCSI standard is to streamline the addition of peripherals to a system. This has been accomplished by standardizing the software interface as well as the hardware interface. In other words, not only is the hardware "plug compatible", but the software is "plug compatible". This compatability supports the use of a generic software driver. There is available from Insite a generic software driver that has been written for use on the iSBC 186/03.

Logically addressed blocks are used when addressing a SCSI device as opposed to the classical cylinder, head, sector addresses. Therefore the number of addressable blocks on a drive is a function of the size of the drive. For example, a 5Mb drive with a sector size of 256 will have logical addresses from 0 to 2,0130. The conversion from logical address to physical address is performed by the SCSI controller.

There are four levels of command compliance to the SCSI standard provided for. The first, and minimum level is called "Standard". For a device to be SCSI compatible it must support all commands of this level. The second level is known as "Extended". These commands are a part of the extended SCSI specification. The third level is known as "Optional". If these commands are implemented by a device then they will follow the definition in the SCSI specification.

The fourth, and last level is the "Vendor Unique" classification. These commands are not defined in the SCSI specification. They have been made available to the vendors to implement commands specific to the vendor's device. There are also command codes that have been reserved for future standardization. (See SCSI Standard). These commands fall the eight categories listed below:

Group 0	Six byte commands including control, data transfer and status
Group 1	Ten byte commands
Groups 2 - 4	Reserved
Group 5	Twelve byte commands
Groups 6 & 7	Vendor-unique commands

Commands are passed to the TARGET in the form of Command Descriptor Blocks (CDB). The CDB includes such information as the operation code, the Logical Unit Number (LUN), block starting address, the number of blocks to transfer and the control byte. Upon completion of the command the TARGET will return status followed by a command completion byte message to the INITIATOR.

The software must be able to build and transmit CDB's as well as handle the return of status. An optional characteristic that would raise the level of operation would be for the software to implement the extended and optional commands while allowing for message passing from the TARGET to the INITIATOR. Since a TARGET with minimal SCSI compliance will be returning status for each command, the software must be able to handle exceptions as well. The SCSI driver that is available from the Insite library is an example of the minimum exception handling acceptable. To implement higher levels of exception handling and message passing, the amount of software necessary would go up as a function of the level of complexity that the error handler implemented. Therefore the level of SCSI compliance will be directly related to the size of the software.

The following flow-charts in Figure C-5 provide an outline of the functions the software is expected to implement.

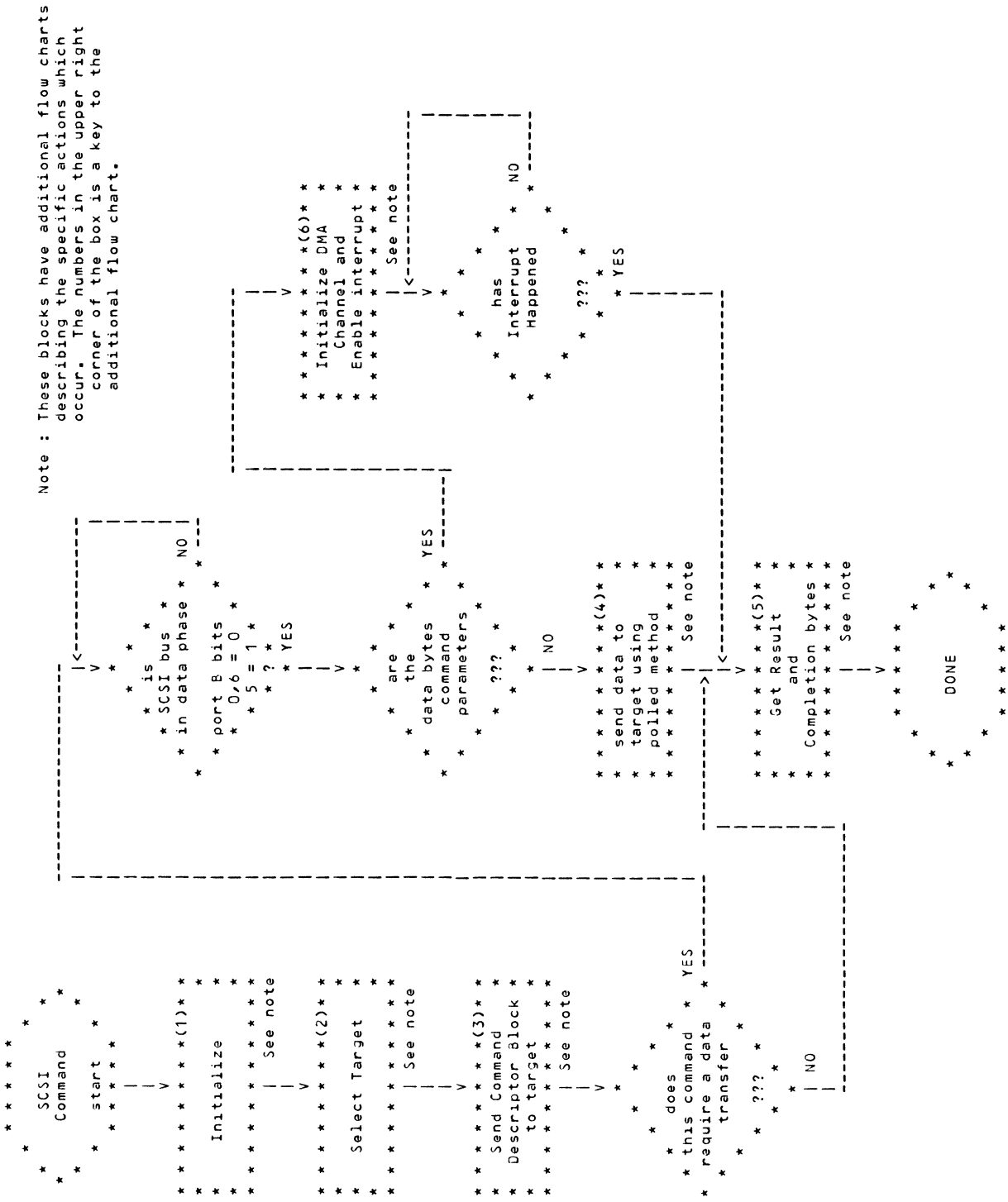


Figure C-5. SCSI Program Requirements

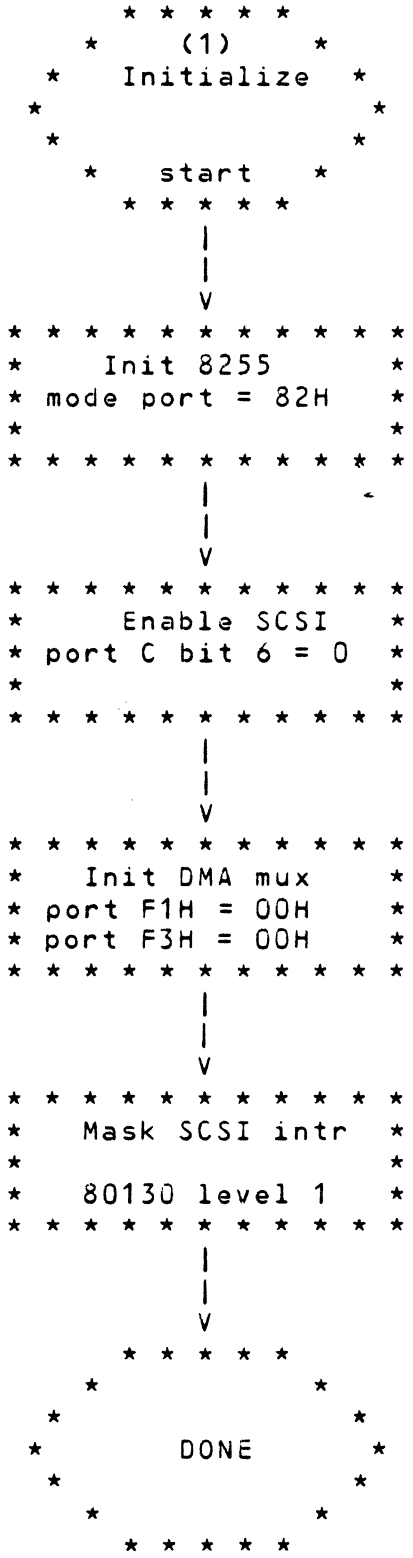


Figure C-5. SCSI Program Requirements (continued)

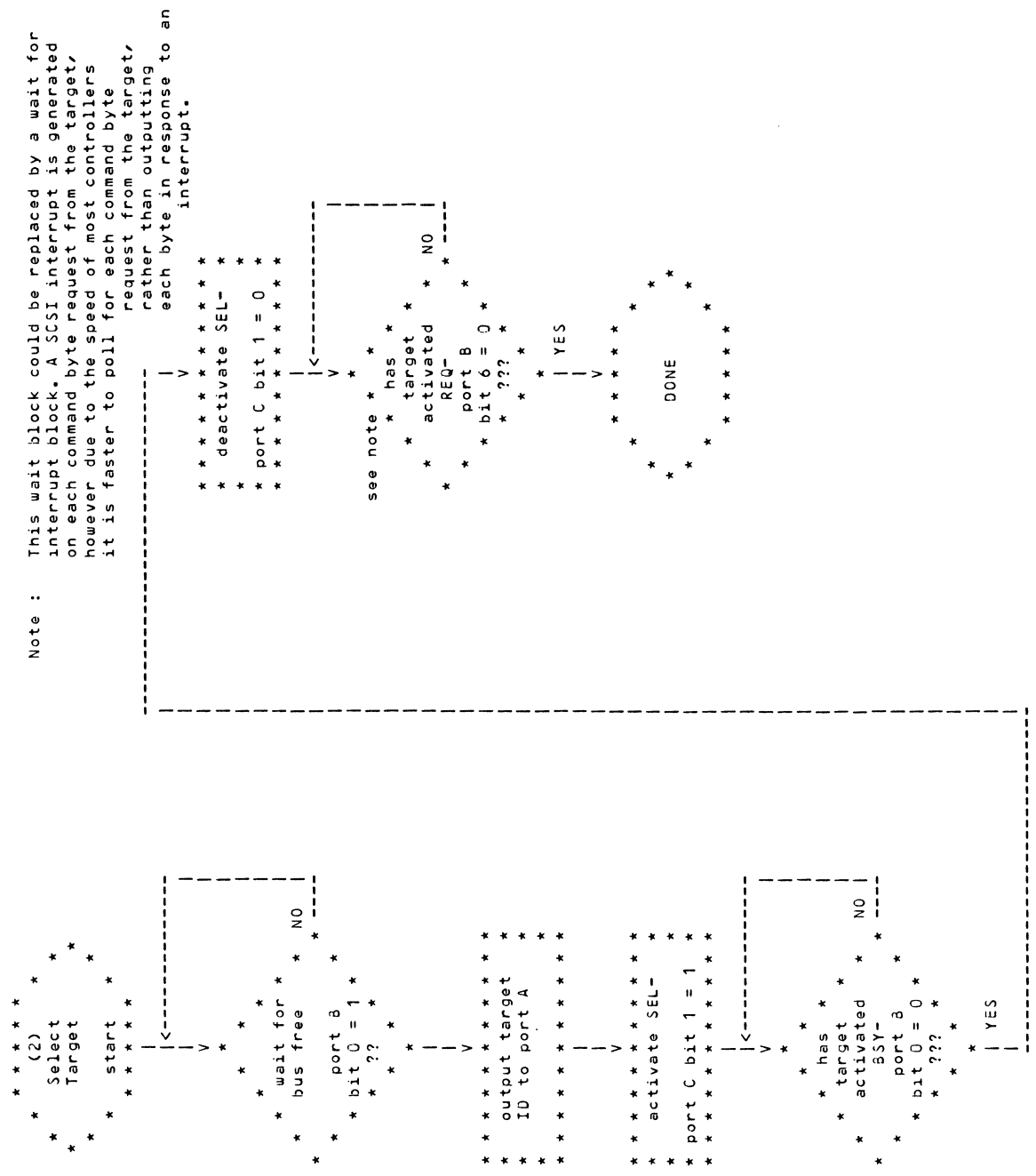


Figure C-5. SCSI Program Requirements (continued)

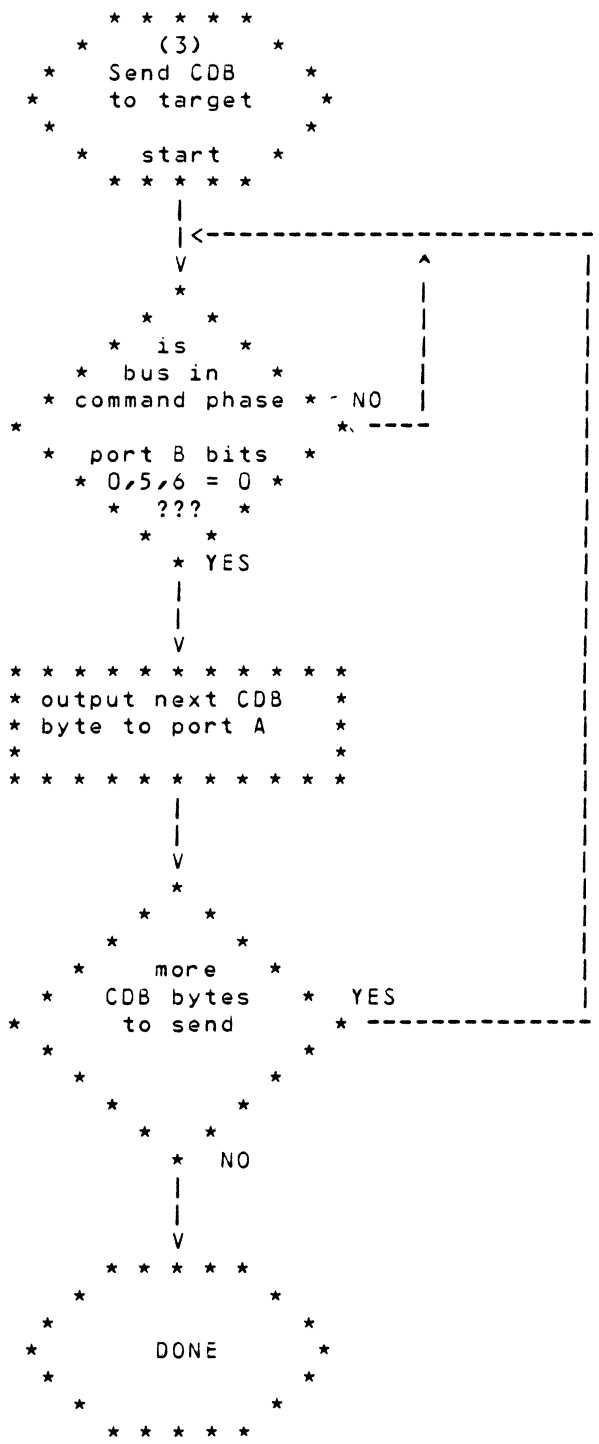


Figure C-5. SCSI Program Requirements (continued)

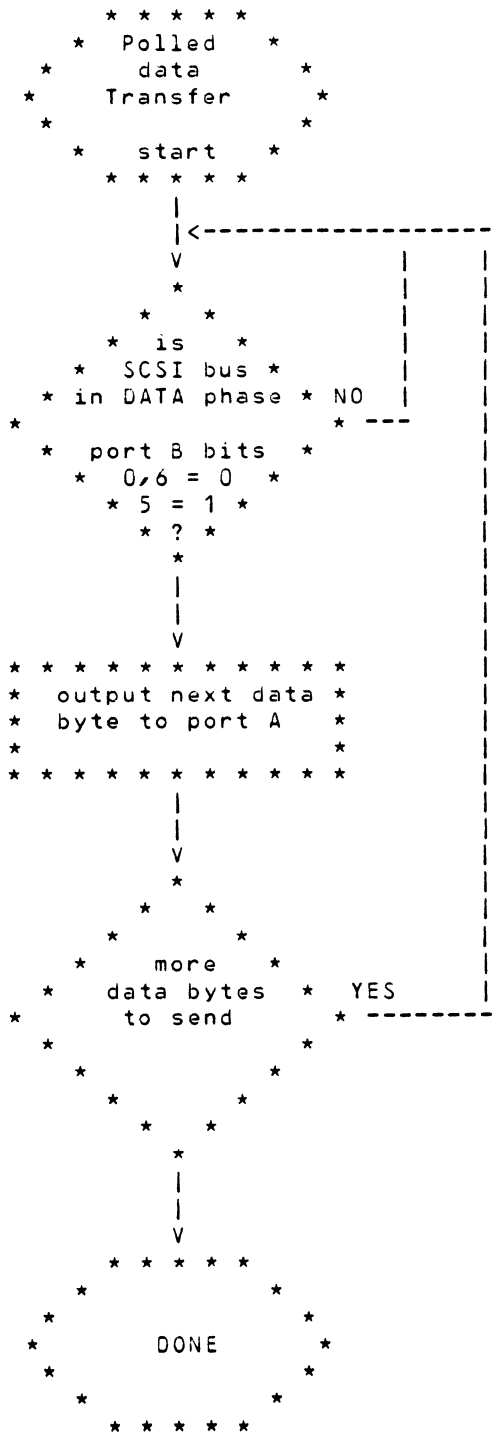


Figure C-5. SCSI Program Requirements (continued)

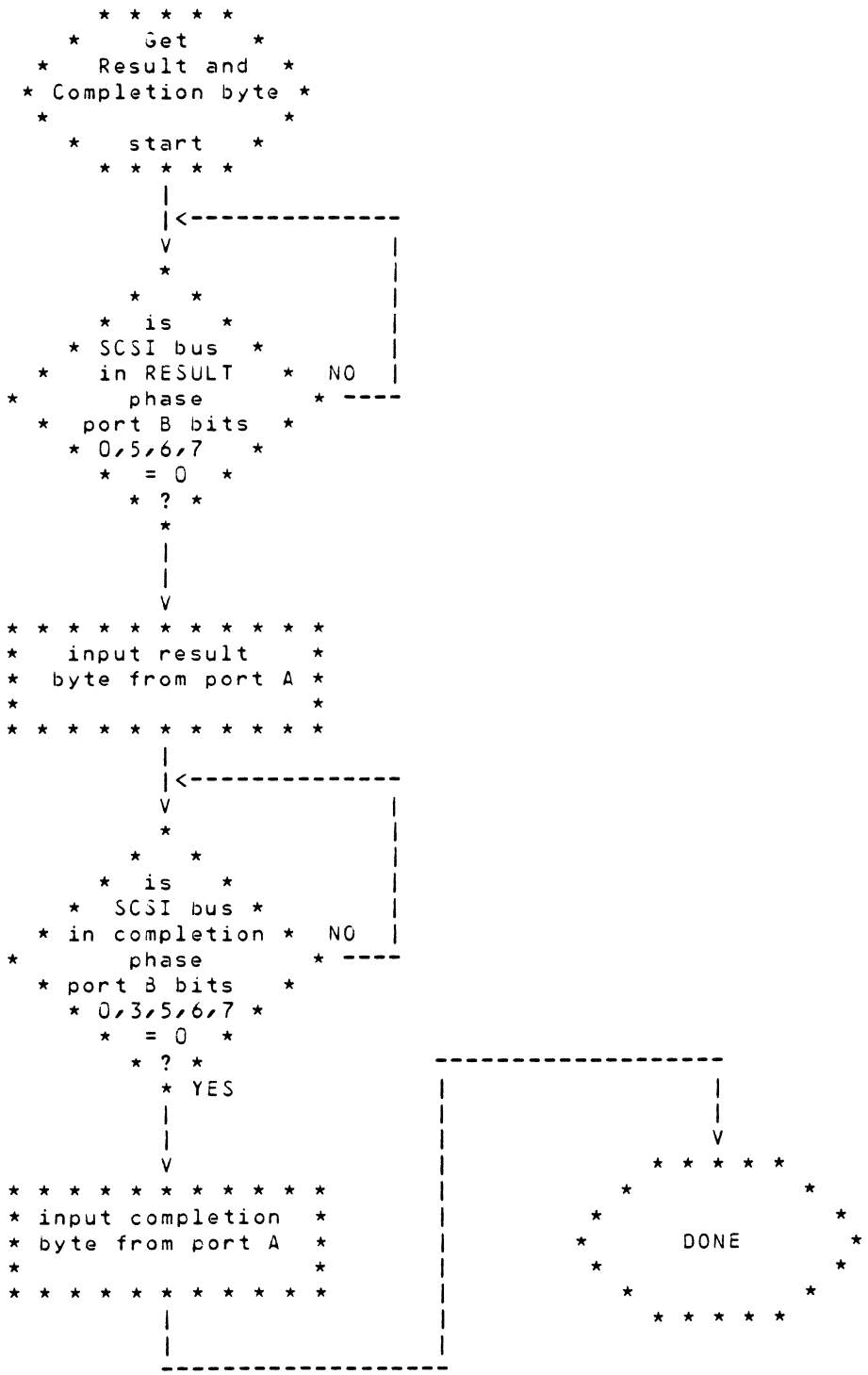


Figure C-5. SCSI Program Requirements (continued)

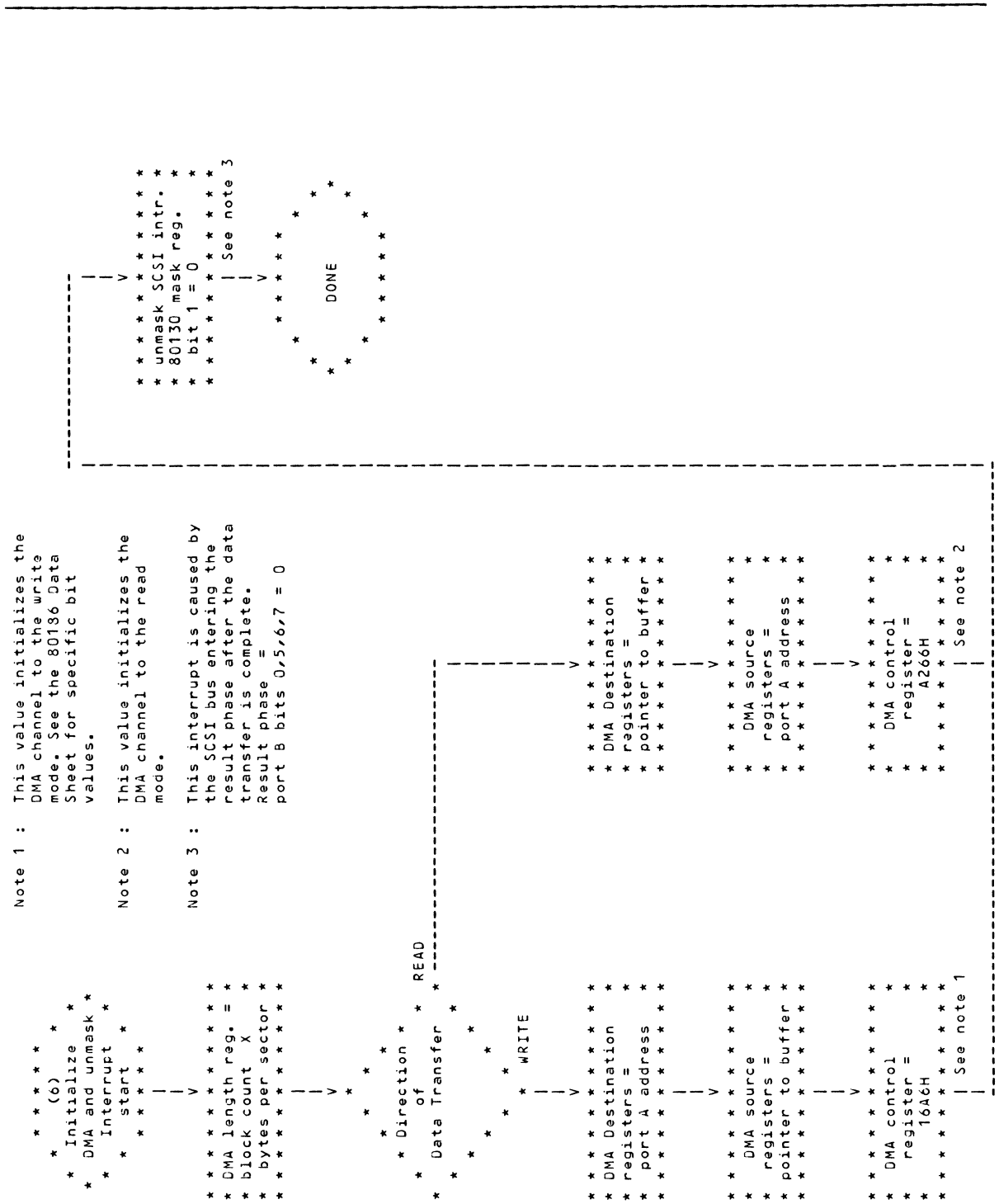


Figure C-5. SCSI Program Requirements (continued)

C.4 CENTRONICS PRINTER INTERFACE

The parallel port interface of the iSBC 186/03 board can be configured to connect to a printer that conforms to the Centronics interface as used on the models 702 or 737 printers. To accomplish this, a 20R4A Programmable Array Logic (PAL) device must be programmed and installed, and the parallel port matrix jumpers must be configured correctly.

The Centronics interface is capable of providing a DMA request signal to the DMA controller when the printer is ready to accept the next character. More than one Centronics controller can be connected to the same printer by controlling PORT C bit 6 of the 8255A Programmable Parallel Interface (PPI) device.

Two interrupts can be provided from the interface via the PPI INT signal. These interrupts may be driven by the FAULT* signal on some printer models or by the SLCT signal on other models.

The first of these interrupt must be inverted at the jumper matrix in order to provide the printer fault interrupt. The select (SLCT) signal and the Paper Empty (PE) signal can both be read via PORT B of the 8255A PPI to determine which fault condition caused the interrupt. The second interrupt must then be wired directly to IR1 of the 80130 in order to provide the printer online interrupt. This second interrupt routine must provide a 1 ms delay for the printer to ready itself to accept a DMA transfer after a fault condition has occurred.

Cables for only two models are given because the cabling is different from printer to printer. Cables for other models need to be modified accordingly.

The PAL code provided in Table C-4 meets the Centronics timing specifications given in Figure C-6. Faster printer interfaces may be implemented by reducing the number of states in the PAL to generate smaller delays between the DATA and the STROBE* signals.

All outputs and inputs to and from the printer are pulled up through 1K ohm SIP resistor packs to +5 volts. All logic levels are TTL compatible (10 MHz maximum). The driver for the data lines is a 74LS245 device.

The following procedure outlines the specific requirements necessary to convert the parallel port on the iSBC 186/03 board to the Centronics configuration.

1. Program the 20R4 PAL with the equations given in Table C-4. After programming the PAL, install the PAL into socket U18 on the iSBC 186/03 board.

2. Refer to Figure C-1 and remove jumpers E5 to E6 and E16 to E25 (these jumpers were installed at the factory).

3. Refer to Figure C-7 and Table C-5 and configure the jumpers as indicated. Figure C-10 shows a schematic diagram of the Centronics interface PAL and jumpers.

4. Verify that a 74LS245 device is installed in socket U15 correctly.

5. Fabricate the printer interface cable assembly in accordance with the data given in Figure C-8. Refer to Table C-6 for pin assignments.

6. Insert the iSBC 186/03 board into your system chassis. Software requirements for the Centronics are discussed in the next section.

Table C-5. Centronics Interface PAL Equations For U18

```
PAL20R4A
PQ006-A
iSBC 186/03 CENTRONICS INTERFACE PAL
INTEL OREGON
CLK NCS NIOWT NC AB1 AB2 PE BUSY SLCT NC NACK GND
NINPUT T302 /DREQ /BLOCK NST0 NST1 NST2 NST3 NST4
/STROBE ENABLE VCC
/NST0 := T302 * /NCS * /AB1 * /AB2 * NST4 * NST3 *
                              NST2 * NST1 * ENABLE +
         NST4 * NST3 * /NST2 * /NST1 * ENABLE +
         NST4 * /NST3 * /NST2 * NST1 * ENABLE +
         NST4 * /NST3 * NST2 * /NST1 * ENABLE +
         /NST4 * /NST3 * NST2 * NST1 * ENABLE +
         /NST4 * /NST3 * /NST2 * /NST1 * ENABLE +
         /NST4 * NST3 * /NST2 * NST1 * ENABLE +
         /NST4 * NST3 * NST2 * /NST1 * ENABLE
/NST1 := NST4 * NST3 * NST2 * /NST0 * ENABLE +
         NST4 * NST3 * /NST1 * NST0 * ENABLE +
         NST4 * /NST3 * /NST2 * /NST0 * ENABLE +
         NST4 * /NST3 * /NST1 * NST0 * ENABLE +
         /NST4 * /NST3 * NST2 * /NST0 * ENABLE +
         /NST4 * /NST3 * /NST1 * NST0 * ENABLE +
         /NST4 * NST3 * /NST2 * /NST0 * ENABLE +
         /NST4 * NST3 * /NST1 * NST0 * ENABLE
/NST2 := NST4 * NST3 * /NST1 * NST0 * ENABLE +
         NST4 * NST3 * /NST2 * /NST0 * ENABLE +
         NST4 * /NST2 * NST1 * NST0 * ENABLE +
         NST4 * /NST3 * /NST2 * /NST0 * ENABLE +
         /NST4 * /NST3 * /NST1 * NST0 * ENABLE +
         /NST4 * /NST3 * /NST2 * /NST0 * ENABLE +
         /NST4 * /NST2 * NST1 * NST0 * ENABLE +
         /NST4 * NST3 * /NST2 * /NST0 * ENABLE
/NST3 := NST4 * NST3 * /NST2 * NST1 * NST0 * ENABLE +
         NST4 * /NST3 * ENABLE +
         /NST4 * /NST3 * NST2 * NST1 * ENABLE +
         /NST4 * /NST3 * NST2 * /NST1 * ENABLE +
         /NST4 * /NST3 * /NST2 * /NST1 * ENABLE +
         /NST4 * /NST3 * /NST2 * NST1 * /NST0 * ENABLE
IF (VCC) /NST4 = /NST3 * NST2 * NST1 * NST0 * ENABLE +
                 /NST4 * /NST3 * ENABLE +
                 /NST4 * NST3 * /NST2 * ENABLE +
                 /NST4 * NST3 * NST2 * /NST1 * ENABLE +
                 /NST4 * NST3 * NST2 * NST1 * /NST0 * ENABLE
IF (ENABLE) STROBE = /NST4 * /NST3 * /NST2 +
                     /NST4 * NST3 * /NST2 +
                     /NST4 * NST3 * NST2
IF (VCC) DREQ = NACK * /BLOCK * ENABLE * /BUSY
IF (VCC) BLOCK = NACK * /NIOWT * /NCS * /AB1 * /AB2 * ENABLE +
                 BLOCK * NACK * ENABLE
```

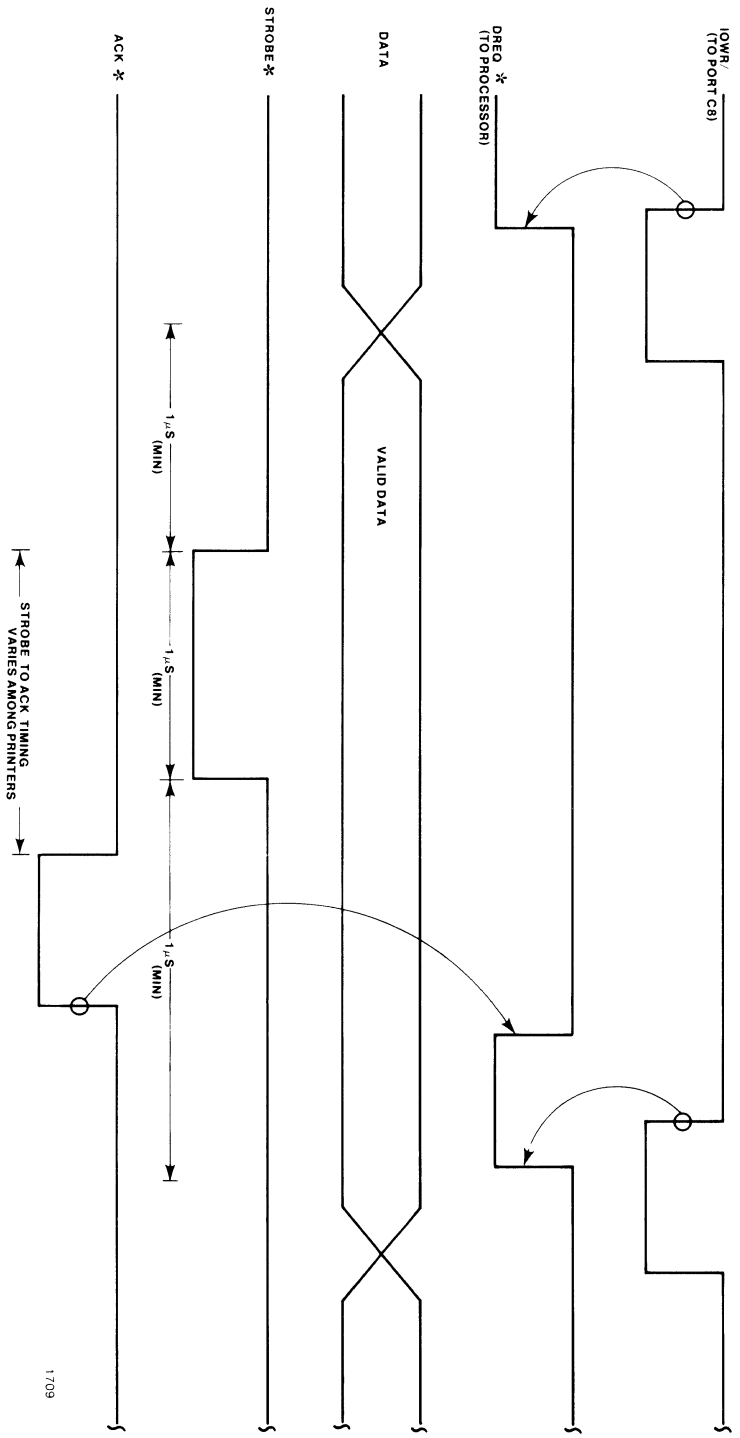
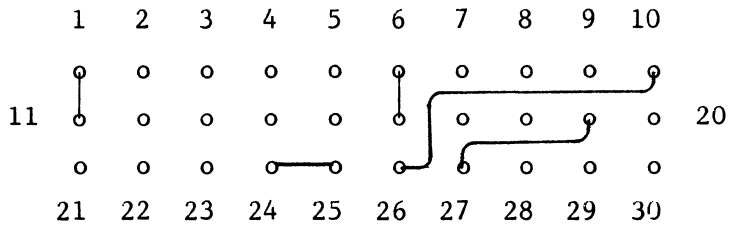


Figure C-6. Centronics PAL Timing Specifications

This side of matrix towards J3



Note: Add jumpers shown in Table C-5.

Figure C-6. Parallel Port Matrix Configuration for Centronics Jumpers

Table C-6. Centronics Interface Jumpers

Jumper Number	Signal Function
E6 to E16	STROBE*
E10 to E26	ACK*
E19 to E27	FAULT*
E24 to E25	BUSY
E64 to E96	PPI INT to inverter
E71 to E79	PPI INT* to IR5
E64 to E83	PPI INT to IR1

Note: * = Active low signal.

Table C-7. Centronics Interface Cable Pin Assignments

Signal	Model 702	Model 737-1	iSBC® 186/03 Pin
STROBE*	1	1	38 (E16 - E6)
Return	19	2	39 (GND)
D1	2	3	2 (PA-0)
Return	20	4	3 (GND)
D2	3	5	4 (PA-1)
Return	21	6	5 (GND)
D3	4	7	6 (PA-2)
Return	22	8	7 (GND)
D4	5	9	8 (PA-3)
Return	23	10	9 (GND)
D5	6	11	10 (PA-4)
Return	24	12	11 (GND)
D6	7	13	12 (PA-5)
Return	25	14	13 (GND)
D7	8	15	14 (PA-6)
Return	26	16	15 (GND)
D8	9	17	16 (PA-7)
Return	27	18	17 (GND)
ACK*	10	19	40 (E10 - E26)
Return	28	20	--
BUSY	11	21	50 (E24 - E25)
Return	29	22	49 (GND)
PE	12	NC	48 (PB-6)
Return	30	NC	47 (NC)
SLCT	13	25**	46 (PB-5)
Return	31	--	45 (NC)
FAULT*	32	--	44 (E19-E27)

Note: ** For printer model 737, SLCT (pin 25) must be tied to J3-44 for fault interrupt.

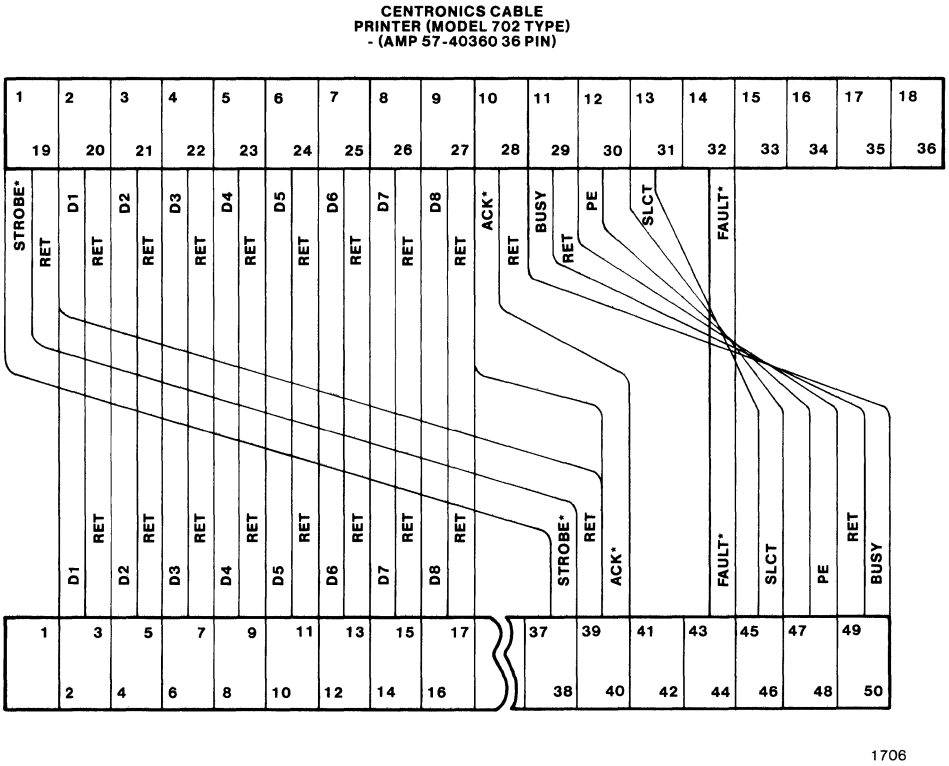


Figure C-8. Cable Fabrication Diagram

Centronics Interface Software Requirements

This application uses one DMA channel of the 80186 and three interrupts. The first interrupt is the Transfer Count interrupt from the 80186. The Transfer Count interrupt signal is generated when the last byte is sent by the DMA controller and signals the end of a DMA transfer. This does not signal that the printer has completed its print only that the DMA Controller has sent its last character to the printer.

The printer may require special characters to cause the print to occur such as a carriage return or a line feed. The interrupt service routine should write the proper interrupt masks to the 80130 and the 80186 devices and send the specific end of interrupt to each controller.

The second interrupt is a Fault interrupt generated by the printer to signal either a deselected condition or a paper empty. When the FAULT* signal becomes active, the printer BUSY signal goes active which blocks any more DMA requests from the Centronics Interface PAL. The FAULT* signal is inverted and routed to an interrupt input to generate an active high interrupt signal. Once a FAULT* interrupt is received, the software should read Port B of the 8255 device to determine which of the two conditions to service (either SLCT or PE). When the printer is deselected, the software must temporarily stop the DMA Controller until the printer comes back on line (which may require operator intervention).

When the printer returns on line from a de-selected condition, the FAULT* signal goes inactive. With the FAULT* signal connected directly to an interrupt input, this condition is used to generate a third (on line) interrupt. When this happens, the software must be delayed for at least 1 millisecond before restarting the DMA controller. The delay is necessary because the printer may generate spurious ACK signals to the Centronics PAL when it comes on line. Figure C-9 specifies the required FAULT* signal timing.

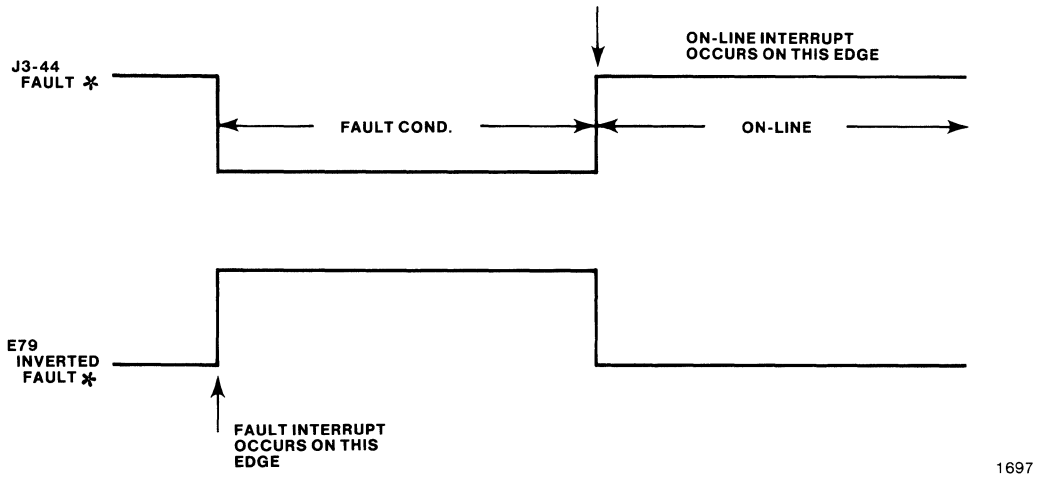


Figure C-9. FAULT* Timing Specification

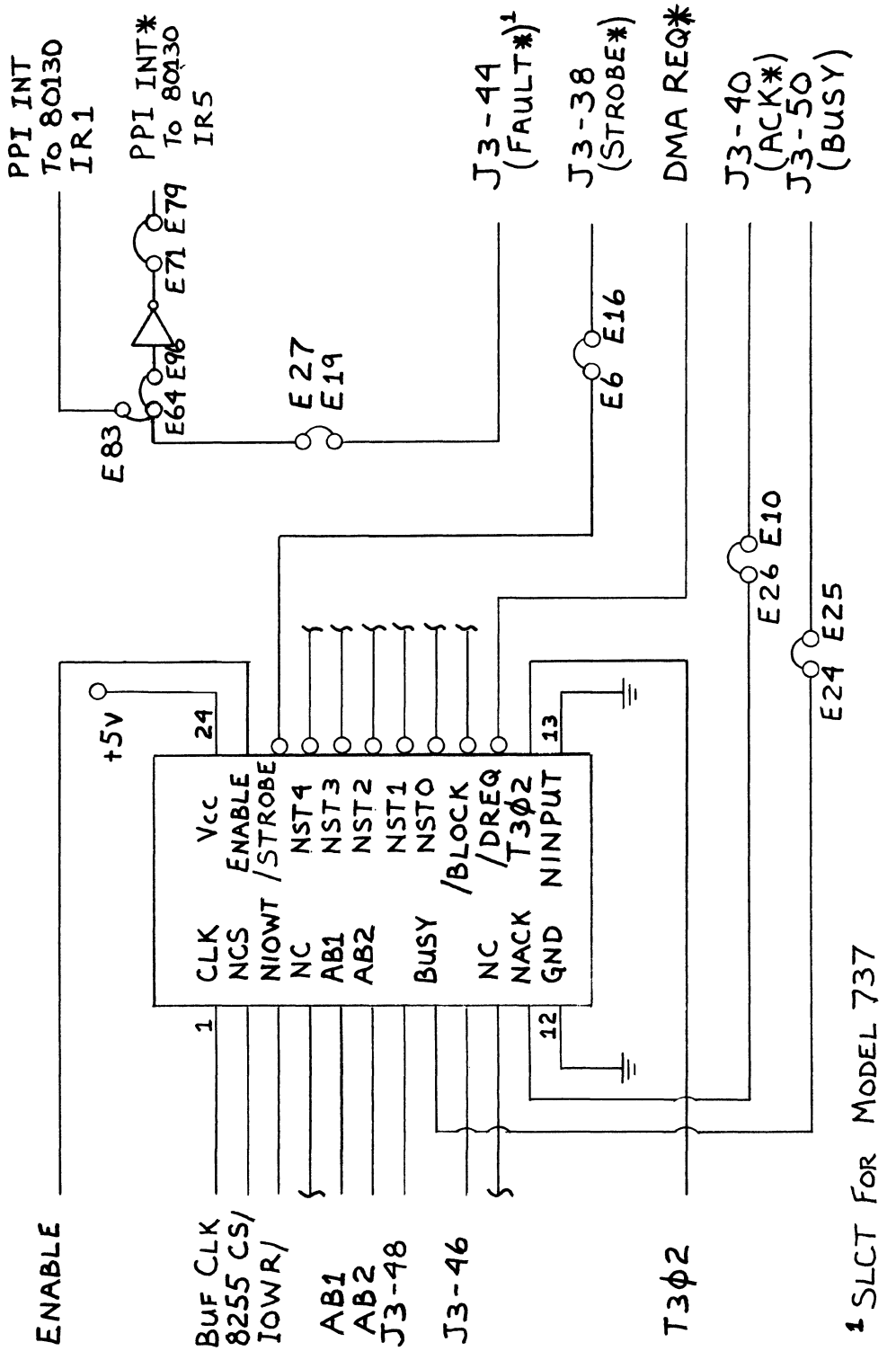


Figure C-10. Centronics PAL with Jumper Connections

***

APPENDIX D. UNIVERSAL MEMORY SITE REQUIREMENTS AND JUMPER CONFIGURATIONS

D.1. INTRODUCTION

This appendix covers three universal memory site topics: the electrical DC requirements for the 28-pin sockets; the timing requirements, as implemented on the iSBC 186/03 board; and the jumper configurations for the universal site matrices.

D.2. DC CHARACTERISTICS

Table D-1 outlines the DC characteristics for the 28-pin memory sites on the iSBC 186/03 board.

Table D-1. Universal Memory Site DC Specifications

Parameter	Minimum	Maximum	Units	Notes
V_{IL}	--	0.8	V	
V_{IH}	2.0	--	V	
I_{IL}	--	$\pm$10	uA	-0.5V $\leq$ Vin $\leq$ 0.4V
I_{IH}	--	$\pm$10	uA	5.25V $\geq$ Vin $\geq$ 2.4V
V_{OL}	--	0.4	V	I_{OL} = 2.1 mA
V_{OH}	2.4	--	V	I_{OH} = -400uA

D.3 JUMPER MATRIX CONFIGURATION

You will find two jumper matrices on the iSBC 186/03 board; one for local memory bank A (socket pair U42/U43 & U75/U76), and one for local memory bank B (socket pair U40/U41 and U73/74. You can configure each matrix independently for a different type of memory device; the following text describes how.

Each of the three jumper matrices consists of 15 stake pins arranged in a two rows. One missing pin in each matrix serves as a key to the orientation of that matrix. The pin arrangement is a standard format, as follows:

Address Bit A13	•	•	To pin 26 of 28 pin site
Address Bit A11	•	•	Vcc
To pin 23 of 28 pin site	•	•	Vcc/PGM
Write Enable Signal WE*	•	•	To pin 27 of 28 pin site
Missing pin (key)		•	A14 Address Bit
NVRAM Enable Signal NVEN*	•	•	To pin 1 of 28 pin site
Ready Signal RDY	•	•	A15 Address Bit
To pin 1 of 28 pin site	•	•	Vcc/Vpp

Configure each matrix by installing jumpers as shown in this appendix to place signals onto the proper pins of the memory devices. On the iSBC 186/03 board the two matrices are located directly above the memory sockets.

JEDEC 28 PIN SITE PINOUT

	NVRAM	iRAM	8Kx8 SRAM	2K × 8 SRAM+	8Kx8 E² PROM	2Kx8 E² PROM	27512	27256	27128	2764			2764	27128	27256	27512	2Kx8 E² PROM	8Kx8 E² PROM	2K × 8 SRAM+	8Kx8 SRAM	iRAM	NVRAM	
CONFIGURABLE	$\overline{NE}$	RDY	N.C.		RDY/$\overline{BSY}$	RDY/$\overline{BUSY}$	A15	VPP	VPP	VPP	1		28	VCC	VCC	VCC	VCC	VCC	VCC		VCC	VCC	VCC	
	N.C.	A12	A12		A12	N.C.	A12	A12	A12	A12	2		27	$\overline{PGM}$	$\overline{PGM}$	A14	A14	$\overline{WE}$	$\overline{WE}$		$\overline{WE}$	$\overline{WE}$	$\overline{WE}$	CONFIGURABLE
	A7	A7	A7	A7	A7	A7	A7	A7	A7	A7	3		26	N.C.	A13	A13	A13	N.C.	N.C.	VCC	N.C.	N.C.	N.C.	CONFIGURABLE
	A6	A6	A6	A6	A6	A6	A6	A6	A6	A6	4		25	A8	A8	A8	A8	A8	A8	A8	A8	A8	A8	
	A5	A5	A5	A5	A5	A5	A5	A5	A5	A5	5	28 PIN	24	A9	A9	A9	A9	A9	A9	A9	A9	A9	N.C.	
	A4	A4	A4	A4	A4	A4	A4	A4	A4	A4	6	SITE	23	A11	A11	A11	A11	N.C.	A11	WE	A11	A11	N.C.	CONFIGURABLE
	A3	A3	A3	A3	A3	A3	A3	A3	A3	A3	7		22	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$/VPP	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	
	A2	A2	A2	A2	A2	A2	A2	A2	A2	A2	8		21	A10	A10	A10	A10	A10	A10	A10	A10	A10	N.C.	
	A1	A1	A1	A1	A1	A1	A1	A1	A1	A1	9		20	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	$\overline{CE}$	
	A$\emptyset$	A$\emptyset$	A$\emptyset$	A$\emptyset$	A$\emptyset$	A$\emptyset$	A$\emptyset$	A$\emptyset$	A$\emptyset$	A$\emptyset$	10		19	O7	O7	O7	O7	I/O_7	I/O_7	I/O_7	I/O_7	I/O_7	I/O_7	
	$I/O_\emptyset$	I/O_O	I/O_O	$I/O_\emptyset$	$I/O_\emptyset$	$I/O_\emptyset$	O$\emptyset$	O$\emptyset$	O$\emptyset$	O$\emptyset$	11		18	O6	O6	O6	O6	I/O_6	I/O_6	I/O_6	I/O_6	I/O_6	I/O_6	
	I/O_1	I/O_1	I/O_1	I/O_1	I/O_1	I/O_1	O1	O1	O1	O1	12		17	O5	O5	O5	O5	I/O_5	I/O_5	I/O_5	I/O_5	I/O_5	I/O_5	
	I/O_2	I/O_2	I/O_2	I/O_2	I/O_2	I/O_2	O2	O2	O2	O2	13		16	O4	O4	O4	O4	I/O_4	I/O_4	I/O_4	I/O_4	I/O_4	I/O_4	
	GND	GND	GND	GND	GND	GND	GND	GND	GND	GND	14		15	O3	O3	O3	O3	I/O_3	I/O_3	I/O_3	I/O_3	I/O_3	I/O_3	

+24 PIN DEVICE

x-658

Figure D-1. Compatible Device Types

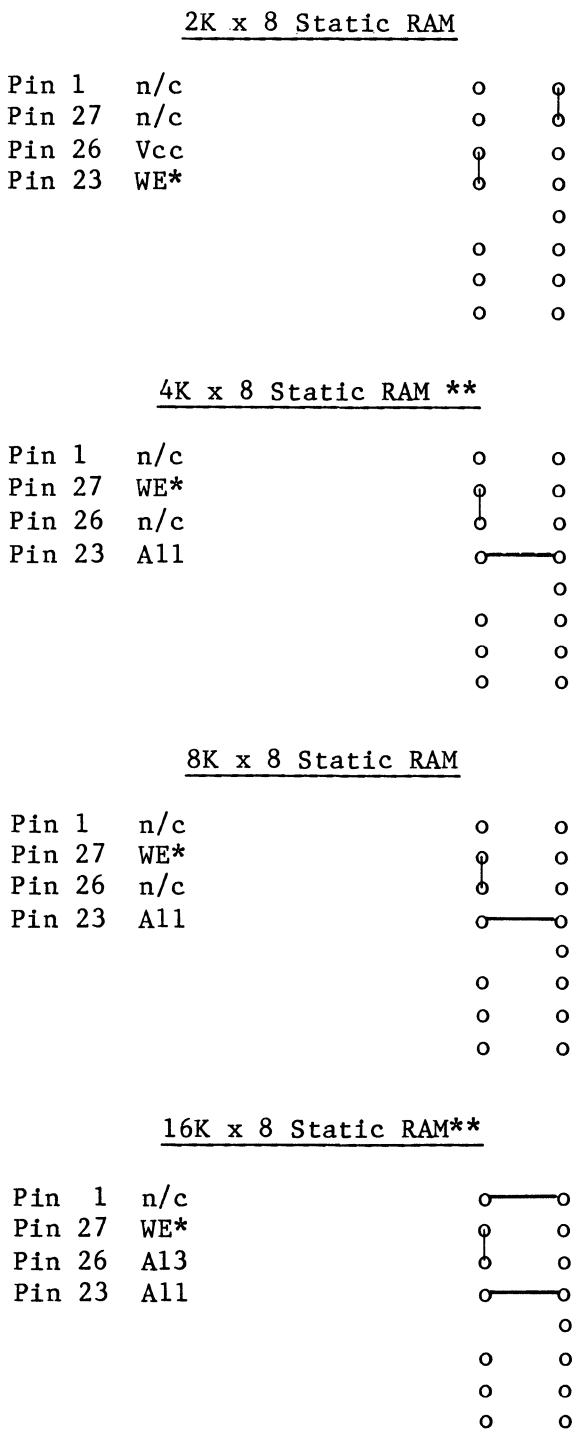


Figure D-2. Jumper Matrix Configurations for Static RAM Devices

Note: ** = Not supported by the memory decode options provided by the default PAL in U59. A user-programmed PAL can be used instead of the default PAL to obtain support.

8K x 8 iRAM

Pin 1	RDY	o o
Pin 27	WE*	o o
Pin 26	n/c	o o
Pin 23	A11	o——o
		o
		o o
		o o
		o o

16K x 8 iRAM**

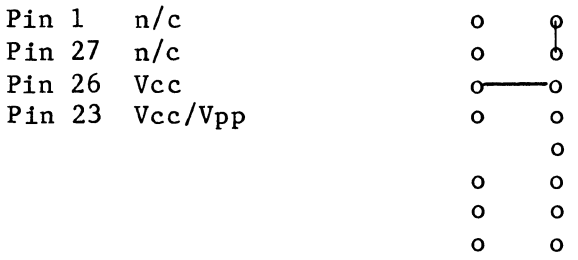
Pin 1	RDY	o——o
Pin 27	WE*	o o
Pin 26	A13	o o
Pin 23	A11	o——o
		o
		o o
		o o
		o o

Figure D-3. Jumper Matrix Configurations for iRAM Devices

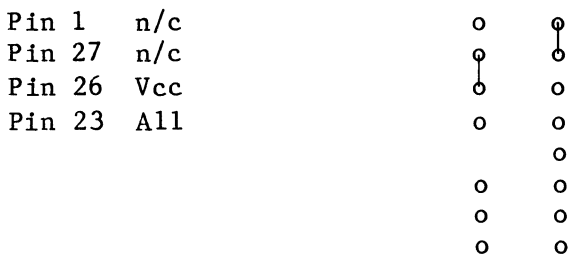
Note: ** = Not supported by the memory decode options provided by the default PAL in U59. A user-programmed PAL can be used instead of the default PAL to obtain support.

Jumper E238-E239 must also be installed when using iRAM devices. The iRAM devices are not allowed in Bank B.

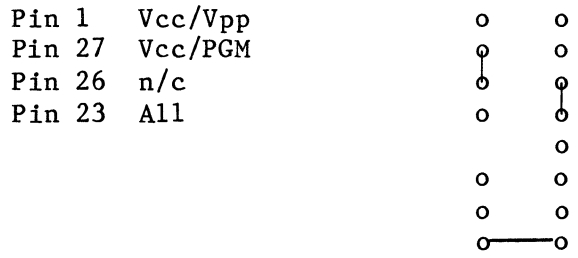
2K x 8 EPROM (2716)**



4K x 8 EPROM (2732A)**



8K x 8 EPROM (2764)



16K x 8 EPROM (27128)

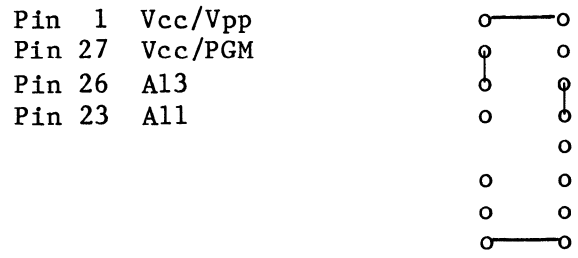
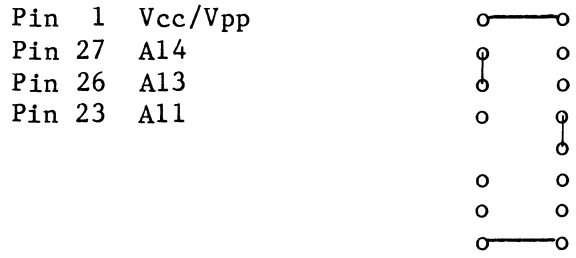


Figure D-4. Jumper Matrix Configurations for EPROM Devices

Note: ** = Not supported by the memory decode options provided by the default PAL in U59. A user-programmed PAL can be used instead of the default PAL to obtain support.

32K x 8 EPROM (27256)



64K x 8 EPROM (27512)

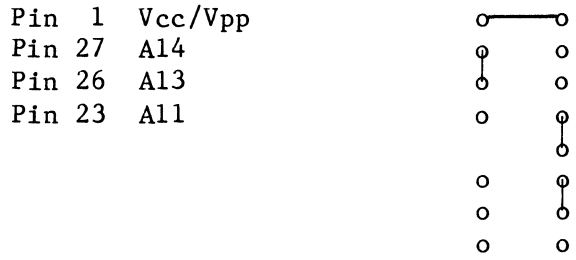
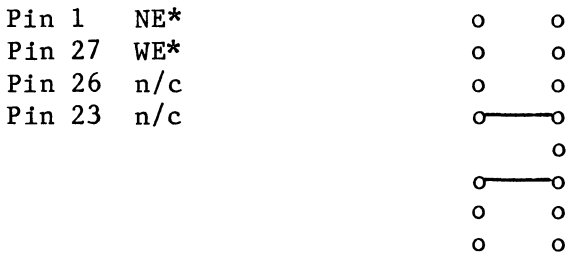
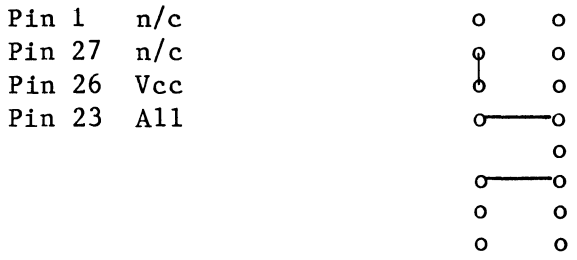


Figure D-4. Jumper Matrix Configurations for EPROM Devices (continued)

512 x 8; 1K x 8; 2K x 8 NVRAM



4K x 8; 8K x 8 NVRAM



16K x 8 NVRAM

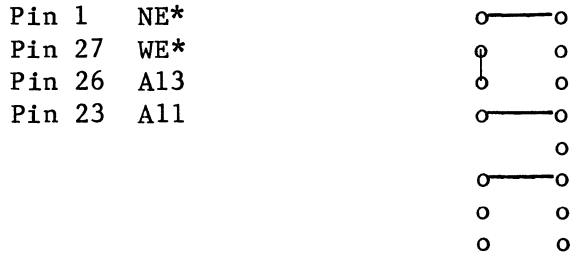
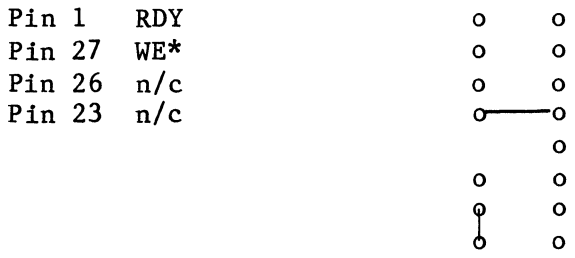


Figure D-5. Jumper Matrix Configurations for NVRAM Devices

2K x 8 EEPROM (2817A)



4K x 8 EEPROM (2865)**

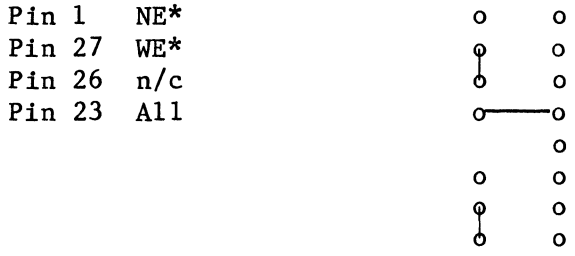


Figure D-6. Jumper Matrix Configurations for EEPROM Devices

Note: ** = Not supported by the memory decode options provided by the default PAL in U59. A user-programmed PAL can be used instead of the default PAL to obtain support.

Table D-2. Normal READ Cycle at 6MHz

	0 Wait		1 Wait		2 Wait	
	Min	Max	Min	Max	Min	Max
t1 Read Cycle time	423ns		589ns		755ns	
t2 Access time from Address		285ns		410ns		535ns
t3 Access time from Chip Enable						
Bank B sites		347ns		513ns		679ns
Bank A sites		302ns		468ns		634ns
t4 Access time from Output Enable		149ns		315ns		481ns
t5 Output Enable high to Data Tri-State		141ns		141ns		141ns
t6 Chip Enable to Ready Low to guarantee $\geq$ n wait states (assumes ready logic is jumpered for n wait states)						
Bank B sites		-----		161ns		327ns
Bank A sites		-----		116ns		282ns
t7 Ready high to Read Data Valid	241ns		241ns		241ns	

Table D-3. Normal WRITE Cycle at 6MHz

		0 Wait		1 Wait		2 Wait	
		Min	Max	Min	Max	Min	Max
t8	Address Valid to WE* high	423ns		589ns		755ns	
t9	Address set up to WE* low	176ns		176ns		176ns	
t10	Chip Enable* to WE* low						
	Bank B sites	136ns		136ns		136ns	
	Bank A sites	91ns		91ns		91ns	
t11	Chip Enable* to WE* high	257ns		423ns		589ns	
t12	Write Data Setup	298ns		464ns		630ns	
t13	Write Data Hold	141ns		141ns		141ns	

Table D-4. iRAM READ Cycle at 6MHz (Bank A Only)

		1 Wait		2 Wait		3 Wait	
		Min	Max	Min	Max	Min	Max
t20	Read Cycle time	589ns		755ns		921ns	
t21	Address Setup to Chip Enable*	106ns		106ns		106ns	
t22	Chip Enable* to Read Data Valid		336ns		559ns		725ns
t23	Chip Enable* High Time	281ns		281ns		281ns	
t24	Output Enable*/Write Enable High to next CE*	322ns		322ns		322ns	
t25	Chip Enable* to Ready Low to quarantee $\geq$ n wait states (assumes external ready circuit is jumpered for n wait states)		96ns		262ns		428ns
t26	Ready High to Read Data Valid	241ns		241ns		241ns	
t27	Output Enable to Read Data Valid		315ns		481ns		647ns

Table D-5. iRAM WRITE Cycle at 6MHz (Bank A Only)

	1 Wait		2 Wait		3 Wait	
	Min	Max	Min	Max	Min	Max
t28 Address Setup to WE* Low	176ns		176ns		176ns	
t29 Chip Enable* Setup to WE* Low	10ns		10ns		10ns	
t30 Write Data Setup to WE* Low	51ns		51ns		51ns	
t31 Write Data Hold from WE* Low	517ns		683ns		849ns	
t32 WE* Low to Next CE* Low	698ns		864ns		1030ns	

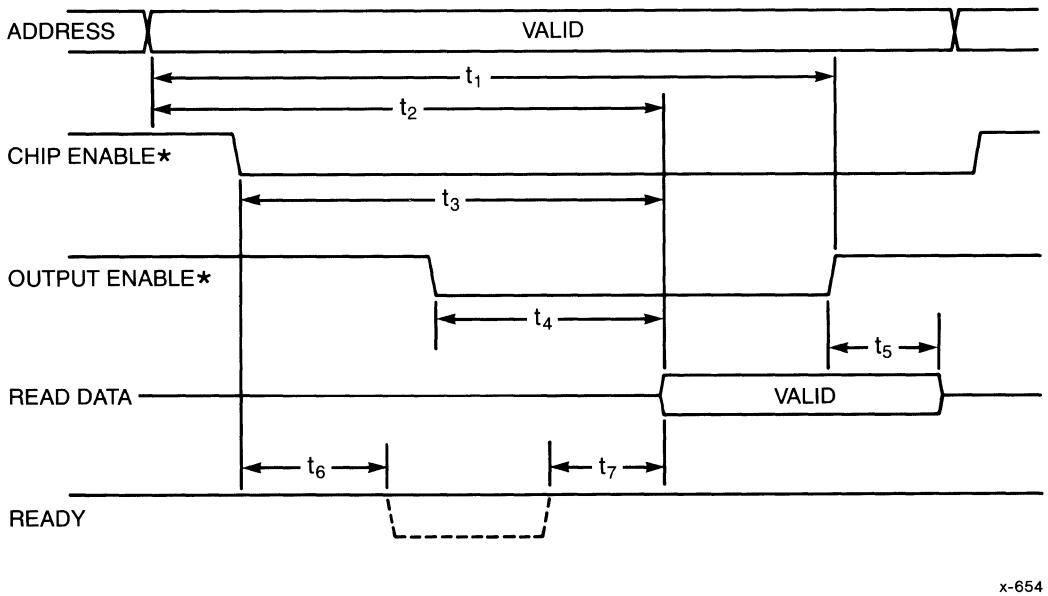


Figure D-7. Normal READ Cycle

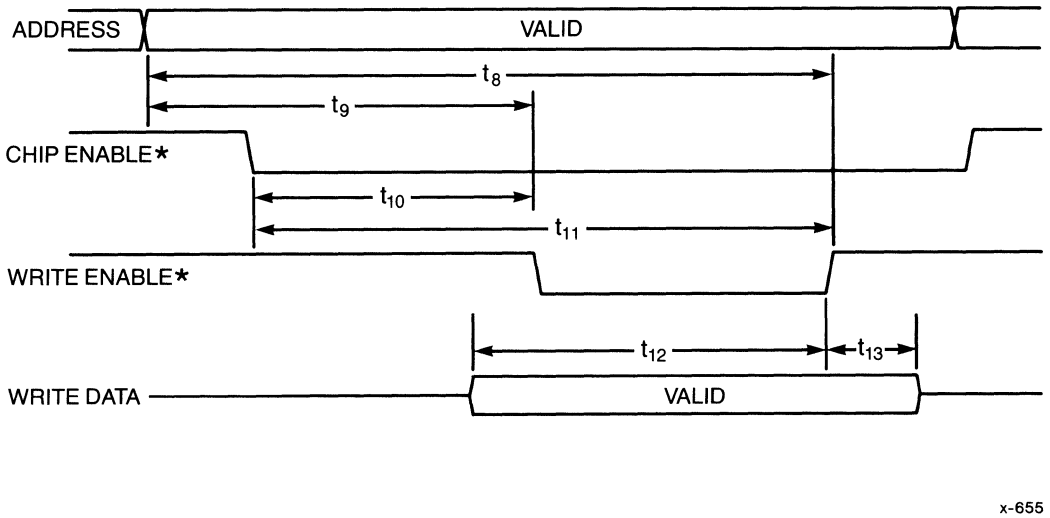


Figure D-8. Normal WRITE Cycle

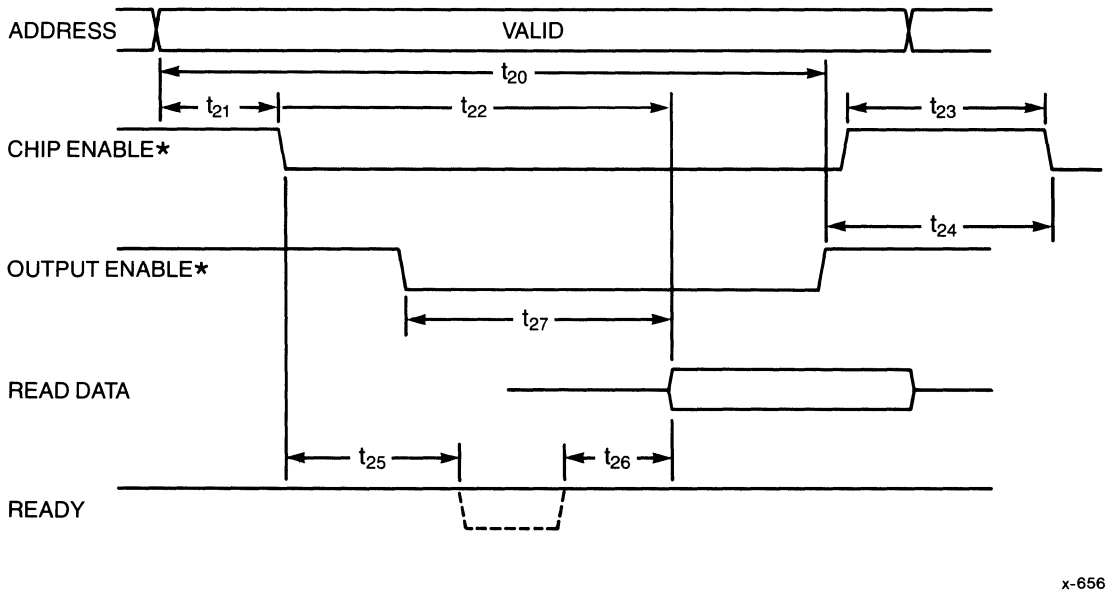


Figure D-9. iRAM READ Cycle

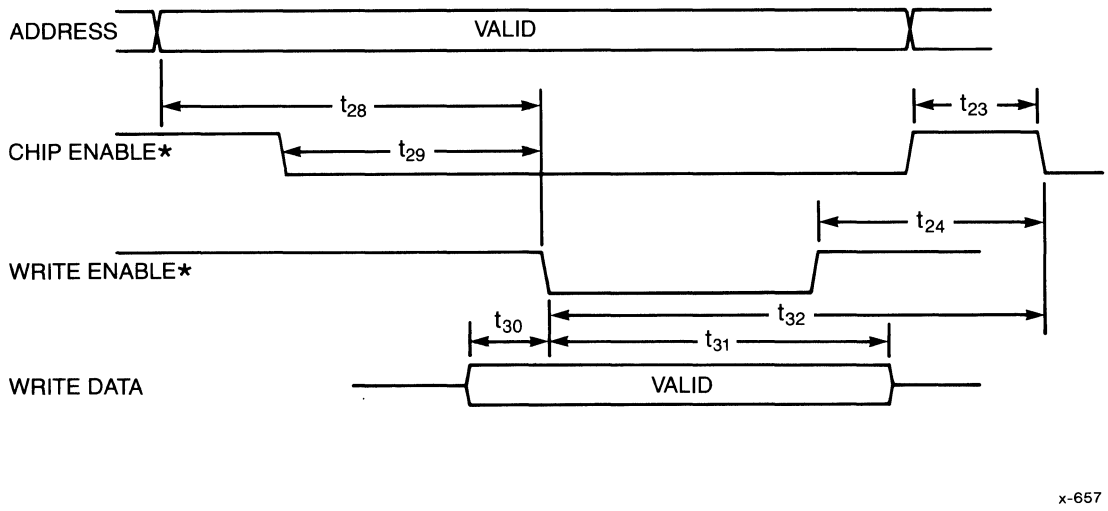


Figure D-10. iRAM WRITE Cycle

***

APPENDIX E. iSBC® 341 MEMORY EXPANSION BOARD INSTALLATION

E.1 INTRODUCTION

This appendix provides information for installing the iSBC 341 Memory Expansion MULTIMODULE Board onto the iSBC 186/03 Single Board Computer. You can expand the number of 28-pin memory sockets on the iSBC 186/03 board from 8 to 12 by installing an iSBC 341 Memory Expansion MULTIMODULE Board. The MULTIMODULE board effectively doubles the on-board RAM capacity.

E.2 iSBC® 341 INSTALLATION

The installation of the iSBC 341 Memory Expansion MULTIMODULE board adds four additional memory locations, labeled U2, U3, U5, and U6. To install the iSBC 341 board, proceed as follows:

1. Unpack the iSBC 341 Memory Expansion MULTIMODULE board.

2. Inspect the iSBC 341 Memory Expansion MULTIMODULE board for damage. If damage exists, follow the instructions for repairs in Chapter 5 of this manual.

3. Modify the jumpers on the iSBC 341 Memory Expansion MULTIMODULE board to provide the required operation; refer to paragraph E.3 for jumper configurations.

4. Install the user-supplied memory devices onto the iSBC 341 Memory Expansion MULTIMODULE board.

CAUTION

The iSBC 186/03 board is designed to accommodate both 24- and 28-pin Intel memory chips in the same socket. Ensure that pin 1 of the device is in the proper location.

5. Trim the leads of the memory devices (installed in step 4) at the end of the connectors.

6. Ensure that system power is off.

7. Remove the iSBC 186/03 board from the backplane and place it on a soft surface (preferably a piece of foam), component side up.

8. Remove the memory devices from locations U43 and U76 on the iSBC 186/03 board, if installed.

9. Hold the iSBC 186/03 board on edge and install the three screws (reference Figure E-1), from the solder side.

10. Place a spacer on each of the screws.

11. Install the iSBC 341 Memory Expansion MULTIMODULE board on the iSBC 186/03 board in the location shown in Figure E-1.

12. Press the iSBC 341 Memory Expansion MULTIMODULE board into place by pressing at locations U1, U4, and U6.

13. Install the three nuts and tighten them finger tight.

14. Tighten the three nuts with a nut driver.

CAUTION

Do not overtighten the screws. Damage to the board could result.

15. Reinstall the memory devices, removed from U43 and U76 on the iSBC 186/03 board in step 8, into locations U1 and U4 on the iSBC 341 Memory Expansion MULTIMODULE Board.

CAUTION

Ensure that the memory devices are properly oriented in their sockets or they could be damaged when power is applied.

E.3 JUMPER CONFIGURATION INFORMATION

The iSBC 341 Memory Expansion MULTIMODULE board is shipped from the factory with the following jumpers installed (refer to the schematic diagram, Figure E-3):

E9-E10	E39-E43	E13-E14	E47-E50
E17-E18	E30-E31	E53-E57	E37-E41
E25-E26	E1-E2	E21-E22	E35-E36

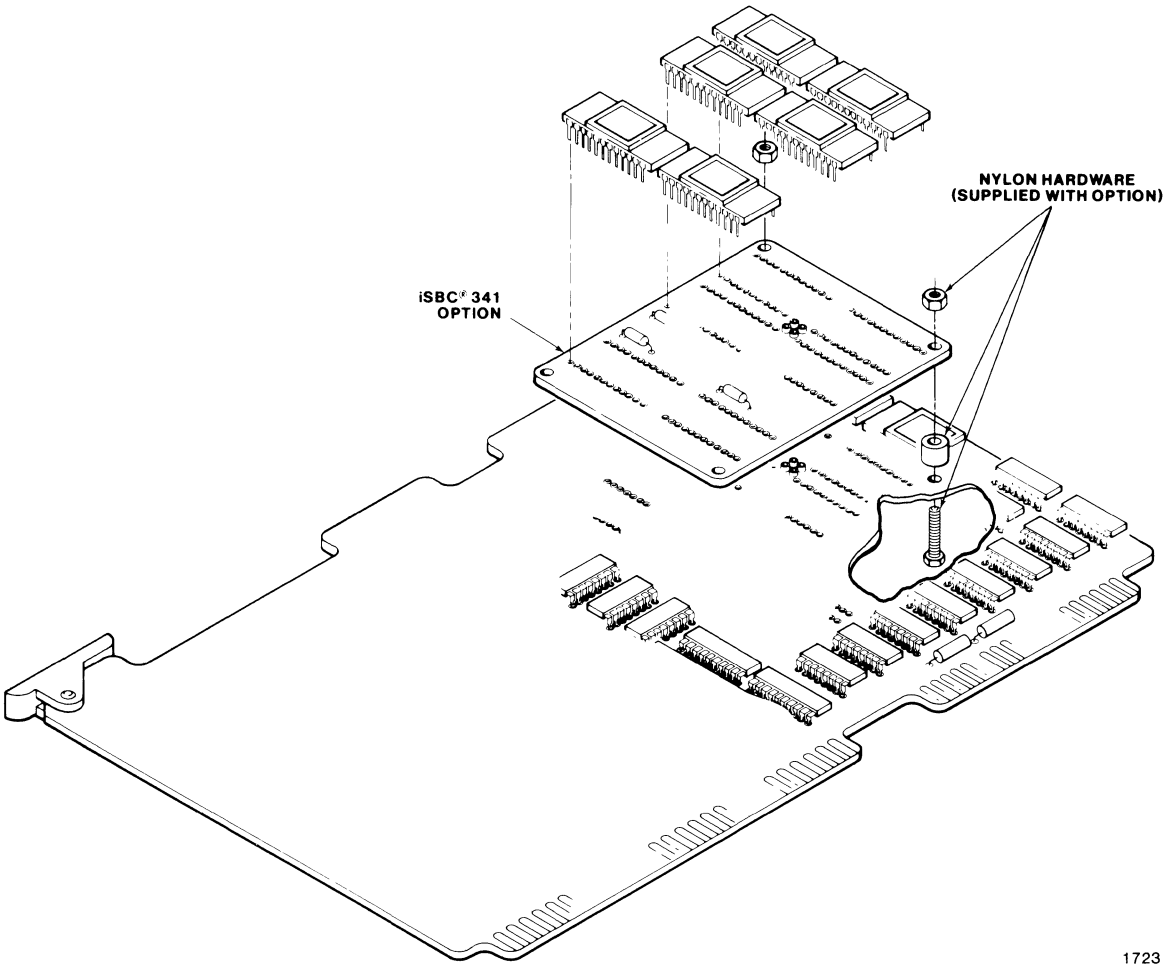


Figure E-1. iSBC® 341 Memory Expansion MULTIMODULE™ Board Orientation

This configuration matches the iSBC 341 board to the configuration of the iSBC 186/03 board. For example, if the iSBC 186/03 board is configured to accept a particular capacity of memory devices, this jumper configuration allows the iSBC 341 board to accept the same capacity and type of memory device.

If the devices installed onto the iSBC 341 Board are different from those installed on the baseboard then jumper modifications are required for installation of SRAM, iRAM, NVRAM, and EEPROM devices onto the iSBC 341 board as shown in Table E-1

Table E-1. Jumper Configurations for Types of Memory Devices

In iSBC® 341 Sockets	Disconnect:	Connect:
STATIC RAM REQUIREMENTS:		
U2 and U5	E39 - E43	E40 - E44
U3 and U6	E37 - E41	E38 - E42
iRAM REQUIREMENTS:		
U2 and U5	E9 - E10 E39 - E43	E7 - E15 E40 - E44
U3 and U6	E1 - E2 E37 - E41	E4 - E11 E38 - E42
EEPROM REQUIREMENTS:		
U2 and U5	E39 - E43 E53 - E57 E30 - E31 E9 - E10	E7 - E15 E40 - E44
U3 and U6	E37 - E41 E47 - E50 E35 - E36 E1 - E2	E4 - E11 E38 - E42
NVRAM REQUIREMENTS:		
U2 and U5	E39 - E43 E53 - E57 E30 - E31 E9 - E10	E6 - E7 E40 - E44
U3 and U6	E37 - E41 E47 - E50 E35 - E36 E1 - E2	E3 - E4 E38 - E42

NOTE

The iSBC 186/03 board does not support EEPROM devices that require a 21 volt programming signal.

E.4 REFERENCE DIAGRAMS

The iSBC 341 board parts location diagram and schematic diagram are provided in Figures E-2 and E-3, respectively.

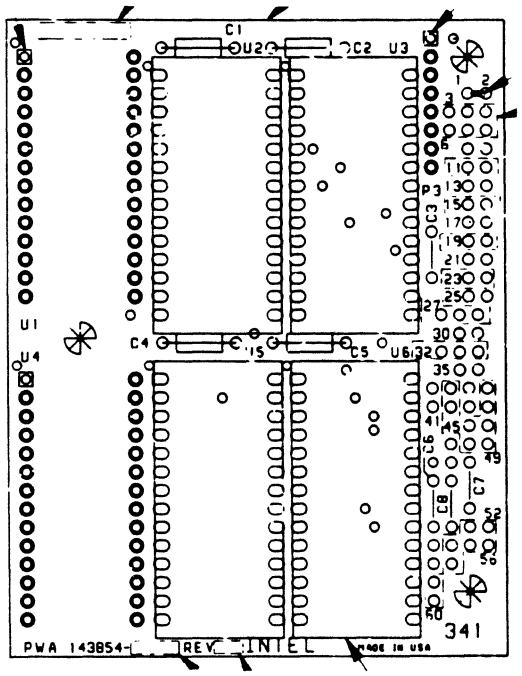


Figure E-2. Parts Location Diagram

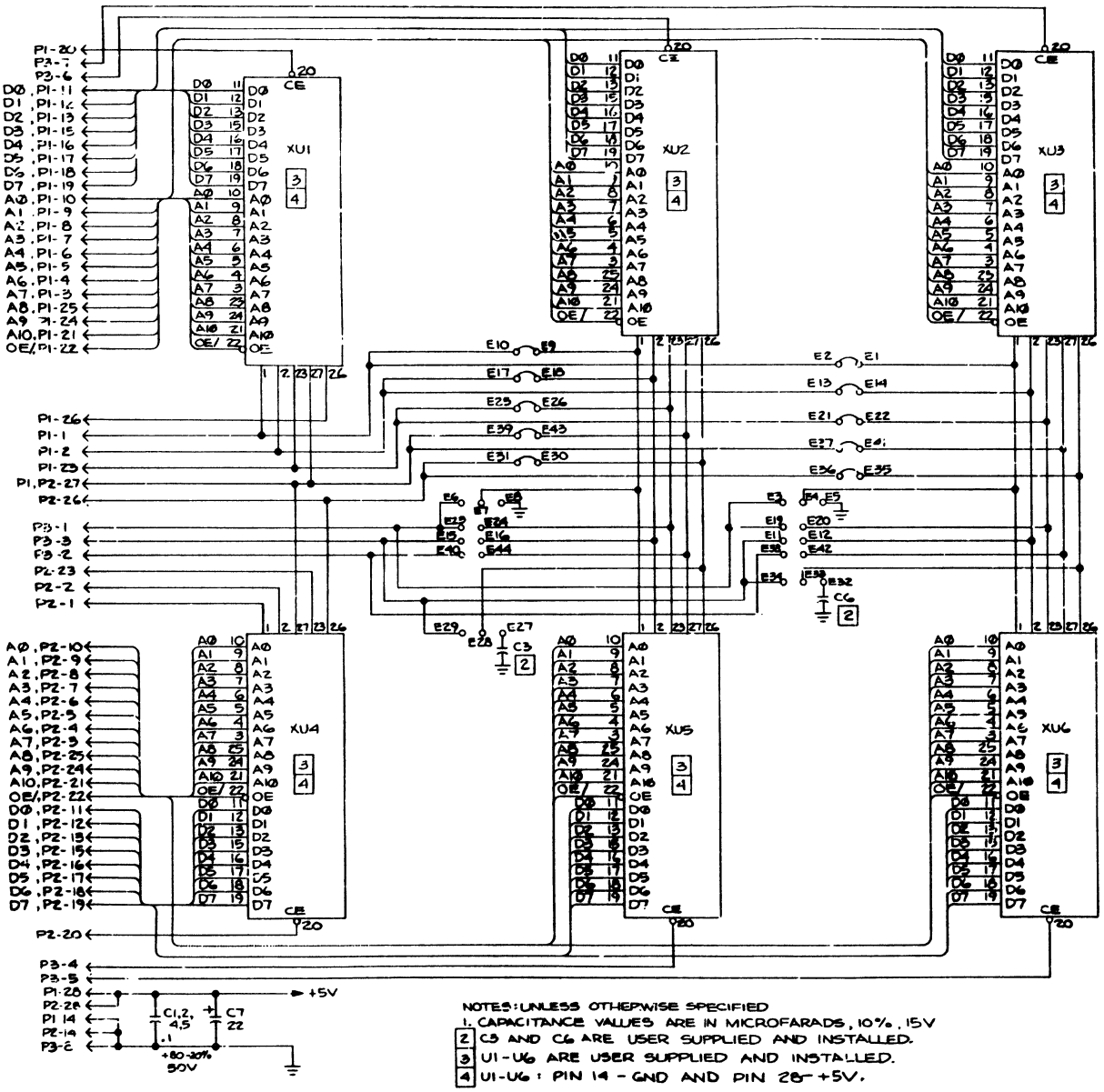


Figure E-3. Schematic Diagram

***

APPENDIX F. iSBC® 186/03 BOARD PROGRAMMING EXAMPLES

F.1 INTRODUCTION

This appendix provides a group of sample initialization procedures for the key programmable devices on the iSBC 186/03 board. Included in this example are initialization procedures for the 80186, the 80130, and the 8274. Refer to the comments within the program for more details.

```
/*****************************************************************


The programming examples for the iSBC 186/03 board includes an example
that initializes the 80186, the 8274 MPSC, and the 80130 PIC on the
iSBC 186/03. Also included is an example that initializes the 80186
DMA controller, and an example that places the 8274 device into
general transmit operation.
More details may be found in the Intel Application Notes entitled
ASYNCHRONOUS COMMUNICATION WITH THE 8274 MULTIPLE PROTOCOL SERIAL
CONTROLLER, AP-134; and SYNCRHRONOUS COMMUNICATION WITH THE 8274
MULTIPLE PROTOCOL SERIAL CONTROLLER, AP-145.

*****************************************************************/

/*80186 REGISTER ADDRESSES*/

declare relo$reg$186        literally 'OFFfeh',
        mpcs$reg$186        literally 'OFFa8h',
        pacs$reg$186        literally 'OFFa8h',
        vcs$reg$186         literally 'OFFa0h',
        eoi$reg$186         literally 'OFF22h',
        mask$reg$186        literally 'OFF28h',
        pri$reg$186         literally 'OFF2ah',
        dma0$int$reg$186    literally 'OFF34h',
        dma1$int$reg$186    literally 'OFF36h';

declare tmr0$186$cntl       literally 'Off56h',
        tmr0$186$mcntb      literally 'Off54h',
        tmr0$186$mcnta      literally 'Off52h',
        tmr0$186$cnt        literally 'Off50h',
        tmr1$186$cntl       literally 'Off5eh',
        tmr1$186$mcntb      literally 'Off5ch',
        tmr1$186$mcnta      literally 'Off5ah',
        tmr1$186$cnt        literally 'Off58h';


/*DMA CONTROL, SOURCE, AND DESTINATION ADDRESSES*/

declare dma0$cntrl$reg      literally 'OFFcah',
        dma0$count$reg      literally 'OFFc8h',
        dma0$dest$h$reg     literally 'OFFc6h',
        dma0$dest$l$reg     literally 'OFFc4h',
        dma0$src$h$reg      literally 'OFFc2h',
        dma0$src$l$reg      literally 'OFFc0h',

        dma1$cntrl$reg      literally 'OFFdah',
        dma1$count$reg      literally 'OFFd8h',
        dma1$dest$h$reg     literally 'OFFd6h',
        dma1$dest$l$reg     literally 'OFFd4h',
        dma1$src$h$reg      literally 'OFFd2h',
        dma1$src$l$reg      literally 'OFFd0h';
```

Figure F-1. iSBC® 186/03 Programming Example

```
/*80186 REGISTER VALUES*/
        /* rmx mode, no trap, io space */
declare relo$reg$value      literally '040FFh', /*rmx mode,no trap,io space*/
        mpcs$reg$value      literally '080bbh', /*peripherals mapped to IO,no

                                                                    lmcs*/
        pacs$reg$value      literally '00038h', /*pcs base at 0h,0 wait state */
        vmcs$reg$value      literally '0FFF8h', /*vcs = 1k block,0 wait state*/
        dma1$cntrl$stopped literally '07684h',
        dma1$cntrl$started literally '07686h',
        dma0$cntrl$stopped literally '0ae64h',
        dma0$cntrl$started literally '0ae66h',
        dma0$cntrl$start$m literally '0b727h',
        dma1$cntrl$start$m literally '0b727h';

/*80130 ADDRESSES*/
declare icw$reg$130         literally '0e0h',
        ocw$reg$130         literally '0e2h';
/*80130 COMMAND VALUES*/
declare icw1$130            literally '039h', /* icw6 needed,level trig */
        icw2$130            literally '020h', /* vector base  IR0 = level 32 */
        icw3$130            literally '058h', /* slaves on IR3, IR4, IR6     */
        icw4$130            literally '00dh', /* fully nested mode           */
        icw6$130            literally '0ffh'; /* LIR=0 for all intr.         */
/*PROGRAM MASK VALUES*/
declare mask$all$130        literally '0ffh',
        unmask$timer$130    literally '0fbh',  /* systick on IR2 level 34 */
        unmask$186          literally '0efh',  /* 186 on IR4 level 36     */
        unmask$8274         literally '0f7h',  /* 8274 on IR3 level 35    */
        unmask$dma0         literally '039h',  /* dma0 on level 2         */
        timer$eoi$130       literally '066h';  /* eoi for IR6             */
/*8274 COMMAND, DATA, AND STATUS ADDRESSES*/
declare command_a_74        literally '0dch',
        command_b_74        literally '0deh',
        status_a_74         literally '0dch',
        status_b_74         literally '0deh',
        data_a_74           literally '0d8h',
        data_b_74           literally '0dah';
/*PARAMETERS TO INITIALIZE DMA MULTIPLEXER*/
declare dma_sel0_on         literally '0f1h',
        dma_sel0_off        literally '0f0h',
        dma_sel1_on         literally '0f3h',
        dma_sel1_off        literally '0f2h',
        dma_sel2_on         literally '0f5h',
        dma_sel2_off        literally '0f4h',
        dma_tdma0_pl        literally '0f6h',
        dma_tdma1_pl        literally '0f7h';

declare int_vectors(256) pointer at (0);
$EJECT
```

Figure F-1. iSBC® 186/03 Programming Example (continued)

```
INIT_186:
DO;
$INCLUDE (:F1:GPORTS.SRC)

init_186: procedure ;

  disable;
  outword(relo$reg$186) = relo$reg$value;
  outword(mpcs$reg$186) = mpcs$reg$value;
  outword(pacs$reg$186) = pacs$reg$value;

/* 80130 pic initialization  */

  output(icw$reg$130) = icw1$130;
  output(ocw$reg$130) = icw2$130;
  output(ocw$reg$130) = icw3$130;
  output(ocw$reg$130) = icw4$130;
  output(ocw$reg$130) = icw6$130;
  output(ocw$reg$130) = mask$all$130;

/* 186 pic initialization  */

  outword(vcs$reg$186) = 040h;    /*  slave level  IRS0 = level 64  */
  outword(dma0$int$reg$186) = 0ah;  /*  dma0 = level 2   masked       */
  outword(pri$reg$186) = 02h;       /*  mask all levels below 2       */
  outword(mask$reg$186) = 03dh;     /*  mask all interrupts           */
end init_186;


/*ROUTINE TO INITIALIZE 80186 TIMERS 0 AND 1 FOR BAUD RATE*/

init_baud: procedure(baud_count_a,baud_count_b) public;
  declare (baud_count_a,baud_count_b) word;

  outword(tmr0$186$cnt) = (baud_count_a/2);
  outword(tmr0$186$mcnta) = ((baud_count_a/2) + 1);
  outword(tmr0$186$mcntb) = ((baud_count_a/2) + 1);
  outword(tmr0$186$cntl) = 0c003h;
  outword(tmr1$186$cnt) = (baud_count_b/2);
  outword(tmr1$186$mcnta) = ((baud_count_b/2) + 1);
  outword(tmr1$186$mcntb) = ((baud_count_b/2) + 1);
  outword(tmr1$186$cntl) = 0c003h;

end init_baud;
END INIT_186;
$eject
```

Figure F-1. iSBC® 186/03 Programming Example (continued)

```
/*************************************************************************/
/*                                                                       */
/*       INITIALIZE THE 8274 FOR SDLC, AND ASYNCRONOUS MODES             */
/*                                                                       */
/*                          CHANNEL A                                    */
/*                                                                       */
/*       1. RESET CHANNEL                                                */
/*       2. EXTERNAL INTERRUPTS ENABLED                                  */
/*       3. NO WAIT                                                      */
/*       4. PIN 10 = RTS                                                 */
/*       5. VECTORED INTERRUPT-8086 MODE                                 */
/*       6. CHANNEL A DMA, CH B INT                                      */
/*       7. TX AND RX = 8 BITS/CHAR                                      */
/*       9. ADDRESS SEARCH MODE                                          */
/*       10.CD AND CTS AUTO ENABLE                                       */
/*       11.X1 CLOCK                                                     */
/*       12.NO PARITY                                                    */
/*       13.SDLC/HDLC MODE                                               */
/*       14.RTS AND DTR                                                  */
/*       15.CCITT - CRC                                                  */
/*       16.TRANSMITTER AND RECEIVER ENABLED                             */
/*       17.7EH = FLAG                                                   */
/*       18.ADDRESS = 055H                                               */
/*                                                                       */
/*                     CHANNEL B                                         */
/*                                                                       */
/*       1.RESET CHANNEL                                                     */
/*       2.INTERRUPT BASE = 80D                                          */
/*       3.ASYNCRONOUS,X16 CLOCK,1.5 STOP BITS,NO PARITY                 */
/*       4.8 RECEIVE BITS, 8 TRANSMIT BITS                               */
/*       5.INTERRUPT ON ALL RECEIVE CHARACTERS AND SPECIALS              */
/*                                                                       */
/*************************************************************************/

INIT_8274:
DO;

INIT_8274:      PROCEDURE;

  DECLARE       C       BYTE;
  DECLARE(COMMAND_A_74,COMMAND_B_74)BYTE;

/* TABLE TO INITIALIZE THE 8274 CHANNEL A IN SDLC MODE,          */
/* AND CHANNEL B IN ASYNC. MODE . CHANNEL A IS DMA, AND          */
/* CHANNEL B IS INTERRUPT.                                       */
/*      FORMAT IS: WRITE REGISTER, REGISTER DATA                 */
$EJECT
```

Figure F-1. iSBC® 186/03 Programming Example (continued)

```
/*CHANNEL A - SDLC/HDLC MODE */

DECLARE TABLE_74_A(*) BYTE DATA
        (00H,18H,       /* CHANNEL RESET */
        00H,80H,        /* RESET TX CRC */
        02H,31H,        /* PIN 10=RTSB, A DMA, B INT, vectored 8086 mode  */
        04H,20H,        /* SDLC/HDLC MODE, NO PARITY */
        07H,07EH,       /* SDLC FLAG */
        01H,0BH,        /* RX DMA ENABLE */
        05H,06BH,       /* NO DTR, RTS, 8 TX BITS, TX ENABLE, TX CRC ENABLE */
        06H,055H,       /* DEFAULT ADDRESS BYTE */
        03H,0CDH,       /* 8 RX BITS, NO AUTO ENABLES NEC, NO HUNT MODE, */
                        /* RX CRC ENABLE */
        00H,10H,        /* RESET EXT/STAT INTS */
        0FFH);          /* END OF INITIALIZATION TABLE */

/*CHANNEL B ASYNC. MODE*/
DECLARE TABLE_74_B(*) BYTE DATA
        (00H,18H,       /* CHANNEL RESET */
        02H,50H,        /* INTERRUPT VECTOR  BASE LEVEL = 80D  */
        04H,48H,        /* ASYNC. MODE,16X CLOCK, 1.5 STOP BITS,NO PARITY*/
        03H,0E1H,       /* 8 RECEIVE,AUTO ENABLED,Rx ENABLED*/
        05H,0EAH,       /* DTR-RTS ENABLE,8 BITS/CHAR.,Tx ENABLE*/
        01H,01CH,       /* INT. ON ALL RECEIVE CHAR.,VARIABLE VECTOR,NO DMA*/
        0FFH);          /* END */

/* INITIALIZE THE 8274 */

C=0;
DO WHILE TABLE_74_B(C) <> 0FFH;
  OUTPUT(COMMAND_B_74) = TABLE_74_B(C);
        C=C+1;
        OUTPUT(COMMAND_B_74) = TABLE_74_B(C);
        C=C+1;
END;

C=0;
DO WHILE TABLE_74_A(C) <> 0FFH;
        OUTPUT(COMMAND_A_74) = TABLE_74_A(C);
        C=C+1;
        OUTPUT(COMMAND_A_74) = TABLE_74_A(C);
        C=C+1;
END;
RETURN;

END;
END INIT_8274;
$EJECT
```

Figure F-1. iSBC® 186/03 Programming Example (continued)

```
/*************************************************************************/
/*                                                                       */
/*                  SAMPLE 8274 INTERRUPT ROUTINES                       */
/*                                                                       */
/*************************************************************************/

ENABLE_8274_INTERRUPTS:
do;
$INCLUDE (:F1:GPORTS.SRC)

level$0$8274: procedure interrupt 80;  /*  ch-B tx buffer empty */

/*OUTPUT CHARACTER TO TRANSMITTER*/
  output(command_a_74) = 38h;          /* eoi to 8274 */
  output(icw$reg$130) = 63h;           /* eoi to 80130 level 3 */
end level$0$8274;


level$2$8274: procedure interrupt 82;  /*  ch-B rx char avail    */

     /* AT THIS POINT,YOU MUST READ THE RECEIVED CHARACTER */
     /* FROM THE 8274 CHANNEL B DATA PORT*/

  output(command_a_74) = 38h;           /* eoi to 8274 */
  output(icw$reg$130) = 63h;            /* eoi to 80130 level 3 */
end level$2$8274;

     /* CHANNEL A RECEIVE CHARACTER AVAILABLE */
     /* START DMA AFTER FIRST RECEIVED CHARACTER*/

level$6$8274: PROCEDURE interrupt 86;

  outword(dma0$cntrl$reg) = dma0$cntrl$started;
  output(command_a_74) = 38h;           /* eoi to 8274 */
  output(icw$reg$130) = 63h;            /* eoi to 80130 level 3 */
  RETURN;
END level$6$8274;

DEFAULT$INT: PROCEDURE INTERRUPT 0;

     /* INTERRUPT ROUTINE TO HANDLE UNKNOWN INTERRUPTS*/

END DEFAULT$INT;
$eject
```

Figure F-1. iSBC® 186/03 Programming Example (continued)

```
/* ENABLE 8274 INTERRUPTS */

     /* ONLY ROUTINES FOR 8274 LEVELS 0,2 AND 6 ARE SHOWN IN      */
     /* THIS EXAMPLE. ALL OTHER INTERRUPT LEVELS MUST BE          */
     /* PROPERLY INITIALIZED.                                     */

enable_8274_interrupts: procedure;
    declare index byte;
    disable;
    do index = 0 to 255;
       int_vectors(index) = interrupt$ptr(default$int);
    end;
      /* NOTE:                                                   */
      /* default$int IS A LOOP THAT INITALIZES ALL INT. VECTORS */
      /* FOR A SPECIFIC "DEFAULT" INTERRUPT PROCEDURE            */
    do;

     call set$interrupt(80,level$0$8274);
     call set$interrupt(82,level$2$8274);
     call set$interrupt(86,level$6$8274);

     output(ocw$reg$130) = unmask$8274;
     enable;
     end;
end;
END ENABLE_8274_INTERRUPTS;
$eject
```

Figure F-1. iSBC® 186/03 Programming Example (continued)

```
/*************************************************************/
/*                                                           */
/* THIS PROCEDURE INITIALIZES THE 80186 DMA CONTROLLER       */
/*                                                           */
/*      1. DRQ0=8274  CHANNEL A RECEIVE                      */
/*      2. DRQ1=8274 CHANNEL A TRANSMIT                      */
/* PARAMETERS USED ARE DEFINED AS:                           */
/*      1. TX_LENGTH-TRANSMIT BUFFER LENGTH                  */
/*      2. RX_LENGTH-RECEIVER BUFFER LENGTH                  */
/*      3. DEST_ADDR_SEG- THE SEGMENT PORTION OF THE         */
/*         DESTINATION MEMORY ADDRESS FOR DMA CHANNEL 0      */
/*      4. DEST_ADDR_OFFSET- THE OFFSET PORTION OF THE       */
/*         DESTINATION MEMORY ADDRESS FOR DMA CHANNEL 0      */
/*      5. SOURCE_ADDR_SEG- SEGMENT PORTION OF SOURCE MEMORY */
/*         ADDRESS FOR DMA CHANNEL 1                         */
/*      6. SOURCE_ADDR_OFFSET- OFFSET PORTION OF SOURCE      */
/*         MEMORY ADDRESS FOR DMA CHANNEL 1                  */
/*                                                           */
/*************************************************************/

INIT_DMA:
DO;
$INCLUDE (:F1:GPORTS.SRC)
INIT_DMA: PROCEDURE;

  declare (tx_length,rx_length,dest_addr_seg,
           dest_addr_offset,source_addr_seg,
           source_addr_offset) word;

/* Make receive buffer 5 bytes larger than transmit buffer */
  RX_LENGTH = TX_LENGTH + 5;

        /*DMA CH0 INITIALIZATION*/
/* dma for the receiver is started in the interrupt handler */

  outword(dma0$cntrl$reg) = dma0$cntrl$stopped;

  outword(dma0$count$reg) = rx_length;

  outword(dma0$src$h$reg) = 0;
  outword(dma0$src$l$reg) = data_a_74;

  outword(dma0$dest$h$reg) = dest_addr_seg;
  outword(dma0$dest$l$reg) = dest_addr_offset;

        /*DMA CH1 INITIALIZATION*/
  outword(dma1$count$reg) = tx_length;
  outword(dma1$dest$h$reg) = 0;
  outword(dma1$dest$l$reg) = data_a_74;
  outword(dma1$src$h$reg) = source_addr_seg;
  outword(dma1$src$l$reg) = source_addr_offset;

        /*DMA REQ. MULTIPLEXER INITITIALIZATION*/
  output(dma_sel0_off) = 00h;       /*ensure that DMA MUX pal     */
  output(dma_sel1_off) = 00h;       /*is in its default condition*/
  output(dma_sel2_off) = 00h;       /*DRQ0=RxDRQA, DRQ1=TxDRQA    */

/*START DMA TO 8274 TRANSMITTER*/
outword(dma1$cntrl$reg) = dma1$cntrl$started;

end;
```

Figure F-1. iSBC® 186/03 Programming Example (continued)

```
MAIN_ROUTINE:
DO;

DECLARE (init_186,enable_8274_interrupts,
         init_8274,init_baud,init_dma) pointer
         external;

declare data_a_74       literally  '0d8h',
        command_a_74    literally  '0dch',
        status_a_74     literally  '0dch';

/* This sample routine calls the initialization routines
   that initialize 80186 control registers, 80186 timer 0 and 1
   for 8274 baud rate generators, 80130 PIC, and the 8274 MPSC.
   The 8274 is initialized so that channel A is in DMA mode,
   SDLC, at 64K baud, x1 clock at 6mhz CPU speed. Channel B is
   asynchronous, interrupt mode, x16 clock, 1.5 stop bits at
   19.2K baud.                                                  */


DECLARE BAUD_COUNT_A LITERALLY '0024D'; /* 64K baud,x1 clock,6mhz
                                           CPU */
DECLARE BAUD_COUNT_B LITERALLY '0078D'; /*19.2 baud,x16 clock,6mhz
                                           CPU */

CALL INIT_186;              /* init 186 registers,80186 and 80130 PICs */
CALL ENABLE_8274_INTERRUPTS;
CALL INIT_8274;             /* initialize the 8274 for chA in DMA,SDLC
                               mode, and chB in interrupt async. mode */
CALL INIT_BAUD (BAUD_COUNT_A,BAUD_COUNT_B);
                            /* initialize the 80186 timers 0 and 1 */
ENABLE;
CALL INIT_DMA;              /* initialize 80186 DMA ch 0 and 1, start
                            /* DMA to 8274 chA transmitter */


/* To transmit data on channel B, output the first charactor
   chB data port. You will be interrupted for the next
   charactor(s) to be transmitted, when all of your data has
   been sent, reset TXint (DMA pending indication in WR0 on the
   last transmitter buffer empty interrupt. You will be interrupted
   for receive charactors when the receive buffer becomes full
   since channel B has been set up in the interrupt mode.       */

/* To transmit an SDLC packet on channel A, write a packet address
   byte to DATA_A_74 port, poll transmit buffer empty in RR0(bit2)
   until empty, and then reset transmit UNDERRUN/EOM through WR0.
   Since the 80186 DMA controller has been set up with the packet
   length and transmit buffer start address, the transmit buffer
   will be "DMA'd" out on channel A. A transmit UNDERRUN/EOM
   interrupt will occur after the full packet has been transmitted.
   The following is a sample routine to accomplish the above. */


OUTPUT (DATA_A_74) = 55H;         /* PACKET ADDRESS BYTE */
   DO WHILE (INPUT(STATUS_A_74) AND 04H) = 04H; /* wait for transmit
                                   /* to start,Txbuffer is full */
      OUTPUT (COMMAND_A_74) = 0C0H;     /* RESET Tx UNDERRUN */
   END;

/* To receive an SDLC packet from channel A, an interrupt will
   be generated by the 8274 on the first receive charactor. The
   interrupt routine will start DMA channel 0 to receive the
   charactors from the 8274. The RxDRQA DMA request signal will go
   active for each receive charactor.                           */

END MAIN_ROUTINE;
$eject
```

Figure F-1. iSBC® 186/03 Programming Example (continued)

***

APPENDIX G. MEMORY DECODE PAL EQUATIONS

G.1. INTRODUCTION

This appendix provides the equations used to program the Memory Decode PAL (U59) to its factory default configuration. Table G-1 provides the factory default PAL code listing. Using the principles shown in the listing, and a new 20L8A PAL, the code can be modified for alternative addressing schemes. This format is compatible with the following brand and model of PAL programming equipment: Data I/O Palasm Adapter and Logic Pak.

Table G-1. Memory Decode PAL (U59) Equations

```
PAL20L8
P0003
SBC 186/03 MEMORY DECODE PAL
INTEL OREGON
AB19 AB18 AB17 AB16 AB15 AB14 AB13 AB12 NLS2 PSIZE1 PSIZE0 GND
RSIZE2 RSIZE1 /8SITES /INTIOACC /4SITES /341ACC /BW0ACC /80130CS
/BW23ACC /BW2ACC RSIZE0 VCC
IF (VCC) BW2ACC = NLS2 * PSIZE0 * PSIZE1 * AB19 * AB18 * AB17 *
                      AB16 * AB15 * /AB14 +
                  NLS2 * /PSIZE0 * PSIZE1 * AB19 * AB18 * AB17 *
                      AB16 * /AB15 +
                  NLS2 * PSIZE0 * /PSIZE1 * AB19 * AB18 * AB17 *
                      /AB16 +
                  NLS2 * /PSIZE0 * /PSIZE1 * AB19 * AB18 * /AB17
IF (VCC) BW23ACC = NLS2 * PSIZE0 * PSIZE1 * AB19 * AB18 * AB17 *
                      AB16 * AB15 +
                   NLS2 * /PSIZE0 * PSIZE1 * AB19 * AB18 * AB17 *
                      AB16 +
                   NLS2 * PSIZE0 * /PSIZE1 * AB19 * AB18 * AB17 +
                   NLS2 * /PSIZE0 * /PSIZE1 * AB19 * AB18
IF (VCC) 80130CS = NLS2 * PSIZE0 * PSIZE1 * RSIZE2 * AB19 * AB18 *
                      AB17 * AB16 * /AB15 * AB14 +
                   NLS2 * /PSIZE0 * PSIZE1 * RSIZE2 * AB19 * AB18 *
                      AB17 * /AB16 * AB15 * AB14 +
                   NLS2 * PSIZE0 * /PSIZE1 * RSIZE2 * AB19 * AB18 *
                      /AB17 * AB16 * AB15 * AB14 +
                   NLS2 * /PSIZE0 * /PSIZE1 * RSIZE2 * AB19 * /AB18 *
                      AB17 * AB16 * AB15 * AB14 +
                   NLS2 * PSIZE0 * PSIZE1 * /RSIZE2 * RSIZE1 *
                      RSIZE0 * AB19 * AB18 * AB17 * /AB16 *
                      AB15 * AB14 +
                   NLS2 * /PSIZE0 * PSIZE1 * /RSIZE2 * RSIZE1 *
                      /RSIZE0 * AB19 * AB18 * /AB17 * AB16 *
                      AB15 * AB14 +
                   NLS2 * PSIZE0 * /PSIZE1 * /RSIZE2 * /RSIZE1 *
                      RSIZE0 * AB19 * /AB18 * AB17 * AB16 *
                      AB15 * AB14
IF (VCC) BW0ACC = NLS2 * RSIZE0 * RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB17 * /AB16 * /AB15 * /AB14 * /AB12 +
                  NLS2 * /RSIZE0 * RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB17 * /AB16 * /AB14 +
                  NLS2 * RSIZE0 * /RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB16 +
                  NLS2 * RSIZE0 * RSIZE1 * /RSIZE2 * AB19 * AB18 *
                      AB17 * AB16 * /AB15 * /AB14 +
                  NLS2 * /RSIZE0 * RSIZE1 * /RSIZE2 * AB19 * AB18 *
                      AB17 * /AB16 * /AB15 +
                  NLS2 * RSIZE0 * /RSIZE1 * /RSIZE2 * AB19 * AB18 *
                      /AB17 * /AB16
IF (VCC) 341ACC = NLS2 * RSIZE0 * RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB17 * /AB16 * /AB15 * /AB14 * AB13 +
                  NLS2 * /RSIZE0 * RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB17 * /AB16 * AB15 +
                  NLS2 * RSIZE0 * /RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      AB17
IF (VCC) 4SITES = NLS2 * RSIZE0 * RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB17 * /AB16 * /AB15 * /AB14 * /AB13 +
                  NLS2 * /RSIZE0 * RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB17 * /AB16 * /AB15 +
                  NLS2 * RSIZE0 * /RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB17 +
                  NLS2 * RSIZE0 * RSIZE1 * /RSIZE2 * AB19 * AB18 *
                      AB17 * AB16 * /AB15 +
                  NLS2 * /RSIZE0 * RSIZE1 * /RSIZE2 * AB19 * AB18 *
                      AB17 * /AB16 +
                  NLS2 * RSIZE0 * /RSIZE1 * /RSIZE2 * AB19 * AB18 *
                      /AB17
IF (VCC) INTIOACC = /NLS2 * AB15 * AB14 * AB13 * AB12
IF (VCC) 8SITES = NLS2 * RSIZE0 * RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                     /AB17 * /AB16 * /AB15 * /AB14 +
                  NLS2 * /RSIZE0 * RSIZE1 * RSIZE2 * /AB19 * /AB18 *
                      /AB17 * /AB16 +
                  NLS2 * RSIZE0 * /RSIZE1 * RSIZE2 * /AB19 * /AB18
```

***

INDEX

***

REQUEST FOR READER'S COMMENTS

Intel's Technical Publications Departments attempt to provide publications that meet the needs of product users. This form lets you participate directly in the publication process. Your comments w us correct and improve our publications. Please take a few minutes to respond.

Please restrict your comments to the usability, accuracy, readability, organization, and completer this publication. If you have any comments on the product that this publication describes, please your Intel representative. If you wish to order publications, contact the Intel Literature Departme page ii of this manual).

1. Please describe any errors you found in this publication (include page number).

2. Does the publication cover the information you expected or required? Please make suggestic improvement.

3. Is this the right type of publication for your needs? Is it at the right level? What other typ publications are needed?

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

5. Please rate this publication on a scale of 1 to 5 (5 being the best rating). _____

NAME _____ DATE _____

TITLE _____

COMPANY NAME/DEPARTMENT _____

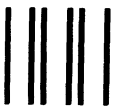
ADDRESS _____

CITY _____ STATE _____ ZIP CODE _____
(COUNTRY)

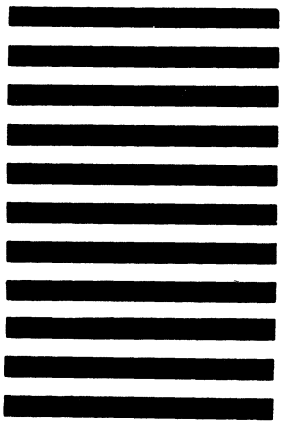
Please check here if you require a written reply. ☐

LIKE YOUR COMMENTS . . .

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



NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 79 BEAVERTON, OR

POSTAGE WILL BE PAID BY ADDRESSEE

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

IO Technical Publications

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

Printed in U.S.A.

SINGLE BOARD COMPUTER

0018/6K/0984/WCP