iSBC® 86/05
SINGLE BOARD COMPUTER
HARDWARE REFERENCE MANUAL

Order Number: 143153-002

Copyright © 1981, 1983 Intel Corporation
Intel Corporation, 3065 Bowers Avenue, Santa Clara, California 95051

REV.	REVISION HISTORY	PRINT DATE
-001	Original Issue	2/81
-002	Corrects technical errors	2/83

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

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

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

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

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

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

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

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:

AEDIT	iLBX	iOSP	MULTIBUS
BITBUS	im	iPDS	MULTICHANNEL
BXP	iMMX	iRMX	MULTIMODULE
COMMputer	Insite	iSBC	Plug-A-Bubble
CREDIT	Intel	iSBX	PROMPT
i	IntelBOS	iSDM	Promware
iATC	Intelevision	iSXM	Ripplemode
I^2ICE	inteligent Identifier	Library Manager	RMX/80
ICE	inteligent Programming	MCS	RUPI
iCS	Inteilec	Megachassis	System 2000
iDBP	Intellink	MICROMAINFRAME	UPI
iDIS			

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.

Printed in USA/OM-035/5K/0583/AP

Copyright © 1983, Intel Corporation

PREFACE

This manual pertains to both versions of the iSBC 86/05 Single Board Computer. It was written specifically to reflect the newer version (145895) of the iSBC 86/05 Single Board Computer. If you have the old version (143240) refer to Appendix C for the differences between these versions. The manual provides general information, installation, and setup instructions, programming guidelines for the on-board, programmable devices, and service information for the iSBC 86/05 Single Board Computer. Related information is provided in the following publications:

- The 8086 Family User's Manual, Order Number: 9800722.
- iSBC® Applications Manual, Order Number: 142687.
- Intel Multibus® Specification, Order Number: 9800683.
- Intel Multibus® Interfacing, Application Note AP-28A.
- MCS-86 Assembly Language Programming Manual, Order Number: 9800640.
- PL/M 86 Programming Manual, Order Number: 9800466.
- Intel iSBX™ Bus Specification, Order Number: 142686.
- Designing iSBX™ Multimodule™ Boards, Application Note AP-96.
- Using the iRMX™ 86 Operating System, Application Note AP-86.
- iSBC® 337 Numeric Data Processor Hardware Reference Manual, Order Number: 142887.
- The 8086 Primer, by Stephen P. Morse. Hayden Book Company, Inc., Rochelle Park, N.J., 1980. ISBN: 08104-5165-4.

CONTENTS

CONTENTS (continued)

CONTENTS (continued)

CONTENTS (continued)

TABLES

TABLES (continued)

FIGURES

***

CHAPTER 1. GENERAL INFORMATION

1.1 INTRODUCTION

The iSBC 86/05 Single Board Computer is a 16-bit computer system on a single printed circuit board and is compatible with the Intel Multibus and iSBX Multimodule architectures (Figure 1-1). It provides a low-cost solution to those users whose requirements do not exceed 8K bytes of on-board static random access memory (RAM) (16K bytes with iSBC 302 expansion module), but want the high-performance capability of the 8 MHz, 16-bit 8086-2 Microprocessor. The iSBC 86/05 board also includes 24 programmable parallel I/O lines, one serial I/O port, three programmable interval timers, and a programmable interrupt controller. Sockets are provided for a maximum of 64K bytes of read only memory (ROM).

On-board memory expansion can be readily accomplished using plug-in memory expansion boards. On-board RAM size may be doubled, to 16K bytes using the optional iSBC 302 RAM Expansion Module. On-board ROM size may be increased to a maximum of 128K bytes, using the iSBC 341 ROM Expansion Module.

Additional on-board I/O capabilities are provided via two iSBX Multimodule connectors. These connectors allow any of the optional iSBX Multimodule boards to be used on the iSBC 86/05 board.

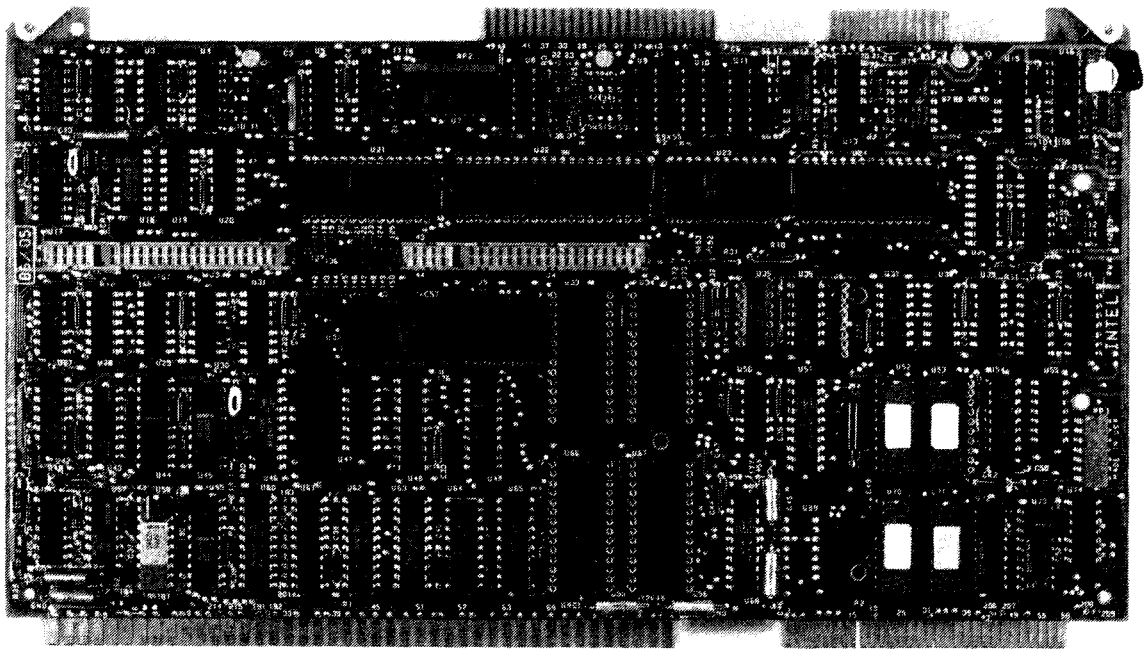


Figure 1-1. iSBC® 86/05 Single Board Computer

1.2 DESCRIPTION

The iSBC 86/05 board is controlled by an Intel 8086-2 microprocessor operating at 8 MHz. Processor support is provided by an Intel 8284A Clock Generator/Driver and an Intel 8288 Bus Controller. The board can be jumpered to operate at 5 MHz, if necessary.

Up to 1 Megabyte of total system memory may be directly addressed by the iSBC 86/05 board. Of this amount, a maximum of 144K bytes may reside on-board (16K RAM + 128K ROM). The 8086 microprocessor can access either 8 or 16 bits of memory at a time, allowing maximum system compatibility.

The 8K bytes of on-board static RAM is implemented with four Intel 2168 (4K X 4 bit) static RAM devices. These high-speed devices require only a +5 volt supply, and are therefore well suited for battery backup applications. The iSBC 302 RAM Expansion Module increases the on-board RAM array to 16K bytes, using an additional four 2168 devices.

The board will accept a wide variety of ROM, PROM, and EPROM devices. Either 24 or 28-pin devices may be used. Four on-board sockets are provided, with expansion provided by the optional iSBC 341 ROM Expansion Module. Refer to Chapter 2 for complete information.

The on-board 8253-5 Programmable Interval Timer provides three independently controlled counters which may be configured to perform a variety of applications including frequency output, rate generator, interval timer and real-time interrupts. One of the counters is used as the CPU interval timer and is connected to IR2 on the interrupt controller.

Serial I/O operation is handled by an Intel 8251A Programmable Communications Interface device. The board supports the RS 232C standard. Baud rates are software programmable using one of the counter outputs from the on-board interval timer in conjunction with the communications interface device.

The iSBC 86/05 board utilizes one Intel 8255A-5 Programmable Peripheral Interface device to control the three, 8-bit, parallel I/O ports. All 24 lines may be configured for a variety of dedicated or general purpose applications. One port is equipped with an Intel 8287 Bus Transceiver. The other two ports are equipped with sockets for line drivers or terminators.

All interrupts, except the 8086-2 non-maskable interrupt (NMI), are handled by the on-board Intel 8259A Programmable Interrupt Controller device. System interrupts can be connected to the interrupt controller via the Multibus lines, and additional interrupts may originate from one or two iSBX Multimodule boards. An on-board interrupt jumper matrix allows interrupt configuration flexibility and provides priority selection.

Two iSBX bus connectors (J3 & J4) are provided on the iSBC 86/05 board. These connectors are designed to expand the board's I/O functions, using special purpose optional iSBX Multimodule boards, such as the iSBX 350 Parallel I/O Multimodule Board. The Multimodule boards reside directly on the iSBC 86/05 board (Figure 1-2). One or two Multimodules may be added, as required by your application.

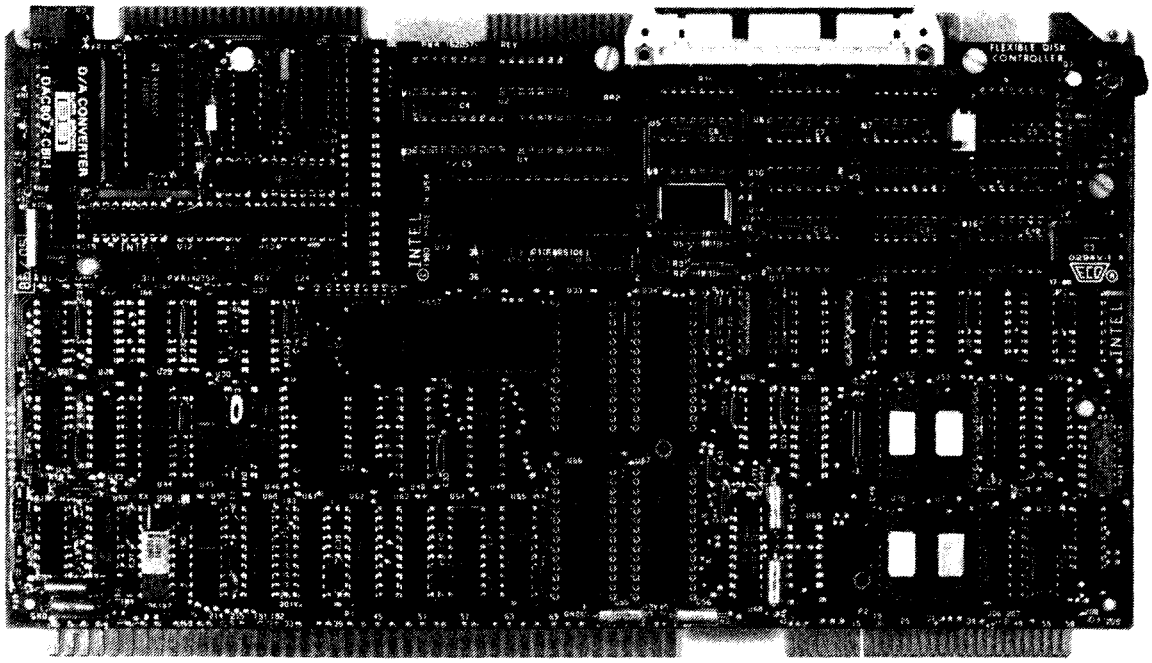


Figure 1-2. iSBC® 86/05 Board With Optional Multimodule Boards

Off-board system access is provided by the Multibus connector (P1) and an auxiliary connector (P2). Off-board peripheral operations are handled through 24 parallel I/O lines (connector J1), a serial communications channel (connector J2), and two iSBX Multimodule connectors.

The iSBC 86/05 board is designed to operate as a full master in any Intel Multibus compatible chassis or backplane. The board may also reside in your own custom chassis, using Multibus compatible connectors (refer to Chapter 2).

1.3 DOCUMENTATION SUPPLIED

Each iSBC 86/05 board is shipped with a corresponding set of schematic diagrams. These drawings should be inserted into the back of this manual for future reference. Refer to Chapter 5 for related information.

1.4 ADDITIONAL EQUIPMENT REQUIRED

The iSBC 86/05 board requires a few optional components for basic operation. Depending on your application, you may need to purchase a parallel I/O connector, a serial I/O connector, and additional RAM if more than 8K bytes are required. Any on-board ROM must also be purchased separately. Chapter 2 provides information for selecting these items.

1.5 SPECIFICATIONS

Specifications of the iSBC 86/05 board are provided in Table 1-1.

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 86/05 board's level of compliance to the 796 BUS SPECIFICATION :

D16 M20 I16 V2 EL

This notation is decoded as follows:

D16 = data path is 8 and/or 16 bits
M20 = memory address path is up to 20 bits
I16 = I/O address path is 8 or 16 bits
V2 = Non-Bus-Vectored interrupts are supported; and
EL = Level-triggered and Edge-triggered interrupts are supported

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 (i.e., reference manual). 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 86/05 board's level of compliance to the iSBX BUS SPECIFICATION:

D16/8

This notation is decoded as follows:

D16/8 = A 16-bit CPU board that can interface to an 8-bit expansion module.

Table 1-1. Board Specifications

CPU	Intel 8086-2	
Operating Rate	8 MHz (default) 5 MHz (optional)	
Single Bus Cycle	125 nanoseconds	
Minimum Processor Bus Cycle (four single cycles)	500 nanoseconds	
MULTIBUS CLOCK	9.830 MHz (BCLK/ & CCLK/)	
PCI Clock Input	1.229 MHz	
PIT Input 0 & 2	1.229 MHz	
PIT Input 1	153.6 KHz	
RAM ACCESS TIME	85 nsecs max, Address To Data	
ROM/PROM/EPROM ACCESS TIME		
8 MHz	285 - 660 nsecs (0 - 3 Wait states)	
5 MHz	520 - 1120 nsecs (0 - 3 Wait states)	
MEMORY CAPACITY	1M Byte	
Maximum On-Board ROM/EPROM	128K Bytes	
Maximum On-Board RAM	16K Bytes	
Remaining Off-Board Expansion	866K Bytes	
MEMORY ADDRESSING	All notation in hexidecimal	
On-Board RAM	0 - 1FFF	
With iSBC 302 Expansion	0 - 3FFF	
On-Board ROM	FE000 - FFFFF	using 2716 devices
	FC000 - FFFFF	using 2732 devices
	F8000 - FFFFF	using 2764 devices
	F0000 - FFFFF	using 27128 devices
With iSBC 341 Expansion	FC000 - FFFFF	using 2716 devices
	F8000 - FFFFF	using 2732 devices
	F0000 - FFFFF	using 2764 devices
	E0000 - FFFFF	using 27128 devices

Table 1-1. Board Specifications (continued)

ON BOARD I/O ADDRESSING		
iSBX Connector J4 (8-bit board)	80 - 9E	Even Bytes Only
iSBX Connector J4 (16-bit board)	80 - 8F	
iSBX Connector J3 (8-bit board)	A0 - BE	Even Bytes Only
iSBX Connector J3 (16-bit board)	A0 - AF	
Interrupt Controller	C0 or C4	ICW1, OCW2, OCW3, Status, & Poll
	C2 or C6	ICW2, ICW3, ICW4, & Masks
Parallel Interface	C8	PPI Port A
	CA	PPI Port B
	CC	PPI Port C
	CE	PPI Control
Interval Timer		
Counter 0	D0	
Counter 1	D2	
Counter 2	D4	
Counter Control	D6	
Serial Interface		
Data	D8 or DC	
Mode or Status	DA or DE	
INTERFACES		
Multibus	All signals TTL compatible	
Parallel I/O	All signals TTL compatible	
Interrupt Requests	All signals TTL compatible	
Interval Timer	All signals TTL compatible	
iSBX Bus	All signals TTL compatible	
Serial I/O	RS 232C compatible, data set	

Table 1-1. Board Specifications (continued)

ELECTRICAL REQUIREMENTS

CONFIGURATION	+5Vdc	+12Vdc*	-12Vdc*
Standard Board, no ROM/EPROM	4.7A	25mA	23mA
Add for four 2716 devices	250mA	--	--
Add for four 2732 devices	320mA	--	--
Add for four 2764 devices	280mA	--	--
Add for four 27128 devices	390mA		
Add for iSBC 302 Option	720mA	--	--
Add for iSBC 341 Option	600mA	--	--
Add for two iSBX Multimodules	6.00A	2.0A	2.0A
Add for iSBC 337 Option	475mA	--	--
STANDBY CURRENT REQUIREMENTS			
Four 2716	25mA	--	--
Four 2732	35mA	--	--
Four 2764	40mA	--	--
iSBC 302 Option	120mA	--	--
iSBC 341 Option	220mA	--	--
BATTERY BACKUP REQUIREMENTS	0.8A	--	--
MAXIMUM OPERATING REQUIREMENTS (with all options)	12.9A	2.025A	2.023A

* +12Vdc & -12Vdc are required for RS232 applications only.

PHYSICAL CHARACTERISTICS

Width	12.00 in. (30.48 cm)
Length	6.75 in. (17.15 cm)
Thickness	0.50 in. (1.27 cm)
Weight	14.0 oz. (388 grams)

ENVIRONMENTAL CHARACTERISTICS

Maximum Power Requirements	119 Watts
Maximum Heat Dissipation	1689 gcal/minute (6.84 Btu/minute)
Operating Temperature Range	0°C - 50°C
Operating Humidity Range	85% max non-condensing

***

CHAPTER 2. PREPARATION FOR USE

2.1 INTRODUCTION

This chapter provides specific information enabling you to install the iSBC 86/05 Single Board Computer into your system. The board's default or factory configuration for RAM addressing, ROM/PROM size, and other variables are described, followed by procedures for altering the default configuration. Using the information in this chapter, you can configure this board to operate in a variety of applications. To completely familiarize yourself with the flexibility of the iSBC 86/05 board, we recommend reading the entire chapter before installation and use.

2.2 UNPACKING AND INSPECTION

Inspect the shipping carton immediately upon receipt for evidence of mishandling during transit. If the shipping carton is severely damaged or water-stained, 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.

For repairs to a product damaged in shipment contact the Intel Product Service Marketing Administration to obtain a return authorization number and further instructions (see Section 4-2). A purchase order will be required to complete the repair. A copy of the purchase order should be submitted to the carrier with your claim.

2.3 INSTALLATION CONSIDERATIONS

There are several general requirements which should be considered prior to board installation and use. These requirements are discussed in the following sections.

2.3.1 MINIMAL OPERATING REQUIREMENTS

The features of the iSBC 86/05 board are described in Chapter 1. In order to operate the board you may need additional equipment. For most applications this will typically be the following:

a. CPU software, residing in on-board ROM/PROM (Section 2.6.1 and 2.6.3).
b. I/O connectors and cables (Sections 2.7; 2.7.1; 2.7.2).
c. Additional on-board RAM, if more than 8K bytes are required (Section 2.6.4).
d. Line drivers or terminators for parallel I/O lines (Section 2.6.2), if parallel I/O is required.

Instructions for installing these items are provided in the sections listed above. Before installing these items, the appropriate jumper connections must be made. Refer to 2.5 for jumper configurations. Physical location of jumper posts on the board are shown in Figure 4-2. Jumper connections are also shown schematically in Figure 4-3.

2.3.1.1 Power Requirements

Three voltages are required for operating the iSBC 86/05 board in most configurations: +5 Vdc, +12 Vdc, and -12 Vdc. All must be within ±5%. However, some configurations do not require all voltages. Power requirements for the various board configurations are listed in Table 1-1. The table includes power required by any optional iSBX Multimodule boards which may be installed.

2.3.1.2 Cooling Requirements

Operating temperature range for the iSBC 86/05 board is 0°C to 50°C. If the board is installed into an Intel system chassis, adequate cooling is provided by the fans supplied. However, if the board is used in another chassis, you must ensure that adequate cooling is provided.

2.4 SYSTEM CONSIDERATIONS

Before installing the iSBC 86/05 board, you need to consider the system environment in which the board is to operate. These considerations include but are not limited to the following:

Memory allocation - Refer to 2.4.1
Timing - Refer to 2.4.2
Arbitration - Refer to 2.4.3
Priority - Refer to 2.4.4
Interrupt structure - Refer to 2.4.5
Interfaces - Refer to 2.4.6
Power Fail/Memory Protect - Refer to 2.4.7

2.4.1 MEMORY ALLOCATION CONSIDERATIONS

The iSBC 86/05 board is shipped from the factory with 8K bytes of RAM. It provides four sockets in which PROM's programmed with your CPU firmware can be installed. The type and the amount of PROM's installed determine the memory size and therefore the addressing range. You need to consider the size of memory needed for your application. Both RAM and PROM expansion Multimodules are available to increase on-board memory size, if needed. After memory size is determined, you need to configure the board to the type of PROM selected and the amount of memory (both RAM and PROM) that is installed on the board.

The iSBC 86/05 board is configured at the factory to recognize two separate on-board 128K byte pages as valid memory address ranges. These pages are used to map memory. One page (the top of memory) is reserved for PROM and the other (bottom of memory) is reserved for RAM. The memory portion reserved is jumper selectable. Refer to 2.5.1 for memory jumper information.

2.4.2 TIMING CONSIDERATIONS

The timing used by the iSBC 86/05 board may be changed. The following paragraphs present information to be considered before making the changes to the timing.

2.4.2.1 Processor Clock

The 8086 microprocessor on the iSBC 86/05 board is configured at the factory to operate at 8 MHz. However, the microprocessor can operate at 5 MHz by removing jumper E181 to E182 to reconfigure the 8284A. If the iSBC 337 Numeric Data Processor is required for your application, the microprocessor has to run at 5 MHz. If you elect to operate the microprocessor at 5 MHz instead of the default 8 MHz, you need to (besides reconfiguring the 8284A) reconfigure the Ready circuit on the board to eliminate at least one wait state for any I/O accesses and two wait states for any acknowledge signal. Refer to Table 2-2 and section 2.5.7.2 for additional information.

2.4.2.2 Fail-Safe Timer Selection

If a non-existent off-board memory address or I/O address is addressed by the 8086 CPU, the iSBC 86/05 board will execute wait states indefinitely, causing the board to cease processing. An on-board fail-safe timer can be jumper selected to prevent this. The fail-safe timer times-out after a delay of approximately 10 milliseconds, giving the CPU a "false" READY signal so that it may resume processing. To enable the failsafe timer, install jumper connection 14 to 15. Notice that the failsafe timer applies only to Multibus (off-board) requests. If you want the fail-safe timer to interrupt processing connect jumper E129 to an interrupt input.

2.4.2.3 Clock Generator

The iSBC 86/05 generates two signals BCLK/ and CCLK/ at a frequency of 9.83 MHz. BCLK/ can be stopped or single-stepped and is used to synchronize bus contention logic. CCLK/ is constant and may be used as a clock source by other boards in the system. Both of these clock signals should be generated by only one board within the system environment. CCLK/ signal is inhibited by removing jumper 191 to 192; BCLK/ signal is inhibited by removing jumper 179 to 180.

2.4.2.4 Interval Timers

The iSBC 86/05 board contains a Programmable Interval Timer that includes three independently controlled counters. These counters are used for on-board I/O and CPU interrupt timing. The input clock frequency to each counter is selectable. The output from counters zero and one are routed directly to the interrupt matrix. These outputs may then be jumpered to the desired on-board interrupt level, or routed off-board via the Multibus system bus lines by jumper connection to one of the outbound posts (194 through 201). The output from counter two is used as the transmit and receive clocks of the 8251A Programmable Communications Interface device.

2.4.3 INTERFACE ARBITRATION

The iSBC 86/05 board contains a 8289 Bus Arbiter that operates in conjunction with the 8288 Bus Controller to interface the 8086 processor to the Multibus System Bus interface. The 8289 Bus Arbiter can operate in several modes, depending on how it is wired and the status of the Common Bus Request (CBRQ/) signal.

2.4.3.1 Common Bus Request

Common Bus Request (CBRQ/), a bidirectional Multibus System Bus interface signal, allows a bus master to retain control of the system bus without contending for it each transfer cycle, as long as no other master is requesting control of the bus. A bus master requesting control of the bus, but not currently controlling it, asserts CBRQ/. This causes the controlling bus master to relinquish control of the bus when the proper surrender conditions exist. (See Table 2-18 for surrender conditions.)

The CBRQ/ pins of all the bus master devices that support CBRQ/ are connected together via the backplane. When a bus master needs a bus resource, it informs the other bus masters that it is requesting the bus by activating CBRQ/ and either BREQ/ or BPRO/. When the controlling master releases the bus, the bus exchange operates as described in Table 2-18.

CBRQ/ minimizes bus access time by allowing a bus master to retain control without contending for it each transfer cycle, as long as no other master is requesting control of the bus.

There are two priority resolution schemes used on the system bus: serial and parallel (see section 2.4.4). When common bus request is used, the operation is identical in either parallel and serial priority resolution schemes.

2.4.3.2 Any Request

The iSBC 86/05 board has a jumper option (ANYRQST) that controls, in conjunction with BPRN/, BPRO/ and also CBRQ/, the condition under which the Multibus System bus interface will be surrendered. The following paragraphs describe this option.

When ANYRQST is jumpered to a low level (jumper 189 to 190), the bus arbiter currently controlling the Multibus System bus interface retains control unless one of the following conditions exist.

1. A higher priority bus master requests the interface (as indicated by the BPRN/ signal going high and BREQ/ going low).

2. The next transfer cycle of the iSBC 86/05 board does not require the use of the interface, and CBRQ/ is low.

When ANYRQST is jumpered to a high level (jumper 188 to 189) and CBRQ/ is jumpered to a low level, it permits the Multibus System bus interface to be surrendered to a higher priority bus master or to a lower priority bus master as though it were a bus master of higher priority. When this option is used, the bus master that is in control surrenders the interface its current cycle is completed.

2.4.4 PRIORITY RESOLUTION CONSIDERATIONS

Your iSBC 86/05 board has been designed as a "full master" board. This means the board is equipped with bus arbitration logic and can acquire and relinquish control of the common system Multibus lines. In order for this system to be effective, a board priority scheme should be established in your system.

If your iSBC 86/05 board is the only master in the system and you have serial priority resolution, you must place it in a cardcage slot that has BPRN/ grounded. (This connection should be verified if the factory configuration has been modified.)

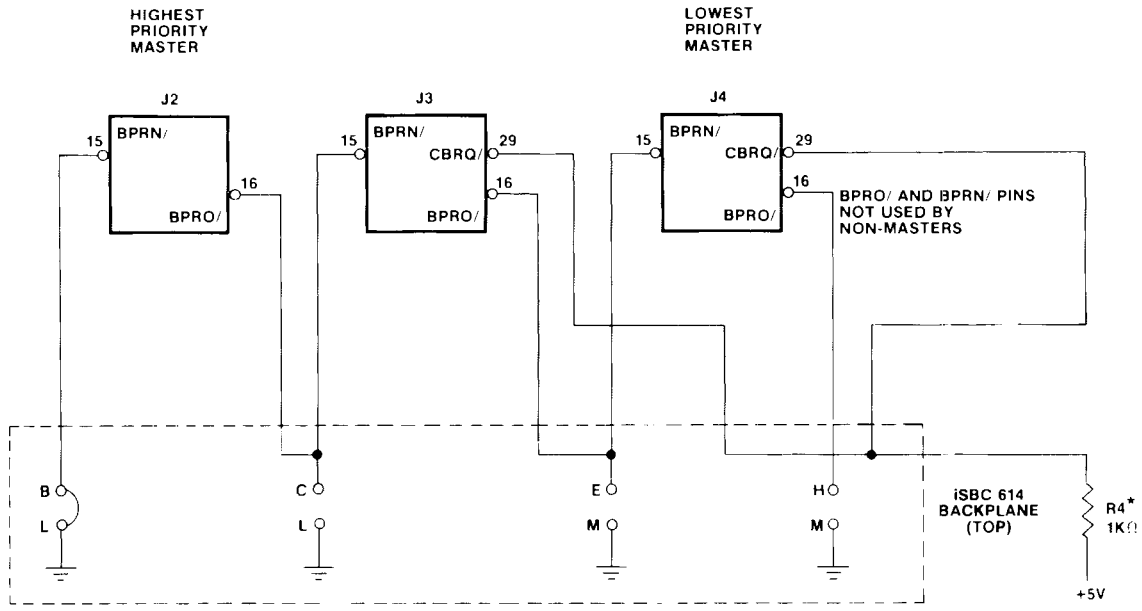
If your system includes more than one master board, you must establish a board priority scheme. In this configuration, you must place the highest priority master board in the slot that has BPRN/ grounded. The remaining slots can be used first for any lower priority master boards and then the remaining slots for any expansion boards. There are two methods of priority resolution available: serial and parallel. These are described in sections 2.4.4.1 and 2.4.4.2, respectively.

Another important consideration in setting up a multimaster system is the Multibus clock signal source. You must ensure only one of the master boards is supplying the BCLK/ & CCLK/ signals to the Multibus lines. All master boards that generate these signals have provisions for disabling these outputs (i.e., preventing the signals from going off-board). The iSBC 86/05 board BCLK/ & CCLK/ signals can be disabled by removing two jumper connections. Refer to Section 2.5.7.1 for this information.

2.4.4.1 Serial Priority Resolution

In the serial priority schemes, board priorities are determined by the interconnections of the BPRO/ and BPRN/ signals of the master boards. Intel system chassis are set up so that priority levels can be determined by board placement, with no user wiring changes required. Alternatively, a user may wire the backplane to modify the standard board position/-priority relationship. If your iSBC 86/05 board resides in the top slot (the highest priority), the board installed in the next lower slot is assigned the next lower priority. The board installed in the lowest slot is assigned the lowest priority in this scheme. Due to the propagation delay of the BPRO/ signal path, this scheme is limited to a maximum of three master boards. In the configuration shown in Figure 2-1, the master board installed in slot J2 has the highest priority and is able to acquire control of the Multibus lines anytime the bus is not busy. This is because the BPRN/ input is always true (tied to ground via jumper connection B to L on the backplane).

If the master board in the top slot desires control of the Multibus lines, it drives its BPRO/ output high (false) and inhibits the BPRN/ inputs to the remaining lower priority master boards. The master board then takes control of the Multibus lines when the current bus cycle is completed. When finished using the Multibus lines, the master board installed in slot J2 pulls its BPRO/ output low (true) and gives the J3 master board the opportunity to acquire Multibus line control. If the J3 master board does not want the Multibus lines, it pulls its BPRO/ output low (true) and gives the J4 master board the opportunity to assume control of the Multibus lines.

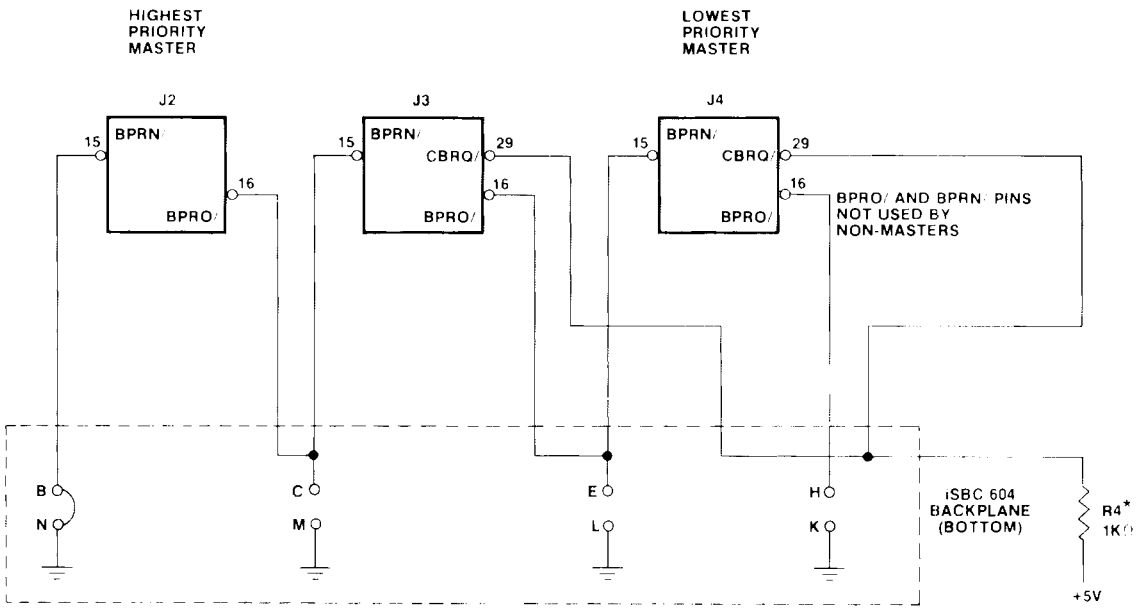


*Pull-up resistor is supplied by the customer.

NOTE: All non CBRQ/ devices must have higher priority. If a non CBRQ/ device is placed at a lower priority, it will not be able to gain control of the Multibus interface.

Top Backplane In Stacked System X-467

Figure 2-1. Serial Priority Resolution Schemes



*Pull-up resistor is supplied by the customer.

NOTE: All non CBRQ/ devices must have higher priority. If a non CBRQ/ device is placed at a lower priority, it will not be able to gain control of the Multibus interface.

Single Backplane In Non-Stacked System X-468

Figure 2-2. Serial Priority Resolution Schemes

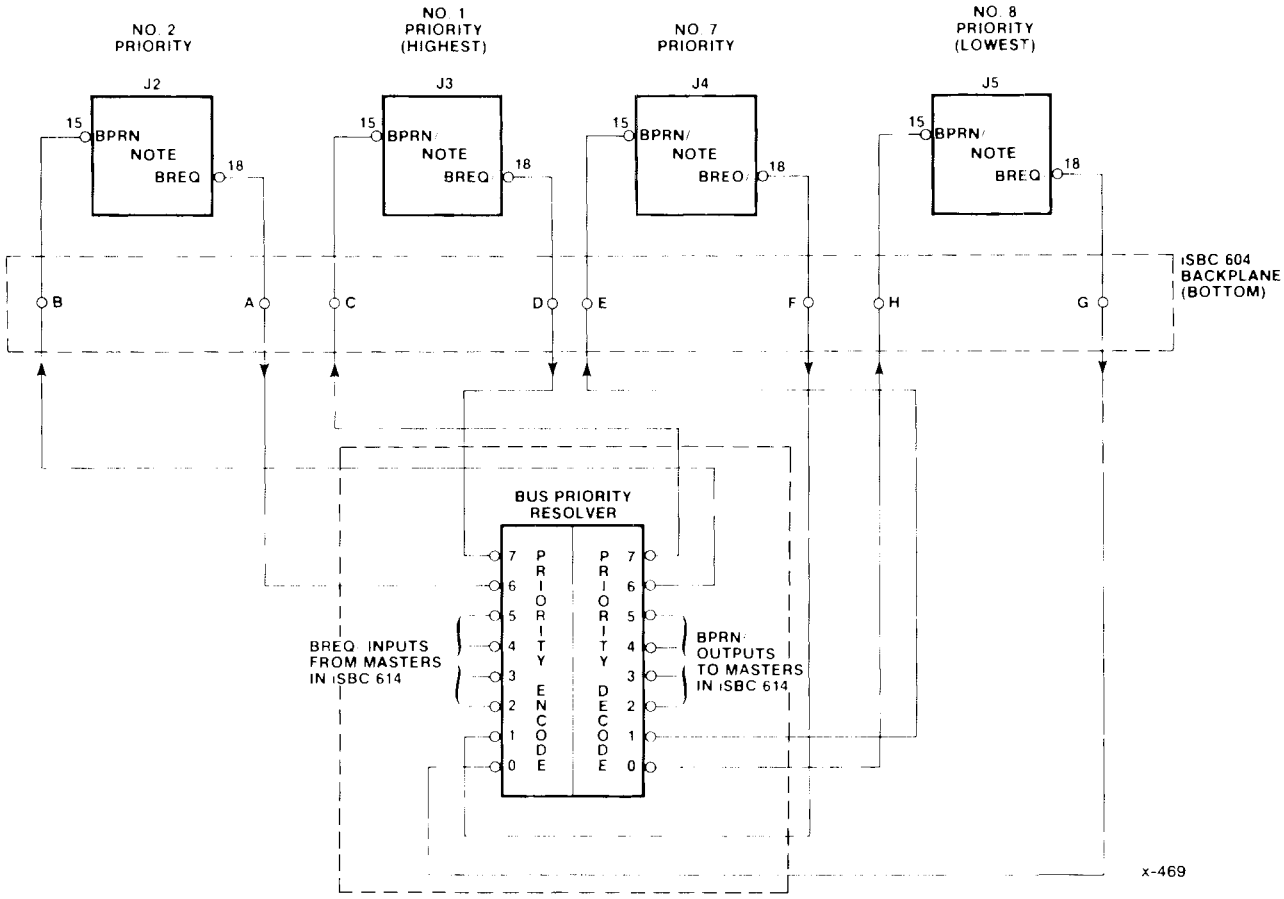
2.4.4.2 Parallel Priority Resolution

A parallel priority resolution scheme allows up to 16 bus masters in a single system to acquire and control the Multibus lines. Figure 2-3 illustrates one method of implementing such a scheme for resolving bus contention in a system using eight bus masters. Notice that the two highest and two lowest priority bus masters are shown installed in the master chassis. The other masters in this example are installed in the expansion chassis.

In the scheme shown in Figure 2-3, the input connections to the priority encoder determine the bus priority, with input 7 having the highest priority and input 0 having the lowest priority. In Figure 2-3, the master board in J3 has the highest priority and the master board in J5 has the lowest priority.

NOTE

In a parallel priority resolution scheme, the BPRO/output must be disabled on all master boards if the BPRO/ signal is bussed on the backplane. Refer to the appropriate hardware reference manual for instructions.



*Note: The user must implement logic, wire to mother board, and provide mounting.

Figure 2-3. Parallel Priority Resolution Scheme

2.4.5 INTERRUPT CONSIDERATIONS

The iSBC 86/05 board supports both bus vectored and non-bus vectored interrupts by using the 8259A Programmable Interrupt Controller (PIC) device. In both types of operation, the on-board PIC must be the master PIC in the system environment. The PIC provides eight interrupt levels. The sources of these eight interrupts can be interfaced to the PIC through the on-board interrupt matrix. The iSBC 86/05 board provides jumper posts for 12 on-board interrupt sources and 11 off-board sources. In addition, the user can utilize the processor's NMI input for a priority interrupt request.

2.4.5.1 Non-Bus Vectored Interrupts

A Non-Bus Vectored (NBV) interrupt is one in which the interrupt vector is generated by the on-board PIC and passed to the on-board processor. An NBV interrupt does not send the interrupt vector across the Multibus interface. An NBV interrupt is generated by asserting one of the on-board 8259 interrupt inputs.

2.4.5.2 Bus Vectored Interrupts

The Bus Vectored (BV) interrupt is similar to the non-bus vectored interrupt scheme except the interrupt vector comes to the processor from an off-board slave PIC rather than from the on-board (master) PIC. The interrupt vector is sent across the Multibus interface to identify the interrupting device. Bus vectored interrupts require additional time compared to non-bus vectored interrupts. If the iSBC 86/05 is operating in the bus vectored mode, it must gain control of the Multibus System bus interface for each interrupt; this causes the on-board processor to execute additional wait states and may cause system contention for use of the bus. Therefore, you must configure the Ready circuit to insert additional wait states into the processor's instruction execution timing when bus vectored interrupts are used. Refer to section 2.5.7.2.

In the factory default configuration, the following four interrupt matrix jumpers are installed:

a. 123 - 124 Timer 0 output to IR2 on PIC
b. 145 - 146 Multibus interrupt INT5/ to IR5 on PIC
c. 120 - 121 Disable NMI input
d. 125 - 132 RxRDY Interrupt to IR0 on PIC

Table 2-13 provides a complete list of possible interrupt jumper configurations on the iSBC 86/05 board. Refer to Section 3.8 for 8259A programming information.

2.4.6 INTERFACE CONSIDERATIONS

The following paragraphs present information to be considered when interfacing the iSBC 86/05 board.

2.4.6.1 Multibus® Interface

The iSBC 86/05 board is designed to interface with the Multibus System bus interface. The Multibus connector P1 and auxiliary connector P2 interface the iSBC 86/05 board signals and power lines to the other boards in your system and the power supply. Where applicable, these signals conform to the Intel Multibus Interface standard. All of this interface information is available in the Multibus specification.

The dual port RAM lock signal originates from the iSBC 86/05 board, and is used to prevent access to any system level dual port RAM by another master. This signal prevents any other system board from accessing its own dual-port RAM available to the Multibus system bus until the master board that has asserted LOCK completes a read-modify-write, or test-and-set operation. The lock signal (LOCK) is generated when a locked Multibus access (either by OVERRIDE or PROCESSOR LOCK) is requested. It exits the board on Multibus pin P1 - 25.

2.4.6.2 Serial I/O Considerations

The 8251A PCI device is used to implement the serial interface for the iSBC 86/05 board. The PCI is programmable and configurable by jumper connections to offer a wide range of serial interface applications. You need to provide a connector that mates with J2 of the iSBC 86/05 board. Refer to section 2.5.3 for jumper information concerned with the serial interface. Refer to section 2.7.2 for serial I/O cabling information. Refer to section 3.6 for information concerned with programming the 8251A device.

2.4.6.3 Parallel I/O Considerations

The 8255A PPI device is used to implement the parallel interface for the iSBC 86/05 board. The PPI is programmable and configurable with jumper connections to offer a wide range of parallel interface applications. You need to provide a connector that mates with J1 of the iSBC 86/05 board. Refer to section 2.5.4 for jumper information concerned with the parallel interface. Refer to section 2.7.1 for parallel I/O cabling information. Refer to section 3.7 for information concerned with the programming the 8255A device.

2.4.6.4 iSBX™ Interface Considerations

The iSBC 86/05 board provides two iSBX connectors (J3 and J4). These connectors permit the installation of one or two particular iSBX multimodule(s) designed to fulfill a specific application. The iSBC 86/05 accesses the installed iSBX Multimodule as though it were an on-board I/O address, eliminating Multibus system bus contention. The iSBX interface information is provided in section 2.5.8.

2.5 JUMPER CONFIGURATIONS

Much of the flexibility of your iSBC 86/05 board is due to the use of jumper connections which may easily be altered from their factory configurations to suit your particular application. Sections 2.5.1 through 2.5.9 describe optional jumper connections for all of the iSBC 86/05 configurations. Table 2-1 lists the jumper connections in numerical order and Table 2-2 lists the factory default configurations.

Table 2-1. iSBC® 86/05 Jumpers

Jumper Pair	Function
E1 thru E11	Wait state selection jumpers for the Ready circuit. Refer to Table 2-16.
E12 and E13	No jumper posts.
E14 to E15*	Enables fail-safe timer to time-out bus operations.
E16	No jumper post.
E17 to E18	Routes +5V to parallel I/O connector J1 pin 50.
E19 to E20*	Routes BUSY/ signal from either the iSBC PROM Expansion Multimodule connector J6 or from on-board to gate the EPROM Ready circuit.
E19 to E21	Routes RDY/ signal from the Multimodule connector J6 to gate the EPROM Ready circuit.
E22	Routes a data bit from the parallel port area to jumper post E73. Optionally, it may be routed through an RS232 driver to the serial I/O connector J2 as either SEC CTS (jumpers E73 to E74 and E76 to E75) or SEC REC SIG DATA (jumpers E73 to E74 and E76 to E78).
E23	Routes the external clock signal from the parallel port area (E62) as a clock input to the Programmable Interval Timer.
E24	Routes a bit from the parallel port area (Override) to generate the Multibus LOCK signal.
E25 to E27*	Connects PIT Gate 1 Control to a pullup resistor allowing counter one to count.
E26	Routes a control bit from the parallel port area to disable/enable the maskable NMI gate.
E28 to E29*	Connects the direction control for the 8255A PPI Port A I/O signal buffer to ground which selects the output mode when installed.
E30	Routes the PB INTR signal input from the parallel port to the interrupt jumper matrix. This is one of two posts used to connect signals from the parallel port to the interrupt matrix. Connects to post E140 of the interrupt matrix.

Table 2-1. iSBC® 86/05 Jumpers (continued)

Jumper Pair	Function
E31	Routes the BUS INTR OUT signal from the parallel port area to a driver for BUS INT OUT to the Multibus.
E32 to E33*	Connects PIT Gate 0 control to a pullup resistor to enable PIT counter 0 to count.
E34	Routes the PA INTR signal input from the parallel port to the interrupt jumper matrix. Connects to post 141 of the interrupt jumper matrix.
E35	Test signal to the 8086-2 CPU. Should be connected to ground if an iSBC 337 NDP Multimodule is not installed to prevent execution of WAIT instructions from hanging-up the CPU.
E36	Provides the PFSN/ signal input from the P2 connector at the parallel I/O port jumper matrix.
E37	No jumper post.
E38 to E42* E39 to E43* E40 to E44* E41 to E45* E46 to E50* E47 to E51* E48 to E52* E49 to E53	Routes the 8255A PPI Port C I/O signals to the Parallel I/O connector J1.
E54 to E55	Connects PPI Port C bit 3 to drive LED (DS1).
E56 to E57*	Selects 1.23 MHz clock input frequency to counter 2 of the 8253-5 PIT device.
E58 to E59*	Selects 1.23 MHz clock input frequency to counter 0 of the 8253-5 PIT device.
E60 to E61*	Selects 156.3 KHz clock input frequency to counter 1 of the 8253-5 PIT device.
E62	Connects to jumper post E23. Generally used to route the EXT CLK signal to the parallel port jumper matrix.
E63	Output from counter 0 of the 8253-5 PIT device.
E64	2.46 MHz clock output.

Table 2-1. iSBC® 86/05 Jumpers (continued)

Jumper Pair	Function
E65	Output from counter 1 of the 8253-5 PIT device.
E66	Provides Power Line Clock input from the P2 connector as optional input to 8253-5 PIT device.
E67	Output from counter 2 of the 8253-5 PIT device.
E68 to E69	Routes +5 V to serial I/O Connector J2 pin 23.
E70 to E71	Routes -12V to serial I/O connector J2 pin 19.
The following six jumper descriptions refer to an RS232 driver. This driver is part of device U14 and has its input connected to jumper post E74 and its output to post E76. This driver can be used by the customer as required.	
E72	Provides the Transmit Clock (TxC) signal adjacent to jumper post E74. May be connected to jumper post E74 as the input of the RS232 driver.
E73	Provides the STxD signal from the parallel port jumper matrix via jumper post E22. May be connected to jumper post E74 as the input to the RS232 driver.
E74	Selects STxD (jumper post E73) or TxC (jumper post E72) as the input to the RS232 driver.
E75 to E76	Routes the output from the RS232 driver to the serial I/O interface connector J2 as Sec CTS signal.
E76 to E77	Routes the output from the RS232 driver to the serial I/O interface connector J2 as the TRANS SIG ELE TIMING signal.
E76 to E78	Routes the output from the RS232 driver to the serial I/O interface connector J2 as the SEC REC SIG DATA signal.
E79 to E80*	Routes the counter 2 output from the 8253 PIT device to the Receive Clock (RxC) input of the 8251A PCI device.
E79 to E81	Routes the Secondary Transmit Clock from the serial I/O interface connector J2 to the Receive Clock (RxC) input of 8251A PCI device.
E82 to E83	Routes the Secondary Transmit Clock from the serial I/O interface connector J2 to the Transmit Clock (TxC) input of the 8251A PCI device.

Table 2-1. iSBC® 86/05 Jumpers (continued)

Jumper Pair	Function
E83 to E84*	Routes the counter 2 output from the 8253 PIT device to the Transmit Clock (TxC) input of the 8251A PCI device.
E85 to E86	Routes +12V to serial I/O connector J2 pin 22.
E87 to E88*	Routes CTS to RTS.
E89 to E90*	Connects 19.6608 MHz clock to divider circuitry.
E91 to E92*	Routes the Data Terminal Ready signal from the serial interface connector J2 to the Data Set Ready input of the 8251A PIC device.
E93 thru E103	No jumper posts.
E104 to E105	Enables additional RAM address space due to the installation of the iSBC 302 RAM Expansion Multimodule.
E105 to E106	Enables additional PROM address space due to the installation of the iSBC 341 PROM Expansion Multimodule.
E107 thru E116	Selects the proper 128K page addresses. (See Table 2-3.)
E117	Provides the SBX2 INTO interrupt signal at the interrupt jumper matrix.
E118	Provides the POWER FAIL INTERRUPT signal from the external power fail sense circuitry via connector P2 at the interrupt jumper matrix.
E119	Provides an external interrupt 0 signal from parallel port connector J1 at the interrupt jumper matrix.
E120 to E121*	Grounds NMI interrupt input to 8086-2 CPU.
E122	Provides the TIMER 1 INTR interrupt signal from PIT counter 1 at the interrupt jumper matrix.
E123 to E124*	Routes the TIMER 0 INTR interrupt signal from PIT counter 0 to the IR2 interrupt level of the PIC.
E125 to E132*	Routes the RxRDY interrupt signal to IR0 interrupt level of the PIC.

Table 2-1. iSBC® 86/05 Jumpers (continued)

Jumper Pair	Function
E126	Provides the interrupt line (SBX2 INT1) from the iSBX Bus Connector J3 at the interrupt jumper matrix.
E127	Input to IR7 interrupt level of the PIC.
E128	Input to IR6 interrupt level of the PIC.
E129	EDGE INTR is the time-out interrupt which is available at the interrupt jumper matrix.
E130	Input to IR4 interrupt level of the PIC.
E131	Input to IR3 interrupt level of the PIC.
E132	Input to IR0 interrupt level of the PIC.
E133	Input to IR1 interrupt level of the PIC.
E134	Provides the TxRDY interrupt signal from the 8251A PCI device at the interrupt jumper matrix .
E135	No jumper post.
E137	Provides the SBX1 INT0 interrupt signal from the iSBX connector J4 at the interrupt jumper matrix.
E138	Provides the SBX1 INT1 interrupt signal from the iSBX connector J4 at the interrupt jumper matrix.
E139	Provides the MINT interrupt signal input from the iSBC 337 Numeric Data Processor at the interrupt jumper matrix.
E140	Provides the PB INTR interrupt signal from the parallel port at the interrupt jumper matrix.
E141	Provides the PA INTR interrupt signal from the parallel port at interrupt jumper matrix.
E136, E142, E144, E143, E146, E147 E148, E149	Output signals from the line receiver device interfacing the MULTIBUS interrupt signals to the interrupt jumper matrix.
E145 to E146*	Routes the Multibus interrupt line INT5/ input to IR5 interrupt level of the PIC.

Table 2-1. iSBC® 86/05 Jumpers (continued)

Jumper Pair	Function
E150	Provides the external POWER LINE CLOCK signal from the P2 connector at interrupt jumper matrix.
E151	No jumper post.
E152 to E153	Enables the Advance memory write command to the EPROM socket in case your devices require this signal. EPROM does not require this signal.
E154 thru E165	No jumper posts.
E166 and E167	Provide access to the option signals (OPT0 or OPT1), on the iSBX Bus connector J3.
E168 to E169	Enables Bus vectored interrupts
E170 to E171*	Routes the clock output from the 8284 to the 8086 clock input.
E172 and E173	Provides access to the option signals (OPT0 or OPT1), on the iSBX Bus connector J4.
E174 thru 176	No jumper posts.
E177 to E178*	Routes the +5V line to the +5VB line. This jumper is connected unless you want battery backup.
E179 to E180*	Routes BCLK/ system clock onto the MULTIBUS Bus interface when installed.
E181 to E182*	When installed allows 8 MHz processor rate; when removed, processor runs at 5 MHz.
E183 to E184*	Routes CBRQ/ signal to the MULTIBUS Bus interface when installed. See Table 2-18.
E184 to E185	Connects CBRQ/ signal to ground. See Table 2-18.
E186 to E187*	Routes BPRO/ signal to the MULTIBUS Bus interface when installed.
E188 to E189*	Connects ANY REQUEST signal of the 8289 device to pullup resistor. See Table 2-18.
E189 to E190	Connects ANY REQUEST signal of the 8289 device to ground. See Table 2-18.

Table 2-1. iSBC® 86/05 Jumpers (continued)

Jumper Pair	Function
E191 to E192*	Routes CCLK/ system clock onto the MULTIBUS Bus interface when installed.
E193	Provides the BUS INTR OUT signal from the parallel port area at the interrupt jumper matrix to generate an interrupt signal onto the Multibus.
E194 thru E201	Provides access to the MULTIBUS Bus interrupt lines (INT0/ through INTR7/).
E202 thru E205	No jumper posts.
E206 to E207	Configures the Multimodule installed into Multimodule connector J4 as a 16-bit device.
E208 to E209	Configures the Multimodule installed into Multimodule connector J3 as a 16-bit device.
E210 to E211	PROM device select jumper. See Table 2-4.
E212 to E213	PROM device select jumper. See Table 2-4.
E214 to E215	Reserved option for test purposes only. Do not use.
E215 to E216*	Enables BUS AEN/ signal to enable Multibus address drivers.
E217 to E218*	When the fail-safe timer is used and the 86/05 board is delayed in obtaining access to the bus such that timeout occurs before bus access is gained, the generation of the READY signal is delayed until bus access is obtained. If connected, you must connect jumper E221 to E222.
E217 to E219	When the fail-safe timer is used and the 86/05 board is delayed in obtaining access to the bus such that timeout occurs before bus access is gained, the READY signal is generated without waiting for bus access. If connected, you must connect jumper E220 to E221.
E220 to E221	Must be used in conjunction with E217 to E219 to route the ANDed XACK/ and BUS AEN/ signals to the Ready 1 line.
E221 to E222*	Must be used in conjunction with E217 and E218 to route XACK/ to Ready 1 line of the 8284A chip.

Table 2-2. iSBC® 86/05 Factory Default Jumpers

Jumper Pair	Function	Figure 4-3 Schematic Sheet
2 - 3	Wait State Generator	2
6 - 7	Wait State Generator	2
10 - 11	Wait State Generator	2
14 - 15	Failsafe Timer	2
19 - 20	PROM BUSY	2
25 - 27	PIT Gate 1 Control	8
28 - 29	Port A Direction	8
32 - 33	PIT Gate 0 Control	8
38 - 42	Port C Bit 5	8
39 - 43	Port C Bit 6	8
40 - 44	Port C Bit 7	8
41 - 45	Port C Bit 4	8
46 - 50	Port C Bit 0	8
47 - 51	Port C Bit 1	8
48 - 52	Port C Bit 2	8
49 - 53	Port C Bit 3	8
56 - 57	PIT Input CLK 2	7
58 - 59	PIT Input CLK 0	7
60 - 61	PIT Input CLK 1	7
79 - 80	RxC Input	7
83 - 84	TxC Input	7
89 - 90	OSC Output	7
91 - 92	DSR/ Input	7
107 - 108	Page Address	5
115 - 116	Page Address	5
120 - 121	NMI Gate	9
123 - 124	Timer 0 INTR to IR2	9
125 - 132	51RxINTR To IR0	9
145 - 146	Multibus INTR To IR5	9
170 - 171	TEST ONLY	2
177 - 178	Battery Defeat	1
179 - 180	BCLK/ To Multibus	4
181 - 182	8 MHz Operation	2
183 - 184	CBRQ/ To Multibus	4
186 - 187	BPRO/ To Multibus	4
188 - 189	See Section 2-24	4
191 - 192	CCLK/ To Multibus	4
217 - 218	Delayed timeout until bus accessed	2
221 - 222	XACK/	2

2.5.1 MEMORY JUMPER INFORMATION

The iSBC 86/05 board is configured at the factory to recognize two separate on-board 128K byte pages as valid memory addresses. (RAM and PROM memory may be located in 1 of 4 128K byte pages each.) Any address within either page is recognized as a valid memory address by the decoding circuitry. With the default jumpers installed, the address range for RAM is from 00000 to 1FFFF (hexadecimal) and the address range for PROM is from E0000 to FFFFF (hexadecimal). These pages may be altered by jumper selection as shown in Table 2-3. Section 2.5.1.1 provides PROM jumper information and section 2.5.1.2 provides the RAM jumper information.

Table 2-3. Page Select Jumpers

Function	Address Range	Jumper Connection
RAM ONLY	00000 - 1FFFF	108 - 107*
ROM/PROM ONLY	20000 - 3FFFF	111 - 107 or 115
RAM ONLY	40000 - 5FFFF	109 - 107 or 115
ROM/PROM ONLY	60000 - 7FFFF	113 - 107 or 115
RAM ONLY	80000 - 9FFFF	110 - 107 or 115
ROM/PROM ONLY	A0000 - BFFFF	114 - 107 or 115
RAM ONLY	C0000 - DFFFF	112 - 107 or 115
ROM/PROM ONLY	E0000 - FFFFF	116 - 115*

Note: * indicates factory default connection. Select any two pages; connect one for RAM and the other for PROM.

2.5.1.1 PROM Configurations

Sockets U33, U34, U66, and U67 are reserved for the ROM/PROM devices. A maximum of 64K bytes may be installed in these four sockets, using four 16K byte 28-pin devices. Refer to Table 2-4 for the jumper configurations to accommodate the various sizes of PROM. Table 2-5 lists the address ranges for the different device types with the factory default jumper configuration installed. Device types can not be mixed; however, empty sockets are allowed (provided they are not addressed). You could also add the optional iSBC 341 ROM/PROM Expansion Module to increase the amount of on-board ROM/PROM. The device types used on the iSBC 341 ROM/PROM Expansion Module can not be mixed and must match the device types used on the iSBC 86/05 board. Refer to Table 2-6 for the address ranges with the iSBC 341 ROM/PROM Expansion Module installed. Refer to paragraph 2.6.3 for the iSBC 341 ROM/PROM Expansion Module installation procedure.

Table 2-4. ROM/PROM Jumper Configurations

	DEVICE TYPE & SIZE			
	2716 2K x 8	2732 4K x 8	2764 8K x 8	27128 16K x 8
U35 Jumpers	1 - 14 2 - 13 3 - 12	1 - 14 3 - 12 6 - 9	1 - 14 6 - 9	1 - 14 6 - 9 7 - 8
Decode PROM Jumpers	None	210 - 211	212 - 213	210 - 211 212 - 213

Table 2-5. ROM/PROM Configurations

BANK NO: SOCKET NO: BYTE TYPE:	0 U66 LOW (EVEN) BYTES ONLY	1 U67 LOW (EVEN) BYTES ONLY	0 U33 HIGH (ODD) BYTES ONLY	1 U34 HIGH (ODD) BYTES ONLY	TOTAL ROM SPACE
DEVICE TYPE & SIZE:	ADDRESS RANGES				
2716 (2Kx8)	FE000- FEFFF	FF000- FFFFF	FE000- FEFFF	FF000- FFFFF	FE000- FFFFF
2732 (4Kx8)	FC000- FDFFF	FE000- FFFFF	FC000- FDFFF	FE000- FFFFF	FC000- FFFFF
2764 (8Kx8)	F8000- FBFFF	FC000- FFFFF	F8000- FBFFF	FC000- FFFFF	F8000- FFFFF
27128 (16Kx8)	F0000- F7FFF	F8000- FFFFF	F0000- F7FFF	F8000- FFFFF	F0000- FFFFF

Table 2-6. Address Ranges With iSBC® 341 Module

BANK NO: SOCKET NO: BYTE TYPE:	3 U 5 LOW (EVEN) BYTES ONLY	2 U 6 LOW (EVEN) BYTES ONLY	3 U 2 HIGH (ODD) BYTES ONLY	2 U 3 HIGH (ODD) BYTES ONLY	TOTAL ROM SPACE
DEVICE TYPE & SIZE:	ADDRESS RANGES				
2716 (2Kx8)	FC000- FCFFF	FD000- FDFFF	FC000- FCFFF	FD000- FDFFF	FC000- FFFFF
2732 (4Kx8)	F8000- F9FFF	FA000- FBFFF	F8000- F9FFF	FA000- FBFFF	F8000- FFFFF
2764 (8Kx8)	F0000- F3FFF	F4000- F7FFF	F0000- F3FFF	F4000- F7FFF	F0000- FFFFF
27128 (16Kx8)	E0000- E7FFF	E8000- EFFFF	E0000- E7FFF	E8000- EFFFF	E0000- FFFFF

2.5.1.2 RAM Configuration Considerations

The iSBC 86/05 board is shipped with 8K bytes of static RAM installed. This RAM memory is not shared with other system devices because the iSBC 86/05 board RAM is not "dual-ported". All on-board RAM must reside in 1 of 4 128K byte pages. Refer to Table 2-3 for Page Select jumper configurations. The only means of expanding on-board RAM is to install the optional iSBC 302 RAM Expansion Module. This doubles the on-board RAM size to 16K bytes. Refer to section 2.6.4 for the iSBC 302 RAM Expansion Module installation procedure.

2.5.2 INTERVAL TIMER JUMPER CONFIGURATIONS

The 8253-5 Programmable Interval Timer (PIT) is configured at the factory with three jumpers installed, as shown in Table 2-7. These three jumpers select the input frequencies to each of the three independent counters within the PIT. Outputs 0 and 1 from the timer are routed directly to the interrupt matrix (Section 2.5.5) as well as jumper posts E63 and E65, respectively. These outputs may then be jumpered to the desired on-board interrupt level, or routed off-board via the Multibus interrupt lines, by connection to one of the outbound posts (194 through 201). Refer to section 3.5 for programming information.

Output 2 is used for the 8251A Programmable Communications Interface (PCI) transmit and receive clocks.

Table 2-7. Interval Timer Input Jumper Configurations

Function	Jumper	Description
2.46 MHz	59 - 64	Optional input to CLK 0
	61 - 64	Optional input to CLK 1
	56 - 64	Optional input to CLK 2
1.23 MHz	58 - 59*	Default input to CLK 0
	56 - 57*	Default input to CLK 2
	57 - 60	Optional input to CLK 1
153.6 KHz	60 - 61*	Default input to CLK 1
Output 0	61 - 63	Optional cascade mode (see text)
Output 1	56 - 65	Optional cascade mode (see text)
Output 2	67 - XX	Optional cascade mode (see text)
External	62 - XX	Connect to 56, 59, 61 for external input (select one only)
Power Line Clock	66 - XX	Connect to 56, 59, 61 for PLC frequency

Note: * Indicates default connection; select one function per input.

2.5.3 SERIAL PORT JUMPER CONNECTIONS

The iSBC 86/05 board serial port is configured at the factory to the RS 232C standard interface. This permits direct interconnection to data terminal equipment via the I/O serial port connector J2. Jumper connections associated with the serial port are summarized in Table 2-8. Connector J2 pin assignments are provided in Table 2-9.

Table 2-8. Serial Port Jumper Configurations

Function	Jumpers	Description
On Board TxC	83 - 84*	Connects PIT output 2 to TxC
On Board RxC	79 - 80*	Connects PIT output 2 to RxC
External TxC	82 - 83	Connects J2-7 to TxC
External RxC	79 - 81	Connects J2-7 to RxC
Secondary TxC	72 - 74	Connects TxC clock to RS232 driver
Secondary TxD	73 - 74	Connects STxD to RS232 Driver
	75 - 76	Connects output of RS232 driver to J2-26
	76 - 78	Connects output of RS232 driver to J2-5
	76 - 77	Connects output of RS232 driver to J2-21
RTS/ to CTS/	87 - 88	Connects on board RTS/ to CTS/.
Voltages	68 - 69	Connects +5 Vdc to J2 - 23
	85 - 86	Connects +12 Vdc to J2 - 22
	70 - 71	Connects -12 Vdc to J2 - 19
DSR Defeat	91 - 92*	Disconnects DSR input when removed

Note: * Indicates default connection installed.

Table 2-9. Connector J2 Pin Assignments

Pin No.	Signal	PCI Function
J2 - 1	Not Used	--------
J2 - 2	Chassis GND	Protective Ground
J2 - 3	Not Used	--------
J2 - 4	Receive Data	RxD Input
J2 - 5	SEC REC SIG	STxD or TxC/TxD
J2 - 6	Transmit Data	TxD Output
J2 - 7	External Clock	TxC/RxC Input
J2 - 8	Clear To Send	CTS/ Input
J2 - 9	Not Used	--------
J2 - 10	Request To Send	RTS/ Output
J2 - 11	Not Used	--------
J2 - 12	Data Terminal Ready	DTR/ Output
J2 - 13	Data Set Ready	DSR/ Input
J2 - 14	Ground	GND
J2 - 15	Not Used	--------
J2 - 16	Not Used	--------
J2 - 17	Not Used	--------
J2 - 18	Not Used	--------
J2 - 19	-12V	--------
J2 - 20	Not Used	--------
J2 - 21	TRANS SIG ELE TIMING	STxD or TxC/TxD
J2 - 22	+12V	--------
J2 - 23	+5V	--------
J2 - 24	Not Used	--------
J2 - 25	Ground	GND
J2 - 26	Secondary TxD or Clock Out	STxD or TxC/TxD

Notes: 1. Odd numbered pins are on component side of board; even pins on trace side.
2. Cable connector numbering convention may not correspond with J2 numbering.
3. / indicates an active low true signal.

2.5.4 PARALLEL PORT JUMPER CONFIGURATIONS

Parallel Port C has a jumper matrix between the 8255A PPI device and the driver/terminator sockets U8 & U9. This arrangement allows a greater amount of flexibility.

Port A is equipped with an 8287 Bus Transceiver installed in socket U7. The control line for the Bus Transceiver is configured at the factory to operate the transceiver in the output only mode by connecting jumper 28 to 29. Two other modes are possible for the transceiver:

1. Input only mode: remove 28 - 29 and install 27 - 28.

2. Programmable mode: remove 28 - 29 or 27 - 28 and install a jumper between post 28 and the bit you select from Port C. (Connect to post on device side of Port C matrix.) The transceiver's direction or mode is then controlled by outputting the appropriate bit state to the device.

   0 out = output only mode
   1 out = input only mode

Parallel Port B operation is determined entirely by software programming and the type of devices installed in sockets U10 & U11. Refer to Table 3-20 for a list of operating modes allowed for each parallel port.

Table 2-10 lists the default connections for all parallel ports and shows the corresponding input/output connector pin numbers. Table 2-11 provides jumper information and descriptions of the optional features associated with the Port C jumper matrix. Table 2-12 is a comprehensive guide to mode restrictions and jumper connections for all three parallel ports.

Before configuring the parallel ports for your application, refer to Section 3.7 for 8255A-5 programming information.

Table 2-10. Parallel Port Default Jumper Connections

Port C Bit	Jumper Conn.	J1 Pin Number
0	46 - 50	24
1	47 - 51	22
2	48 - 52	20
3	49 - 53	18
4	41 - 45	26
5	38 - 42	28
6	39 - 43	30
7	40 - 44	32
Port A Bit	**Jumper Conn.**	**J1 Pin Number**
0	NONE	48
1		46
2		44
3		42
4		40
5		40
6		38
7		36
Note: Driver/Terminators not installed at factory.		

Table 2-10. Parallel Port Default Jumper Connections (continued)

Port B Bit	Jumper Conn.	J1 Pin Number
0	NONE	16
1		14
2		12
3		10
4		8
5		6
6		4
7		2
Note: Driver/Terminators not installed at factory.		

Table 2-11. Parallel Port C Jumper Configurations

Function	Jumpers	Description
OUTPUTS:		
EXT CLK/	23 - XX	Connect to desired jumper post from the parallel port matrix; driver terminator socket must have terminator. See Table 2-7.
OVERRIDE/	24 - XX	Software programmable Multibus override control. Connect to desired bit from parallel port matrix.
SECONDARY TxD	22 - XX	Software programmable transmit channel. Connect to desired bit from parallel port matrix.
PA INTR	34 - XX	Parallel port interrupt "A". Software programmable on-board interrupt. Connect to desired bit from parallel port matrix. See Table 2-13 for associated required jumper connection.
PB INTR	30 - XX	Parallel port interrupt "B". Software programmable on-board interrupt. Connect to desired bit from parallel port matrix. See Table 2-13 for associated required jumper connection.

Table 2-11. Parallel Port C Jumper Configurations (continued)

Function	Jumpers	Description
TEST/	36 - XX	Software programmable TEST/ input. When asserted, causes the 8086 to either execute WAIT states or become idle.
BUS INTR OUT	31 - XX	Software programmable Multibus (System) interrupt output. Requires additional connection from jumper post 193 to desired output post (194 through 201). See schematic sheet 9 for levels.
GATE 0 CNTRL	33 - XX	Software programmable gate input for 8253A PIT. Connect to desired bit from parallel port matrix.
GATE 1 CNTRL	25 - XX	Software programmable gate input for 8253A PIT. Connect to desired bit from parallel port matrix.
NMI MASK/	26 - XX	Software programmable means to switch the 8086 NMI input on or off. A low disables the NMI input gate. Connect to desired bit from the parallel port matrix.
DS1	54 - 55	Software programmable indicator lamp. Lamp is connected to bit 3 of Port C. Install jumper to enable.
PORT A DIRECTION	28 - XX	Controls direction (input or output) of Port A. Refer to Section 2.5.4
INPUT:		
PFSN/	37 - XX	Power fail sense line. This line is an output from an off-board latch which indicates the occurrence of a power failure. PFSN/ may be read by the parallel port, in conjunction with a battery backup power-on sequence. Refer to Section 2.5.10. Connect post 37 to one of the available input lines.

Note: XX denotes variable bit choice. Only one function per bit is allowed.

Table 2-12. Parallel Port Jumpers & Restrictions

Port	Mode	Driver (D) Terminator	Jumper Configuration			Port	Restrictions
			Delete	Add	Effect		
A	0 input	8287:U7	28-29	28-27	8287=input enabled.	B	None; can be Mode 0 or 1, input or output.
						C	None; can be Mode 0 or 1, input or output, unless Port B is in Mode 1.
A	0 Output (latched)	8287; U7		28-29	8287=output enabled.	B	None; can be Mode 0 or 1, input or output.
						C	None; can be Mode 0 or 1, input or output, unless Port B is in Mode 1.
A	1 Input (strobed)	8287: U7 T: U8 D: U9	28-29	27-28	8287=input enabled.	B	None; can be Mode 0 or 1, input or output.
			53-49 38-42 and 49-53	34-53 41-45 42-49	Connects PA interrupt to Bit 3. Connects J1 pin 26 to STBa input. Connects IBFa output to J1 pin 18.	C	Port C bits perform the following: Bits 0, 1, 2 control Port B if Port B is in Mode 1. Bit 3 - Port A interrupt jumper matrix. Bit 4 - Port A Strobe (STB) input. Bit 5 - Port A Input buffer full (IBF) output.

Table 2-12. Parallel Port Jumpers & Restrictions (continued)

Port	Mode	Driver (D) Terminator	Jumper Configuration			Port	Restrictions
			Delete	Add	Effect		
			53-49	34-53	Connects INTA output to interrupt matrix.		Bits 6,7 of Port C input or output (both must be in same direction.)
A	1 Output (latched)	8287: U7 T: U8 D: U9		28-29	8287=output enabled.	B	None; can be in Mode 0 or 1, input or output.
						C	Port C bits perform the following: Bits 0, 1, 2 - control Port B if Port B is in Mode 1.
			49-53	34-53	Connects INTa output interrupt matrix.		Bit 3 - Port A interrupt PA INTR to interrupt jumper matrix. Bits 4, 5 - Port C input or output (both must be in same direction.
				39-43	Connects J1 pin 30 to ACKa input.		Bit 6 - Port A Acknowledge (ACK) input.
			40-44	44-49	Connects OBFa output to J1 pin		Bit 7 - Port A Output Buffer Full (OBF) output.
A	2 bi-direct-ional	8287: U7 T: U8	28-29	28-39	Allows ACKa input to control 8287 in/out direction.	B	None; can be in Mode 0 or 1, input or output.

Table 2-12. Parallel Port Jumpers & Restrictions

Port	Mode	Driver (D) Terminator	Jumper Configuration			Port	Restrictions
			Delete	Add	Effect		
						C	Port C bits perform the following: Bit 0 - Can only be used for jumper option. Bits 1,2 - Can be used for input or output if Port B is in Mode 0.
			49-53 38-42 and 50-46	34-53 41-45 42-46 39-43	Connects INTa output to interrupt matrix. Connects J1 pin 26 to STB input. Connects IBF output to J1 pin 24. Connects J1 pin 30 to ACK input. Connects OBFa output to J1 pin 18.		Bit 3 - Port A interrupt PA INTR to interrupt matrix. Bit 4 - Port A strobe STB input. Bit 5 - Port A Input Buffer Full (IBF) output. Bit 6 - Port A Acknowledge (ACK) input. Bit 7 - Port A Output Buffer Full (OBF) output.
B	0 Input	T:U10, U11	None	None		B	None
						C	None; Port C can be in Mode 0, input or output, if Port A is also in Mode 0.
B	0 Output latched	D:U10, U11	None	None		B	None
						C	None; Port C can be in Mode 0, input or output, if Port A is also in Mode 0.

Table 2-12. Parallel Port Jumpers & Restrictions (continued)

Port	Mode	Driver (D) Terminator	Jumper Configuration			Port	Restrictions
			Delete	Add	Effect		
B	1 Input strobed	T:U8, U10, and U11 D: U9		47-51	Connects IBFb output to J1 pin 22.	A	None
						C	Port C bits perform the following: Bit 0 - Port B interrupt PB INTR to interrupt jumper matrix. Bit 1 - Port B Input Buffer Full (IBF) output. Bit 2 - Port B Strobe (STB) input. Bit 3 - If Port A is in Mode 0, bit 3 can be input or output. Otherwise, bit 3 is reserved. Bits 4, 5, 6, 7 depend on Port A Mode.
			50-46	30-50	Connects INTb output to interrupt matrix.		
			40-44 and 48-52	40-52	Connects J1 pin 32 to STBb input.		
B	1 Output latched	T: U8 D: U9, U10, U11		47-51	Connects OBFb output to J1 pin 22.	A	None
						C	Port C bits perform the following: Bit 0 - Port B interrupt PBINTR to interrupt jumper matrix. Bit 1 - Port B Output Buffer Full (OBF) output.
			46-50	30-50	Connects INTb output to interrupt matrix.		

Table 2-12. Parallel Port Jumpers & Restrictions (continued)

Port	Mode	Driver (D) Terminator	Jumper Configuration			Port	Restrictions
			Delete	Add	Effect		
			40-44 and 48-52	40-52	Connects J1 pin 32 to ACK input.		Bit 2-Port B Acknowledge (ACK) input. Bit 3- If Port A is in Mode 0, bit 3 an be input or output. Otherwise, bit 3 is reserved. Bits 4, 5, 6, 7 depend on Port A Mode.
C	0 Input	T: U8	None	41-45 38-42 39-43 40-44	Connects bit 4 to J1 pin 26. Connects bit 5 to J1 pin 28. Connects bit 6 to J1 pin 30. Connects bit 7 to J1 pin 32.	A	Port A must be in Mode 0 for all four bits to be available.
						B	Port B must be in Mode 0 for all four bits to be available.
C	0 Input	T: U9	None	46-50 47-51 48-52	Connects bit 0 to J1 pin 24. Connects bit 1 to J1 pin 22. Connects bit 2 to J1 pin 20.	A	Port A must be in Mode 0 for all 4 bits to be available.
						B	Port B must be in Mode 0 for all 4 bits to be available.
C	0 Output	D:U8	None	Same as Port C (upper) Mode 0 input.		A	Same as for Port C (Upper) Mode 0 Input.
C	0 Output	D: U9	None	Same as Port C (lower) Mode 0.		B	Same as for Port C (lower) Mode 0 Input.

2.5.5 INTERRUPT MATRIX JUMPER CONFIGURATIONS

The iSBC 86/05 board provides jumper posts for 12 on-board interrupt sources and 11 off-board sources. Any eight of these these sources can be interfaced to the 8259A Programmable Interrupt Controller (PIC) through the on-board interrupt matrix. The PIC provides eight interrupt levels. In addition, the 8086 CPU NMI input can be used for high priority interrupt requests.

In the factory default configuration, the following four interrupt matrix jumpers are installed:

a. 123 - 124 Timer 0 output to IR2 on PIC
b. 145 - 146 Multibus interrupt INT5/ to IR5 on PIC
c. 120 - 121 Disable NMI Mask gate
d. 125 - 132 RxRDY Interrupt

Table 2-13 provides a complete list of possible interrupt jumper configurations on the iSBC 86/05 board. Table 2-14 lists the output jumper configurations. Refer to Section 3.8 for 8259A programming information.

In addition, the iSBC 86/05 board supports Multibus vectored interrupts from off-board slave 8259A interrupt controllers. Refer to Section 2.5.6 for information on Multibus vectored interrupts.

The following sections provide brief descriptions of all interrupt request lines which are part of the interrupt matrix or related to the iSBC 86/05 board interrupt structure.

2.5.5.1 iSBX™ Multimodule™ Interrupts (SBX1 INT0,1; SBX2 INT0,1)

Two interrupt request lines are available for each iSBX Multimodule board installed on the iSBC 86/05 board.

Table 2-13. Interrupt Matrix Jumper Configurations

Matrix Inputs	Description	Jumper Posts
iSBX 2 INT0	J3 Multimodule INT0	117
iSBX 2 INT1	J3 Multimodule INT1	126
iSBX 1 INT0	J4 Multimodule INT0	137
iSBX 1 INT1	J4 Multimodule INT1	138
TIMER0 INTR	PIT Output 0	123
TIMER1 INTR	PIT Output 1	122
PA INTR	Parallel Port INT A	141
PB INTR	Parallel Port INT B	140
51TxINTR	PCI Transmit INT	134
51RxINTR	PCI Receive INT	125
PLC	Power Line Clock	150
MINT	8087 Math Chip INT	139
PFIN/	Power Fail INT	118
EXT INTR0	External INT from J1-50	119
EDGE INTR	Edge Sensitive Mode INT	129
INT0/	Multibus INT from P1-41	144
INT1/	Multibus INT from P1-42	136
INT2/	Multibus INT from P1-39	142
INT3/	Multibus INT from P1-40	143
INT4/	Multibus INT from P1-37	147
INT5/	Multibus INT from P1-38	146
INT6/	Multibus INT from P1-35	149
INT7/	Multibus INT from P1-36	148
IR0	Interrupt Controller input	132
IR1	Interrupt Controller input	133
IR2	Interrupt Controller input	124
IR3	Interrupt Controller input	131
IR4	Interrupt Controller input	130
IR5	Interrupt Controller input	145
IR6	Interrupt Controller input	128
IR7	Interrupt Controller input	127
NMI	Maskable NMI gate input	120
GND	Ground	121

Table 2-14. Multibus® Interrupt Output Jumper Configuration

Output Line	P1 Pin	Multibus Jumper Post	Bus INTR Out Post
INT0/	41	199	193
INT1/	42	201	
INT2/	39	195	
INT3/	40	197	
INT4/	37	194	
INT5/	38	196	
INT6/	35	200	
INT7/	36	198	

Note: Connect one jumper only from desired Multibus jumper post to the Bus Interrupt Output post (193). This option also requires a parallel port jumper matrix connection - see section 2.5.4.

2.5.5.2 Interval Timer Outputs (TIMER0 and TIMER1 INTR)

These two lines come directly from the 8253A Interval Timer. The timer 0 line is jumpered at the factory to interrupt request line INT2 (123 to 124). The timer 1 output line is not connected at the factory.

2.5.5.3 Parallel Port Interrupts A, B (PA INTR & PB INTR)

These two lines are software programmable interrupt lines. Connect each line to the desired interrupt request input. Refer to Section 2.5.4 for instructions on installing the parallel port matrix jumpers required for this option.

2.5.5.4 Transmit and Receive Interrupts (51TxINTR & 51RxINTR)

These signals originate at the 8251A Programmable Communications Interface (PCI) device. The signal 51TxINTR is equivalent to TxRDY on the PCI, and when true, indicates that the PCI is ready to accept a data character from the CPU. Likewise, 51RxINTR is equivalent to RxRDY, and when true, indicates that the PCI contains a data character to be read by the CPU. Refer to the Intel Component Data Catalog for additional PCI information.

2.5.5.5 Power Line Clock (PLC)

This external signal is supplied by a chassis, or similar circuit. It enters the board via auxiliary connector pin P2-31 and is specified at 120 Hz (double the AC line frequency).

2.5.5.6 Math Interrupt (MINT)

This signal originates from the optional iSBC 337 Numeric Data Processor Multimodule board. This interrupt is used only in conjunction with this option.

2.5.5.7 Power Fail Interrupt (PFIN/)

Furnished by the iSBC Power Supply (or equivalent), this signal indicates that an AC line power failure has occurred and DC voltage loss is imminent. This signal can be jumpered to the NMI input on the 8086 CPU or to any of the 8259 PIC inputs. It is used in conjunction with a user written power down routine and battery backup scheme. Refer to Section 2.5.10 for additional battery backup information.

2.5.5.8 External Interrupt 0 (EXT INTR0)

This external interrupt signal enters the board via parallel port connector P1-50. The incoming signal is inverted by the iSBC 86/05 board, therefore a low state (level mode) or a high-to-low transition (edge sensitive mode) will activate the interrupt request.

2.5.5.9 Non-Maskable Interrupt Input Mask

The 8086 CPU Non-Maskable Interrupt (NMI) input may be configured to be enabled and disabled by the software. The NMI input is intended mainly for catastrophic error handling. The fact that NMI interrupts a program even when interrupts are disabled can be both an advantage and a hazard. The use of NMI must be coordinated with the software and application environment in which the board is used. In some cases a power failure interrupt can be connected to NMI, but you must consider the action taken by the software on both power-down and power-up. Connecting the NMI input to a system power failure interrupt may be suitable for some software applications, but not all software.

Specifically, multi-tasking software that is expected to continue operation (as opposed to restarting) after a power failure may not function correctly if a power fail interrupt is connected to NMI. The problem occurs because the software disables interrupts to protect the integrity of its internal data structures while they are being modified.

Disabling interrupts; however, does not disable NMI. If an NMI occurs while a data structure is partially modified and the NMI service routine uses this data structure or does not perform a normal return (common with power-fail routines), the data structure will be corrupted.

If the interrupt service routine for NMI does not use any of the data structures and does a normal interrupt return on completion, then, it is safe to use. An advantageous way to use the NMI input is for debugging. An operator input (such as connecting the NMI input to an interrupt pushbutton on an Intel system chassis) would allow operator intervention as an aid in software debugging. Connecting the NMI input from an error signal resulting from a memory parity error or bus time-out could also be an aid in debugging both hardware and software.

Two jumper connections are required to enable this option: first, remove jumper connection 120 - 121 and install a jumper between interrupt matrix post 120 and the desired interrupt source. Then install another jumper between post 26 and the desired parallel Port C bit jumper post. A high state on this line enables the mask gate. A low state disables the mask gate preventing any non-maskable interrupts from reaching the 8086 CPU.

2.5.5.10 Multibus® Interrupt Output Option (BUS INTR OUT)

The iSBC 86/05 board has an optional interrupt output configuration which is used to drive Multibus system bus interrupt sources. This would allow you to issue an interrupt request on a system Multibus line with one of the parallel Port C bits. Two connections are required for this scheme: one jumper connection from the desired Port C bit to jumper post 31 and another jumper connection from jumper post 193 and the desired Multibus Interrupt Line. Refer to tables 2-13 and 2-14.

2.5.6 MULTIBUS® VECTORED INTERRUPTS

The iSBC 86/05 board has the capability to service interrupt requests which originate with a request to a slave, off-board 8259A Programmable Interrupt Controller (PIC). The slave INTR output is connected to the master PIC on the iSBC 86/05 board via the Multibus lines. This type of interrupt request is called a Bus Vectored Interrupt. In general, a bus vectored interrupt should be of lower priority than interrupt requests which are input directly to the master PIC. The iSBC 86/05 board is configured at the factory to accept bus vectored interrupts. To disable this feature, you must install jumper 168 - 169.

The on-board PIC (master) may be interfaced with two slave PIC devices. This arrangement leaves the master PIC with six inputs (IR2 through IR7) that can be used to handle the various on-board interrupt functions. The example scheme is implemented by programming the master PIC to handle IR0 & IR1 as bus vectored interrupts.

Each interrupt input (IR0 through IR7) to the master PIC may be individually programmed to be bus vectored or non-bus vectored. In the bus vectored mode, the slave PIC generates the interrupt vector address, and in the non-bus vectored mode the master PIC generates the interrupt vector address.

Slave PIC devices must be identified as such during their initialization sequences (with ICW3). The master PIC must also be initialized to support slave PIC devices. Section 3.8 describes 8259A programming and provides initialization examples.

2.5.7 ON-BOARD TIMING JUMPER SELECTION

The iSBC 86/05 board is configured at the factory to operate at 8 MHz. However, the board may be operated at 5 MHz by removing jumper 181 - 182.

2.5.7.1 Bus Clock & Constant Clock Selection

Bus Clock (BCLK/) and Constant Clock (CCLK/) are standard Multibus signals, common to most Intel iSBC boards. One and only one Bus Master should generate these clocks. If more than one Bus Master is used in your system, these clock signals should be disabled from all but the one Bus Master. CCLK/ is 180 degrees out of phase from BCLK/. The frequency of these signals is 9.83 MHz. Table 2-15 provides BCLK/ & CCLK/ jumper information:

Table 2-15. BCLK and CCLK Jumper Configurations

Signal	Jumpers In	Multibus® Pin
BCLK/	179 - 180*	P1 - 13
CCLK/	191 - 192*	P1 - 31

Notes: Either signal may be disabled by removing the appropriate jumper connection.
*Indicates factory default connection.

2.5.7.2 Wait State Generator Selection

The iSBC 86/05 board utilizes a wait state generator to allow the 8086 processor to wait for on-board addressed devices. Wait states are generated for all PROM, I/O, and interrupt requests. The number of wait states for each function is jumper selectable, allowing maximum utilization of processor time according to your system configuration. Table 2-16 provides the possible jumper configurations for the wait state generator and provides the maximum address to data times for each function.

Table 2-16. Wait State Generator Jumpers & Times

Function	Jumper Connection	Number of Waits	8 MHz Time**	5 MHz Time**
PROM Wait	6 - 5	0	216	364
	6 - 4	1	341	564
	6 - 7*	2	466	764
	6 - 8	3	591	964
I/O Wait	1 - 2	1	372	597
	2 - 3*	2	497	797
INT ACK Wait	9 - 10	2	497	797
	10 - 11*	3	622	997

Notes: * indicates factory default connection.
** indicates all times specified in nanoseconds; chip enable To Data.

2.5.8 iSBX™ Multimodule™ Board Mode Jumper Selection

The iSBC 86/05 board supports both 8-bit and 16-bit iSBX Multimodule expansion boards. There is a jumper connection for each iSBX connector, which specifies whether it is an 8-bit Multimodule expansion board or a 16-bit Multimodule expansion board. Table 2-17 defines the jumper connections.

Table 2-17. iSBX™ Jumper Configuration 8/16 Bit

iSBX™ Connector	Jumpers	8-Bit	16-Bit
J3	208 - 209	OUT*	IN
J4	206 - 207	OUT*	IN

Note: * indicates factory default connection.

2.5.9 Bus Arbiter Jumper Configuration

The control of the Multibus system bus interface depends on the jumper configuration of the 8289 Bus Arbiter. Table 2-18 lists the different jumper configurations and the conditions under which the Multibus interface is surrendered.

Table 2-18. 8289 Bus Arbiter Jumper Configurations

Configuration Number	Jumper Connector	CBRQ/	ANYRQST	Description
1	183 - 184* 189 to 190	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.
2	188 - 189* 183 - 184*	Low	High	The Bus Arbiter that has control of the of the Multibus interface will surrender control to the Bus Arbiter that is pulling CBRQ/ low, regardless of its priority, upon completion of the current bus cycle.
		High	High	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.
3	184 - 185 188 - 189*	Low	High	The Bus Arbiter that has control of the Multibus interface will surrender the use of the Multibus interface after each transfer cycle.

2.5.10 Power Fail Battery Backup Provisions

In an optional mode, the iSBC 86/05 board may be configured for battery backup operation. This means you may have a dc battery or an uninterrupted power supply connected to the board which is used to preserve memory during an ac power failure. In order for the battery backup scheme to function, your power supply must provide the following signals:

a. PFIN/ Power Fail Interrupt. Asserted at least 8 milliseconds before dc voltages are lost.

b. MPRO/ Memory Protect. Asserted at least 50 microseconds before dc voltages are lost.

c. PFSN/ Power Fail Sense. The output of an external, battery powered latch which indicates a power failure has occurred.

To implement a typical battery backup scheme on the iSBC 86/05 board, the following connections are required:

a. Connect +5 Volt battery positive leads to auxiliary connector pins P2-3 and P2-4.

b. Connect battery return leads to auxiliary connector pins P2-1 and P2-2.

c. Remove jumper connection 177 - 178.

d. Connect power supply PFIN/ line to auxiliary connector P2-19.

e. Connect the Power Fail Interrupt from the power supply (Jumper post E118) to an interrupt input as required by your software.

f. Install a jumper between parallel port matrix post E36 and the desired Port C input bit.

g. Connect power supply MPRO/ line to auxiliary connector P2-20.

h. Connect the PFSN/ (not supplied on the iSBC 86/05 board) line to auxiliary connector P2-17. PFSN/ is the output of an external, battery powered latch which indicates that a power failure has occurred. This latch is reset by PFSR/, which can be implemented with an unused Port C bit.

In this typical battery backup configuration, if a power failure occurs, the power supply asserts PFIN/ which in turn initiates an interrupt, placing the board in a power fail interrupt routine. Contents of various internal registers are stored in RAM, which is then locked up when MPRO/ is asserted. When power is restored, the PFSN/ signal is read by the parallel port, indicating a power failure has occurred. Your power on routine could then read the register contents from RAM and reset the PFSN/ signal before executing, thereby restarting the system.

2.6 INSTALLATION

Installation consists of installing the selected PROM/PROM devices onto the board, refer to 2.6.1. If additional PROM/ROM is required, refer to 2.6.3. If additional RAM is required, refer to 2.6.4. If line drivers or terminators are required for your application, refer to 2.6.2. To configure your serial I/O port and parallel I/O ports to your specific application, refer to 2.5.3 and 2.5.4. Once all the jumpers and components have been installed and the backplane configured to implement your system requirements, install the iSBC 86/05 board in place within your system chassis. Refer to 2.6.6.

Refer to section 2.6.6 if an iSBX Multimodule board is needed to suit your particular application needs. Refer to section 2.7 for connector information.

2.6.1 ROM/PROM INSTALLATION

Sockets U33, U34 & U66, U67 are reserved for the ROM/PROM devices. A maximum of 64K bytes may be installed in these four sockets, using four 16K byte 28-pin devices. A summary of compatible device types, and capacity is provided in Table 2-4. Device types may not be mixed, but empty sockets are allowed (provided they are not addressed).

Table 2-5 correlates the banks, bytes, socket numbers and addresses. Notice that ROM/PROM space is arranged into two banks (0 & 1), with each bank having a low byte and a high byte. When the 8086-2 CPU addresses a full 16-bit word, the low (or even) byte and the high (or odd) byte comprise the word. The CPU also has the capability to address either byte separately.

Before installing the devices on the board, several jumper connections are required to specify device size and power scheme. Table 2-4 summarizes the possible ROM/PROM jumper connections.

CAUTION

Never install any device onto a board when power is applied. Damage to the board, device, and power supply could result.

CAUTION

The ROM/PROM sockets are 28-pin sockets which are used for both 24-pin & 28-pin devices. When inserting devices, ensure that pin 1 of the ROM/PROM device corresponds with pin 1 of the socket. Use the upper white dot for 28-pin devices & the lower white dot for 24-pin devices (Figure 2-4).

Table 2-6 provides address ranges when the optional iSBC 341 ROM/PROM Expansion Module is installed on the iSBC 86/05 board. This option should be installed only when your total on-board ROM/PROM requirements exceed 64K bytes. The addresses listed include ROM/PROM space already on the iSBC 86/05 board. Refer to Section 2.6.3 for iSBC 341 Module installation information.

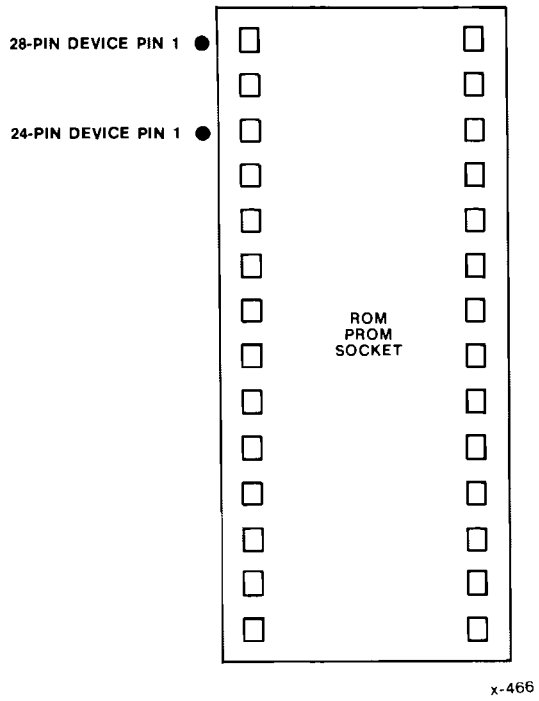


Figure 2-4. ROM/PROM Device Insertion

2.6.2 LINE DRIVERS & I/O TERMINATORS

When using parallel Ports B and C, line drivers or terminators are required for operation. The iSBC 86/05 board is equipped with a bidirectional bus transceiver for parallel Port A. Sockets U8 through U11 provide for the installation of either line driver or I/O terminator devices. Table 2-19 lists the types of terminators and line drivers which are recommended for this purpose.

Table 2-19. Line Driver and Terminator Circuits

Line Drivers	Current		I/O Terminators
	IOL	IOH or IIH	
7400 I	16 mA	-400uA	iSBC 901
7403 I, OC	16 mA	40uA	
7408 NI	16 mA	-800uA	iSBC 902
7409 NI, OC	16 mA	40uA	
I = Inverting; NI = Non-Inverting; OC = Open Collector;			

2.6.3 iSBC® 341 PROM/ROM EXPANSION MODULE INSTALLATION

The optional iSBC 341 ROM/PROM Expansion Module is designed to increase the amount of iSBC 86/05 on-board ROM/PROM. The size of the devices used on the module must match the size of the devices used on the board. For example, if the iSBC 86/05 board is equipped with 4K x 8 EPROM devices, the iSBC 341 module must also use 4K x 8 EPROM devices. The following procedure is recommended for iSBC 341 module installation:

1. Turn off power and remove iSBC 86/05 board from the system.

2. Ensure that the jumpers of U35 are correct for the PROM device used (refer to Table 2-4). Carefully remove the ROM/PROM device from board socket U34 and install it into socket U1 on the iSBC 341 module. Similarly, remove the ROM/PROM device from board socket U67 and install it into socket U4 on the iSBC 341 module. Be sure to install these two devices in the sockets indicated. Byte order will be reversed if the ROM/PROM devices are incorrectly installed. Refer to CAUTION notice in Section 2.6.1 for related socket information.

3. Carefully insert all remaining ROM/PROM devices into the iSBC 341 module. Refer to CAUTION notice in Section 2.3.2 for related socket information.

4. Using the hardware supplied with the iSBC 341 module, install the module onto the iSBC 86/05 board as shown in Figure 2-5. The module connector fits directly into ROM/PROM sockets U34 and U67, and board connector J6.

5. Install jumper connection 105 - 106 on the iSBC 86/05 board.

6. Installation is complete. The iSBC 86/05 board is now ready to be installed into your system cardcage.

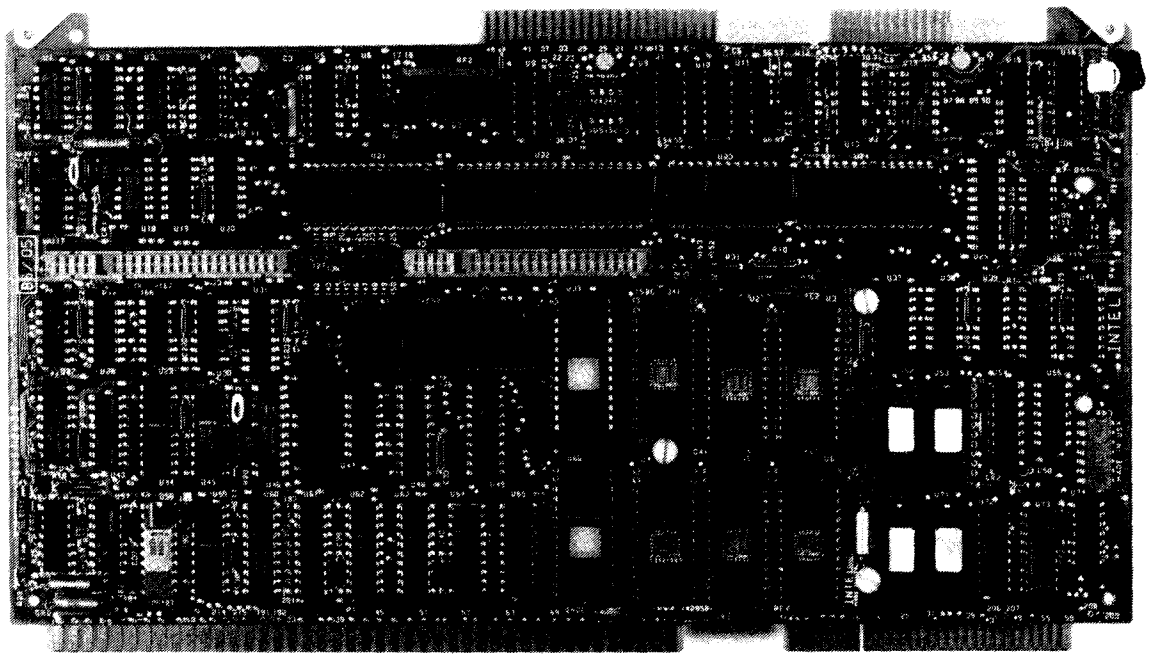


Figure 2-5. iSBC® 341 Module Insertion

2.6.4 iSBC® 302 RAM MODULE INSTALLATION

The iSBC 86/05 board is shipped with 8K bytes of static RAM in place. Intel 2168 RAM devices are used in this configuration. The only means of expanding on-board RAM is by installing the optional iSBC 302 RAM Expansion Module. This will double the on-board RAM to 16K bytes. The following procedure is recommended for installing the RAM module:

1. Turn power off and remove the iSBC 86/05 board from its system.

2. Carefully remove the RAM devices from board sockets U53 and U71. Save these devices for use on the iSBC 302 module.

3. Using the hardware supplied with the iSBC 302 module, install the module onto the iSBC 86/05 board as shown in Figure 2-6. The iSBC 302 module connector pins fit directly into board sockets U53 & U71, and board connector J7. Ensure that all pins fit correctly before tightening the hardware.

4. Install the two RAM devices removed in step (b) into module sockets U1 and U4. Ensure pin 1 of the device matches pin 1 on the socket.

5. Install jumper 104 - 105 on the iSBC 86/05 board.

6. On-board RAM space with 16K bytes covers address range 0 - 3FFF (hexadecimal) with the board in the default configuration.

7. Installation is complete. The iSBC 86/05 board is now ready to be installed into your system cardcage. Refer to 2.6.6.

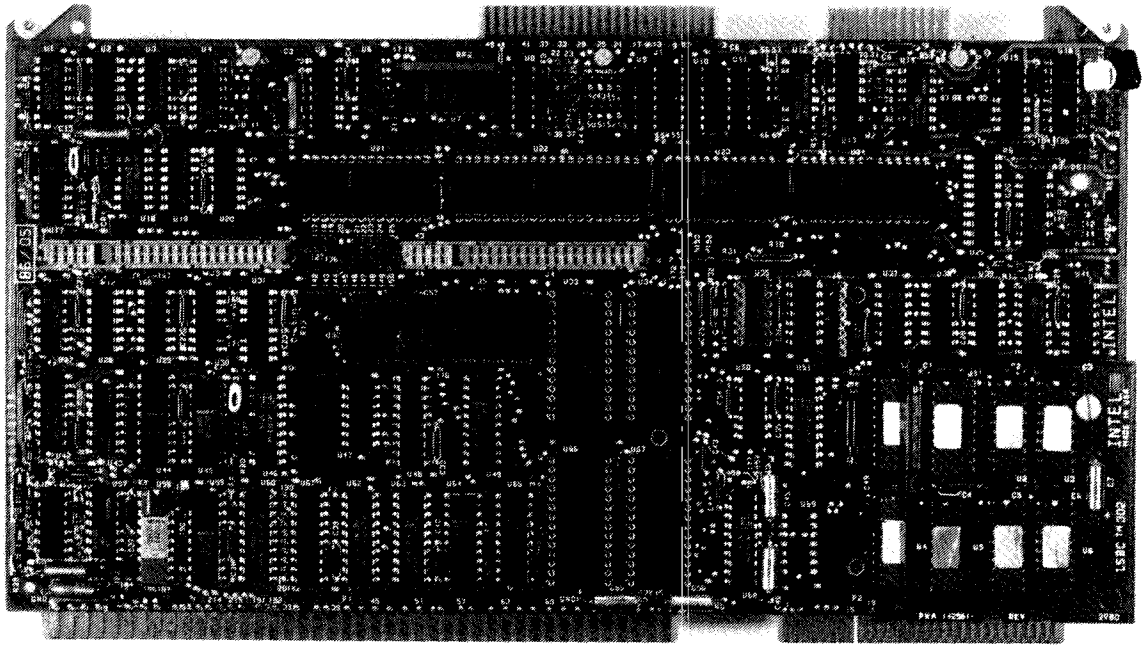


Figure 2-6. iSBC® 302 Module Insertion

2.6.6 iSBX™ MULTIMODULE™ BOARD INSTALLATION

The iSBC 86/05 board is equipped with two iSBX (single board expansion) bus connectors (J3 & J4). This bus allows on-board I/O expansion, using optional iSBX Multimodule boards. These boards should not be confused with the optional RAM and ROM/PROM expansion boards which are interfaced to the on-board memory bus. Connectors J3 & J4 may be used only for iSBX Multimodule boards.

For installation instructions, refer to the specific iSBX Multimodule board Hardware Reference Manual. When a Multimodule board is installed, the iSBC 86/05 board's power requirements will increase by the amount specified in the Multimodule board reference manual.

2.6.6 FINAL INSTALLATION

In an iSBC Single Board Computer based system, install the iSBC 86/05 board in the cardcage slot which corresponds to your priority scheme. Ensure that an auxiliary connector is installed in the cardcage if any of the iSBC 86/05 board P2 signals are used in your system.

CAUTION

Always turn off the system power supply before installing or removing the iSBC 86/05 board from its system, or before installing or removing any I/O cables. Failure to observe these precautions may result in damage to the board.

2.7 CONNECTOR INFORMATION

For systems applications, the iSBC 86/05 board is designed for installation into a standard Intel Multibus cardcage assembly. For OEM applications, the board may be interfaced to other hardware by means of separately purchased Multibus compatible connectors. Table 2-20 lists recommended suppliers for such connectors.

Parallel and serial I/O connector information is also supplied in Table 2-20. For related information on parallel I/O and serial I/O cabling, refer to Sections 2.7.1 and 2.7.2.

2.7.1 PARALLEL I/O CABLING

Parallel I/O Ports A, B, & C are controlled by the 8255A-5 Parallel Peripheral Interface (PPI) device at location U22, and are connected to external equipment via edge connector J1. Pin assignments for edge connector J1 are provided in Table 2-21. Bit order for Port C may be altered by jumper connection. Refer to Section 2.5.4 for information.

For maximum reliability, the transmission path from the I/O source to the iSBC 86/05 board should be limited to a maximum of 3 meters (10 feet). Recommended bulk cable types are provided in Table 2-22.

2.7.2 SERIAL I/O CABLING

Pin assignments and signal names for the serial I/O port interface connector (J2) are listed in Table 2-23. For OEM applications where cables will be made for the iSBC 86/05 board, it is important to note that the mating connector for J2 has 26 pins, whereas the RS 232C connector has 25 pins. Consequently, when connecting the 26-pin mating connector to 25-conductor flat cable, be sure that the cable makes contact with pins 1 & 2 of the mating connector and not pin 26. Table 2-24 provides pin correspondence between connector J2 and the RS 232C connector. When attaching the connector to J2 be sure that the PC connector is oriented properly with respect to pin 1 on the board. Refer to the footnote at the bottom of Table 2-23.

Table 2-20. User Furnished Connector Details

Connector	No. of Pins	Connector Type	Vendor	Vendor Part No.	Intel Part No.
Parallel I/O Conn. J1	25/50	Flat Crimp	3M 3M AMP ANSLEY SAE	3415-0001 W/ears 3415-0000 W/ears 88083-1 609-5015 S06750 Series	102211-003
		Soldered Pierced Tail	GTE SYLVANIA MASTERITE MICRO PLASTICS VIKING	6AD01-25-1A1-DD NDD8GR25-DR-H-X MP-0100-25-DP-1 3KH25/9JN5	102237-001
		Wire Wrap	VIKING TI ITT CANNON	3KH25/JND5 H421011-25 EC4A050A1A	
Serial I/O Conn. J2	13/26	Soldered Mounting Holes	AMP EDAC	1-583715-1 345-026-520-202	102233-001
		Flat Crimp	3M AMP	3462-0001 88373-5	102210-001
		Soldered Pierced Tail	EDAC	345-026-500-201	
		Wire Wrap	EDAC	345-026-540-201	

Table 2-20. User Furnished Connector Details

Connector	No. of Pins	Connector Type	Vendor	Vendor Part No.	Intel Part No.
iSBX Multi-Module Conn J3/J4	44	Soldered PCB	VIKING	293-001	
	36	Soldered PCB	VIKING	292-001	iSBC 960-5
Multibus Conn. P1	43/86	Soldered PCB Mount	ELFAB VIKING	BS1562043PBB 2KH43/9AMK12	102247-001
		Wire Wrap No Ears	EDAC ELFAB	337-086-0540-201 BW1562D43PBB	102248-001
		Wire Wrap With .128 Mounting Holes	EDAC ELFAB	337-086-540-202 BW1562A43PBB	102273-001
Aux. Conn. P2	30/60	Wire Wrap	EDAC ELFAB	345-060-520-202 BW1020A30PBB	102238-001
		With Mounting Holes (.128)	TI VIKING	H421121-30 3KH30/9JNK	
		Wire Wrap No Ear	EDAC ELFAB	345-060-540-201 BW1020D30PBB	102241-001

Notes:
1. Connector heights are not guaranteed to conform to OEM packaging equipment.
2. Wirewrap pin heights are not guaranteed to conform to OEM packaging equipment.
3. Connector numbering convention may not agree with board connector numbers.

Table 2-21. Connector J1 Pin Assignments

Pin No.	Function	Pin No.	Function
1	Ground	2	Port B bit 7
3	Ground	4	Port B bit 6
5	Ground	6	Port B bit 5
7	Ground	8	Port B bit 4
9	Ground	10	Port B bit 3
11	Ground	12	Port B bit 2
13	Ground	14	Port B bit 1
15	Ground	16	Port B bit 0
17	Ground	18	Port C bit 3
19	Ground	20	Port C bit 2
21	Ground	22	Port C bit 1
23	Ground	24	Port C bit 0
25	Ground	26	Port C bit 4
27	Ground	28	Port C bit 5
29	Ground	30	Port C bit 6
31	Ground	32	Port C bit 7
33	Ground	34	Port A bit 7
35	Ground	36	Port A bit 6
37	Ground	38	Port A bit 5
39	Ground	40	Port A bit 4
41	Ground	42	Port A bit 3
43	Ground	44	Port A bit 2
45	Ground	46	Port A bit 1
47	Ground	48	Port A bit 0
49	Ground	50	EXT INTR0/

Notes: 1. All odd-numbered pins are on component side of board. Pin 1 is the right-most pin when viewed from the component side of the board with the extractors at the top.

2. Cable connector pin numbering convention may not agree with the board connector pin numbering convention.

Table 2-22. Bulk Cable Types

Flat cable, 50 conductor w/o ground plane	3M	3306-50
Flat cable, 50 conductor with ground plane	3M	3380-50
Woven cable, 25 pair,	3M	3321-25

Table 2-23. Connector J2 Pin Assignments

Pin No.	Signal	PCI Function
J2 - 1	Not Used	--------
J2 - 2	Chassis GND	Protective Ground
J2 - 3	Not Used	--------
J2 - 4	Receive Data	RxD Input
J2 - 5	See J2 - 26	See J2 - 26
J2 - 6	Transmit Data	TxD Output
J2 - 7	External Clock	TxC/RxC Input
J2 - 8	Clear To Send	CTS/ Input
J2 - 9	Not Used	--------
J2 - 10	Request To Send	RTS/ Output
J2 - 11	Not Used	--------
J2 - 12	Data Terminal Ready	DTR/ Output
J2 - 13	Data Set Ready	DSR/ Input
J2 - 14	Ground	GND
J2 - 15	Not Used	--------
J2 - 16	Not Used	--------
J2 - 17	Not Used	--------
J2 - 18	Not Used	--------
J2 - 19	-12 Vdc	-12 Vdc Output
J2 - 20	Not Used	--------
J2 - 21	See J2 - 26	See J2 - 26
J2 - 22	+12 Vdc	+12 Vdc Output
J2 - 23	+5 Vdc	+5 Vdc Output
J2 - 24	Not Used	--------
J2 - 25	Ground	GND
J2 - 26	Secondary TxD or Clock Out	STxD or TxC/TxD

Notes: 1. Odd numbered pins are on component side of board; even pins on trace side.
2. Cable connector numbering convention may not correspond with J2 numbering.

Table 2-24. RS232 Signals Pin Correspondence

PC Conn. J2	RS232 Conn.	PC Conn. J2	RS232 Conn.
1	14	14	7
2	1	15	21
3	15	16	8
4	2	17	22
5	16	18	9
6	3	19	23
7	17	20	10
8	4	21	24
9	18	22	11
10	5	23	25
11	19	24	12
12	6	25	N/C
13	20	26	13

***

CHAPTER 3. PROGRAMMING INFORMATION

3.1 INTRODUCTION

Several Intel programmable devices reside on the iSBC 86/05 board. This chapter provides programming information for these devices and gives typical examples for most applications. Memory and I/O addressing are provided in tabular form, for quick reference.

3.2 MEMORY ADDRESSING

The iSBC 86/05 board may accommodate up to 64K bytes of on-board ROM. Four sockets are provided for ROM devices. The amount of on-board ROM may be doubled to 128K bytes by adding the optional iSBC 341 ROM Expansion Module (refer to section 2.5.1). Table 3-1 provides the addressing for the various ROM configurations possible on the iSBC 86/05 board using the default page select configuration.

Assuming default page select configuration, the configuration for 2716 devices (2K X 8) indicates an on-board ROM address range from FE000 to FFFFF (hexadecimal). However the address range can be moved, refer to Table 2-3. In the maximum configuration, without the iSBC 341 ROM Expansion Module, the address range would be from F0000 to FFFFF.

Table 3-1. On-Board ROM Addresses

Device Type & Size	Address Range	Total Space
2716 (2K X 8)	FE000 - FFFFF	8K
2732 (4K X 8)	FC000 - FFFFF	16K
2764 (8K X 8)	F8000 - FFFFF	32K
27128 (16K X 8)	F0000 - FFFFF	64K
ROM Addresses With iSBC 341 ROM Module		
2716 (2K X 8)	FC000 - FFFFF	16K
2732 (4K X 8)	F8000 - FFFFF	32K
2764 (8K X 8)	F0000 - FFFFF	64K
27128 (16K X 8)	E0000 - FFFFF	128K
Note: Device sizes cannot be mixed.		

In the factory configuration, 8K bytes of RAM reside on-board. RAM addressing in the default factory configuration is assigned from 0000 to 1FFF (hexadecimal). With the optional iSBC 302 RAM Expansion Module installed, RAM addressing becomes 0000 - 3FFF. Table 3-2 summarizes the default on-board RAM addressing.

Table 3-2. On-Board RAM Addresses

Configuration	Addresses	Total Size
Factory Default	0 - 1FFF	8K
With iSBC 302	0 - 3FFF	16K

If non-existent system memory is addressed, instruction execution will stop unless the fail-safe timer is enabled. If the fail-safe timer is enabled, it generates an Acknowledge signal after approximately 10 milliseconds, allowing processing to resume. For failsafe timer jumper information, refer to section 2.4.2.2.

When the CPU is addressing on-board memory, an internal PROM or RAM Acknowledge signal is automatically generated. When the CPU is addressing off-board system memory via the Multibus lines, the CPU must first gain control of the Multibus lines and, after the Memory Read or Memory Write command is given, it must execute wait states until a Transfer Acknowledge signal is received from the addressed memory.

3.3 I/O ADDRESSING

The on-board 8086 CPU communicates with the programmable devices through a sequence of I/O read and I/O write commands. Each device has a specific fixed (dedicated) address, or group of addresses, through which commands and or data are issued or accepted. All of the fixed on-board I/O addresses are listed in Table 3-3. In addition to the board's programmable I/O devices, the iSBX Multimodules have specific addresses assigned to them. The iSBX I/O addresses are listed in the Table 3-4.

3.4 SYSTEM INITIALIZATION

When power is initially applied to the board, the reset signal (RESET) is automatically generated by the 8284A Clock Generator/Driver. This clears the 8086 internal counters, instruction registers, and the interrupt enable circuitry. The first instruction is then executed from memory location FFFF0. This location normally contains a JMP instruction which directs the processor to the actual program beginning.

Table 3-3. I/O Port Addresses

Address	Device	Input Function	Output Function
C0 or C4	8259A PIC	ICW1,OCW2,OCW3	Status & Poll
C2 or C6	8259A PIC	ICW2,ICW3,ICW4,OCW1 Mask	OCW1 Mask
C8	8255A PPI	Port A	Port A
CA	8255A PPI	Port B	Port B
CC	8255A PPI	Port C	Port C
CE	8255A PPI	Control Word	None
D0	8253 PIT	Counter 0	Counter 0
D2	8253 PIT	Counter 1	Counter 1
D4	8253 PIT	Counter 2	Counter 2
D6	8253 PIT	Control Word	None
D8 or DC	8251A PCI	Data	Data
DA or DE	8251A PCI	Mode or Command Word	Status

Table 3-4. iSBX™ Multimodule™ Connector I/O Port Addresses

I/O Port Addresses	iSBX™ Connector	Function Performed
80, 82, 84, 86, 88, 8A, 8C, 8E	J4	Read/Write low byte (both 8-bit and 16-bit boards), or word transfer (16-bit boards only). Activates MCS0/ to Multimodule boards.
81, 83, 85, 87, 89, 8B, 8D, 8F	J4	Read/Write high byte transfer (16-bit boards only). Activates MCS1/ to Multimodule boards.
90, 92, 94, 96, 98, 9A, 9C, 9E	J4	Read/Write Byte transfer (8-bit boards only). Activates MCS1/ to Multimodule boards.
A0, A2, A4, A6, A8, AA, AC, AE	J3	Read/Write low byte transfer (both 8-bit and 16- bit boards), or word transfer (16-bit boards only). Activates MCS0/ to Multimodule boards.
A1, A3, A5, A7, A9, AB, AD, AF	J3	Read/Write high byte transfer (16-bit boards only). Activates MCS1/ to Multimodule boards.
B0, B2, B4, B6, B8, BA, BC, BE	J3	Read/Write Byte transfer (8-bit boards only). Activates MCS1/ to Multimodule boards.

The RESET signal also is routed to all other iSBC boards in the system (as INIT/) via Multibus line P1 - 14. On-board, the RESET signal is routed to the 8255A parallel interface, the 8251A serial interface, the interrupt acknowledge circuitry, and the iSBX connectors. The RESET signal causes:

a. The 8251A serial interface device to idle and wait for a set of command words; and

b. The 8255A parallel ports to enter mode 0, input.

The reset/initialize signal can also be generated by an auxiliary reset switch, such as on a system front panel. This switch should be connected to P2 - 38 (AUX RESET/) on the auxiliary connector or alternatively to the Multibus INIT/ line.

Another switch may be used to generate an on-board only RESET. This switch should be connected to P2 - 36 (BD RESET/) on the auxiliary connector.

3.5 8253 INTERVAL TIMER PROGRAMMING

The 8253A Programmable Interval Timer (PIT) includes three independently controlled counters which are used for on-board I/O timing and CPU interrupts. Each counter has its own input and output and can be programmed to operate in one of six different modes. In addition, the iSBC 86/05 board interval timer configuration provides several jumper selectable clock rates which can be used for counter inputs. The output from counter 0 is jumpered to IR2 on the interrupt controller so it can be used as the CPU interrupt interval timer. It also could be jumpered as an input to counter 1. The output from counter 1 could be routed either to the interrupt matrix or off-board via the parallel interface. The output from counter 2 is used exclusively for the baud rate timer, clocking the 8251A serial interface controller.

Each counter of the PIT is individually programmed by writing a control word to the I/O port address.

Jumper configurations for the PIT clock inputs are summarized in section 2.5.2. In the default configuration, the following frequencies are jumpered to the PIT:

1.23 MHz to CLK 0 Input
153.60 KHz to CLK 1 Input
1.23 MHz to CLK 2 Input

Sections 3.5.1 through 3.5.7 describe interval timer mode control, addressing, initialization, and operation as implemented on the iSBC 86/05 board.

3.5.1 ADDRESSING

As listed in Table 3-5 the PIT is accessed using the following four I/O addresses: D0, D2, D4, D6. Addresses D0, D2 and D4 are used in loading and reading the count that is latched in Counters 0, 1, and 2, respectively. Address D6 is used in writing the mode control word to the desired counter.

Table 3-5. PIT Register Address

CS/	RD/	WR/	A_1	A_0	Activity	I/O Address (hex)
0	1	0	0	0	Load Counter No. 0	D0
0	1	0	0	1	Load Counter No. 1	D2
0	1	0	1	0	Load Counter No. 2	D4
0	1	0	1	1	Write Mode Word	D6
0	0	1	0	0	Read Counter No. 0	D0
0	0	1	0	1	Read Counter No. 1	D2
0	0	1	1	0	Read Counter No. 2	D4
0	0	1	1	1	No-Operation 3-State	D6
1	X	X	X	X	Disable 3-State	--
0	1	1	X	X	No-Operation 3-State	--

Note: X = Irrelevant Bit

3.5.2 MODE CONTROL WORD AND COUNT

Before you can use one or all of the counters, they must be programmed for mode and count. The mode can be programmed at any time with the Mode Control Word (Figure 3-1). This word specifies which counter is selected, what its mode is, and the type of count byte(s) which will subsequently be sent.

The mode control word (Figure 3-1) does the following:

a. Selects counter to be loaded.

b. Selects counter operating mode.

c. Selects one of the following four counter read/load functions:

   (1) Counter latch (for stable read operation).

   (2) Read or load most-significant byte only.

   (3) Read or load least-significant byte only.

   (4) Read or load least-significant byte first, then most-significant byte.

d. Sets counter for either binary or BCD count.

Each counter must be initialized prior its use. Initialization is described in section 3.5.3.

3.5.3 INITIALIZATION

To initialize the PIT, perform the following:

a. Write a mode control word to the control register for each counter to be used. Note that all mode control words are written to Port D6, because the mode control word must specify the counter being programmed. (Refer to Figure 3-1). Table 3-6 provides a sample routine for writing a mode control word to all three counters.

Table 3-6. Typical PIT Control Word Subroutine

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

        PUBLIC   INTTMR

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

        END
```

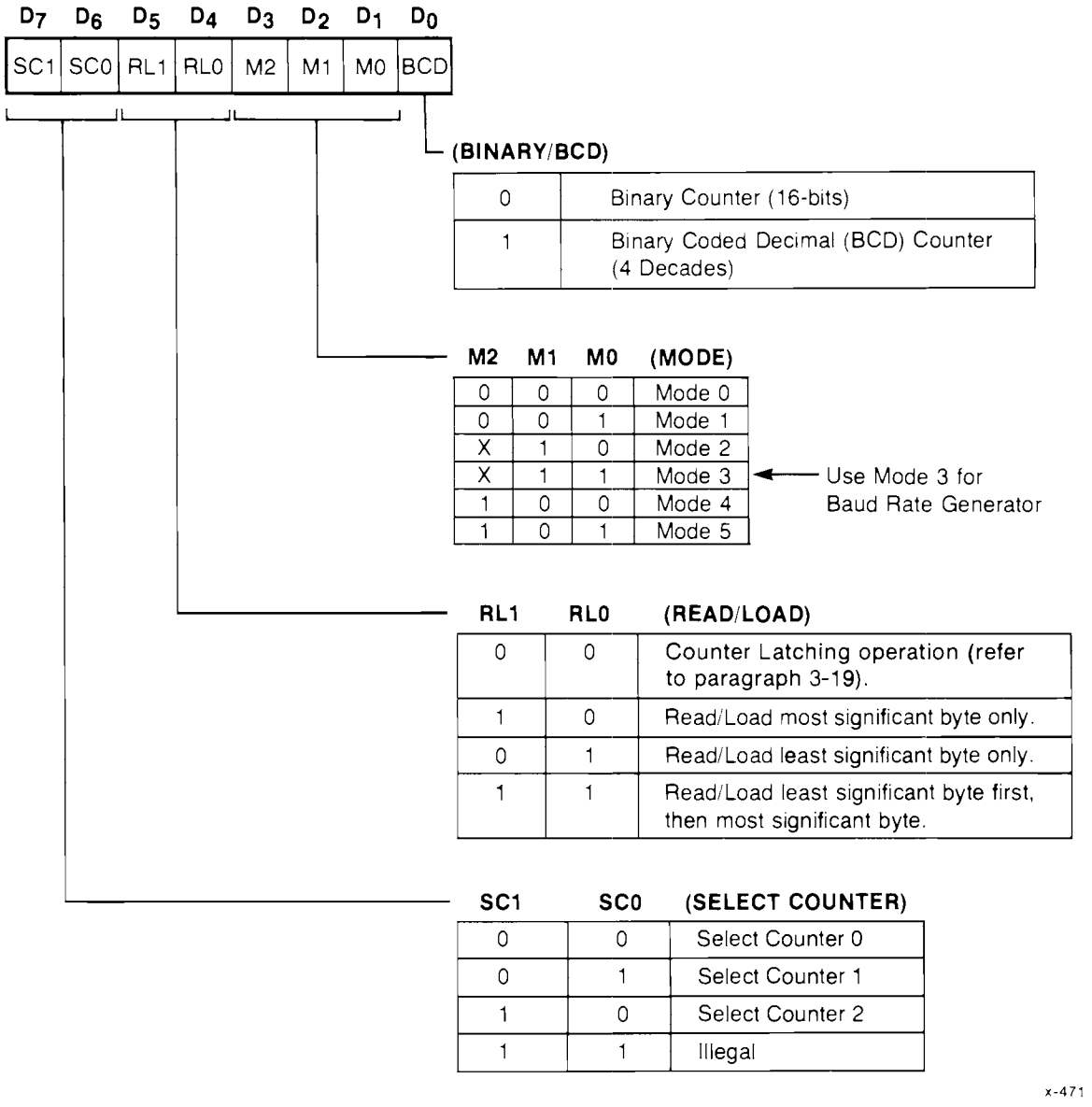


Figure 3-1. PIT Mode Control Word Format

b. Load the down-count number into the counter. The mode control word defines the number of bytes to be sent; either one or two bytes. Assuming the mode control word has selected a 2-byte load, load least-significant byte of the number into Counter 0 at port D0. Table 3-7 provides a sample subroutine for loading a 2-byte count value.

Table 3-7. Typical PIT Counter Value Load Subroutine

```
;LOAD0 LOADS COUNTER 0 FROM CX, CH IS MSB, CL IS LSB.
;USES-D,E:  DESTROYS-AL.

        PUBLIC   LOAD0

LOAD0:  MOV      AL,CL          ;GET LSB
        OUT      0D0H,AL
        MOV      AL,CH          ;GET MSB
        OUT      0D0H,AL
        RET

        END
```

c. Load most-significant byte of count into Counter 0 at port D0.

NOTE

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

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

To read the count of a particular counter, proceed as follows (a typical counter read subroutine is given in Table 3-8):

a. Write counter register latch control word (Figure 3-2) to port D6. Control word specifies desired counter and selects counter latching operation. Bits D0 - D3 are irrelevant.

b. Perform a read operation of desired counter, refer to Table 3-5 for counter addresses.

Table 3-8. Typical PIT Counter Read Subroutine

```
;READ1 READS COUNTER 1 ON-THE-FLY INTO CX.  MSB IN CH, LSB IN CL.
;DESTROYS-AL,CX.

          PUBLIC    READ1

READ1:    MOV       AL,40H          ;MODE WORD FOR LATCHING COUNTER 1 VALUE
          OUT       0D6H,AL
          IN        AL,0D2H         ;LSB OF COUNTER
          MOV       CL,AL
          IN        AL,0D2H         ;MSB OF COUNTER
          MOV       CH,AL
          RET

          END
```

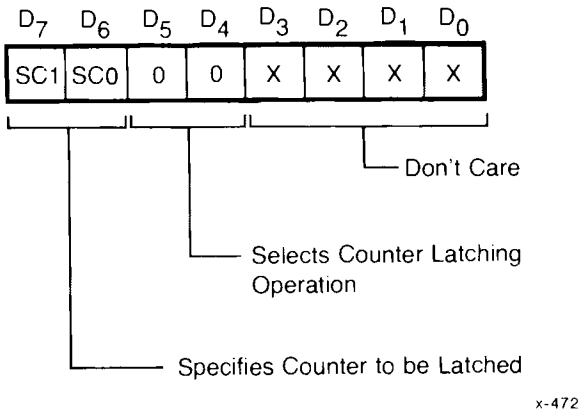


Figure 3-2. PIT Counter Register Latch Control Word Format

NOTE

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

Notice that the Read/Load sequence specified by the Mode Control Word with the RL bits (Figure 3-1) must be followed when these bytes are sent to the PIT. These bytes do not necessarily have to follow the associated Mode Control Word. For example, if you select RL1 = 0 and RL0 = 1, indicating Read/Load least significant byte only, your desired count must be placed in the least significant count byte. This is especially

important when using counts which require two bytes. If this procedure is followed for each counter, the PIT can be programmed in any convenient sequence. For example, Mode control words for each counter can be loaded-first followed by the count byte. Figure 3-3 illustrates a typical PIT programming sequence.

PROGRAMMING FORMAT

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

ALTERNATE PROGRAMMING FORMAT

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

Figure 3-3. PIT Programming Sequence Examples

Because the PIT counters are down-counters, the values loaded into the count registers are decremented. Loading all zeros into a count register results in the maximum count possible: 2^{16} for binary numbers and 10^4 for BCD numbers.

The count mode selected in the control word controls the counter output. As shown in Figure 3-1, the PIT device can operate in any of six modes:

a. Mode 0: Interrupt on terminal count. In this mode, Counters 1 and 2 can be used for auxiliary functions such as generating real-time interrupts. After the count value is loaded into the count register, the counter output goes low and remains low until the terminal count is reached. The output then goes high until either the count register or the mode control register is reloaded.

b. Mode 1: Programmable one-shot. In this mode, the output of Counter 0 and or Counter 1 will go low on the count following the rising edge of the GATE input which can be jumpered to be controlled by an on-board parallel Port. The output will go high on terminal count. If a new count value is loaded while the output is low, it will not affect the duration of the one-shot pulse until the succeeding trigger. The current count can be read at any time without affecting the one-shot pulse. The one-shot is retriggerable, hence the output will remain low for the full count after any rising edge of the gate input.

c. Mode 2: Rate generator. In this mode, the counter generates output pulses at a programmed interval. The output pulse is low for one period of the clock input. The interval from one output pulse to the next equals the number of input counts in the count register. If the count register is reloaded between output pulses, the present interval is not affected but the subsequent interval will reflect the new value. The gate input, when low, forces the output to go high. When the gate input goes high, the counter starts from the initial count. Thus, the gate input may be used to synchronize the counter. When Mode 2 is set, the output will remain high until after the count register is loaded; thus, the count can be synchronized by software.

d. Mode 3: Square wave generator/Frequency Divider. Mode 3 is the primary operating mode for all three counters on the 8253. In this mode, the counter output remains high until one half of the count value in the count register has been decremented (for even numbers). The output then goes low for the other half of the count. If the value in the count register is odd, the counter output is high for (N+1)/2 counts, and low for (N-1)/2 counts.

e. Mode 4: Software triggered strobe. After this mode is set, the output goes high. When the count is loaded, the counter begins counting. On terminal count, the output will go low for one input clock period and then go high again. If the count register is reloaded between output pulses, the present count will not be affected, but the subsequent period will reflect the new value. The count is inhibited while the gate input is low. Reloading the count register restarts the counting for the new value.

f. Mode 5: Hardware triggered strobe. The counter starts counting on the rising edge of the gate input. The output goes low for one clock period when the terminal count is reached. The counter is retriggerable; the output will not go low until the full count after the rising edge of the gate input.

Table 3-9 provides a summary of the counter operation versus the gate inputs.

Table 3-9. PIT Counter Operation Vs. Gate Inputs

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

3.5.4 OPERATION

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

3.5.4.1 Counter Read

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

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

NOTE

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

3.5.4.2 Clock Frequency/Divide Ratio Selection

Table 2-6 lists the default and optional timer input frequencies to Counters 0 through 2. The timer input frequencies are divided by the counters to generate the OUT0 Interrupt Clock (Counter 0), OUT1 Clock (Counter 1), and the 8251A Baud Rate Clock (Counter 2).

Each counter must be programmed with a down-count number, or count value N. When count value N is loaded into a counter, it becomes the clock divisor. To derive N for either synchronous or asynchronous RS 232C operation, use the procedures described in the following paragraphs.

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

$$N = C/B$$

where N is the count value,
B is the desired Baud rate, and
C is 1.23 MHz, the input clock frequency.
Thus, for a 4800 Baud rate, the required count value (N) is:

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

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

Asynchronous Mode. In the asynchronous mode, the TXC and/or RXC rates equal the Baud rate times one of the following multipliers: 16 or 64. Therefore, the count value is determined by:

$$N = C/BM$$

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

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

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

If the binary equivalent of count value N = 16 is loaded into Counter 2, then the output frequency is 4800 X 16 Hz, which is the desired clock rate for asynchronous mode operation. Count values (N) versus rate multiplier (M) for each Baud rate are listed in Table 3-10. A simplified version of this table, with hex notations, is provided in Table 3-11.

NOTE

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

Table 3-10. Count Values And Rate Multipliers

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

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

Table 3-11. PIT Baud Rate Factors

Desired Baud Rate	Baud Rate Multiplier			Frequency in KHz to 8251A	Count Value Loaded into 8253	
	Synchronous	Asynchronous				
	1	16	64		DEC	HEX
9600	X			9600	128	80
		X		153.6	8	8
4800	X			4800	256	100
		X		76.8	16	16
			X	307.2	4	4
2400	X			2400	512	200
		X		38.4	32	20
			X	153.6	8	8
1200	X			1200	1024	400
		X		19.2	64	40
			X	76.8	16	16
600	X				2048	800
		X		9.6	128	80
			X	38.4	32	20
300	X				4096	1000
		X		4.8	256	100
			X	19.2	64	40
150	X				8192	2000
		X		2.4	512	200
			X	9.6	128	80
110	X				11171	26A3
		X		1.76	698	26A
			X	7.04	175	AF
75	X				16384	4000
		X		12	1024	400
			X	48	256	100

Isosynchronous Mode. In some cases it is desirable to operate in a hybrid mode which involves transmitting data with the asynchronous format using a synchronous modem. This occurs when an increase in operating speed is required without a change in the basic protocol of the system. This hybrid technique is known as isosynchronous and involves the generation of the start and stop bits associated with the asynchronous format, while still using the modem clock for bit synchronization.

In the asynchronous mode the PCI can be programmed to accept clock rates at 16 or 64 times the required baud rate. Isosynchronous operation is a special case of asynchronous with the multiplier rate programmed as one instead of 16 or 64.

3.5.4.3 Rate Generator/Interval Timer

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

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

	Rate Generator (frequency)		Real-Time Interrupt (interval)	
	Minimum	Maximum	Minimum	Maximum
Single Timer[1] (Counter 0)	18.75 Hz	614.4 kHz	1.63 usec	53.3 msec
Single Timer[2] (Counter 1)	2.344 Hz	76.8 kHz	13 usec	426.67 msec
Dual Timer[3] (0 and 1 in Series)	0.00029 Hz	307.2 kHz	3.26 usec	58.25 minutes

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

3.5.4.4 Interrupt Timer

To program an interval timer to interrupt upon reaching terminal count, program the appropriate timer for the correct operating mode (Mode 0) in the control word. Then load the count value (N), which is derived by:

$$N = TC$$

where N is the count value for Counter 1,
T is the desired interrupt time interval in seconds, and
C is the internal clock frequency (Hz).

Table 3-13 shows the count value (N) required for several time intervals (T) that can be generated for Counters 0 and 1 when the input clock is 1.23M Hz.

Table 3-13. PIT Timer Intervals & Timer Counts

T	N*
10 usec	12
100 usec	123
1 msec	1229
10 msec	12288
50 msec	61440

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

3.6 8251A PCI PROGRAMMING

The PCI converts parallel output data into serial data, and can be programmed to support virtually any data format (including IBM Bi-Sync) for half- or full-duplex operation. The PCI also converts serial input data into parallel data format.

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

3.6.1 MODE INSTRUCTION FORMAT

The Mode instruction word defines the general characteristics of the PCI and must follow a reset operation. Once the Mode instruction word has been written into the PCI, sync characters or command instructions may be loaded, as required by the mode selected. The mode control word is written to the PCI control port. The Mode instruction word defines the following:

a. For Sync Mode:

(1) Character length
(2) Parity enable
(3) Even/odd parity generation and check
(4) External sync detect
(5) Single or double character sync

b. For Async Mode:

(1) Baud rate factor (X16 or X64)
(2) Character length
(3) Parity enable

Instruction word and data transmission formats for synchronous and asynchronous modes are shown in Figures 3-4 through 3-8.

3.6.2 SYNC CHARACTERS

Sync characters are written to the PCI, following the mode word, in the synchronous mode only. The PCI can be programmed for either one or two sync characters; the format of the sync characters is at the option of the programmer. Sync characters are written to the PCI control port following the mode word, but before any control words.

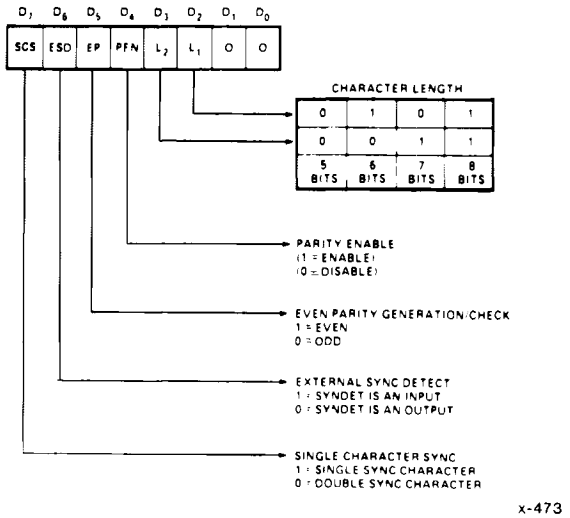


Figure 3-4. PCI Synchronous Mode Instruction Word Format

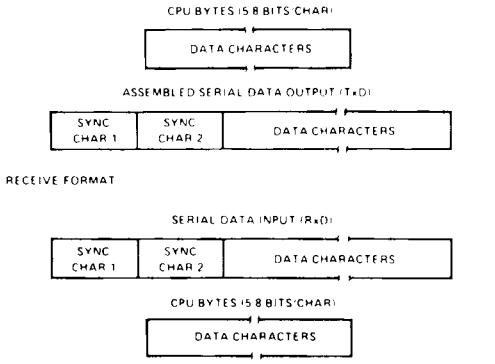


Figure 3-5. PCI Synchronous Mode Transmission Format

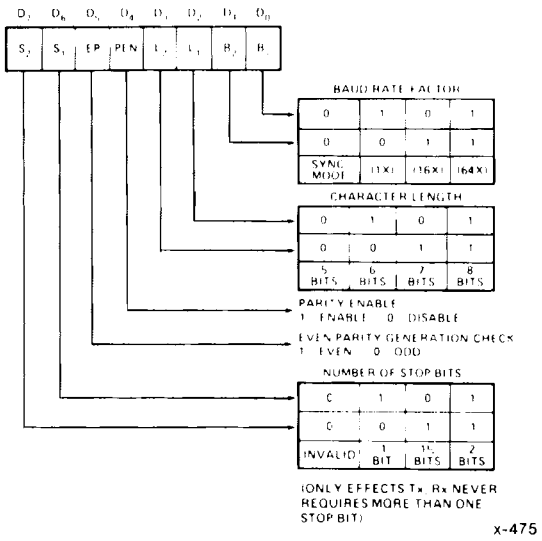
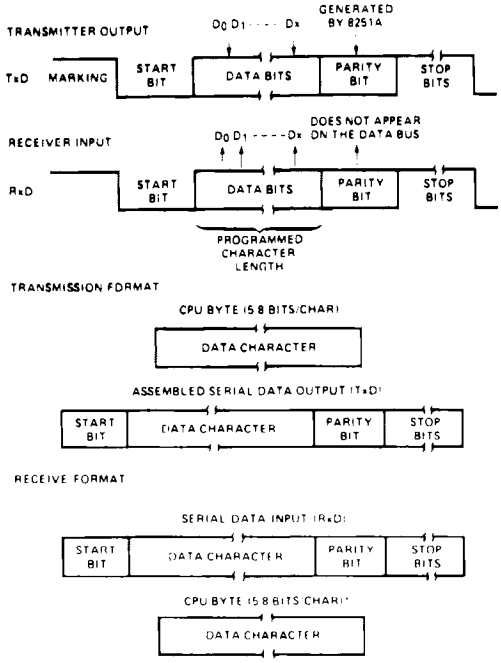


Figure 3-6. PCI Asynchronous Mode Instruction Word Format



x-476

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

Figure 3-7. PCI Asynchronous Mode Transmission Format

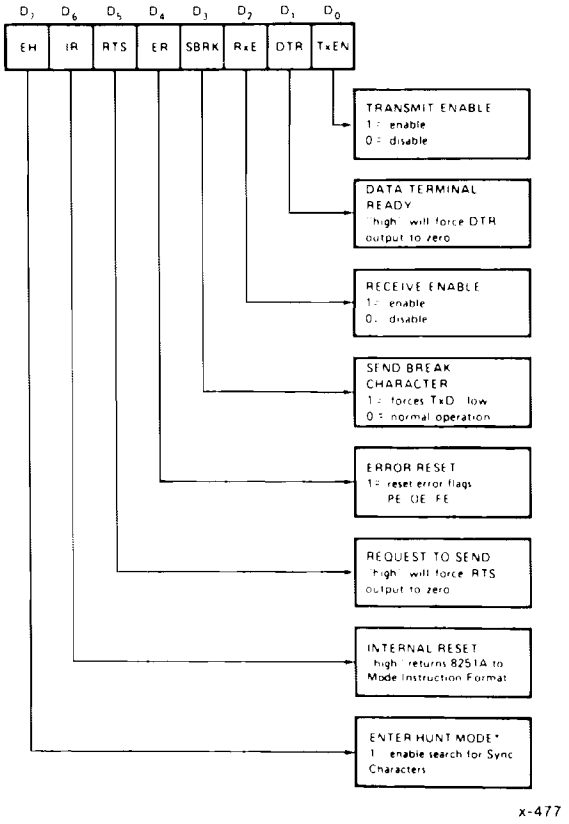


Figure 3-8. PCI Command Instruction Word Format

3.6.3 COMMAND INSTRUCTION FORMAT

The Command instruction word shown in Figure 3-8 controls the operation of the addressed PCI. A Command instruction must follow the mode and/or sync words. Once the Command instruction is written, data can be transmitted or received by the PCI.

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

After initialization, always read the PCI status and check for the TxRDY bit prior to writing either data or command words to the PCI. This ensures that any prior input is not overwritten and lost. Note that issuing a Command instruction with bit 6 (IR) set will return the PCI to the Mode instruction format.

3.6.4 RESET

To change the Mode instruction word, the PCI must receive a Reset command. This can be either a hardware reset or a reset generated by bit 6 of the Command instruction. The next word written to the PCI after a Reset command is assumed to be a Mode instruction. Similarly, for sync mode, the next word after a Mode instruction is assumed to be one or more sync characters. All control words written into the PCI after the Mode instruction (and/or the sync character) are assumed to be Command instructions. If you are going to reset the PCI with a command instruction and it is in an unknown state, you should first write three zeros to the control port. This executes the worst-case initialization sequence of sync mode with two sync characters and ensures that the reset command is accepted.

3.6.5 ADDRESSING

The PCI device uses two consecutive pairs of addresses. The lower of the two addresses in each pair is used to read and write I/O data; the upper address in each pair is used to write mode and command words, write sync characters, and to read the PCI status. (Refer to Table 3-14).

Table 3-14. PCI Address Assignments

I/O Address (hexadecimal)	Command	Function	Direction
DA or DE	OUTPUT	CONTROL	CPU to PCI
D8 or DC	OUTPUT	DATA	CPU to PCI
DA or DE	INPUT	STATUS	PCI to CPU
D8 OR DC	INPUT	DATA	PCI to CPU

3.6.6 INITIALIZATION

A typical PCI initialization and I/O data sequence is presented in Figure 3-9. An example of a typical Initialization subroutine is presented in Table 3-15. Initialize the PCI device as follows:

a. Disable PCI interrupt.

b. Write four bytes of zeros to the PCI command port. This inserts zeros into the registers of the PCI enabling a subsequent RESET command. (Delay for at least 6.5 us between each write.)

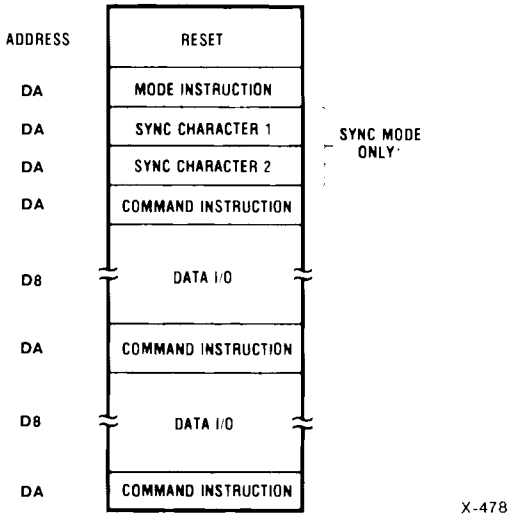
c. Issue a PCI reset command.

d. Delay for at least 6.5 us.

e. Write PCI mode instruction word. One function of the mode word is to specify synchronous or asynchronous operation. If synchronous mode is selected, write one or two sync characters as required.

f. Delay for at least 6.5 us.

g. Write PCI command instruction word.



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

Figure 3-9. Typical PCI Initialization & Data I/O Sequence

Table 3-15. Typical PCI Initialization Subroutine

TYPICAL DELAY SUBROUTINE

```
;Delay gets count value from stack
;Destroys -BX

     PUBLIC DELAY

DELAY: POP  BX    ;Get count value from stack
     PUSH   CX    ;Save CX register
     PUSHF        ;Save flags
     MOV    CX,BX      ;Move count to CX register
TAG1:  LOOP TAG1 ;Delay loop
     POPF         ;Restore flags
     POP    CX    ;Restore CX register
     RET

     END



;Uses DELAY
;PCI mode word initializes for 2 stop bits, no parity,
;7 bit character length, and X16 baud rate factor.  PCI command word
;initializes for TX and RX enable, error reset, DTR and RTS low.
;Destroys- AX, BX, CX, flags

     EXTRN  DELAY

     MOV    BX,03H     ;Delay count value
     MOV    AL,00H     ;Do nothing PCI control word
     MOV    CX,04H     ;Loop value to output four zeros
TAG2:  OUT  0DAH,AL    ;to the PCI to insure that it
     PUSH   BX    ;is ready to receive a mode word.
     CALL   DELAY
     LOOP   TAG2

     MOV    AL,40H     ;Reset PCI
     OUT    0DAH,AL    ;PCI write
     PUSH   BX    ;Delay
     CALL   DELAY
     MOV    AL,0CAH    ;PCI mode word
     OUT    0ADH,AL    ;PCI write
     PUSH   BX    ;Delay
     CALL   DELAY
     MOV    AL,037H    ;PCI command word
     OUT    0ADH,AL    ;PCI write
     RET

     END
```

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

First, reset the PCI device by writing a Command instruction to Port 00DA (or 00DE). The Command instruction must have bit 6 set (IR=1); all other bits are immaterial.

NOTE

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

Next write a Mode instruction word to the PCI. (See Figures 3-4 through 3-9.) A typical subroutine for writing Command instructions after initialization is given in Table 3-16.

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

Finally, write a Command instruction word to the PCI. Refer to Figure 3-8 and Table 3-16.

IMPORTANT: During initialization, the 8251A PCI requires a minimum recovery time of 6.5 microseconds (16 PCI clock cycles) between back-to-back writes in order to set up its internal registers. This recovery time can be satisfied by the CPU performing several dummy instructions between the back-to-back writes to the 8251A to create a minimum delay of 6.5 microseconds. The following example will create a delay of approximately 7.2 microseconds.

```
        MOV     AL,04EH     ;PCI MODE WORD
        OUT     0DAH,AL     ;FIRST PCI WRITE
        MOV     CX,3        ;DELAY
TAG:    LOOP    TAG         ;DELAY
        MOV     AL,037H     ;PCI COMMAND WORD
        OUT     0DAH,AL     ;SECOND PCI WRITE
```

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

3.6.7 OPERATION

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

NOTE

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

Prior to any operating change, a new command word must be written with the appropriate change in command bits. (Refer to Figure 3-8 and Table 3-16.)

Data Input/Output. For data receive or transmit operations, perform a read or write, respectively, to the PCI. Tables 3-17 and 3-18 provide examples of typical character read and write subroutines.

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

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

Table 3-16. Typical PCI Command Instruction Subroutine After Initialization

```
;CMD 2 OUTPUTS CONTROL WORD TO USART FROM AL REGISTER.
;USES-AL, STAT0; DESTROYS-NOTHING.

            PUBLIC    CMD2,51INT
            EXTRN     STAT0

CMD2:       PUSH      AX
            PUSHF
LP:         CALL      STAT0
            AND       AL,1              ;CHECK TXRDY
            JZ        LP                ;TXRDY MUST BE TRUE
            POPF
            POP       AX
51INT:      OUT       0DAH,AL           ;ENTER HERE FOR INITIALIZATION
            RET

            END
```

Table 3-17. Typical PCI Data Character Read Subroutine

```
;RX1 READS DATA CHARACTER FROM USART INTO REG AL.
;USES-STAT0; DESTROYS-AL, FLAGS.

            PUBLIC    RX1,RXA1
            EXTRN     STAT0

RX1:        CALL      STAT0
            AND       AL,2              ;CHECK FOR RXRDY TRUE
            JZ        RX1
RXA1:       IN        AL,0D8H           ;ENTER HERE IF RXRDY IS TRUE
            RET

            END
```

Table 3-18. Typical PCI Data Character Write Subroutine

```
;TX1 WRITES DATA CHARACTER FROM REG AL TO USART.
;USES-AL, STAT0; DESTROYS-FLAGS.

            PUBLIC    TX1,TXA1
            EXTRN     STAT0

TX1:        PUSH      AX
TX11:       CALL      STAT0
            AND       AL,1            ;CHECK FOR TXRDY TRUE
            JZ        TX11
            POP       AX
TXA1:       OUT       0D8H,AL         ;ENTER HERE IF TXRDY IS TRUE
            RET

            END
```

The TXRDY and RXRDY outputs of the PCI are available at the priority interrupt jumper matrix. If, for instance, TXRDY and RXRDY are input to the 8259A PIC, the PIC resolves the priority and interrupts the CPU. TXRDY and RXRDY are also available in the status word. (Refer to paragraph 3.7.1.)

Status Read. The CPU can determine the status of a serial I/O port by issuing an I/O Read Command to the upper port (00DA or 00DE) of the PCI. The format of the status word is shown in Figure 3-10. A typical status read subroutine is given in Table 3-19.

Table 3-19. Typical PCI Status Read Subroutine

```
;STAT0 READS STATUS FROM USART.
;DESTROYS-AL.

            PUBLIC    STAT0

STAT0:      IN        AL,0DEH         ;GET STATUS
            RET

            END
```

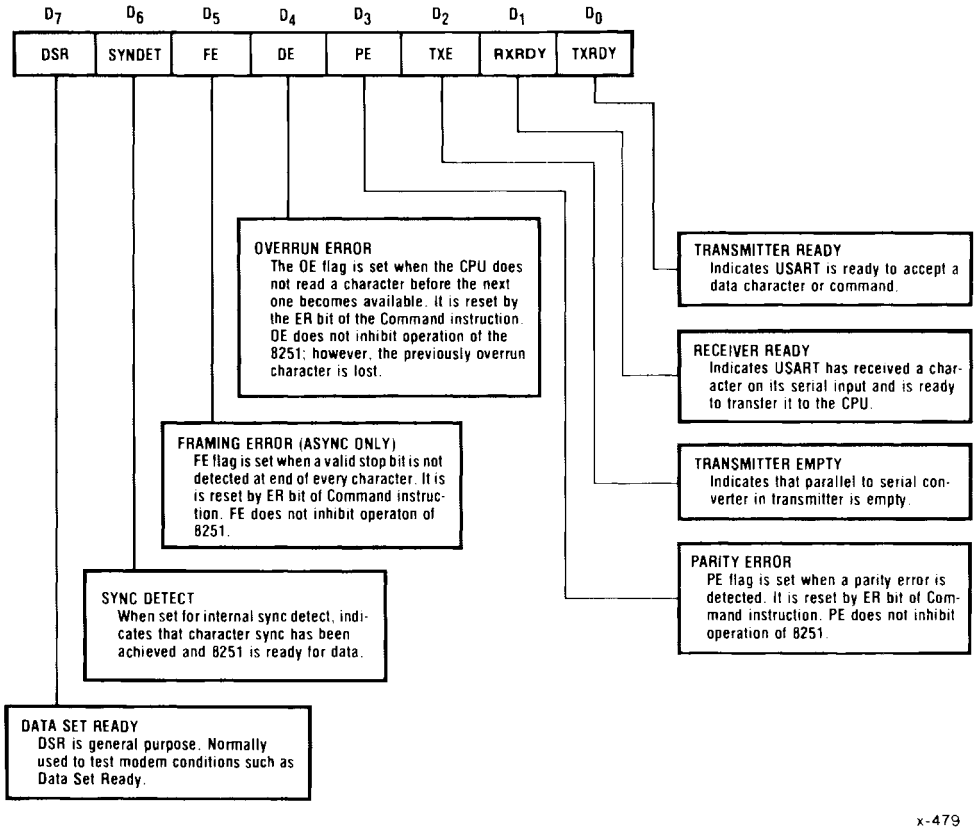


Figure 3-10. PCI Status Read Format

3.7 8255A PPI PROGRAMMING

The iSBC 86/05 board has a total of 24 parallel I/O lines, grouped into three Ports: A, B, and C. All lines exit the board via connector J1. One 8255A PPI device is used to control all three ports. Line identification is provided in Table 2-21.

Each of the three parallel I/O ports may be programmed independently. However, as implemented on the iSBC 86/05 board, some lines have restricted use in certain modes due to the configuration of the input and output devices. The modes allowed on the iSBC 86/05 board are listed in Table 3-20. Notice that each half of port C may be programmed independently.

Default jumpers set the port A bus transceivers to the output (transmit) mode. Optional jumper connections allow the bus transceivers to be set to either the input mode or a bit-programmable input/output mode. Refer to Table 2-10 for complete jumper information.

Ports B and C do not have bus transceivers installed at the factory. Line drivers or terminators can be installed for these ports as described in section 2.6.2.

In order to use any of the parallel port lines, the 8255A PPI device must first be initialized and programmed for the desired mode and direction of data flow. Sections 3.7.1 through 3.7.4 provide this information.

3.7.1 CONTROL WORD FORMAT

The control word format (shown in Figure 3-11) is used to initialize the PPI port. Group A (control word bits 3 through 6) defines the operating mode for Port A and the upper four bits of Port C. Group B (control word bits 0 thru 2) defines the operating mode for Port B and the lower four bits of Port C. (Refer to Table 3-21 for port identification). Bit 7 of the control word controls the mode set flag. Control words are sent to port CE (Table 3-21). There are restrictions associated with the use of certain ports in modes 1 and 2. Refer to Table 2-12 for restrictions and refer to the Intel Component Data Catalog for more information.

Table 3-20. Parallel Port Configuration

Port A	Mode 0, input Mode 0, output (latched) Mode 1, input (strobed) Mode 1, output (latched) Mode 2, bidirectional
Port B	Mode 0, input Mode 0, output (latched) Mode 1, input (strobed) Mode 1, output (latched)
Port C*	Mode 0, 8-bit input Mode 0, 8-bit output (latched) Mode 0, split (4-bit input, 4-bit output)
* Control mode may depend on mode of other ports; see Table 2-10.	

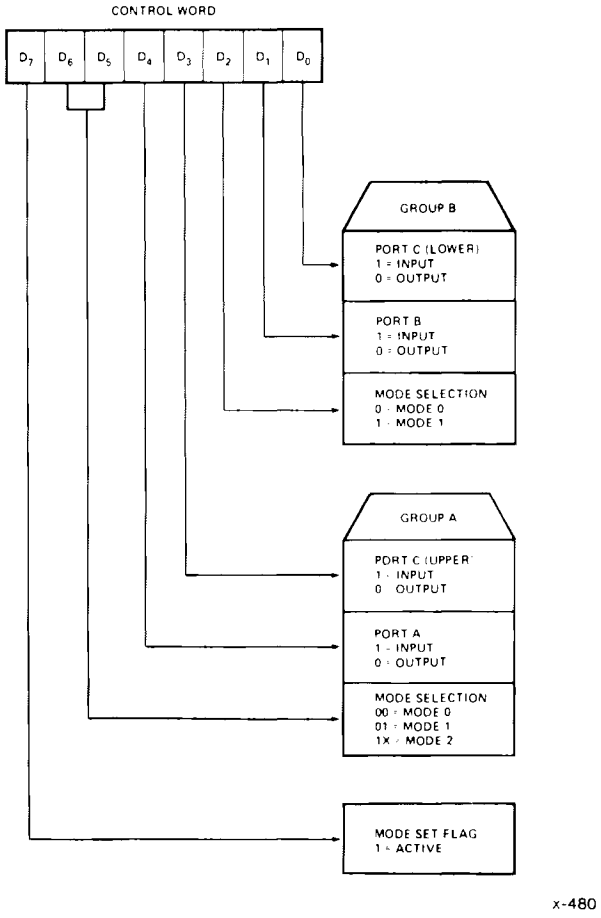


Figure 3-11. PPI Control Word Format

Table 3-21. Parallel Port I/O Addresses

8255A Device Port	Address (hexadecimal)
8255A Port (A)	C8
8255A Port (B)	CA
8255A Port (C)	CC
8255A Control	CE

3.7.2 ADDRESSING

The PPI uses four consecutive even addresses (00C8 through 00CE) for data transfer and for port control. (Refer to Table 3-21.)

3.7.3 INITIALIZATION

To initialize the PPI, write a control word to port 00CE. Refer to Figure 3-11 and Table 3-22 and assume that the control word is 92 (hexadecimal). This initializes the PPI as follows:

a. Mode Set Flag active

b. Port A (00C8) set to Mode 0 Input

c. Port C (00CC) upper set to Mode 0 Output

d. Port B (00CA) set to Mode 0 Input

e. Port C (00CC) lower set to Mode 0 Output

3.7.4 OPERATION

The primary considerations in determining how to operate each of the three I/O ports are:

a. Choice of operating mode (as defined in Table 3-20);

b. Direction of data flow (input, output or bidirectional), (see Table 3-25); and

c. Choice of driver/terminator networks.

After the PPI has been initialized, the operation is completed by simply performing a read or a write to the appropriate port. A typical read subroutine for Port A is given in Table 3-23.

A typical write subroutine for Port C is given in Table 3-24. As shown in Figure 3-12, any of the Port C bits can be selectively set or cleared by writing a control word to Port 00CE.

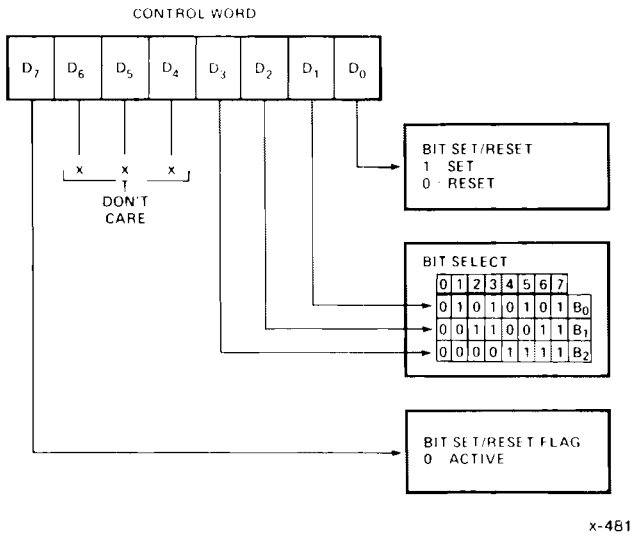


Figure 3-12. PPI Port C Bit Set/ Reset Control Word Format

3.7.4.1 Single Bit Set/Reset Feature

Any of the eight bits of Port C (board port CC) can be Set or Reset using a single output instruction (see Figure 3-12). This feature reduces software requirements in Control-based applications.

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output ports.

3.7.4.2 Mode Combinations

Table 3-25 summarizes the various mode combinations possible with ports A and B of each PPI, and indicates how each port C bit can be used. This table can serve as a useful starting point for selecting your particular configuration. Once you select the desired mode combination and the port C bit assignments are made, refer to the jumper configuration Table (2-10) for implementation details.

Table 3-22. Typical PPI Initialization Subroutine

```
;INTPAR INITIALIZES PARALLEL PORT MODES.
;DESTROYS-AL.

          PUBLIC    INTPAR

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

          END
```

Table 3-23. Typical PPI Port Read Subroutine

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

          AREAD

AREAD:    IN            AL,0C8H          :GET BYTE
          RET

          END
```

Table 3-24. Typical PPI Port Write Subroutine

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

          PUBLIC        COUT

COUT:     OUT           0CCH,AL          ;OUTPUT BYTE
          RET

          END
```

Table 3-25. Parallel I/O Interface Configurations

Configuration Number	PPI Port A (C8)	PPI Port B (CA)	PPI Port C (CC) Lower				PPI Port C (CC) Lower			
			C_0	C_1	C_2	C_3	C_4	C_5	C_6	C_7
1	Mode 0-In	Mode 0-I/O	-I/O-				-I/O-			
2	Mode 0-Out	Mode 0-I/O	-I/O-				-I/O-			
3	Mode 0-In	Mode 1-I/O	R	R	R	I	O	O	O	U
4	Mode 0-In	Mode 1-I/O	R	R	R	O	I	I	I	U
5	Mode 0-Out	Mode 1-I/O	R	R	R	I	O	O	O	U
6	Mode 0-Out	Mode 1-I/O	R	R	R	O	I	I	I	U
7	Mode 1-In	Mode 0-I/O	I	I	I	R	R	R	I	I
8	Mode 1-In	Mode 0-I/O	O	O	O	R	R	R	I	I
9	Mode 1-Out	Mode 0-I/O	I	I	I	R	O	O	R	R
10	Mode 1-Out	Mode 0-I/O	O	O	O	R	I	I	R	R
11	Mode 1-In	Mode 1-I/O	R	R	R	R	R	R	I	I
12	Mode 1-In	Mode 1-I/O	R	R	R	R	R	R	R	O
13	Mode 1-Out	Mode 1-I/O	R	R	R	R	I	I	R	R
14	Mode 1-Out	Mode 1-I/O	R	R	R	R	O	O	R	R
15	Mode 2-B	Mode 0-I/O	U	I	I	R	R	R	R	R
16	Mode 2-B	Mode 0-I/O	U	O	O	R	R	R	R	R
17	Mode 2-B	Mode 1-I/O	R	R	R	R	R	R	R	R

Notes:

I - Input
O - Output
I/O - Input or Output
B - Bidirectional
R - Reserved
U - Unused - Due to the jumper changes necessary to implement the functions of the reserved bits for this configuration and because there are only four lines available per device (driver for PPI output; terminator for PPI input), this bit is unused. However, it may be used to connect to the serial I/O interface or the Interval Timer.

3.8 8259A PIC PROGRAMMING

The 8259A PIC functions as the interrupt manager in a system environment. It accepts requests from the peripheral equipment, determines which of the incoming requests is of the highest importance (priority), ascertains whether the incoming request has a higher priority value than the level currently being serviced, and may issue an interrupt to the CPU based on this determination.

The on-board master 8259A PIC handles up to eight priority interrupts and has the capability of expanding the number of priority interrupts by cascading one or more of its interrupt input lines with slave 8259A PIC's. (Refer to paragraph 2.5.5.)

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

3.8.1 INTERRUPT PRIORITY MODES

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

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

3.8.1.1 Fully Nested Mode

In this mode, the PIC input signals are assigned a priority from 0 through 7. Interrupt IR0 has the highest priority and IR7 has the lowest priority. When an interrupt request is acknowledged by the CPU, the highest priority request is sent to the CPU by the PIC. Lower priority interrupts are inhibited; higher priority interrupts will be able to generate an interrupt that will be acknowledged if the CPU has enabled its own interrupt input through software. An End-Of-Interrupt (EOI) command from the CPU is required to reset the PIC for the next interrupt.

3.8.1.2 Special Fully Nested Mode

This mode must be used only when one or more PIC's are slaved to the master PIC, in which case the priority is resolved within both the master and the slave PIC's. The slave PIC's actually send the interrupt ID to the CPU. The master PIC arbitrates simultaneous interrupt requests from more than one slave. When PIC's are slaved to a master, the master must be operated in special fully nested mode and the slaves in the fully nested mode.

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

a. While an interrupt from a slave PIC is being serviced, that particular PIC is not locked out from the master PIC priority logic. That is, further interrupts of higher priority within this slave PIC will be recognized and the master PIC will initiate an interrupt request to the CPU.

b. When exiting the interrupt service routine, the software must check to determine if another interrupt is pending from the same slave PIC. This is done by sending an End-Of-Interrupt (EOI) command to the slave PIC and then reading its In-Service Register (ISR). If the IS register is clear (empty), an EOI command is sent to the master PIC. If the IS register is not clear (interrupt pending), no EOI command should be sent to the master PIC.

3.8.1.3 Automatic Rotating Mode

In this mode, the interrupt priority rotates. Once an interrupt on a given input is serviced, that interrupt assumes the lowest priority. Thus, if there are a number of simultaneous interrupts, the priority will rotate among the interrupts in numerical order. For example, if interrupts IR4 and IR6 request service simultaneously, IR4 will receive the highest priority. After service, the priority level rotates so the IR4 has the lowest priority and IR5 assumes the highest priority. In the worst case, seven other interrupts are serviced before IR4 again has the highest priority. Of course, if IR4 is the only request, it is serviced promptly. The priority shifts when the PIC receives an End-Of-Interrupt (EOI) command.

3.8.1.4 Specific Rotating Mode

In this mode, the software can change interrupt priority by specifying the bottom priority, which automatically sets the highest priority. For example, if IR5 is assigned the bottom priority, IR6 assumes the highest priority. In specific rotating mode, the priority can be rotated by writing a Specific Rotate at EOI (SEOI) command to the PIC. This command contains the BCD code of the interrupt being serviced; that interrupt is reset as the bottom priority. In addition, the bottom priority interrupt can be fixed at any time by writing a command word to the appropriate PIC.

3.8.1.5 Special Mask Mode

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

a. Write an End-Of-Interrupt (EOI) command which enables all interrupt levels that are not masked.

b. Set the Special Mask Mode which enables all levels that are not masked except for the levels being serviced.

Normally, when an interrupt level is being serviced, the lower priority interrupts are disabled. However, it is possible to enable the lower priority interrupts with the Special Mask Mode. In this mode, the lower priority lines are enabled until the Special Mask Mode is reset. Higher priorities are not affected.

3.8.1.6 Poll Mode

One way to use this mode is with the CPU internal Interrupt Enable flip-flop cleared (interrupts disabled) and a software subroutine used to initiate a Poll command. In the Poll Mode, the addressed PIC treats an I/O Read Command as an interrupt acknowledge, sets its In-Service flip-flop if there is a pending interrupt request, and returns the interrupt priority level to the CPU. Used in this manner, the PIC is not used to manage interrupts, but is used to prioritize and report service requests to the CPU.

Another way to use the poll mode is to connect a poll mode PIC to a master or slave PIC input. When used this way, an interrupt is generated, but the interrupt service routine must read the poll mode PIC to identify the interrupt source. Using this mode, more than 64 interrupt sources could be handled. A maximum configuration would have a master PIC, eight slave PIC's, and eight poll mode PIC's connected to each slave for a total of 512 interrupt sources.

3.8.2 STATUS READ

Interrupt request inputs are handled by the following three internal PIC registers:

a. Interrupt Request Register (IRR), which contains all interrupt levels that are requesting service.

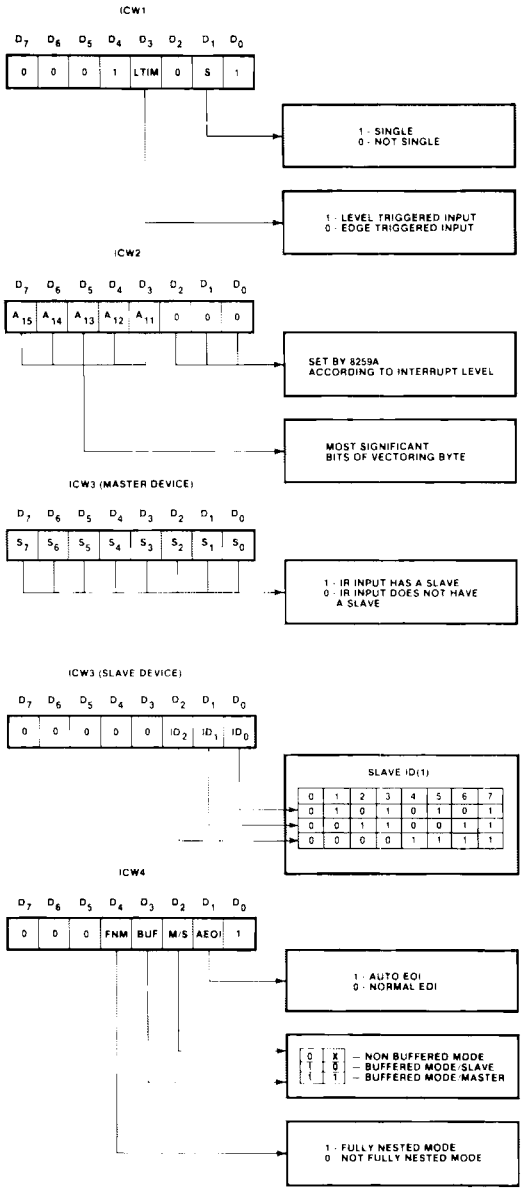
b. In-Service Register (ISR), which contains all interrupt levels that are being serviced.

c. Interrupt Mask Register (IMR), which contains the interrupt request lines which are masked.

These registers can be read by following the instructions in Table 3-29.

3.8.3 INITIALIZATION COMMAND WORDS

The on-board master PIC and each external slave PIC requires a separate initialization sequence to work in a particular mode. The initialization sequence requires three Initialization Command Words (ICW's) for a single PIC system and requires four ICW's for a each PIC in a system with slave PIC's. The ICW formats are shown in Figure 3-13.



x-482

NOTE 1: SLAVED ID IS EQUAL TO THE CORRESPONDING MASTER IR INPUT.
NOTE 2: X INDICATES "DON'T CARE".

Figure 3-13. PIC Initialization Command Word Formats

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

a. Bits 0 and 4 are both 1's and identify the word as ICW1 for an 8086 CPU operation.

b. Bit 1 denotes whether or not the PIC is employed in a multiple PIC configuration. For a single master PIC configuration (no slaves), bit 1 = 1; for a master with one or more slaves, bit 1 = 0.

NOTE

Bit 1 = 0 when programming a slave PIC.

c. Bit 3 establishes whether the interrupts are requested by a positive-true level input or requested by a low-to-high transition. This applies to all input requests handled by the PIC. In other words, if bit 3 = 1, a low-to-high transition is required to request an interrupt on any of the eight levels handled by the PIC.

The second Initialization Command Word (ICW2) represents the vectoring byte (identifier) and is required by the 8086 CPU during interrupt processing. ICW2 consists of the following:

a. Bits D3-D7 (A11-A15) represent the five most significant bits of the vector byte. These are supplied by the programmer. The value of these bits determines the address of the interrupt vector table in memory.

b. Bits D0-D2 represent the interrupt level requesting service. These bits are provided by the 8259A during interrupt processing. These bits should be programmed as 0's when initializing the PIC.

NOTE

The 8086 CPU multiplies the vector byte by four. This value is then used by the CPU as the vector address.

Table 3-26 lists the vector byte contents for interrupts IR0-IR7.

Table 3-26. Interrupt Vector Byte

Data Bit Level	D7	D6	D5	D4	D3	D2	D1	D0
IR7	A15	A14	A13	A12	A11	1	1	1
IR6	A15	A14	A13	A12	A11	1	1	0
IR5	A15	A14	A13	A12	A11	1	0	1
IR4	A15	A14	A13	A12	A11	1	0	0
IR3	A15	A14	A13	A12	A11	0	1	1
IR2	A15	A14	A13	A12	A11	0	1	0
IR1	A15	A14	A13	A12	A11	0	0	1
IR0	A15	A14	A13	A12	A11	0	0	0

The third Initialization Command Word (ICW3) is required only if bit 1 = 0 in ICW1, specifying that multiple PIC's are used; i.e., one or more PIC's are slaved to the on-board master PIC. ICW3 programming can be in one of two formats: master mode format and slave mode format.

a. For master mode, the D0-D7 (S0-S7) bits correspond to the IR0-IR7 bits of the master PIC and designate which levels have slave PIC's connected to them. For example, if a slave PIC is connected to the master PIC IR3 input, code bit 3 = 1.

b. For a slave PIC, the D0-D2 (ID0-ID2) bits identify the master IR line that the slave PIC is connected to. The slave compares its cascade input (generated by the master PIC) with these bits and, if they are equal, the slave releases an interrupt vector byte upon the reception of the second INTA during interrupt processing. For example, if a slave is connected to the master interrupt line IR5, code bits ID0-ID2 = 101.

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

a. Bit D0 is a 1 to identify that the PIC is used in an iAPX 86 system.

b. Bit D1 (AEOI) programs the End-Of-Interrupt function. Code bit 1 = 1 if an EOI is to be automatically generated by the hardware when the interrupt is acknowledged by the CPU. Code bit 1 = 0 if an EOI command is to be generated by software before returning from the interrupt service routine.

c. Bit D2 (M/S) specifies if ICW4 is addressed to a master PIC or a slave PIC. For example, code bit 2 = 1 in ICW4 for the master PIC. If bit D3 (BUF) is zero, bit D2 has no function.

d. Bit D3 (BUF) specifies whether the 8259A is operating in the buffered or nonbuffered mode. Fox example, code bit 3 = 1 for buffered mode.

   The master PIC in an iSBC 86/05, with or without slaves, must be operated in the buffered mode.

e. Bit D4 (SFNM) programs the fully nested or special fully nested mode. (Refer to paragraph 3.8.1.1 and 3.8.1.2.)

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

- Master PIC -- No Slaves

  ICW1
  ICW2
  ICW4

- Master PIC - With Slave(s)

  ICW1
  ICW2
  ICW3
  ICW4

- Each Slave PIC

  ICW1
  ICW2
  ICW3
  ICW4

3.8.4 OPERATION COMMAND WORDS

After being initialized, the master and slave PIC's can be programmed at any time for various operating modes. The Operation Command Word (OCW) formats are shown in Figure 3-14 and discussed in paragraph 3.8.7.

3.8.5 ADDRESSING

The master PIC uses Ports 00C0 and 00C2 to write initialization and operation command words and Ports 00C4 and 00C6 to read status, poll, and mask bytes. Addresses for the specific functions are provided in Table 3-3 and Table 3-9.

Slave PIC's, if employed, are accessed via the Multibus interface and their port addresses are determined by the hardware that contains the slave PIC's.

3.8.6 INITIALIZATION

To initialize the PIC's (master and slaves), proceed as follows (Table 3-26 provides a typical PIC initialization subroutine for a PIC operated in the non-bus vectored mode (no slave PIC's); Table 3-27 and 3-28 are typical master PIC and slave PIC initialization subroutines for the bus vectored mode):

a. Disable system interrupts by executing a CLI (Clear Interrupt Flag) instruction.

b. Initialize master PIC by writing ICW's in the following sequence:

   (1) Write ICW1 to Port 00C0 and ICW2 to Port 00C2.

   (2) If slave PIC's are used, write ICW3 and ICW4 to Port 00C2. If no slave PIC's are used, omit ICW3 and write ICW4 only to Port 00C2.

c. Initialize each slave PIC by writing ICW's in the following sequence: ICW1, ICW2, ICW3, and ICW4.

d. Enable system interrupts by executing an STI (Set Interrupt Flag) instruction.

NOTE

Each PIC independently operates in the fully nested mode (paragraph 3.8.1.1) or the special fully nested mode (paragraph 3.8.1.2) after initialization and before an Operation Control Word (OCW) programs it otherwise.

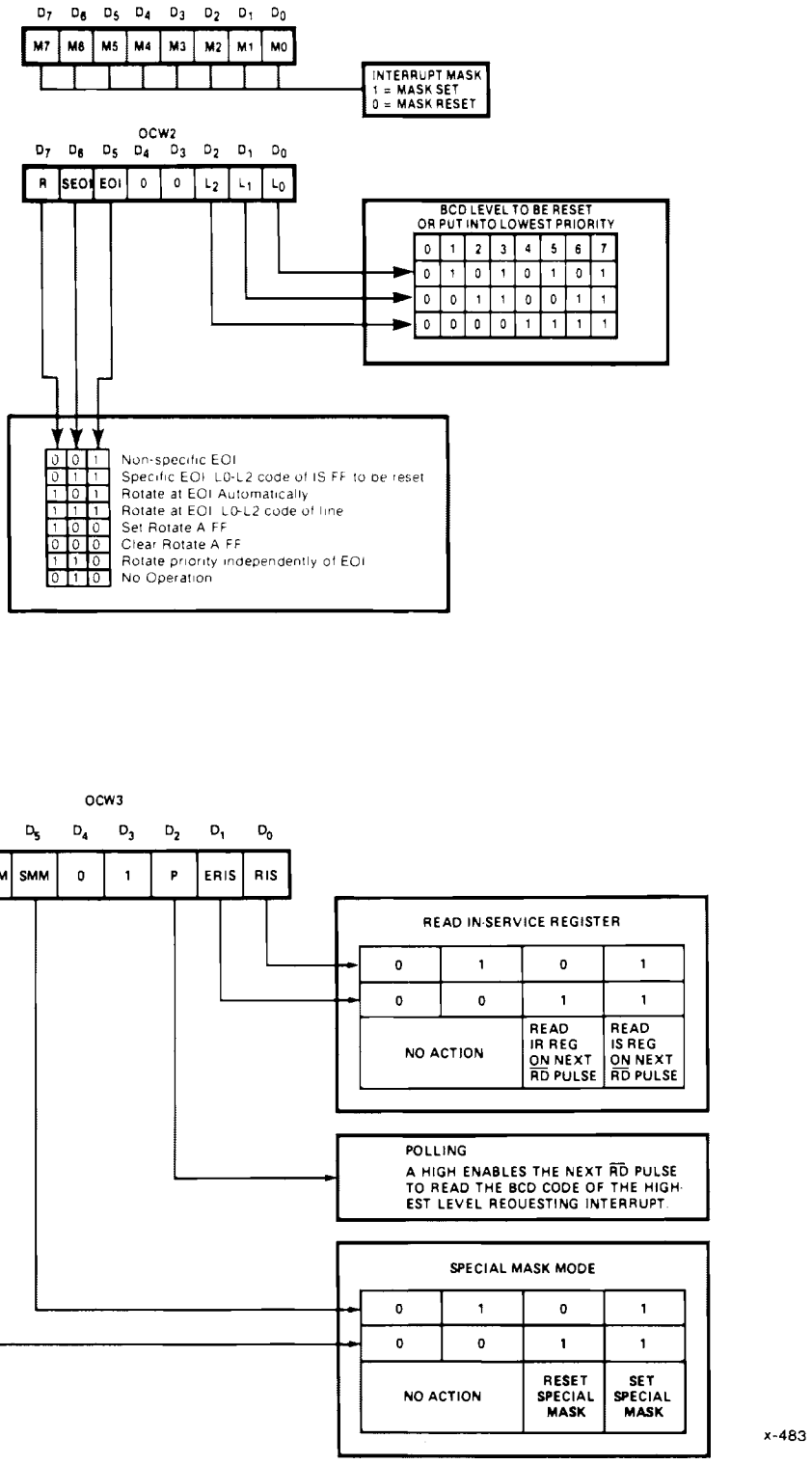


Figure 3-14. PIC Operation Control Word Formats

Table 3-27. Typical PIC Initialization Subroutine (NBV Mode)

```
;INT49 INITIALIZES THE PIC.  A 32-BYTE ADDRESS BLOCK BEGINNING WITH
;00020H IS SET UP FOR INTERRUPT SERVICE ROUTINES.
;PIC MASK IS SET, DISABLING ALL PIC INTERRUPTS.
;PIC IS IN SPECIAL FULLY NESTED MODE, NON-AUTO EOI.
;USES SMASK; DESTROYS-A.

          PUBLIC    INT59
          EXTRN     SMASK

INT49:    MOV       AL,13H
          OUT       0C0H,AL             ;ICW1 TO PIC
          MOV       AL,08H
          OUT       0C2H,AL             ;ICW2 TO PIC
          MOV       AL,1DH
          OUT       0C2H,AL             ;ICW4 TO PIC
          MOV       AL,0FFH
          CALL      SMASK
          RET

          END
```

Table 3-28. Typical Master PIC Initialization Subroutine (BV Mode)

```
;INTMA INITIALIZES MASTER PIC WITH A SINGLE SLAVE ATTACHED
;TO THE 0 LEVEL INTERRUPT INPUT.
;PIC MASK IS SET WITH ALL PIC INTERRUPTS DISABLED.
;MASTER PIC IS SPECIAL FULLY NESTED, NON-AUTO EOI.
;USES SMASK; DESTROYS AL.

          PUBLIC    INTMA
          EXTRN     SMASK

INTMA:    MOV       AL,11H              ;ICW1
          OUT       0C0H,AL
          MOV       AL,08H              ;ICW2
          OUT       0C2H,AL
          MOV       AL,01H              ;ICW3
          OUT       0C2H,AL
          MOV       AL,1DH              ;ICW4
          OUT       0C2H,AL
          MOV       AL,0FFH
          CALL      SMASK
          RET

          END
```

Table 3-29. Typical Slave PIC Initialization Subroutine (BV Mode)

```
;INTSL INITIALIZES A SLAVE PIC LOCATED AT ADDRESS BLOCK
;BEGINNING WITH 00040H.
;PIC IS FULLY NESTED, NON-AUTO EOI.
;PIC IS IDENTIFIED AS SLAVE 0.
;USES-SETI, DESTROYS-AL.

          PUBLIC    INTSL

INTSL:    MOV       AL,11H              ;ICW1
          OUT       0C0H,AL
          MOV       AL,10H              ;ICW2
          OUT       0C2H,AL
          MOV       AL,00H              ;ICW3
          OUT       0C2H,AL
          MOV       AL,19H              ;ICW4
          OUT       0C2H,AL
          RET

          END
```

3.8.7 OPERATION

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

a. Auto-rotating priority

b. Specific rotating priority.

c. Status read of Interrupt Request Register (IRR).

d. Status read of In-Service Register (ISR).

e. Interrupt mask bits set, reset, or read.

f. Special mask mode set or reset.

Table 3-30 lists details of the above operations. Note that an End-Of-Interrupt (EOI) or a Specific End-Of-Interrupt (SEOI) command is required at the end of each interrupt service routine to reset the ISR unless the Automatic End-Of-Interrupt function has been selected. The EOI command is used in the fully nested and auto-rotating priority modes and the SEOI command, which specifies the bit to be reset, is used in the specific rotating priority mode. Tables 3-31 through 3-35 provide typical subroutines for the following:

a. Read IRR (Table 3-31).

b. Read ISR (Table 3-32).

c. Set mask register (Table 3-33).

d. Read mask register (Table 3-34).

e. Issue EOI command (Table 3-35).

Table 3-30. PIC Operation Procedure

Operation	Procedure
Auto-Rotating Priority Mode	To set: In OCW2, write a Rotate Priority at EOI command (A0H) to Port 00C0. To terminate interrupt and rotate priority: In OCW2, write EOI command (20H) to Port 00C0.

Table 3-30. PIC Operation Procedures (continued)

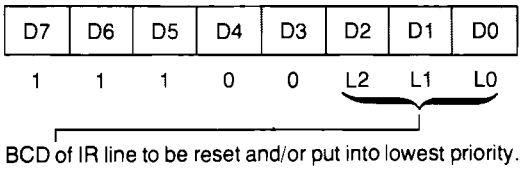
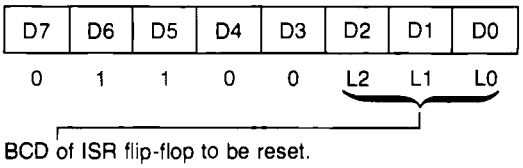
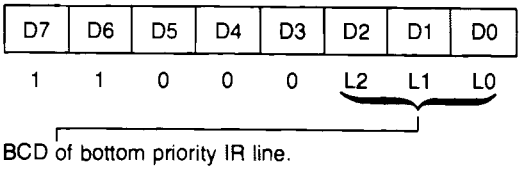
Operation	Procedure
Specific Rotating Priority Mode	To set: In OCW2, write a Rotate Priority at SEOI command in the following format to Port 00C0:  To terminate interrupt and rotate priority: In OCW2, write an SEOI command in the following format to Port 00C0.  To rotate priority without EOI: In OCW2, write a command word in the following format to Port 00C0: 

Table 3-30. PIC Operation Procedures (continued)

Operation	Procedure
Interrupt Request Register (IRR) Status	The IRR stores a "1" in the associated bit for each IR input line that is requesting an interrupt. To read the IRR (refer to note): (1) Write 0AH to Port 00C0. (2) Read Port 00C0. Status is as follows: D7 D6 D5 D4 D3 D2 D1 D0 IR line: 7 6 5 4 3 2 1 0
In-Service Register (ISR) Status	The ISR stores a "1" in the associated bit for interrupt levels that are being serviced. The ISR is updated when an EOI command is issued. To read the ISR (refer to note): (1) Write 0BH to Port 00C0. (2) Read Port 00C0. Status is as follows: D7 D6 D5 D4 D3 D2 D1 D0 IR line: 7 6 5 4 3 2 1 0 Be sure to reset ISR bit at End-of-Interrupt when in the following modes: Auto-Rotating (both types) and Special Mask. To reset ISR in OCW2, write: D7 D6 D5 D4 D3 D2 D1 D0 0 1 1 0 0 L2 L1 L0 BCD identifies bit to be reset.

Table 3-30. PIC Operation Procedures (continued)

Operation	Procedure
Interrupt Mask Register	To set mask bits OCW1, write the following mask byte to Port 00C2. D7 D6 D5 D4 D3 D2 D1 D0 IR Bit Mask: M7 M6 M5 M4 M3 M2 M1 M0 1 = Mask Set, 0 = Mask Reset To read mask bits, read Port 00C2.
Special Mask Mode	The Special Mask Mode enables the unmasked interrupt levels that are lower priority than the level being serviced. To set, write 68H to Port 00C0. To reset, write 48H to Port 00C0.
Note: The PIC "remembers" if is is set-up to read the IRR or ISR contents. It is not necessary to re-write OCW3 to re-read the register that was previously selected for reading.	

Table 3-31. Typical PIC Interrupt Request Register Read Subroutine

```
;RR0 READS PIC INTERRUPT REQUEST REG.
:DESTROYS-AL.

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

          END
```

Table 3-32. Typical PIC In-Service Register Read Subroutine

```
;RISO READS PIC IN-SERVICE REGISTER.
;DESTROYS-A.

          PUBLIC     RISO

RISO:     MOV        AL,0BH          ;OCW3 RIS INSTRUCTION TO PIC
          OUT        0C0H,AL
          IN         AL,0C0H
          RET

          END
```

Table 3-33. Typical PIC Set Mask Register Subroutine

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

          PUBLIC     SMASK

SMASK:    OUT        0C2H,AL
          RET

          END
```

Table 3-34. Typical PIC Mask Register Read Subroutine

```
;RMASK READS PIC MASK REG INTO AL REG.
;DESTROYS-AL.

          PUBLIC     RMASK

RMASK:    IN         AL,0C2H
          RET

          END
```

Table 3-35. Typical PIC End-Of-Interrupt Command Subroutine

```
;EOI ISSUES END-OF-INTERRUPT TO PIC.
;DESTROYS-AL

          PUBLIC   EOI

EOI:      MOV      AL,20H              ;NON-SPECIFIC EOI
          OUT      0C0H,AL
          RET

          END
```

3.9 8086 INTERRUPT HANDLING

The 8086 CPU has two interrupt input request lines: Interrupt Request (INTR) and Non-Maskable Interrupt Request (NMI). All of the interrupt requests handled by the 8259A interrupt controller are connected to the INTR input. The NMI input on the iSBC 86/05 board is not used in the factory default configuration, but can be reconfigured for use as required for a particular application. Refer to Section 2.5 for complete jumper instructions. When both NMI and INTR occur during the execution of an instruction, NMI is serviced first.

Normally, interrupt requests are not serviced until completion of the instruction that was in progress when the interrupt request occurred.

But there are several exceptions. For example, when a repeated operation (such as a block move) is performed, interrupts are accepted after completion of an iteration but before completion of the entire repeated operation.

There are several cases when interrupt service is delayed until completion of the instruction following the instruction that was in progress when the interrupt occurs.

An example of this delay is that interrupts are disabled for one instruction whenever a segment register is loaded. This allows, for example, a new value to be loaded into SS:SP without the risk of an interrupt occurring sometime after SS is loaded, but before SP is loaded.

Section 3.9.1 provides a summary of the NMI input functions and Section 3.9.2 summarizes INTR functions. For a complete discussion of 8086 interrupt handling, refer to THE 8086 FAMILY USER'S MANUAL.

3.9.1 NON-MASKABLE INTERRUPT (NMI)

The NMI input has the higher priority of the two interrupt inputs. The worst case response to NMI is during a multiply, divide, or variable shift instruction.

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

a. Pushes the flag registers into the stack (same as PUSHF).

b. Clears the Interrupt Flag (same as CLI). This disables all maskable interrupts.

c. Transfers control with an indirect call through vector location 00008.

The NMI input is intended mainly for catastrophic error handling or debugging. Upon completion of the service routine, an IRET instruction restores the flags and returns to the interrupted program.

3.9.2 MASKABLE INTERRUPT (INTR)

The INTR input has the lower priority of the two interrupt inputs. When INTR goes active, the CPU performs the following (assuming the Interrupt Flag is set):

a. Issues two acknowledge signals; upon receipt of the second acknowledge signal, the interrupting device (master or slave PIC) will respond with a one-byte interrupt identifier.

b. Pushes the Flag registers onto the stack (same as a PUSHF instruction.

c. Clears the Interrupt Flag, thereby disabling further maskable interrupts.

d. Multiplies by four (4) the binary value (X) contained in the one-byte identifier from the interrupting device.

e. Transfers control with an indirect call through location 4X.

Upon completion of the service routine, an IRET instruction restores the CPU flags and returns to the interrupted program.

3.9.2.1 Master PIC Byte Identifier

The master (on-board) PIC responds to the second acknowledge signal from the CPU only if the interrupt request is from a non-slaved device (a device that is connected directly to one of the master PIC IR inputs).

The master PIC has eight IR inputs numbered IR0 through IR7, which are identified by a 3-bit binary number. Thus, if an interrupt request occurs on IR5, the master PIC responds to the second acknowledge signal from the CPU by outputting the vector address on the data bus to be read by the CPU. The address is formed by adding the interrupt level to the value supplied in ICW2, and multiplying it by 4.

3.9.2.2 Slave PIC Byte Identifier

Each slave PIC is initialized with a 3-bit identifier (ID) in ICW3. These three bits will form a part of the byte identifier transferred to the CPU in response to the second acknowledge signal.

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

***

CHAPTER 4. SERVICE INFORMATION

4.1 INTRODUCTION

This chapter contains the service and repair assistance instructions, replacement parts list and diagram, jumper post location diagram, and schematic diagrams.

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

a. Date you received the product.
b. Complete part number of the product (including dash number). On boards, this number is usually silk-screened onto the board. On other MCSD products, it is usually stamped on a label.
c. Serial number of product. On boards, this number is usually stamped on the board. On other MCSD products, the serial number is usually stamped on a label.
d. Shipping and billing addresses.
e. Purchase order number for billing purposes if your Intel product warranty has expired.
f. Extended warranty agreement information, if applicable.

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

Regional Telephone Numbers:

Western Region:	602-869-4951
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 Product Service marketing Administration personnel.

4.3 REPLACEMENT PARTS

A complete list of replacement parts is provided in Table 4-1. This list provides the part number, manufacturer, description and quantity of the item. Notice that each item is referenced in the parts location diagram. Table 4-2 provides the full name of the manufacturer which is abbreviated in Table 4-1. Some of the parts are available from any normal commercial source, and should be ordered by their generic description. These items are called out as CML, rather than listing a specific part number. Figure 4-1 shows the location of each iSBC 86/05 referenced part in Table 4-1. Figures 4-4 & 4-6 show the location of the iSBC 341 ROM Expansion Module parts and the iSBC 302 RAM Expansion Module parts.

4.4 SERVICE DIAGRAMS

The following schematic diagrams are included in this chapter:

Figure 4-3	iSBC 86/05 Board
Figure 4-5	iSBC 341 ROM Expansion Module
Figure 4-7	iSBC 302 RAM Expansion Module

Notice that a functional description of each jumper connection on a particular schematic sheet is referenced on the schematic.

The schematic diagrams are current when the manual is printed. However, minor revisions to the diagrams may occur between manual printings. Therefore, Intel provides photocopies of the current schematic diagrams with the board, when it is shipped from the factory. These diagrams should be inserted into this manual for future reference. In most instances, the diagrams shipped with the board will be identical to those printed in the manual.

4.5 INTERNAL SIGNALS

Internal board signals which traverse from one sheet to another in Figure 4-3 are identified by a single or double alpha character within a box (e.g., G AN). The signal mnemonic is shown adjacent to the boxed character, along with the source or destination sheet number. Signals coming into the sheet are shown on the left side of the diagram. Conversely, signals leaving the sheet are shown on the right side.

To follow a signal from one sheet to another, read the sheet number and boxed character, then look for the same boxed character on the indicated sheet. For example, if you are going to trace the path of MRDC/ when it exits sheet 4, the first step would be to turn to the indicated sheet. Since MRDC/ will be entering sheet 6, as indicated on sheet 4 look for the BH symbol on the left side of the sheet. Notice that the inputs on the sheet also list the source sheet number (sheet 4 in this example).

Each signal will keep the same boxed character throughout Figure 4-3. This will enable you to trace the signal to any sheet with minimal effort.

The internal board signal mnemonics are listed and defined in Table 4-3. The signals are listed according to boxed code alphabetical order.

Signals which do not have boxed codes are either board inputs or outputs. These signals are described in Chapter 2.

4.6 JUMPER LOCATIONS

Jumper post locations are shown in Figure 4-2. This drawing is provided for use as a quick reference in locating the physical location of a jumper post on the iSBC 86/05 board. Jumper locations are also listed on each schematic sheet, with a brief description of the jumper's function.

Table 4-1. iSBC® 86/05 Replacement Parts List

Reference Designator	Description	Mfr Part No.	Mfr Code	Qty
4-1	iSBC 86/05 Single Board Computer	142879	INTEL	1
C1-6,8-11, 16-31,33-44, 49,50,52, 57	Capacitor, cer. 0.1uf, 50V +80-20%	OBD	COML	43
C12,15,32	Capacitor, mica 10pf, 500V 5%	OBD	COML	3
C13,45,46, 54,55	Capacitor, cer. 10uf, 10V 20%	OBD	COML	5
C47	Capacitor, tant. 6.8uf, 35V 20%	OBD	COML	1
C53,56	Capacitor, tant. 22uf, 15V, 10%	OBD	COML	2
CR1,2	Diode, 1N 4148	OBD	COML	2
DS1	Diode, LED, red low profile	521-9180	DIALIGHT	1
J3,4	iSBX connector, 44-pin	68-369	VIKING	2
J5,6,7	Socket (20-pin strip	7195-295-5	EMC	1
L1	Inductor, 4.7uh, 10%	101808-023	COML	1
R1,10,14, 17-19,31,32	Resistor, 10K, 1/4W, 5%	OBD	COML	8
R2	Resistor, 220K, 1/4W,5%	OBD	COML	1
R3-5,7,12, 26	Resistor, 1K, 1/4W, 5%	OBD	COML	6
R6,11,13, 16,25,30	Resistor, 5.6K, 1/4W, 5%	OBD	COML	6
R8,9	Resistor, 560, 1/4W, 5%	OBD	COML	2
R9,20	Resistor, 100K, 1/4W, 5%	OBD	COML	2
R33,34	Resistor, 510, 1/4W, 5%	OBD	COML	2
RP1	Resistor pack, 10K, 6-pin,3/4W 2%	OBD	COML	1
RP2	Resistor pack, 10K, 10-pin, 1-1/4W, 2%	OBD	COML	1
RP3	Resistor pack, 1K, 14-pin, 1-1/2W, 2%	OBD	COML	1
U1,18,36	NAND gate, 2 input	74LS00	TI	3
U2,28	AND gate, 2 input	74LS08	TI	2
U3,55	OR gate, 2 input	74S32	TI	2
U4	Flip-flop, D type	74S175	TI	1
U5	Multivibrator, mono. retrig	74LS123	TI	1
U6	Inverter	74LS04	TI	1
U7,49,64,65	Bus Transceiver	8287	INTEL	4
U12	Counter, sync. 4-bit	74S163	TI	1
U13	Counter, sync. 4-bit	74LS163	TI	1
U14	Line Receiver	75188N	TI	1

Table 4-1. iSBC® 86/05 Replacement Parts List (continued)

Reference Designator	Description	Mfr Part No.	Mfr Code	Qty
U15	Line Receiver	75189AN	TI	1
U16,17	Clock Generator/Driver	8224	INTEL	2
U19,39	NAND gate, 3 input	74S10	TI	2
U20	Flip-flop, D type	74LS74	TI	1
U21	Prog Interrupt Controller	8259A	INTEL	1
U22	Prog. Peripheral Interface	8255A-5	INTEL	1
U23	Prog. Interval Timer	8253-5	INTEL	1
U24	Prog. Comm. Interface	8251A	INTEL	1
U25,48	Bus Transceiver	8286	INTEL	2
U26,41	Decoder, 3 to 8	74S138	TI	2
U27	NOR gate, 2 input	74S02	TI	1
U29,38	Inverter	74S04	TI	2
U30,50	NAND gate, 2 input	74LS00	TI	2
U31	Bus Driver	DM8097N	NSC	1
U32	Microprocessor 16-bit	8086-2	INTEL	1
U36	NOR gate, 5 input	74S260	TI	1
U40,54	Decoder, 2 to 4	74S139	TI	2
U42	Inverter, OC High Voltage	7406	TI	1
U43	Flip-flop JK, Neg. Edge Trig	74112	TI	1
U44,59	Bus Controller	8288	INTEL	2
U45	Clock Generator/Driver	8284A	INTEL	1
U46,51,68	Latch, D type	74S373	TI	3
U47,56,74	AND gate, 2 input	74S08	TI	3
U52,53,70,71	RAM, static 4K x 4	2168R	INTEL	4
U57	NAND buffer	74S37	TI	1
U58	Bus Arbiter	8289	INTEL	1
U60,62,63	Buffer/Driver/Receiver	74S240	TI	3
U61	Buffer/Driver/Receiver	74S240	TI	1
U72	Pre-programmed PROM (I/O)	143635-001	INTEL	1
U73	Pre-programmed PROM (Memory)	143636-001	INTEL	1
XU7	Socket, 20-pin DIP	DILB20P-108	BURNDY	1
XU8-11	Socket, 14-pin DIP	DILB14P-108	BURNDY	1
XU32	Socket, 40-pin DIP	540-AG11D	AUGAT	1
XU33,34,66 67	Socket, 28-pin DIP	528-AG37D	AUGAT	4
XU35	Socket, 14-pin DIP	514-AG37D	AUGAT	1
XU52,53,70 71	Socket, 20-pin DIP	520-AG37D	AUGAT	4
Y1	Crystal, 19.6608 MHz	OBD	CRYSTEK	1
Y2	Crystal, 15 MHz	OBD	CRYSTEK	1
U3	Crystal, 24 MHz	OBD	CRYSTEK	1
-	Plug, shorting, 2 position	530153-2	AMP	30
-	Plug, shorting, U35	8136-475G1	AUGAT	3

Table 4-2. Manufacturer's Names

Mfr. Code	Manufacturer
AMP	AMP Incorporated
AUGAT	Augat Incorporated
BURNDY	Burndy Corporation
CRYSTEK	Crystek Crystals Corporation
DIALIGHT	Dialight Corporation
EMC	EMC Controls Incorporated
NSC	National Semiconductor Corporation
TI	Texas Instruments Incorporated
VIKING	Viking Connectors Incorporated

Note: OBD = Order by description
CML = Any commercial source

Table 4-3. List of Internal Signal Mnemonics

Code	Mnemonic	Description
A	BUS AEN/	Bus Address Enable
D	ON BD ADR/	On-board address
E	INTA LOCK/	Interrupt acknowledge lock
F	TEST/	Test input to 8086, from parallel interface
G	INTR/	Interrupt request input to 8086
H	NMI	Non-maskable interrupt to 8086
I	RST/	Inverted RESET
J	RST	RESET
K	LOCK/	Bus lock from 8086
L	BHE/S7	Byte high enable/status bit 7 output from 8086
M	A19/S6	Address/status bit output from 8086
N	A18/S5	Address/status bit output from 8086
O	A17/S4	Address/status bit output from 8086
P	A16/S3	Address/status bit output from 8086
Q	AD0-AD15	Internal data/address bus lines
R	S2/	Status bit output from 8086
S	S1/	Status bit output from 8086
T	S0	Status bit output from 8086
U	ALE	Address latch enable output from 8086
V	BHE/	Byte high enable
W	A0-A19	Internal extended address bus lines
X	W/R	Write enable signal to RAM array

Table 4-3. List of Internal Signal Mnemonics (continued)

Code	Mnemonic	Description
Y	MEM/I/O	Memory or I/O selector
Z	D0-D7	Low byte data lines
AC	DEN	Data enable bit from 8288
AD	SYS CLK/	Inverted 8 MHz on-board clock from 8284
AE	MMPRES/	Multimodule board present indicator
AF	CLK	8 MHz on-board clock from 8284
AG	RAM 0 HIGH/	RAM array 0 high byte enable
AH	RAM 1 HIGH/	RAM array 1 high byte enable
AI	RAM 0 LOW/	RAM array 0 low byte enable
AJ	RAM 1 LOW/	RAM array 1 low byte enable
AK	PROM 0 LOW/	PROM array 0 low byte enable
AL	PROM 1 LOW/	PROM array 1 low byte enable
AM	PROM 2 LOW/	PROM array 2 low byte enable
AN	PROM 3 LOW/	PROM array 3 low byte enable
AO	PROM 0 HIGH/	PROM array 0 high byte enable
AP	PROM 1 HIGH/	PROM array 1 high byte enable
AQ	PROM 2 HIGH/	PROM array 2 high byte enable
AR	PROM 3 HIGH/	PROM array 3 high byte enable
AS	8251A CS/	PCI (USART) chip select
AT	8253A CS/	PIT (Interval Timer) chip select
AU	8255A CS/	PPI (Parallel Port) chip select
AV	8259A CS/	PIC (Interrupt Controller) chip select
AW	SBX2 CS0/	Multimodule 1, chip select 0
AX	SBX2 UCS1/	Unqualified chip select 1
AY	SBX1 CS0/	Multimodule 1, chip select 0
AZ	SBX1 UCS1/	Unqualified chip select 1
BA	BCLK	Multibus clock
BB	OVERRIDE/	Multibus override
BC	MCLK/	iSBX Multimodule clock (same as BCLK/)
BE	INTA/	Interrupt Acknowledge
BF	IORC/	I/O Read command from 8288
BG	AIOWC/	Advanced I/O write command from 8288
BH	MRDC/	Memory read command from 8288
BI	AMWTC/	Advanced memory write command from 8288
BJ	DT/R	Data transmit/receive signal from 8288
BK	ALE/	Address latch enable from 8288
BN	PU1	Pull-up resistor
BO	INTA LOCK	Interrupt acknowledge lock
BP	59 INTA/	Interrupt acknowledge from PIC
BQ	LOCAL INTA2	Local interrupt acknowledge
BR	MWAIT2/	iSBX Multimodule 2 wait signal
BS	MWAIT1/	iSBX Multimodule 1 wait signal
BT	RAM/PROM	RAM or PROM select from memory decode
BU	BUSY/	Memory busy

Table 4-3. List of Internal Signal Mnemonics (continued)

Code	Mnemonic	Description
BV	RDY/	Memory ready
BX	EXT CLK/	External (off-board) clock for PIT
BY	GATE0 CNTRL	Gate 0 line for PIT
BZ	GATE1 CNTRL	Gate 1 line for PIT
CA	STXD	Secondary Transmit data output
CB	SITX	Serial Interface Transmit Interrupt
CC	SIRX	Serial Interface Receive Interrupt
CE	TIMER0 INTR	PIT output 0 to PIC
CF	TIMER1 INTR	PIT output 1 to PIC
CG	PA INTR	Parallel port "A" interrupt to PIC
CH	CB INTR	Parallel port "B" interrupt to PIC
CI	BUS INTR OUT	Multibus interrupt output
CJ	NMI MASK/	Non-maskable interrupt mask
CK	SBX2 INT0	Multimodule 2, interrupt 0
CL	SBX2 INT1	Multimodule 2, interrupt 1
CM	SBX1 INT0	Multimodule 1, interrupt 0
CN	SBX1 INT1	Multimodule 1, interrupt 1
CO	59 DEN/	Data bus enable from PIC
SS/	PLC	Power Line Clock from power supply
UU	PU2	Pull-up resistor number 2
XX	I/O CS/	I/O data bus enable
ZY	BUS INTA/	Multibus interrupt acknowledge
ZZ	BUS LOCK/	Multibus lock

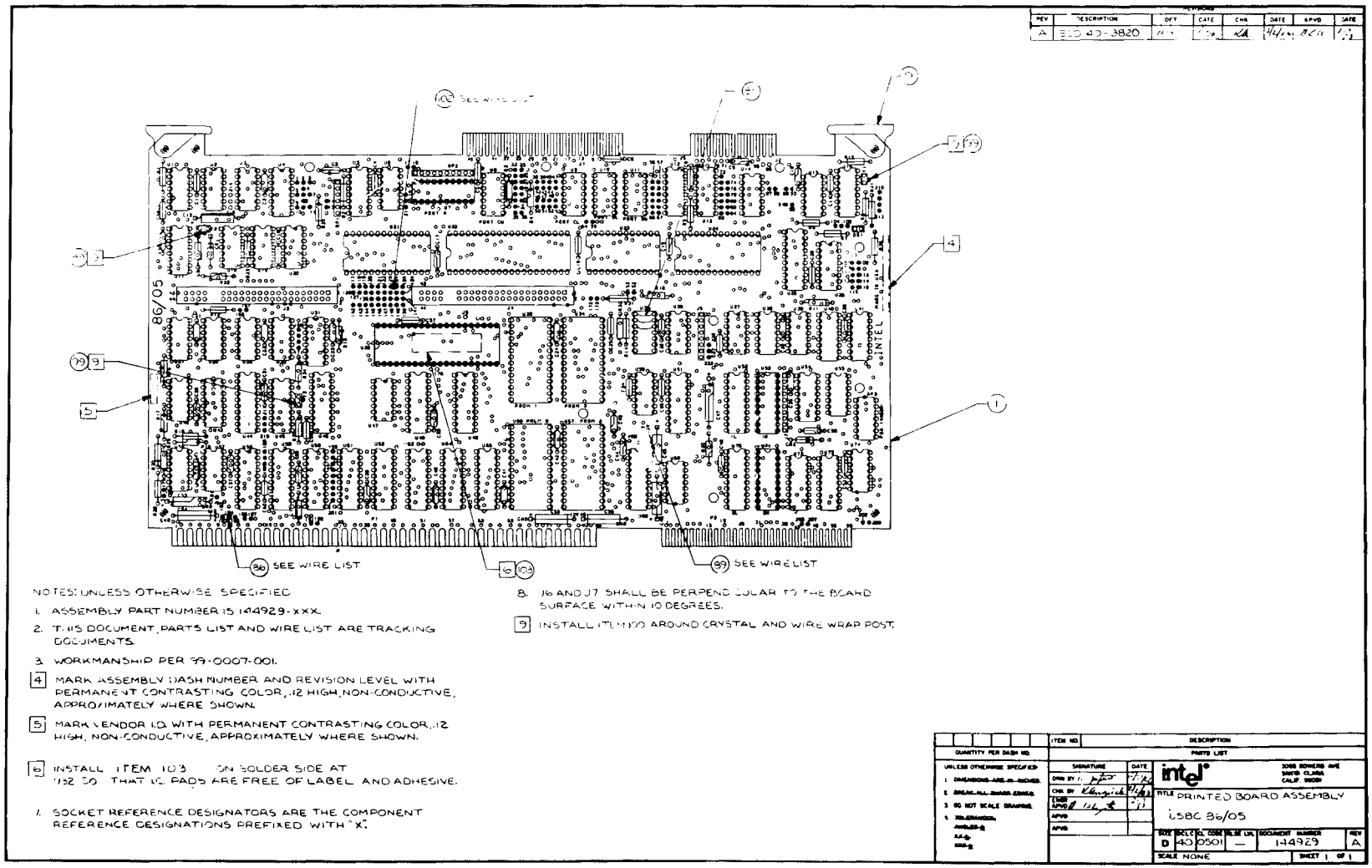


Figure 4-1. iSBC® 86/05 Board Parts Location Drawing

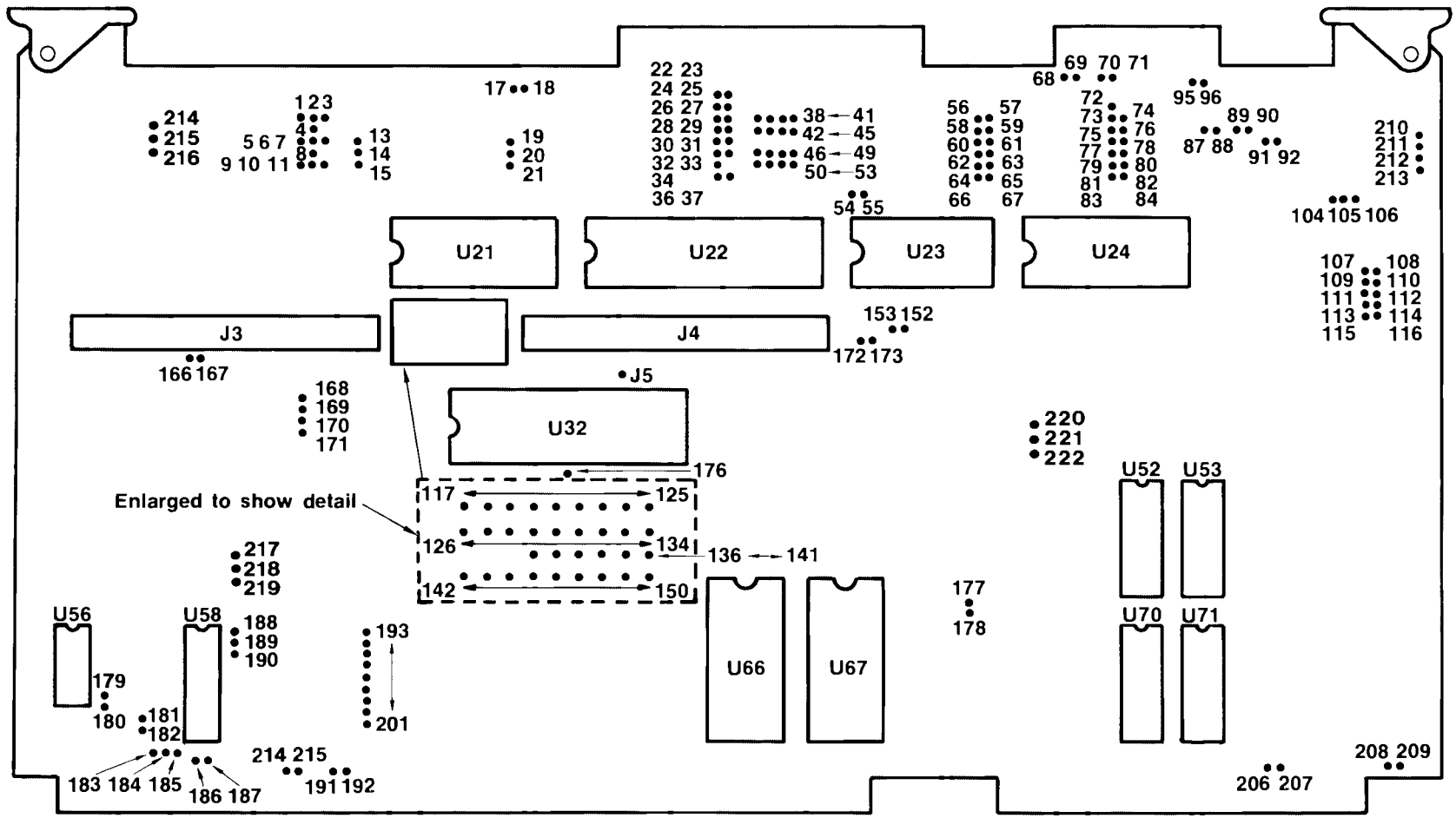


Figure 4-2. iSBC® 86/05 Board Jumper Post Location Drawing

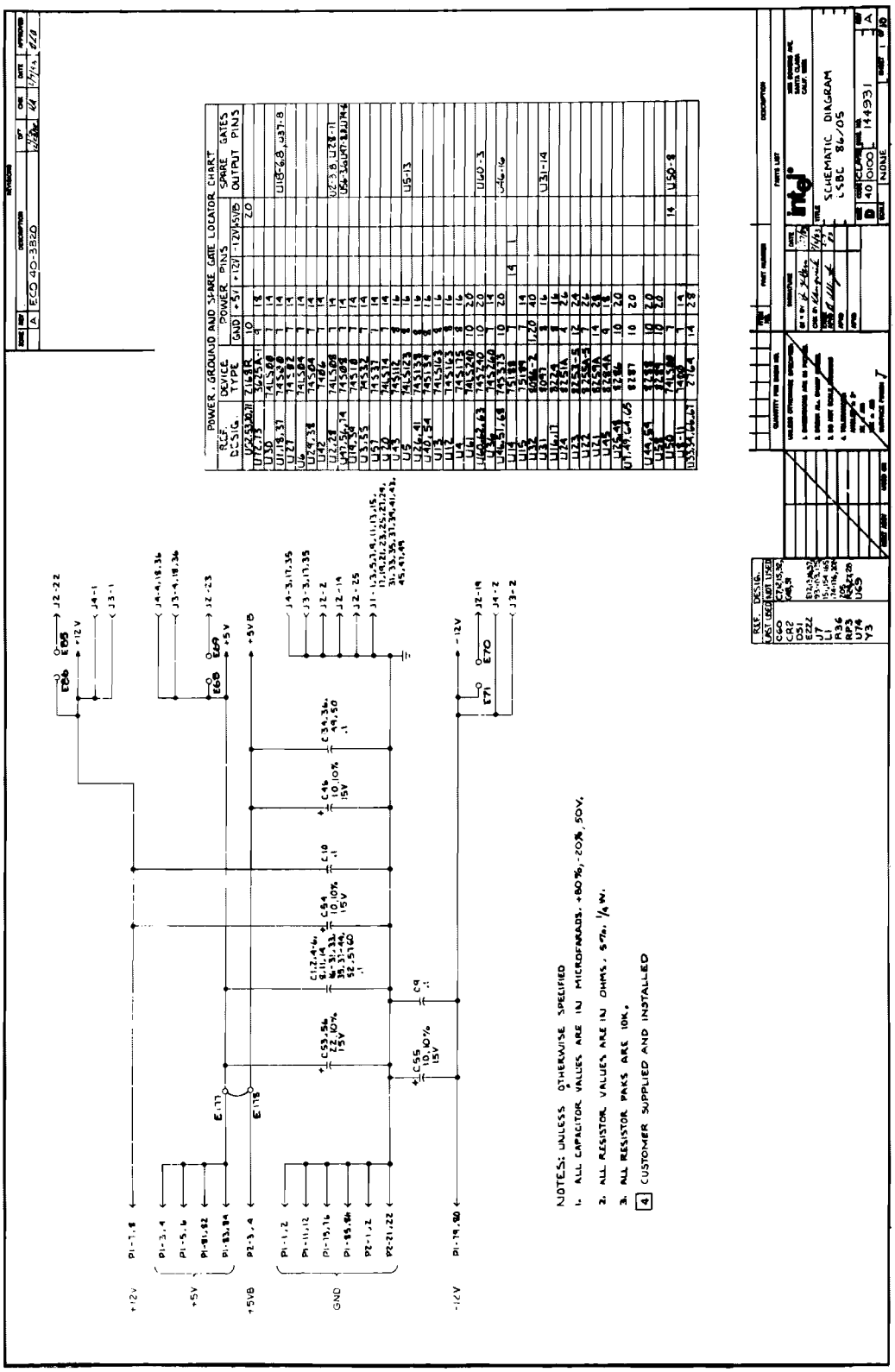


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 1 of 10)

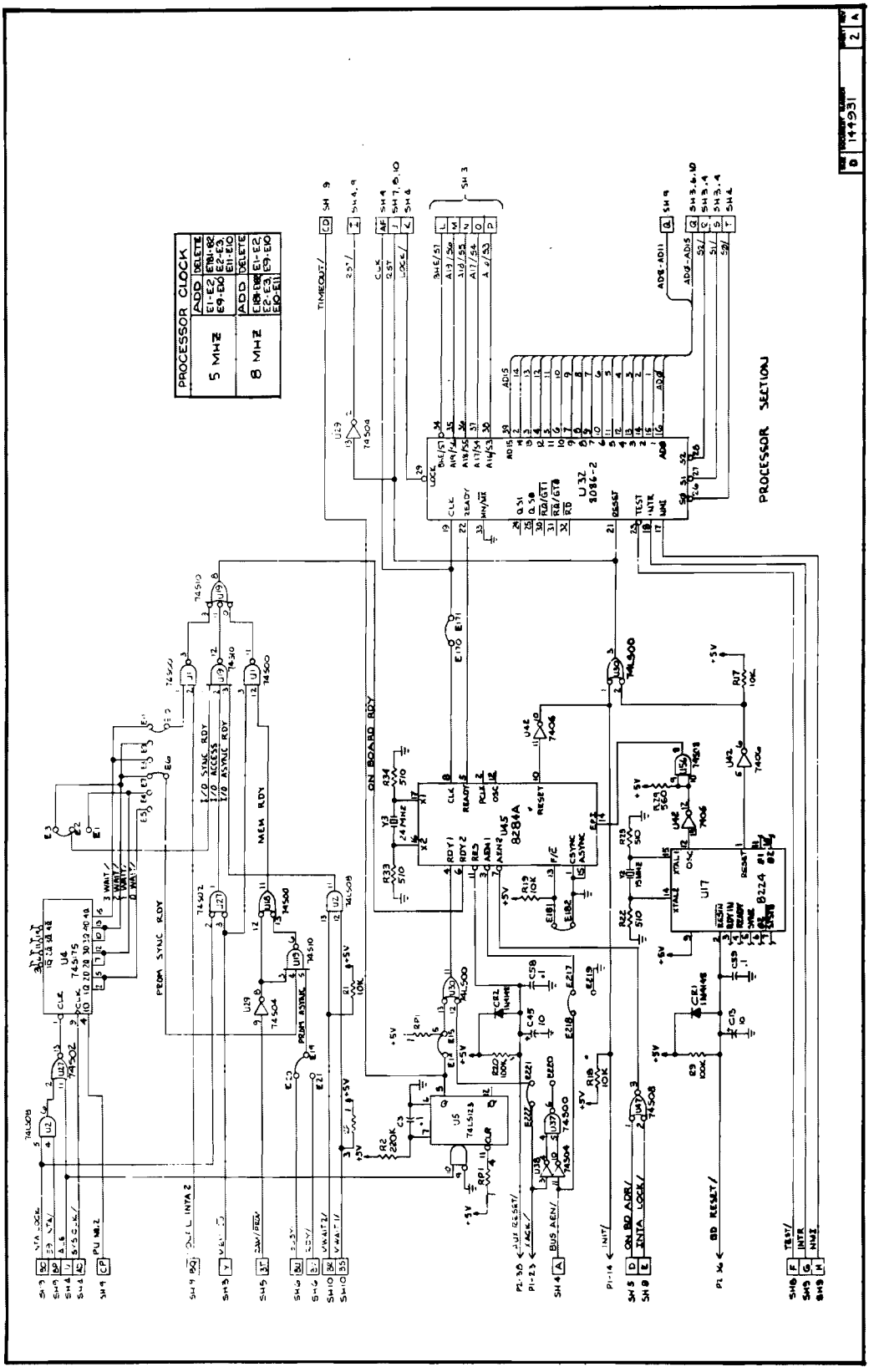


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 2 of 10)

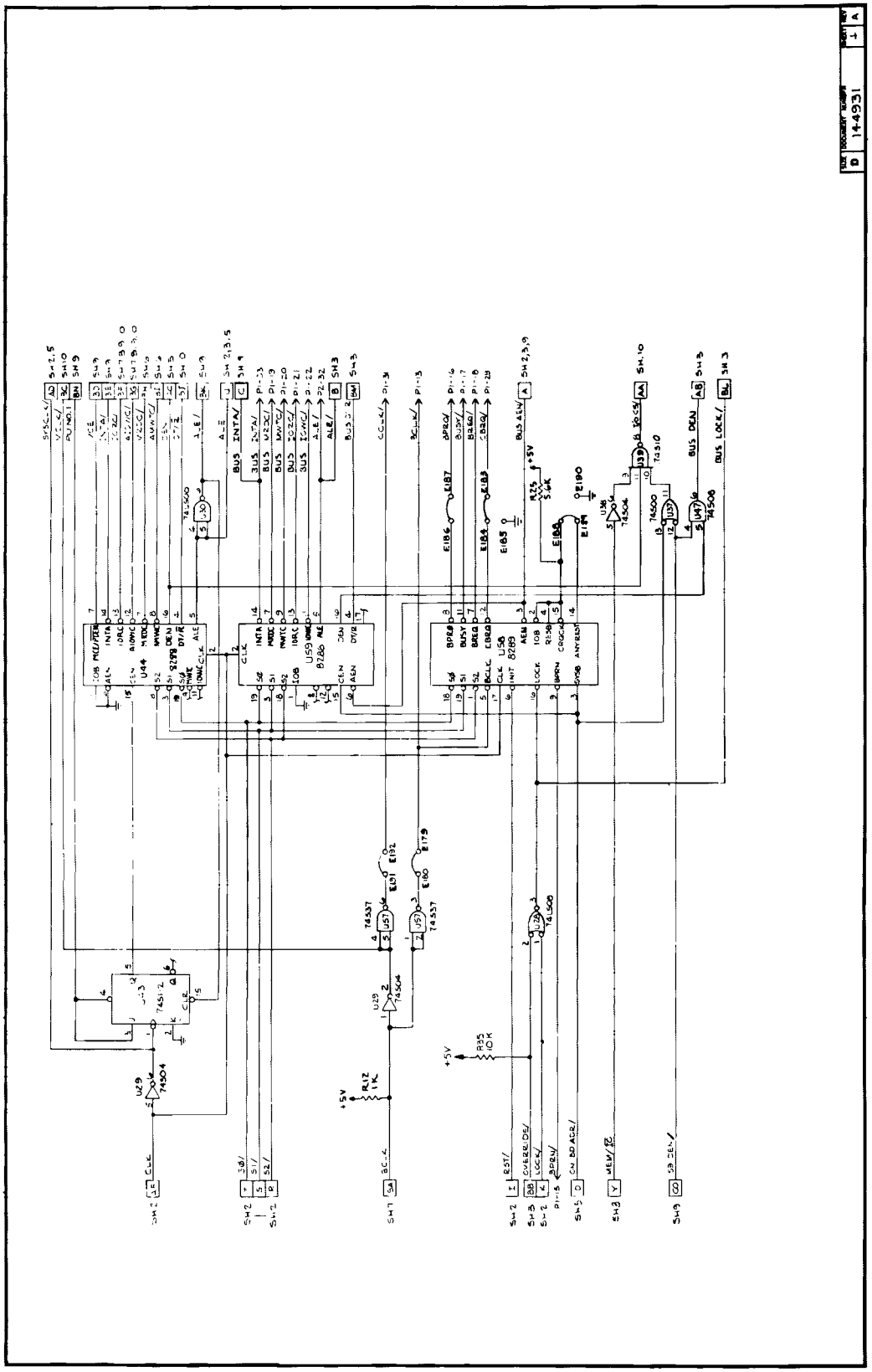


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 3 of 10)

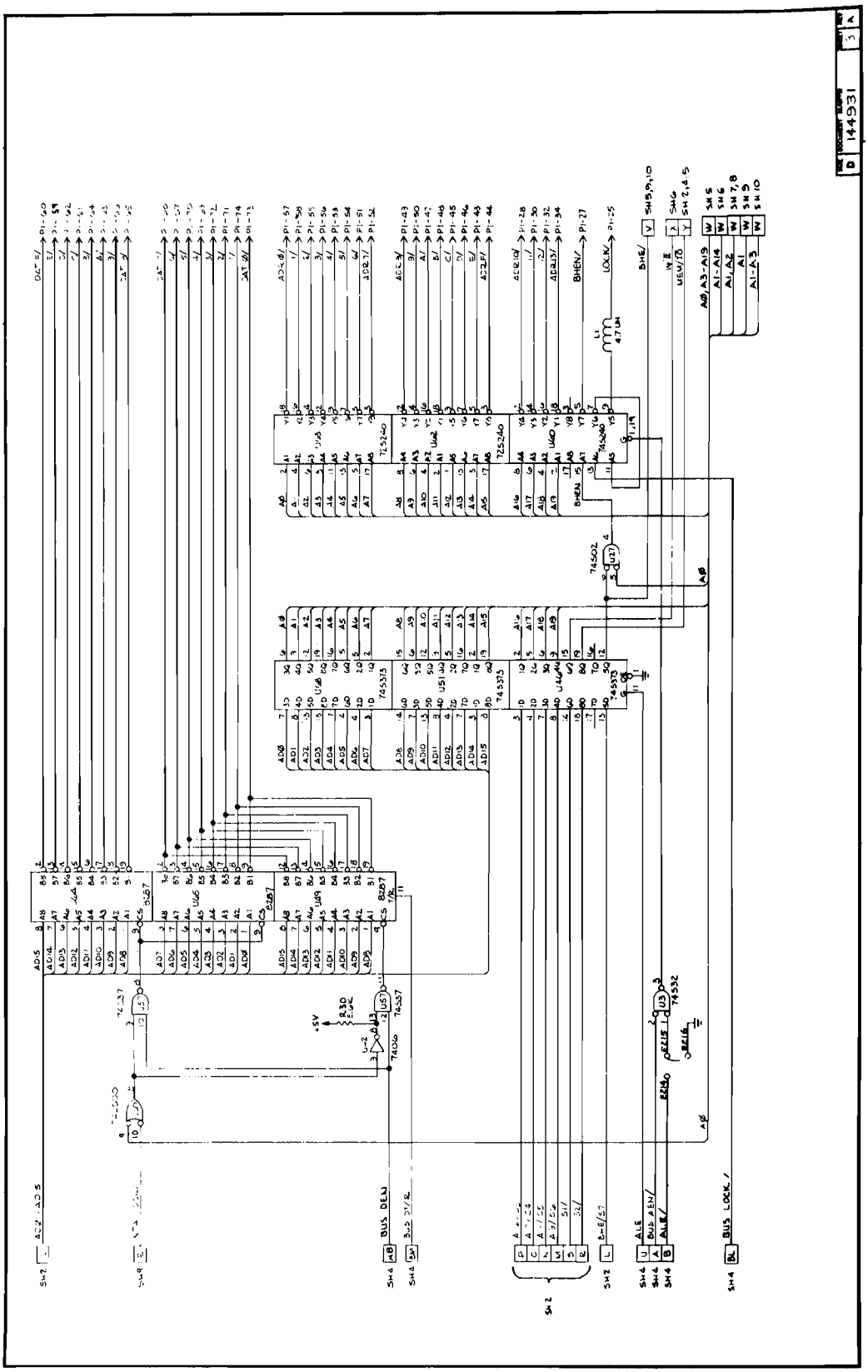


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 4 of 10)

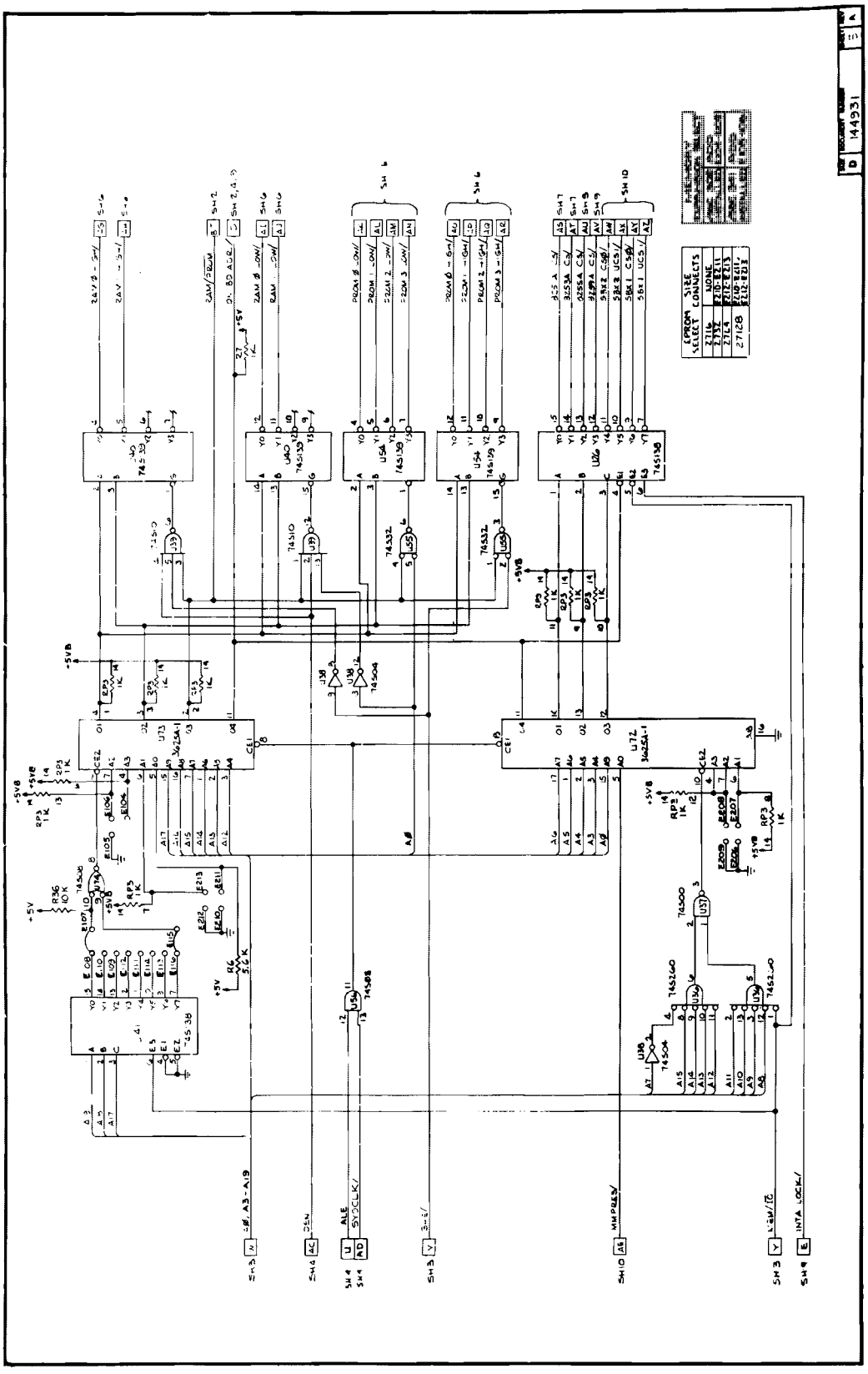


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 5 of 10)

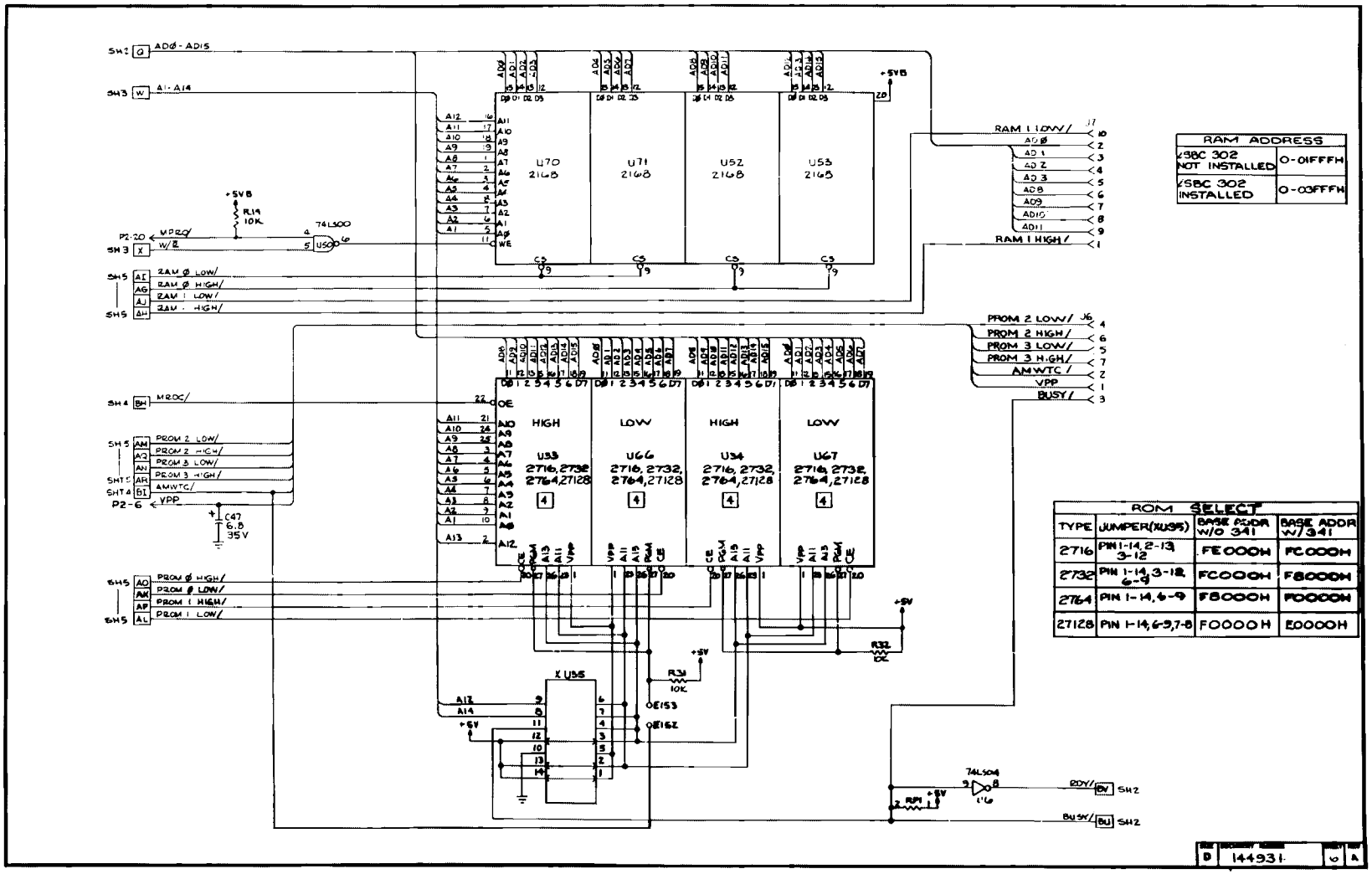


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 6 of 10)

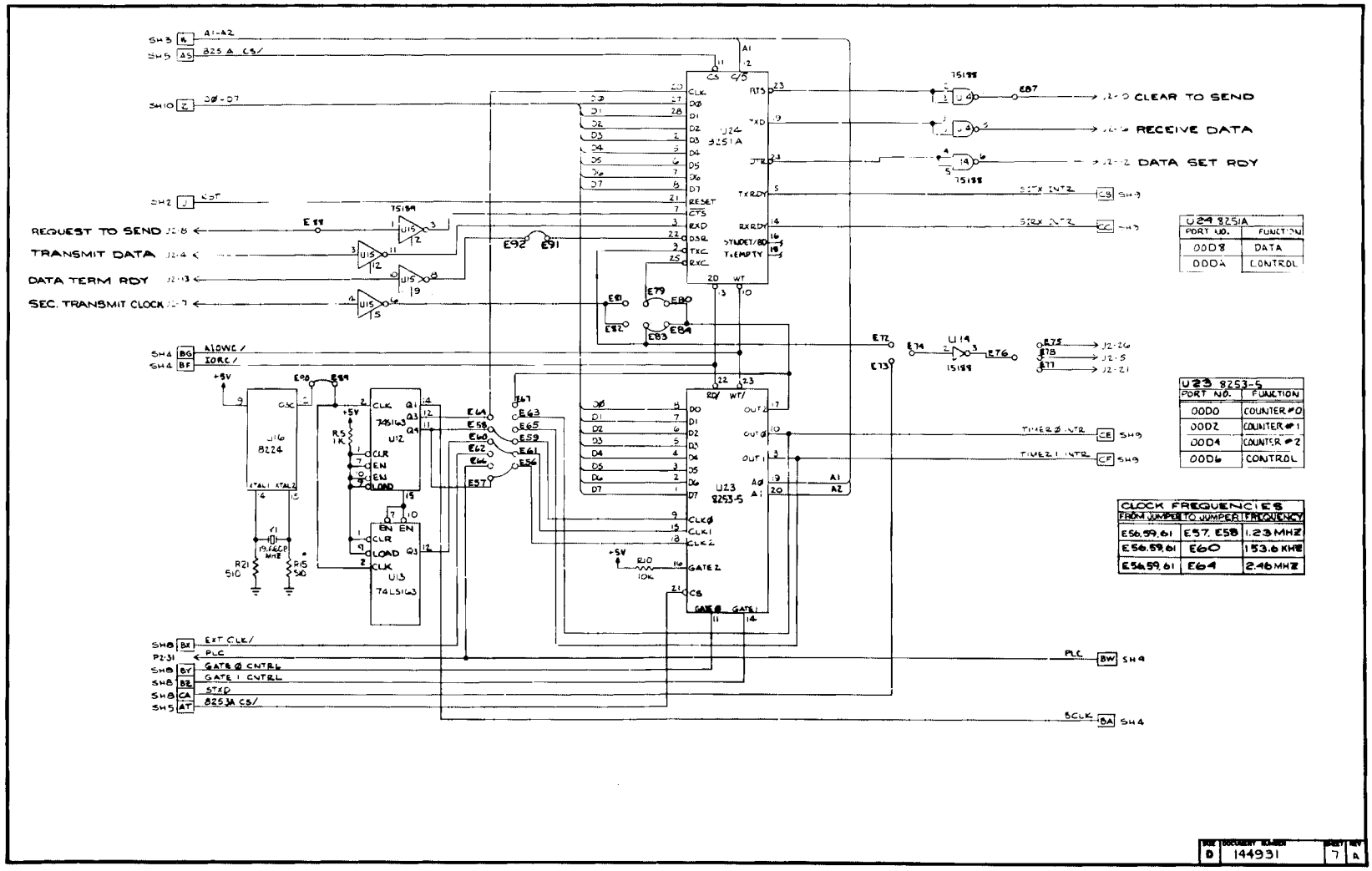


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 7 of 10)

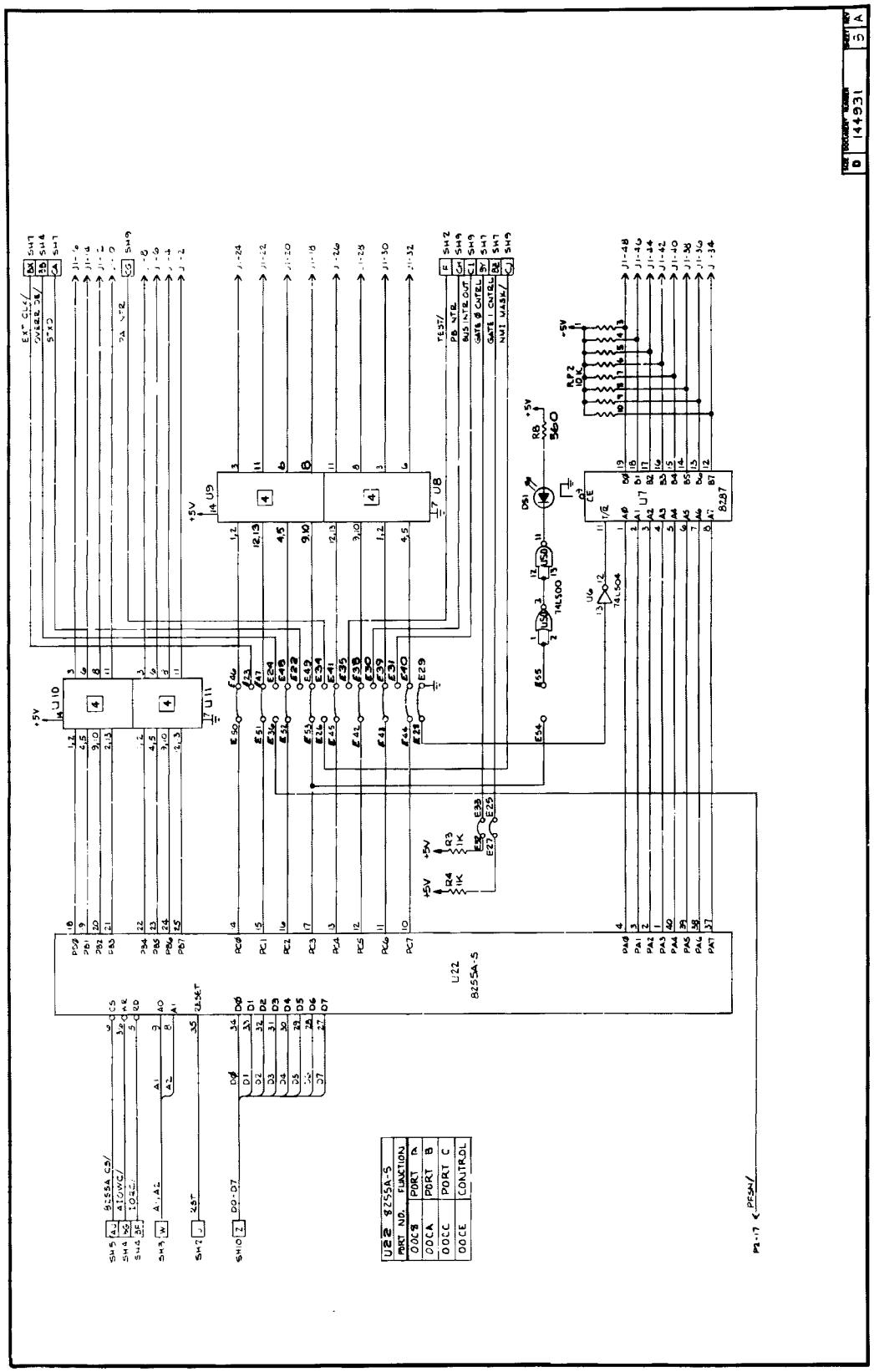


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 8 of 10)

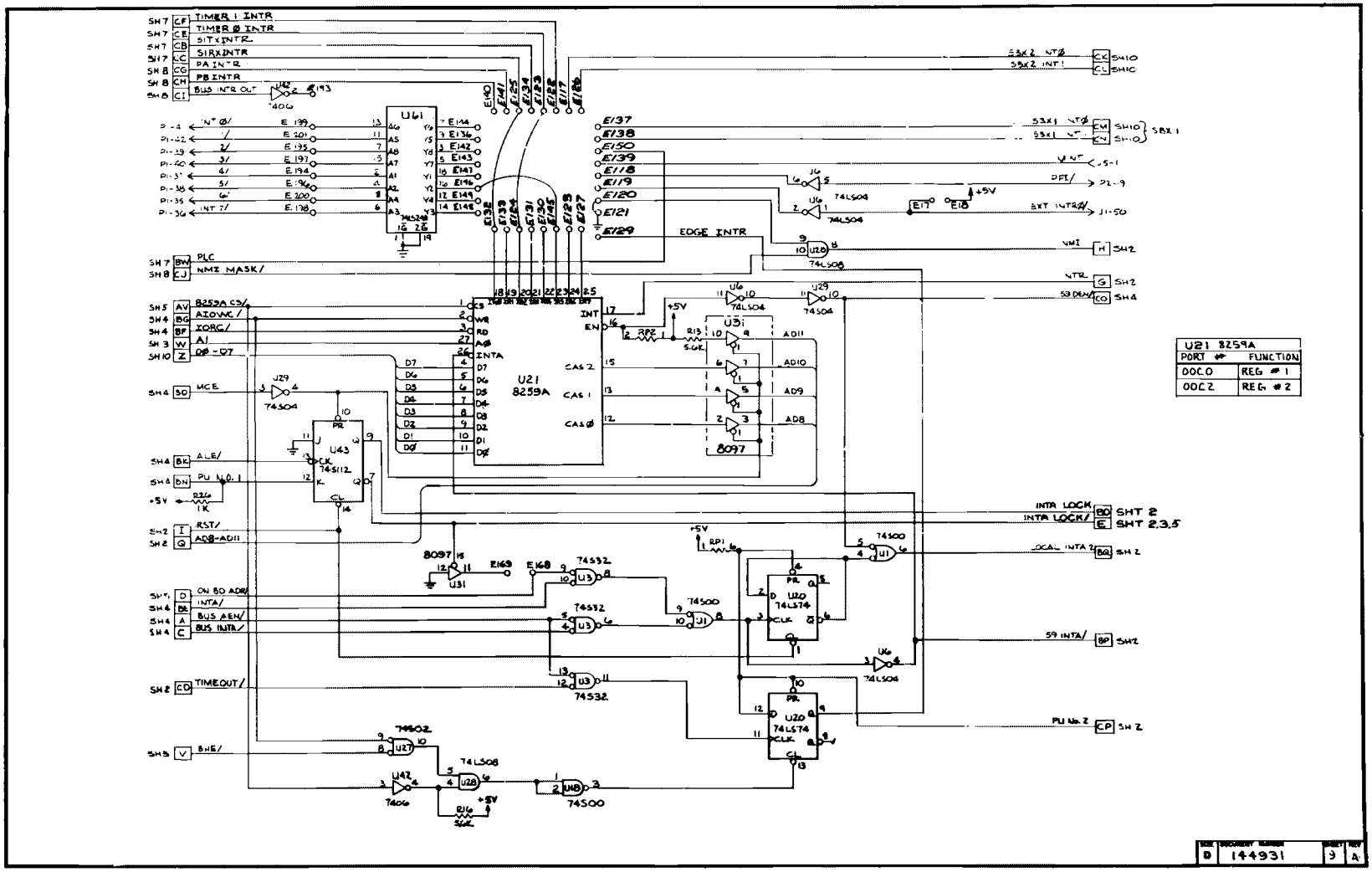


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 9 of 10)

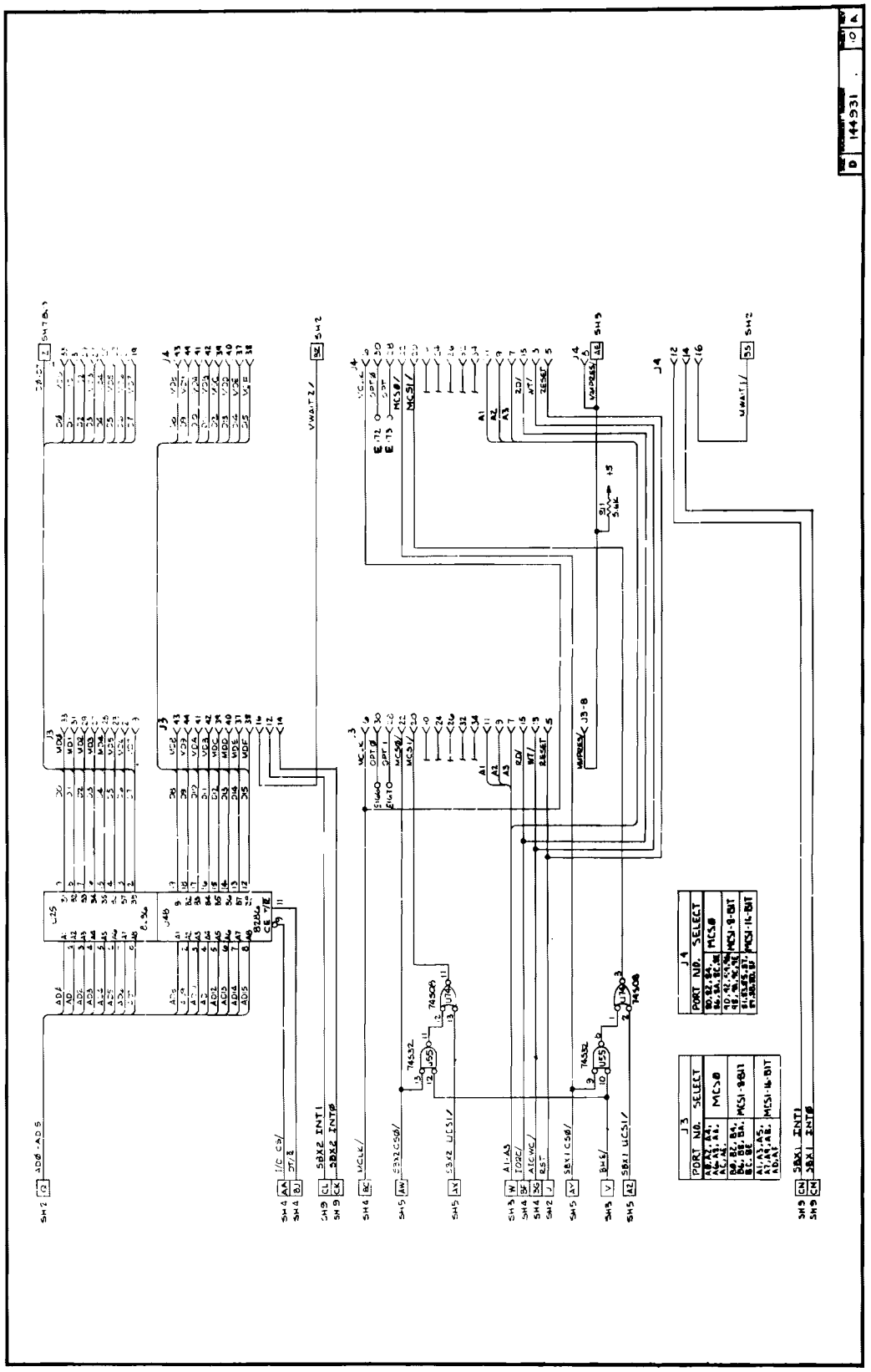


Figure 4-3. iSBC® 86/05 Board Schematic Diagram (Sheet 10 of 10)

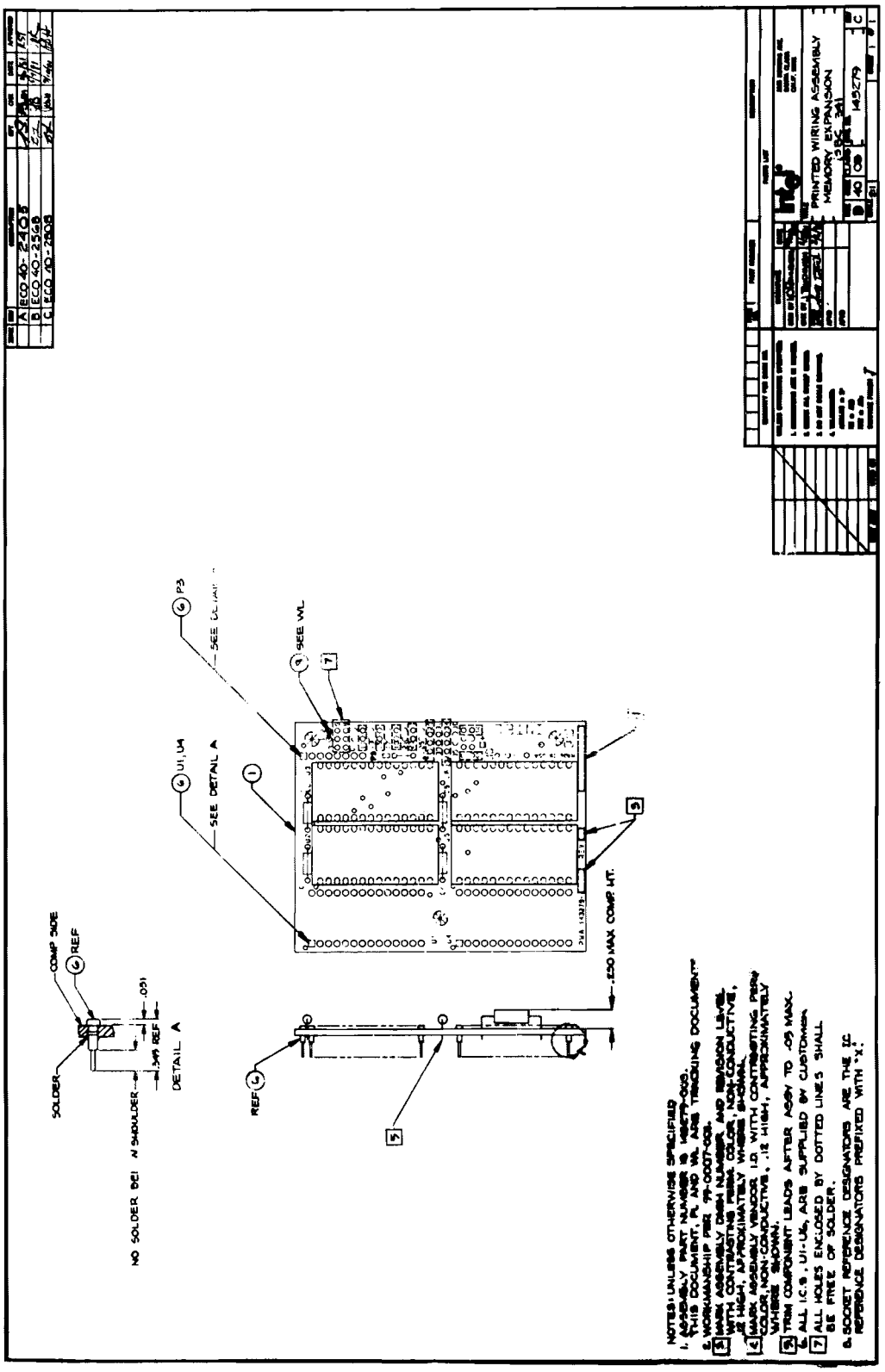


Figure 4-4. iSBC® 341 ROM Expansion Module Parts Location Diagram

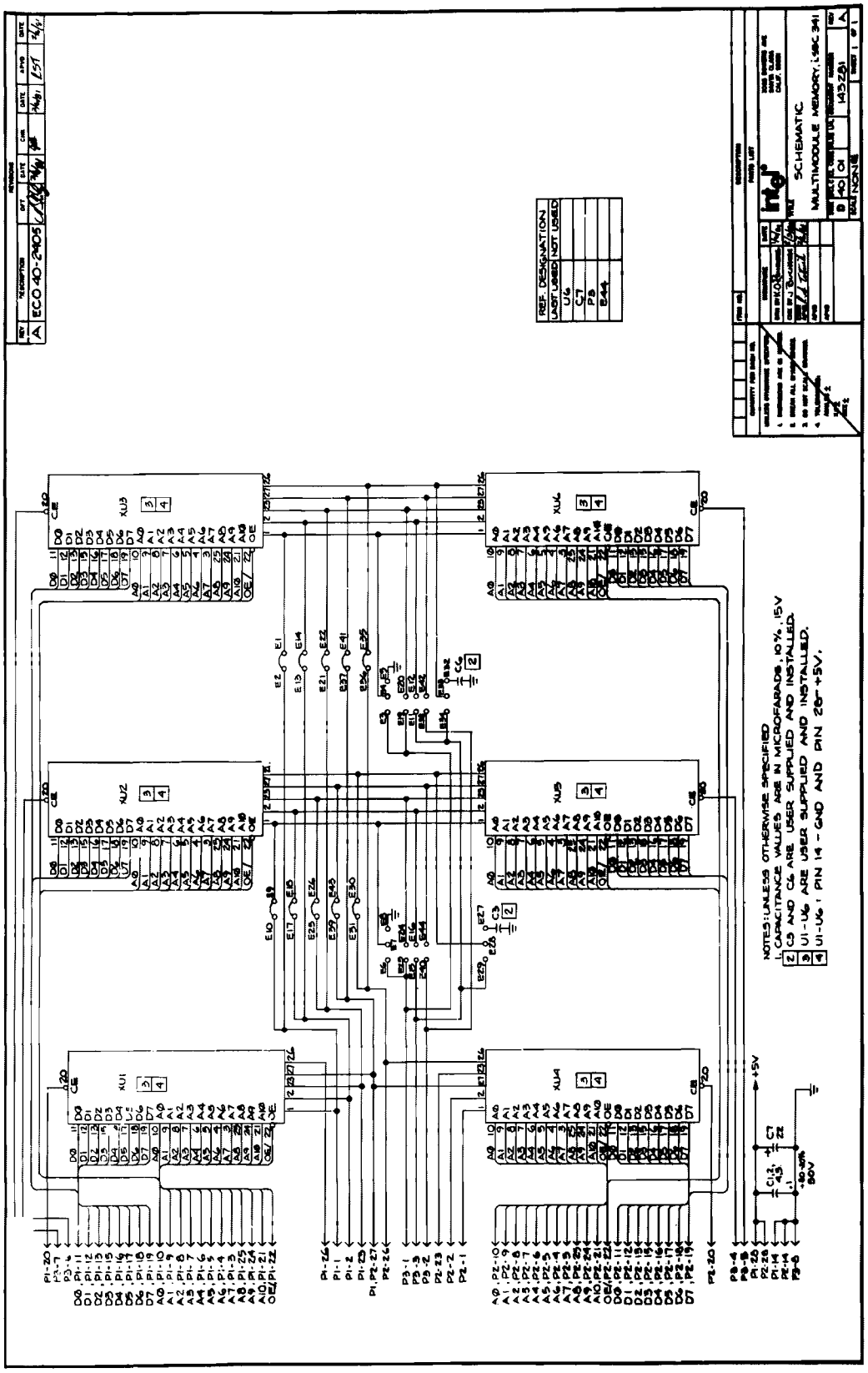


Figure 4-5. iSBC® 341 ROM Expansion Module Schematic Diagram

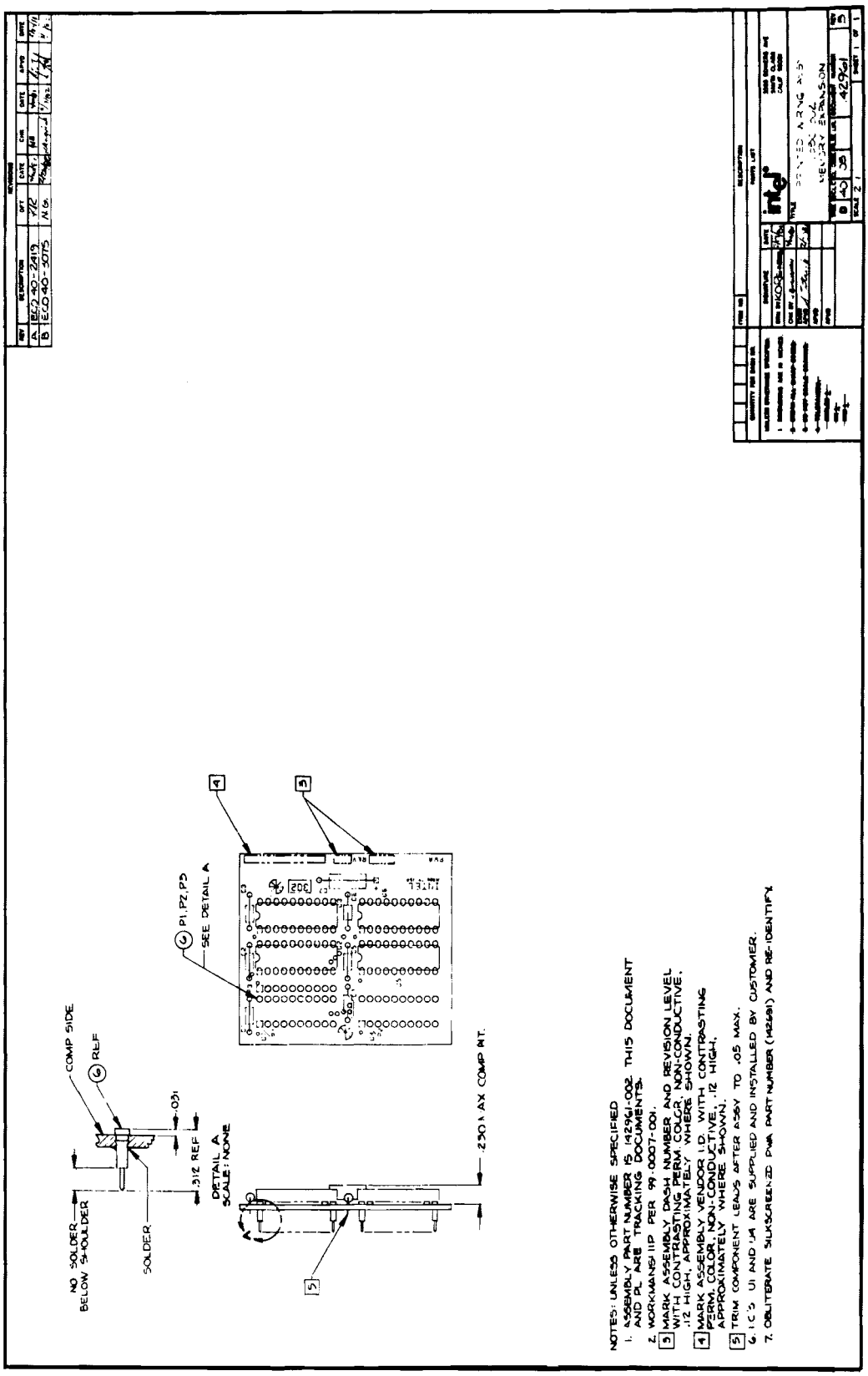


Figure 4-6. iSBC® 302 RAM Expansion Module Parts Location Diagram

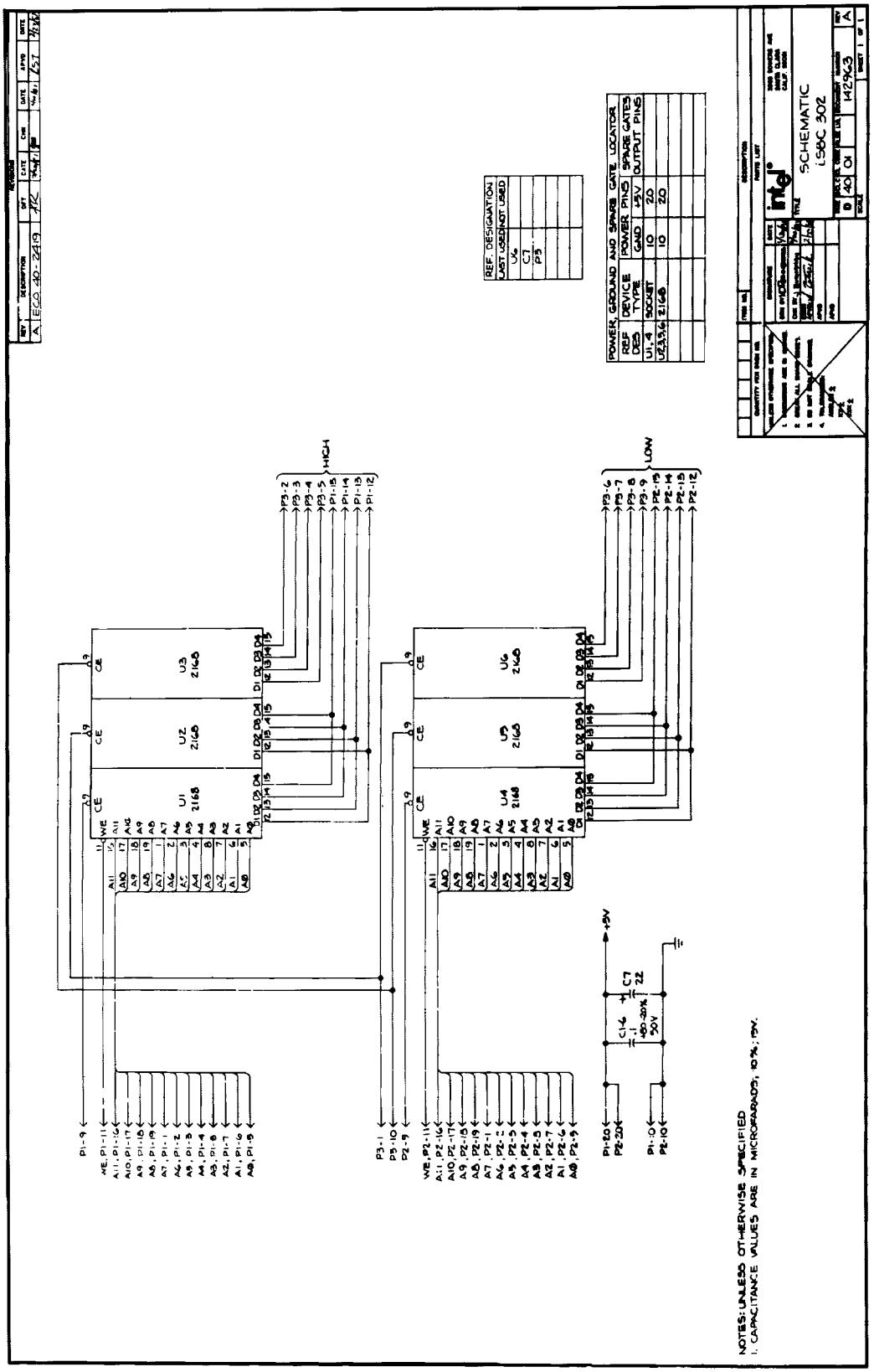


Figure 4-7. iSBC® 302 RAM Expansion Module Schematic Diagram

APPENDIX A. INTERFACE SIGNAL INFORMATION

Multibus connector P1 and auxiliary connector P2 interface the iSBC 86/05 board signals and power lines to the other boards in your system and the power supply. Where applicable, these signals conform to the Intel Multibus Interface standard. Pin assignments for iSBC 86/05 board connectors P1 & P2 are listed in Tables A-1 & A-3, respectively. Signal definitions are provided in Tables A-2 & A-4, respectively.

The signal names indicate whether or not the signal; lines on the Multibus System Bus are active high or active low. If the signal name ends with a slash (/), then the logical-electrical state relationship for that signal is:

Logical State	Electrical Signal Level	At Receiver	At Driver
0	H = TTL high state	5.25V≥H≥2.0V	5.25V≥H≥2.4V
1	L = TTL low state	.8V≥L≥-.5V	.5V≥L≥0V

If the signal name has no slash, then the logical-electrical state relationship for that signal is:

Logical State	Electrical Signal Level	At Receiver	At Driver
0	L = TTL low state	.8V≥L≥.5V	.5V≥L≥0V
1	H = TTL high state	5.25V≥H≥2.0V	5.25V≥H≥2.4V

These specifications are based on TTL where the power source is 5 volts $\pm$ 5%, referenced to ground (GND).

DC and AC characteristics of the P1 signals used on the iSBC 86/05 board are provided in Tables A-5 and A-6, respectively. Refer to the board timing diagram (Figure A-1) for parameter identification. Auxiliary connector P2 signal characteristics are listed in Table A-7.

Parallel I/O DC signal characteristics are listed in Table A-8.

Connector pin assignments for the iSBX Bus connectors (J3 and J4) are listed in Table A-9. The iSBX Bus signal descriptions are listed in Table A-10.

Table A-1. Multibus® Interface Connector P1 Pin Assignments

	(COMPONENT SIDE)			(CIRCUIT SIDE)		
	PIN[1]	MNEMONIC	DESCRIPTION	PIN[1]	MNEMONIC	DESCRIPTION
POWER SUPPLIES	1	GND	Signal GND	2	GND	Signal GND
	3	+5V	+5Vdc	4	+5V	+5Vdc
	5	+5V	+5Vdc	6	+5V	+5Vdc
	7	+12V	+12Vdc	8	+12V	+12Vdc
	9	-5V	-5Vdc	10	-5V	-5Vdc
	11	GND	Signal GND	12	GND	Signal GND
BUS CONTROLS	13	BCLK/	Bus Clock	14	INIT/	Initialize
	15	BPRN/	Bus Priority In	16	BPRO/	Bus Priority Out
	17	BUSY/	Bus Busy	18	BREQ/	Bus Request
	19	MRDC/	Mem Read Cmd	20	MWTC/	Mem Write Cmd
	21	IORC/	I/O Read Cmd	22	IOWC/	I/O Write Cmd
	23	XACK/	XFER Acknowledge	24	INH1/	Inhibit RAM[2]
BUS CONTROLS AND ADDRESS	25	LOCK/	Dual Port Lock	26	INH2/	Inhibit ROM[2]
	27	BHEN/	Byte High Enable	28	AD10/	
	29	CBRQ/	Common Bus Request	30	AD11/	Address
	31	CCLK/	Constant Clk	32	AD12/	Bus
	33	INTA/	Interrupt Acknowledge	34	AD13/	
INTERRUPTS	35	INT6/	Parallel	36	INT7/	Parallel
	37	INT4/	Interrupt	38	INT5/	Interrupt
	39	INT2/	Requests	40	INT3/	Requests
	41	INT0/		42	INT1/	
ADDRESS	43	ADRE/		44	ADRF/	
	45	ADRC/		46	ADRD/	
	47	ADRA/	Address	48	ADRB/	Address
	49	ADR8/	Bus	50	ADR9/	Bus
	51	ADR6/		52	ADR7/	
	53	ADR4/		54	ADR5/	
	55	ADR2/		56	ADR3/	
	57	ADR0/		58	ADR1/	
DATA	59	DATE/		60	DATF/	
	61	DATC/		62	DATD/	
	63	DATA/		64	DATB/	
	65	DAT8/	Data	66	DAT9/	Data
	67	DAT6/	Bus	68	DAT7/	Bus
	69	DAT4/		70	DAT5/	
	71	DAT2/		72	DAT3/	
	73	DAT0/		74	DAT1/	
POWER SUPPLIES	75	GND	Signal GND	76	GND	Signal GND
	77	—	Reserved	78	—	Reserved
	79	-12V	-12Vdc	80	-12V	-12Vdc
	81	+5V	+5Vdc	82	+5V	+5Vdc
	83	+5V	+5Vdc	84	+5V	+5Vdc
	85	GND	Signal GND	86	GND	Signal GND

1. All odd-numbered pins (1, 3, 5 . . . 85) are on component side of the board. Pin 1 is the left-most pin when viewed from the component side of the board with the extractors at the top. All unassigned pins are reserved.
2. Not Used on the iSBC 86/05 board.

Table A-2. Multibus® Interface Signal Functions

Signal	Functional Description
ADR0/-ADRF/ ADR10/-ADR13/	*Address.* These 20 lines transmit the address of the memory location or I/O port to be accessed. For memory access, ADR0/ (when active low) enables the even byte (DAT0/-DAT7/) on the Multibus interface; i.e., ADR0/ is active low for all even addresses. ADR13/ is the most significant address ibt.
BCLK/	*Bus Clock.* Used to synchronize the bus contention logic on all bus masters. When generated by the iSBC 86/12A board, BCLK/ has a period of 108.5 nanoseconds (9.22 MHz) with a 35-65 percent duty cycle.
BHEN/	*Byte High Enable.* When active low, enables the odd byte (DAT8/-DATF/) onto the Multibus interface.
BPRN/	*Bus Priority In.* Indicates to a particular bus master that no higher priority bus master is requesting use of the bus. BPRN/ is synchronized with BCLK/.
BPRO/	*Bus Priority Out.* In serial (daisy chain) priority resolution schemes, BPRO/ must be connected to the BPRN/ input of the bus master with the next lower bus priority.
BREQ/	*Bus Request.* In parallel priority resolution schemes, BREQ/ indicates that a particular bus master requires control of the bus for one or more data transfers. BREQ/ is synchronized with BCLK/.
BUSY/	*Bus Busy.* Indicates that the bus is in use and prevents all other bus masters from gaining control of the bus. BUSY/ is synchronized with BCLK/.
CBRQ/	*Common Bus Request.* Indicates that a bus master wishes control of the bus but does not presently have control. As soon as control of the bus is obtained, the requesting bus controller raises the CBRQ/ signal.
CCLK/	*Constant Clock.* Provides a clock signal of constand frequency for use by other system modules. When generated by the iSBC 86/12A board, CCLK/ has a period of 108.5 nanoseconds (9.22 MHz) with a 35-65 percent duty cycle.
DAT0/-DATF/	*Data.* These 16 bidirectional data lines transmit and receive data to and from the addressed memory location or I/O port. DATF/ is the most-significant bit. For data word operations, DAT0-DAT7/ is the even byte and DAT8/-DATF/ is the odd byte.
LOCK	*Dual Port RAM Lock.* Disables system dual Port RAM when asserted and jumper is installed See section 2-25.
INIT/	*Initialize.* Resets the entire system to a known internal state.
INTA/	*Interrupt Acknowledge.* This signal is issued in response to an interrupt request.
INT0/-INT7/	*Interrupt Request.* These eight lines transmit Interrupt Requests to the appropriate interrupt handler. INT0 has the highest priority.
IORC/	*I/O Read Command.* Indicates that the address of an I/O port is on the Multibus interface address lines and that the output of that port is to be read (placed) onto the Multibus interface data lines.
IOWC/	*I/O Write Command.* Indicates that the address of an I/O port is on the Multibus interface interface address lines and that the contents on the Multibus interface data lines are to be accepted by the addressed port.
MRDC/	*Memory Read Command.* Indicates that the address of a memory location is on the Multibus interface address lines and that the contents of that location are to be read (placed) on the Multibus interface data lines.
MWTC/	*Memory Write Command.* Indicates that the address of a memory location is on the Multibus interface address lines and that the contents on the Multibus interface data lines are to be written into that location.
XACK/	*Transfer Acknowledge.* Indictes that the addressed memory location has completed the specified read or write operation. That is, data has been placed onto or accepted from the Multibus interface data lines.

Table A-3. Connector P2 Pin Assignments

Pin Assignment	Signal Mnemonic	Description
P2-1, 2 P2-21, 22	Signal GND	Battery Ground
P2-3 P2-4	+5V AUX	Battery +5V Power Input
P2-6	Reserved	Reserved
P2-17	PFSN/	Power Fail Sense
P2-19	PFIN/	Power Fail Interrupt
P2-20	MPRO/	Memory Protect
P2-31	PLC	Power Line Clock
P2-32	ALE	Bus Master ALE
P2-36	BD RESET/	Board Only Reset
P2-38	AUX RESET/	System Reset Switch Input

Table A-4. Connector P2 Signal Definitions

PFIN/	*Power Fail Interrupt.* This input from the power supply interrupts the CPU when a power failure occurs. See section 2-33.
PFSN/	*Power Fail Sense.* This line is the output of a latch which indicates a power failure has occurred. It is reset by PFSR/ and must be powered by the standby power source. See section 2-33.
MPRO/	*Memory Protect.* When true, this externally generated signal prevents access to the on-board RAM during periods of uncertain DC power. See section 2-33.
ALE	*Address Latch Enable.* Indicates the 8086 CPU is operating. Typically, this signal is used to drive a front panel RUN indicator.
BD RESET/	*Board Reset.* This signal resets the iSBC 86/05 board only. It will not reset other boards in the system.
AUX RESET/	*Auxiliary Reset.* Typically this RESET signal is generated by a front panel switch. The signal is functionally equivalent to INIT/.

Table A-5. iSBC® 86/05 Board DC Characteristics

Signals	Symbol	Parameter Description	Test Conditions	Min.	Max.	Units
AACK/, XACK/	V_{OL}	Output Low Voltage	I_{OL}= 16 mA		.04	V
	V_{OH}	Output High Voltage	I_{OH} = -3 mA	2.0		V
	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.4V		-2.2	mA
	I_{IH}	Input Current at High V	V_{IN} = 2.4V		-1.4	mA
	*C_L	Capacitive Load			15	pF
ADR0/-ADRF/ ADR10/-ADR13/	V_{OL}	Output Low Voltage	I_{OL} = 32 mA		0.55	V
	V_{OH}	Output High Voltage	I_{OH} = 3 mA	2.4		V
	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.45V		-0.50	mA
	I_{IH}	Input Current at High V	V_{IN} = 5.25V		50	μA
	I_{LH}	Output Leakage High	V_O = 5.25V		-0.50	mA
	I_{LL}	Output Leakage Low	V_O = 0.45V		-0.50	mA
	*C_L	Capacitive Load			18	pF
BCLK/	V_{OL}	Output Low Voltage	I_{OL} = 59.5 mA		0.5	V
	V_{OH}	Output High Voltage	I_{OH} = -3 mA	2.7		V
	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.45V		-0.5	mA
	I_{IH}	Input Current at High V	V_{IN} = 5.25V		40	μA
	*C_L	Capacitive Load			15	pF
BHEN/	V_{OL}	Output Low Voltage	I_{OL} = 16 mA		0.4	V
	V_{OH}	Output High Voltage	I_{OH} = -2.0 mA	2.4		V
	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.4V		1.6	mA
	I_{IH}	Input Current at High V	V_{IN} = 2.4V		40	μA
	*C_L	Capacitive Load			15	pF
BPRN/	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.4V		-0.5	mA
	I_{IH}	Input Current at High V	V_{IN} = 5.25V		50	μA
	*C_L	Capacitive Load			18	pF
BPRO/	V_{OL}	Output Low Voltage	I_{OL} = 5.0 mA		0.45	V
	V_{OH}	Output High Voltage	I_{OH} = -0.4 mA	2.4		V
	*C_L	Capacitive Load			15	pF
BREQ/	V_{OL}	Output Low Voltage	I_{OL} = 50 mA		0.45	V
	V_{OH}	Output High Voltage	I_{OH} = -0.4 mA	2.4		V
	*C_L	Capacitive Load			10	pF
BUSY/, CBRQ/, INTROUT/ (OPEN COLLECTOR)	V_{OL}	Output Low Voltage	I_{OL} = 20 mA		0.45	V
	V_{IL}	Input Low Voltage			0.4	V
	V_{IH}	Input High Voltage		2.4		V
	I_{IL}	Input Current at Low V	V_{IN}= 0.45V		-0.5	mA
	I_{IH}	Input Current at High V	V_{IN} = 5.25V		40	μA
	*C_L	Capacitive Load			20	pF

Table A-5. iSBC® 86/05 Board DC Characteristics (continued)

Signals	Symbol	Parameter Description	Test Conditions	Min.	Max.	Units
CCLK/	V_{OL}	Output Low Voltage	I_{OL} = 60 mA		0.5	V
	V_{OH}	Output High Voltage	I_{OH} = -3 mA	2.7		V
	*C_L	Capacitive Load			15	pF
DAT0/-DAT7/	V_{OL}	Output Low Voltage	I_{OL} = 32 mA		0.45	V
	V_{OH}	Output High Voltage	I_{OH} = -5 mA	2.4		V
	V_{IL}	Input Low Voltage			0.80	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.45V		-0.20	mA
	I_{LH}	Output Leakage High	V_O = 5.25V		100	μA
	*C_L	Capacitive Load			18	pF
LOCK/	V_{OL}	Output Low Voltage	I_{OL} = 32		0.8	V
	V_{OH}	Output High Voltage	I_{OH} = -2	2.0		V
	*C_L	Capacitive Load			300	pF
INIT/ (SYSTEM RESET)	V_{OL}	Output Low Voltage	I_{OL} = 4.4 mA		0.4	V
	V_{OH}	Output High Voltage	OPEN COLLECTOR			
	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.4V		-4.2	mA
	I_{IH}	Input Current at High V	V_{IN} = 2.4V		-1.4	mA
	*C_L	Capacitive Load			15	pF
INT0/-INT7/	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.4V		-1.6	mA
	I_{IH}	Input Current at High V	V_{IN} = 2.4V		40	μA
	*C_L	Capacitive Load			18	pF
IORC/, IOWC/	V_{OL}	Output Low Voltage	I_{OL} = 32 mA		0.45	V
	V_{OH}	Output High Voltage	I_{OH} = 5 mA	2.4		V
	I_{LH}	Output Leakage High	V_O = 5.25V		100	μA
	I_{LL}	Output Leakage Low	V_O = 0.45V		-100	μA
	*C_L	Capacitive Load			15	pF
INTA/, MRDC/, MWTC/	V_{OL}	Output Low Voltage	I_{OL} = 30 mA		0.45	V
	V_{OL}	Output High Voltage	I_{OH} = -5 mA	2.4		V
	V_{IL}	Input Low Voltage			0.95	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.45V		-2.0	mA
	I_{IH}	Input Current at High V	V_{IN} = 5.25		1000	μA
	*C_L	Capacitive Load			25	pF

*Capacitive load values are approximations.

Table A-6. iSBC® 86/05 AC Characteristics

Parameter	Description	Minimum	Maximum	Units
t_{BCY}	Bus Clock Period	100	D.C.	ns
t_{BW}	Bus Clock Width	0.35 t_{BCY}	0.65 t_{BCY} (not restricted)	
t_{SKEW}	BCLK/skew		3	ns
t_{PD}	Standard Bus Propagation Delay		3	
t_{AS}	Address Set-Up Time (at Slave Board)	50		ns
t_{DS}	Write Data Set Up Time	50		ns
t_{AH}	Address Hold Time	50		ns
t_{DHW}	Write Data Hold Time	50		ns
t_{DXL}	Read Data Set Up Time To XACK	0		ns
t_{DHR}	Read Data Hold Time	0	65	ns
t_{XAH}	Acknowledge Hold Time	0	65	ns
t_{XACK}	Acknowledge Time	0	8	μs
t_{CMD}	Command Pulse Width	100	9.5	μs
t_{INTA}	INTA/ Width	250		ns
t_{CSEP}	Command Separation	100		ns
t_{BREQL}	↑BCLK/ to BREQ/ Low Delay	0	35	ns
t_{BREQH}	↓BCLK/ to BREQ/ High Delay	0	35	ns
t_{BPRNS}	BPRN/ to ↓BCLK/ Setup Time	22		ns
t_{BUSY}	BUSY/ delay from ↓BCLK/	0	70	ns
t_{BUSYS}	BUSY/ to ↓BCLK/ Setup Time	25		ns
t_{BPRO}	↓BCLK/ to BPRO/ (CLK to Priority Out)	0	40	ns
t_{BPRNO}	BPRN/ to BPRO/ (Priority In to Out)	0	30	ns
t_{CBRO}	↓BCLK/ to CBRQ/ (CLK to Common Bus Request)	0	60	ns
t_{CBRQS}	CBRQ/ to ↓BCLK/ Setup Time	35		ns
t_{XCD}	XACK↓ to Command Delay	0	1500	ns
t_{BSYO}	CBRQ/↓ and BUSY/↓ to BUSY/↑	—	12	μs
t_{CCY}	C-clock Period	100	110	ns
t_{CW}	C-clock Width	0.35 t_{CCY}	0.65 t_{CCY}	ns
t_{INIT}	INIT/Width	5		ms
t_{INITS}	INIT/ to MPRO/ Setup Time	100		ns
t_{PBD}	Power Backup Logic Delay	0	200	ns
t_{PFINW}	PFIN/ Width	2.5		ms
t_{MPRO}	MPRO/ Delay	2.0	2.5	ms

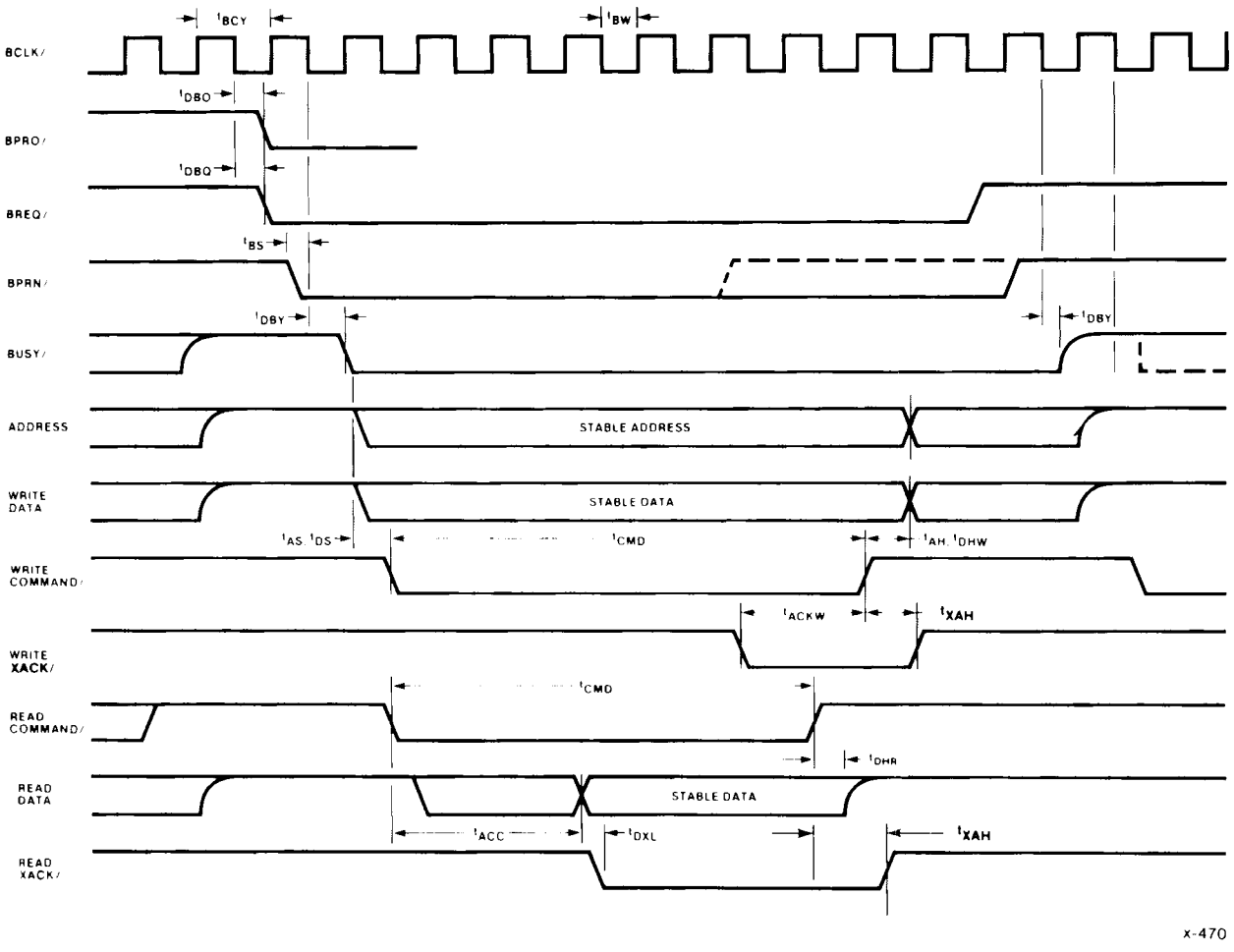


Figure A-1. iSBC® 86/05 AC Timing Diagram

Table A-7. Auxiliary Connector P2 Signal DC Characteristics

Signals	Symbol	Parameter Description	Test Conditions	Min.	Max.	Units
ALE	V_{OL}	Output Low Voltage	I_{OL} = 8 mA		0.45	V
	V_{OH}	Output High Voltage	I_{OH} = -1.0 mA	2.4		V
	*C_L	Capacitive Load			20	pF
PFIN/	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.4		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.4V		-0.4	mA
	I_{IH}	Input Current at High V	V_{IN} = 2.4V		20	μ
	*C_L	Capacitive Load			20	pF
MPRO/	V_{IL}	Input Low Voltage			0.80	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.45V		-6.0	mA
	I_{IH}	Input Current at High V	V_{IN} = 5.25V		250	μA
	*C_L	Capacitive Load			15	pF
AUX RESET/	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.6		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.45V		-0.25	mA
	I_{IH}	Input Current at High V	V_{IN} = 5.25V		10	μA
	*C_L	Capacitive Load			10	μF

*Capacitance load values are approximations.

Table A-8. Parallel I/O Signal DC Characteristics

Signals	Symbol	Parameter Description	Test Conditions	Min.	Max.	Units
Port C8 Bidirectional Drivers	V_{OL}	Output Low Voltage	I_{OL} = 32 mA		0.45	V
	V_{OH}	Output High Voltage	I_{OH} = -5 mA	2.4		V
	V_{IL}	Input Low Voltage			0.95	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.45V		-5.25	mA
	*C_L	Capacitive Load			18	pF
8255A Driver/Receiver	V_{OL}	Output Low Voltage	I_{OL} = 1.7 mA		0.45	V
	V_{OH}	Output High Voltage	I_{OH} = -200 μA	2.4		V
	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.45		10	μA
	I_{IH}	Input Current at High V	V_{IN} = 5.0		10	μA
	*C_L	Capacitive Load			18	pF
EXT INTR0/	V_{IL}	Input Low Voltage			0.8	V
	V_{IH}	Input High Voltage		2.0		V
	I_{IL}	Input Current at Low V	V_{IN} = 0.4V		-1.0	mA
	I_{IH}	Input Current at High V	V_{IN} = 2.4V		-0.8	mA
	*C_L	Capacitive Load			30	pF

*Capacitive load values are approximations.

Table A-9. iSBX™ Bus Connector Pin Assignments

Pin	Mnemonic	Description	Pin	Mnemonic	Description
43	MD8	MDATA 8	44	MD9	MDATA 9
41	MDA	MDATA A	42	MDB	MDATA B
39	MDC	MDATA C	40	MDD	MDATA D
37	MDE	MDATA E	38	MDF	MDATA F
35	GND	SIGNAL GROUND	36	+5V	+5 Volts
33	MD0	MDATA BIT 0	34	—	RESERVED
31	MD1	MDATA BIT 1	32	—	RESERVED
29	MD2	MDATA BIT 2	30	OPT0	OPTION 0
27	MD3	MDATA BIT 3	28	OPT1	OPTION 1
25	MD4	MDATA BIT 4	26	—	RESERVED
23	MD5	MDATA BIT 5	24	—	RESERVED
21	MD6	MDATA BIT 6	22	MCS0/	M CHIP SELECT 0
19	MD7	MDATA BIT 7	20	MCS1/	M CHIP SELECT 1
17	GND	SIGNAL GROUND	18	+5V	+5 Volts
15	IORD/	IO READ COMMAND	16	MWAIT/	M WAIT
13	IOWRT/	IO WRITE COMMAND	14	MINTR0	M INTERRUPT 0
11	MA0	M ADDRESS 0	12	MINTR1	M INTERRUPT 1
9	MA1	M ADDRESS 1	10	—	RESERVED
7	MA2	M ADDRESS 2	8	MPST/	M PRESENT
5	MRESET	M RESET	6	MCLK	M CLOCK
3	GND	SIGNAL GROUND	4	+5V	+5 Volts
1	+12V	+12 Volts	2	-12V	-12 Volts

Table A-10. iSBX™ Bus Signal Description

IORD/	Commands the Multimodule board to perform the read operation.
IOWRT/	Commands the Multimodule board to perform the write operation.
MRESET/	Initializes the Multimodule board to a known internal state.
MCS0/	Chip select. Selects I/O addresses 80-8F on the J4 Multimodule board and addresses A0-AF on the J3 Multimodule board.
MCS1/	Chip select. Selects I/O addresses 90-9F on the J4 Multimodule board and addresses B0-BF on the J3 Multimodule board.
MA0-2	Least three bits of the I/O address. Used in conjunction with the chip select and command lines.
MPST/	Multimodule present indicator. Informs iSBC 86/05 board that a Multimodule board(s) is installed.
MINTR0-1	Interrupt request lines from the Multimodule board to the iSBC 86/05 board interrupt matrix.
MWAIT/	Causes iSBC 86/05 board to execute wait states until Multimodule board is ready to respond.
MCLK/	9.68 MHz Multimodule board timing reference from iSBC 86/05 board.
OPT0-1	Optional use lines. May be used for additional interrupt request lines.
MD0-F	Bidirectional data lines.

***

APPENDIX B. DECODE PROM MEMORY MAPS

The iSBC 86/05 board uses one Intel pre-programmed PROM in the memory decode circuitry. Table B-1 is the PROM memory decode map for the memory decode PROM (U73). Table B-2 is the PROM memory decode map for the I/O decode PROM (U72).

Table B-1. Memory Decode PROM (U73)Map

000	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04
010	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04	04
020	05	05	05	05	05	05	05	05	0F	0F	0F	0F	0F	0F	0F	0F
030	05	05	05	05	05	05	05	05	0F	0F	0F	0F	0F	0F	0F	0F
040	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
050	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
060	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
070	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
080	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
090	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0A0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0B0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0C0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0D0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0E0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0F0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
100	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
110	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
120	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
130	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
140	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
150	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
160	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
170	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
180	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
190	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1A0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1B0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1C0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1D0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1E0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1F0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F

Table B-1. Memory Decode PROM (U73) Map (continued)

200	02	0F	0F	0F	0F	0F	0F	0F	02	0F	0F	0F	0F	0F	0F	0F
210	02	0F	0F	0F	0F	0F	0F	0F	02	0F	0F	0F	0F	0F	0F	0F
220	02	0F	0F	0F	0F	0F	0F	0F	02	0F	0F	0F	0F	0F	0F	0F
230	02	0F	0F	0F	0F	0F	0F	0F	02	0F	0F	0F	0F	0F	0F	0F
240	02	0F	0F	0F	0F	0F	0F	0F	02	0F	0F	0F	0F	0F	0F	0F
250	02	0F	0F	0F	0F	0F	0F	0F	02	0F	0F	0F	0F	0F	0F	0F
260	02	0F	0F	0F	0F	0F	0F	0F	02	0F	0F	0F	0F	0F	0F	0F
270	02	0F	0F	0F	0F	0F	0F	0F	02	0F	0F	0F	0F	0F	0F	0F
280	03	0F	0F	0F	0F	0F	0F	0F	03	0F	0F	0F	0F	0F	0F	0F
290	03	0F	0F	0F	0F	0F	0F	0F	03	0F	0F	0F	0F	0F	0F	0F
2A0	03	0F	0F	0F	0F	0F	0F	0F	03	0F	0F	0F	0F	0F	0F	0F
2B0	03	0F	0F	0F	0F	0F	0F	0F	03	0F	0F	0F	0F	0F	0F	0F
2C0	03	0F	0F	0F	0F	0F	0F	0F	03	0F	0F	0F	0F	0F	0F	0F
2D0	03	0F	0F	0F	0F	0F	0F	0F	03	0F	0F	0F	0F	0F	0F	0F
2E0	03	0F	0F	0F	0F	0F	0F	0F	03	0F	0F	0F	0F	0F	0F	0F
2F0	03	0F	0F	0F	0F	0F	0F	0F	03	0F	0F	0F	0F	0F	0F	0F
300	00	02	0F	0F	00	0F	0F	0F	00	02	0F	0F	00	0F	0F	0F
310	00	02	0F	0F	00	0F	0F	0F	00	02	0F	0F	00	0F	0F	0F
320	00	02	0F	0F	00	0F	0F	0F	00	02	0F	0F	00	0F	0F	0F
330	00	02	0F	0F	00	0F	0F	0F	00	02	0F	0F	00	0F	0F	0F
340	00	03	0F	0F	00	0F	0F	0F	00	03	0F	0F	00	0F	0F	0F
350	00	03	0F	0F	00	0F	0F	0F	00	03	0F	0F	00	0F	0F	0F
360	00	03	0F	0F	00	0F	0F	0F	00	03	0F	0F	00	0F	0F	0F
370	00	03	0F	0F	00	0F	0F	0F	00	03	0F	0F	00	0F	0F	0F
380	01	00	02	0F	01	00	0F	0F	01	00	02	0F	01	00	0F	0F
390	01	00	02	0F	01	00	0F	0F	01	00	02	0F	01	00	0F	0F
3A0	01	00	03	0F	01	00	0F	0F	01	00	03	0F	01	00	0F	0F
3B0	01	00	03	0F	01	00	0F	0F	01	00	03	0F	01	00	0F	0F
3C0	01	01	00	02	01	01	00	0F	01	01	00	02	01	01	00	0F
3D0	01	01	00	03	01	01	00	0F	01	01	00	03	01	01	00	0F
3E0	01	01	01	00	01	01	01	00	01	01	01	00	01	01	01	00
3F0	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01

Table B-2. I/O Decode PROM (U72) Map

000	06	0F	06	0F	06	0F	06	0F	06	0F	06	0F	06	0F	06	0F
010	06	0F	06	0F	06	0F	06	0F	06	0F	06	0F	06	0F	06	0F
020	06	0F	06	0F	07	0F	07	0F	06	0F	06	0F	07	0F	07	0F
030	06	0F	06	0F	07	0F	07	0F	06	0F	06	0F	07	0F	07	0F
040	04	0F	04	0F	0F	0F	0F	0F	0F	0F	0F	0F	04	0F	04	0F
050	04	0F	04	0F	0F	0F	0F	0F	0F	0F	0F	0F	04	0F	04	0F
060	04	0F	04	0F	0F	0F	0F	0F	0F	0F	0F	0F	05	0F	05	0F
070	04	0F	04	0F	0F	0F	0F	0F	0F	0F	0F	0F	05	0F	05	0F
080	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03	03
090	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02	02
0A0	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01	01
0B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0C0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0D0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0E0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
0F0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
100	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
110	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
120	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
130	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
140	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
150	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
160	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
170	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
180	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
190	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1A0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1B0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1C0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1D0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1E0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
1F0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F

Table B-2. I/O Decode PROM (U72) Map (continued)

200	07	0F	07	0F	0F	0F	0F	0F	07	0F	07	0F	0F	0F	0F	0F
210	07	0F	07	0F	0F	0F	0F	0F	07	0F	07	0F	0F	0F	0F	0F
220	07	0F	07	0F	0F	0F	0F	0F	07	0F	07	0F	0F	0F	0F	0F
230	07	0F	07	0F	0F	0F	0F	0F	07	0F	07	0F	0F	0F	0F	0F
240	05	0F	05	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
250	05	0F	05	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
260	05	0F	05	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
270	05	0F	05	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
280	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
290	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
2A0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
2B0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
2C0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
2D0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
2E0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
2F0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
300	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
310	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
320	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
330	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
340	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
350	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
360	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
370	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
380	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
390	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
3A0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
3B0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
3C0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
3D0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
3E0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F
3F0	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F	0F

***

APPENDIX C. iSBC® 86/05 BOARD DIFFERENCES

This Appendix presents the differences between the old version of the iSBC 86/05 board (143240) and the new version (145895). Table C-1 lists the difference in jumper designations between the old and the new versions of the iSBC 86/05 boards. Figure C-1 shows the jumper post layout of the old version. Figure 4-2 shows the jumper post layout of the newer version. Figure C-2 are those pages from the older schematic that have been changed to incorporate the new features of the newer version. Figure 4-3 is the schematic pages of the new version.

Table C-1. Jumper Differences

New Version Jumpers	Old Version Jumpers	Function
	E214 to E215	Provides LOCK/ signal to the Multibus connector P1 pin 25. The new version has changed the function of these jumper posts and an inductor (L1) added in its place.
E214 to E215		This is a function change to existing jumper posts from the old version to the new version. The new function of this jumper is for a reserved option for test purposes only. Do not use.
E215 to E216*		Jumper posts E216 has been added in the new version. It enables BUS AEN/ signal to enable Multibus address drivers.
E217 to E218*		These are new jumper posts added in the new version. When the fail-safe timer is used and the 86/05 board is delayed in obtaining access to the bus such that timeout occurs before bus access is gained, the generation of the READY signal is delayed until bus access is obtained. If connected, you must connect jumper E221 to E222.

Table C-1. Jumper Differences (continued)

New Version Jumpers	Old Version Jumpers	Function
E217 to E219		These are new jumper posts added in the new version. When the fail-safe timer is used and the 86/05 board is delayed in obtaining access to the bus such that timeout occurs before bus access is gained, the READY signal is generated without waiting for bus access. If connected, you must connect jumper E220 to E221.
E220 to E221		These are new jumper posts added in the new version. This jumper must be used in conjunction with E217 to E219 to route the ANDed XACK/ and BUS AEN/ signals to the Ready 1 line.
E221 to E222*		These are new jumper posts added in the new version. This jumper must be used in conjunction with E217 and E218 to route XACK/ to Ready 1 line of the 8284A chip.

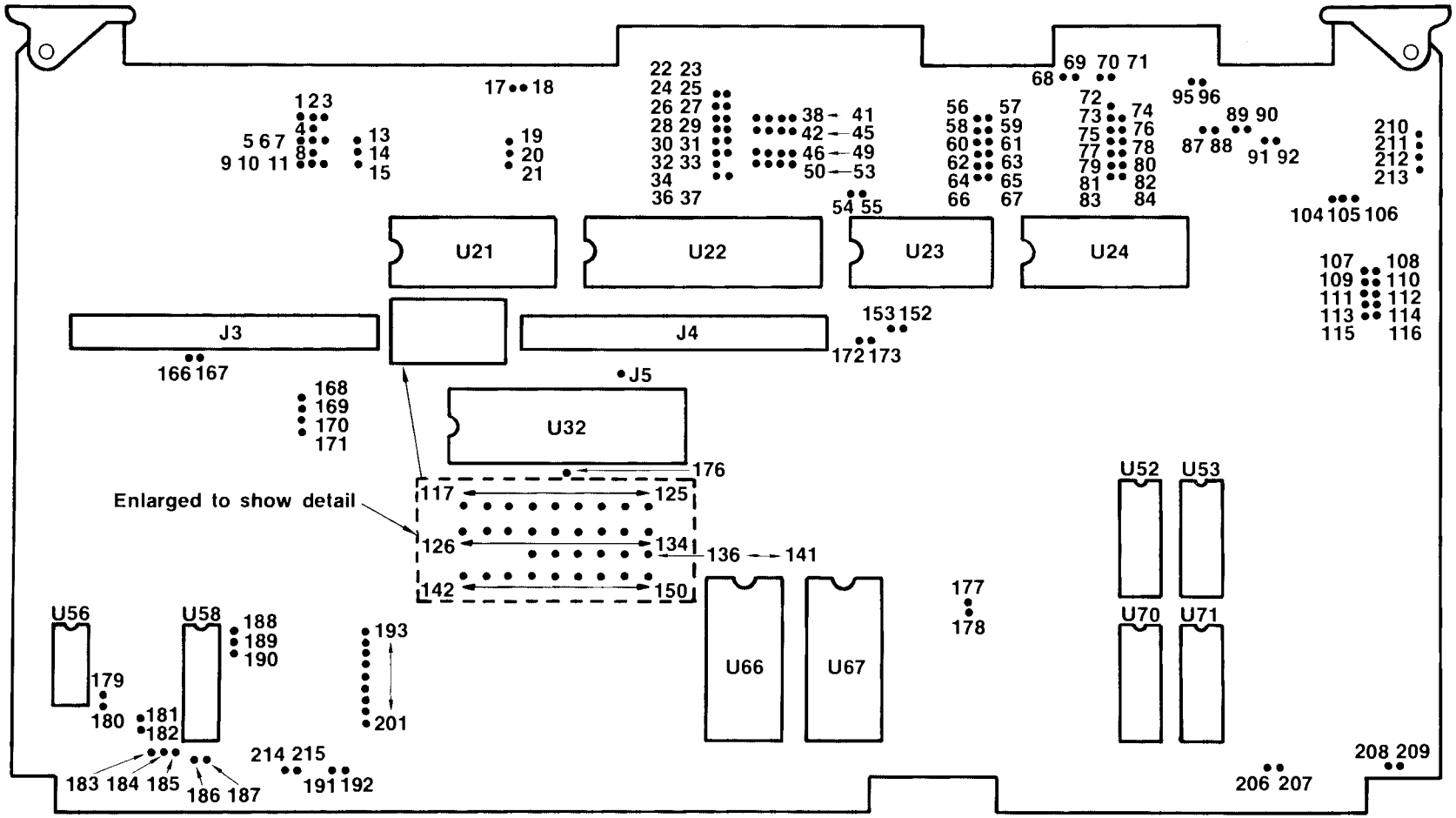


Figure C-1. iSBC® 86/05 Board Jumper Post Location Drawing

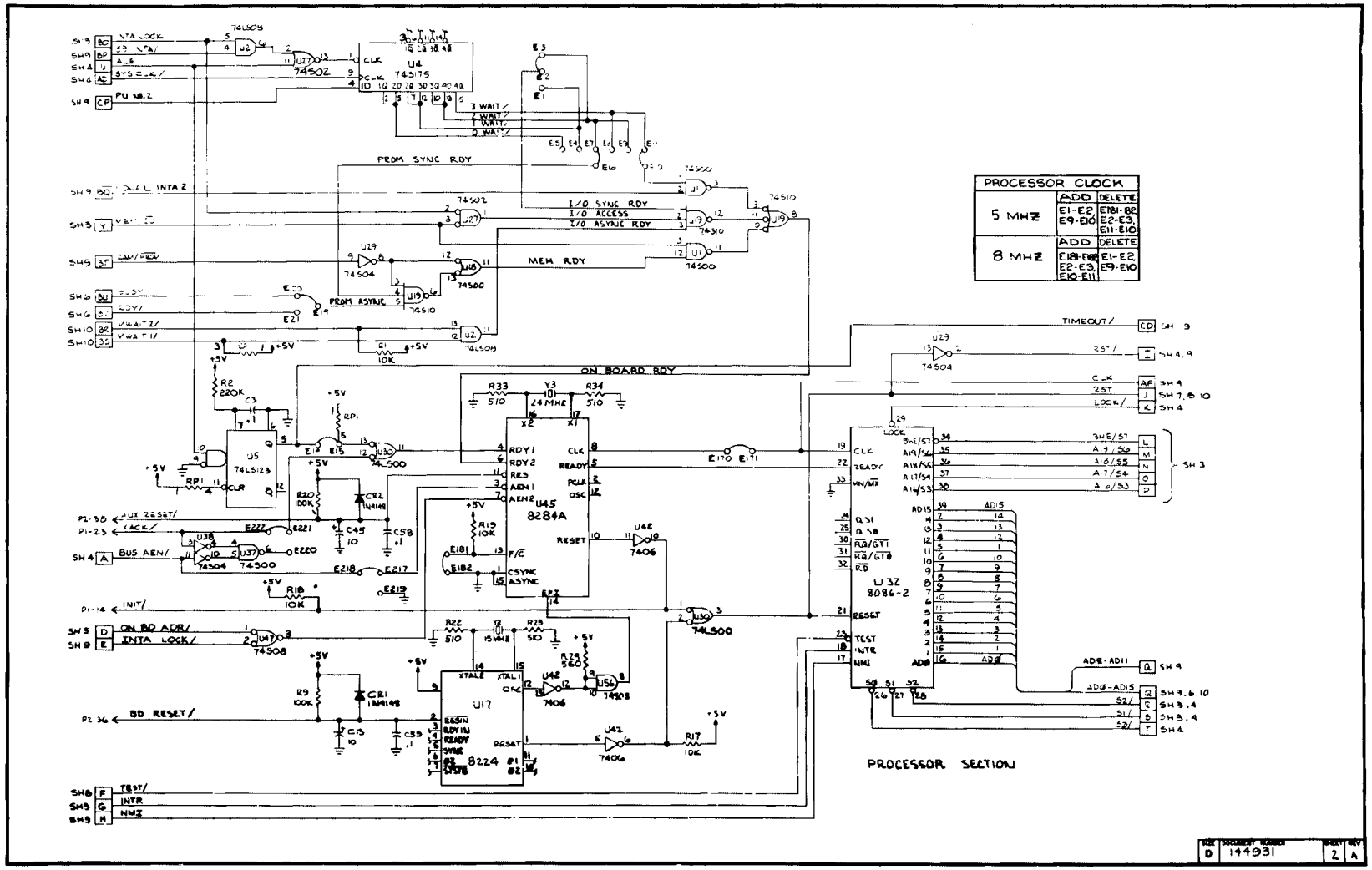


Figure C-2. iSBC® 86/05 Board Schematic Diagram (Sheet 1 of 2)

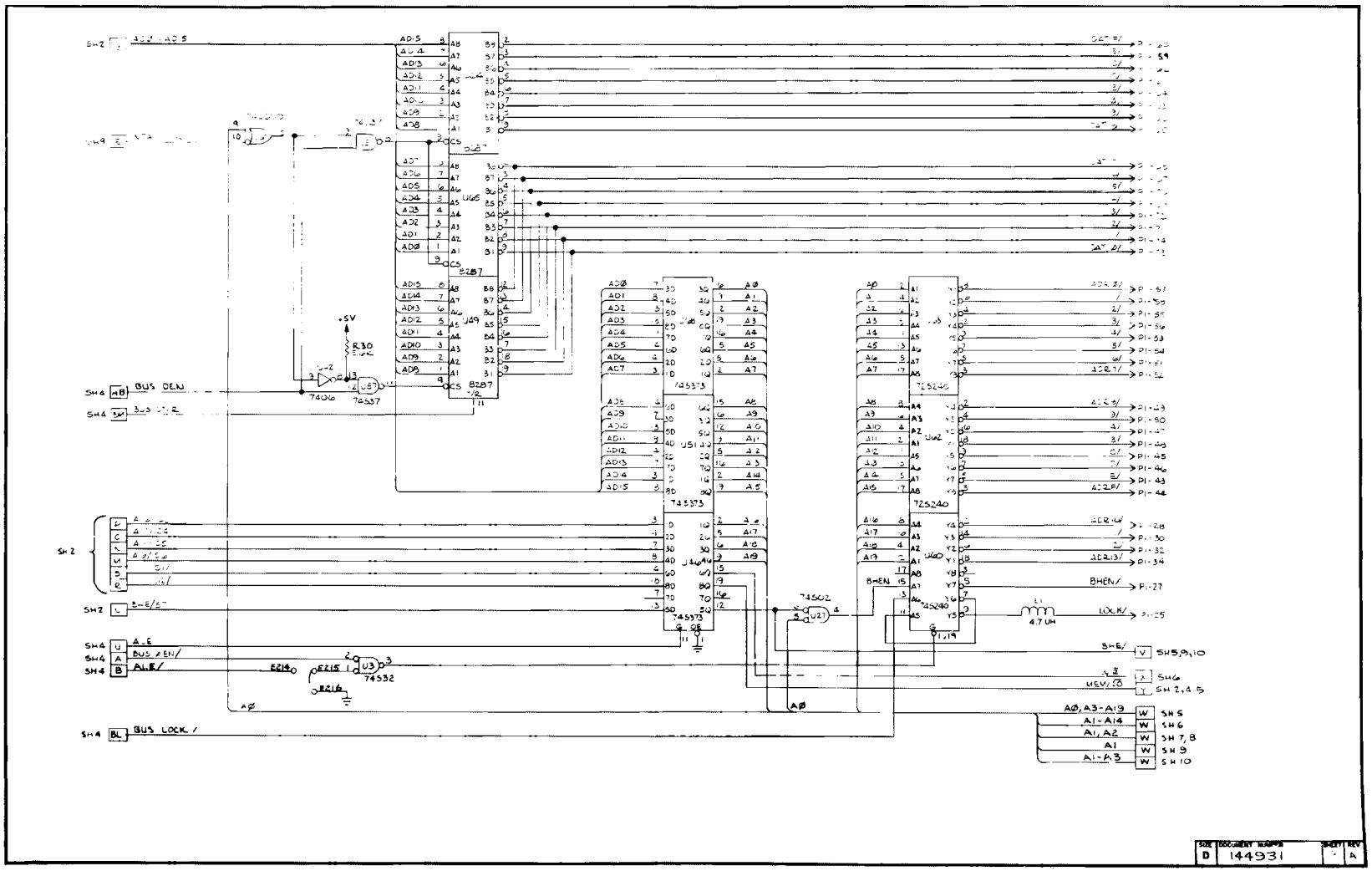


Figure C-2. iSBC® 86/05 Board Schematic Diagram (Sheet 2 of 2)

***

INDEX

REQUEST FOR READER'S COMMENTS

Intel's Technical Publications Departments attempt to provide publications that meet the needs of all Intel product users. This form lets you participate directly in the publication process. Your comments will help us correct and improve our publications. Please take a few minutes to respond.

Please restrict your comments to the usability, accuracy, readability, organization, and completeness of this publication. If you have any comments on the product that this publication describes, please contact your Intel representative. If you wish to order publications, contact the Intel Literature Department (see 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 suggestions for improvement.

3. Is this the right type of publication for your needs? Is it at the right level? What other types of 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. ☐

WE'D LIKE YOUR COMMENTS . . .

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

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 79 BEAVERTON, OR

POSTAGE WILL BE PAID BY ADDRESSEE

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

OMO Technical Publications