AD2000 User's Manual

Real Time Devices, Inc.

"Accessing the Analog World"™

ISO9001 and AS9100 Certified

AD2000
User's Manual

REAL TIME DEVICES, INC.
820 North University Drive
Post Office Box 906
State College, Pennsylvania 16804
Phone: (814) 234-8087
FAX: (814) 234-5218

Published by
Real Time Devices, Inc.
820 N. University Dr.
P.O. Box 906
State College, PA 16804

Printed in U.S.A.

Rev. B 9241

TABLE OF CONTENTS

Page

INTRODUCTION

How to Use This Manual....i-2
When You Need Help....i-2

CHAPTER 1 — QUICK START—GETTING YOUR AD2000 RUNNING

What Comes With Your AD2000....1-1
The Hardware....1-1
Functions You Can Set....1-1
Setting the Base I/O Address....1-2
Installing the AD2000 in Your Computer....1-4
The Software....1-6
Demo Disk....1-6
Backing Up Your Disk....1-6
Initializing Your AD2000....1-7
Selecting an Analog Input Channel....1-8
Setting the Input Gain....1-9
Taking an A/D Reading....1-10

CHAPTER 2 — FUNCTIONAL DESCRIPTION

Analog-to-Digital Conversion Circuitry....2-1
Multiplexers....2-1
Gain Control Circuitry....2-2
Sample and Hold Circuitry....2-2
A/D Converter....2-2
Programmable Peripheral Interface....2-3
Programmable Interval Timer (PIT)....2-3

CHAPTER 3 — JUMPER SETTINGS

S1 — Analog Input Signal Type DIP Switch....3-2
P2 — Base I/O Address Header Connector....3-2
P3 — Programmable Interval Timer (PIT) I/O Header Connector....3-2

P4, P5, and P7 — Interrupt Header Connectors....3-4
P4 — EXTINT and PPI INTRA Interrupts....3-5
P5 — PIT Output Interrupts....3-5
P7 — A/D End-of-Convert (EOC) Interrupt....3-5
P6 — End-of-Convert (EOC) Monitor Header Connector....3-6
P9 — A/D Converter Voltage Range Header Connector....3-6

Page

CHAPTER 4 — PROGRAMMING YOUR AD2000

Selecting an Analog Input Channel.................... 4-1
Setting the Input Gain.................... 4-1
Taking an A/D Reading.................... 4-1
Programming the Programmable Peripheral Interface.................... 4-1
Programming the Programmable Interval Timer.................... 4-2
Hardware Interrupts.................... 4-3
A/D End-of-Convert (EOC) Signal.................... 4-3
PPI Interrupts.................... 4-3
PIT Interrupts.................... 4-3

CHAPTER 5 — CALIBRATION PROCEDURES

Required Equipment.................... 5-1
A/D Calibration.................... 5-1
Unipolar Calibration.................... 5-2
Bipolar Calibration.................... 5-2
Gain Circuitry Calibration.................... 5-3

APPENDIX A — AD2000 Specifications....................A-1
APPENDIX B — Connector Pin Assignments....................B-1
APPENDIX C — Component Data Sheets....................C-1
APPENDIX D — Configuring the AD2000 for SIGNAL*MATH....................D-1
APPENDIX E — Configuring the AD2000 for ATLANTIS....................E-1
APPENDIX F — Warranty....................F-1

LIST OF ILLUSTRATIONS

Figure		Page
i-1	Typical Laboratory Setup	i-1
1-1	AD2000 Board Layout	1-2
1-2	Base I/O Address Header Connector, P2	1-4
1-3	A/D Conversion Word Format	1-10
2-1	AD2000 Functional Block Diagram	2-1
2-2	EOC Timing Diagram	2-2
3-1	AD2000 Board Layout	3-1
3-2	DIP Switch S1	3-2
3-3	PIT I/O Header Connector P3	3-3
3-4	PIT Functional Block Diagram	3-4
3-5	Interrupt Header Connector P4	3-5
3-6	Interrupt Header Connector P5	3-5
3-7	Interrupt Header Connector P7	3-6
3-8	EOC Monitor Header Connector P6	3-6
3-9	A/D Converter Voltage Range Header Connector P9	3-6
4-1	PPI Mode Definition Format	4-2
5-1	AD2000 Board Layout	5-1

LIST OF TABLES

Table		Page
1-1	AD2000 Board Functions and Settings	1-3
1-2	AD2000 I/O Map	1-5
5-1	A/D Converter Bit Weights	5-3
5-2	A/D Converter Readings for Gain Calibration	5-3

INTRODUCTION

This manual shows you how to operate and provides technical data for Real Time Devices' AD2000 multifunction data acquisition board. The AD2000 features 12-bit high-speed multichannel differential or single-ended analog-to-digital conversion. This versatile interface allows your IBM PC/XT/AT or compatible computer to effectively operate in the real-time environment of data acquisition and control to sense and generate analog and digital signals. Figure i-1 shows a typical laboratory setup using a PC for data collection.

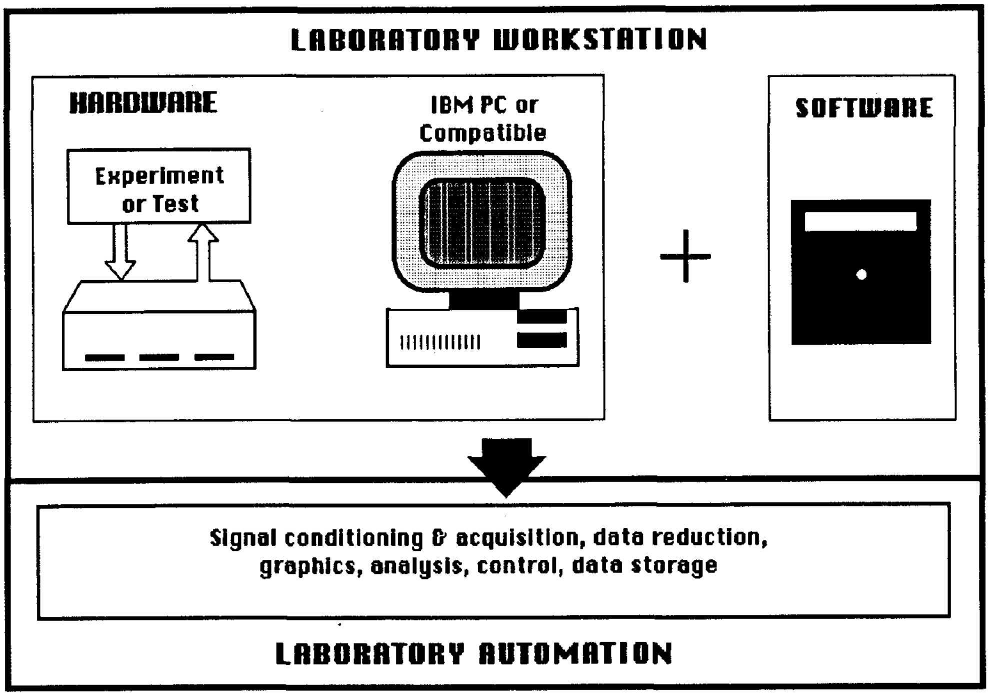


Fig. i-1 — Typical Laboratory Setup

The AD2000 features a high-resolution (12-bit) analog-to-digital and converter, digital I/O, and timer/counters that provide flexibility for many applications. Its six-layer construction, including separate power and ground planes, enhances board performance and low-noise characteristics. It plugs directly into any unused expansion slot (short or full-size) in the computer. All external I/O connections, including PC bus-sourced power, are accessible at the rear panel of the computer when the board is installed.

Several of the AD2000's functions can be readily adapted for your specific requirements. Through programming and/ or jumper or switch settings made on the board, you can:

- Select the base I/O address,
- Choose 8 differential or 16 single-ended analog input channels,
- Select the active channel,
- Select the channel gain,
- Select the analog input voltage range and polarity,
- Control 16 TTL/CMOS-compatible digital I/O lines,
- Control three 16-bit, 8 MHz timer/counter circuits (the programmable interval timer),
- Monitor the A/D conversion using the end-of-convert (EOC) signal,
- Generate interrupt signals.

Many of these functions are set up at the factory, based on typical data collection requirements and customer specifications when ordering. Therefore, you can successfully install and run the AD2000 with minimal understanding about changing and controlling them. On the other hand, you may want to understand everything about your board so that you can effectively use each feature. With this in mind, this manual provides basic information to get the board up and running, as well as detailed information for a full understanding of each function.

How to Use This Manual

This manual is designed to help you install and get your AD2000 running quickly, while also including sufficient detail about each board function. Begin by reading Chapter 1 in order to use your board as quickly as possible. This chapter and the accompanying demonstration software included with your AD2000 package will allow you to promptly use your interface. To fully understand and control the AD2000 functions, read Chapters 2 through 4. Chapter 5 contains board calibration procedures.

The chapters and appendixes in this manual are described in detail below.

> Chapter 1, "Quick Start—Getting Your AD2000 Running," provides the instructions necessary to install the board and use its basic functions. The information contained in this chapter does not cover how to change the board setup, except for the base I/O address.
>
> Chapter 2, "Functional Description," provides a block diagram and a functional discussion of the board.
>
> Chapter 3, "Jumper Settings," describes each header or jumper circuit on the board and how it is controlled.
>
> Chapter 4, "Programming Your AD2000," describes how the board can be programmed using the demonstration software.
>
> Chapter 5, "Calibration Procedures," provides instructions for board calibration.
>
> Appendix A, "AD2000 Specifications," contains a complete listing of board specifications.
>
> Appendix B, "Connector Pin Assignments," contains the pinouts of the external I/O connectors and the mating connectors' part numbers.
>
> Appendix C, "Component Data Sheets," contains manufacturers' data sheets for major board components.
>
> Appendix D, "Configuring the AD2000 for SIGNAL*MATH," contains information about setting board jumpers and and initializing the board to run the SIGNAL*MATH acquisition and analysis program.
>
> Appendix E, "Configuring the AD2000 for ATLANTIS," contains information about setting board jumpers to run the ATLANTIS data acquisition and real-time monitoring program.
>
> Appendix F, "Warranty," contains board warranty information.

When You Need Help

When you are working with the AD2000 interface board, this manual and the demo software included in your package will provide sufficient information to properly control all of the board's functions. If, however, after carefully reviewing the manual, you are unable to obtain proper responses from the board, Real Time Devices' technical staff is ready to assist you. For assistance, call (814) 234-8087 during regular business hours, eastern standard time or eastern daylight time, or send a FAX requesting assistance to (814) 234-5218. Be sure to include your company's name, your name, your telephone number, and a brief description of the problem.

CHAPTER 1

QUICK START—GETTING YOUR AD2000 RUNNING

To get started using your AD2000 interface board, you must:

- Select by jumper a base I/O address which does not contend with any other peripheral device.
- Install the board into your PC.
- Connect a signal to one of the analog input channels.
- Run the AD2000 software.

Unless you have other requirements, these steps are all that are necessary to use your AD2000 board.

This chapter explains how to install your AD2000 and use its basic functions. You will learn how to:

- Change the base I/O address setting,
- Install the board in your PC,
- Initialize the board,
- Select the analog input channel and gain,
- Take an A/D reading.

This chapter allows you to immediately start using the basic functions of your AD2000 board for data collection applications. This chapter does not explain how to control the more intricate board functions such as the programmable interval timer, the various digital I/O configurations, or interrupts, nor does it explain how to change hardware-controlled settings except for the base I/O address. The functions not covered here are described in Chapters 2 through 4.

What Comes With Your AD2000

The standard AD2000 board package includes:

1	AD2000 5.5-inch (140mm) interface board (fits short slot)
1	AD2000 demo disk
1	user's manual

Additional items, such as the AD2000 2-cable set (order number XK40-1), extender boards or SIGNAL*MATH or ATLANTIS application software, are available for this board and are included on an as-ordered basis.

All signals on your board are made easily accessible with Real Time Devices' XB40 I/O extender board and XC40 expansion cable. The extender board has two 20-pin terminal strips and a prototype area to support any special circuitry you may require to condition the signals. For example, if you are prototyping solid-state relays or optoisolators, this can easily be done with an XB40. The expansion cable terminates in a 40-pin wire-wrap header connector suitable for installation in standard 0.1 inch spacing perf-board material available from most electronic distributors.

The Hardware

The AD2000 interface board is shown in Figure 1-1. A complete listing of the board specifications is contained in Appendix A. The AD2000 has several features which are user-controlled through hardware or software. Most of the hardware-controllable features are jumper-controlled; the remaining are switch-controlled.

All of the board components are mounted on a 5.5-inch printed circuit board which fits in any unused expansion slot (short or full-size) in an IBM PC/XT/AT or compatible computer. Two 40-pin connectors on the board, P8 and P15, accommodate all of the board's external I/O. In operation, these connectors are cabled so that all 80 lines are accessible at the rear panel of the computer (see the board installation instructions later in this chapter).

Functions You Can Set

To allow the AD2000 interface board to be adapted to your needs, several functions can be set up to perform specific tasks by changing the hardware configuration or through software. Table 1-1 lists each function you can control, the factory (or default) setting if applicable, and where in this manual you can find information about its settings.

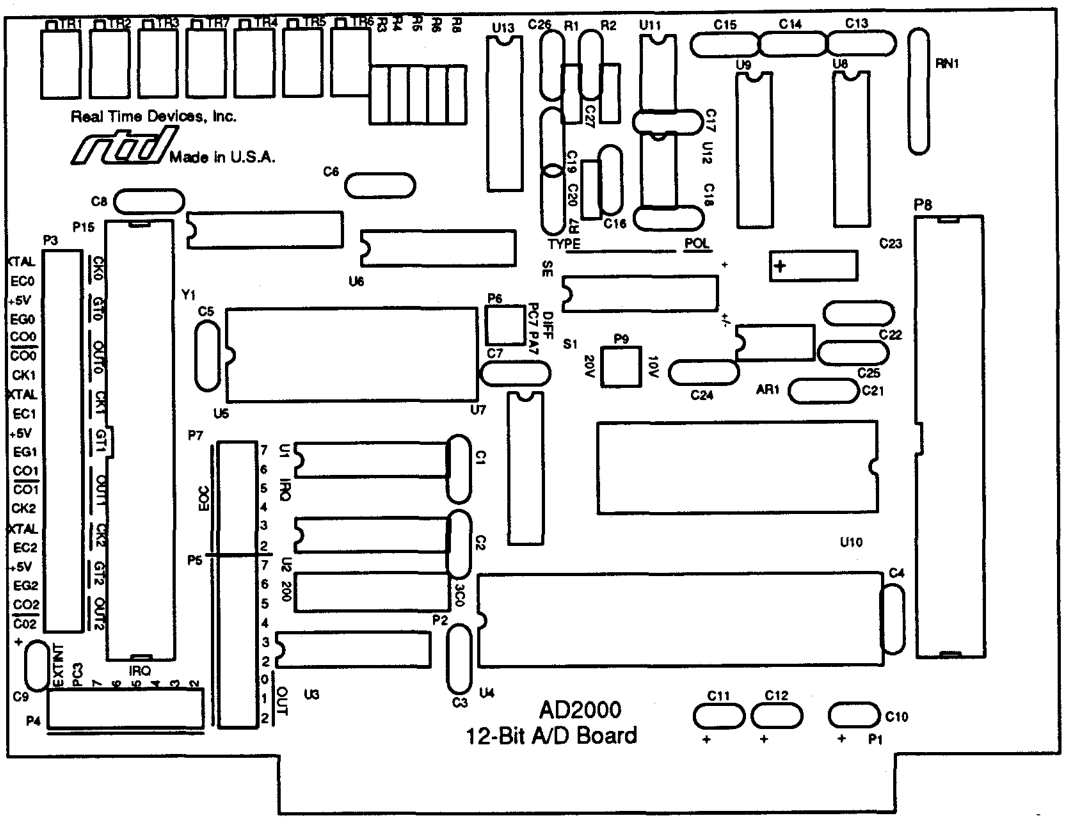


Fig. 1-1 — AD2000 Board Layout

The functions which you can control through hardware are:

- Base I/O address,
- Analog input channel type,
- Analog input channel voltage range and polarity,
- End-of-convert monitor,
- PIT timer/counters (hardware and software),
- Interrupts.

The functions which you can control through software are:

- Analog input channel selection,
- Analog input gain selection,
- Digital I/O,
- PIT timer/counters (software and hardware),
- Board initialization.

Setting the Base I/O Address

Starting with the base I/O address (BA), the AD2000 board uses 12 address locations in your computer's I/O space. Table 1-2 lists the I/O map for the AD2000. It is important to recognize that some of your computer's I/O address locations will already be occupied by internal I/O and other peripherals. If your AD2000 board tries to use I/O address locations already in use by another device in your system, address contention will result. Hence, the board will not operate, or at best will operate erratically.

I/O address contention is one of the most common problems encountered when adding an interface device to your computer system. To avoid this problem, a base I/O address jumper circuit is provided on the AD2000 board. By changing the position of the jumper on the header connector labeled P2 (located just to the left of center, near the bottom of the board), the base I/O address setting can be changed to any one of eight locations.

Table 1-1—AD2000 Board Functions and Settings

FUNCTION	FACTORY SETTING	USER INFORMATION
Base I/O Address	300 hex (768 decimal)	To change this setting, see "Setting the Base I/O Address," Chapter 1
Analog Input Channel Type	8 differential channels	To select 16 single-ended channels, see S1 discussion, Chapter 3
Analog Input Channel Selection	Software-controllable	See "Selecting an Analog Input Channel," Chapter 1, and demo disk
Analog Input Gain Selection	Software-controllable	See "Setting the Input Gain," Chapter 1, and demo disk
Analog Input Voltage Range and Polarity	User-specified when ordering	To change these settings, see S1 and P9 discussions, Chapter 3.
End-of-Convert (EOC) Monitor	Connected to PA7	See P6 discussion, Chapter 3.
Digital I/O		
16 I/O Lines from PPI	Software-controllable	See "Programming the PPI," Chapter 4 and demo disk
Programmable Inverval Timer (PIT) Circuitry		
Modes	Software-controllable	See "Programming the PIT," Chapter 4 and demo disk
I/O Configuration	Clock Input: 5 MHz Gate Input: +5 V Clock Output: To P8	See P3 discussion, Chapter 3
Interrupts	Disabled	See P4, P5, and P7 discussions, Chapter 3, and "Interrupt Considerations," Chapter 4

Figure 1-2 shows the base I/O address header connector, P2, with the jumper installed at the factory-set location of 300 hex. The jumper must be installed vertically across one of the eight pairs of pins on P2. The hexadecimal base I/O address setting corresponding to each pair of pins, from left to right, is as follows:

200 240 280 2C0 300 340 380 3C0

For example, if the base I/O address is changed to 280 hex, then for the 12 operations listed in Table 1-2, BA equals 280. Thus, to send the channel selection and gain data to port B of the PPI, its address of BA + 1 becomes 281 hex.

If the factory setting of 300 hex will cause contention in your system, position the jumper to the desired base I/O address setting. Once you have set the base I/O address, make a note of its value on the table inside the back cover of this manual. You will need to know this setting for use in your programs.

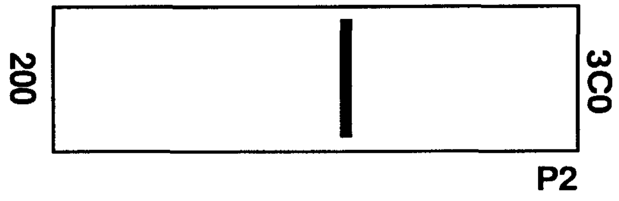


Fig. 1-2 — Base I/O Address Connector, P2

Installing the AD2000 in Your Computer

Before installing the AD2000 in your computer, make sure that the base I/O address has been properly selected and all the hardware settings have been configured to support your requirements. This chapter explains how to control the base I/O address. Other hardware settings are set at the factory, as listed in Table 1-1, and remain at their factory settings unless you change them. The interrupts generated by your AD2000 are disabled (not connected) when you receive your board. If you intend to use the interrupts, they must be configured appropriately before installing the board. Information about these and other functions not covered in this chapter is provided in Chapters 2 through 4. Use these chapters as necessary to configure your board before installation.

To install your AD2000, follow these step-by-step procedures:

1. TURN OFF THE POWER TO YOUR COMPUTER FIRST. Refer to the owner's manual for your computer, and remove the top cover.
2. Select an unused expansion slot (short or full-size) in which to install your board and remove its corresponding blank bracket from the rear panel of the computer by removing the screw at the top of the bracket.
3. Before placing the board into the computer, two ribbon cable assemblies must be installed on board connectors P8 and P15. If you have purchased the AD2000 cable set, first install the twisted pair cable on analog I/O connector P8. Then install the standard cable on P15. Each cable is a 40-line external I/O cable which extends through the connector slot in the rear panel of the computer. Both cables run through a single slot where they provide 80 lines of external I/O to the board. This configuration allows substantial board I/O through a single expansion port in your computer. Appendix B lists the signal carried on each pin of these connectors. To install the cables:
   a. Remove the strain relief clamp attached to the AD2000 bracket located on the right side of the board.
   b. Connect the socket connector to board connector for each cable. When installing, observe the connector keying and press firmly to make sure that the socket connector is fully seated on the board. Each cable provided is labeled with the connector's P number for easy identification. The cables have strain reliefs on one connector and not on the other. The connector without the strain relief is to be installed on the board. After both cables are installed on the board, position them so that they pass over the flange in the board's bracket.

Table 1-2—AD2000 I/O Map

FUNCTION	A4	A3	A2	A1	A0	R/W	BA + HEX
PPI							
Port A	0	0	0	0	0	R/W	0
Port B (Channel Sel & Gain)	0	0	0	0	1	W	1
Port C	0	0	0	1	0	R/W	2
Control Word	0	0	0	1	1	W	3
A/D Conversion Circuitry							
Start 12-bit Conversion	0	0	1	x	0	W	4 or 6
Start 8-bit Conversion	0	0	1	x	1	W	5 or 7
Read MSB	0	0	1	x	0	R	4 or 6
Read LSB	0	0	1	x	1	R	5 or 7
Programmable Interval Timer							
Counter 0	1	0	1	0	0	R/W	14
Counter 1	1	0	1	0	1	R/W	15
Counter 2	1	0	1	1	0	R/W	16
Control Word	1	0	1	1	1	W	17

NOTE: x = don't care setting

c. Re-attach the clamp to the bracket using the hardware supplied with your AD2000, securing the ribbon cables in place.

4. After checking that the cables are correctly installed on the board, orient the board inside the computer so that the cables extend through the rear panel opening and the card edge connector lines up with the expansion slot connector. Then, press down on the metal bracket tab and the top of the board until the board is firmly seated in the expansion slot connector.
5. Secure the bracket back in place with the screw and put the cover back on your computer.

Now your board is ready to be connected via the external connectors at the rear of the computer. After these connections have been made, the board is ready for operation.

The Software

The AD2000 operates under software control. Programming includes the analog input channel selection and gain, control of the the A/D conversion, the programmable peripheral interface, and the programmable interval timer. The analog input channel and gain selections and taking an A/D reading are covered in this chapter. Digital I/O control and control of the programmable interval timer are more complex, and are described in Chapter 4, "Programming Your AD2000."

Regardless of what programming language you use, you can write programs that control the AD2000 board. The demonstration disk which accompanies your AD2000 contains examples in Turbo C, Turbo Pascal, and BASIC.

Nearly all modern MS-DOS-based PC languages have I/O reference instructions. These are the instructions to control the data transfers to and from the I/O ports. Consult your programming language reference to find these instructions for your favorite language. Listed below are the I/O reference instructions used by some common languages.

	BASIC	TURBO PASCAL	TURBO C
input:	INP	Port	inportb
output:	OUT	Port	outportb

Demo Disk

Included with your AD2000 is a demo disk which provides programming instructions and example programs for controlling the functions of your interface board. This demo disk is divided into directories, each of which is named according to the language used to write the programs it contains. The files within each directory contain example programs and a documentation file with general information. In addition, your demo disk contains a README.DOC file which provides programming information for your board.

Each example program shows you how to control a particular board function, such as selecting an input channel or input gain, controlling the A/D converter, controlling digital data transfers, and setting the timer/counter circuitry. These programs should be used to become familiar with these functions.

Backing Up Your Disk

The demo disk provided with the AD2000 is a double-sided format which can be read by all DOS versions 1.1 and above. Before using the software included with your board, make a backup copy of the disk. You may make as many backups as you need. To copy the original to any other DOS-formatted disk, insert the disk to be copied into drive A of your computer, and from DOS enter:

```
COPY A:*.* B:  (or other destination drive specifier)
```

Initializing Your AD2000

Before you can operate the AD2000, it must be initialized. This step must be executed every time you start up, reset, or reboot the computer. This sets up the PPI to properly communicate with the A/D converter circuitry. If the board is not initialized, it will not respond to the software commands and will probably lock up, requiring you to reboot your system.

As described earlier, the AD2000 uses 12 address locations in the computer's I/O space. These address locations start with the base I/O address (BA) and go through BA + 17 (hex). BA + 8 through BA + 13 are not used. Table 1-2 provides the AD2000 I/O map, defining what function each of the 12 addresses controls. Recall that the base I/O address is factory-set at 300 hex. On the demo disk, the base I/O address is usually stored in the variable "board." Remember to use the correct base I/O address in the demo disk programs or your own programs. The demo disk explains how to change the base I/O address in the programs.

The AD2000 is initialized by simply writing a control byte to the PPI control register mapped at the I/O location base address + 3 (hex). The control byte must conform to this general form:

1xxx x00x   where x = don't care

This ensures that the eight I/O lines making up port B of the PPI, which are used to control the multiplexer and gain circuitry, are configured as outputs. The don't care (x) positions control the direction of the remaining 16 digital I/O lines available on the PPI. These lines can be configured as inputs, outputs, or in other more complex configurations.

For example, when the control byte bit pattern is:

100000000   (decimal 128)

the AD2000 is initialized as follows:

```
out base_address+3,128
```

When this value is used to initialize the AD2000, the eight port C lines of the PPI will all be configured as outputs. You can transfer data to these lines with the command:

```
out base_address+2,data
```

If instead, the decimal value 137 (1000 1001) is used to initialize the AD2000, the port C lines will be set up as inputs. You can input data from port C with the command:

```
data = inp(base_address+2)
```

Note that port A, bit 7 (PA7) of the PPI is factory-set to monitor the end-of-convert (EOC) signal. The PPI must be programmed so that port A is an input if you are going to monitor the EOC signal through PA7. The control byte must then conform to the general form of 1xx1 x00x, where the underlined 1 is the data bit which sets up port A as an input.

A functional description of the PPI is contained in Chapter 2, "Functional Description," and hardware configurations are described in Chapter 3, "Jumper Settings." Information about how you can control the digital I/O lines is contained in Chapter 4, "Programming Your AD2000," and is not covered here because of its complexity.

As mentioned earlier, the eight lines of port B are used to select the analog input channel and gain. The four LSBs, PB (for Port B) 0 through PB3, control the channel selection, and the four MSBs, PB4 through PB7, control the gain selection. The bit assignment of this port is:

MSBs	LSBs	
$\underbrace{7\ 6\ 5\ 4}$	$\underbrace{3\ 2\ 1\ 0}$	PPI Port B (Base Address +1)
gain select	channel select	
0000 = 1x	0000 = channel 1	
0001 = 2x	0001 = channel 2	
0010 = 4x	0010	
0100 = 8x	0011	
1000 = 16x	0100	
	0101	
	0110 .	
	0111 .	
	1000 .	
	1001	
	1010	
	1011	
	1100	
	1101	
	1110 = channel 15	
	1111 = channel 16	

After the AD2000 is initialized, the port B register is loaded with the default setting of 0000 0000. This selects channel 1 as the input channel with a gain of 1. To change this value, for example, to a gain of 2x on channel 16, enter these commands:

BA + 1 (hex)	selects port B
0001 1111	sets gain to 2x and channel to 16

Recall that the board's default channel setting is eight differential channels. Therefore, only the channel select binary values for channels 1 through 8 apply. Channels 9 through 16 are used in the single-ended channel mode only.

Now your board is initialized and ready to operate. The following sections describe how to select the analog input channel, set the input gain, and take an A/D reading. Mastering these operations will allow you to effectively use your board for data acquisition applications.

Selecting an Analog Input Channel

After the AD2000 has been initialized you can select the analog input channel. The analog input channel is selected by writing to port B of the PPI, mapped at I/O location base address (BA) + 1.

The input channel and the input gain can be set individually by setting only the four LSBs (channel select) or only the four MSBs (gain) of the eight-bit control word sent to port B. Before you change either the input channel or the gain, you MUST preserve the current state of port B. Failure to do so will result in changing both the channel select and the gain when you intended to change only one of these two settings.

The general algorithm for setting the channel (changing just the four LSBs of the control word while preserving the four MSBs) is:

```
1. Read the current state of port B:
        current_state = inp(base_address+1)
2. Preserve the upper four bits since they contain gain data:
        current_state = current_state AND $F0
3. Logically OR the current_state with the desired channel number minus 1:
        current_state = current_state OR (channel - 1)
4. Write it back out to port B:
        out base_address+1,current_state
```

A BASIC program to select channel 2 is:

```
100 BASE_ADDRESS% = 768
110 CHANNEL% = 2
120 STATUS% = INP(BASE_ADDRESS% + 1)
130 STATUS% = STATUS% AND &HF0
140 STATUS% = STATUS% OR (CHANNEL% -1)
150 OUT BASE_ADDRESS% + 1,STATUS%
```

Setting the Input Gain

The gain is set by writing to the upper four bits of port B at BA + 1. The bit pattern for each of the five gain values supported by the hardware are:

0000 = gain of 1
0001 = gain of 2
0010 = gain of 4
0100 = gain of 8
1000 = gain of 16

It is recommended that no other bit patterns be used when setting the gain.

The general algorithm for setting the gain is:

```
1. Read the current state of port B:
        current_state = inp(base_address+1)
2. Preserve the lower four bits since they contain channel information:
        current_state = current_state AND $0F
3. Logically OR the current_state with a bit pattern that activates the desired gain:
        current_state = current_state OR gain bit pattern:
                1x bit pattern = 0
                2x bit pattern = 16
                4x bit pattern = 32
                8x bit pattern = 64
                16x bit pattern = 128
4. Write the current_state back to port B:
        out base_address+1,current_state
```

A BASIC program to set a gain of 2 is:

```
100 BASE_ADDRESS% = 768
110 GAIN% = 2
120 STATUS% = INP(BASE_ADDRESS% + 1)
130 STATUS% = STATUS% AND &H0F
140 IF GAIN% = 1 GOTO 160
150 STATUS% = STATUS% OR (GAIN% * 8)
160 OUT BASE_ADDRESS% +1,STATUS%
```

Taking an A/D Reading

After you have selected an analog input channel and set the gain, you can take an A/D reading. It is important to note that once the gain and channel are set, they stay at those settings until you change them; that is, they are latched. You do not have to set the gain or channel every time you take a reading.

Each time an A/D conversion is completed, an end-of-convert (EOC) signal is generated to signify the end of the conversion. This signal can be used in a number of ways. One way is to use this line to monitor the A/D conversion status. Setting up the EOC signal to be monitored involves configuring bit 7 of PPI port A or port C as an input line and connecting the EOC signal to it. This procedure is detailed in Chapter 3, "Jumper Settings." The EOC signal is factory-set to be monitored through PA7 on header connector P6.

The general algorithm for taking an A/D reading is:

1. Start a 12-bit conversion by writing to base_address + 4 (or 6):

   `out base_address+4,0`

   (Note that the value you send is not important. The act of writing to this I/O location is the key to starting a conversion.)
2. Delay at least 20 microseconds or monitor PPI port A or C, bit 7 for a transition. Polling permits the fastest data acquisition.
3. Read the least significant bit from base_address + 5 (or 7):

   `lsb% = inp(base_address% +5)`
4. Read the most significant bit from base_address + 4 (or 6):

   `msb% = inp(base_address% +4)`
5. Combine them into the 12-bit result by shifting the LSB four bits to the right. The MSB must also be weighted correctly:

   `result% = (msb% * 16) + (lsb%/16)`

For a 12-bit conversion, the A/D data read is left justified in a 16-bit word, with the least significant four bits equal to zero, as shown in Figure 1-3. Because of this, the two bytes of A/D data read must be scaled to obtain a valid A/D reading. Once it is calculated, the reading can be correlated to a voltage value by scaling it, in the case of bipolar input ranges (±5 or ±10 volts), and then multiplying by the appropriate bit weight, as shown in the table at the top of the following page:

MSB															LSB
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
DB12	DB11	DB10	DB9	DB8	DB7	DB6	DB5	DB4	DB3	DB2	DB1	0	0	0	0

Fig. 1-3 — A/D Conversion Word Format

Input Range	Scale Factor	Bit Weight
±5 volts	Subtract 2048	2.4414 mV
±10 volts	Subtract 2048	4.8828 mV
0 to +10 volts	None	2.4414 mV

For example, if the A/D reading is 1024 and the input range used is ±5 volts, the analog input voltage is calculated as follows:

(1024 - 2048) bits * 2.4414 mV/bit = -2.49999 volts.

For a ±10 volt input range, the voltage is calculated as follows:

(1024 - 2048) bits * 4.8828 mV/bit = -4.99999 volts.

For a 0 to +10 volt input range, no scaling is required and the voltage is calculated as follows:

1024 bits * 2.4414 mV/bit = 2.49999 volts.

The input voltage range and polarity are factory-set according to customer specifications when ordering the board. If, after receiving your board, you wish to change the input voltage, see Chapter 3, "Jumper Settings." Whenever the voltage polarity is changed (unipolar to bipolar or vice versa), the A/D converter should be recalibrated as described in Chapter 5, "Calibration Procedures."

Note that eight-bit A/D conversions can also be performed. This is accomplished by writing to I/O location BA + 5 (or 7). While an eight-bit conversion has a lower resolution than the 12-bit conversion, it is performed much more rapidly, in about 13 microseconds. A 12-bit conversion takes about 20 microseconds. Therefore, when speed is essential, you can use the eight-bit conversion capability.

CHAPTER 2

FUNCTIONAL DESCRIPTION

This chapter describes the major functions of the AD2000 interface board. Figure 2-1 shows a block diagram of the board. The functions discussed in the following sections are:

- Analog-to-digital conversion circuitry
- Programmable peripheral interface (PPI) circuitry
- Programmable interval timer (PIT) circuitry

Analog-to-Digital Conversion Circuitry

The main function of the AD2000 interface board is to provide high-speed analog-to-digital conversion capability for data acquisition. The analog-to-digital (A/D) conversion circuitry receives inputs from eight differential or 16 single-ended analog channels, selects one active channel, and performs an analog-to-digital conversion of the voltage value read at that channel. The conversion throughput rate is typically 38 kHz.

Multiplexers

Two eight-bit analog multiplexers are used to connect either one of 16 single-ended or one of eight differential analog channels to the gain circuitry. The leftmost three switches on DIP switch S1 set up the multiplexer at IC location U9 to receive either single-ended or differential inputs. When these three switches are up, the multiplexer is configured for single-ended inputs, and when they are down, the multiplexer is configured for differential inputs. Note that these three switches are always set as a group to the same position (see "S1 Switch Settings," Chapter 3). A channel is selected through software control, by writing to port B of the PPI, as described in Chapter 1.

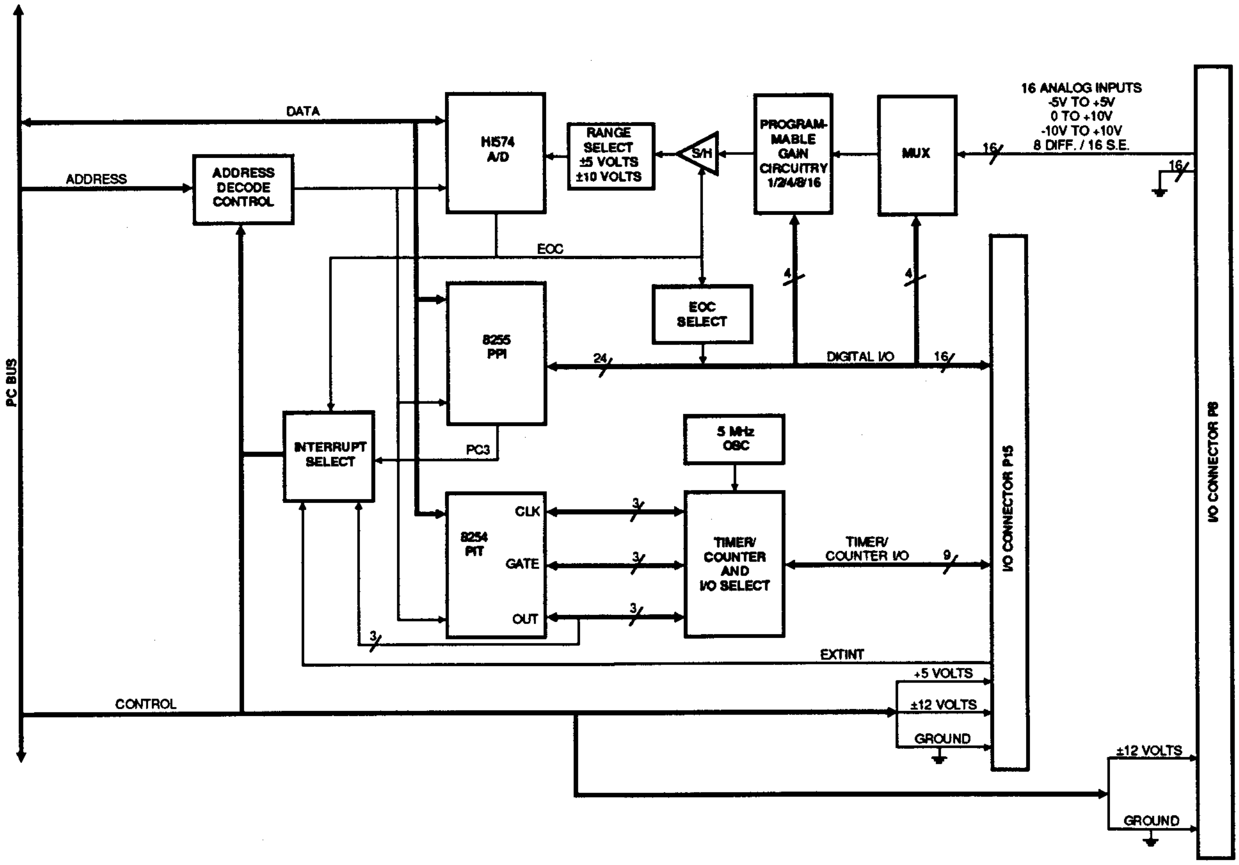


Fig. 2-1 — AD2000 Functional Block Diagram

Gain Control Circuitry

The programmable gain control circuitry can provide a gain factor of 1, 2, 4, 8, or 16. The gain selection is made by writing to port B of the PPI, as described in Chapter 1. The gain factor is controlled by the setting of four analog switches. For a gain of 2, 4, 8, or 16, this write operation will close one of the four switches; for a gain factor of 1, all switches are open. Note that programming gain factors other than the five listed here is not recommended.

Sample and Hold Circuitry

A sample and hold (S/H) amplifier is used between the gain control circuitry output and the A/D input to ensure that dynamic analog signals are accurately digitized by the A/D converter. The .001 µF hold capacitor used in this circuit is a polystyrene type selected for its low dielectric absorption. Its low value minimizes the acquisition time (6 microseconds, typical), and minimizes hold step voltage and droop. The sample and hold time and rate are determined by the EOC signal generated by the A/D converter and fed back into the S/H circuit. When the EOC signal is high (logic 1), the amplifier samples the analog input; when the EOC signal is low (logic 0), the amplifier holds the input.

A/D Converter

The A/D converter is a high-speed 12-bit conversion IC which performs conversions in approximately 20 microseconds. Eight-bit conversions can also be performed when speed is more critical than resolution. An eight-bit conversion takes about 13 microseconds, allowing rapid conversions of dynamic analog inputs. The converter supports 10- or 20-volt analog input signals; however, it cannot support a 20-volt unipolar input range because its supply voltage in the AD2000 application is only +12 volts. The analog input voltage ranges supported by the AD2000 are listed in the specifications in Appendix A. Calibration circuitry is included for unipolar and bipolar calibration of the A/D converter. Calibration procedures are described in Chapter 5.

An 8- or 12-bit conversion is initiated by a write operation to the appropriate I/O address. Once a conversion is begun, the conversion status can be monitored by reading the A/D converter status (STS) signal which is output from the A/D converter IC and inverted before being made available to other circuitry on the board as the end-of-convert (EOC) signal. The EOC signal can be monitored by one of two digital input lines on the PPI, PA7 or PC7. Note that if either line is selected as the EOC monitor, a jumper must be installed for the selected line on P6 and that line must be configured as an input. The EOC signal is factory-set to be monitored through PA7 on P6. The EOC signal is low (logic 0) during a conversion. Figure 2-2 shows the EOC timing diagram. Also, the three-state A/D output buffers remain in a high-impedance state, and, therefore, data cannot be read. While a conversion is in progress, any transitions of the digital inputs which control the conversion will be ignored, so that the conversion cannot be prematurely terminated or restarted. Once the conversion is complete (EOC is now high, or logic 1), the A/D data can be read in two bytes, the MSB and the LSB, in any order. For a 12-bit conversion, the data is left-justified in a 16-bit word. In the case of an eight-bit conversion, the data is completely contained in the eight-bit MSB.

Refer to Chapter 1, "Taking an A/D Reading," and the demo disk for more information about using the A/D converter.

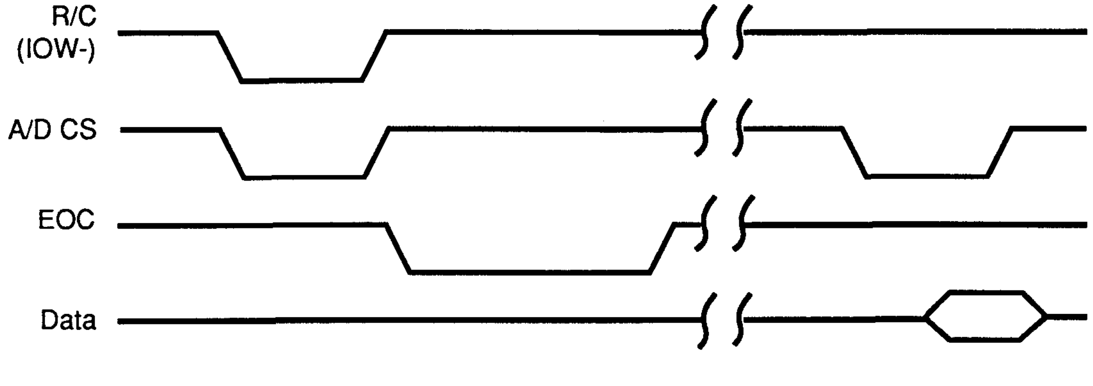


Fig. 2-2 —EOC Timing Diagram

Programmable Peripheral Interface

The Programmable Peripheral Interface (PPI) provides 16 TTL/CMOS digital I/O lines which can be configured in a number of ways to support user requirements. The lines available for digital I/O are port A and port C. The 8255 PPI has a total of 24 digital I/O lines, eight of which are used to control the A/D channel selection and gain circuitry, and therefore are not available to the user. The remaining 16 lines are available at external I/O connector P15. The 24 lines are grouped into three eight-bit ports, port A, port B, and port C. Port C is further subdivided into two four-bit ports, port C lower (PC0-PC3) and port C upper (PC4-PC7) in certain modes of operation. The PPI data sheet is included in Appendix C.

The eight bits of port B are reserved for A/D channel selection and gain control, and cannot be configured for I/O use. Ports A and C can be configured in any of the three operating modes described below:

> Mode 0 — Basic input/output. Provides simple input and output operations for each port. Data is written to or read from a specified port.
>
> Mode 1 — Strobed input/output. Provides a means for transferring I/O data to or from port A or port B in conjunction with strobes or handshaking signals.
>
> Mode 2 — Strobed bidirectional input/output. Provides a bidirectional means of communicating with another device on a single eight-bit bus. Handshaking signals are similar to mode 1. This mode applies to port A only.

In mode 0, all four ports (A, B, C lower, and C upper) are available as I/O lines. Sixteen configurations are possible in this mode, and any port can be configured as an input or an output. The outputs are latched, but the inputs are not latched.

In mode 1, the four ports are grouped into two groups. Each group contains one eight-bit data port (port A or port B) and one four-bit control/data port (port C lower or port C upper) which is used for control and status of the eight-bit port. The eight-bit data port in each group can be configured as an input or an output. Both inputs and outputs are latched.

In mode 2, port A is an eight-bit bidirectional bus and port C is a five-bit control port. Port B cannot be used in this mode, but is available for use in mode 0 or mode 1 while port A is in mode 2. Both inputs and outputs are latched.

The PPI is configured by writing a control word to the appropriate I/O address location, as described in Chapter 4, "Programming Your AD2000."

The control word can also be used to individually set or reset the port C bits. This feature allows any bit of port C to be set or reset without affecting the other port C bits. The data sheet included in Appendix C explains this feature.

The PPI can also be used to generate interrupts in mode 1 or mode 2 operation. In these modes, the interrupt enable (INTE) mask is used to enable the INTRA and INTRB interrupt signals. Note that the INTRB signal for PPI cannot be used since port B of this PPI is always configured as mode 0 output and is reserved for channel selection and gain control. Interrupt functions are further explained in the data sheet in Appendix C.

The AD2000 board provides a header connector which can jumper the A/D converter end-of-convert (EOC) signal to a PPI bit where it can be monitored to provide A/D conversion status. The EOC signal can be jumpered to either PA7 (port A, bit 7) or PC7 (port C, bit 7). The default setting of the jumper is PA7. The port used to monitor the EOC signal must be configured as a mode 0 input port.

Programmable Interval Timer (PIT)

The programmable interval timer (PIT) can be configured for a variety of timing and counting functions. This versatile IC contains three independently clocked 16-bit timer/counter circuits, TC0, TC1, and TC2, which operate as down counters. These down counters can resolve time increments down to 125 nanoseconds. This circuit's most common application is to provide accurate time delays under software control. Upon command, the PIT can count out a programmed delay and interrupt the PC when it has finished its tasks. All three counter outputs are brought out to external I/O connector P15.

The three 16-bit timer/counters are each loaded by two one-byte write operations to the appropriate I/O location. The bytes are latched into a 16-bit internal count register, where they are stored until the count sequence starts. The countdown starts when the count register contents are transferred (in parallel) to the down counter. The timer/counter circuits can be programmed for binary or BCD countdowns.

A 5 MHz crystal oscillator on the AD2000 can be used to clock any timer/counter circuit. Or, the timer/counter can be clocked by a source external to the board through external I/O connector P15. Rates of dc to 8 MHz can be used to clock the timer/counters.

Each timer/counter can be configured for one of six modes of operation. These modes are:

Mode 0 — Interrupt on end of count. The OUT signal changes from low to high when the countdown is completed.

Mode 1 — Re-triggerable one-shot. A low-level pulse triggered by the GT input is output on the OUT pin.

Mode 2 — Rate generator.

Mode 3 — Square wave generator.

Mode 4 — Software-triggered strobe.

Mode 5 — Hardware-triggered strobe (re-triggerable).

The timer/counter count modes, as well as the count type (binary or BCD), read/write mode, and counter/timer selection mode, are all part of the control word which is written to the PIT control register to initialize the circuit. When the PC is powered up, the timer/counter circuits are not defined until the appropriate control words are written to the circuits to program them for operation. Initialization is required only once after a power-up reset occurs. Detailed information about the PIT, including the control word format, is given in the data sheet in Appendix C. Appendix D contains programming notes for some PIT applications.

The three timer/counter circuits are independent. However, they can be cascaded for countdowns which are longer than one 16-bit field can support. For example, TC0's OUT signal can be connected to TC1's CK signal, and TC1's OUT signal can be connected to TC2's CK signal. When configured this way, the PIT can accommodate extremely long countdowns. This configuration is described in the application notes in Appendix D.

One of the three timer/counter outputs, TC0 OUT, TC1 OUT, or TC2 OUT, can also be used as a PC interrupt. These signals are brought out to board header connector P5 where one (and only one) can be selected for connection to any one IRQ channel, IRQ2 through IRQ7. Chapter 3, "Jumper Settings," and Chapter 4, "Programming Your AD2000," describe these interrupts in more detail.

CHAPTER 3

JUMPER SETTINGS

This chapter describes the AD2000 board settings you can control on DIP switch S1 and various header connectors. You can use this chapter to tailor your board's functions to your specific application before installing it in your computer, or to change the board's configuration as you learn more about its operation and special features. In this chapter, you will learn about each setting and how to set switches or install jumpers to achieve the desired operation of your board. Before changing any settings, you should have a functional knowledge of the circuit you are setting up (see Chapter 2). Remember that all of the settings described in this chapter have been factory-set, or, as in the case of the interrupt signals, are disabled. Therefore, you do not have to do any further set-up of the board in order for it to operate in your system as described in Chapter 1. The descriptions in this chapter allow you to change factory settings, or to tailor your board to take full advantage of its built-in versatility.

There are one DIP switch and several header connectors which allow you to control various board functions. These are shown in the board layout of Figure 3-1 and are presented as follows:

S1 — Analog Input Signal Type DIP Switch

P2 — Base I/O Address Header Connector

P3 — Programmable Interval Timer (PIT) I/O Header Connector

P4, P5, and P7 — Interrupt Header Connectors

P6 — End-of-Convert (EOC) Monitor Header Connector

P9 — A/D Converter Voltage Range Header Connector

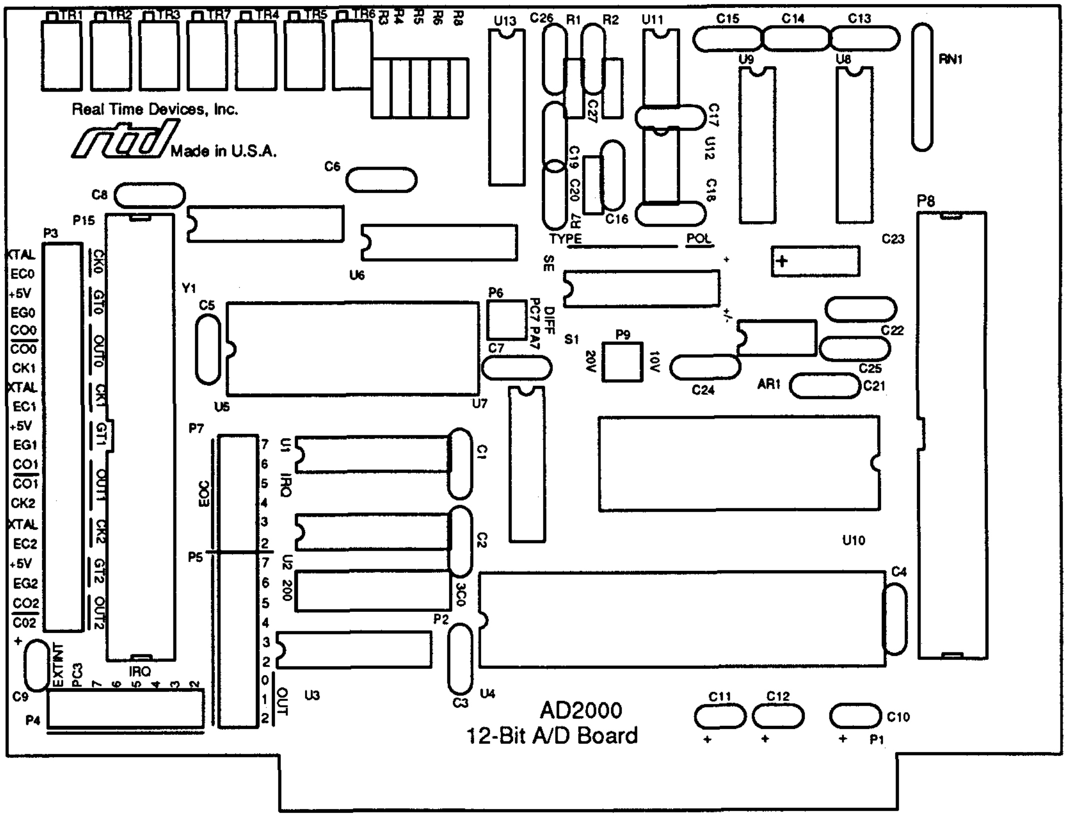


Fig. 3-1 — AD2000 Board Layout

S1 — Analog Input Signal Type DIP Switch

DIP switch S1, shown in Figure 3-2, configures the multiplexers for single-ended or differential inputs and selects a unipolar or bipolar input voltage range. The first three switches on S1 operate as a group. When these are in the UP position, the multiplexers are configured for single-ended inputs; when they are in the DOWN position, the multiplexers are configured for differential inputs. Note that these three switches must all be set to the same position (UP or DOWN) for the multiplexers to function properly. The remaining switch, S1-4, controls the input voltage polarity. When this switch is in the UP position, the input voltage range is unipolar; when it is in the DOWN position, the voltage range is bipolar. This switch, coupled with the voltage range selection set on header connector P9, determines the analog input voltages supported by the A/D converter. Note that whenever the polarity is changed, the A/D converter circuitry should be calibrated as described in Chapter 5. The switch settings are clearly labeled on the board to eliminate errors when configuring S1.

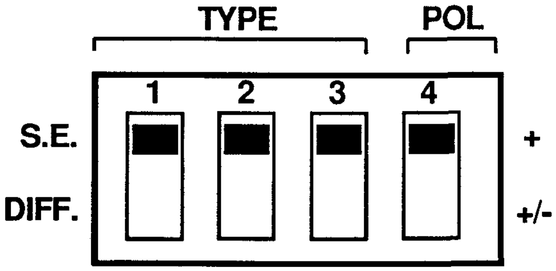


Fig. 3-2 — DIP Switch S1

P2 — Base I/O Address Header Connector

Header connector P2 controls the 12 computer I/O address locations used by the board. The base I/O address location is set by jumpering one of the eight positions on the P2 header connector. The base I/O address is factory-set to 300 hex (768 decimal), with the jumper installed across the pair of pins fifth from the left on the connector. The base I/O address setting is fully explained in Chapter 1, "Base I/O Address Setting," and is not repeated here. Note the importance of this setting with respect to the possibility of address contention with other devices in your computer. Be sure to examine this possibility if you experience board failure when you first attempt to operate the board in your computer.

P3 — Programmable Interval Timer (PIT) I/O Header Connector

Header connector P3, shown in Figure 3-3, controls the programmable interval timer (PIT). The PIT contains three independent 16-bit timer/counter circuits, as described in Chapter 2. Each timer/counter has three I/O signals associated with it: a clock, a gate, and an output. P3 can be configured in a number of ways to provide maximum versatility in applying this device to your particular application. Each timer/counter is factory-set for XTAL clock input, +5V gate input, and CO output. Figure 3-4 shows a block diagram of the PIT.

For ease in configuring this circuitry, the header connector is partitioned into three functional groups: TC0, TC1, and TC2, which correspond to timer/counter 0, timer/counter 1, and timer/counter 2, respectively. These designations also correspond to the manufacturer's designations, as shown on the data sheet included in Appendix C. Starting from the top of P3, the first group of pins on the right side are labeled CK0, GT0, and OUT0, the three I/O signals for TC0. The signals on the left side for TC0 are labeled XTAL, EC0, +5V, EG0, CO0, and $\overline{\text{CO0}}$ (this signal has a bar over top of the signal name on the board as the inverse designation). The groups of signals for TC1 and TC2 are identical to TC0, except that each has a CK input on the left side of the header connector. Note that each signal name on the right side of the connector (CK, GT, and OUT) spans a group of two or three pins. Each group can have only one jumper installed at any time. The following paragraphs describe how these signals can be used in the PIT circuit. An "x" is used in place of 0, 1, or 2 in the signal names whenever the application can be applied to any or all of the three timer/counter circuits.

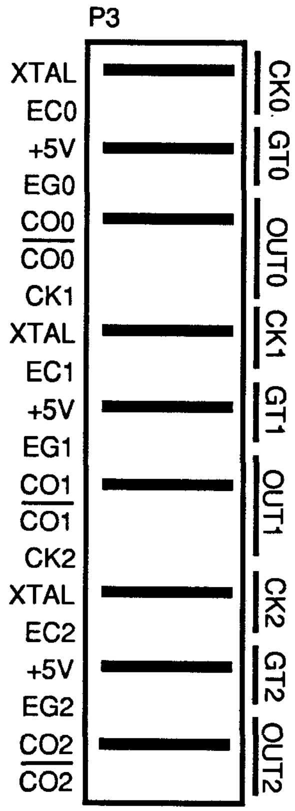


Fig. 3-3 — PIT I/O Header Connector P3

Counter Inputs:

XTAL — This input to all three timer/counter circuits is from the 5 MHz crystal oscillator, labeled Y1, located in the upper left area of the board. By connecting XTAL to the CKx input on the right side of the connector with a jumper placed horizontally between the pins, the 5 MHz clock is applied to the timer/counter circuit. If required by your application, the XTAL frequency can be changed by installing a different crystal oscillator at Y1. Note, however, that the maximum frequency at which the PIT will operate is 8 MHz.

ECx — This input allows an external clock, other than the XTAL signal, to control the timing of the corresponding timer/counter circuit. This pin can be horizontally jumpered to the CKx input on the right side of the connector, in place of the XTAL source. The ECx signals are brought onto the board through external I/O connector P15 (see Table B-3 in Appendix B).

Gate Inputs:

+5V — This input, if connected to the GTx input by placing a jumper horizontally between the two pins, places the associated timer/counter circuit in an enabled state at all times.

EGx — This input can be horizontally jumpered to the GTx input on the right side of the connector to provide an external gate input instead of the +5 volts input. The EGx signals are brought onto the board through external I/O connector P15 (see Table B-3 in Appendix B).

Counter Outputs:

COx — This output can be horizontally jumpered to the corresponding OUT pin on the right side of the connector so that the clock output signal can be routed to external I/O connector P15 (see Table B-3 in Appendix B).

$\overline{\text{COx}}$ — This output can be horizontally jumpered to the corresponding OUT pin on the right side of the connector to provide the inverse of the clock output signal to external I/O connector P15 (see Table B-3 in Appendix B).

CKx — This input connects the output of one timer/counter to the clock input of the next timer/counter. CKx is provided for TC1 and TC2 only, and is connected to the output of the previous timer/counter (TC0 or TC1) by placing a jumper horizontally between the pins. These connections are used to cascade the timer/counters for longer time delays than are supported by a single timer/counter circuit.

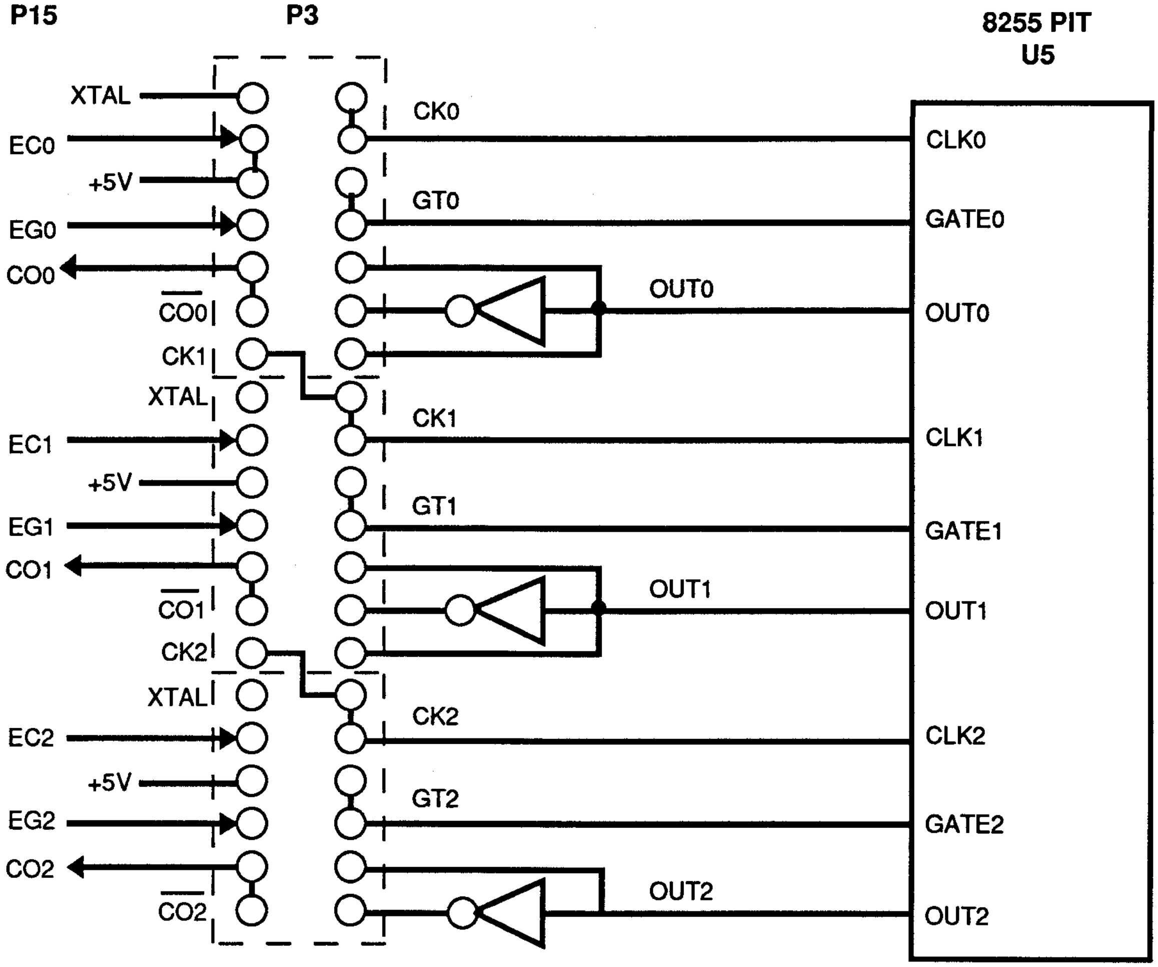


Fig. 3-4 — PIT Functional Block Diagram

P4, P5, and P7 — Interrupt Header Connectors

Header connectors P4, P5, and P7 are used to jumper various signals generated by the AD2000 circuitry to the PC's interrupt channels. The interrupt channels available on the board are IRQ2 through IRQ7. Note that only one interrupt in the computer system can be connected to an interrupt channel at any given time.

Before attempting to use interrupts, you should be familiar with the procedure for initializing the interrupt vectors and the PC's *interrupt* controller, and setting up the interrupt handling routines. These procedures are beyond the scope of this manual, but must be understood to effectively use interrupts in you computer system.

Be careful to avoid contention when selecting the interrupt channels used, both with the signals on the AD2000 as well as with other devices within your computer. To avoid contention, use the table inside the back cover of this manual to record the interrupt channels you use with the AD2000 board.

It is also very important to note that the AD2000 interrupt sources are TTL totem-pole (push/pull) type outputs; they are **not** open-collector. Therefore, do not attempt to connect one of these interrupts to any other interrupt output.

The following paragraphs describe the interrupts available on your AD2000 board.

P4 — EXTINT and PPI INTRA Interrupts

Header connector P4 is used to select EXTINT or PPI INTRA for connection to one of the computer's interrupt channels IRQ2 through IRQ7. EXTINT is provided to accommodate an interrupt signal generated external to the AD2000 and routed onto the board through external I/O connector P15 (see Table B-3 in Appendix B). PPI INTRA (labeled PC3 on the board) is generated by the PPI. This interrupt is generated during PPI mode 1 or mode 2 operation only. One of these two signals can be jumpered to one of the available computer interrupt channels IRQ2 through IRQ7 by first placing a jumper vertically across the pins of the signal chosen and then placing a second jumper vertically across the pins of the selected IRQ channel. Figure 3-5 shows header connector P4 with jumpers installed so that PPI INTRA is connected to IRQ2.

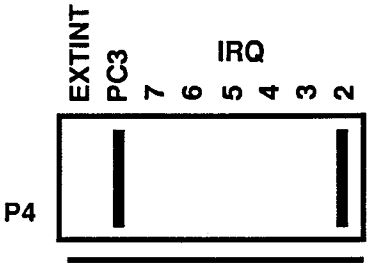


Fig. 3-5 — Interrupt Header Connector P4

P5 — PIT Output Interrupts

Header connector P5, shown in Figure 3-6, is used to jumper one of the three PIT outputs, OUT0, OUT1, or OUT2, to one of the computer's interrupt channels IRQ2 through IRQ7. As in the case of P4, two jumpers must be installed to connect a PIT output to an interrupt channel. First, install a jumper horizontally across the pins of the PIT output selected. Then install a second jumper across the pins of the interrupt channel selected. Figure 3-6 shows jumpers installed so that OUT2 is connected to IRQ3.

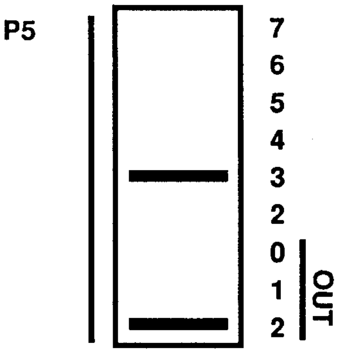


Fig. 3-6 — Interrupt Header Connector P5

P7 — A/D End-of-Convert (EOC) Interrupt

Header connector P7, shown in Figure 3-7, is used to jumper the A/D converter's end-of-convert (EOC) signal to one of the computer's interrupt channels IRQ2 through IRQ7. The EOC signal is connected to an IRQ channel by installing a single jumper horizontally across the pins of the IRQ channel selected. Figure 3-7 shows the EOC signal connected to IRQ4.

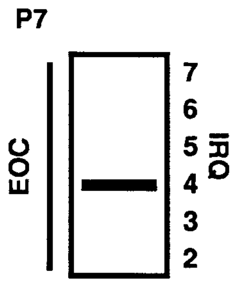


Fig. 3-7 — Interrupt Header Connector P7

P6 — End-of-Convert (EOC) Monitor Header Connector

As described above, the A/D converter end-of-convert (EOC) signal can be used to generate an interrupt. If this signal is not used as an interrupt, it can be used as a status monitor of the A/D conversion process. Header connector P6 provides two lines through which the EOC can be monitored from the PPI, PA7 or PC7. One of these two digital I/O lines is selected for EOC monitoring by installing a jumper horizontally across the appropriate pair of pins. The digital I/O line selected, PA7 or PC7, must be configured as a mode 0 input (see Chapter 4, "Programming Your AD2000"). Figure 3-8 shows P6 with a jumper installed in the factory-set position for EOC monitoring through PA7.

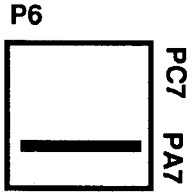


Fig. 3-8 — EOC Monitor Header Connector P6

P9 — A/D Converter Voltage Range Header Connector

Header connector P9, shown in Figure 3-9, is used to select the analog input voltage range of the A/D converter. A jumper is installed vertically across the pins marked 10V to support a 10-volt range (0 to 10 volts or -5 to +5 volts), or across the pins marked 20V to support a 20-volt range (-10 to +10 volts). The setting of this jumper, coupled with the setting of DIP switch S1-4 which selects a unipolar or a bipolar range, determines the input voltage range of the A/D converter. P9 is configured at the factory according to the customer's specifications for the input voltage range. The valid settings of P9 and S1-4 are summarized in the table below:

Range	P9 Setting	S1-4 Setting
-5 to +5 volts	10V (right)	DOWN (bipolar)
0 to +10 volts	10V (right)	UP (unipolar)
-10 to +10 volts	20V (left)	DOWN (bipolar)

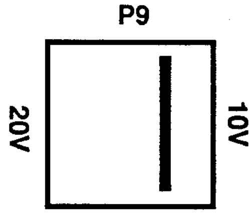


Fig. 3-9 — A/D Converter Voltage Range Header Connector P9

CHAPTER 4

PROGRAMMING YOUR AD2000

All communication with the AD2000 interface board is done by strobing data to and from the board using the I/O reference instructions. Most operations involve the transfer of data to or from the components' internal registers. However, some operations require only that a particular I/O address be written to; the data written is irrelevant. These I/O locations are referenced to the AD2000 base I/O address (BA) determined by the jumper setting of connector P2. Chapter 1 describes the base I/O address considerations and configuration.

The data collection and support functions controlled through software include the analog input channel selection and gain, control of the the A/D conversion, the programmable peripheral interface, and the programmable interval timer. Because they are integral to the basic operation of the board, the analog input channel and gain selections and taking an A/D reading are covered in Chapter 1. Digital I/O control through the PPI and control of the programmable interval timer are more complex, and are described in this chapter.

The demonstration disk which accompanies your AD2000 contain examples in Turbo C, Turbo Pascal, and BASIC.

Nearly all modern MS-DOS-based PC languages have I/O reference instructions. These are the instructions to control the data transfers to and from the I/O ports. Consult your programming language reference to find these instructions for your favorite language.

Selecting an Analog Input Channel

See this section in Chapter 1.

Setting the Input Gain

See this section in Chapter 1.

Taking an A/D Reading

See this section in Chapter 1.

Programming the Programmable Peripheral Interface

The programmable peripheral interface (PPI) has three eight-bit parallel I/O ports, port A, port B, and port C, which can be configured for a variety of applications. The PPI has 16 lines available at external I/O connector P15; the eight bits of port B (PB0-PB7) are used for channel selection and gain control and cannot be used for other functions.

The PPI ports can be operated in one of three modes. The mode of operation and the signal direction of each port (input or output) are controlled by an eight-bit control word written to an internal register. Two bits define the mode selection: mode 0, mode 1, or mode 2. Four bits configure the I/O direction: one bit to control PA0-PA7, one bit to control PB0-PB7, one bit to control PC0-PC3, and one bit to control PC4-PC7. Port C is divided into two four-bit fields so that it can provide status and control for port A if desired in your application. The control word is defined in Figure 4-1.

The PPI is configured by writing a control word to it's internal control register. Upon power-up, all ports are configured as mode 0 inputs. The PPI is written to during board initialization so that port B is set up as a mode 0 output to configure it for channel selection and gain control functions. Chapter 1, "Initializing Your AD2000," describes this procedure.

Because the PPI can be configured for a wide range of operating modes and programming requirements, it is heavily dependent on correctly understanding how to use the proper control byte to configure the PPI for your application. The demo disk includes example programs that show how to select the common operating modes. Reading the source code is highly recommended.

For more information about the operation of the PPI, see the data sheet included in Appendix C.

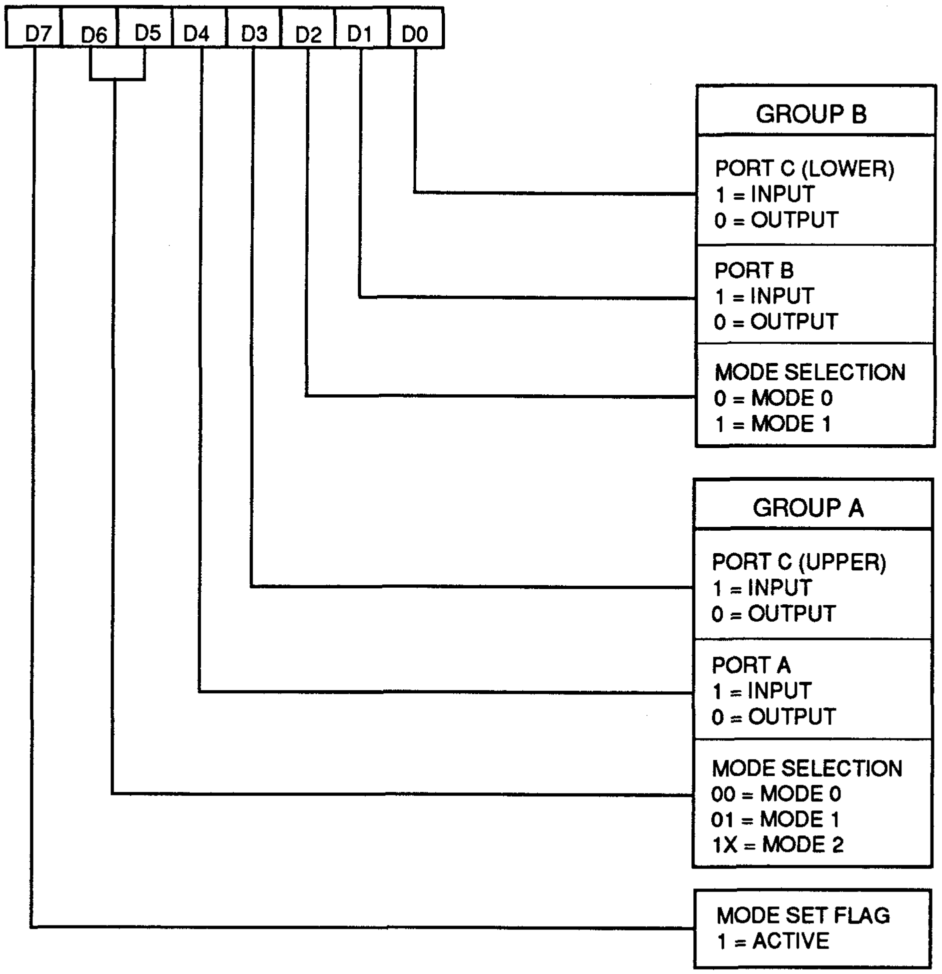


Fig. 4-1 — PPI Mode Definition Format

Programming the Programmable Interval Timer

The programmable interval timer (PIT) can be configured for a variety of timing and counting functions. The PIT's versatility is supplemented by the use of header connector P3 for jumpering various I/O options. Chapter 3, "Jumper Settings," describes this connector.

The PIT consists of three independent 16-bit down counters. The counters are initialized for operation in any of six modes by writing data to the appropriate control word for each counter. Counter data is then written to or read from each of the counters by accessing three additional internal registers. The data is set up in a two-byte format, each byte serially accessible on the data bus. The I/O locations that control the PIT are listed below from Table 1-2.

PIT FUNCTION	A4	A3	A2	A1	A0	R/W	BA + HEX
Counter 0	1	0	1	0	0	R/W	14
Counter 1	1	0	1	0	1	R/W	15
Counter 2	1	0	1	1	0	R/W	16
Control Word	1	0	1	1	1	W	17

Your specific requirements will determine how the individual timer/counters should be configured. The data sheet included in Appendix C provides the information required to control the PIT.

The software included on the demo disk shows example programs for controlling some of the PIT operating modes. In addition, some typical applications are presented in the programmable interval timer application notes in Appendix D. Included are examples requiring two or more counters to be cascaded.

The signals generated by the OUT pins for any of the counters may be connected to one of the PC's interrupt channels using jumpers installed at connector P5. Refer to the "Hardware Interrupts" section below for more information on using the OUT signals to generate interrupts.

Hardware Interrupts

Three jumper connectors, P4, P5, and P7, are provided on the AD2000 to enable interrupts generated by the A/D converter, the PIT, the PPI, and an external source to the PC's interrupt channels IRQ2 through IRQ7. Chapter 3, "Jumper Settings," explains how these header connectors can be configured.

Before you attempt to use interrupts, be sure you are familiar with the procedure for initializing the interrupt vectors and the PC's interrupt controller, and setting up the interrupt handling routines. Reference 1 in Appendix E provides a good description of the PC's system interrupts.

A/D Converter End-of-Convert (EOC) Signal

The A/D converter EOC signal can be used to generate an interrupt to the PC. An interrupt will occur (through the selected interrupt channel) to indicate a conversion is complete approximately 20 microseconds after the conversion is initiated. The EOC signal is inverted before being applied to the interrupt channel. It makes a low-to-high transition at the completion of each conversion cycle, and remains high until another conversion is initiated. The timing of the EOC signal is shown in Figure 2-2, Chapter 2.

PPI Interrupts

The PPI INTRA (PC3) interrupt generated in PPI mode 1 and mode 2 operation can be jumpered to any of the PC interrupt channels IRQ2 through IRQ7. The timing of this interrupt is shown on the PPI data sheet included in Appendix C.

The PPI interrupt must be enabled by writing a "1" to the INTE mask bit of the PPI as described in the data sheet under "Interrupt Control Functions." The INTE mask bit is disabled during power-up reset and whenever the PPI mode is changed.

PIT Interrupts

One of the OUT0, OUT1, or OUT2 signals generated by the PIT can be jumpered to a PC interrupt channel using connector P5.

When using a PIT OUT signal as an interrupt, you must be very careful to ensure that the PC system's programmable interrupt controller (PIC) is properly configured to ignore interrupts on the selected interrupt channel immediately after power-up. This is necessary because the PIT must first be initialized to define the desired mode(s) of operation. Prior to initialization, the mode, count, and output of all counters are undefined. If the system interrupts are not disabled, the counter outputs may cause erratic system behavior.

CHAPTER 5

CALIBRATION PROCEDURES

This chapter contains calibration procedures for the A/D converter input voltage range and the A/D converter gain. The offset and full-scale performance of the AD2000 A/D converter is factory-calibrated according to the specifications that were given when your order was placed. The gain circuitry is also factory-calibrated before the board is shipped. The following procedure allows you to quickly verify the accuracy of these circuits. This procedure should be done approximately every six months, whenever inaccurate readings are suspected, or whenever the voltage range is changed.

Calibration is performed with a properly configured AD2000 installed in the PC. Apply power to the computer and allow the AD2000 circuitry to stabilize for 15 minutes.

Required Equipment

The following equipment is required for calibration:

- Precision Voltage Source: 0 to ±10 volts
- Digital Voltmeter: 5-1/2 digit
- Small Screwdriver (for trimpot adjustment)

Figure 5-1 shows the board layout. The trimpots referenced in the following procedures are grouped in the upper left area of the board.

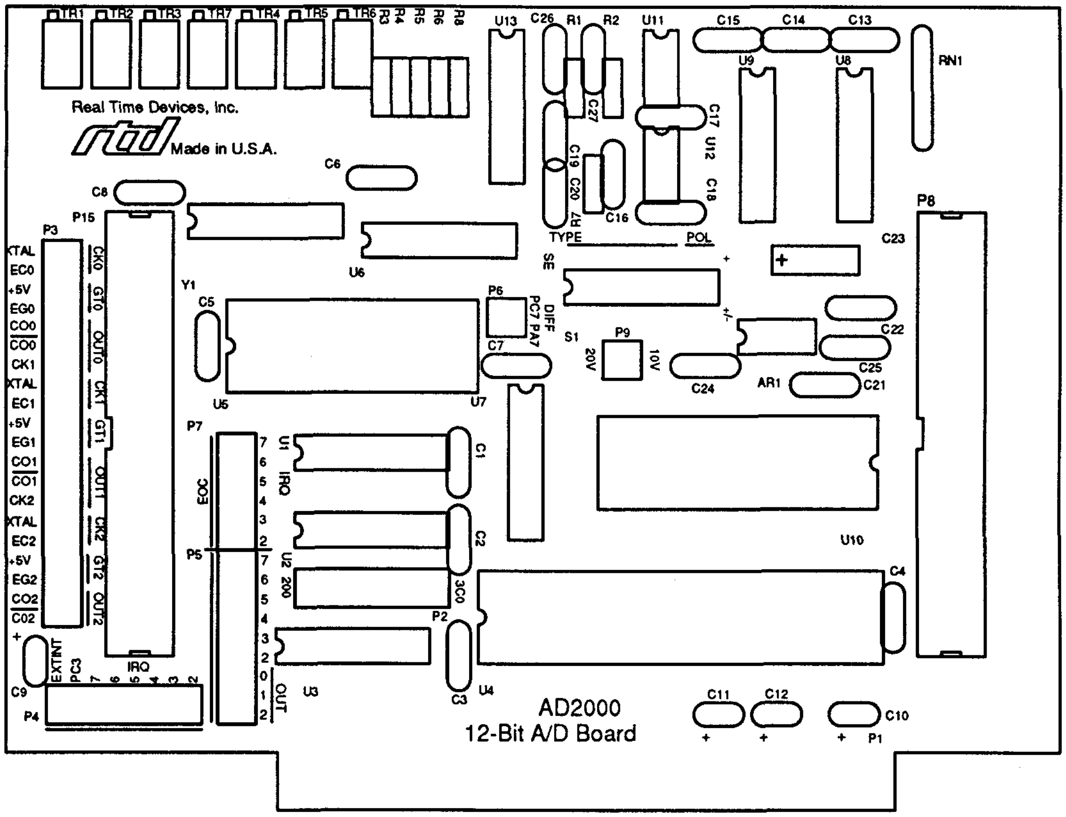


Fig. 5-1 — AD2000 Board Layout

A/D Calibration

During this procedure, connections must be made to some of the analog inputs on external I/O connector P8, available at the rear panel of the computer. The pin assignments for this connector are given in Table B-2, Appendix B.

Two adjustments are necessary to completely calibrate the A/D converter for unipolar or bipolar operation. These affect the offset and full-scale performance of the AD2000 circuitry. Both calibration steps are performed using trimpots TR5 and TR6 or TR6 and TR7. Trimpot TR5 or TR7 is used to zero the offset error of the A/D converter and trimpot TR6 is used for full-scale adjustment. In the following procedure, use analog input channel 1 and set it for a gain of 1. This is accomplished by writing all zeroes to I/O address location BA + 1. Be certain that position 4 of switch S1 is set for the desired polarity and the jumper on connector P9 is set for 10V.

Unipolar Calibration

Two adjustments are necessary to calibrate the A/D converter for the unipolar voltage range of 0 to +10 volts, one for offset and one for full scale. To adjust the offset, a very low analog input voltage, shown under the "Offset" heading in the following table, is connected to the channel 1 input of the multiplexer (P8-1). The ground reference of this signal should be connected to P8-2. While continuously displaying 12-bit A/D conversions, adjust TR7 until the A/D data flickers between the two values listed in the table under "Offset."

After the offset adjustment is made, TR6 is used to adjust the full-scale value. While the full-scale input voltage listed in the table is not the actual full-scale voltage for an ideal 0 to +10 volt range, it is the maximum voltage at which the A/D conversion is guaranteed to be linear. Any value above this voltage may not be linear and thus may adversely affect calibration. After connecting the full-scale voltage listed in the table to the channel 1 input, adjust TR6 until the data flickers between the two values in the table under "Full Scale."

Unipolar Calibration (0 to +10 volts range)		
	Offset (TR7)	**Full Scale (TR6)**
Input Voltage	+1.22070 millivolts	+9.49829 volts
A/D Data	0000 0000 0000 0000 0000 0001	1111 0011 0010 1111 0011 0011

Bipolar Calibration

Whether you are selecting the bipolar input voltage range of -5 to +5 volts or -10 to +10 volts, the following calibration procedure can only be performed with the board configured for a -5 to +5 volt input voltage range. This means that the jumper on header connector P9 must be installed across the 10V pins. If you are using the -10 to +10 volt range, reposition the jumper on P9 across the 20V pins **after** you perform the calibration procedures below.

Two adjustments are necessary to calibrate the A/D converter for bipolar voltage ranges, one for offset and one for full scale. To adjust the offset, connect the voltage shown under the "Offset" heading in the table below to the channel 1 input of the multiplexer. While continuously displaying 12-bit A/D conversions, adjust TR5 until the data flickers between the two values listed in the table under "Offset." Next, connect the full-scale voltage listed in the table to the channel 1 input and adjust TR6 until the data flickers between the two values in the table under "Full Scale."

Bipolar Calibration (-5 to +5 volts or -10 to +10 volts range)		
	Offset (TR5)	**Full Scale (TR6)**
Input Voltage	-4.99878 volts	+4.99634 volts
A/D Data	0000 0000 0000 0000 0000 0001	1111 1111 1110 1111 1111 1111

Table 5-1 provides a reference for the ideal input voltage for the A/D converter for each bit weight in each voltage range. This table shows the ideal full-scale (all ones) value in the first line and decrements by one bit weight each line thereafter. Note that these values are for 12-bit A/D conversions, and are not valid when using the converter to perform more rapid eight-bit conversions. Note that the voltage values in the table are in millivolts.

Table 5-1 — A/D Converter Bit Weights

	Ideal Input Voltage (millivolts)		
A/D Bit Weight	±5 Volts	±10 Volts	0 to +10 Volts
4095 (Full-Scale)	+4997.6	+9995.1	+9997.6
2048	0000.0	0000.0	+5000.0
1024	-2500.0	-5000.0	+2500.0
512	-3750.0	-7500.0	+1250.0
256	-4375.0	-8750.0	+625.00
128	-4687.5	-9375.0	+312.50
64	-4843.8	-9687.5	+156.250
32	-4921.9	-9843.8	+78.125
16	-4960.9	-9921.9	+39.063
8	-4980.5	-9960.9	+19.5313
4	-4990.2	-9980.5	+9.7656
2	-4995.1	-9990.2	+4.8828
1	-4997.6	-9995.1	+2.4414
0	-5000.0	-10000.0	0.0000

Gain Circuitry Calibration

Four trimpots, TR1 through TR4, are used to adjust the gain circuitry, one for each of the gains 2, 4, 8, and 16. To calibrate this circuitry, apply an input voltage of +39.063 millivolts to the input of channel 1. Next, by writing the correct word to the BA +1 I/O location, set the gain to 2 and adjust trimpot TR1 to obtain the 12-bit A/D converter output for your board's voltage range, as listed in Table 5-2. Then, repeat this procedure for each of the remaining three gain settings, adjusting the appropriate trimpot until achieving the correct value listed in the table.

Table 5-2 — A/D Converter Readings for Gain Calibration

		Input Voltage Range		
Gain	Trimpot	±5 Volts	±10 Volts	0 to +10 Volts
2	TR1	1000 0010 0000	1000 0001 0000	0000 0010 0000
4	TR2	1000 0100 0000	1000 0010 0000	0000 0100 0000
8	TR3	1000 1000 0000	1000 0100 0000	0000 1000 0000
16	TR4	1001 0000 0000	1000 1000 0000	0001 0000 0000

APPENDIX A

AD2000 SPECIFICATIONS

AD2000 SPECIFICATIONS

(Typical at 25°C)

Interface:	IBM PC/XT/AT compatible	
	Jumper-selectable base address, I/O mapped	
	Jumper-selectable interrupts	
Analog Inputs:	8 differential or 16 single-ended inputs, switch-selectable	
	Input impedance, each channel	>10 megohms
	Gains, software selectable	1, 2, 4, 8, or 16
	Gain error	0.5% typ, 1% max
	Input options: 10-volt range* (Option 1)	Bipolar ±5V
	Guaranteed Linearity	±5V
	10-volt range* (Option 2)	Unipolar 0 to +10V
	Guaranteed Linearity	0 to +9.5V
	20-volt range* (Option 3)	Bipolar ±10 V
	Guaranteed Linearity	±9.5V
	Range	Jumper-selectable
	Polarity	Switch-selectable
	Settling time	3 µsec max
	Common mode input voltage	±10V
	Overvoltage protection	±35 Vdc
	*Erratic readings can occur beyond specified input voltage ranges.	
A/D Converter:	Type	Successive approximation
	Resolution: 10-volt range	12 bits (2.44 mV/bit)
	20-volt range	12 bits (4.88 mV/bit)
	Chip-selectable conversion speed: Option 0	20 µsec typ, 25 µsec max
	Option 1	12 µsec typ, 15 µsec max
	Option 2	8 µsec typ, 9 µsec max
	Linearity	±1 bit typ
	Sample-and-hold acquisition time	6 µsec max
	Throughput	38 kHz
Counter/Timer:	Three 16-bit, 8 MHz down counters	
Digital I/O Lines:	16 TTL/CMOS-compatible	
Miscellaneous I/Os:	±12V, +5V, PC bus-sourced	
	Ground, PC bus-sourced	
	One external interrupt input	
Power Requirements:	+5 Volts	260 mA
	+12 Volts	30 mA
	-12 Volts	35 mA
I/O Connectors:	Two 40-pin box headers (one dedicated to analog signals only)	
	All 80 signals exit through one rear panel slot in the PC	

Environmental:	Operating temperature	0 to +70°C
	Storage temperature	-40 to +85°C
	Humidity	0 to 90% non-condensing
Size:	Height	3.875" (99 mm)
	Width	5.50" (140 mm)

APPENDIX B

CONNECTOR PIN ASSIGNMENTS

CONNECTOR PIN ASSIGNMENTS

Table B-1—Mating External I/O Connectors for P8 and P15

Connector No.	Manufacturer	Part Number
P8	3M	3417-7040
P15	Mil C-83503	M83503/7-09

Table B-2—P8 Connector Pin Assignments

Pin No.	Signal Name	Pin No.	Signal Name
	DIFF / SE		
1	AIN1+ / AIN1	2	GND
3	AIN1- / AIN9	4	GND
5	AIN2+ / AIN2	6	GND
7	AIN2- / AIN10	8	GND
9	AIN3+ / AIN3	10	GND
11	AIN3- / AIN11	12	GND
13	AIN4+ / AIN4	14	GND
15	AIN4- / AIN12	16	GND
17	AIN5+ / AIN5	18	GND
19	AIN5- / AIN13	20	GND
21	AIN6+ / AIN6	22	GND
23	AIN6- / AIN14	24	GND
25	AIN7+ / AIN7	26	GND
27	AIN7- / AIN15	28	GND
29	AIN8+ / AIN8	30	GND
31	AIN8- / AIN16	32	GND
33	N.C.	34	GND
35	N.C.	36	GND
37	+12 VOLTS	38	GND
39	-12 VOLTS	40	GND

Table B-3—P15 Connector Pin Assignments

Pin No.	Signal Name	Pin No.	Signal Name
1	GND	2	EXTINT
3	PA7	4	PA6
5	PA5	6	PA4
7	PA3	8	PA2
9	PA1	10	PA0
11	GND	12	GND
13	PC7	14	PC6
15	PC5	16	PC4
17	PC3	18	PC2
19	PC1	20	PC0
21	GND	22	GND
23	EXTCLK0	24	EXTGATE0
25	CLKOUT0 / CLKOUT0-	26	GND
27	EXTCLK1	28	EXTGATE1
29	CLKOUT1 / CLKOUT1-	30	GND
31	EXTCLK2	32	EXTGATE2
33	CLKOUT2 / CLKOUT2-	34	GND
35	+5 VOLTS	36	+5 VOLTS
37	GND	38	GND
39	+12 VOLTS	40	-12 VOLTS

APPENDIX C

COMPONENT DATA SHEETS

Intel 82C55A Programmable Peripheral Interface
Data Sheet Reprint

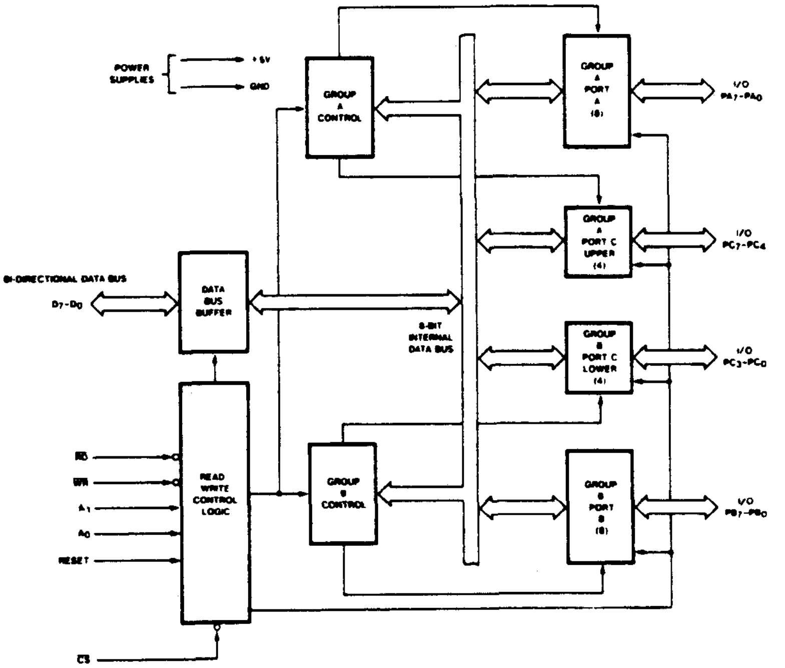
82C55A
CHMOS PROGRAMMABLE PERIPHERAL INTERFACE

- **Compatible with all Intel and Most Other Microprocessors**
- **High Speed, "Zero Wait State" Operation with 8 MHz 8086/88 and 80186/188**
- **24 Programmable I/O Pins**
- **Low Power CHMOS**
- **Completely TTL Compatible**
- **Control Word Read-Back Capability**
- **Direct Bit Set/Reset Capability**
- **2.5 mA DC Drive Capability on all I/O Port Outputs**
- **Available in 40-Pin DIP and 44-Pin PLCC**
- **Available in EXPRESS**
  — Standard Temperature Range
  — Extended Temperature Range

The Intel 82C55A is a high-performance, CHMOS version of the industry standard 8255A general purpose programmable I/O device which is designed for use with all Intel and most other microprocessors. It provides 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. The 82C55A is pin compatible with the NMOS 8255A and 8255A-5.

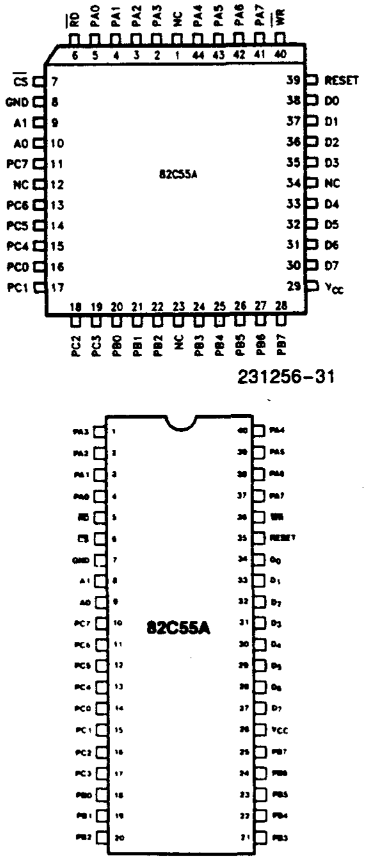
In MODE 0, each group of 12 I/O pins may be programmed in sets of 4 and 8 to be inputs or outputs. In MODE 1, each group may be programmed to have 8 lines of input or output. 3 of the remaining 4 pins are used for handshaking and interrupt control signals. MODE 2 is a strobed bi-directional bus configuration.

The 82C55A is fabricated on Intel's advanced CHMOS III technology which provides low power consumption with performance equal to or greater than the equivalent NMOS product. The 82C55A is available in 40-pin DIP and 44-pin plastic leaded chip carrier (PLCC) packages.



231256-1

Figure 1. 82C55A Block Diagram



231256-2

Figure 2. 82C55A Pinout

Diagrams are for pin reference only. Package sizes are not to scale.

September 1987
Order Number: 231256-004

Table 1. Pin Description

<table>
<tr><th rowspan="2">Symbol</th><th colspan="2">Pin Number</th><th rowspan="2">Type</th><th rowspan="2" colspan="6">Name and Function</th></tr>
<tr><th>Dip</th><th>PLCC</th></tr>
<tr><td>PA_{3-0}</td><td>1–4</td><td>2–5</td><td>I/O</td><td colspan="6"><b>PORT A, PINS 0–3:</b> Lower nibble of an 8-bit data output latch/buffer and an 8-bit data input latch.</td></tr>
<tr><td>$\overline{RD}$</td><td>5</td><td>6</td><td>I</td><td colspan="6"><b>READ CONTROL:</b> This input is low during CPU read operations.</td></tr>
<tr><td>$\overline{CS}$</td><td>6</td><td>7</td><td>I</td><td colspan="6"><b>CHIP SELECT:</b> A low on this input enables the 82C55A to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and WR are ignored otherwise.</td></tr>
<tr><td>GND</td><td>7</td><td>8</td><td></td><td colspan="6"><b>System Ground</b></td></tr>
<tr><td rowspan="16">A_{1-0}</td><td rowspan="16">8–9</td><td rowspan="16">9–10</td><td rowspan="16">I</td><td colspan="6"><b>ADDRESS:</b> These input signals, in conjunction $\overline{RD}$ and $\overline{WR}$, control the selection of one of the three ports or the control word registers.</td></tr>
<tr><th>A_1</th><th>A_0</th><th>$\overline{RD}$</th><th>$\overline{WR}$</th><th>$\overline{CS}$</th><th>Input Operation (Read)</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>Port A - Data Bus</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Port B - Data Bus</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>Port C - Data Bus</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Control Word - Data Bus</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><th>Output Operation (Write)</th></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>Data Bus - Port A</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Data Bus - Port B</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>Data Bus - Port C</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Data Bus - Control</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><th>Disable Function</th></tr>
<tr><td>X</td><td>X</td><td>X</td><td>X</td><td>1</td><td>Data Bus - 3 - State</td></tr>
<tr><td>X</td><td>X</td><td>1</td><td>1</td><td>0</td><td>Data Bus - 3 - State</td></tr>
<tr><td colspan="6"></td></tr>
<tr><td colspan="6"></td></tr>
<tr><td>PC_{7-4}</td><td>10–13</td><td>11,13–15</td><td>I/O</td><td colspan="6"><b>PORT C, PINS 4–7:</b> Upper nibble of an 8-bit data output latch/buffer and an 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.</td></tr>
<tr><td>PC_{0-3}</td><td>14–17</td><td>16–19</td><td>I/O</td><td colspan="6"><b>PORT C, PINS 0–3:</b> Lower nibble of Port C.</td></tr>
<tr><td>PB_{0-7}</td><td>18–25</td><td>20–22,
24–28</td><td>I/O</td><td colspan="6"><b>PORT B, PINS 0–7:</b> An 8-bit data output latch/buffer and an 8-bit data input buffer.</td></tr>
<tr><td>V_{CC}</td><td>26</td><td>29</td><td></td><td colspan="6"><b>SYSTEM POWER:</b> + 5V Power Supply.</td></tr>
<tr><td>D_{7-0}</td><td>27–34</td><td>30–33,
35–38</td><td>I/O</td><td colspan="6"><b>DATA BUS:</b> Bi-directional, tri-state data bus lines, connected to system data bus.</td></tr>
<tr><td>RESET</td><td>35</td><td>39</td><td>I</td><td colspan="6"><b>RESET:</b> A high on this input clears the control register and all ports are set to the input mode.</td></tr>
<tr><td>$\overline{WR}$</td><td>36</td><td>40</td><td>I</td><td colspan="6"><b>WRITE CONTROL:</b> This input is low during CPU write operations.</td></tr>
<tr><td>PA_{7-4}</td><td>37–40</td><td>41–44</td><td>I/O</td><td colspan="6"><b>PORT A, PINS 4–7:</b> Upper nibble of an 8-bit data output latch/buffer and an 8-bit data input latch.</td></tr>
<tr><td>NC</td><td></td><td>1, 12,
23, 34</td><td></td><td colspan="6">No Connect</td></tr>
</table>

82C55A FUNCTIONAL DESCRIPTION

General

The 82C55A is a programmable peripheral interface device designed for use in Intel microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 82C55A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 82C55A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 82C55A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 82C55A.

Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control Logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Control Group A - Port A and Port C upper (C7–C4)
Control Group B - Port B and Port C lower (C3–C0)

The control word register can be both written and read as shown in the address decode table in the pin descriptions. Figure 6 shows the control word format for both Read and Write operations. When the control word is read, bit D7 will always be a logic "1", as this implies control word mode information.

Ports A, B, and C

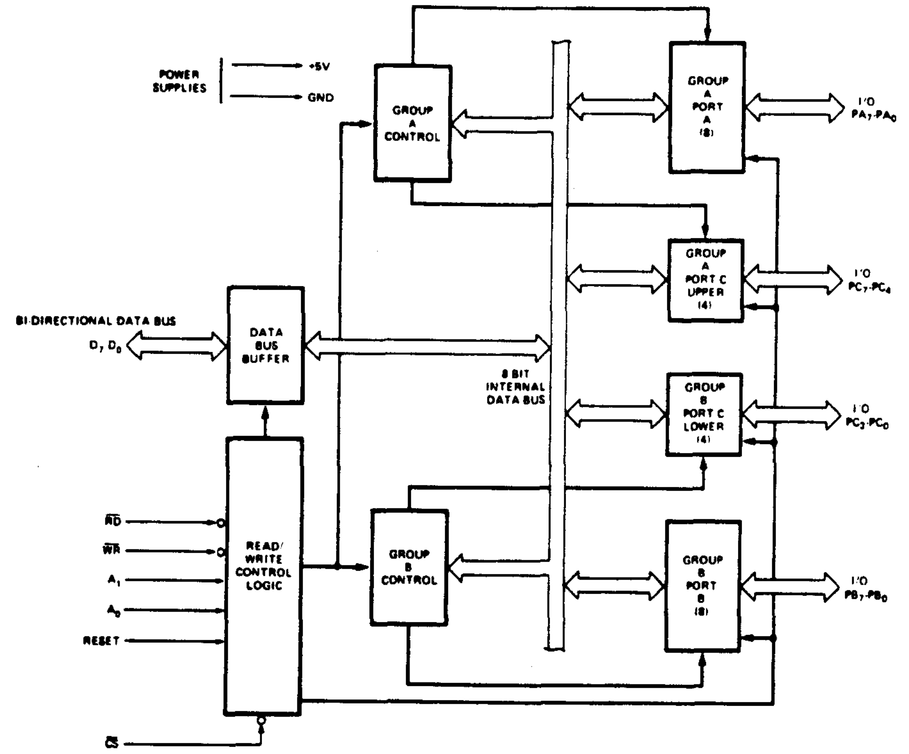
The 82C55A contains three 8-bit ports (A, B, and C). All can be configured in a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 82C55A.

Port A. One 8-bit data output latch/buffer and one 8-bit input latch buffer. Both "pull-up" and "pull-down" bus hold devices are present on Port A.

Port B. One 8-bit data input/output latch/buffer. Only "pull-up" bus hold devices are present on Port B.

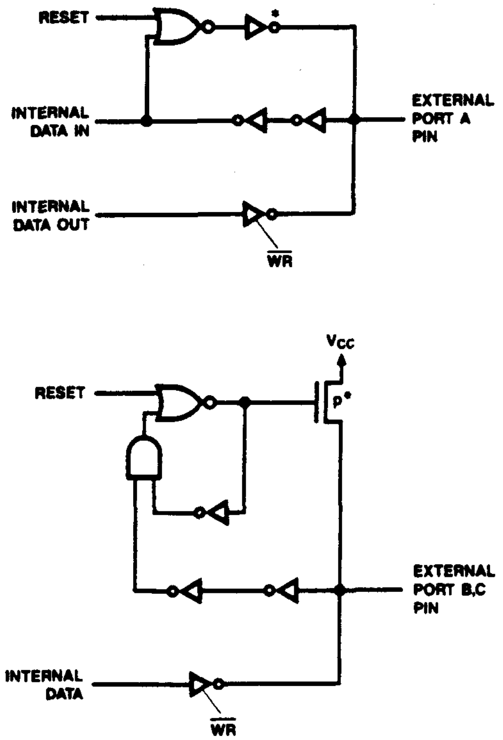
Port C. One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B. Only "pull-up" bus hold devices are present on Port C.

See Figure 4 for the bus-hold circuit configuration for Port A, B, and C.



231256-3

Figure 3. 82C55A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions



231256-4

***NOTE:**
Port pins loaded with more than 20 pF capacitance may not have their logic level guaranteed following a hardware reset.

Figure 4. Port A, B, C, Bus-hold Configuration

82C55A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that can be selected by the system software:

Mode 0 — Basic input/output
Mode 1 — Strobed Input/output
Mode 2 — Bi-directional Bus

When the reset input goes "high" all ports will be set to the input mode with all 24 port lines held at a logic "one" level by the internal bus hold devices (see Figure 4 Note). After the reset is removed the 82C55A can remain in the input mode with no additional initialization required. This eliminates the need for pullup or pulldown devices in "all CMOS" designs. During the execution of the system program, any of the other modes may be selected by using a single output instruction. This allows a single 82C55A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

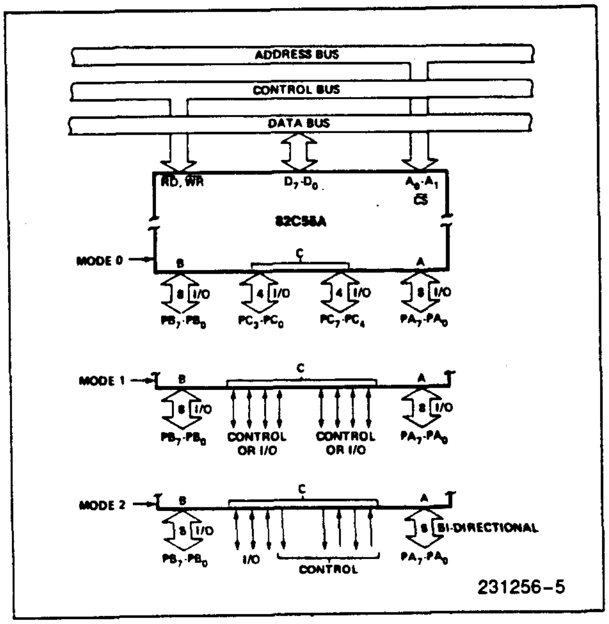


Figure 5. Basic Mode Definitions and Bus Interface

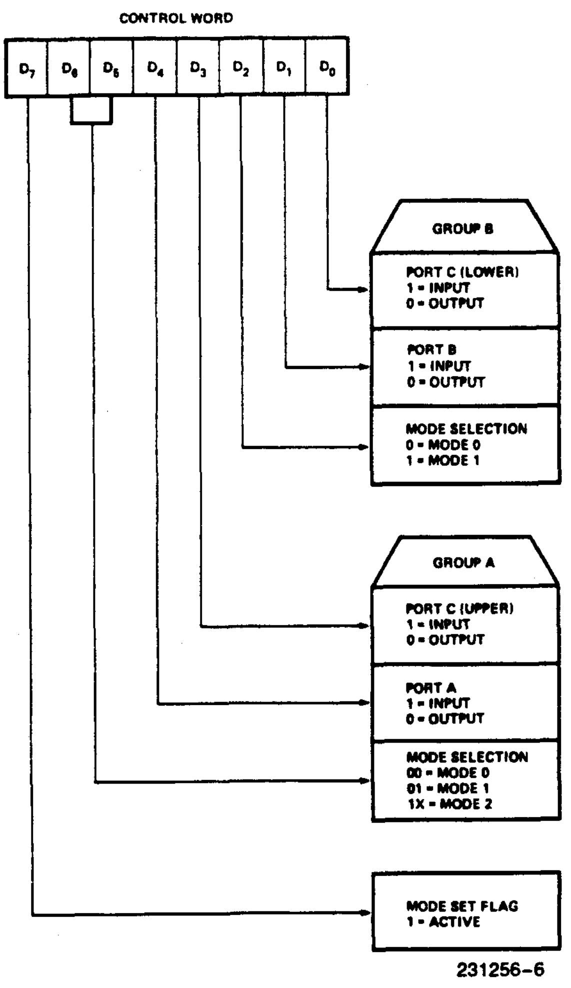


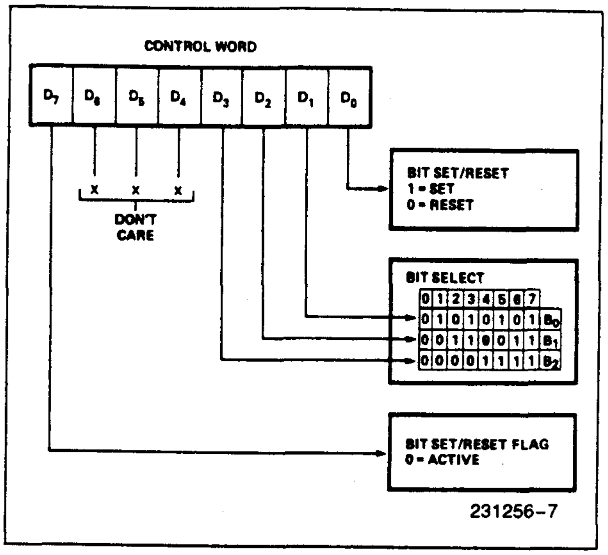
Figure 6. Mode Definition Format

The mode definitions and possible mode combinations may seem confusing at first but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 82C55A has taken into account things such as efficient PC board layout, control signal definition vs PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

Single Bit Set/Reset Feature

Any of the eight bits of Port C can be Set or Reset using a single OUTput instruction. 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.



0	1	2	3	4	5	6	7	
0	1	0	1	0	1	0	1	B_0
0	0	1	1	0	0	1	1	B_1
0	0	0	0	1	1	1	1	B_2

Figure 7. Bit Set/Reset Format

Interrupt Control Functions

When the 82C55A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the Programmer to disallow or allow a specific I/O device to interrupt the CPU without affecting any other device in the interrupt structure.

INTE flip-flop definition:

(BIT-SET)—INTE is SET—Interrupt enable
(BIT-RESET)—INTE is RESET—Interrupt disable

Note:
All Mask flip-flops are automatically reset during mode selection and device Reset.

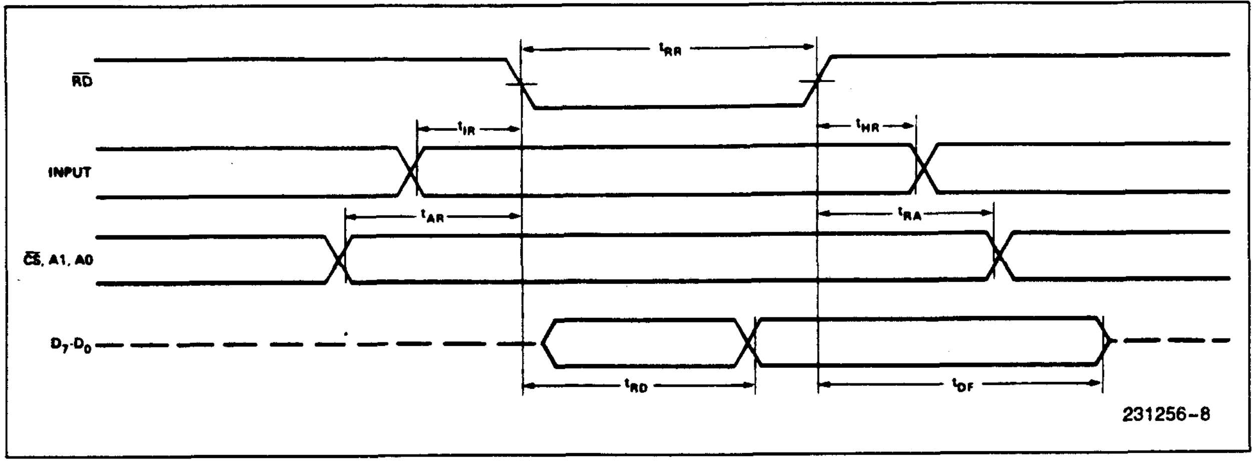
Operating Modes

Mode 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No "handshaking" is required, data is simply written to or read from a specified port.

Mode 0 Basic Functional Definitions:

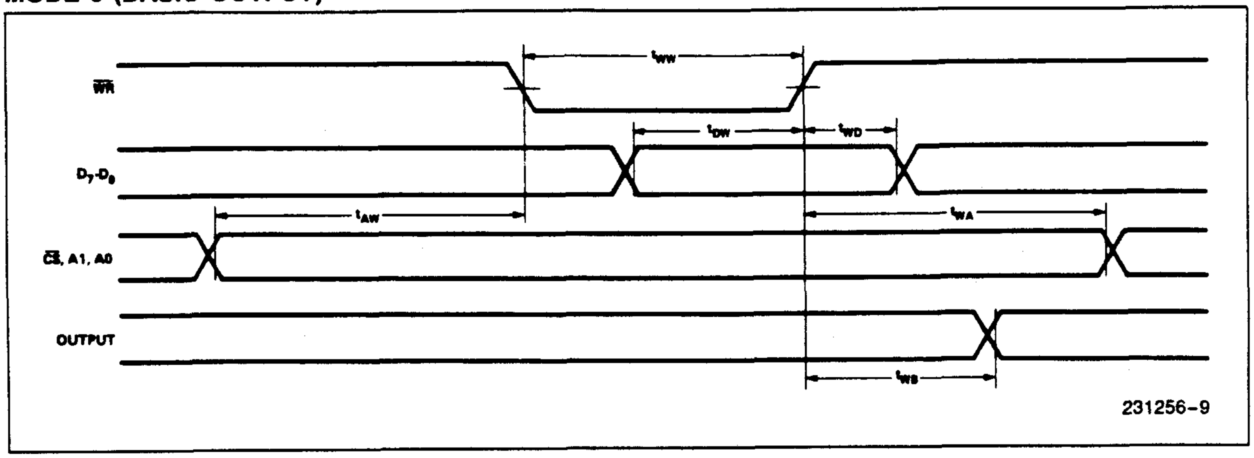
- Two 8-bit ports and two 4-bit ports.
- Any port can be input or output.
- Outputs are latched.
- Inputs are not latched.
- 16 different Input/Output configurations are possible in this Mode.

MODE 0 (BASIC INPUT)



231256-8

MODE 0 (BASIC OUTPUT)

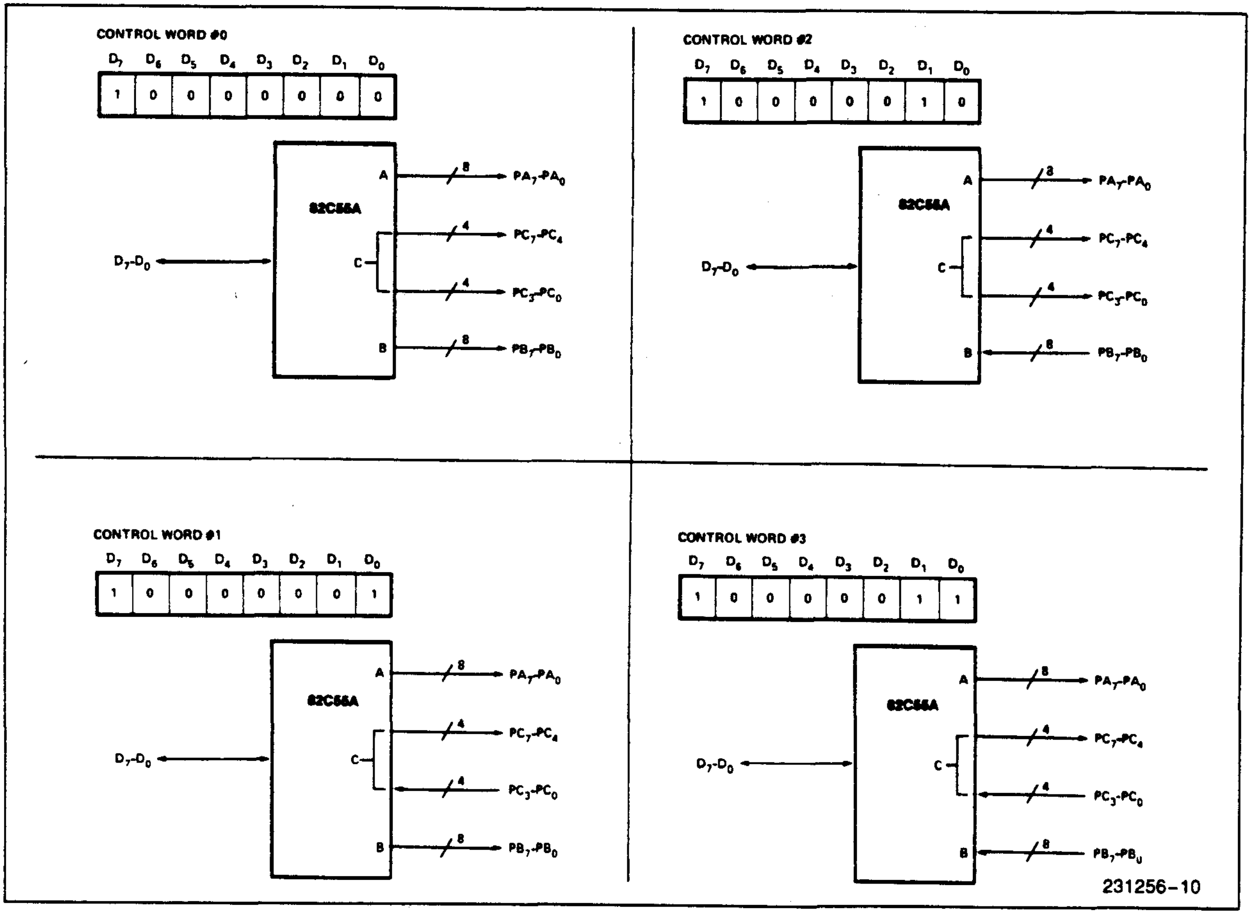


231256-9

MODE 0 Port Definition

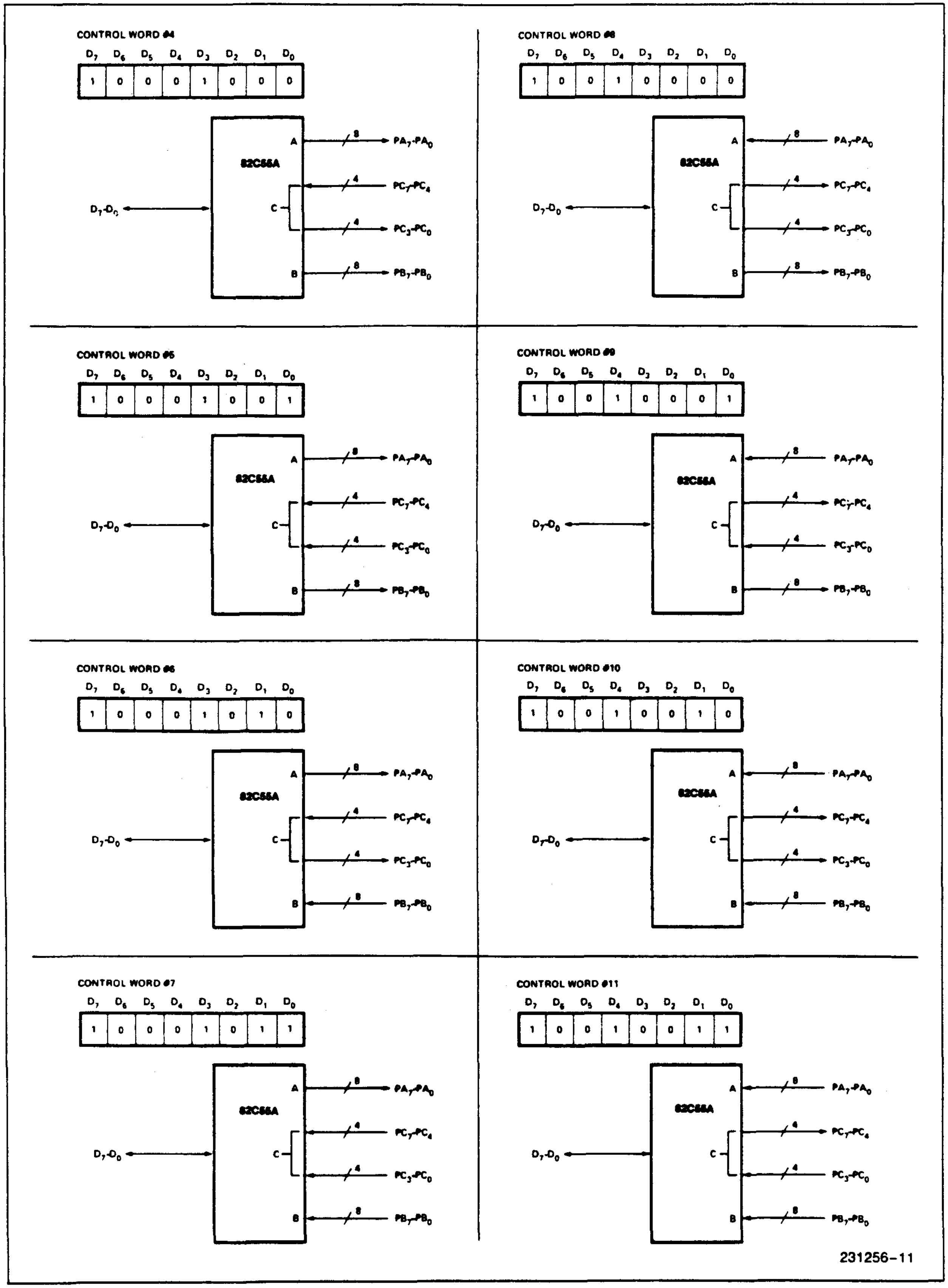
A		B		GROUP A			GROUP B	
D_4	D_3	D_1	D_0	PORT A	PORT C (UPPER)	#	PORT B	PORT C (LOWER)
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUTPUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INPUT
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUTPUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INPUT
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUTPUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INPUT
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUTPUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INPUT
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUTPUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INPUT
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUTPUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INPUT
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUTPUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	INPUT
1	1	1	0	INPUT	INPUT	14	INPUT	OUTPUT
1	1	1	1	INPUT	INPUT	15	INPUT	INPUT

MODE 0 Configurations



231256-10

MODE 0 Configurations (Continued)



CONTROL WORD #4

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	0	0	0

CONTROL WORD #5

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	0	0	1

CONTROL WORD #6

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	0	1	0

CONTROL WORD #7

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	0	1	0	1	1

CONTROL WORD #8

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	0	0	0	0

CONTROL WORD #9

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	0	0	0	1

CONTROL WORD #10

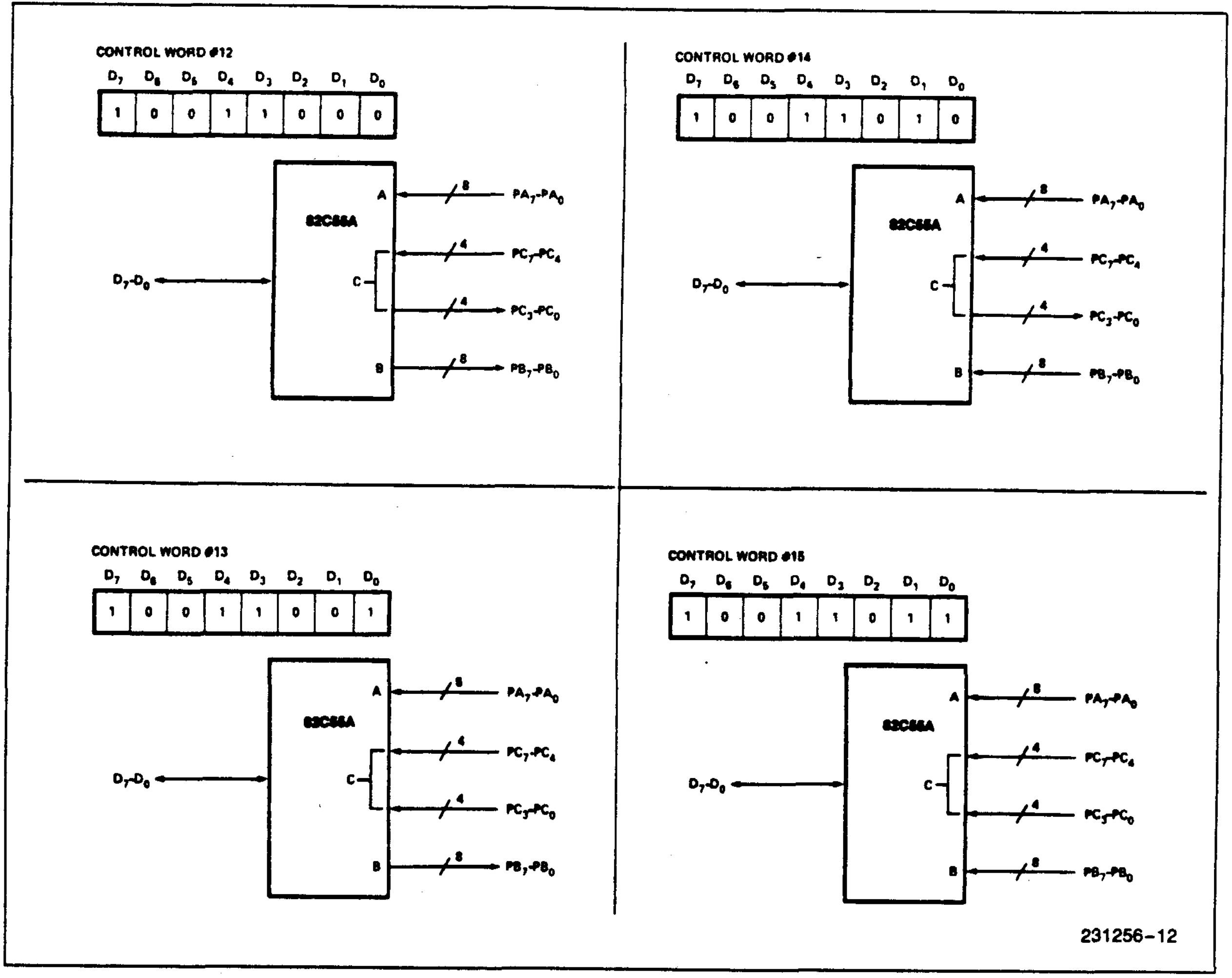
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	0	0	1	0

CONTROL WORD #11

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	0	0	1	0	0	1	1

231256-11

MODE 0 Configurations (Continued)



231256-12

Operating Modes

MODE 1 (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or "handshaking" signals. In mode 1, Port A and Port B use the lines on Port C to generate or accept these "handshaking" signals.

Mode 1 Basic functional Definitions:

- Two Groups (Group A and Group B).
- Each group contains one 8-bit data port and one 4-bit control/data port.
- The 8-bit data port can be either input or output Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit data port.

Input Control Signal Definition

$\overline{STB}$ (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch; in essence, an acknowledgement. IBF is set by $\overline{STB}$ input being low and is reset by the rising edge of the $\overline{RD}$ input.

INTR (Interrupt Request)

A "high" on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the $\overline{STB}$ is a "one", IBF is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{RD}$. This procedure allows an input device to request service from the CPU by simply strobing its data into the port.

INTE A

Controlled by bit set/reset of PC_4.

INTE B

Controlled by bit set/reset of PC_2.

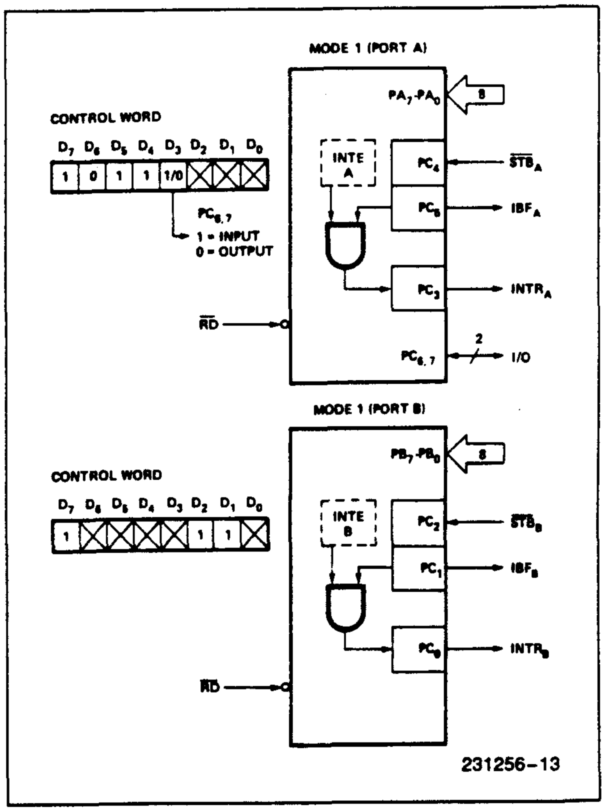


Figure 8. MODE 1 Input

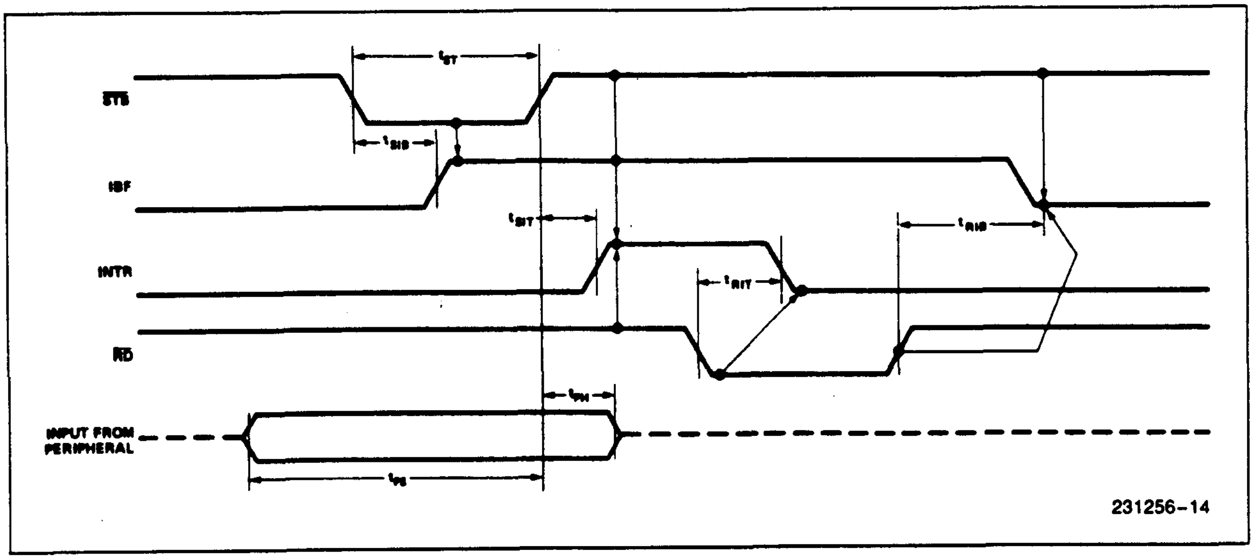


Figure 9. MODE 1 (Strobed Input)

Output Control Signal Definition

$\overline{\text{OBF}}$ (Output Buffer Full F/F). The $\overline{\text{OBF}}$ output will go "low" to indicate that the CPU has written data out to the specified port. The $\overline{\text{OBF}}$ F/F will be set by the rising edge of the $\overline{\text{WR}}$ input and reset by $\overline{\text{ACK}}$ Input being low.

$\overline{\text{ACK}}$ (Acknowledge Input). A "low" on this input informs the 82C55A that the data from Port A or Port B has been accepted. In essence, a response from the peripheral device indicating that it has received the data output by the CPU.

INTR (Interrupt Request). A "high" on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when $\overline{\text{ACK}}$ is a "one", $\overline{\text{OBF}}$ is a "one" and INTE is a "one". It is reset by the falling edge of $\overline{\text{WR}}$.

INTE A

Controlled by bit set/reset of PC_6.

INTE B

Controlled by bit set/reset of PC_2.

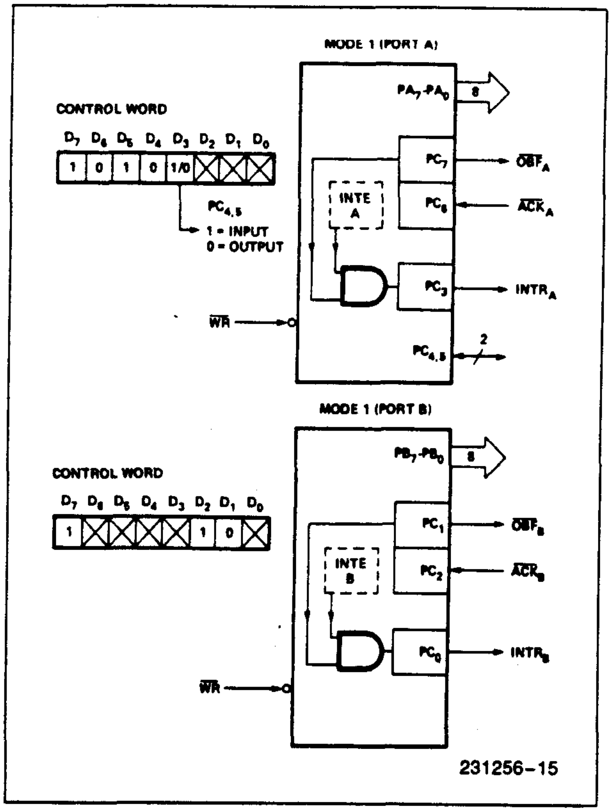


Figure 10. MODE 1 Output

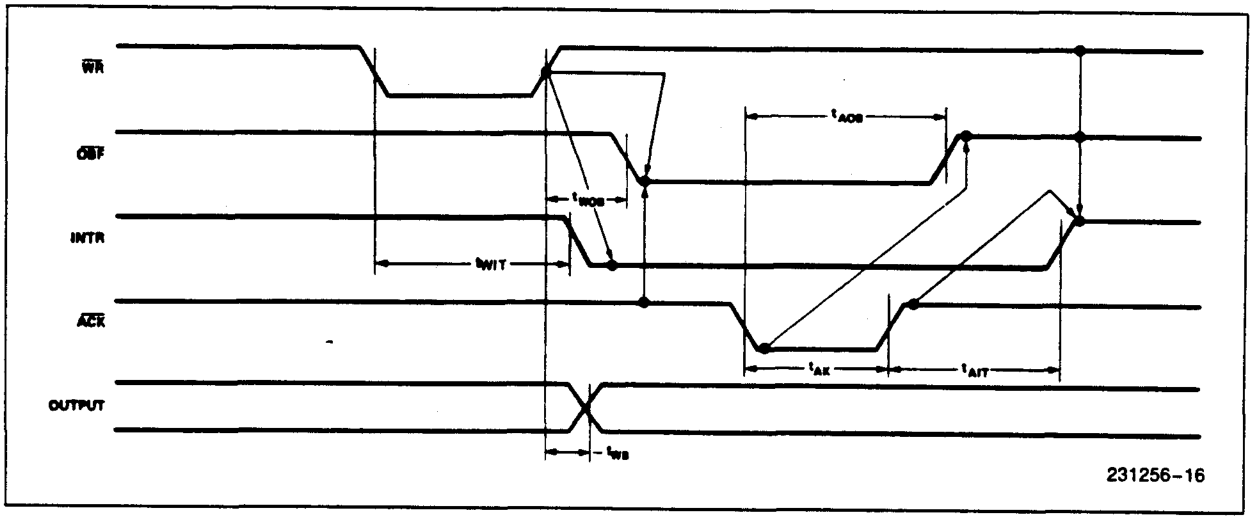


Figure 11. MODE 1 (Strobed Output)

Combinations of MODE 1

Port A and Port B can be individually defined as input or output in Mode 1 to support a wide variety of strobed I/O applications.

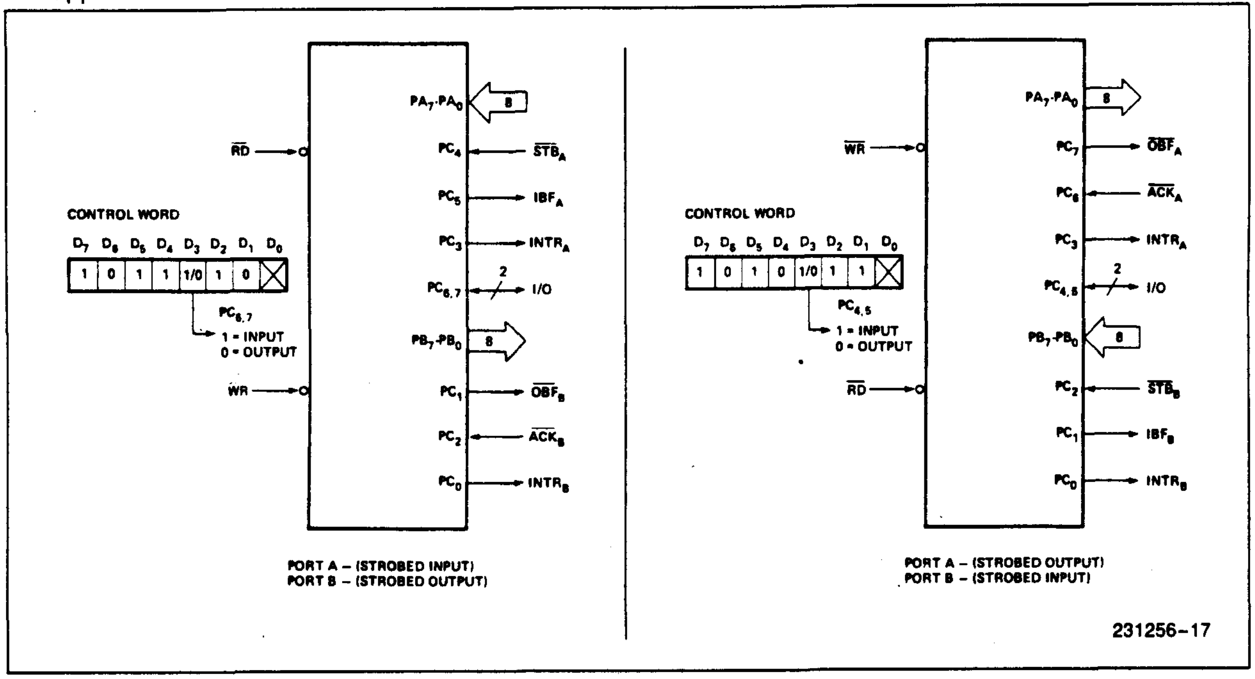


Figure 12. Combinations of MODE 1

Operating Modes

MODE 2 (Strobed Bidirectional Bus I/O). This functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bidirectional bus I/O). "Handshaking" signals are provided to maintain proper bus flow discipline in a similar manner to MODE 1. Interrupt generation and enable/disable functions are also available.

MODE 2 Basic Functional Definitions:

- Used in Group A **only**.
- One 8-bit, bi-directional bus port (Port A) and a 5-bit control port (Port C).
- Both inputs and outputs are latched.
- The 5-bit control port (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A).

Bidirectional Bus I/O Control Signal Definition

INTR (Interrupt Request). A high on this output can be used to interrupt the CPU for input or output operations.

Output Operations

$\overline{\text{OBF}}$ (Output Buffer Full). The $\overline{\text{OBF}}$ output will go "low" to indicate that the CPU has written data out to port A.

$\overline{\text{ACK}}$ (Acknowledge). A "low" on this input enables the tri-state output buffer of Port A to send out the data. Otherwise, the output buffer will be in the high impedance state.

INTE 1 (The INTE Flip-Flop Associated with $\overline{\text{OBF}}$). Controlled by bit set/reset of PC_6.

Input Operations

$\overline{\text{STB}}$ (Strobe Input). A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F). A "high" on this output indicates that data has been loaded into the input latch.

INTE 2 (The INTE Flip-Flop Associated with IBF). Controlled by bit set/reset of PC_4.

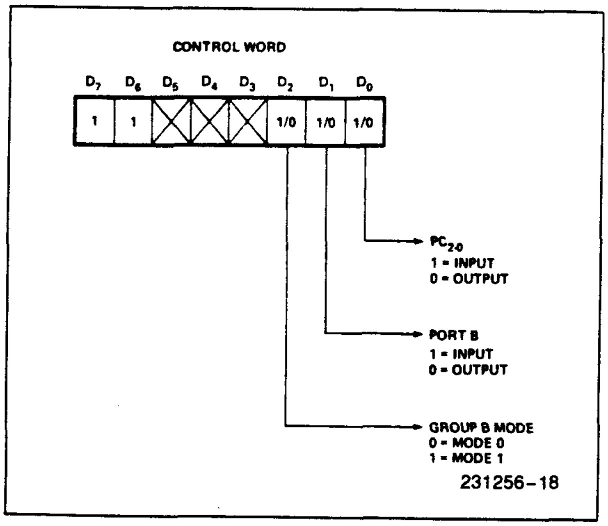


Figure 13. MODE Control Word

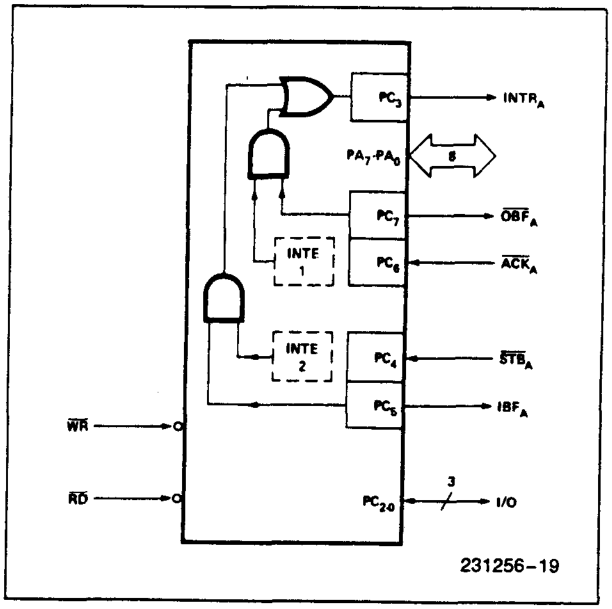


Figure 14. MODE 2

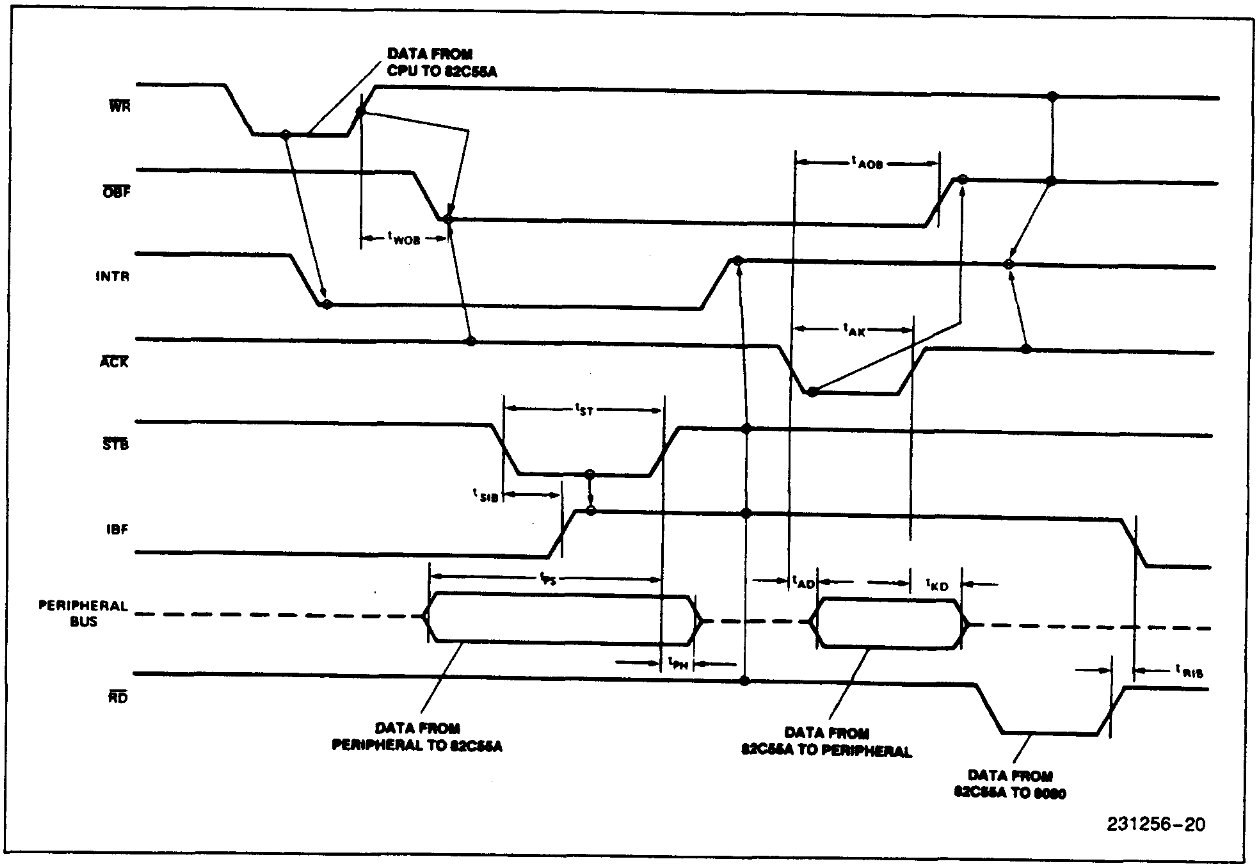


Figure 15. MODE 2 (Bidirectional)

NOTE:
Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$, and $\overline{STB}$ occurs before $\overline{RD}$ is permissible.
$(INTR = IBF \bullet \overline{MASK} \bullet \overline{STB} \bullet \overline{RD} + \overline{OBF} \bullet \overline{MASK} \bullet \overline{ACK} \bullet \overline{WR})$

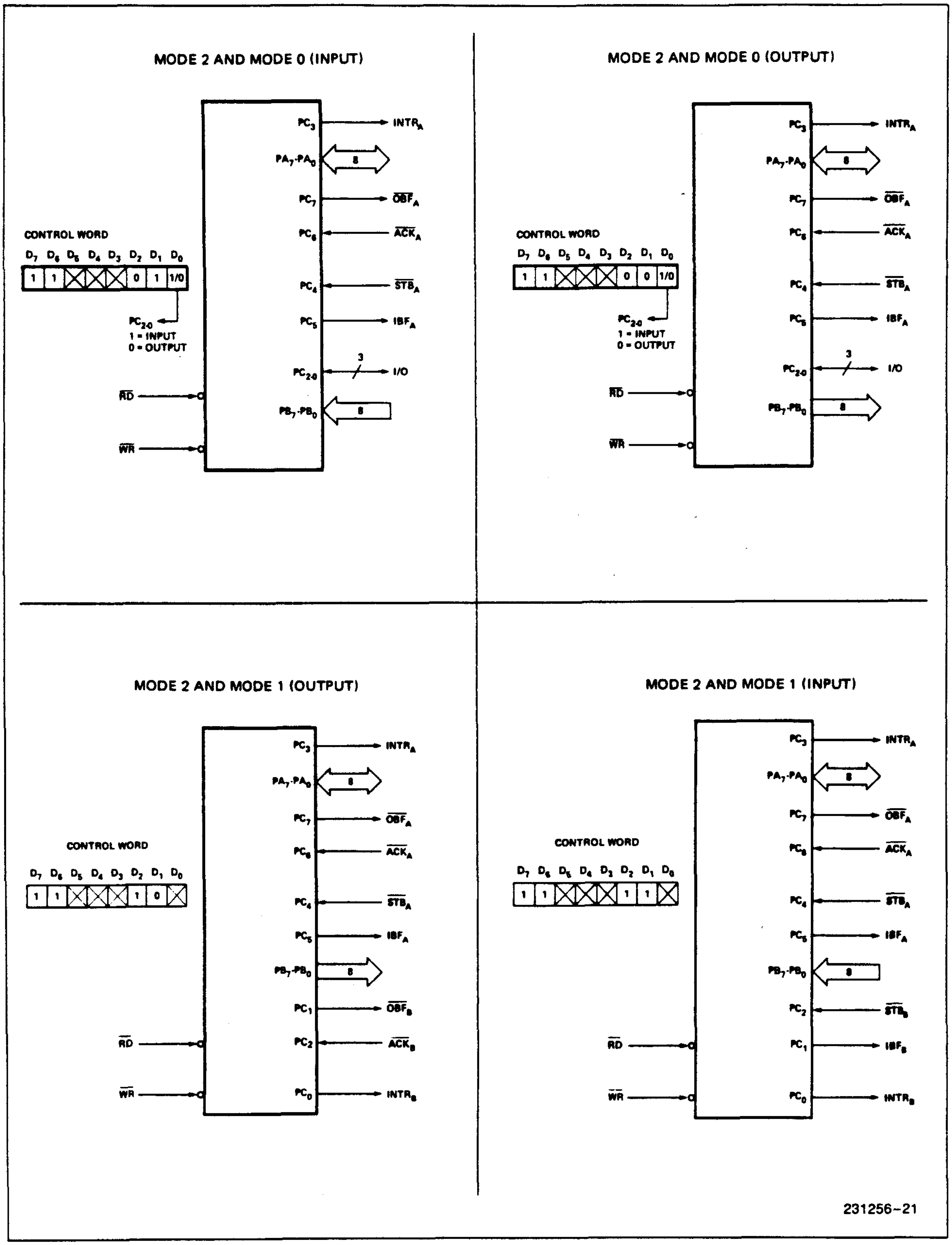


Figure 16. MODE ¼ Combinations

Mode Definition Summary

	MODE 0		MODE 1		MODE 2
	IN	OUT	IN	OUT	GROUP A ONLY
PA_0	IN	OUT	IN	OUT	⟷
PA_1	IN	OUT	IN	OUT	⟷
PA_2	IN	OUT	IN	OUT	⟷
PA_3	IN	OUT	IN	OUT	⟷
PA_4	IN	OUT	IN	OUT	⟷
PA_5	IN	OUT	IN	OUT	⟷
PA_6	IN	OUT	IN	OUT	⟷
PA_7	IN	OUT	IN	OUT	⟷
PB_0	IN	OUT	IN	OUT	—
PB_1	IN	OUT	IN	OUT	—
PB_2	IN	OUT	IN	OUT	—
PB_3	IN	OUT	IN	OUT	—
PB_4	IN	OUT	IN	OUT	—
PB_5	IN	OUT	IN	OUT	—
PB_6	IN	OUT	IN	OUT	—
PB_7	IN	OUT	IN	OUT	—
PC_0	IN	OUT	INTR_B	INTR_B	I/O
PC_1	IN	OUT	IBF_B	$\overline{\text{OBF}}_B$	I/O
PC_2	IN	OUT	$\overline{\text{STB}}_B$	$\overline{\text{ACK}}_B$	I/O
PC_3	IN	OUT	INTR_A	INTR_A	INTR_A
PC_4	IN	OUT	$\overline{\text{STB}}_A$	I/O	$\overline{\text{STB}}_A$
PC_5	IN	OUT	IBF_A	I/O	IBF_A
PC_6	IN	OUT	I/O	$\overline{\text{ACK}}_A$	$\overline{\text{ACK}}_A$
PC_7	IN	OUT	I/O	$\overline{\text{OBF}}_A$	$\overline{\text{OBF}}_A$

MODE 0 OR MODE 1 ONLY

Special Mode Combination Considerations

There are several combinations of modes possible. For any combination, some or all of the Port C lines are used for control or status. The remaining bits are either inputs or outputs as defined by a "Set Mode" command.

During a read of Port C, the state of all the Port C lines, except the $\overline{\text{ACK}}$ and $\overline{\text{STB}}$ lines, will be placed on the data bus. In place of the $\overline{\text{ACK}}$ and $\overline{\text{STB}}$ line states, flag status will appear on the data bus in the PC2, PC4, and PC6 bit positions as illustrated by Figure 18.

Through a "Write Port C" command, only the Port C pins programmed as outputs in a Mode 0 group can be written. No other pins can be affected by a "Write Port C" command, nor can the interrupt enable flags be accessed. To write to any Port C output programmed as an output in a Mode 1 group or to change an interrupt enable flag, the "Set/Reset Port C Bit" command must be used.

With a "Set/Reset Port C Bit" command, any Port C line programmed as an output (including INTR, IBF and $\overline{\text{OBF}}$) can be written, or an interrupt enable flag can be either set or reset. Port C lines programmed as inputs, including $\overline{\text{ACK}}$ and $\overline{\text{STB}}$ lines, associated with Port C are not affected by a "Set/Reset Port C Bit" command. Writing to the corresponding Port C bit positions of the $\overline{\text{ACK}}$ and $\overline{\text{STB}}$ lines with the "Set/Reset Port C Bit" command will affect the Group A and Group B interrupt enable flags, as illustrated in Figure 18.

Current Drive Capability

Any output on Port A, B or C can sink or source 2.5 mA. This feature allows the 82C55A to directly drive Darlington type drivers and high-voltage displays that require such sink or source current.

Reading Port C Status

In Mode 0, Port C transfers data to or from the peripheral device. When the 82C55A is programmed to function in Modes 1 or 2, Port C generates or accepts "hand-shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is no special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

INPUT CONFIGURATION

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
I/O	I/O	IBF_A	INTE_A	INTR_A	INTE_B	IBF_B	INTR_B
GROUP A					GROUP B		

OUTPUT CONFIGURATIONS

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
$\overline{\text{OBF}}_A$	INTE_A	I/O	I/O	INTR_A	INTE_B	$\overline{\text{OBF}}_B$	INTR_B
GROUP A					GROUP B		

Figure 17a. MODE 1 Status Word Format

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
$\overline{\text{OBF}}_A$	INTE_1	IBF_A	INTE_2	INTR_A			
GROUP A					GROUP B		

(Defined By Mode 0 or Mode 1 Selection)

Figure 17b. MODE 2 Status Word Format

Interrupt Enable Flag	Position	Alternate Port C Pin Signal (Mode)
INTE B	PC2	$\overline{\text{ACK}}_B$ (Output Mode 1) or $\overline{\text{STB}}_B$ (Input Mode 1)
INTE A2	PC4	$\overline{\text{STB}}_A$ (Input Mode 1 or Mode 2)
INTE A1	PC6	$\overline{\text{ACK}}_A$ (Output Mode 1 or Mode 2

Figure 18. Interrupt Enable Flags In Modes 1 and 2

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias....0°C to + 70°C
Storage Temperature ....− 65°C to + 150°C
Supply Voltage ....− 0.5 to + 8.0V
Operating Voltage ....+ 4V to + 7V
Voltage on any Input..........GND−2V to + 6.5V
Voltage on any Output ..GND−0.5V to V_{CC} + 0.5V
Power Dissipation ....1 Watt

**Notice: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D.C. CHARACTERISTICS

T_A = 0°C to 70°C, V_{CC} = +5V ±10%, GND = 0V (T_A = −40°C to +85°C for Extended Temperture)

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	−0.5	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC}	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2.5 mA
V_{OH}	Output High Voltage	3.0 V_{CC} − 0.4		V V	I_{OH} = −2.5 mA I_{OH} = −100 μA
I_{IL}	Input Leakage Current		±1	μA	V_{IN} = V_{CC} to 0V (Note 1)
I_{OFL}	Output Float Leakage Current		±10	μA	V_{IN} = V_{CC} to 0V (Note 2)
I_{DAR}	Darlington Drive Current	±2.5	(Note 4)	mA	Ports A, B, C R_{ext} = 500Ω V_{ext} = 1.7V
I_{PHL}	Port Hold Low Leakage Current	+50	+300	μA	V_{OUT} = 1.0V Port A only
I_{PHH}	Port Hold High Leakage Current	−50	−300	μA	V_{OUT} = 3.0V Ports A, B, C
I_{PHLO}	Port Hold Low Overdrive Current	−350		μA	V_{OUT} = 0.8V
I_{PHHO}	Port Hold High Overdrive Current	+350		μA	V_{OUT} = 3.0V
I_{CC}	V_{CC} Supply Current		10	mA	(Note 3)
I_{CCSB}	V_{CC} Supply Current-Standby		10	μA	V_{CC} = 5.5V V_{IN} = V_{CC} or GND Port Conditions If I/P = Open/High O/P = Open Only With Data Bus = High/Low $\overline{CS}$ = High Reset = Low Pure Inputs = Low/High

NOTES:
1. Pins A_1, A_0, $\overline{CS}$, $\overline{WR}$, $\overline{RD}$, Reset.
2. Data Bus; Ports B, C.
3. Outputs open.
4. Limit output current to 4.0 mA.

CAPACITANCE

T_A = 25°C, V_{CC} = GND = 0V

Symbol	Parameter	Min	Max	Units	Test Conditions
C_{IN}	Input Capacitance		10	pF	Unmeasured pins returned to GND f_c = 1 MHz[5]
$C_{I/O}$	I/O Capacitance		20	pF	

NOTE:

[5] Sampled not 100% tested.

A.C. CHARACTERISTICS

T_A = 0° to 70°C, V_{CC} = +5V ±10%, GND = 0V

T_A = −40°C to +85°C for Extended Temperature

BUS PARAMETERS

READ CYCLE

Symbol	Parameter	82C55A-2		Units	Test Conditions
		Min	Max		
t_{AR}	Address Stable Before $\overline{RD}$ ↓	0		ns	
t_{RA}	Address Hold Time After $\overline{RD}$ ↑	0		ns	
t_{RR}	$\overline{RD}$ Pulse Width	150		ns	
t_{RD}	Data Delay from $\overline{RD}$ ↓		120	ns	
t_{DF}	$\overline{RD}$ ↑ to Data Floating	10	75	ns	
t_{RV}	Recovery Time between $\overline{RD}/\overline{WR}$	200		ns	

WRITE CYCLE

Symbol	Parameter	82C55A-2		Units	Test Conditions
		Min	Max		
t_{AW}	Address Stable Before $\overline{WR}$ ↓	0		ns	
t_{WA}	Address Hold Time After $\overline{WR}$ ↑	20		ns	Ports A & B
		20		ns	Port C
t_{WW}	$\overline{WR}$ Pulse Width	100		ns	
t_{DW}	Data Setup Time Before $\overline{WR}$ ↑	100		ns	
t_{WD}	Data Hold Time After $\overline{WR}$ ↑	30		ns	Ports A & B
		30		ns	Port C

OTHER TIMINGS

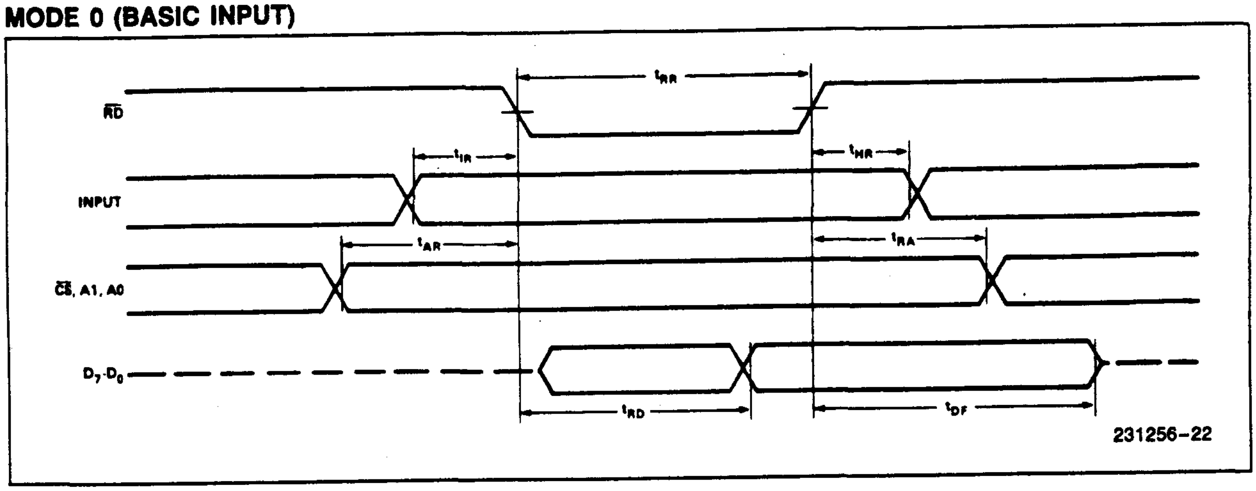
Symbol	Parameter	82C55A-2		Units Conditions	Test
		Min	Max		
t_{WB}	$\overline{WR}$ = 1 to Output		350	ns	
t_{IR}	Peripheral Data Before $\overline{RD}$	0		ns	
t_{HR}	Peripheral Data After $\overline{RD}$	0		ns	
t_{AK}	$\overline{ACK}$ Pulse Width	200		ns	
t_{ST}	$\overline{STB}$ Pulse Width	100		ns	
t_{PS}	Per. Data Before $\overline{STB}$ High	20		ns	
t_{PH}	Per. Data After $\overline{STB}$High	50		ns	
t_{AD}	$\overline{ACK}$ = 0 to Output		175	ns	
t_{KD}	$\overline{ACK}$ = 1 to Output Float	20	250	ns	
t_{WOB}	$\overline{WR}$ = 1 to $\overline{OBF}$ = O		150	ns	
t_{AOB}	$\overline{ACK}$ = 0 to $\overline{OBF}$ = 1		150	ns	
t_{SIB}	$\overline{STB}$ = 0 to IBF = 1		150	ns	
t_{RIB}	$\overline{RD}$ = 1 to IBF = 0		150	ns	
t_{RIT}	$\overline{RD}$ = 0 to INTR = 0		200	ns	
t_{SIT}	$\overline{STB}$ = 1 to INTR = 1		150	ns	
t_{AIT}	$\overline{ACK}$ = 1 to INTR = 1		150	ns	
t_{WIT}	$\overline{WR}$ = 0 to INTR = 0		200	ns	see note 1
t_{RES}	Reset Pulse Width	500		ns	see note 2

NOTE:

1. INTR ↑ may occur as early as $\overline{WR}$ ↓.
2. Pulse width of initial Reset pulse after power on must be at least 50 μSec. Subsequent Reset pulses may be 500 ns minimum.

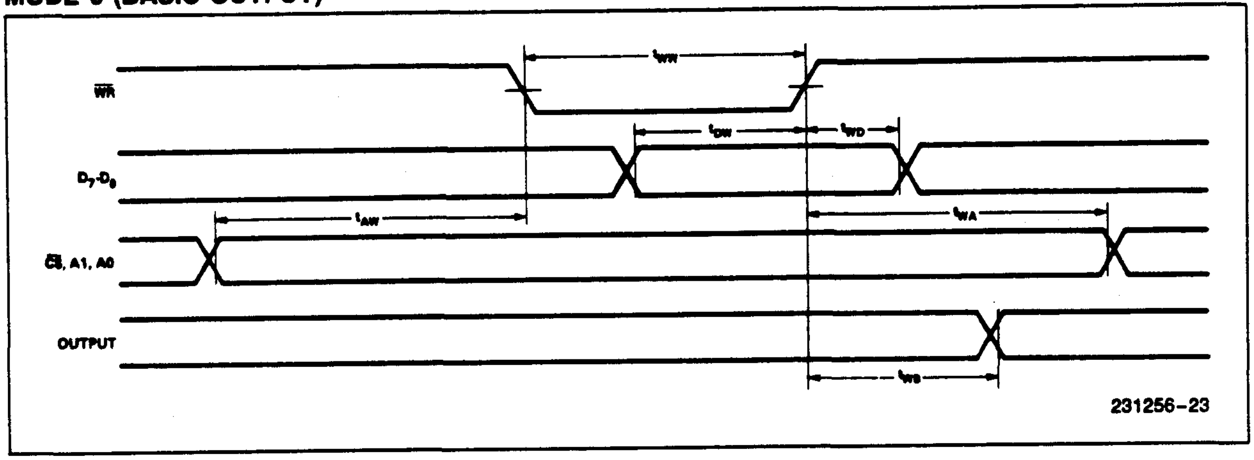
WAVEFORMS

MODE 0 (BASIC INPUT)



231256–22

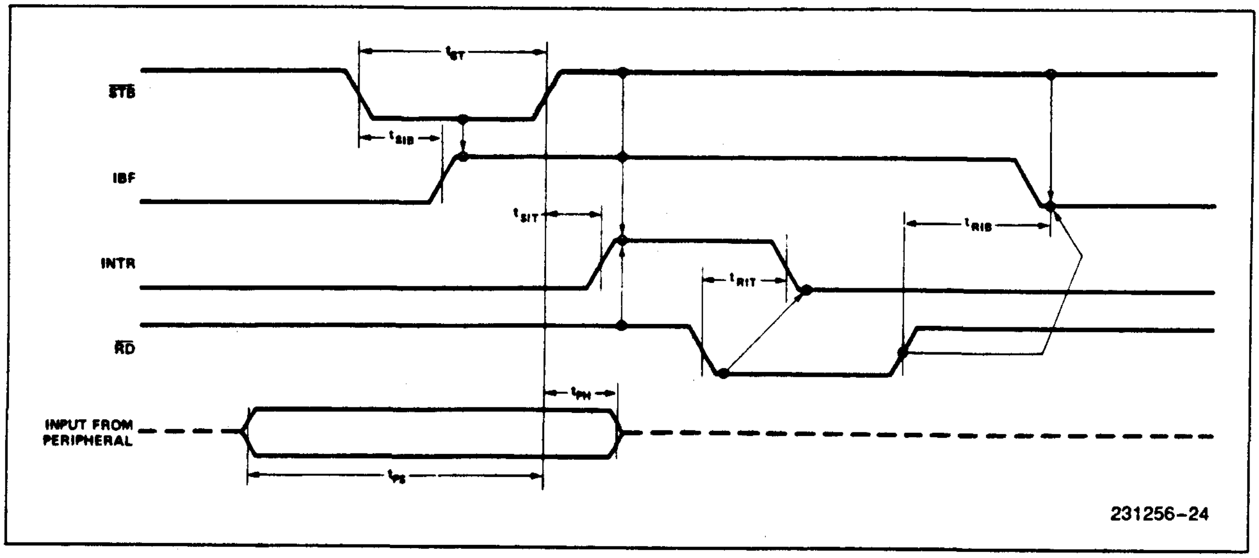
MODE 0 (BASIC OUTPUT)



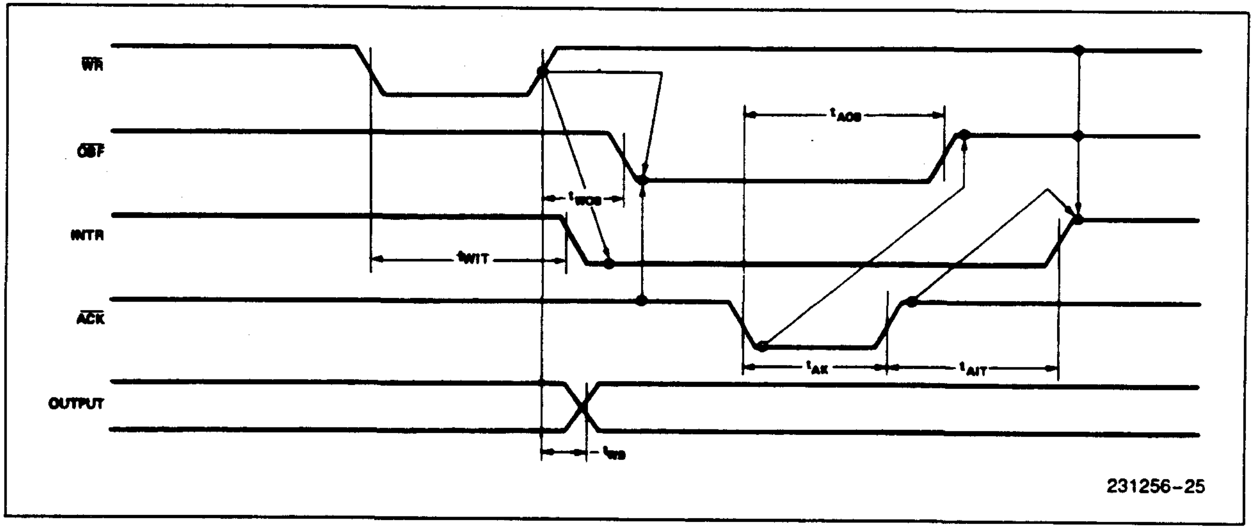
231256–23

WAVEFORMS (Continued)

MODE 1 (STROBED INPUT)

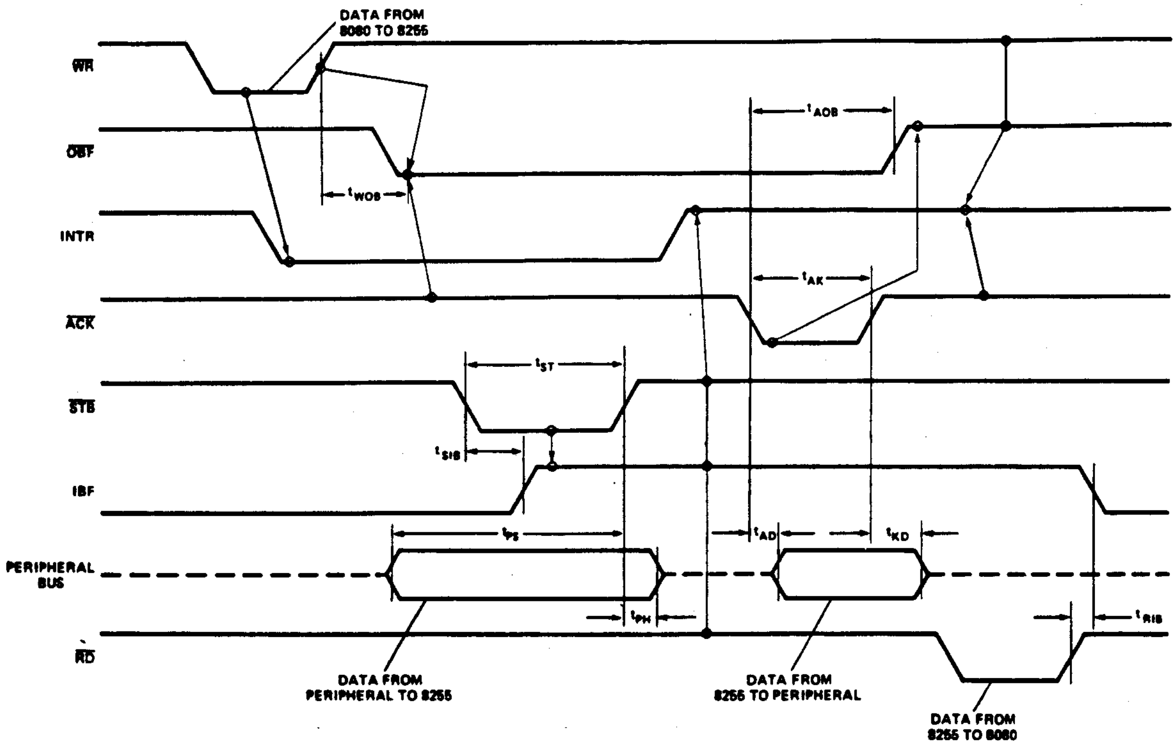


MODE 1 (STROBED OUTPUT)



WAVEFORMS (Continued)

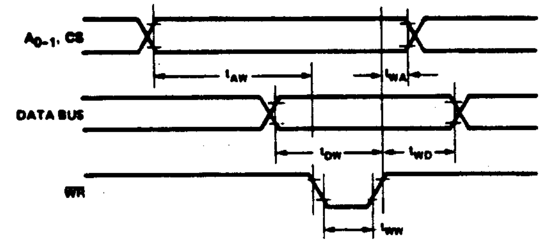
MODE 2 (BIDIRECTIONAL)



231256-26

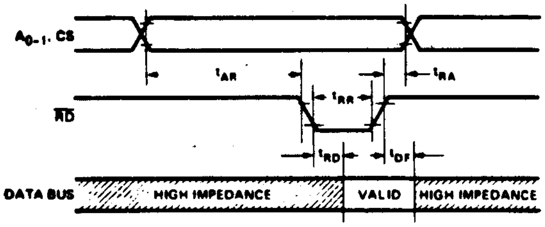
Note:
Any sequence where $\overline{WR}$ occurs before $\overline{ACK}$ AND $\overline{STB}$ occurs before $\overline{RD}$ is permissible.
(INTR = IBF • $\overline{MASK}$ • $\overline{STB}$ • $\overline{RD}$ + $\overline{OBF}$ • $\overline{MASK}$ • $\overline{ACK}$ • $\overline{WR}$)

WRITE TIMING



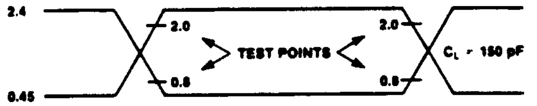
231256-27

READ TIMING



231256-28

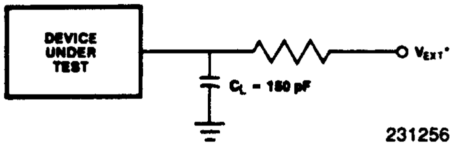
A.C. TESTING INPUT, OUTPUT WAVEFORM



231256-29

A.C. Testing Inputs Are Driven At 2.4V For A Logic 1 And 0.45V For A Logic 0 Timing Measurements Are Made At 2.0V For A Logic 1 And 0.8 For A Logic 0.

A.C. TESTING LOAD CIRCUIT



231256-30

*V_{EXT} Is Set At Various Voltages During Testing To Guarantee The Specification. C_L Includes Jig Capacitance.

Intel 82C54 Programmable Interval Timer
Data Sheet Reprint

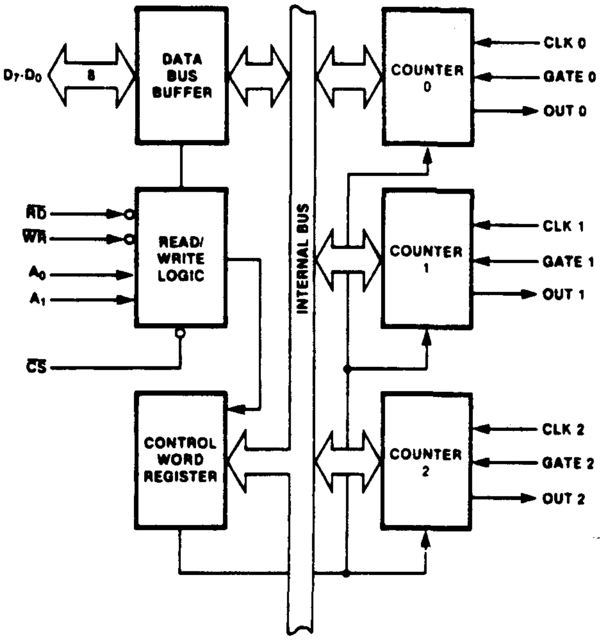
82C54
CHMOS PROGRAMMABLE INTERVAL TIMER

- **Compatible with all Intel and most other microprocessors**
- **High Speed, "Zero Wait State" Operation with 8 MHz 8086/88 and 80186/188**
- **Handles Inputs from DC to 8 MHz**
  — 10 MHz for 82C54-2
- **Available in EXPRESS**
  — Standard Temperature Range
  — Extended Temperature Range
- **Three independent 16-bit counters**
- **Low Power CHMOS**
  — I_{CC} = 10 mA @ 8 MHz Count frequency
- **Completely TTL Compatible**
- **Six Programmable Counter Modes**
- **Binary or BCD counting**
- **Status Read Back Command**
- **Available in 24-Pin DIP and 28-Pin PLCC**

The Intel 82C54 is a high-performance, CHMOS version of the industry standard 8254 counter/timer which is designed to solve the timing control problems common in microcomputer system design. It provides three independent 16-bit counters, each capable of handling clock inputs up to 10 MHz. All modes are software programmable. The 82C54 is pin compatible with the HMOS 8254, and is a superset of the 8253.

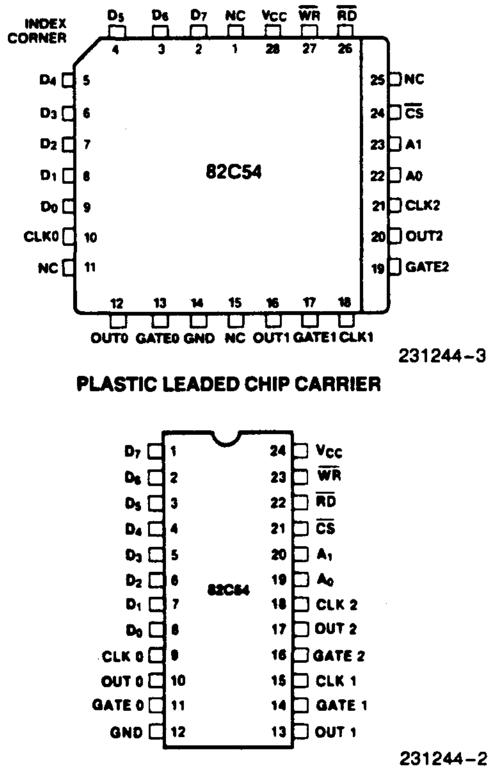
Six programmable timer modes allow the 82C54 to be used as an event counter, elapsed time indicator, programmable one-shot, and in many other applications.

The 82C54 is fabricated on Intel's advanced CHMOS III technology which provides low power consumption with performance equal to or greater than the equivalent HMOS product. The 82C54 is available in 24-pin DIP and 28-pin plastic leaded chip carrier (PLCC) packages.



231244-1

Figure 1. 82C54 Block Diagram



231244-3

PLASTIC LEADED CHIP CARRIER

231244-2

Diagrams are for pin reference only.
Package sizes are not to scale.

Figure 2. 82C54 Pinout

September 1989
Order Number: 231244-005

Table 1. Pin Description

<table>
<tr><th rowspan="2">Symbol</th><th colspan="2">Pin Number</th><th rowspan="2">Type</th><th rowspan="2">Function</th></tr>
<tr><th>DIP</th><th>PLCC</th></tr>
<tr><td>D_7-D_0</td><td>1-8</td><td>2-9</td><td>I/O</td><td>Data: Bidirectional tri-state data bus lines, connected to system data bus.</td></tr>
<tr><td>CLK 0</td><td>9</td><td>10</td><td>I</td><td>Clock 0: Clock input of Counter 0.</td></tr>
<tr><td>OUT 0</td><td>10</td><td>12</td><td>O</td><td>Output 0: Output of Counter 0.</td></tr>
<tr><td>GATE 0</td><td>11</td><td>13</td><td>I</td><td>Gate 0: Gate input of Counter 0.</td></tr>
<tr><td>GND</td><td>12</td><td>14</td><td></td><td>Ground: Power supply connection.</td></tr>
<tr><td>OUT 1</td><td>13</td><td>16</td><td>O</td><td>Out 1: Output of Counter 1.</td></tr>
<tr><td>GATE 1</td><td>14</td><td>17</td><td>I</td><td>Gate 1: Gate input of Counter 1.</td></tr>
<tr><td>CLK 1</td><td>15</td><td>18</td><td>I</td><td>Clock 1: Clock input of Counter 1.</td></tr>
<tr><td>GATE 2</td><td>16</td><td>19</td><td>I</td><td>Gate 2: Gate input of Counter 2.</td></tr>
<tr><td>OUT 2</td><td>17</td><td>20</td><td>O</td><td>Out 2: Output of Counter 2.</td></tr>
<tr><td>CLK 2</td><td>18</td><td>21</td><td>I</td><td>Clock 2: Clock input of Counter 2.</td></tr>
<tr><td>A_1, A_0</td><td>20-19</td><td>23-22</td><td>I</td><td>Address: Used to select one of the three Counters or the Control Word Register for read or write operations. Normally connected to the system address bus.
<table>
<tr><th>A_1</th><th>A_0</th><th>Selects</th></tr>
<tr><td>0</td><td>0</td><td>Counter 0</td></tr>
<tr><td>0</td><td>1</td><td>Counter 1</td></tr>
<tr><td>1</td><td>0</td><td>Counter 2</td></tr>
<tr><td>1</td><td>1</td><td>Control Word Register</td></tr>
</table></td></tr>
<tr><td>$\overline{CS}$</td><td>21</td><td>24</td><td>I</td><td>Chip Select: A low on this input enables the 82C54 to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and $\overline{WR}$ are ignored otherwise.</td></tr>
<tr><td>$\overline{RD}$</td><td>22</td><td>26</td><td>I</td><td>Read Control: This input is low during CPU read operations.</td></tr>
<tr><td>$\overline{WR}$</td><td>23</td><td>27</td><td>I</td><td>Write Control: This input is low during CPU write operations.</td></tr>
<tr><td>V_{CC}</td><td>24</td><td>28</td><td></td><td>Power: +5V power supply connection.</td></tr>
<tr><td>NC</td><td></td><td>1, 11, 15, 25</td><td></td><td>No Connect</td></tr>
</table>

FUNCTIONAL DESCRIPTION

General

The 82C54 is a programmable interval timer/counter designed for use with Intel microcomputer systems. It is a general purpose, multi-timing element that can be treated as an array of I/O ports in the system software.

The 82C54 solves one of the most common problems in any microcomputer system, the generation of accurate time delays under software control. Instead of setting up timing loops in software, the programmer configures the 82C54 to match his requirements and programs one of the counters for the desired delay. After the desired delay, the 82C54 will interrupt the CPU. Software overhead is minimal and variable length delays can easily be accommodated.

Some of the other counter/timer functions common to microcomputers which can be implemented with the 82C54 are:

- Real time clock
- Even counter
- Digital one-shot
- Programmable rate generator
- Square wave generator
- Binary rate multiplier
- Complex waveform generator
- Complex motor controller

Block Diagram

DATA BUS BUFFER

This 3-state, bi-directional, 8-bit buffer is used to interface the 82C54 to the system bus (see Figure 3).

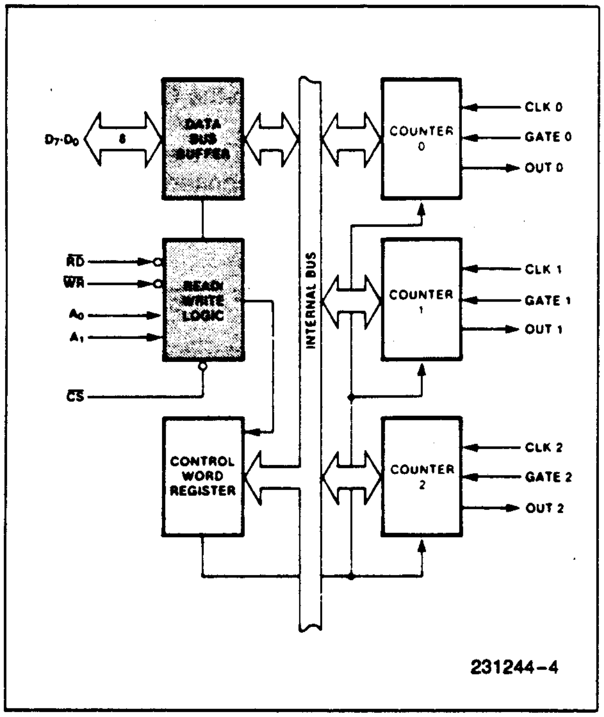


Figure 3. Block Diagram Showing Data Bus Buffer and Read/Write Logic Functions

READ/WRITE LOGIC

The Read/Write Logic accepts inputs from the system bus and generates control signals for the other functional blocks of the 82C54. A_1 and A_0 select one of the three counters or the Control Word Register to be read from/written into. A "low" on the $\overline{RD}$ input tells the 82C54 that the CPU is reading one of the counters. A "low" on the $\overline{WR}$ input tells the 82C54 that the CPU is writing either a Control Word or an initial count. Both $\overline{RD}$ and $\overline{WR}$ are qualified by $\overline{CS}$; $\overline{RD}$ and $\overline{WR}$ are ignored unless the 82C54 has been selected by holding $\overline{CS}$ low.

CONTROL WORD REGISTER

The Control Word Register (see Figure 4) is selected by the Read/Write Logic when A_1, A_0 = 11. If the CPU then does a write operation to the 82C54, the data is stored in the Control Word Register and is interpreted as a Control Word used to define the operation of the Counters.

The Control Word Register can only be written to; status information is available with the Read-Back Command.

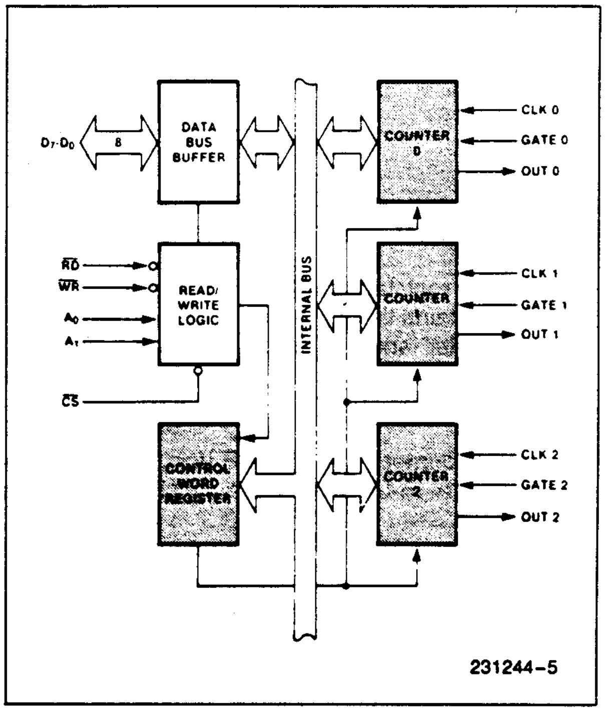


Figure 4. Block Diagram Showing Control Word Register and Counter Functions

COUNTER 0, COUNTER 1, COUNTER 2

These three functional blocks are identical in operation, so only a single Counter will be described. The internal block diagram of a single counter is shown in Figure 5.

The Counters are fully independent. Each Counter may operate in a different Mode.

The Control Word Register is shown in the figure; it is not part of the Counter itself, but its contents determine how the Counter operates.

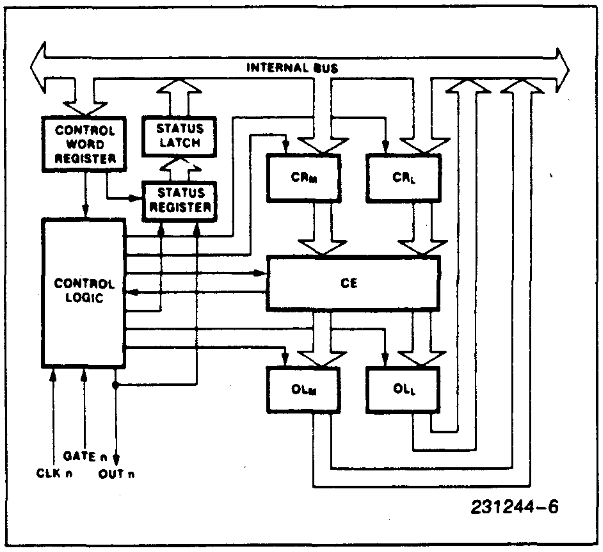


Figure 5. Internal Block Diagram of a Counter

The status register, shown in the Figure, when latched, contains the current contents of the Control Word Register and status of the output and null count flag. (See detailed explanation of the Read-Back command.)

The actual counter is labelled CE (for "Counting Element"). It is a 16-bit presettable synchronous down counter.

OL_M and OL_L are two 8-bit latches. OL stands for "Output Latch"; the subscripts M and L stand for "Most significant byte" and "Least significant byte" respectively. Both are normally referred to as one unit and called just OL. These latches normally "follow" the CE, but if a suitable Counter Latch Command is sent to the 82C54, the latches "latch" the present count until read by the CPU and then return to "following" the CE. One latch at a time is enabled by the counter's Control Logic to drive the internal bus. This is how the 16-bit Counter communicates over the 8-bit internal bus. Note that the CE itself cannot be read; whenever you read the count, it is the OL that is being read.

Similarly, there are two 8-bit registers called CR_M and CR_L (for "Count Register"). Both are normally referred to as one unit and called just CR. When a new count is written to the Counter, the count is stored in the CR and later transferred to the CE. The Control Logic allows one register at a time to be loaded from the internal bus. Both bytes are transferred to the CE simultaneously. CR_M and CR_L are cleared when the Counter is programmed. In this way, if the Counter has been programmed for one byte counts (either most significant byte only or least significant byte only) the other byte will be zero. Note that the CE cannot be written into; whenever a count is written, it is written into the CR.

The Control Logic is also shown in the diagram. CLK n, GATE n, and OUT n are all connected to the outside world through the Control Logic.

82C54 SYSTEM INTERFACE

The 82C54 is treated by the systems software as an array of peripheral I/O ports; three are counters and the fourth is a control register for MODE programming.

Basically, the select inputs A_0, A_1 connect to the A_0, A_1 address bus signals of the CPU. The $\overline{CS}$ can be derived directly from the address bus using a linear select method. Or it can be connected to the output of a decoder, such as an Intel 8205 for larger systems.

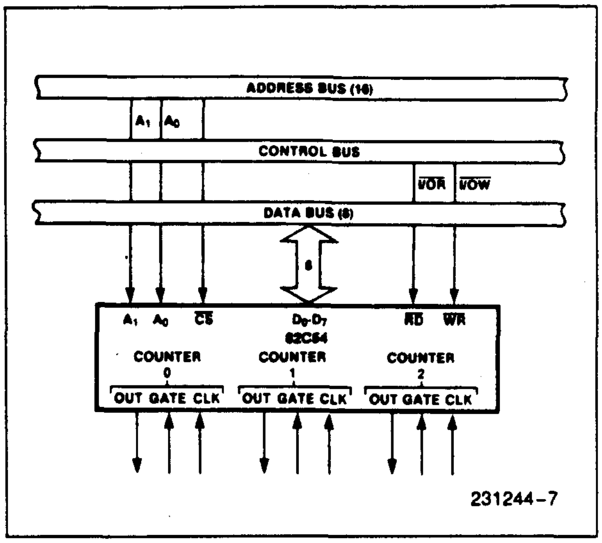


Figure 6. 82C54 System Interface

OPERATIONAL DESCRIPTION

General

After power-up, the state of the 82C54 is undefined. The Mode, count value, and output of all Counters are undefined.

How each Counter operates is determined when it is programmed. Each Counter must be programmed before it can be used. Unused counters need not be programmed.

Programming the 82C54

Counters are programmed by writing a Control Word and then an initial count. The control word format is shown in Figure 7.

All Control Words are written into the Control Word Register, which is selected when A_1, A_0 = 11. The Control Word itself specifies which Counter is being programmed.

By contrast, initial counts are written into the Counters, not the Control Word Register. The A_1, A_0 inputs are used to select the Counter to be written into. The format of the initial count is determined by the Control Word used.

Control Word Format

$A_1, A_0 = 11 \quad \overline{CS} = 0 \quad \overline{RD} = 1 \quad \overline{WR} = 0$

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC1	SC0	RW1	RW0	M2	M1	M0	BCD

SC — Select Counter:

SC1	SC0	
0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	Read-Back Command (See Read Operations)

RW — Read/Write:

RW1	RW0	
0	0	Counter Latch Command (see Read Operations)
0	1	Read/Write least significant byte only.
1	0	Read/Write most significant byte only.
1	1	Read/Write least significant byte first, then most significant byte.

M — MODE:

M2	M1	M0	
0	0	0	Mode 0
0	0	1	Mode 1
X	1	0	Mode 2
X	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

BCD:

0	Binary Counter 16-bits
1	Binary Coded Decimal (BCD) Counter (4 Decades)

NOTE: Don't care bits (X) should be 0 to insure compatibility with future Intel products.

Figure 7. Control Word Format

Write Operations

The programming procedure for the 82C54 is very flexible. Only two conventions need to be remembered:

1) For each Counter, the Control Word must be written before the initial count is written.
2) The initial count must follow the count format specified in the Control Word (least significant byte only, most significant byte only, or least significant byte and then most significant byte).

Since the Control Word Register and the three Counters have separate addresses (selected by the A_1, A_0 inputs), and each Control Word specifies the Counter it applies to (SC0, SC1 bits), no special instruction sequence is required. Any programming sequence that follows the conventions above is acceptable.

A new initial count may be written to a Counter at any time without affecting the Counter's programmed Mode in any way. Counting will be affected as described in the Mode definitions. The new count must follow the programmed count format.

If a Counter is programmed to read/write two-byte counts, the following precaution applies: A program must not transfer control between writing the first and second byte to another routine which also writes into that same Counter. Otherwise, the Counter will be loaded with an incorrect count.

		A_1	A_0
Control Word —	Counter 0	1	1
LSB of count —	Counter 0	0	0
MSB of count —	Counter 0	0	0
Control Word —	Counter 1	1	1
LSB of count —	Counter 1	0	1
MSB of count —	Counter 1	0	1
Control Word —	Counter 2	1	1
LSB of count —	Counter 2	1	0
MSB of count —	Counter 2	1	0

		A_1	A_0
Control Word —	Counter 2	1	1
Control Word —	Counter 1	1	1
Control Word —	Counter 0	1	1
LSB of count —	Counter 2	1	0
MSB of count —	Counter 2	1	0
LSB of count —	Counter 1	0	1
MSB of count —	Counter 1	0	1
LSB of count —	Counter 0	0	0
MSB of count —	Counter 0	0	0

		A_1	A_0
Control Word —	Counter 0	1	1
Counter Word —	Counter 1	1	1
Control Word —	Counter 2	1	1
LSB of count —	Counter 2	1	0
LSB of count —	Counter 1	0	1
LSB of count —	Counter 0	0	0
MSB of count —	Counter 0	0	0
MSB of count —	Counter 1	0	1
MSB of count —	Counter 2	1	0

		A_1	A_0
Control Word —	Counter 1	1	1
Control Word —	Counter 0	1	1
LSB of count —	Counter 1	0	1
Control Word —	Counter 2	1	1
LSB of count —	Counter 0	0	0
MSB of count —	Counter 1	0	1
LSB of count —	Counter 2	1	0
MSB of count —	Counter 0	0	0
MSB of count —	Counter 2	1	0

NOTE:
In all four examples, all counters are programmed to read/write two-byte counts.
These are only four of many possible programming sequences.

Figure 8. A Few Possible Programming Sequences

Read Operations

It is often desirable to read the value of a Counter without disturbing the count in progress. This is easily done in the 82C54.

There are three possible methods for reading the counters: a simple read operation, the Counter Latch Command, and the Read-Back Command. Each is explained below. The first method is to perform a simple read operation. To read the Counter, which is selected with the A1, A0 inputs, the CLK input of the selected Counter must be inhibited by using either the GATE input or external logic. Otherwise, the count may be in the process of changing when it is read, giving an undefined result.

COUNTER LATCH COMMAND

The second method uses the "Counter Latch Command". Like a Control Word, this command is written to the Control Word Register, which is selected when A_1, A_0 = 11. Also like a Control Word, the SC0, SC1 bits select one of the three Counters, but two other bits, D5 and D4, distinguish this command from a Control Word.

A_1, A_0 = 11; $\overline{CS}$ = 0; $\overline{RD}$ = 1; $\overline{WR}$ = 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
SC1	SC0	0	0	X	X	X	X

SC1, SC0 - specify counter to be latched

SC1	SC0	Counter
0	0	0
0	1	1
1	0	2
1	1	Read-Back Command

D5,D4 - 00 designates Counter Latch Command

X - don't care

NOTE:
Don't care bits (X) should be 0 to insure compatibility with future Intel products.

Figure 9. Counter Latching Command Format

The selected Counter's output latch (OL) latches the count at the time the Counter Latch Command is received. This count is held in the latch until it is read by the CPU (or until the Counter is reprogrammed). The count is then unlatched automatically and the OL returns to "following" the counting element (CE). This allows reading the contents of the Counters "on the fly" without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one Counter. Each latched Counter's OL holds its count until it is read. Counter Latch Commands do not affect the programmed Mode of the Counter in any way.

If a Counter is latched and then, some time later, latched again before the count is read, the second Counter Latch Command is ignored. The count read will be the count at the time the first Counter Latch Command was issued.

With either method, the count must be read according to the programmed format; specifically, if the Counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other; read or write or programming operations of other Counters may be inserted between them.

Another feature of the 82C54 is that reads and writes of the same Counter may be interleaved; for example, if the Counter is programmed for two byte counts, the following sequence is valid.

1. Read least significant byte.
2. Write new least significant byte.
3. Read most significant byte.
4. Write new most significant byte.

If a Counter is programmed to read/write two-byte counts, the following precaution applies; A program must not transfer control between reading the first and second byte to another routine which also reads from that same Counter. Otherwise, an incorrect count will be read.

READ-BACK COMMAND

The third method uses the Read-Back command. This command allows the user to check the count value, programmed Mode, and current state of the OUT pin and Null Count flag of the selected counter(s).

The command is written into the Control Word Register and has the format shown in Figure 10. The command applies to the counters selected by setting their corresponding bits D3,D2,D1 = 1.

A0, A1 = 11 $\overline{CS}$ = 0 $\overline{RD}$ = 1 $\overline{WR}$ = 0

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	1	$\overline{\text{COUNT}}$	$\overline{\text{STATUS}}$	CNT 2	CNT 1	CNT 0	0

D_5: 0 = Latch count of selected counter(s)
D_4: 0 = Latch status of selected counter(s)
D_3: 1 = Select counter 2
D_2: 1 = Select counter 1
D_1: 1 = Select counter 0
D_0: Reserved for future expansion; must be 0

Figure 10. Read-Back Command Format

The read-back command may be used to latch multiple counter output latches (OL) by setting the $\overline{\text{COUNT}}$ bit D5 = 0 and selecting the desired counter(s). This single command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read (or the counter is reprogrammed). That counter is automatically unlatched when read, but other counters remain latched until they are read. If multiple count read-back commands are issued to the same counter without reading the count, all but the first are ignored; i.e., the count which will be read is the count at the time the first read-back command was issued.

The read-back command may also be used to latch status information of selected counter(s) by setting $\overline{\text{STATUS}}$ bit D4 = 0. Status must be latched to be read; status of a counter is accessed by a read from that counter.

The counter status format is shown in Figure 11. Bits D5 through D0 contain the counter's programmed Mode exactly as written in the last Mode Control Word. OUTPUT bit D7 contains the current state of the OUT pin. This allows the user to monitor the counter's output via software, possibly eliminating some hardware from a system.

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
OUTPUT	NULL COUNT	RW1	RW0	M2	M1	M0	BCD

D_7 1 = Out Pin is 1
0 = Out Pin is 0
D_6 1 = Null count
0 = Count available for reading
D_5-D_0 Counter Programmed Mode (See Figure 7)

Figure 11. Status Byte

NULL COUNT bit D6 indicates when the last count written to the counter register (CR) has been loaded into the counting element (CE). The exact time this happens depends on the Mode of the counter and is described in the Mode Definitions, but until the count is loaded into the counting element (CE), it can't be read from the counter. If the count is latched or read before this time, the count value will not reflect the new count just written. The operation of Null Count is shown in Figure 12.

THIS ACTION:	CAUSES:
A. Write to the control word register:[1]	Null count = 1
B. Write to the count register (CR);[2]	Null count = 1
C. New count is loaded into CE (CR → CE);	Null count = 0

[1] Only the counter specified by the control word will have its null count set to 1. Null count bits of other counters are unaffected.

[2] If the counter is programmed for two-byte counts (least significant byte then most significant byte) null count goes to 1 when the second byte is written.

Figure 12. Null Count Operation

If multiple status latch operations of the counter(s) are performed without reading the status, all but the first are ignored; i.e., the status that will be read is the status of the counter at the time the first status read-back command was issued.

Both count and status of the selected counter(s) may be latched simultaneously by setting both $\overline{\text{COUNT}}$ and $\overline{\text{STATUS}}$ bits D5,D4 = 0. This is functionally the same as issuing two separate read-back commands at once, and the above discussions apply here also. Specifically, if multiple count and/or status read-back commands are issued to the same counter(s) without any intervening reads, all but the first are ignored. This is illustrated in Figure 13.

If both count and status of a counter are latched, the first read operation of that counter will return latched status, regardless of which was latched first. The next one or two reads (depending on whether the counter is programmed for one or two type counts) return latched count. Subsequent reads return unlatched count.

Command								Description	Results
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0		
1	1	0	0	0	0	1	0	Read back count and status of Counter 0	Count and status latched for Counter 0
1	1	1	0	0	1	0	0	Read back status of Counter 1	Status latched for Counter 1
1	1	1	0	1	1	0	0	Read back status of Counters 2, 1	Status latched for Counter 2, but not Counter 1
1	1	0	1	1	0	0	0	Read back count of Counter 2	Count latched for Counter 2
1	1	0	0	0	1	0	0	Read back count and status of Counter 1	Count latched for Counter 1, but not status
1	1	1	0	0	0	1	0	Read back status of Counter 1	Command ignored, status already latched for Counter 1

Figure 13. Read-Back Command Example

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A_1	A_0	
0	1	0	0	0	Write into Counter 0
0	1	0	0	1	Write into Counter 1
0	1	0	1	0	Write into Counter 2
0	1	0	1	1	Write Control Word
0	0	1	0	0	Read from Counter 0
0	0	1	0	1	Read from Counter 1
0	0	1	1	0	Read from Counter 2
0	0	1	1	1	No-Operation (3-State)
1	X	X	X	X	No-Operation (3-State)
0	1	1	X	X	No-Operation (3-State)

Figure 14. Read/Write Operations Summary

Mode Definitions

The following are defined for use in describing the operation of the 82C54.

CLK PULSE: a rising edge, then a falling edge, in that order, of a Counter's CLK input.

TRIGGER: a rising edge of a Counter's GATE input.

COUNTER LOADING: the transfer of a count from the CR to the CE (refer to the "Functional Description")

MODE 0: INTERRUPT ON TERMINAL COUNT

Mode 0 is typically used for event counting. After the Control Word is written, OUT is initially low, and will remain low until the Counter reaches zero. OUT then goes high and remains high until a new count or a new Mode 0 Control Word is written into the Counter.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

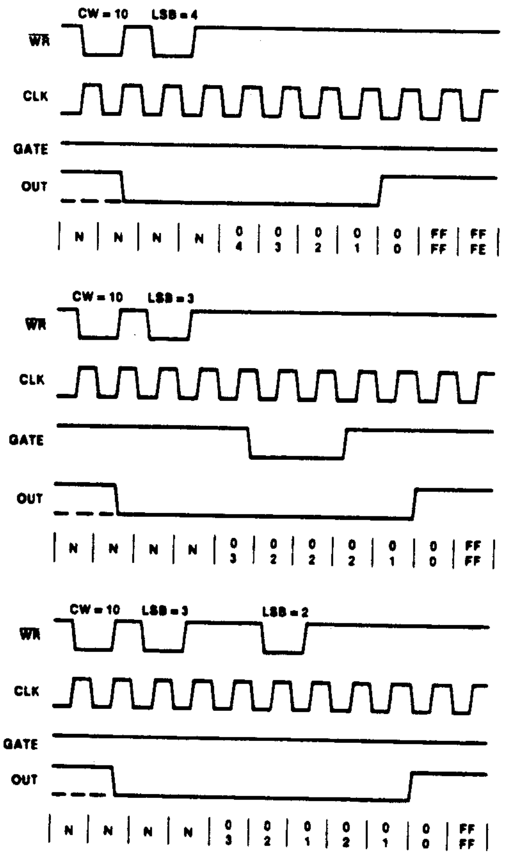
After the Control Word and initial count are written to a Counter, the initial count will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not go high until N + 1 CLK pulses after the initial count is written.

If a new count is written to the Counter, it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

1) Writing the first byte disables counting. OUT is set low immediately (no clock pulse required).
2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the counting sequence to be synchronized by software. Again, OUT does not go high until N + 1 CLK pulses after the new count of N is written.

If an initial count is written while GATE = 0, it will still be loaded on the next CLK pulse. When GATE goes high, OUT will go high N CLK pulses later; no CLK pulse is needed to load the Counter as this has already been done.



231244-8

NOTE:
The Following Conventions Apply To All Mode Timing Diagrams:
1. Counters are programmed for binary (not BCD) counting and for Reading/Writing least significant byte (LSB) only.
2. The counter is always selected ($\overline{CS}$ always low).
3. CW stands for "Control Word"; CW = 10 means a control word of 10, hex is written to the counter.
4. LSB stands for "Least Significant Byte" of count.
5. Numbers below diagrams are count values.
The lower number is the least significant byte.
The upper number is the most significant byte. Since the counter is programmed to Read/Write LSB only, the most significant byte cannot be read.
N stands for an undefined count.
Vertical lines show transitions between count values.

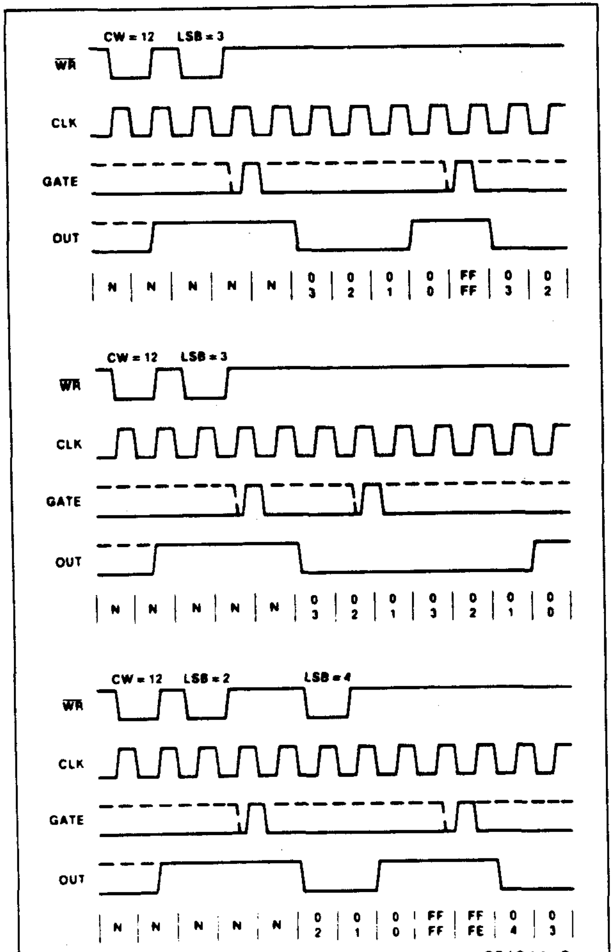
Figure 15. Mode 0

MODE 1: HARDWARE RETRIGGERABLE ONE-SHOT

OUT will be initially high. OUT will go low on the CLK pulse following a trigger to begin the one-shot pulse, and will remain low until the Counter reaches zero. OUT will then go high and remain high until the CLK pulse after the next trigger.

After writing the Control Word and initial count, the Counter is armed. A trigger results in loading the Counter and setting OUT low on the next CLK pulse, thus starting the one-shot pulse. An initial count of N will result in a one-shot pulse N CLK cycles in duration. The one-shot is retriggerable, hence OUT will remain low for N CLK pulses after any trigger. The one-shot pulse can be repeated without rewriting the same count into the counter. GATE has no effect on OUT.

If a new count is written to the Counter during a one-shot pulse, the current one-shot is not affected unless the Counter is retriggered. In that case, the Counter is loaded with the new count and the one-shot pulse continues until the new count expires.


CW = 12 LSB = 3
WR
CLK
GATE
OUT
N N N N N 0/3 0/2 0/1 0/0 FF/FF 0/3 0/2
CW = 12 LSB = 3
WR
CLK
GATE
OUT
N N N N N 0/3 0/2 0/1 0/3 0/2 0/1 0/0
CW = 12 LSB = 2 LSB = 4
WR
CLK
GATE
OUT
N N N N N 0/2 0/1 0/0 FF/FF FF/FE 0/4 0/3

231244–9

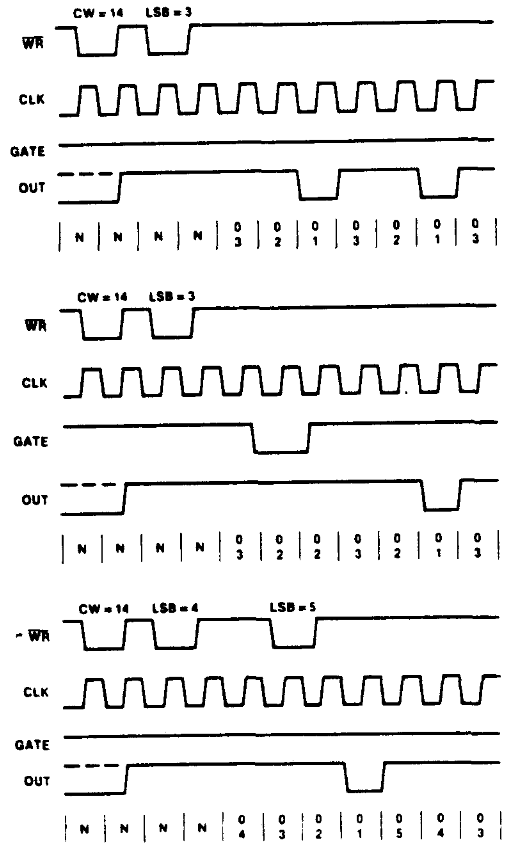
Figure 16. Mode 1

MODE 2: RATE GENERATOR

This Mode functions like a divide-by-N counter. It is typicially used to generate a Real Time Clock interrupt. OUT will initially be high. When the initial count has decremented to 1, OUT goes low for one CLK pulse. OUT then goes high again, the Counter reloads the initial count and the process is repeated. Mode 2 is periodic; the same sequence is repeated indefinitely. For an initial count of N, the sequence repeats every N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low during an output pulse, OUT is set high immediately. A trigger reloads the Counter with the initial count on the next CLK pulse; OUT goes low N CLK pulses after the trigger. Thus the GATE input can be used to synchronize the Counter.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. OUT goes low N CLK Pulses after the initial count is written. This allows the Counter to be synchronized by software also.


CW = 14 LSB = 3
WR
CLK
GATE
OUT
N N N N 0/3 0/2 0/1 0/3 0/2 0/1 0/3
CW = 14 LSB = 3
WR
CLK
GATE
OUT
N N N N 0/3 0/2 0/2 0/3 0/2 0/1 0/3
CW = 14 LSB = 4 LSB = 5
WR
CLK
GATE
OUT
N N N N 0/4 0/3 0/2 0/1 0/5 0/4 0/3

231244–10

NOTE:
A GATE transition should not occur one clock prior to terminal count.

Figure 17. Mode 2

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current period, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the new count. Otherwise, the new count will be loaded at the end of the current counting cycle. In mode 2, a COUNT of 1 is illegal.

MODE 3: SQUARE WAVE MODE

Mode 3 is typically used for Baud rate generation. Mode 3 is similar to Mode 2 except for the duty cycle of OUT. OUT will initially be high. When half the initial count has expired, OUT goes low for the remainder of the count. Mode 3 is periodic; the sequence above is repeated indefinitely. An initial count of N results in a square wave with a period of N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low while OUT is low, OUT is set high immediately; no CLK pulse is required. A trigger reloads the Counter with the initial count on the next CLK pulse. Thus the GATE input can be used to synchronize the Counter.

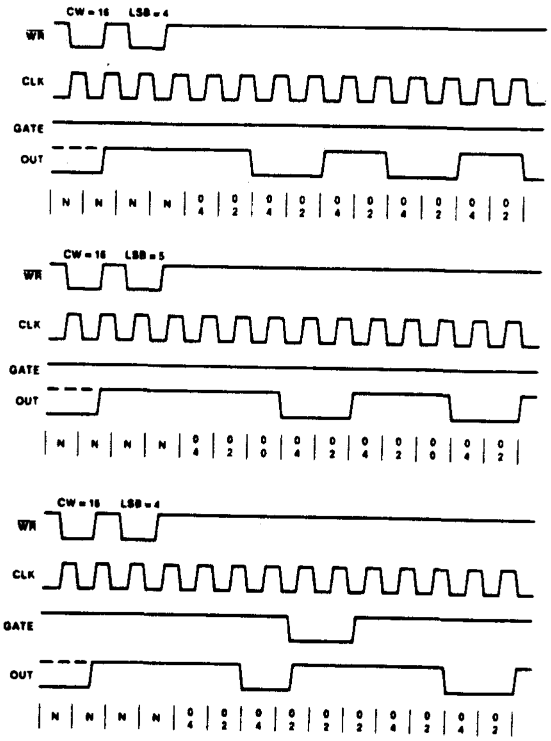
After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This allows the Counter to be synchronized by software also.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current half-cycle of the square wave, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the new count. Otherwise, the new count will be loaded at the end of the current half-cycle.

Mode 3 is implemented as follows:

Even counts: OUT is initially high. The initial count is loaded on one CLK pulse and then is decremented by two on succeeding CLK pulses. When the count expires OUT changes value and the Counter is reloaded with the initial count. The above process is repeated indefinitely.

Odd counts: OUT is initially high. The initial count minus one (an even number) is loaded on one CLK pulse and then is decremented by two on succeeding CLK pulses. One CLK pulse *after* the count expires, OUT goes low and the Counter is reloaded with the initial count minus one. Succeeding CLK pulses decrement the count by two. When the count expires, OUT goes high again and the Counter is reloaded with the initial count minus one. The above process is repeated indefinitely. So for odd counts, OUT will be high for (N + 1)/2 counts and low for (N − 1)/2 counts.


CW = 16 LSB = 4
WR
CLK
GATE
OUT
N N N N 0 4 0 2 0 4 0 2 0 4 0 2 0 4 0 2 0 4 0 2
CW = 16 LSB = 5
WR
CLK
GATE
OUT
N N N N 0 4 0 2 0 0 0 4 0 2 0 4 0 2 0 0 0 4 0 2
CW = 16 LSB = 4
WR
CLK
GATE
OUT
N N N N 0 4 0 2 0 4 0 2 0 2 0 2 0 4 0 2 0 4 0 2

231244-11

NOTE:
A GATE transition should not occur one clock prior to terminal count.

Figure 18. Mode 3

MODE 4: SOFTWARE TRIGGERED STROBE

OUT will be initially high. When the initial count expires, OUT will go low for one CLK pulse and then go high again. The counting sequence is "triggered" by writing the initial count.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

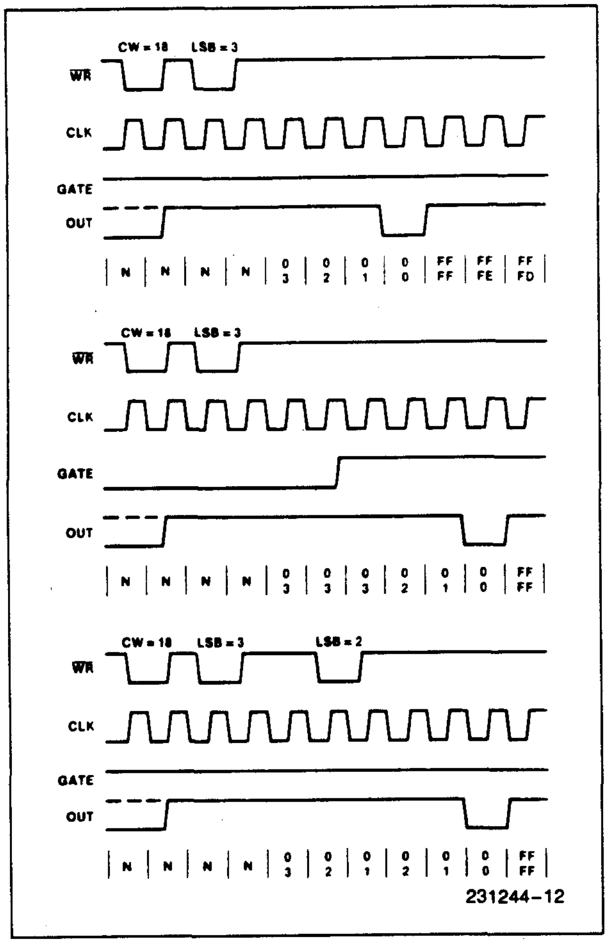
After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 CLK pulses after the initial count is written.

If a new count is written during counting, it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

1) Writing the first byte has no effect on counting.

2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the sequence to be "retriggered" by software. OUT strobes low N+1 CLK pulses after the new count of N is written.


CW=18 LSB=3
WR
CLK
GATE
OUT
N N N N 0 3 0 2 0 1 0 0 FF FF FF FE FF FD
CW=18 LSB=3
WR
CLK
GATE
OUT
N N N N 0 3 0 3 0 3 0 2 0 1 0 0 FF FF
CW=18 LSB=3 LSB=2
WR
CLK
GATE
OUT
N N N N 0 3 0 2 0 1 0 2 0 1 0 0 FF FF

231244-12

Figure 19. Mode 4

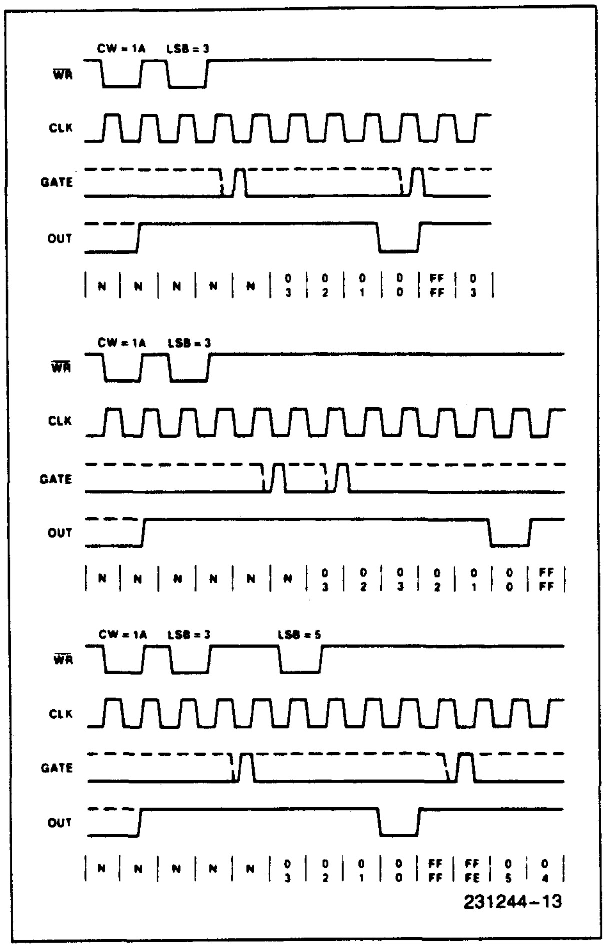
MODE 5: HARDWARE TRIGGERED STROBE (RETRIGGERABLE)

OUT will initially be high. Counting is triggered by a rising edge of GATE. When the initial count has expired, OUT will go low for one CLK pulse and then go high again.

After writing the Control Word and initial count, the counter will not be loaded until the CLK pulse after a trigger. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N+1 CLK pulses after a trigger.

A trigger results in the Counter being loaded with the initial count on the next CLK pulse. The counting sequence is retriggerable. OUT will not strobe low for N + 1 CLK pulses after any trigger. GATE has no effect on OUT.

If a new count is written during counting, the current counting sequence will not be affected. If a trigger occurs after the new count is written but before the current count expires, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from there.


CW=1A LSB=3
WR
CLK
GATE
OUT
N N N N N 0 3 0 2 0 1 0 0 FF FF 0 3
CW=1A LSB=3
WR
CLK
GATE
OUT
N N N N N N 0 3 0 2 0 3 0 2 0 1 0 0 FF FF
CW=1A LSB=3 LSB=5
WR
CLK
GATE
OUT
N N N N N 0 3 0 2 0 1 0 0 FF FF FF FE 0 5 0 4

231244-13

Figure 20. Mode 5

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

Figure 21. Gate Pin Operations Summary

MODE	MIN COUNT	MAX COUNT
0	1	0
1	1	0
2	2	0
3	2	0
4	1	0

NOTE:
0 is equivalent to 2^{16} for binary counting and 10^4 for BCD counting

Figure 22. Minimum and Maximum Initial Counts

Operation Common to All Modes

Programming

When a Control Word is written to a Counter, all Control Logic is immediately reset and OUT goes to a known initial state; no CLK pulses are required for this.

GATE

The GATE input is always sampled on the rising edge of CLK. In Modes 0, 2, 3, and 4 the GATE input is level sensitive, and the logic level is sampled on the rising edge of CLK. In Modes 1, 2, 3, and 5 the GATE input is rising-edge sensitive. In these Modes, a rising edge of GATE (trigger) sets an edge-sensitive flip-flop in the Counter. This flip-flop is then sampled on the next rising edge of CLK; the flip-flop is reset immediately after it is sampled. In this way, a trigger will be detected no matter when it occurs—a high logic level does not have to be maintained until the next rising edge of CLK. Note that in Modes 2 and 3, the GATE input is both edge- and level-sensitive. In Modes 2 and 3, if a CLK source other than the system clock is used, GATE should be pulsed immediately following $\overline{WR}$ of a new count value.

COUNTER

New counts are loaded and Counters are decremented on the falling edge of CLK.

The largest possible initial count is 0; this is equivalent to 2^{16} for binary counting and 10^4 for BCD counting.

The Counter does not stop when it reaches zero. In Modes 0, 1, 4, and 5 the Counter "wraps around" to the highest count, either FFFF hex for binary counting or 9999 for BCD counting, and continues counting. Modes 2 and 3 are periodic; the Counter reloads itself with the initial count and continues counting from there.

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias	0°C to 70°C
Storage Temperature	−65° to +150°C
Supply Voltage	−0.5 to +8.0V
Operating Voltage	+4V to +7V
Voltage on any Input	GND −2V to +6.5V
Voltage on any Output	GND−0.5V to V_{CC} + 0.5V
Power Dissipation	1 Watt

**Notice:* *Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D.C. CHARACTERISTICS

(T_A=0°C to 70°C, V_{CC}=5V± 10%, GND=0V) (T_A = −40°C to +85°C for Extended Temperature)

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	−0.5	0.8	V	
V_{IH}	Input High Voltage	2.0	V_{CC} + 0.5	V	
V_{OL}	Output Low Voltage		0.4	V	I_{OL} = 2.5 mA
V_{OH}	Output High Voltage	3.0 V_{CC} − 0.4		V V	I_{OH} = −2.5 mA I_{OH} = −100 μA
I_{IL}	Input Load Current		±2.0	μA	V_{IN}=V_{CC} to 0V
I_{OFL}	Output Float Leakage Current		±10	μA	V_{OUT}=V_{CC} to 0.0V
I_{CC}	V_{CC} Supply Current		20	mA	Clk Freq= 8MHz 82C54 10MHz 82C54-2
I_{CCSB}	V_{CC} Supply Current-Standby		10	μA	CLK Freq = DC $\overline{\text{CS}}$ = V_{CC}. All Inputs/Data Bus V_{CC} All Outputs Floating
I_{CCSB1}	V_{CC} Supply Current-Standby		150	μA	CLK Freq = DC $\overline{\text{CS}}$ = V_{CC}. All Other Inputs, I/O Pins = V_{GND}, Outputs Open
C_{IN}	Input Capacitance		10	pF	f_c = 1 MHz
$C_{I/O}$	I/O Capacitance		20	pF	Unmeasured pins returned to GND[5]
C_{OUT}	Output Capacitance		20	pF	

A.C. CHARACTERISTICS

(T_A = 0°C to 70°C, V_{CC} = 5V ±10%, GND =0V) (T_A = −40°C to +85°C for Extended Temperature)

BUS PARAMETERS (Note 1)

READ CYCLE

Symbol	Parameter	82C54		82C54-2		Units
		Min	Max	Min	Max	
t_{AR}	Address Stable Before $\overline{\text{RD}}$ ↓	45		30		ns
t_{SR}	$\overline{\text{CS}}$ Stable Before $\overline{\text{RD}}$ ↓	0		0		ns
t_{RA}	Address Hold Time After $\overline{\text{RD}}$ ↑	0		0		ns
t_{RR}	$\overline{\text{RD}}$ Pulse Width	150		95		ns
t_{RD}	Data Delay from $\overline{\text{RD}}$ ↓		120		85	ns
t_{AD}	Data Delay from Address		220		185	ns
t_{DF}	$\overline{\text{RD}}$ ↑ to Data Floating	5	90	5	65	ns
t_{RV}	Command Recovery Time	200		165		ns

NOTE:

1. AC timings measured at V_{OH} = 2.0V, V_{OL} = 0.8V.

A.C. CHARACTERISTICS (Continued)

WRITE CYCLE

Symbol	Parameter	82C54		82C54-2		Units
		Min	Max	Min	Max	
t_{AW}	Address Stable Before $\overline{WR}$ ↓	0		0		ns
t_{SW}	$\overline{CS}$ Stable Before $\overline{WR}$ ↓	0		0		ns
t_{WA}	Address Hold Time After $\overline{WR}$ ↑	0		0		ns
t_{WW}	$\overline{WR}$ Pulse Width	150		95		ns
t_{DW}	Data Setup Time Before $\overline{WR}$ ↑	120		95		ns
t_{WD}	Data Hold Time After $\overline{WR}$ ↑	0		0		ns
t_{RV}	Command Recovery Time	200		165		ns

CLOCK AND GATE

Symbol	Parameter	82C54		82C54-2		Units
		Min	Max	Min	Max	
t_{CLK}	Clock Period	125	DC	100	DC	ns
t_{PWH}	High Pulse Width	60(3)		30(3)		ns
t_{PWL}	Low Pulse Width	60(3)		50(3)		ns
T_R	Clock Rise Time		25		25	ns
t_F	Clock Fall Time		25		25	ns
t_{GW}	Gate Width High	50		50		ns
t_{GL}	Gate Width Low	50		50		ns
t_{GS}	Gate Setup Time to CLK ↑	50		40		ns
t_{GH}	Gate Hold Time After CLK ↑	50(2)		50(2)		ns
T_{OD}	Output Delay from CLK ↓		150		100	ns
t_{ODG}	Output Delay from Gate ↓		120		100	ns
t_{WC}	CLK Delay for Loading(4)	0	55	0	55	ns
t_{WG}	Gate Delay for Sampling(4)	−5	50	−5	40	ns
t_{WO}	OUT Delay from Mode Write		260		240	ns
t_{CL}	CLK Set Up for Count Latch	−40	45	−40	40	ns

NOTES:

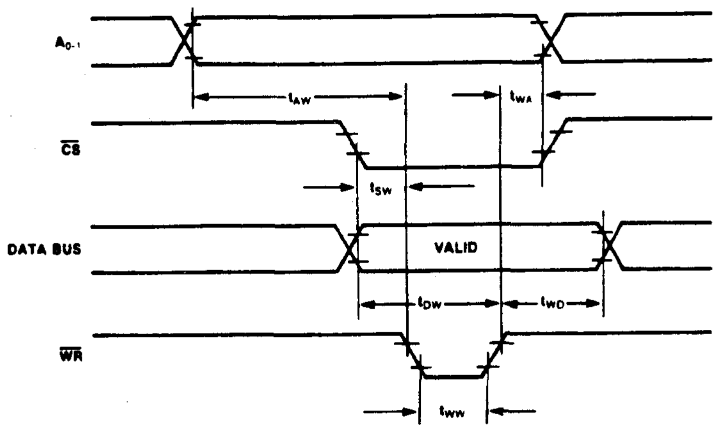
2. In Modes 1 and 5 triggers are sampled on each rising clock edge. A second trigger within 120 ns (70 ns for the 82C54-2) of the rising clock edge may not be detected.
3. Low-going glitches that violate t_{PWH}, t_{PWL} may cause errors requiring counter reprogramming.
4. Except for Extended Temp., See Extended Temp. A.C. Characteristics below.
5. Sampled not 100% tested. $T_A = 25°C$.
6. If CLK present at T_{WC} min then Count equals N+2 CLK pulses, T_{WC} max equals Count N+1 CLK pulse. T_{WC} min to T_{WC} max, count will be either N+1 or N+2 CLK pulses.
7. In Modes 1 and 5, if GATE is present when writing a new Count value, at T_{WG} min Counter will not be triggered, at T_{WG} max Counter will be triggered.
8. If CLK present when writing a Counter Latch or ReadBack Command, at T_{CL} min CLK will be reflected in count value latched, at T_{CL} max CLK will not be reflected in the count value latched. Writing a Counter Latch or ReadBack Command between T_{CL} min and T_{WL} max will result in a latched count vallue which is ± one least significant bit.

EXTENDED TEMPERATURE ($T_A = -40°C$ to $+85°C$ for Extended Temperature)

Symbol	Parameter	82C54		82C54-2		Units
		Min	Max	Min	Max	
t_{WC}	CLK Delay for Loading	−25	25	−25	25	ns
t_{WG}	Gate Delay for Sampling	−25	25	−25	25	ns

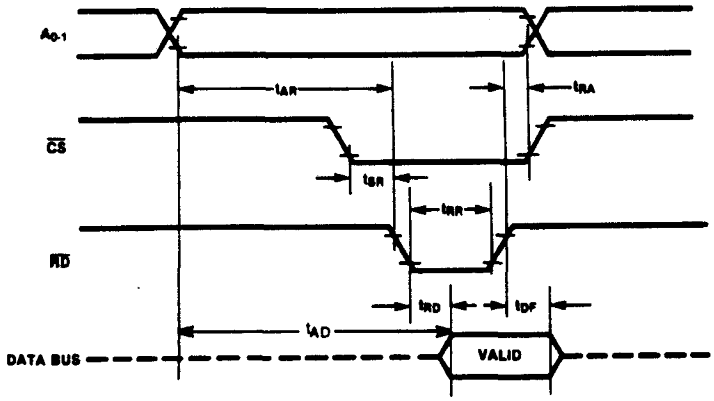
WAVEFORMS

WRITE



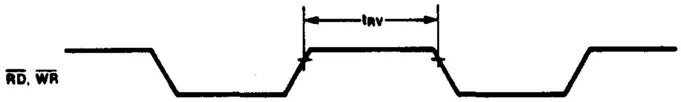
231244–14

READ



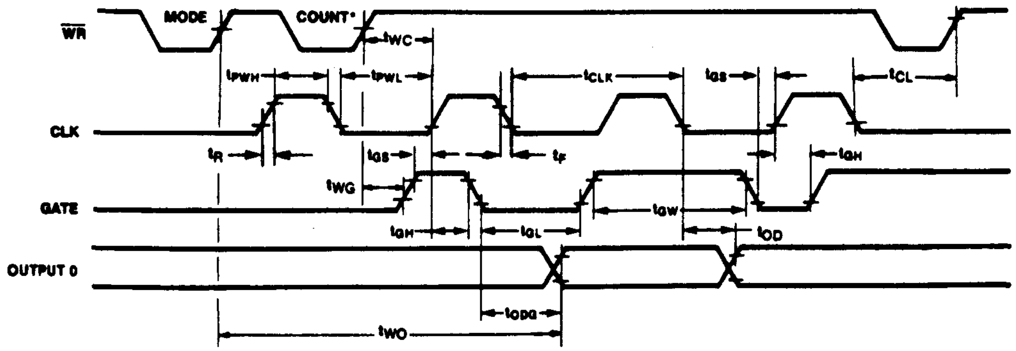
231244–15

RECOVERY



231244–16

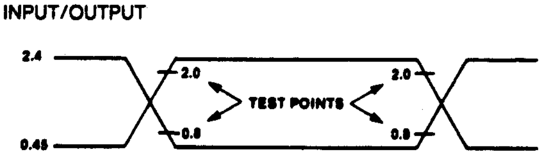
CLOCK AND GATE



231244-17

* Last byte of count being written

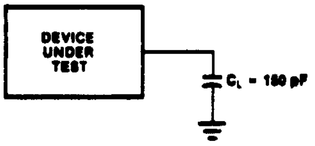
A.C. TESTING INPUT, OUTPUT WAVEFORM



231244-18

A.C. Testing: Inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0." Timing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0."

A.C. TESTING LOAD CIRCUIT



231244-19

C_L = 150 pF
C_L includes jig capacitance

APPENDIX D

CONFIGURING THE AD2000 FOR SIGNAL*MATH

Jumper Settings

When running SIGNAL*MATH, you have to change some of the AD2000's on-board jumpers from their factory-set positions. Before using SIGNAL*MATH on the AD2000 board, check the following jumpers:

- P2 — Base address
- P3 — 8254 timer/counter I/O configuration
- P4, P5 & P7 — Interrupts
- P6 — End-of-Convert Monitor

The board layout is shown in Figure D-1.

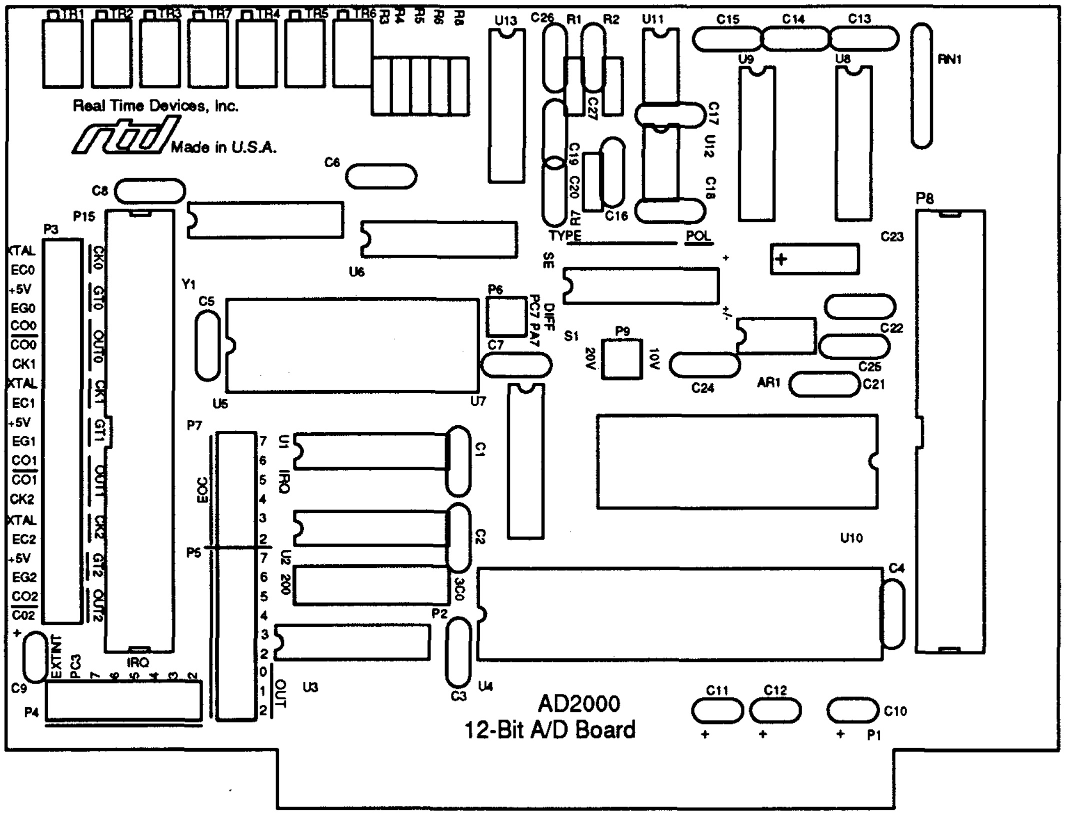


Fig. D-1 — AD2000 Board Layout

P2 — Base Address

SIGNAL*MATH assumes that the base address of your AD2000 is the factory setting of 300 hex (768 decimal). If you change this setting, you **must** run the ADAINST program and reset the base address.

NOTE: When using the ADAINST program, you can enter the base address in decimal or hexadecimal notation. When entering a hex value, you must precede the number by a dollar sign (for example, $300).

P3 — 8254 Timer/Counter I/O Configuration

The 8254 must be configured with the six jumpers placed between the pins as shown in Figure D-2. After setting the jumpers, verify that each is in the proper location. Any remaining jumpers must be removed from the P3 header connector.

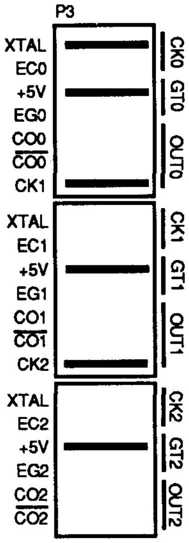


Fig. D-2 — 8254 Timer/Counter Jumpers, P3

P4, P5 & P7 — Interrupts

To select IRQ channels and interrupt sources for SIGNAL*MATH, you must install two jumpers on P5 and one jumper on P7. First install a jumper on P5-OUT2, and a second jumper across the pair of P5 pins for the IRQ channel you select.Then, install a jumper on the end-of-convert interrupt header, P7, across the pins of your desired IRQ channel. The IRQ selected on P7 **must be different from** the IRQ set on P5! Figure D-3 shows OUT2 jumpered to IRQ3 and EOC jumpered to IRQ4. Make sure that no jumpers are installed on P4.

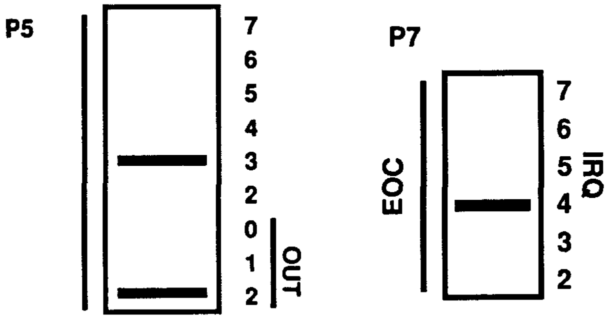


Fig. D-3 — Timer/Counter Out & End-of-Convert Interrupt Jumpers, P5 & P7

P6 — End-of-Convert Monitor

When running SIGNAL*MATH, place a jumper between EOC and PA7, as shown in Figure D-4.

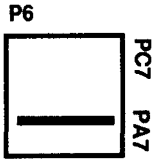


Fig. D-4 — End-of-Convert Monitor Jumper, P6

Running ADAINST

After the jumpers are set and the AD2000 board is installed in the computer, you are ready to configure SIGNAL*MATH so that it is compatible with your board's settings. This is done by running the ADAINST driver installation program. After running the program, open AD2000.EXE from the *Open a File* menu. You will see a screen similar to the screen shown in Figure D-5 below. The factory default settings are shown in the illustration. Your settings may or may not match the default settings, depending on whether you have made changes to these settings before.

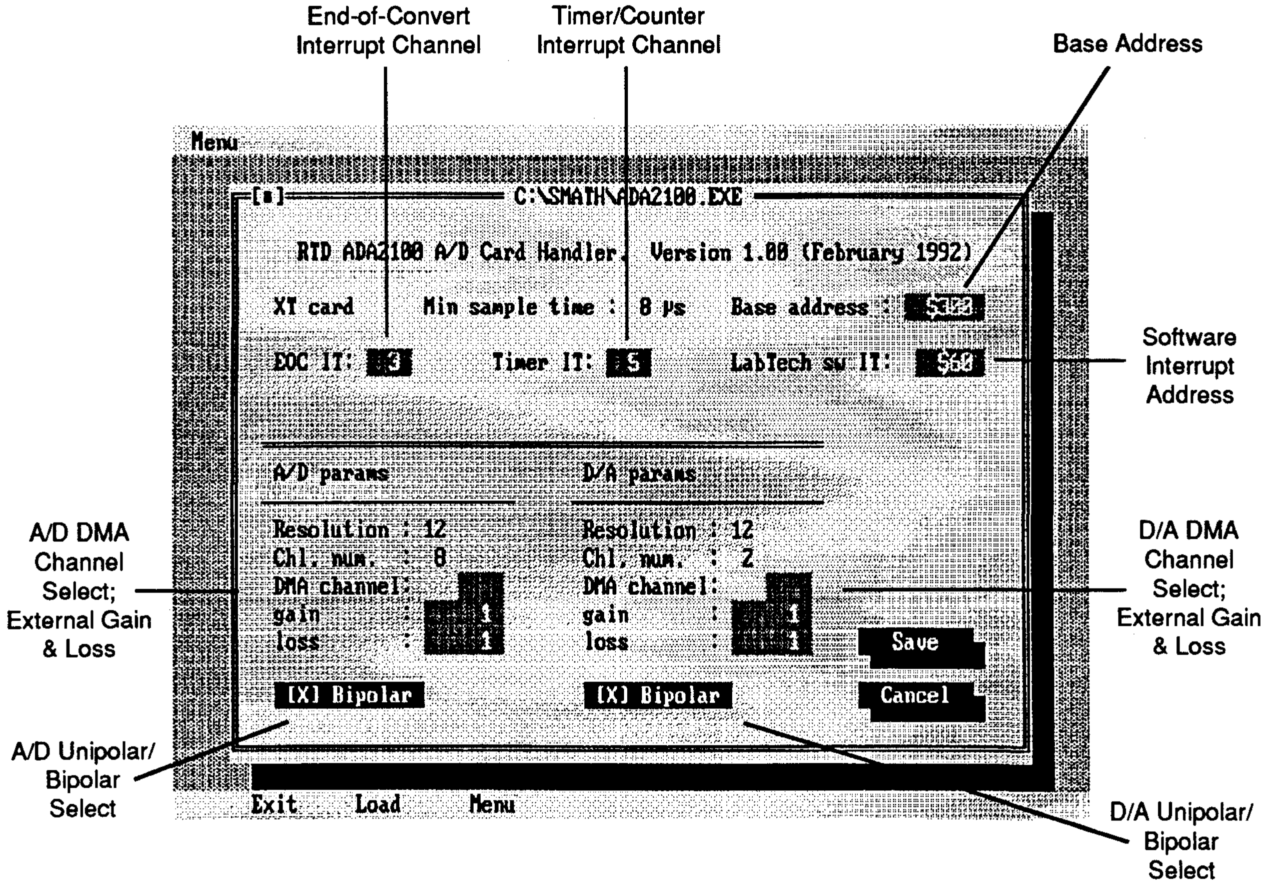
Base Address. The board's base address setting is entered in the upper right block, as shown in the diagram. The factory setting for all Real Time Devices boards is 300 hex (768 decimal). The base address can be entered as a decimal or hexadecimal value (hex values must be preceded by a dollar sign (for example, $300)). Refer to your board's manual if you need help in determining the correct value to enter.

EOC IT (End-of-Convert Interrupt). In this block, enter the IRQ channel number which corresponds to your jumper setting on P7.

Timer IT (Timer/Counter Interrupt). In this block, enter the IRQ channel number which corresponds to your jumper setting on P5.

LabTech SW IT (LABTECH NOTEBOOK Software Interrupt). This sets the software interrupt address where LABTECH NOTEBOOK's labLINX driver is installed. The factory setting is $60. This setting can be ignored when running SIGNAL*MATH.

A/D Parameters. Six A/D board parameters are listed: resolution, number of channels, active DMA channel, gain, loss, and input voltage polarity.



```
Menu
[■]──────────── C:\SMATH\ADA2100.EXE ──────────────

  RTD ADA2100 A/D Card Handler.  Version 1.00 (February 1992)

 XT card      Min sample time :  8 µs    Base address : $300

 EOC IT: 3         Timer IT: 5           LabTech sw IT:  $60

 A/D params                 D/A params

 Resolution : 12            Resolution : 12
 Chl. num.  :  8            Chl. num.  :  2
 DMA channel:               DMA channel:
 gain       :    1          gain       :    1
 loss       :    1          loss       :    1       Save

 [X] Bipolar                [X] Bipolar             Cancel

 Exit   Load    Menu
```

Fig. D-5 — ADAINST.EXE Screen

Resolution and number of channels are fixed by the program for your board.

The DMA channel number block is not valid on the AD2000, and should be left blank.

The next two blocks, gain and loss, are provided so that you can make adjustments for **external** gain or loss, **other than** the programmable gain settings available on the board. If your input signal is externally attenuated, then you can adjust for this by setting a value other than 1 for loss. If you have an external gain factor, then you can adjust for this condition. Numbers must be entered as whole decimal values. The factory default setting for gain and loss is 1.

For a bipolar input range, an X should be placed before Bipolar on the screen (default setting). For unipolar operation, remove the X.

D/A Parameters. These six blocks are not used on the AD2000, and should be left blank.

APPENDIX E

CONFIGURING THE AD2000 FOR ATLANTIS

Jumper Settings

When running ATLANTIS, you have to change some of the AD2000's on-board jumpers from their factory-set positions. Before using ATLANTIS on the AD2000 board, check the following jumpers:

- P2 — Base address
- P3 — 8254 timer/counter I/O configuration
- P4, P5 & P7 — Interrupts
- P6 — End-of-Convert Monitor

The board layout is shown in Figure E-1.

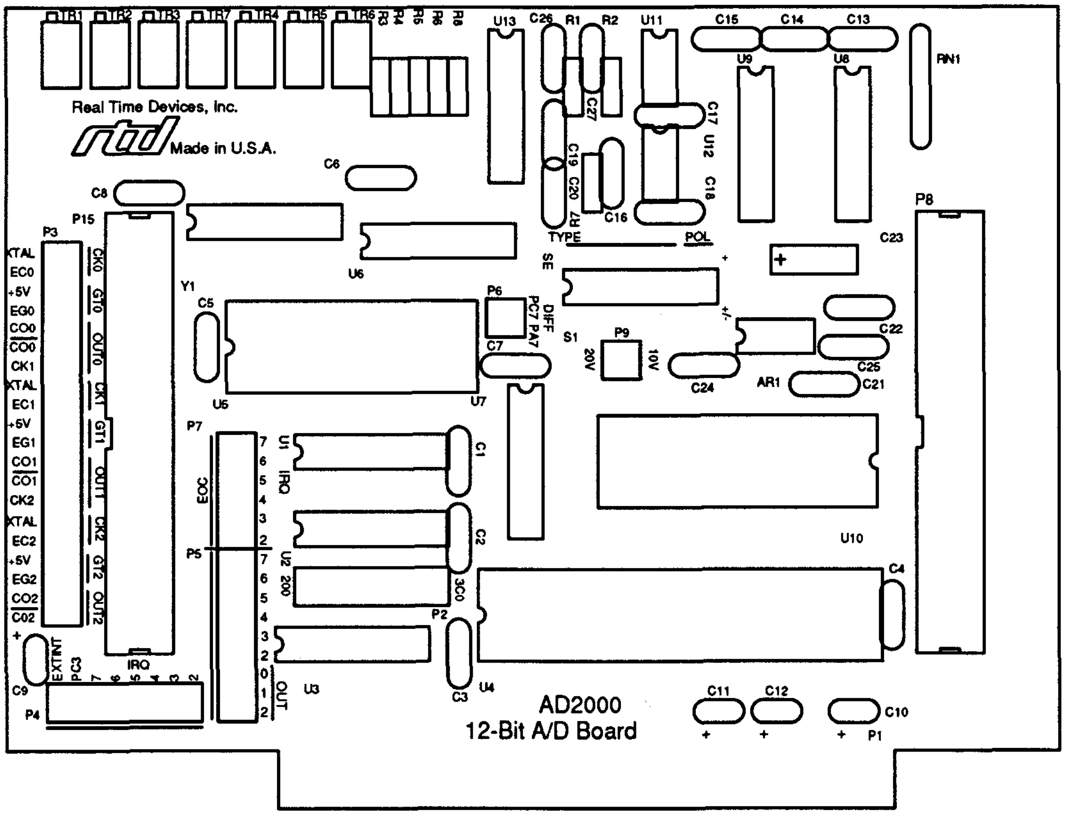


Fig. E-1 — AD2000 Board Layout

P2 — Base Address

ATLANTIS assumes that the base address of your AD2000 is the factory setting of 300 hex (see Chapter 1). If you changed this setting, you **must** run the ATINST program and reset the base address.

NOTE: The ATINST program requires the base address to be entered in **decimal** notation.

P3 — 8254 Timer/Counter I/O Configuration

The 8254 must be configured with the six jumpers placed between the pins as shown in Figure E-2. After setting the jumpers, verify that each is in the proper location. Any remaining jumpers must be removed from the P3 header connector.

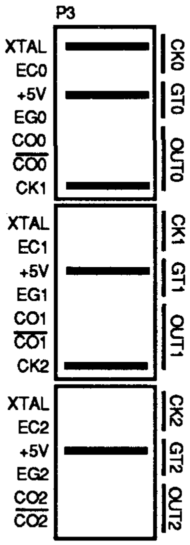


Fig. E-2 — 8254 Timer/Counter Jumpers, P3

P4, P5 & P7 — Interrupts

To select an IRQ channel and an interrupt source for ATLANTIS, you must install two jumpers on P5, the timer/counter output interrupt header. Jumpers must be installed across the OUT2 pins and across the pins of your desired IRQ channel. Figure E-3 shows OUT2 jumpered to IRQ3. Make sure that no jumpers are installed across the IRQ pins on header connectors P4 and P7.

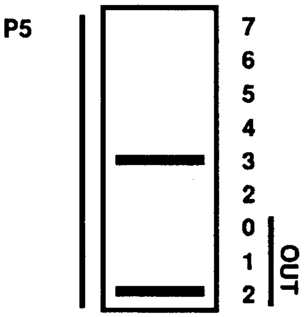


Fig. E-3 — End-of-Convert Interrupt Jumper, P7

P6 — End-of-Convert Monitor

When running ATLANTIS, place a jumper between EOC and PA7, as shown in Figure E-4.

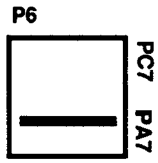


Fig. E-4 — End-of-Convert Monitor Jumper, P6

APPENDIX F

WARRANTY

LIMITED WARRANTY

Real Time Devices, Inc. warrants the hardware and software products it manufactures and produces to be free from defects in materials and workmanship for one year following the date of shipment from REAL TIME DEVICES. This warranty is limited to the original purchaser of product and is not transferable.

During the one year warranty period, REAL TIME DEVICES will repair or replace, at its option, any defective products or parts at no additional charge, provided that the product is returned, shipping prepaid, to REAL TIME DEVICES. All replaced parts and products become the property of REAL TIME DEVICES. **Before returning any product for repair, customers are required to contact the factory for an RMA number.**

THIS LIMITED WARRANTY DOES NOT EXTEND TO ANY PRODUCTS WHICH HAVE BEEN DAMAGED AS A RESULT OF ACCIDENT, MISUSE, ABUSE (such as: use of incorrect input voltages, improper or insufficient ventilation, failure to follow the operating instructions that are provided by REAL TIME DEVICES, "acts of God" or other contingencies beyond the control of REAL TIME DEVICES), OR AS A RESULT OF SERVICE OR MODIFICATION BY ANYONE OTHER THAN REAL TIME DEVICES. EXCEPT AS EXPRESSLY SET FORTH ABOVE, NO OTHER WARRANTIES ARE EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND REAL TIME DEVICES EXPRESSLY DISCLAIMS ALL WARRANTIES NOT STATED HEREIN. ALL IMPLIED WARRANTIES, INCLUDING IMPLIED WARRANTIES FOR MECHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED TO THE DURATION OF THIS WARRANTY. IN THE EVENT THE PRODUCT IS NOT FREE FROM DEFECTS AS WARRANTED ABOVE, THE PURCHASER'S SOLE REMEDY SHALL BE REPAIR OR REPLACEMENT AS PROVIDED ABOVE. UNDER NO CIRCUMSTANCES WILL REAL TIME DEVICES BE LIABLE TO THE PURCHASER OR ANY USER FOR ANY DAMAGES, INCLUDING ANY INCIDENTAL OR CONSEQUENTIAL DAMAGES, EXPENSES, LOST PROFITS, LOST SAVINGS, OR OTHER DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PRODUCT.

SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES FOR CONSUMER PRODUCTS, AND SOME STATES DO NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.

THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE.

AD2000 User-Selected Options	
Base I/O Address:	
(hex)	(decimal)
IRQ Channel Selection:	
A/D EOC	IRQ CHANNEL:
PIT OUT0	IRQ CHANNEL:
PIT OUT1	IRQ CHANNEL:
PIT OUT2	IRQ CHANNEL:
PPI INTRA (PC3)	IRQ CHANNEL:
EXTINT	IRQ CHANNEL:
A/D EOC/PPI Bit Assignment:	
A/D EOC	PA7: PC7: