NEAR EAST UNIVERSITY

Faculty of Engineering

DEPARTMENT OF COMPUTER ENGINEERING

PC14AT/LP EPROM PROGRAMMER

GRADUATION PROJECT
COM 400

Students: BARIŞ BİÇER
&
SELLAHATTİN GÜNER
Supervisor: Assoc.Prof.Dr.Doğan İBRAHİM

Nicosia - 2001

TABLE OF CONTENTS

ABSTRACT

EPROM memories are very widely used in microprocessor based equipment.These memories are normaly programmed using EPROM programmer devices here programming pulse are applied to the devices to program them. The EPROM to be programmed is connected to the PC via an input-output card and the devices is programmed from the PC using the Visual Basic programming language.

The project shows that highly sophisticated hadrware based projects can be developed using the Visual Basic programming language.

INTRODUCTION

Memories are an important part of any kind of microprocessor system.There are several types of memories: Random Access Memories (RAM) are used to store temporary data , such as variables in a program. These memories have the advantages that they are read-write type memories here data can be written to any location , or data from any location can be read. The disadvantage of RAM type memories is that they are volatile and the data is lost after the removal of power.

Read Only Memories (ROM) are normally programmed during the manufacturing process. Data is only read from these memories and the data is not lost after the removal of power. ROM memories are used to store the program codes.

Electrically Programmable Read Only Memories (EPROM) can be programmed using a suitable EPROM programming device. The device must be erased before programming and EPROM eraser devices are used for this purpose. The devices are programmed by sending the addresses and the data and the sending a programming pluse with a specified width. EPROM memories are used in microprocessor systems to store the program codes during the development process.

This project is about developing a Personal Computer (PC) based EPROM programming device. The project has both hardware and software components. A special digital input-output card is connected to the ISA bus of the PC. The input-output pins of this card are connected to the address , data and the program pins of a connector here the 2764 type EPROM memories are attached. A Visual Basic program is developed on the PC which reads the user data file in hexadecimal format and then programs the EPROM accordingly. The program can also verify the EPROM, display the contents of the EPROM and display the contents of the user file hexadecimal format.

Chapter 1 Describes Eprom ,Eeprom and Flash Eprom
Chapter 2 Describtes general eproms and pin codes.

Chapter 3 Describes Standart Programming Algorith ,DC Programming Characteristics and AC.Programming characteristics

Chapter 4 Describes Intelligent Programming Algoritm, DC Programming Characteristics and AC.Programming characteristics

Chapter 5 Describes ISA bus and pin configuration

Chapter 6 Describes Eraser Eprom how is make

Chapter 7 Describes Data coding and data coding tables.

Chapter 8 Describes how make the eprom 2764 program.

Chapter 9 Describe we use to control cart (PC14AT/LP)

CONCLUSION

The project has described the design of a PC based EPROM programmer which can be used to program the popular 2764 type 8K byte EPROM to the PC. A Visual Basic program was developed on the PC which read the user file and loaded the data in this file to the EPROM. The hardware was built and the programmer was tested successfully.

The programmer can be developed further by including the other popular EPROM memory devices , such as the 2732 , 27128 and the 27256. Another possible improvementto the project is to include multi-gang programming facility here more than one EPROM memory can be programmed at the same time. This will require the modification of both the software and the hardware.

1. EPROM SIZE RATING

EPROMs are rated in k-bite here k is equal to 1,024 and the EPROM number generally (but not always) reflects the size. But when we talk about memory size, we speak in terms of k-bytes. To change bits to bytes, simply divide the number of bits by 8 to get bytes.

A couple of examples will clear things up:

The 2716 EPROM number ends in 16 and thus is 16 k-bits in size or 16 * 1,024 or 16,384 bits. Now, 16,384 bits divided by 8 gives us 2,048 bytes or 2 k-bytes. Thus a 2716 is a 16 k-bit EPROM, but is most often expressed as being 2 k-bytes in size.

Some EPROMs are word wide or 16 bits (2 bytes) wide. These EPROMs are also rated in bits, such as 27C1024 ; a 1 M-bit (Mega-bit), 1,000 k-bits, 128 k-bytes, or 64 k-word EPROM. Such EPROMs come in 40 pin packages to allow for the extra pins needed.

1.1 WHAT IS A ROM

ROM stands for **R**ead **O**nly **M**emory. They are programmed at the factory at the time of manufacture with a special mask, thus called a masked ROM. This is the cheapest way to manufacture ROMs once you need more than 10,000 at a time. The drawback is, if there is even one little bug in the software, that pile of 10,000 ROMs becomes worthless. Be aware that some masked ROMs are unreadable by EPROM programmers

1.2 WHAT IS A PROM

PROMs (**P**rogrammable **R**ead **O**nly **M**emory) consist of an array of fuses and thus can only be programmed one-time. Programming is accomplished with a current (instead of a voltage as are EPROMs) and requires a different type of programmer.

1.3 WHAT IS AN EPROM

EPROM (**E**rasable **P**rogrammable **R**ead **O**nly **M**emory) can be programmed and erased enabling them to be re-used. Erasure is accomplished using an UV (Ultra Violet) light source that shines through a quartz erasing window in the EPROM package.

There also are OTP (**O**ne **T**ime **P**rogrammable) EPROMs, sometimes called OTPROMs (**O**ne **T**ime **P**rogrammable **R**ead **O**nly **M**emory), that are identical to an erasable EPROM but lack an erasing window to reduce costs. To reduce the cost these EPROMs

come in a windowless plastic carrier, which is cheaper than the costly ceramic package required for the erasing window. They can be programmed one time only, so these are used after the code is bug free.

1.4 WHAT IS AN EEPROM

An EEPROM (**E**lectrically **E**rasable **P**rogrammable **R**ead **O**nly **M**emory) is similar to an EPROM but the erasure is accomplished using an electric field instead of an UV light source. This eliminates the need of a window. Usually, EEPROM refers to a device that requires a programmer or special voltage to program it.

1.5 WHAT IS A FLASH EPROM

A flash EPROM is similar to an EEPROM except that flash EPROMs are erased all at once while a regular EEROMs can erase one bye at a time. In- circuit writing and erasing is possible because no special voltages are required. To accomplish in-circuit operation, you have to write special application software routines. Flash EPROMs are also called nonvolatile memory.

1.6 HOW DOES AN EPROM WORK

EPROM memory cells use floating gate technology. A floating gate is a gate with a special capacitor for its only electrical connection. This special capacitor takes on an electrical charge in a quantum physics effect called tunneling. The presence of a charge determines the value (1 or 0) of the memory cell. In our example below, a room with a very narrow door represents the memory cell. People in the room represent electrons with their associated charge. These people can only enter or exit through a much too narrow door with much pushing or shoving to represent the tunneling effect.

Think of a room with about 30 people acting as the electron charge. A full room of people represents a '1'; when empty a zero. When an EPROM is erased, all 30 people are pushed into the room and provide the charge that we call '1'. When we program an EPROM bit, we shove these people until they pop back out by applying a pulse of high voltage to the memory cell. This pulse drives the people out of the room changing the bit from a '1' to a '0'.

When programming a bit we can only change a 1 to a 0 because changing a 0 to a 1 requires erasing. To erase an EPROM, we apply an UV (Ultra-Violet) light (that shines

directly on the chip) to drive our imaginary people back into the room. Erasure works on the whole EPROM not individual bits.

An EPROM cell is really an analog device. If it were digital, we would only have an empty room or a room with one person in it. Instead, a cell sensor circuitry compares how many people are in the room to a reference to determine if the cell represents a 1 or 0. With more than 15 people in the room, the cell is considered a 1.

When programming an EPROM you have to apply the specified programming voltage for the specified time. Too short a time or too low a voltage and not all the people get shoved out of the room. When you don't program an EPROM properly and you still have 5 people in the room, it will slow down the cell sensor circuitry, which slows down the read access time of the EPROM or might even corrupt the data. A similar thing can happen if you don't erase the EPROM long enough.

On the other hand, if you apply too high of a program voltage or over erase the EPROM, it is equivalent to blowing the doorframe right out of the wall! Our imaginary people now enter and exit the room, milling around on their own whims and we no longer have information storage! The long and short of it is you need to double check Vpp programming voltages and follow the recommended time for erasing (don't store unused chips in the eraser!). More about this later under the heading: erasing an EPROM.

(A little of the topic here, you may have seen little solid state recording devices, sometimes used in answering machines, that record voice grade audio. The trick to getting so much audio into storage is by storing analog values into EEPROM cells! The recorded cells have intermediate values that are not compared when read but out put as a voltage. When the memory cells are played back in consecutive order, the continuously changing values form an audio signal!)

2. GENERAL EPROMS

EPROM is the most popular type of PROM used in microcomputer system. This device can be programmed and erased and reprogrammed many times over by the user.

EPROM use a floating -gate avalanche-injection MOS (FAMOS) transistor cell to store charge. Applying a special programming voltage (Vpp) cause a high electric field to be

developed in the channel region of the transistor. This in turn causes electrons to jump the silicon dioxide barrier between the channel region and the floating gate. During programming the select gate is given a positive bias whitch helps attract these electrons to the floating-gate electrode. Because the floating gate is surrounded by silicon dioxide (an exellent insulator), the injected charge is effectively trapped. The storage period is projected by Intel to exceed 20 years. Cells with trapped charge cause the transistor to be biased ON, whereas those cells without trapped charge are biased OFF. Blank EPROMs have no trapped charge and each cell stores a logic 1. The EPROM can be erased by subjecting each gate to ultraviolet light, whitch has a wavelength of 2537 angstroms. The electron on the floating gate absorb photons from ultraviolet light source and acquire enough energy to reverse the programming process and return to the substrate. EPROM are packaged in special ceramic packages with quartz windows to allow erasure by EPROM eraser. The erasing process take about 15 to 20 minutes. After programming an EPROM window should be covered with an opaque label because normal room fluorescent lighting could erase the device (Intel reports that approximately three years of exposure to fluorescent light or one week of direct sunlight would be required). Eprom can be programmed in two ways. The first method is consistent with the standard programming technique used for the lower density device such as 2716 and 2732. In this technique, pin 1, Vpp, is raised to 25V or 21V, the chip enable, data applied to output D0~D7, and the desired address applied to A0~A12. Programming is accomplished by applying a 45~55 ms active low TTL level pulse to pin 27, the PGM pin. Data written, verified and the next address selected. The second way is what Intel refers to as an intelligent programming algorithm, the fastest programming algorithm. The advantage in using this algorithm is to reduce the programming time from several minutes to a few minutes (eg. 2764 type, from nearly 7 minutes to less than 1.5 minutes). Vcc equal 6.0V for this mode. The 2732 type and up also support and intelligent identifier mode. In this mode, address line A9 is rised to +12V. Two identifier bytes may then be read from the EPROM by forcing A0 low and then high. the first byte represents a manufacturer code and the second byte a device code. This is intended to allow commercial EPROM programmers to read the device and automatically select the proper pinning and programming algorithm.

2.1 EPROM 2716 TYPE

General description

2048-word x 8-bit U.V. Erasable and Programmable Read Only Memory.

The 2716 is a 2048-word by 8-bit erasable and electrically programmable ROM. This device is packaged in a 24 pin dual-in-lin package with transparent lid. The transparent lid on the package allow the memory content to be erased with ultraviolet light.

Features

- Single Power Supply .....+5V ± 5%
- Simple Programming Program Voltage: +25V-DC, Program with one 50 ms Pulse
- Static ..... No clocks Required
- Inputs and Outputs TTL Compatible During Both Read and Program Mode
- Absolute Max. Rating of Vpp Pin 26.5V
- Low Stand-by Current ..... 35 mA (max)

Modes selection

Table 1

MODE	**PINS**			
	$\overline{CE}$ (18)	**$\overline{OE}$/Vpp (20)**	**Vcc (24)**	**Outputs (9~11, 13~17)**
Read	V_{IL}	V_{IL}	+5V	D out
Stand-by	V_{IH}	X	+5V	Hi-Z
Program	V_{IL}	Vpp	+5V	D in
Verify	V_{IL}	V_{IL}	+5V	D out
Inhibit	V_{IH}	Vpp	+5V	Hi-Z

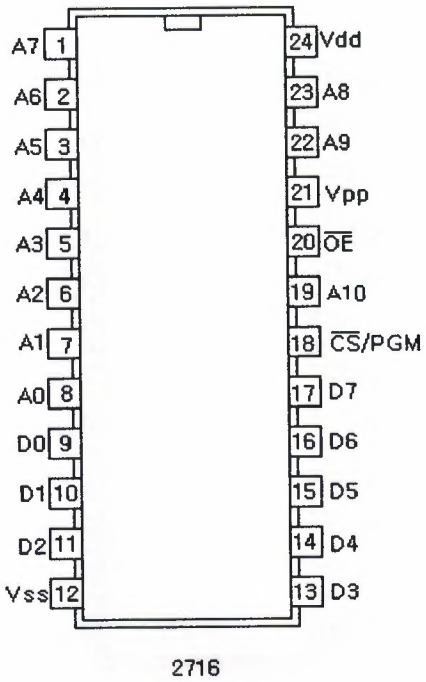


Fig 1. Eprom 2716

2.2 EPROM 2732 TYPE

General description

4096-word x 8-bit U.V. Erasable and Programmable Read Only Memory.

The 2732 is a 4096-word by 8-bit erasable and electrically programmable ROM. This device is packaged in a 24 pin dual-in-lin package with transparent lid. The transparent lid on the package allow the memory content to be erased with ultraviolet light. Optional A type (2732-A) has low programming voltage +21V.

Features

- Single Power Supply ..... +5V ± 5% or +6V ± 5%
- Simple Programming Program Voltage: +25V-DC, Program with one 50 ms Pulse
- Intelligent Programming Program Voltage +21V-DC, Program with 4 x n(1) ms Pulse
- Static ..... No clocks Required
- Inputs and Outputs TTL Compatible During Both Read and Program Mode
- Absolute Max. Rating of Vpp Pin ae26.5V
- Low Stand-by Current ..... 35 mA (max)

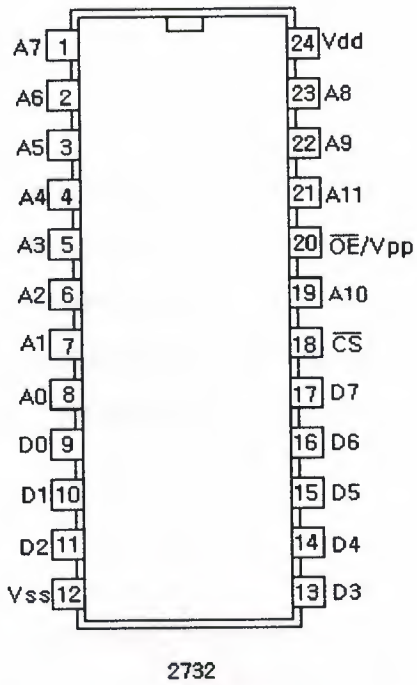


Fig 2.Eprom 2732

Modes selection

Table 2.

MODE	**PINS**			
	~~CE~~ (18)	**~~OE~~/Vpp (20)**	**Vcc (24)**	**Outputs (9~11, 13~17)**
Read	V_{IL}	V_{IL}	Vcc	D out
Stand-by	V_{IH}	X	Vcc	Hi-Z
Program	V_{IL}	Vpp	Vcc	D in
Verify	V_{IL}	V_{IL}	Vcc	D out
Inhibit	V_{IH}	Vpp	Vcc	Hi-Z

2.3 EPROM 2764 TYPE

General description

8192-word x 8-bit U.V. Erasable and Programmable Read Only Memory.

The 2764 is a 8192-word by 8-bit erasable and electrically programmable ROM. This device is packaged in a 28 pin dual-in-lin package with transparent lid. The transparent lid on the package allow the memory content to be erased with ultraviolet light. Optional A type (2764-A) has low programming voltage +12.5V.

Features

- Single Power Supply ..... +5V ± 5% or +6V ± 5%
- Simple Programming Program Voltage: +21V-DC, Program with one 50 ms Pulse
- Intelligent Programming Program Voltage +21V-DC, Program with 4 x n(1) ms

Pulse

- Static ..... No clocks Required
- Inputs and Outputs TTL Compatible During Both Read and Program Mode
- Absolute Max. Rating of Vpp Pin 26.5V
- Low Stand-by Current ..... 35 mA (max)

Modes selection

Table 3.

MODE	**PINS**					
	~~CE~~ (20)	**~~OE~~ (22)**	**Vpp (1)**	**~~PGM~~ (27)**	**Vcc (28)**	**Outputs (11~13, 15~19)**
Read	V_{IL}	V_{IL}	Vcc	V_{IH}	Vcc	D out
Stand-by	V_{IH}	X	Vcc	X	Vcc	Hi-Z
Program	V_{IL}	X	Vpp	V_{IL}	Vcc	D in
Verify	V_{IL}	V_{IL}	Vpp	V_{IH}	Vcc	D out
Inhibit	V_{IH}	X	Vpp	X	Vcc	Hi-Z

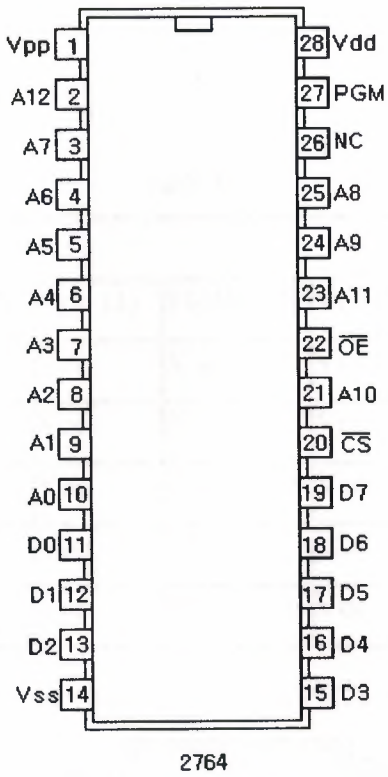


Fig 3.Eprom 2764

2.4 EPROM 27128 TYPE

General description

16384-word x 8-bit U.V. Erasable and Programmable Read Only Memory.

The 27128 is a 16384-word by 8-bit erasable and electrically programmable ROM. This device is packaged in a 28 pin dual-in-lin package with transparent lid. The transparent lid on the package allow the memory content to be erased with ultraviolet light.

Optional A type (27128-A) has low programming voltage +12.5V.

Features

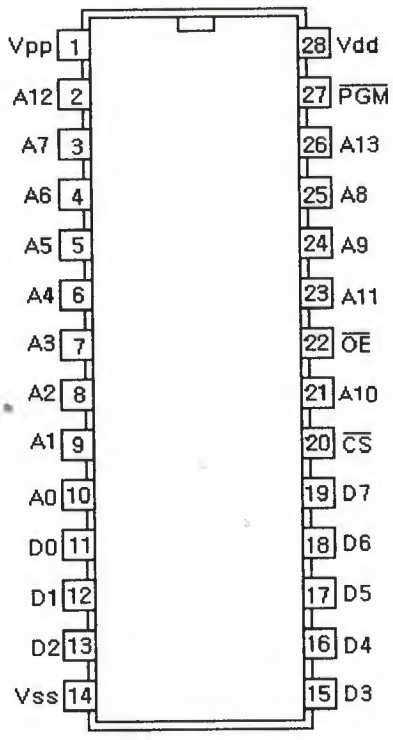
- Single Power Supply ..... +5V ± 5% or +6V ± 5%
- Simple Programming Program Voltage: +21V-DC, Program with one 50 ms Pulse
- Intelligent Programming Program Voltage +21V-DC, Program with 4 x n(1) ms Pulse

- Static ..... No clocks Required
- Inputs and Outputs TTL Compatible During Both Read and Program Mode
- Absolute Max. Rating of Vpp Pin 26.5V
- Low Stand-by Current ..... 35 mA (max)

Modes selection

Table 4.

MODE	PINS					
	~~CE~~ (20)	~~OE~~ (22)	Vpp (1)	~~PGM~~ (27)	Vcc (28)	Outputs (11~13, 15~19)
Read	V_{IL}	V_{IL}	Vcc	V_{IH}	Vcc	D out
Stand-by	V_{IH}	X	Vcc	X	Vcc	Hi-Z
Program	V_{IL}	X	Vpp	V_{IL}	Vcc	D in
Verify	V_{IL}	V_{IL}	Vpp	V_{IH}	Vcc	D out
Inhibit	V_{IH}	X	Vpp	X	Vcc	Hi-Z



27128, 27C128

2.5 EPROM 27256 TYPE

General description

32768-word x 8-bit U.V. Erasable and Programmable Read Only Memory.

The 27256 is a 32768-word by 8-bit erasable and electrically programmable ROM. This device is packaged in a 28 pin dual-in-lin package with transparent lid. The transparent lid on the package allow the memory content to be erased with ultraviolet light.

Features

- Single Power Supply ..... +5V ± 5% or +6V ± 5%
- Simple Programming Program Voltage: +12.5V-DC, Program with one 50 ms Pulse
- Intelligent Programming Program Voltage +12.5V-DC, Program with 4 x n(1) ms

Pulse

- Static ..... No clocks Required
- Inputs and Outputs TTL Compatible During Both Read and Program Mode
- Absolute Max. Rating of Vpp Pin 14.0V
- Low Stand-by Current ..... 35 mA (max)
- Low Power Dissipasion (active mode) ..... 20 mW/MHz typ.

Modes selection

Table 5

MODE	**PINS**				
	$\overline{CE}$ (20)	**$\overline{OE}$ (22)**	**Vpp (1)**	**Vcc (28)**	**Outputs (11~13, 15~19)**
Read	V_{IL}	V_{IL}	Vcc	Vcc	D out
Stand-by	V_{IH}	X	Vcc	Vcc	Hi-Z
Program	V_{IL}	V_{IH}	Vpp	Vcc	D in
Verify	V_{IL} / V_{IH}	V_{IL}	Vpp	Vcc	D out
Inhibit	V_{IH}	V_{IH}	Vpp	Vcc	Hi-Z

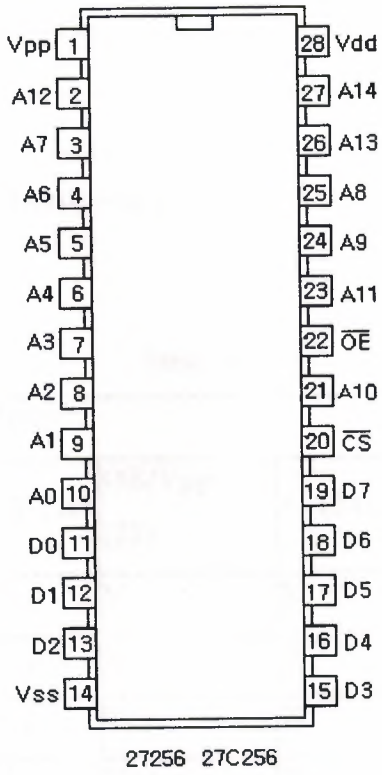


Fig 5.Eprom 27256

2.6 EPROM 27512 TYPE

General description

65536-word x 8-bit U.V. Erasable and Programmable Read Only Memory.

The 27512 is a 65536-word by 8-bit erasable and electrically programmable ROM. This device is packaged in a 28 pin dual-in-lin package with transparent lid. The transparent lid on the package allow the memory content to be erased with ultraviolet light.

Features

- Single Power Supply ..... +5V ± 5% or +6V ± 5%
- Simple Programming Program Voltage: +12.5V-DC, Program with one 50 ms Pulse
- Intelligent Programming Program Voltage +12.5V-DC, Program with 4 x n(1) ms

Pulse

- Static ..... No clocks Required

- Inputs and Outputs TTL Compatible During Both Read and Program Mode
- Absolute Max. Rating of Vpp Pin 14.0V
- Low Stand-by Current ..... 40 mA (max)
- Low Active Power ..... 125 mA (max)
- Intelligent Programming Identifier at A_9

Modes selection

Table 6.

MODE	PINS				
	~~CE~~ (20)	~~OE~~/Vpp (22)	A_9 (24)	Vcc (28)	Outputs (11~13, 15~19)
Read	V_{IL}	V_{IL}	X	Vcc	D out
Stand-by	V_{IH}	X	X	Vcc	Hi-Z
Program	V_{IL}	Vpp	X	Vcc	D in
Verify	V_{IL}	Vpp	X	Vcc	D out
Inhibit	V_{IH}	Vpp	X	Vcc	Hi-Z
Intelligent Identifier	V_{IL}	V_{IL}	V_H	Vcc	Code

Notes

- X = don't care (V_{IL} or V_{IH})
- All voltage ± 5%
- V_H = 12.0V ± 0.5V
- Vcc = +5V ± 5% (standard programming) or +6V ± 5% (intelligent programming)
- Vpp = +25V ± 0.5V, +21V ± 0.5V or +12.5V ± 0.5V depends on the types

3. STANDARD PROGRAMMING ALGORITHM

For programming, CE should be kept TTL low at all times while Vpp is kept at 21V. When the address and data are stable, a 50 msec, active low, TTL program pulse is applied to PGM input. A program pulse must be applied at each address location to be programmed. You can program any location at any time either individually, sequentially, or at random. The program pulse has a maximum width 55 msec.

3.1 D.C. PROGRAMMING CHARACTERISTICS:

T_A=25±5°C,Vcc=5V±5%,Vpp=21V±0.5V

Table 7.

Symbol	Parameter	Limits			Test Conditions
		Min	Max	Units	
I_{IL}	Input Current (All Inputs)		10	μA	V_{IN}=V_{IL} or V_{IH}
V_{OL}	Output Low Voltage During Verify		0.45	V	I_{OL}=2.1mA
V_{OH}	Output High Voltage During Verify	2.4		V	I_{OH}=-400μA
V_{IL}	Input Low Level (All Input)	-0.1	0.8	V	
V_{IH}	Input High Level	2.0	Vcc+1	V	
I_{CC1}	Vcc Supply Current (Inhibit)		40	mA	~~CE~~=V_{IH}
I_{CC2}	Vcc Supply Current (Program & Verify)		100	mA	
I_{PP2}	Vpp Supply Current (Program)		30	mA	~~CE~~=V_{IL}=~~PGM~~
I_{PP3}	Vpp Supply Current (Verify)		5	mA	~~CE~~=V_{IL} ~~PGM~~=V_{IH}
I_{PP4}	Vpp Supply Current (Inhibit)		5	mA	~~CE~~=V_{IH}
V_{ID}	A9 Intelligent Identifier Voltage	11.5	12.5	V	

3.2 A.C. PROGRAMMING CHARACTERISTICS:

T_A=25±5°C,Vcc=5V±5%,Vpp=21V±0.5V

Table 8.

Symbol	Parameter	Limits				Test Conditions
		Min	Typ	Max	Units	
t_{AS}	Address Setup Time	2			µs	
t_{OES}	OE Setup Time	2			µs	
t_{DS}	Data Setup Time	2			µs	
t_{AH}	Address Hold Time	0			µs	
t_{DH}	Data Hold Time	2			µs	
t_{DFP}[2]	Output Enable to Output Float Delay	0		130	ns	
t_{VS}	Vpp Setup Time	2			µs	
t_{PW}	PGM Pulse Width During Programming	45	50	55	ms	
t_{CES}	CE Setup Time	2			µs	
t_{OE}	Data Valid from OE			150	ns	

3.3 A.C. CONDITION OF TEST

- Input Rise and Fall Times (10% to 90%).....20ns
- Input Pulse Levels.....0.45V to 2.4V
- Input Timing Reference Level.....0.8V and 2.0V
- Output Timing Reference Level.....0.8V and 2.0V

- Notes :

1. Vcc must be applied simultaneously or before Vpp and removed simultaneously or after Vpp.
2. This parameter is only sampled and is not 100% tested. Output Float is defined as the point where data is no longer driven, see the timing diagram.

3.4 STANDARD PROGRAMMING WAVEFORMS

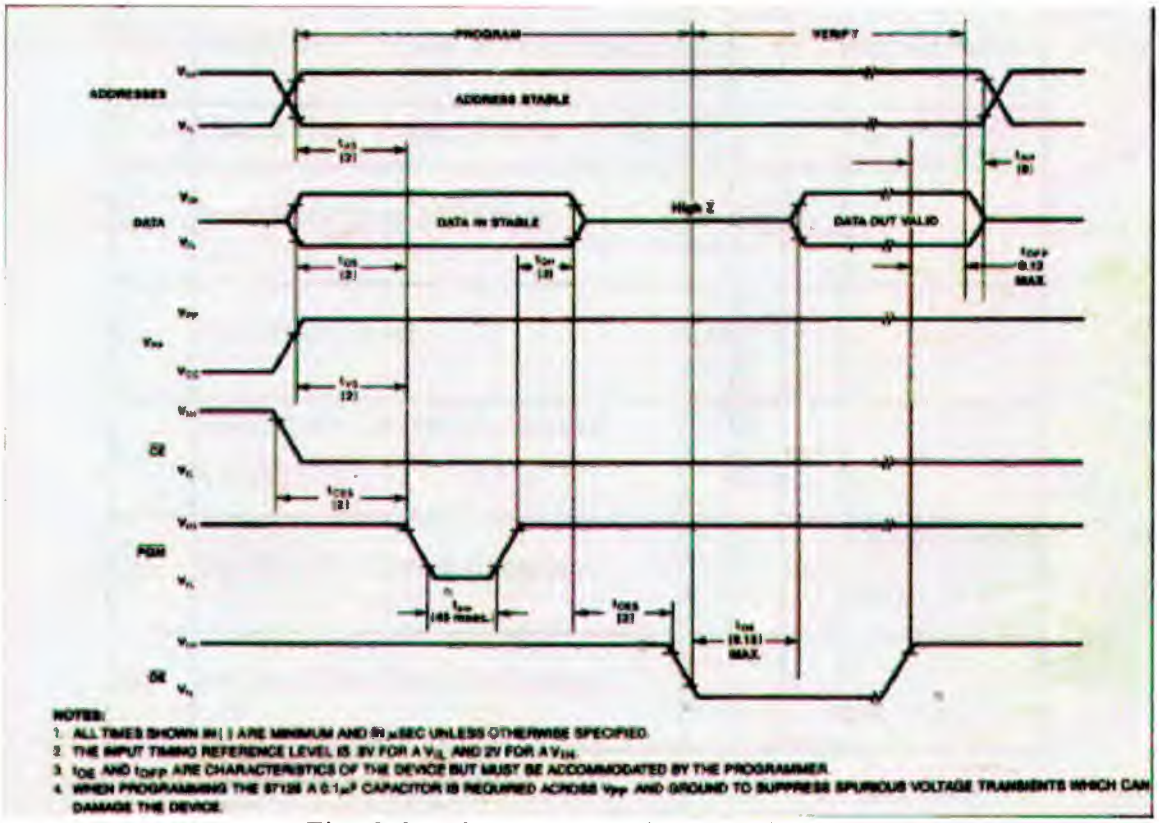


Fig 6. Standart programming waveforms

4. INTELLIGENT PROGRAMMING ALGORITHM

4.1 D.C. PROGRAMMING CHARACTERISTICS:

T_A=25±5°C,Vcc=6V±0.25V,Vpp=21V±0.5V

Table 9.

Symbol	Parameter	Limits			Test Conditions
		Min	Max	Units	
I_{IL}	Input Current (All Inputs)		10	µA	$V_{IN}=V_{IL}$ or V_{IH}
V_{OL}	Output Low Voltage During Verify		0.45	V	I_{OL}=2.1mA
V_{OH}	Output High Voltage During Verify	2.4		V	I_{OH}=-400µA
V_{IL}	Input Low Level (All Input)	-0.1	0.8	V	
V_{IH}	Input High Level	2.0	Vcc	V	
I_{CC2}	Vcc Supply Current (Program & Verify)		100	mA	
I_{PP2}	Vpp Supply Current (Program)		30	mA	~~CE~~=V_{IL}=~~PGM/WE~~
V_{ID}	A9 Intelligent Identifier Voltage	11.5	12.5	V	

4.2 A.C. PROGRAMMING CHARACTERISTICS:

T_A=25±5°C,Vcc=5V±5%,Vpp=21V±0.5V

Table 10.

Symbol	Parameter	Limits				Test Conditions
		Min	Typ	Max	Units	
t_{AS}	Address Setup Time	2			µs	
t_{OES}	OE Setup Time	2			µs	
t_{DS}	Data Setup Time	2			µs	
t_{AH}	Address Hold Time	0			µs	
t_{DH}	Data Hold Time	2			µs	
t_{DFP}[4]	Output Enable to Output Float Delay	0		130	ns	
t_{VPS}	Vpp Setup Time	2			µs	
t_{VCS}	Vcc Setup Time	2			µs	
t_{PW}	PGM/WE Initial Program Pulse Width	0.95	1.0	1.05	ms	
t_{OPW}	PGM/WE Over/Program Pulse Width	3.8		63	ms	
t_{CES}	CE Setup Time	2			µs	
t_{OE}	Data Valid from OE			150	ns	

4.3 A.C. CONDITION OF TEST

- Input Rise and Fall Times (10% to 90%).....20ns
- Input Pulse Levels.....0.45V to 2.4V
- Input Timing Reference Level.....0.8V and 2.0V
- Output Timing Reference Level.....0.8V and 2.0V

- Notes :

1. Vcc must be applied simultaneously or before Vpp and removed simultaneously or after Vpp.
2. The length of the over program pulse will vary from 3.8 msec to 63 msec as a function of the iteration counter value X.
3. Initial program pulse width tolerance is 1 msec±5%.
4. This parameter is only sampled as is not 100% tested. Ouput float is defined as the point where data is no longer driven, see timing diagram on the following page.

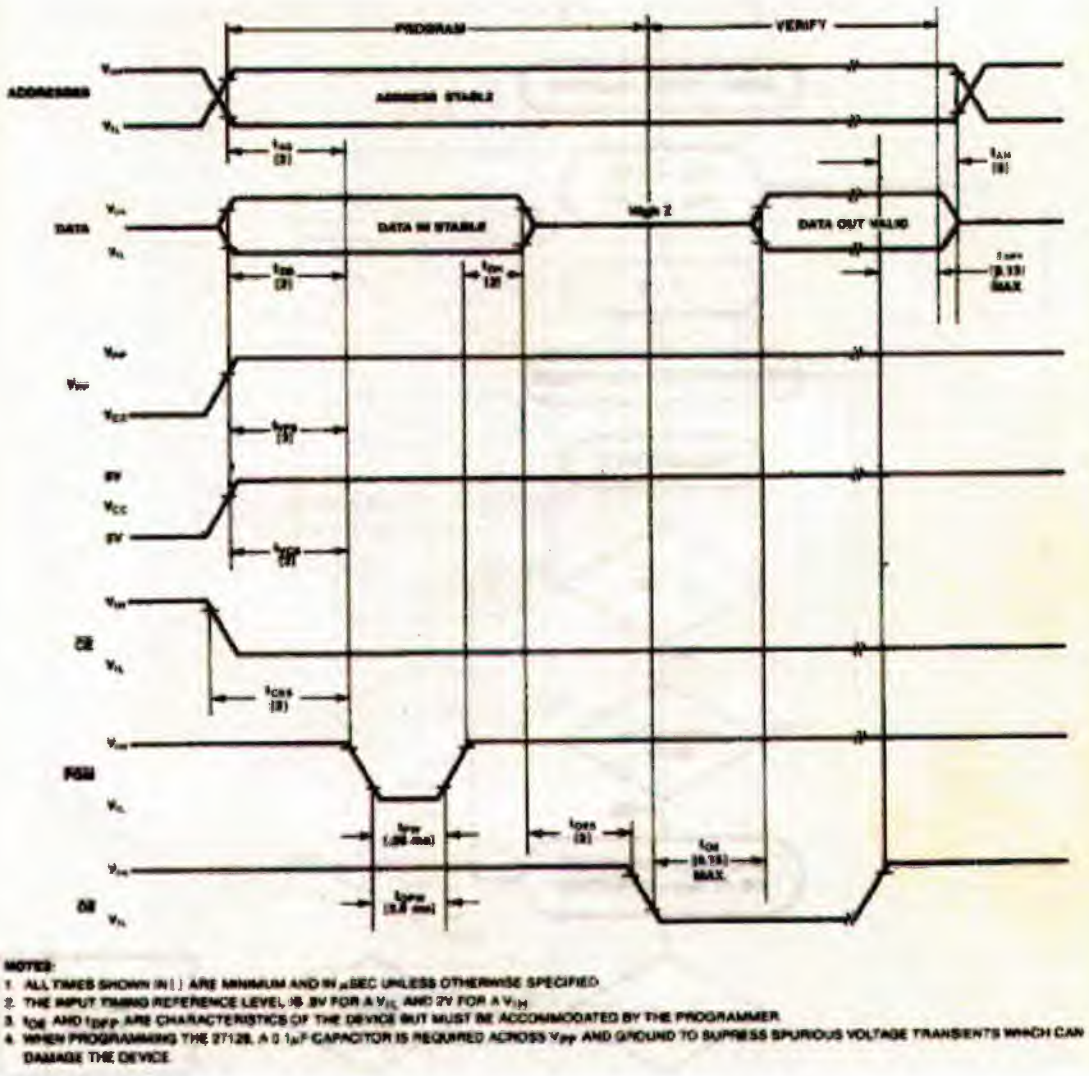
4.4 INTELLIGENT PROGRAMMING ALGORITHM

The Intelligent programming algorithm is the prefered programming method since it allows intel EPROM to be programmed in a significantly faster time than the standard 50 msec per byte programming routine. Typical programming times for 27128s are on the order of two minutes, which is a six-fold reduction in programming time from the standard method. This fast algorithm result in improved reliability characteristics over the standard 50 msec algorithm. A flowchart of 27128 intelligent programming algorithm is shown in figure 3. This is compatible with the 2764 intelligent programming algorithm.

This fast algorithm assures reliable programming through the "closed loop" technique of margin checking. To ensure reliable program margin the intelligent programming algorithm utilizes two different pulse types: initial and overprogram. The duration of the initial programming pulse is one millisecond, which will then be followed by a longer overprogram pulse of length 4X msec. X in an iteration counter and is equal to the number of the initial one millisecond pulse applied to a particular 27128 location, before a correct verify occurs. Up to 15 one millisecond pulses per byte are provided for before the overprogram pulse is applied.

The entire sequence of program pulses and byte verifications is performed at Vcc=6.0V and Vpp=21.0V. When the intelligent programming cycle has been commpleted, all bytes should be compared to the original data with Vcc=Vpp=5.0V.

4.5 INTELLIGENT PROGRAMMING WAVEFORMS



NOTES:
1. ALL TIMES SHOWN IN ( ) ARE MINIMUM AND IN μSEC UNLESS OTHERWISE SPECIFIED.
2. THE INPUT TIMING REFERENCE LEVEL IS .8V FOR A V_{IL} AND 2V FOR A V_{IH}.
3. t_{OE} AND t_{DFP} ARE CHARACTERISTICS OF THE DEVICE BUT MUST BE ACCOMMODATED BY THE PROGRAMMER.
4. WHEN PROGRAMMING THE 27128, A 0.1μF CAPACITOR IS REQUIRED ACROSS V_{PP} AND GROUND TO SUPPRESS SPURIOUS VOLTAGE TRANSIENTS WHICH CAN DAMAGE THE DEVICE.

Fig 7. Intelligent programming waveforms

4.6 INTELLIGENT PROGRAMMING FLOWCHART

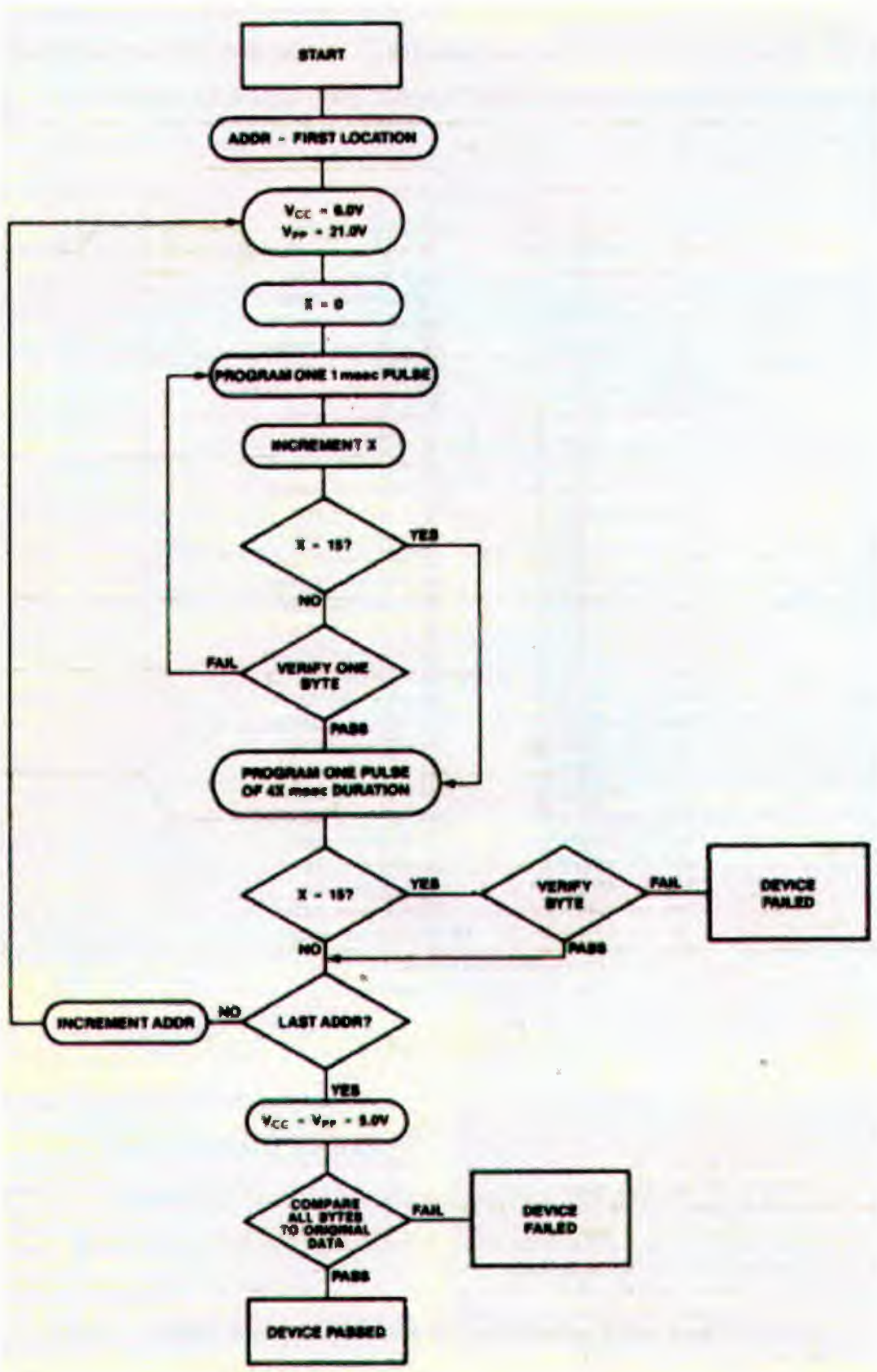


Fig 8. Intelligent programming flowchart

5.ISA BUS

5.1 BRIEF DESCRIPTION AND IO DEVICE DESIGN

The purpose of this web page is to introduce you to the PC ISA bus and to show you how to do simple IO designs using this bus. First, consider a picture of the bus shown

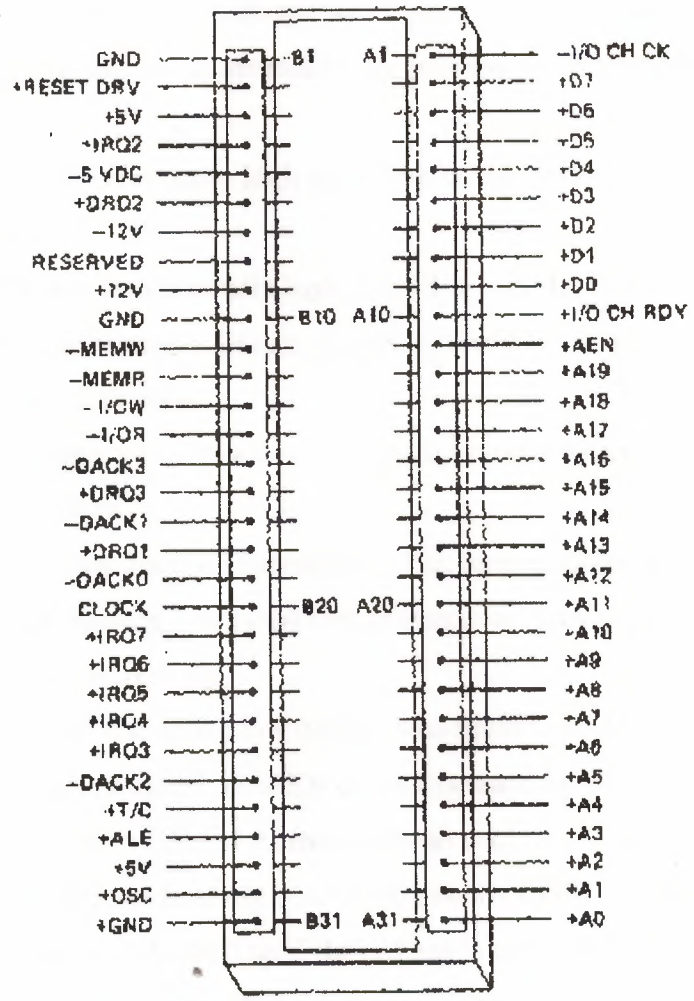


Fig 10. Isa bus

5.2 ISA PIN DESCRIPTIONS

DATA - Lines D0-D7 form the biirectional data bus. The data bus is driven by a bidirectional buffer that is only enabled at times corresponding to data availability (iowc/, iorc/, memw/, memr).

ADDRESS - Output lines A0-A19 form the address bus. These lines are fully demultiplexed and stable during a full bus cycle.

ALE - Address Latch Enable - This output signal comes directly from the bus controller IC and provides timing information for decoding the address lines. It is not needed for bus decoding since the address lines are already demultiplexed on the ISA bus.

AEN - Address Enable - This output signal allows the IO device to distinguish between processor bus cycles and DMA bus cycles. A high on AEN indicates that a DMA cycle is occuring and that the address, data and control lines are under the control of the DMA controller. Peripheral IO devices that do not have DMA capability should insure that they only decode address that are generated by the processor (AEN='0') and not a DMA controller.

IO CHANNEL RDY - This normally high input line can be pulled low by a slow device to inserve wait states.

IO CHANNEL CHECK - This normally high input line is pulled low to indicate a memory or IO device pari6ty error. In turn, the parity error will cause a non-maskable interrupt (NMI) of Type 2 to occur.

RESET DRV - This output signal is active high during power-on and can be used to reset or initialize IO devices.

DRQ1-DRQ3 - These input lines are connected to the corresponding DMA request pins on the DMA controller. Raising a selected line generates a DMA request. DMA channel 0 is reserved for memory refreshing.

DACK0-DACK3 - These four active low output lines provide DMA acknowledge signals for the four DMA channesl. DACK0 can not be used by other devices but is useful since it indicates that a DRAM memory refresh cycle is occuring.

IRQ2-IRQ7 - Interrupt request lines are connected directly to the PC interrupt controller. A line should be held high until the request is serviced by the appropriate interrupt service routine. IRQ0 and IRQ1 are reserved for use on the system board by the time of day and keyboard interupts and are not generally available. Other lines may be used by other devices as well.

OSC - The output of the system oscillator, typically around 8-20MHz (traditionally 14.318MHz).

CLK - Traditionally the processor clock signal with a 33% duty cycle (4.77MHz). More recent PC systems will have frequencies in the range of 4-10MHz. The frequency of this clock output should always be measured to insure that it is used properly.

POWER SUPPLY LINES - All voltages available on the system board are available on the ISA bus. These include +5Vdc and ground, +12 and -12Vdc, and -5Vdc. The current draw from each of these lines should be carefully controlled.

interface design

- Any IO device interfaced to the PC bus must decode an address, decide whether the address is originating from the host processor or a DMA device, determine the direction of data transfer and, if appropriate, either accept data or provide data at the proper time.
- The minimum set of signals that must be used in an interface design are: AEN, A0-A9, IOWC/ and IORC/.
- ADDRESS Decoding - AEN and A0-A9 are used to decode a 10 bit ISA address generated by the processor. AEN must be low. Only A0-A9 are used for IO address (*not* memory address) decoding by convention with the original PC. The upper address lines are ignored.
- IOWC/ and IORC/ are used in conjunction with address decoding to generate a device select pulse (DSP) that can be used to strobe data into a latch (IOWC/) or enable data onto the data bus (IORC/) to be read by the host processor.

Simple output port design

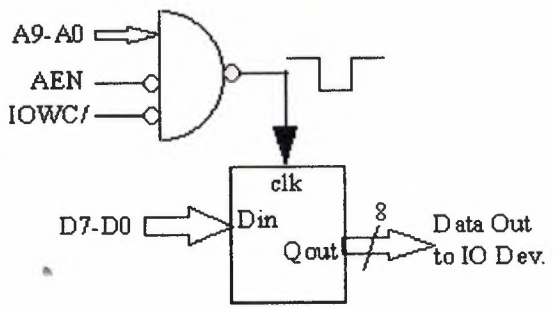


Fig 11.Simple output design

Note in this design that you need an address decoder that produces a negative true output pulse (the NAND gate) in response to a selected address (probably in the range of 300-330h) and only when AEN is low and IOWC/ is low. The trailing edge of the DSP signal goes into the clock strobe of the 8 bit latch and is used to latch the data from the ISA bus at the correct time.

Simple input port design

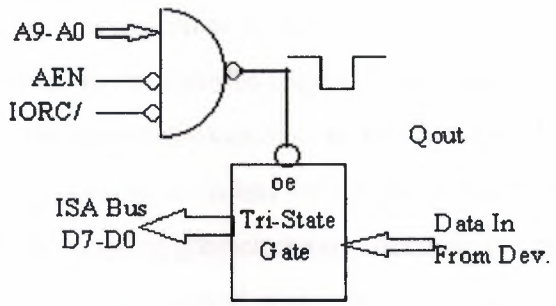


Fig 12.Simple input design

The design for an input port is similar to the design for an output port with the exception that you need an 8-bit tri-state gate to buffer the data from the IO device to the ISA bus. The gate is an essential component and is there to protect the ISA bus from being driven at the wrong time. The address and control signal decode logic provides the Read device select pulse signal (negative true) that determines when data from the IO device is passed to the ISA data bus to be read by the processor.

6. ERASER EPROM

6.1 PREFACE

An EPROM can be reprogrammed if we erased the data. How do the erase work ? EPROM can be erased used an ultraviolet light. Usually EPROM being erased by used a specific ultraviolet lamp. This lamp has a wave-length about 2537 Angstrom or less than 400 nm. Even with sun light or neon light can erased EPROM, it is the reason why an EPROM must cover its quartz crystal window with an opaque label. Research showed that a 2716 type should be erased after 3 years by exposured with neon light continuously. With sun light, it's about one week.

The best way to erased EPROM is by exposure the crystal window with 253.7 nm ultraviolet light and intensity 12 mW/m^2. The window has a distance about 2 to 3 cm.

The exposure must apply as long as about 10 to 40 minutes (usually 15 to 20 minutes is enough).

What is the philosophy of erased an EPROM ?

The philosophy is that when we erased an EPROM, we make the data stored in it to be logic 1 all. EPROM use a floating-gate avalanche-injection MOS (FAMOS) transistor cell to store charge. Applying a special programming voltage (Vpp) cause a high electric field to be developed in the channel region of the transistor. This is in turn causes electrons to jump the silicon dioxide barrier between the channel region and the floating gate. During programming the select gate is given a positive bias which helps attract these electrons to the floating gate electrode. Because the floating gate is surrounded by silicon dioxide, the injected charge is effectively trapped. Cell with trapped charge cause the transistor to be biased on, where as those cells without trapped charge are biased off. Blank EPROMs have no trapped charge and each cell store a logic 1. The electrons on the floating gate absorb photons from the ultraviolet light source and acquire enough energy to reverse the programming process and return to the substrate.

6.2 ERASER

There are many type of EPROM eraser sold in the market, but the price is too high. If we want to do some work, we can build an EPROM eraser with a low cost. Here it is.

To get an ultraviolet lamp, we can used a specific lamp from philips, i.e.: TUV 6 W. this lamp has a wave-length that match for our purpose. This lamp usually used for sterilization medical devices. May be this type quiet hard to find. Another type is Germicidal lamp. This is also for sterilize. I have used this one and seems to work good. More about Germicidal lamp please visit Sci.Electronics FAQ: Cheap EPROM Eraser

Last search about Germicidal lamp, I got a catalog info at my office from 'EYE' maker, one of the largest lamp maker. Here if you want to see it.

Note that neon lamp type usually used for aquarium accessory can not be used, because the wave-length is not match.

The lamp is about 5 to 10 W. Almost the same as the general neon lamp. The glass is clear, and the fillament can be seen directly. The length about 15 to 25 cm depends on

the rating. Used a 10 W / 220 V ballast coil and a starter. Figure I-1. shows the schematic diagram for this lamp connection (the same like conventional neon lamp).

Warning!!!

Be careful when apply circuit to main line, allways check and recheck the circuit before connecting it in the first time, and also keep in mind that do not see the lamp light directly because ultraviolet light can make you blind !

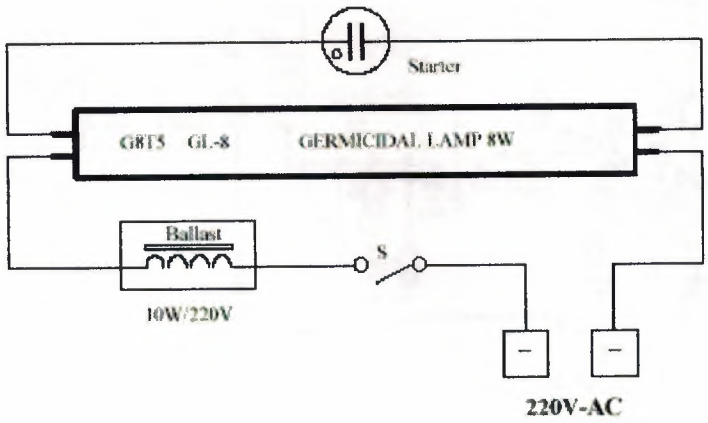


Fig 13. Neon lamp wiring schematic.

- Part list for the eraser circuit :

1. Neon ballast coil, 10W/220V..... 1 pce
2. Neon Starter, 10 - 65W..... 1 pce
3. Germicidal lamp, type: G8T5, GL-8, 8W..... 1 pce
4. Neon fitting..... 2 pcs
5. Electrical switch, 2A/220V..... 1 pce
6. Main outlet plug connector, 2A/220V..... 1 pce

The erasing time of an EPROM variate about 10 to 40 minutes. It is a pity to waste our time to waiting it while watch the clock round until the EPROM has to be erased. Here a circuit to get the perfect time while it is erased, we can do some another work and let the eraser work by it self. Figure I-2. shows this timer circuit diagram.

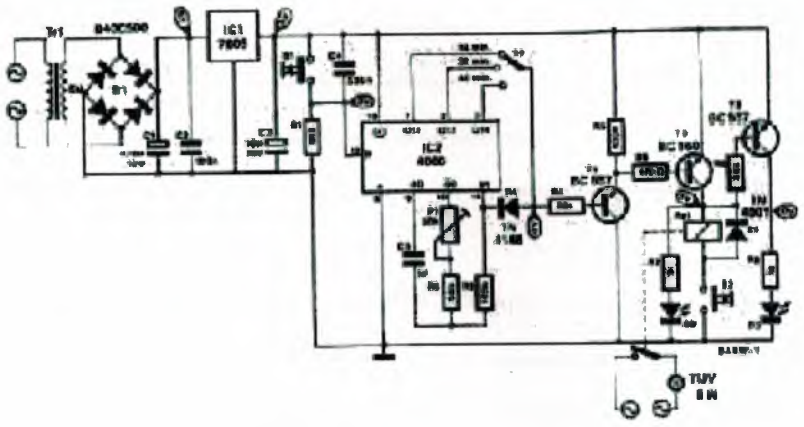


Fig.14. Timer circuit diagram.

- Part list for the timer circuit :
- Resistors :

1. R1=10 kOhm ..... 1 pce
2. R2=180 kOhm ..... 1 pce
3. R3, R4, R6=56 kOhm ..... 3 pcs
4. R5=100 kOhm ..... 1 pce
5. R7, R8=1 kOhm ..... 2 pcs
6. R9=470 Ohm ..... 1 pce
7. Variable P1=50 kOhm ..... 1 pce

- Capasitors :

1. C1=470 uF/16V (electrolith) ..... 1 pce
2. C2=100 nF (ceramic) ..... 1 pce

3. C3=10 uF/10V (electrolith) ..... 1 pce
4. C4=330 nF (ceramic) ..... 1 pce
5. C5=1 uF (non polar) ..... 1 pce

- Diodes :

1. D1=1N 4001 ..... 1 pce
2. D2=Red LED ..... 1 pce
3. D3=Yellow LED ..... 1 pce
4. D4=1N 4148 ..... 1 pce
5. Bridge BD=1N 4001 ..... 4 pcs

- Semiconductors :

1. Transistor T1, T3=BC 557 ..... 2 pcs
2. Transistor T2=BC 160 ..... 1 pce
3. IC1=7805 ..... 1 pce
4. IC2=4060 ..... 1 pce

- Others :

1. Transformer Tr1=500mA/6V sec, 220V pri ..... 1 pce
2. Relay Re1=5V/100mA or 6V/100mA ..... 1 pce
3. Push on switch S1 ..... 1 pce
4. Microswitch S2 .....1 pce
5. Selector switch 1P3T (rotary) S3 ..... 1 pce

6.3 CONSTRUCTION

Because the lamp must not be seen directly when on so we must make the make box fully closed. Figure I-3. shows an example of the construction box. The dimension depends on the lamp size.

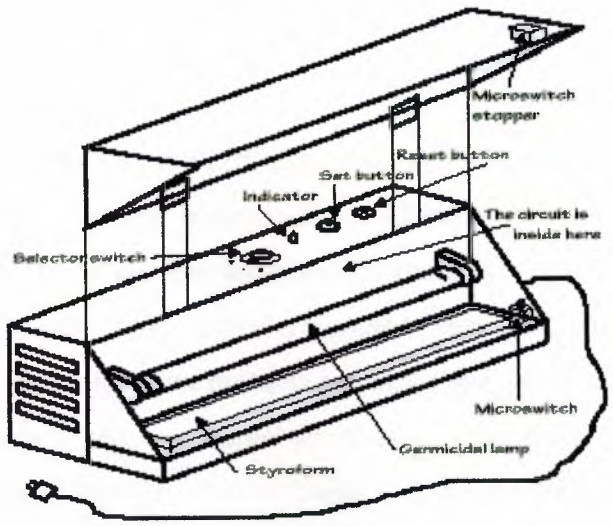


Fig. 15. EPROM eraser box construction.

More attention to mounting the microswitch in the cover box. It must be on when the cover is fully closed or the circuit is never work Also it is not right if the microswitch has to be on when the cover not fully closed yet. Set the distance between the ICs and lamp about 2 to 3 cm length. You can add some hole in the bottom of box for ventilation, because when the lamp is on, the temperature inside may be rised. This was not showed in the figure, but I had added it in my prototype. There is some gap between the styroform and the bottom of box. So, you can make this hole. Use small hinges to open or close the cover. The box divide by two section. The back section is to mounting the timer PCB, ballast coil and lamp starter. Also some panel in the top of box for push button set/reset, power indicator lamp, and the selector switch to choose the right time to exposure the EPROM.

7. DATA CODING

7.1 DATA CODING

Smart as they are, computers have difficulty reading ordinary text. While these alphabets work well for physical representations of letters and words, they fail in the realm of electronics. Oddly, ancient scribes never thought of applying digital techniques to their fledgling alphabets. Only in modern times have people sought to standardize a correspondence between digital bit patterns and alphabetic characters.

Four major systems have been developed for encoding characters as data. These include Binary Coded Decimal, Extended Binary Coded Decimal Interchange Code, the American Standard Code for Information Interchange, and UniCode.

7.2 BCD

When electronic calculators first appeared, the code was obvious: assign a binary code to each of the ten numerals commonly used in our favored decimal system. The shortest code that works is four bits, sufficient to encode 16 symbols. The leftover six can be used for mathematical operators or whatever you like.

The basic code that uses four bits for the ten numerals is called Binary Coded Decimal or BCD and is still used in some data systems. Table D.1 lists the simple BCD code.

Table 10. Binary coded decimal

Binary code	**Numeral**
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7

1000	8
1001	9

Useful as it is, BCD doesn't go far enough. It encodes only numbers. Adding letters and control information requires something more. Engineers were happy to come up with more, even too much more.

7.3 EBCDIC

When IBM developed its 360-series of mainframe computers, it developed its own eight-bit data code to encompass the alphabet. Building on the foundation of BCD, IBM extended the code by adding four more bits and created what it called the Extended Binary Coded Decimal Interchange Code or EBCDIC.

In the EBCDIC system as developed by IBM, characters were not assigned to all of the potential code values, leaving many of them undefined. Although this code is still used by many larger computer systems, few PC applications understand it. With any luck, you will never encounter EBCDIC files when working with your PC. For the sake of completeness, however, Table D.2 lists EBCDIC codes.

Table 11. The extended binary coded decimal interchange code

Decimal	**Hex**	**Symbol or mnemonic**	**Function**
0	0	NUL	Null
1	1	SOH	Start of heading (indicator)
2	2	STX	Start of text (indicator)
3	3	ETX	End of text (indicator)
4	4	PF	Punch off
5	5	HT	Horizontal tab
6	6	LC	Lower case
7	7	DEL	Delete
8	8		
9	9		

10	A	SMM	Start of Manual Message
11	B	VT	Vertical tab
12	C	FF	Form feed
13	D	CR	Carriage return
14	E	SO	Shift out
15	F	SI	Shift in
16	10	DLE	Data link escape
17	11	DC1	Device control 1
18	12	DC2	Device control 2
19	13	TM	Tape mark
20	14	RES	Restore
21	15	NL	New line
22	16	BS	Backspace
23	17	IL	Idle
24	18	CAN	Cancel
25	19	EM	End of medium
26	1A	CC	Cursor control
27	1B	CU1	Customer use 1
28	1C	IFS	Interchange file separator
29	1D	IGS	Interchange group separator
30	1E	IRS	Interchange record separator
31	1F	IUS	Interchange unit separator
32	20	DS	Digit select
33	21	SOS	Start of significance
34	22	FS	Field separator
35	23		
36	24	BYP	Bypass
37	25	LF	Line feed
38	26	ETB	End of transmission block
39	27	ESC	Escape

40	28		
41	29		
42	2A	SM	Set mode
43	2B	CU2	Customer use 2
44	2C		
45	2D	ENQ	Enquiry
46	2E	ACK	Acknowledge
47	2F	BEL	Bell
48	30		
49	31		
50	32	SYN	Synchronous idle
51	33		
52	34	PN	Punch on
53	35	RS	Reader stop
54	36	UC	Upper case
55	37	EOT	End of transmission
56	38		
57	39		
58	3A		
59	3B	CU3	Customer use 3
60	3C	DC4	Device control 4
61	3D	NAK	Negative acknowledge
62	3E		
63	3F	SUB	Substitute
64	40	SP	Space
65	41		
66	42		
67	43		
68	44		
69	45		

70	46		
71	47		
72	48		
73	49		
74	4A	Cent sign	
75	4B		
76	4C	<	Less than sign
77	4D	(	Open parenthesis
78	4E	+	Plus sign
79	4F	|	Logical OR
80	50	&	Ampersand
81	51		
82	52		
83	53		
84	54		
85	55		
86	56		
87	57		
88	58		
89	59		
90	5A	!	Exclamation mark
91	5B	$	Dollar sign
92	5C	.	Period
93	5D	)	Close parenthesis
94	5E	;	Semi-colon
95	5F		
96	60	-	Minus sign, hyphen
97	61	/	Slash
98	62		
99	63		

100	64		
101	65		
102	66		
103	67		
104	68		
105	69		
106	6A		
107	6B	,	Comma
108	6C	%	Percent sign
109	6D	_	Underscore
110	6E	>	Greater than sign
111	6F	?	Question mark
112	70		
113	71		
114	72		
115	73		
116	74		
117	75		
118	76		
119	77		
120	78		
121	79		
122	7A	:	Colon
123	7B	#	Number sign
124	7C	@	At sign
125	7D	'	Single quote
126	7E	=	Equal sign
127	7F	"	Double quote
128	80		
129	81	A	

130	82	B	
131	83	C	
132	84	D	
133	85	E	
134	86	F	
135	87	G	
136	88	H	
137	89	İ	
138	8A		
139	8B	Open curly bracket	
140	8C	Bar	
141	8D	Close curly bracket	
142	8E	Tilde	
143	8F		
144	90		
145	91	J	
146	92	K	
147	93	L	
148	94	M	
149	95	N	
150	96	O	
151	97	P	
152	98	Q	
153	99	R	0
154	9A		
155	9B		
156	9C		
157	9D		
158	9E		
159	9F		

160	A0		
161	A1		
162	A2	S	
163	A3	T	
164	A4	U	
165	A5	V	
166	A6	W	
167	A7	X	
168	A8	Y	
169	A9	Z	
170	A A		
171	AB		
172	AC		
173	A D		
174	AE		
175	AF		
176	B0		
177	B1		
178	B2		
179	B3		
180	B4		
181	B5		
182	B6		
183	B7		
184	B8		
185	B9		
186	BA		
187	BB		

188	BC		
189	BD		
190	BE		
191	BF		
192	C0		
193	C1	A	
194	C2	B	
195	C3	C	
196	C4	D	
197	C5	E	
198	C6	F	
199	C7	G	
200	C8	H	
201	C9	I	
202	CA		
203	CB		
204	CC		
205	CD		
206	CE		
207	CF		
208	D0		
209	D1	J	
210	D2	K	
211	D3	L	
212	D4	M	
213	D5	N	
214	D6	O	
215	D7	P	
216	D8	Q	
217	D9	R	

218	DA		
219	DB		
220	DC		
221	DD		
222	DE		
223	DF		
224	E0		
225	E1		
226	E2	S	
227	E3	T	
228	E4	U	
229	E5	V	
230	E6	W	
231	E7	X	
232	E8	Y	
233	E9	Z	
234	EA		
235	EB		
236	EC		
237	ED		
238	EE		
239	EF		
240	F0	0	Zero
241	F1	1	One
242	F2	2	Two
243	F3	3	Three
244	F4	4	Four
245	F5	5	Five

246	F6	6	Six
247	F7	7	Seven
248	F8	8	Eight
249	F9	9	Nine
250	FA		
251	FB		
252	FC		
253	FD		
254	FE		
255	FF		

7.4 ASCII

In small computer systems and the Internet, the most popular system for coding alphabetic characters is the American Standard Code for Information Interchange or ASCII. Originally put to work when serial communications was the common link between computers and terminals, and seven bit words were commonplace, the basic ASCII code uses seven bits to encode all the letters of the alphabet, numerals, punctuation marks, and a range of message formatting codes. In PC storage, of course, a byte is the standard unit of measure, and adding a bit doubles the range of symbols that the ASCII code can identify. Many eight-bit elaborations of the basic ASCII code have consequently been developed.

The basic 128 characters are generally inviolate. The first 32 characters are reserved as control codes, instructions that tell data processing equipment how to handle the data. Alphabetic characters are stored in two ranges, from 65 through 90 for the capital letters "A" through "Z" and from 97 to 122 for lower case "a" through "z." The two ranges work neatly together because the codes for a specific capital and lowercase letter will always differ by only one bit. Adding 20 (Hex) to the code of a capital letter results in the code for its lowercase equivalent. The numerals run from 48 (representing zero) to 57 (representing nine).

Table 12. Lists the basioc seven-bit ascii code.

Decimal	Hex	Symbol	Mnemonic or function
0	0	^@	NUL (Used as a fill character)
1	1	^A	SOH
2	2	^B	STX
3	3	^C	ETX
4	4	^D	EOT
5	5	^E	ENQ
6	6	^F	ACK
7	7	^G	BEL
8	8	^H	BS
9	9	^I	HT
10	A	^J	LF
11	B	^K	VT
12	C	^L	FF
13	D	^M	CR
14	E	^N	SO
15	F	^O	SI
16	10	^P	DLE
17	11	^Q	DC1
18	12	^R	DC2
19	13	^S	DC3
20	14	^T	DC4
21	15	^U	NAK
22	16	^V	SYN
23	17	^W	ETB
24	18	^X	CAN
25	19	^Y	EM
26	1A	^Z	SUB

27	1B	^[	ESC
28	1C	^\	FS
29	1D	^]	GS
30	1E	^^	RS
31	1F	^_	US
32	20	SP	Space character
33	21	!	Exclamation mark
34	22	"	Double quotes
35	23	#	Pound sign
36	24	$	Dollar sign
37	25	%	Percent sign
38	26	&	Ampersand
39	27	'	Single quote
40	28	(	Open parenthesis
41	29	)	Close parenthesis
42	2A	*	Asterisk
43	2B	+	Plus sign
44	2C	,	Comma
45	2D	-	Minus sign (hyphen)
46	2E	.	Period
47	2F	/	Slash
48	30	0	Zero
49	31	1	One
50	32	2	Two
51	33	3	Three
52	34	4	Four
53	35	5	Five
54	36	6	Six
55	37	7	Seven
56	38	8	Eight

57	39	9	Nine
58	3A	:	Colon
59	3B	;	Semi-colon
60	3C	<	Less than sign
61	3D	=	Equals sign
62	3E	>	Greater than sign
63	3F	?	Question mark
64	40	@	At sign
65	41	A	
66	42	B	
67	43	C	
68	44	D	
69	45	E	
70	46	F	
71	47	G	
72	48	H	
73	49	I	
74	4A	J	
75	4B	K	
76	4C	L	
77	4D	M	
78	4E	N	
79	4F	O	
80	50	P	
81	51	Q	
82	52	R	
83	53	S	
84	54	T	
85	55	U	
86	56	V	

87	57	W	
88	58	X	
89	59	Y	
90	5A	Z	
91	5B	[	Open bracket
92			
5C	\	Backslash	
93	5D	]	Close bracket
94	5E	^	Caret
95	5F	_	Underscore
96	60	`	
97	61	A	
98	62	B	
99	63	C	
100	64	D	
101	65	E	
102	66	F	
103	67	G	
104	68	H	
105	69	I	
106	6A	J	
107	6B	K	
108	6C	L	
109	6D	M	
110	6E	N	
111	6F	O	
112	70	P	
113	71	Q	
114	72	R	
115	73	S	

116	74	T	
117	75	U	
118	76	V	
119	77	W	
120	78	X	
121	79	Y	
122	7A	Z	
123	7B	{	Open curly bracket
124	7C	|	Bar
125	7D	}	Close curly bracket
126	7E	~	Tilde
127	7F		

You're likely to run into two different sets of eight-bit extensions to the ASCII code. When working with DOS, you're most likely to use the IBM extended character set that puts many of the extra codes to work specifying additional symbols that are often used for drawing block graphics on monitors and printed output. Windows has its own Windows extended character set that omits the block graphics (after all, they are hardly necessary for an interface built around bit-mapped graphics) and instead includes more foreign language characters and symbols.

7.5 UNICODE

The eight-bit ASCII code cannot handle all the characters and symbols used by all languages world-wide. Some languages have thousands of distinct characters. If the PC is to be useful throughout the world, it requires some means of accommodating a wider range of characters. The UniCode Worldwide Character Standard was designed to bridge this language gap. By using a 16-bit code for individual characters, UniCode has the potential to encode 65,536 distinct symbols. The downside is, of course, any program must reserve twice the space to stored individual characters.

UniCode has been incorporated into the latest operating system designs and their file systems. Directory entries in new file systems, for example, make allowances for 16-bit character entries.

UniCode makes a distinction between characters and glyphs. Under the UniCode definition, a glyph is the visual representation of a character. The character is the underlying concept, the understood meaning of the symbol. A glyph is what prints; the character is what you understand the glyph to mean. According to the UniCode design, a character has no inherent image of its own. A font, under this definition, therefore, is a collection of glyphs rather than characters.

UniCode is also language neutral. Although it encodes the symbols used by many different languages, merely examining a list of the characters used does not in itself reveal what language is being encoded. UniCode requires a higher level protocol to define the language being encoded.

In its current version, 2.0, UniCode includes characters not only for most major language writing systems in use in the world today[md]a total of 25 different scripts[md]but it also includes symbols for classical and historic languages. A total of 38,885 characters are currently defined for use with languages in Africa, Asia, Europe, the Middle East, North and South America, and Oceana.

Even at this, the current version is not definitive. The symbol needs for some language systems are still being defined and eventually will be accommodated into future versions.

The breadth of the code is so large the complete standard requires its own book, The UniCode Standard, Version 2.0 Addison-Wesley, 1996. ISBN 0-201-48345-9.

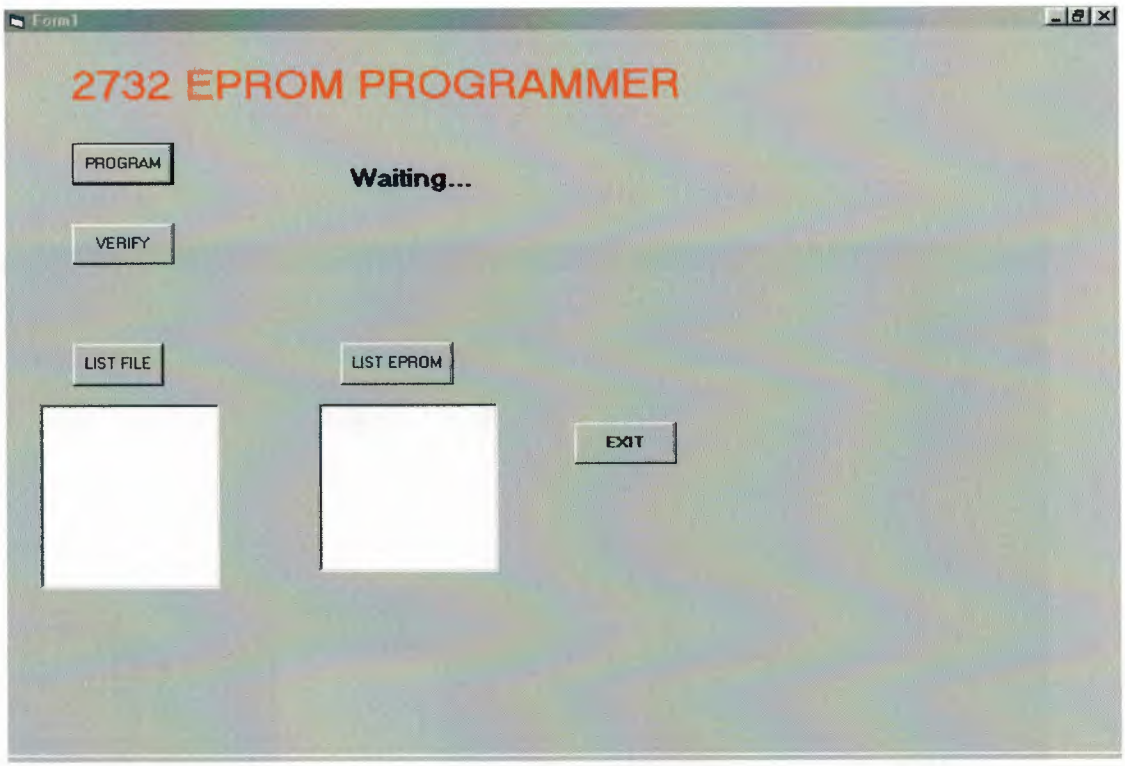
8.EPROM PROGRAM

```
Private Sub Command1_Click()
'Ilk olarak EPROM bosmu degılmı bul
For i = 0 To 8191
   If j <> 255 Then
      MsgBox "EPROM not blank"
      GoTo ext
   End If
Next i
' Bu noktada EPROM bostur
```

```
'Programlamaya basla

'adres 0 gonder
'bilgi 0 gonder
'50 ms pulse gonder
Sleep (50)      ' 50 ms gecikme

ext:
End Sub

Private Sub Command2_Click()
'EPROMu oku, FILE oku ve ikisini mukayese et.
'eger ayni degilse, hata var ve dur (MSGBOX)
End Sub

Private Sub Command3_Click()
'epromu listele
' adresleri gonder 0 dan 8191 e kadar.  Her adres icin
' data yı oku ve LIST2 de goster

'PORT A ve B output  PORT C input
vb_out &H303, &H89

For i = 0 To 31
   vb_out &H301, i
   For k = 0 To 255
      vb_out &H300, k
      j = vb_in(&H302)
      List2.AddItem Hex(j)
   Next k
Next i

End Sub
```

```
Private Sub Command4_Click()
End
End Sub

Private Sub Command5_Click()
Dim S As String
Dim i, j As Integer

Open "c:\eprom.txt" For Input As #1
While Not EOF(1)
   Input #1, S
   List1.AddItem S
Wend

Close #1
End Sub

Private Sub Form_Load()
Dim base, outdat As Integer

address = 0
Form1.WindowState = vbMaximized
Label2.Caption = "Waiting..."
base = &H303
outdat = &H80
vb_out &H303, &H80
j = 0
End Sub

Private Sub Timer1_Timer()
If j = 0 Then
   vb_out &H300, 1
   j = 1
Else
```

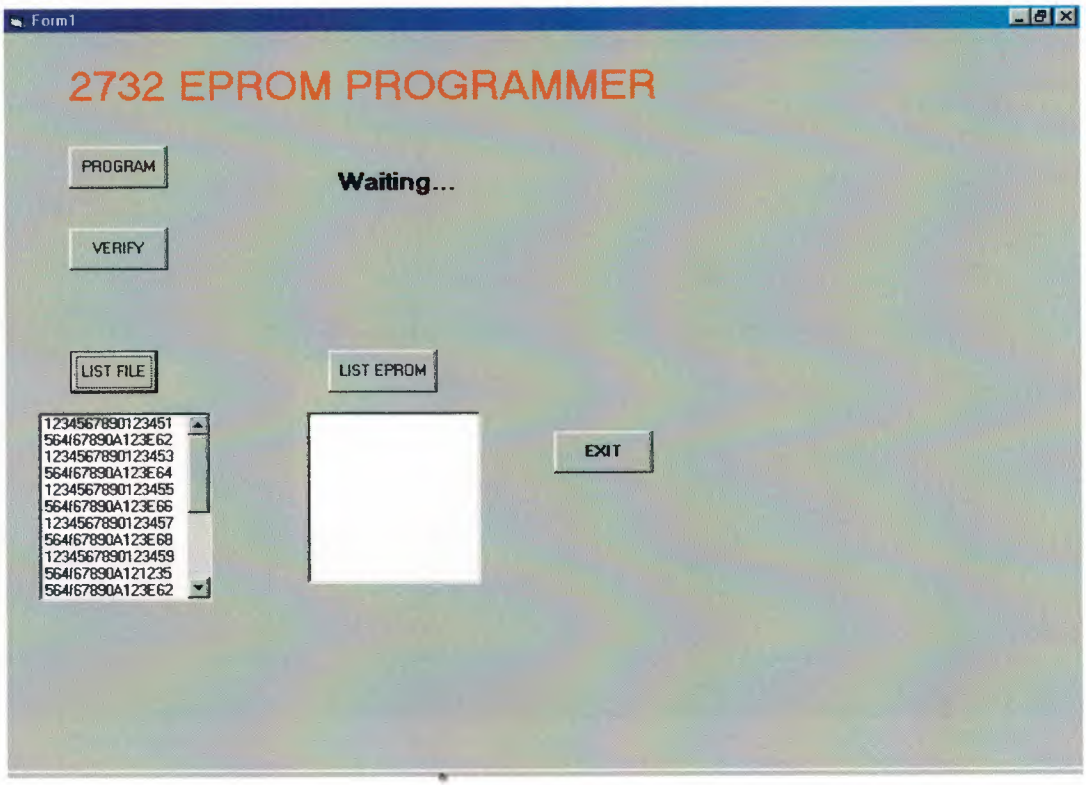
```
    vb_out &H300, 0
    j = 0
End If

End Sub
```



Main menu 2764 eprom

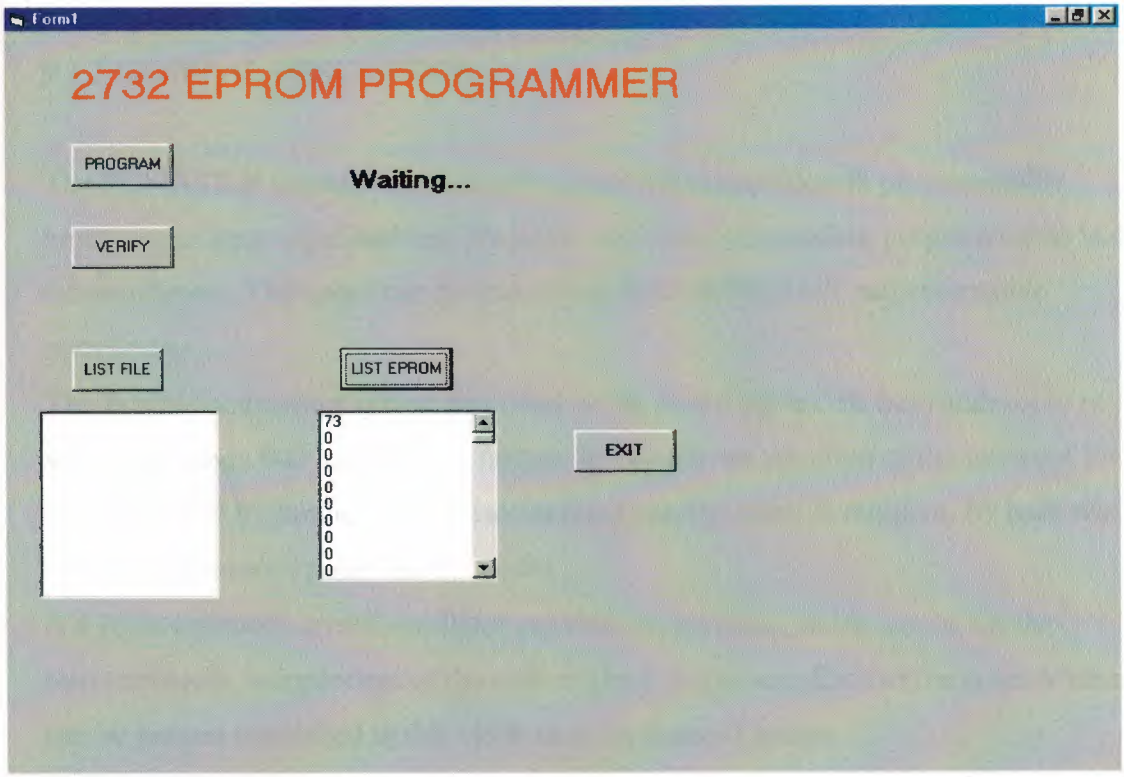
There are five buttons in the main menu. In the list file button we will enter the new eproms, but there are hexdecimal numbers. There are too many file there. But we put one of them. If we chose one of file then we load it to eprom.

When we load the file than we will verify them to correct it or not. If we chose the exit buttom we will stop the program.



List file button

When we press the list file button, if we loading any file to list file it will list all of it. There are to many files in list file button. In that file when we need a file and than we will load any file there.



List eprom form

Eprom listing button is use for eprom listing .We send from 0 start by 8191 addressing all for adress read detay and showing the eprom list box.

9. MODEL PC14AT AND PC14LP (LOW POWER) PROGRAMMABLE DIGITAL INPUT/OUTPUT AND COUNTER/TIMER BOARD

9.1 GENERAL INFORMATION

9.1.1 GENERAL DESCRIPTION

The PC14ATILP is a half sized plug-in board which provides 48 programmable input/output lines organised into six ports, and three independent programmable 16 bit counter/timers. The board can be installed in IBM PCIXTIA T and compatible computers.

The flexible addressing system provided on the board allows the base address to be set within the range 000 to FFO_{16}. A further facility allows selection of the interrupt levels IRO2 to IRO7 by jumper, or if a user-defined configuration is required, by hard wiring to particular control points on the board.

A 4 MHz on-board crystal oscillator provides an accurate, stable source for the counter/timers, independent of the system clock frequency. Each of the counter/timers can be jumper connected to this clock or to an external source.

Sixteen LEDs are mounted on the PC14AT (not PC14LP) board to continuously indicate the state of each bit of the A ports of both 82C55 Programmable Peripheral Interface (PPI) ICs.

This manual also covers the operation of PC 14LP (909 040 67) which is the low power version of PC 14AT. The only major operating difference between the two boards is that the LEDs and their respective drivers are not fitted to the PC14LP .The power requirement of the PC14lP is reduced to 5 V c 55 mA (PC14AT requires 175 mA).

9.1.2 FEATURES

48 programrnable input/output lines, TTL cornpatible .3 control modes for the programrnable I/O ports .3 independent programrnable counter/timers .6 control modes for the counter/timers .Count rate up to 5 MHz, binary or BCD .4 MHz on-board crystal oscillator

Flexible addressing and interrupt selection

LED display of individual bits (PC14AT only) .Socketed I/O devices for easy repair

Low power (LP) version available for portable applications

9.1.3 WHAT THE PACKAGE CONTAINS

The package as delivered from Amplicon Liveline Ltd. contains:

1. The plug-in board, PC36AT or PC36LP as ordered, in a protective bag When removing the board, observe the precautions outlined in paragraph 2.1. The board type is identified by a marker in the appropriate position to indicate. The model.
2. The included distribution software on a high density 3 1/2 inch diskette (Amplicon Part NQ 868 931 88).

NOTE: Soflware items are supplied on 3 1/2 inch diskettes. Please request copies on 5 1/4 inch diskettes if this size is more convenient.

3. This Instruction Manual.

Any additional accessories (maling conneclors, optional software, applications notebook elc.) may be packed separately.

9.1.4 CONTACTLNG AMPLLCON LIVELINE LIMITED FOR SUPPORT OR SERVICE

The PC14AT and PC14LP boards are designed and manufactured by Amplicon Liveline Limited and maintenance is available throughout the supported life of the product.

9.1.4.1 TECHNICAL SUPPORT

Should the PC14AT/LP board appear defective, please check the information in this manual and any 'Help' or 'READ.ME' files appropriate to the program in use to ensure that the product is being correctly applied.

If an application problem persists, please request Technical Support on one of the following numbers.

Telephone. UK 01273 608 331

Inlernational	+44 1273 608 331
Fax: UK	01273 570215
International	+44 1273 570 215
CompuServe Mailbox 1D	100144,1403
Inlemel	100144.140314 COMPUSERVE.COM

9.1.4.2 REPAIRS

If the PC14ATfLP board requires repair then please return the goods enclosing a repair order delailing the nature of the faut. If the board is still under warranty, which period is 12 months from the date of shipment, there will be no repair charge.

For traceability when processing returned goods, a Returned Materials Authorisation (RMA) procedure is in operation. Before returning the goods, please request an individual RMA number by contacting Amplicon Customer Services by telephone or fax on the above numbers.

Give the reason for the return and, if the goods are still under warranly, the original invoice number and date. Repair turnaround time is normally five working days but the Service Engineers will always try to co-operate if there is a particular problem of time pressure.

Please mark the RMA number on the outside of the packaging to ensure that the package is accepted by the Goods Inwards Department.

Address repairs to: Cuslomer Services Department AMPLICON LIVELINE LIMITED Cenlenary Industrial Eslale Brighlon, East Sussex BN2 4AW England

9.2 GETTING STARTED

9.2.1 GENERAL INFORMATION

The PC14ATILP provides 48 lines of user programmable digital input/output and three 16 bit counter/timers. The board is supplied complete with demonstration software written in Borland Turbo Pascal and BASICA (GWBASIC). The source code for the Turbo Pascal program is supplied, and is compatible with versions 4 and above. The source code for BASICA is also supplied, which is compatible with GWBASIC and other variants. A copy of the language will be needed if the user wishes to edit the BASIC code, but as it is saved in ASCII format, the code may be listed by any word processor.

9.2.1 INSTALLING THE PC14AT/LP BOARD

To install the peripheral card in the host computer, please refer to the hardware manual supplied with the machine for instructions on how to remove the cover and install devices into the I/O channel expansion slots. However, ALWAYS SWITCH OFF the power before installing or removing a device. The PC14AT/LP may be located in any available slot in the machine provided that there is space to provide extemal access for the I/O ribbon cables and no restriction is placed on that slot by the manufacturer of the machine.

⚠ CAUTION

Some of the components on the board are susceptible to electrostatic discharge, and proper handling precautions should be observed. As a minimum, an earthed wrist strap must be worn when handling the PC14AT/LP outside its protective bag.

Full static handling procedures are defined in British Standards Publication BS5753.

When removed from the bag, inspect the board for any obvious signs of damage and notify Amplicon if such damage is apparent. Do not plug a damaged board into the host computer. Keep the protective bag for possible future use in transporting the board.

9.2.2 BACKING UP THE DEMONSTRATION SOTTWARE DISK

It is important that a backup copy of the supplied disk is made, and the original stored in a safe place. The software can be copied onto another blank disk by using the MS-DOS command.

DISKCOPY A: A:

on a single drive machine, or

DISKCOPY A: B:

on a dual drive machine.

Always use the copy for your work.

9.2.3 INSTALLING THE SOFTWARE ON A HARD DISK

If the software is to be installed on a fixed hard disk, make sure that you are logged on to drive C: and use the DOS command

MD\PC14AT

to make a suitable directory on drive C:, then change to this directory by typing

CD\PC14AT <RETURN>

Put the PC14AT software disk in drive A: and then type

COPY A:*.* C: <RETURN>

This will copy all the files into the newly made PC14AT directory.

9.2.4 LOADING THE PROGRAM

1. Boot up
2. Log on to the drive and directory containing the PC14AT demo and enter PC14AT to load the Pascal demo. The BASIC demo is PC14AT.BAS and can be run in BASICA or GWBASIC
3. Follow the Menu driven instructions on the screen
4. Type the READ.ME file for the latest information on this demo program

9.3. INPUT/OUTPUT SPECIFICATIONS

Input and output are via the 82C55 Programmable Peripheral Interface and the 82C53 Counter/Timer ICs. The appendices provide complete specifications for these devices. Brief input/output and related specifications are:

The 82C55 programmable peripheral interface

PPI 82C55 Inputs

'Low' input voltage -0.3 V 10 +08 V) TTL

'High' input voltage +2.2 V 10 +5.3 V) compatible

When an input is left open circuit, its high input impedance makes it susceptible to static charge and interference and the state may be indeterminate. Ensure that signals to any input lines are within the above limits with unused input lines grounded or masked out in software .

PPI82C55 Outputs

'Low' oulpul voltage +0.4 V max at +2.5 mA) TTL

'High' oulpul voltage +3.7 V min at -2.5 mA) compatible

The 82C53 counter/timer (cmos programmable interval timer)

82C53 Inputs (Clock and Gat8 Inputs 0-2)

'Low' inpul vollage -0.3 V to +0.8 V) TTL

'High' input vollage +2.2 V to +5.3 V) compatible

82C53 Outputs (OUT 0 -OUT 2)

'Low' output voltage +0.45 V max al +4 mA) TTL

'High' output voltage +3.7 V min at -1 mA) compatible

The crystal oscillator

Frequency 4 MHz

Frequency tolerance +- 50 ppm at 25°C

Frequency stability +- 50 ppm over 0 to +60°C

DC output voltages

The power rails of the host PC are available on the I/O connectors. Subject to the constraints of the host power supply , the following currents can be drawn from these connector outputs.

+12 VDC at 50 mA

-5 VDC at 50 mA

+5VDC at 100mA

-12VDC at 50mA

it is important that any connected external devices operate within the above limits.

9.4. USER SETTINGS

9.4.1 BOARD BASE ADDRESS

The PC14AT/LP can have its base address situated within the range 000_{16} to FFO_{16}. This feature provides the fiexibility to avoid any contention in I/O mapping that may arise with some clones and allows the use of muitiple cards fitted in the PC expansion slots.

Factory setting

The board's base address is sel at the factory to be 300_{16}

Customer configured base address

The board's base address can be selected as any sixteenth address within the range 000_{16} to $FF0_{16}$ by means of the appropriate settings of switch SW1. This switch bank comprises a row of eight single-pole, single-throw switches wilh each 'up' or 'ON' position selecting a logic 0, and each 'down' or 'OFF' position selecting a logic 1. The most significant hex digit is coded by the four most left switches and the middle hex digit is coded by the four most right switches of SW1 .The least significant hex digit is always 0.

Figure 4.1 below shows the factory default setting of 300_{16}

Most significant digit 0011 = 3_{16}

Middle digit 0000 = 0_{16}

Least significant Default =0_{16}

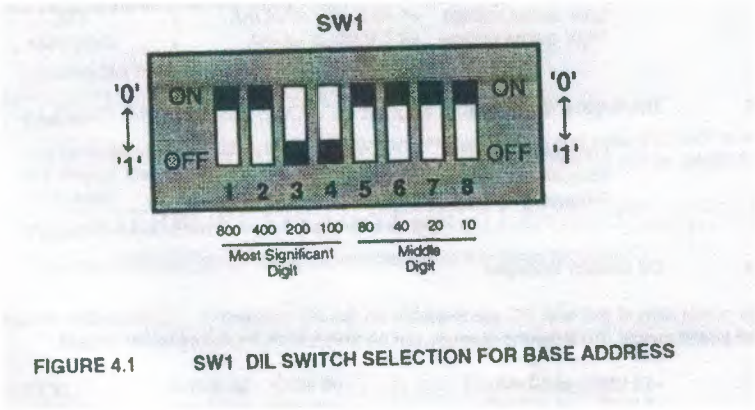


Fig 16.Switch selection for base address

9.4.2 INPUT/OUTPUT ADDRESS SPACE USED BY THE PC14AT/LP

The table in figure 4.2 shows the location of the PPI and counter/timer registers in the I/O space above the base address.

ADDRESS	FUNCTION
BA + 00_{16}	QA2 82C55 PPI - 1 Port A
BA + 01_{16}	QA2 82C55 PPI - 1 Port B
BA + 02_{16}	QA2 82C55 PPI - 1 Port C
BA + 03_{16}	QA2 82C55 PPI - 1 Control Register
BA + 04_{16}	QA3 82C55 PPI - 2 Port A
BA + 05_{16}	QA3 82C55 PPI - 2 Port B
BA + 06_{16}	QA3 82C55 PPI - 2 Port C
BA + 07_{16}	QA3 82C55 PPI - 2 Control Register
BA + 08_{16}	QA7 82C53 Counter/Timer 0
BA + 09_{16}	QA7 82C53 Counter/Timer 1
BA + $0A_{16}$	QA7 82C53 Counter/Timer 2
BA + $0B_{16}$	QA7 82C53 Control Register

FIGURE 4.2 ADDRESS SPACE FUNCTIONS

Fig 17.Address space functions

Noles to figure 17:

1. BA=Base Address as sel on SW1.
2. Regisler address is Base Address plus offset.
3. Offset to be added to the BA is stated as a Hex value.

9.4.3 INTERRUPT REQUEST (IRO) LEVEL SELECTION

A single interrupt is available on a selected level by jumper selting. Further user delined interrupts can be hard-wired on the board, see paragraph 4.3 3.

The patchable interrupt is connected via a non-inverting buffer, from port C3 01 PPI -1 (0A2) and can be used in two ways.

1. With PPI -1 in mode 1 or 2, port C3 can be used to interrupt the PC when input on port A is requesting service. See the Appendix 82C55 data sheet for full details of usage. The handshaking lines ISTBA (por1 C4) and IBFA (port C5) are available on I/O pins PL1-24 and PL1-23 respectively.
2. With PPI -1 in mode 0 and port C3 programmed as input, the input line to port C3 on I/O pin PL1-29 can be used for a buffered external inlerrupt to the PC.

IRQ factory setting

The factory setting for the patchable interrupt level is IRQ7, but note that this level may be in use by a line printer (LPT) connected to the computer. This IRQ 7 position is illustrated in the IRQ Level jumper setting diagram in figure 4.3

IRQ level jumper setting

An interrupt level must be chosen that is not otherwise used in the system. Note that interrupts cannot normally be shared, and if the interrupt is not specifically supported by the software in use, then leave all IRO levels free as a connection may conflict with others on the PC bus.

To assist the user in choosing a suitable interrupt level. the following table shows the available levels on the PC314AT/LP and normal usage of all *PC/AT* hardware interrupts.

Table 13

PC14AT/LP Jumper 1	IRQ Name	Interrupt Number	Usage Description	
–	0	8	Timer)	Not available on the
–	1	9	Keyboard)	Bus Connectors
2	IRQ2	A	IRQ 8 - 15	
3	IRQ3	B	COM2 or SDLC	
4	IRQ4	C	COM1 or SDLC	
5	IRQ5	D	LPT	
6	IRQ6	E	Floppy Disk	
7	IRQ7	F	LPT	
–	IRQ8	70	Real Time Clock)	
–	IRQ9	71	Re-directed to IRQ2)	
–	IRQ10	72	Unassigned)	Not available on
–	IRQ11	73	Unassigned)	the PC36AT/LP
–	IRQ12	74	Unassigned)	
–	IRQ13	75	Co-processor)	
–	IRQ14	76	Hard Disk)	
–	IRQ15	77	Unassigned)	

The interrupt source (Port C3) is buffered and permanently connected to the upper row of the jumper patch block J5. The six pins on the other row of jumper block J5 are connected to interrupt levels IRO2, IRO3. IRO4. IROS, IRO6 and IRO7 respectively. The required IRO level for the interrupt signal is therefore easily selected by positioning a single jumper on block J5 The factory default setting is IRO level 7 as illustrated in figure 18.

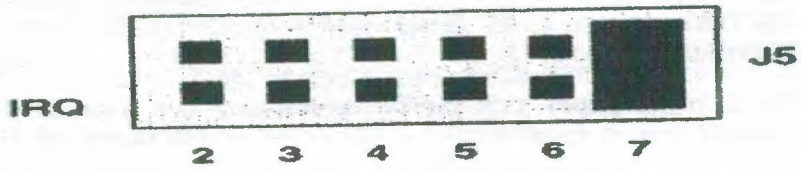


FIGURE 4.3 IRQ LEVELS

Fig 18 IRQ levels

User-Defined interrupts

Normal interrupt usage is as described before in paragraph but for special applications, tenninal points connected to the appropriate Port C control lines of the 82CSS PPIs and the OUT lines of the 82CS3 Counter/Timers enable the user to select interrupts at the required IRQ level for these devices.

The selection is made by hard wiring on the board which requires soldering a wire to the particular tenninal point and wire wrapping it to the requisite IRQ post at the lower row of jumper block J5. For example, to connect Counter/Timer 2 Output to interrupt level IRQ5, a connection has to be made between tenninal point TP6 and the post labelled IRQ5 on the lower row of jumper block J5. (No jumper in position IRQ5).

The terminal points and their functions are listed in the table shown in figure 4.4 below.

Connection	Interrupt	Availability
Jumper J5	PPI - 1 Port A/C3 External Interrupt PL1-29	PPI-1 modes 1 and 2 PPI-1 PC3 mode 0 Input
Terminal TP1	PPI - 2 Port B/C0 External Interrupt PL2-26	PPI-2 mode 1 PPI-2 PC0 mode 0 Input
Terminal TP2	PPI - 2 Port A/C3 External Interrupt PL2-30	PPI-2 modes 1 and 2 PPI-2 PC3 mode 0 Input
Terminal TP3	PPI - 1 Port B/C0 External Interrupt PL1-25	PPI-1 mode 1 PPI-1 PC0 mode 0 Input
Terminal TP4	Counter/Timer 0 OUT0	All modes
Terminal TP5	Counter/Timer 1 OUT1	All modes
Terminal TP6	Counter/Timer 2 OUT2	All modes

FIGURE 4.4 INTERRUPT REQUEST LEVELS IRQ2 - 7 INTERCONNECTIONS.

Fig 19.Interrupt request levels IRQ2 –7 interconnections

Noles to figure 19:

1.The Iower pins of jumper block J5 are those connected to the IRO lines.

2. TP7 is a test point and is not an interrupt connection.

3. AII six interrupt lines may be used subject to no conflict in the system. Level IRQ2 has the highest and IRQ7 the Iowest priority.

4. The user defined interrupts are not buffered and care should be exercised in applying external inlerrupts. These points are provided for flexible usage and tests have shown satisfactory operation with a variety of host PCs. The user should confirm proper operation with the PC in use.

9.4.4 COUNTER/TIMER CLOCK SELECTIONS

The 4 MHz on-board clock signal is available at jumper blocks J1 , J2 and J3 for patching to the Counter/Timer Clock inputs, this signal also being available as an output on PL2/5. The opposite posts on each of these jumper blocks are brought out to pins on the rear connector, so allowing the alternative of TTL compatible external clock signals to be used.

The clock signal used as the source for a particular Timer/Counler is selected by positioning its jumper to either the lower position for the internal 4 MIIz clock or the **upper** position to input an externally generated clock signal.

Jumper	Counter/Timer	External Clock I/P (Max 5 MHz, TTL comp)	Factory Setting
J1	0	PL1-8	Internal - Lower
J2	2	PL1-11	External - Upper
J3	1	PL1-14	External - Upper

FIGURE 4.5 CLOCK INPUT CONNECTIONS TO THE COUNTER/TIMERS

Fig 20. Clock input connections to the counter/timer

AII outputs, clock inputs and gate inputs of the 82C53 Counter/Timer are available on the user I/O connector PL 1 .The inverted output of counter 0 is also available on PL2 pin 3. External inter-connections can be made to configure the three counters to the user's requirements.

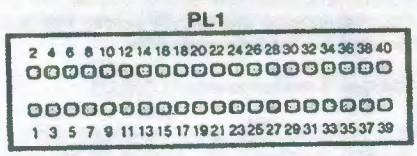
9.4.5 SOCKETED INTEGRATED CIRCUITS

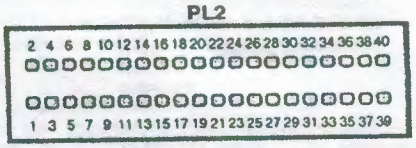
Of all the components on the PC14ATILP board, the unprotected integrated circuits associated with external signals are the most vulnerable to abuse. Incorrect connections, short circuits, transients or excessive signallevels can occur and cause the interface chips to be permanently damaged. Such potential problems are recognised, and the susceptible interface ICs on the PC14AT/LP are mounted in sockets, so that it damage is suspected, the component can be easy replaced. The socketed ICs are
QA2, QA3 82C55 Programmable Peripheral interface
QA7 82C53 Programmable Interval Counter/Timer
Normel precautions should be taken when handling the board for IC replacement. .

9.5. ELECTRICAL CONNECTIONS

9.5.1 USER CONNECTIONS

Inputs and Outputs to the PPI and Counter/Timer devices on the PC14AT/LP are connected via two forty pin IDC headers on the board, PL 1 and PL2. Cables to these connectors are usually ribbon type and can be folded and dressed to pass through the slot provided in the rear mounting bracket. Be careful to observe pin numbering, and to plug these un-polarised connectors correctly. Pins 1 and 40 are labelled for each connector on the board.





Pin NO	PL1	Pin NO	PL1
1	Ground	21	PPI-1 PC7
2	Ground	22	PPI-1 PC6
3	PPI-1 PA3	23	PPI-1 PC5
4	N/C	24	PPI-1 PC4
5	PPI-1 PA2	25	PPI-1 PC0
6	PPI-1 PA1	26	PPI-1 PC1
7	PPI-1 PA0	27	PPI-1 PC2
8	Counter 0 CLK0	28	PPI-1 PB7
9	Counter 0 OUT0	29	PPI-1 PC3
10	Counter 0 GAT0	30	PPI-1 PB6
11	Counter 2 CLK2	31	PPI-1 PB0
12	Counter 2 OUT2	32	PPI-1 PB5
13	Counter 2 GAT2	33	PPI-1 PB1
14	Counter 1 CLK1	34	PPI-1 PB4
15	Counter 1 GAT1	35	PPI-1 PB2
16	Counter 1 OUT1	36	PPI-1 PB3
17	PPI-1 PA4	37	–5 Volts Out
18	PPI-1 PA5	38	+5 Volts Out
19	PPI-1 PA6	39	–12 Volts Out
20	PPI-1 PA7	40	+12 Volts Out

Pin NO	PL2	Pin NO	PL2
1	Ground	21	PPI-2 PC6
2	Ground	22	PPI-2 PC7
3	Counter 0 /OUT0	23	PPI-2 PC4
4	N/C	24	PPI-2 PC5
5	4 MHz Clock O/P	25	PPI-2 PC1
6	N/C	26	PPI-2 PC0
7	N/C	27	PPI-2 PB7
8	N/C	28	PPI-2 PC2
9	N/C	29	PPI-2 PB6
10	N/C	30	PPI-2 PC3
11	N/C	31	PPI-2 PB5
12	N/C	32	PPI-2 PB0
13	PPI-2 PA1	33	PPI-2 PB4
14	PPI-2 PA0	34	PPI-2 PB1
15	PPI-2 PA3	35	PPI-2 PB3
16	PPI-2 PA2	36	PPI-2 PB2
17	PPI-2 PA5	37	–5 Volts Out
18	PPI-2 PA4	38	+5 Volts Out
19	PPI-2 PA7	39	–12 Volts Out
20	PPI-2 PA6	40	+12 Volts Out

FIGURE 5.1 INPUT/OUTPUT CONNECTORS PL1 AND PL2 PIN DESIGNATIONS

Note to Figure 5.1 : The 82C55 input/output lines are designated as follows:

PPI - 1 refers to 82C55 QA2
PPI - 2 refers to 82C55 QA3
PA refers to Port A
PB refers to Port B
PC refers to Port C
0 to 7 is the particular line on the designated port.

Thus PPI-1 PA0 refers to 82C55 QA2 Port A line 0.

Fig 21.Input/Output connectors PL1 and PL2 pin designations

Optional cables and screw terminal assemblies are available from Amplicon Liveline **Ltd** to simplity making the connections to the PC14AT/LP .Two of each part are required and the parts are:

908 919 47 40 way ribbon cable One metre long with IDC connectors 908 919 46 40 screw terminal assembly for DIN rail mounting

9.5.2 MAIN 110 BUS BACK-PLANE CONNECTIONS

Connection to the computer is made through the I/O channel main bus 62 Pin Connector. (Pins B1 and A1 are at the bracket end of the board). Pin connections are shown below. For further information please consult the technical reference manual for the host computer.

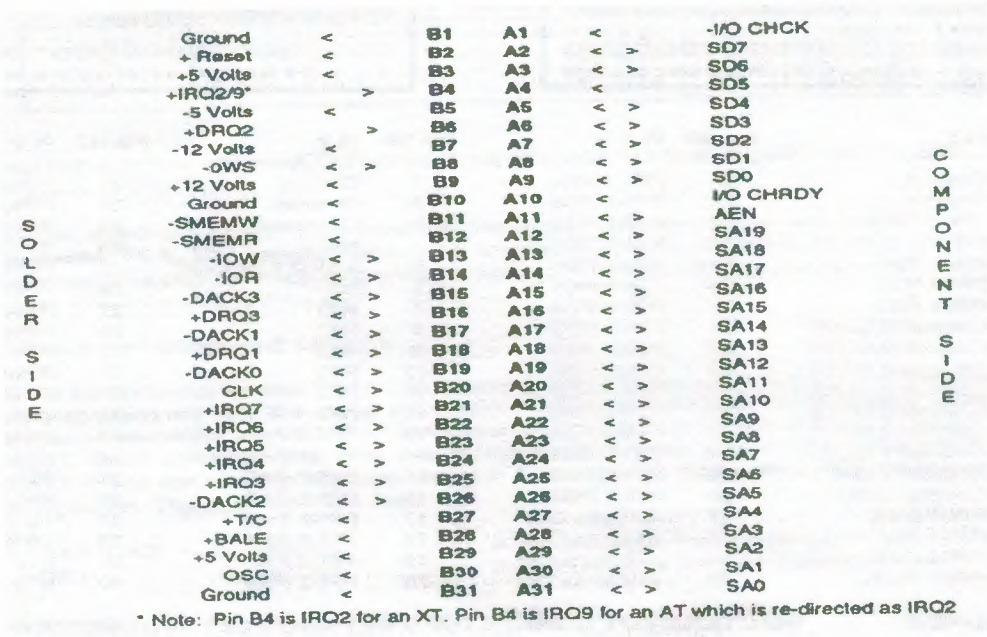


FIGURE 5.2 MAIN PC BUS BACK-PLANE CONNECTOR PIN ASSIGNMENTS

Fig 22.Main pc bus black-plane connector pin assigments

9.6. PROGRAMMING

9.6.1 DEMONSTRATLON DISK

The PC14AT or PC14LP is supplied with a demonstration diskette containing several routines

written in BASICA and Borland Turbo Pascal. The four files on the disk are:

PC14AT.BAS

PC14AT.PAS

PC14AT.EXE

READ.ME

Basic Demonstration program

The BASIC demonstration program is written in the IBM standard BASICA language, and saved in ASCII format. The program can be run in BASICA, GWBASIC or other 100% compatible BASIC. minor changes will be required to run in QuickBASIC.

To assist those who wish to modify and enter the program in another version of BASIC, the program is in ASCII format, so allowing it to be 'typed' or listed using any text editor.

To run the demonstration program:

1. Load the BASIC interpreter .
2. Loada and run "PC14AT".

Pascal Demonstration program

Both source (PC14AT.PAS) and executable (PC14AT.EXE) versions of the demonstration program are provided on the disk.

To run the Pascal version of the demonstration program.

1. Boot up.
2. Log on to the disk drive or directory containing the demonstration software and type PC14AT <RETURN>.

Running the Demonstrations

The BASIC and Pascal versions of the demonstrations are very similar in their operations and are run in the same way.

The first screen asks the user to enter the Base Address selected for the board, and this address must be the same as that set up on switch SW1. Having done this, the user is then presented with a menu from which the required demonstrations can be selected The four menu items comprise:

1. All ports output a square wave.
2. BCD counler (Can be viewed on Ihe LEDs, if filted). 3. Wave pattern (Can be viewed on the LEDs, if filted). 4. Relurn 10 DOS,

9.6.2 COPYRIGHT

The software on the demonstration disk is copyright Amplicon liveline Ltd. Any user who has purchased a PC14AT/LP may use the software, or any part of it, for use in his own programs, or for resale when delivered wilh a PC14AT or PC14LP.

9.6.3 PROGRAMMING THE 82C53 COUNTERRRIMERS

The 82C53 Programmable Interval Timer provides three 16 bit counter/timers which can be independently programmed to operate in any one of six modes with BCD or Binary count functions

The six modes are:

1.Mode 0: Inlerrupt on Terminal Count.

2.Mode 1: Programmable One Shot.

3.Mode 2: Rate Generator.

4. Mode3: Square Rale Generator.

5.Mode4: Software Triggered Strobe.

6. Mode5: Hardware Triggered Strobe.

More detailed information on the operation of these various modes is contained in the data sheet shown in Appendix 82CS3.

82C53 control word

The function of a particular 16 bit timer/counter is established by writing an 8 bit control word 10 the control register, The control word is followed by the count value selling and execution of the desired operation

This 8 bit control word consists of four fields as follows.

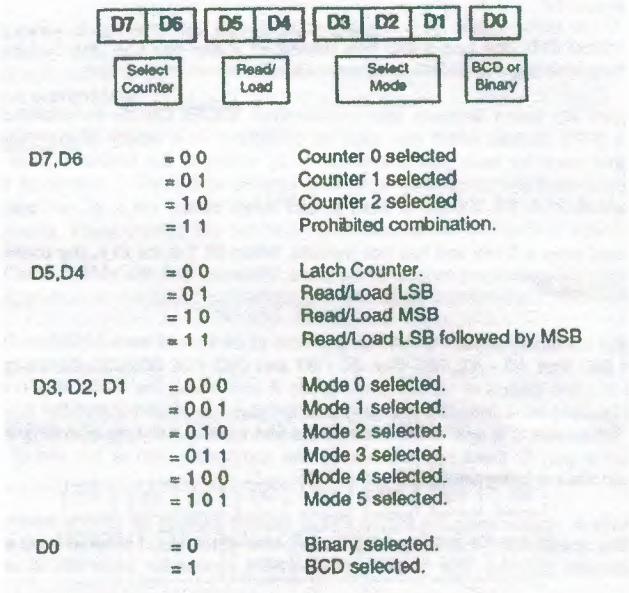


Fig 23

82C53 PROGRAMMING EXAMPLES

Example 1

To select Counter 1 to Mode 3, loading/reading low order byte followed by high order byte in binary , the control word is:

0 1 1 1 0 1 1 0 = 76_{16}.

This value has to be loaded to the control register whose address is Base Address + B_{16}. Assuming that the board base address is 300_{16}, the BASICA statement OUT &II30B, &H76 will load the control register with 76_{16}.

The value of the count has now to be loaded to the counter. The address of Counter 1 is base address +9_{16} which, in our example would be 309. To load the value 50 decimal to this counter the BASICA statement OUT &II309, &II32 .OUT &II309, &II00 is used it should be noted that both the low order and high order bytes have to be loaded even though the high order byte, as in the above example, is zero.

Example 2

To read the current count on Counter 1 without affecting the counting operation the counter has to be latched To do this the control word 0 1 0 0 0 1 1 0 (46_{16}) is loaded to the control register by OUT &H308. &H46. The two bytes then have to be read from

the latch using the command IN &II308 to read the low order byte followed by IN &H308 to read the high order byte. The two byte. The two bytes MUST be read before attempling 10 execule another OUT instruclion on the same counter.

9.6.4 PROGRAMMING THE 82C55 PROGRAMMABLE PERIPHERAL INTERFACES

This digital I/O board features two uncommitted 82C55 CMOS Programmable Peripheral Interface (PPI) devices which can each be configured in a variety of operating modes. The operational mode for each port is eslablished by writing to the control register of the 82C55. This control word also eslablishes whether the port is configured 10 operate as input, output or bi-directional.

The control word is 8 bits and has two formats. When bit 7 is set to 1, the control word is used to establish the operational modes for the ports. When set to 0, the control word is used to write bits to Port C for status/control purposes.

The digital I/O facilily of Ihe PC14AT/LP provides all 24 lines of each 82C55 on three 8 bit ports, labelled DIO Port AO -A7, DIO Port BO -B7 and DIO Port CO -C7. These ports are further divided into two groups of 12 bits each, group A comprising the 8 bits of port A and the high order 4 bits of port C, and Group B comprising the 8 bits of port B and the low order 4 bits of port C. When port C is split in this mode, note that when a half byte of data is being read from or written to port C, those four bits occupy the appropriale high or low end of a full byte, the other four bits not being used.

Full details of programming the 82C55 PPI at register level, in its various modes, are shown in the device specification in the Appendix 82C55, and examples of programming are given on the demonstration diskette. The three basic operating modes for each 82C55 are summarised below.

Mode 0 (BASIC INPUT/OUTPUT)

This mode is the power-up default with all ports set as input.

In mode 0. the PPI provides simple input/output operations. No control signals are required and the ports defined as input reflect the current state of digital signals on the lines (no latching). The lines of output defined ports are set to zero on the mode change, and when a port is Ioaded, the outputs are tatched to that value

and when a port is loaded,the outputs are latched to that value.
All 24 bits of each PPI can be used for input or output arranged as any combination of two 8 bit ports and two 4 bit ports.

Mode 1 (STROBED INPUT/OUTPUT)

Mode 1 provides I/O operations on Group A and/or Group B each with a simple handshake protocol. In either group, the 4 bit port is used for status and control of the associated 8 bit port. An IRQ facility in this mode is available on the PC14AT/LP
Each 8 bit port can be used uni-directionally for either input or output operations, both being latched.

Mode 2 (STROBED BI-DIRECTIONAL INPUT/OUTPUT)

This mode of operation can be applied to group A only, and provides one 8 bit bi-directional data port and one 5 bit control and status port with IRQ lacility Both input and output operations are latched. Port B can be used in mode 0 or 1 while port A is in mode 2.

Mixed Mode

The ports of the PPI can be programmed to operate in a mixed combination of modes that in some cases leave bits of port C unused for control or status purposes. These unused bits can be programmed for use as inputs or outputs.
Further information on the 82C55 operational modes will be lound in Appendix 82C55.

82C55 Control Word

The function of the ports is established by writing an 8 bit control word to the 82C55 control register. The format of the word is as follows:

D7 = 0 Bit SeVReset active. = 1 Mode definition active.

When D7 is set to 0, any one of the eight bits of port C can be set or reset using a single OUT instruction.

D6,D5 = 0 0 Group A ports selected to Mode 0.

= 0 1 Group A ports selected to Mode 1.

= 1 0 Group A ports selected to Mode 2.

= 1 1 Group A ports selected to Mode 2.

D4 = 0 Port A selected for output.

= 1 Port A selected for input.

D3 = 0 Por1 C upper selected for output

= 1 Port C upper selected for input.

D2 = 0 Group B ports selected to Mode 0.

= 1 Group B ports selected to Mode 1

D1 = 0 Port B selected for output

= 1 Por1 B selecled for input.

DO = 0 Por1 C Iower selected for output

= 1 Port C Iower selected for input

82C55 Programming Examples

In the following examples it is assumed that the base address of the board is $01BO_{16}$ and that 82C55 # 1 (QA2) is being programmed.

Example 1

To select all ports to operate in Mode 0 as output ports the control word to be loaded into the control register is:

1 0 0 0 0 0 0 0= 80_{16}

The BASICA command OUT &H1B3, &H80 will load the control word into the control register of 82C55 # 1.

A further BASIC command such as OUT &H1 BO, & HFF will output the value FF from Port A.

Example 2

To select Group A to Mode 1 as input and Group B to Mode 0 as output the required control word is:

1 0 1 1 1 0 0 0 = $B8_{16}$

Example 3

To select Group A to Mode 2 and Group B to Mode 0 as output requires the control word. 1 1 x x x 0 0 0 = (say) CO_{16} Where x = don't care.

Further examples of control words are contained in Appendix 82C55.

APPENDICES

APPENDIX A.

Wait State Generator Setting

Earlier PC14AT/lP boards incorporated an on-board wait state generator (OA10) to enable ~ to operate

reliabty in a wide range of *PClXTIAT* 286, 386 and 486 machines. The need for this was because some machines, that are otherwise IBM compatible, operated the VQ expansion bus at clock frequencies higher than the accepted 8 MHz. This wait state generator can now be fitted as a special option.

For correct operation at these higher frequencies it becomes necessary to slow down the bus interface signals, locally, on the PC14AT/lP board. The degree of retardation can be adjusted to give optimum performance in any machine. Being local to the PC14AT/LP board, this slowing down in no way impairs the performance of the host computer.

The expansion bus frequency is not necessarily the same as that of the main processor clock. A computer which is specified as a 12, 16 MIlz or higher processor could well have an expansion bus frequency of 8 MHz. Unless explicitly stated in the machine specification there is no easy way to establish the speed of the expansion bus.

If the expansion speed is known, then use the following table to set the appropriate number of wait states. If the bus speed is not known, it is suggested that the number of wait states is left at zero (No jumper). if the PC14AT/LP functions correctly Leave the setting at zero wait states. if operation is erratic, increase the number of wait states. Introducing wait states can cause some machines to hang up, otherwise no harm can be done by setting the number of wait states too high, however the response of the PC14ATILP board will not be optimum.

N° of Wait States	Expansion Bus Speed	J4 Jumper Setting
0	Up to 8 MHz	No Jumper
1	8 to 10 MHz	3
2	10 to 12 MHz	4
3	12 MHz and above	5

Fig 24.

Appendix 82C53 Contents

Sample Manufacturer5 Data Sheets. The 82C53 CMOS Programmable Interval Timer

Appendix 82C55 Contents

Sample Manufacturer5 Data Sheets. The 82C55 CMOS Programmable Peripheralinterface

APPENDIX B

Full circuit diagrams and the layout of the components on the PC14AT and PC14LP printed clrcuit board are given in this appendix.

Drawing Number 893140. PC14AT Schematic

Drawing Number 893143. PC14A T Assembly Details Drawing Number 904066. PC14LP Schematic

Drawing Number 904067 .PC14lP Assembly Details ---

APPENDIX 82C53 CMOS PROGRAMMABLE INTERVAL TIMER GENERAL DESCRIPTIDN

The MSMB2C53.5RSIGSIJS and MSM62Ç53.2RSıGSIJS are programmable universal timers desogned or use in microcomputer systems. based on silicon gate CMOS

technology. in requires a standby current 0' only 1 00 ,.A (max) when the chip os in the nonselected state During timer operation. power consumption 05 still very now with only 5 mA Imaxl at 5 MIIz of current requored.

The devices consost to three independent counters and can count up to a maximum of 5 MIIz (MSM62C53.5) and 6 MIIz (MSM62C53.21 The timer features six dotierent counter modes. and binary count BCD count func. toons Count values can be set in byte or word units. and all functions are freely programmable

Features

.Maximum operating frequency of 5 MIIz .Six counter modes available for each counter IMSM82C53-51 .binary and decimal counting possible

.Maximum operating frequency of 8 MIIz (MSM82C5321 .24pin DIP (MSM82C535RSIMSM82C532RS)

.High sp..d and low power consumption achie...d .32.pin 'i8t package (MSM82C535GSIMSM82C53.2GSI through silicon gate CMOS technology .28pin PLCC Pockoge (MSM82C53.5JSIMSM82C53.2JSI

.Compliely static operation

.Thr.. independen, 16.bit down-counters .3V 10 6V single power supply

Functional block diagram ="",, "'c

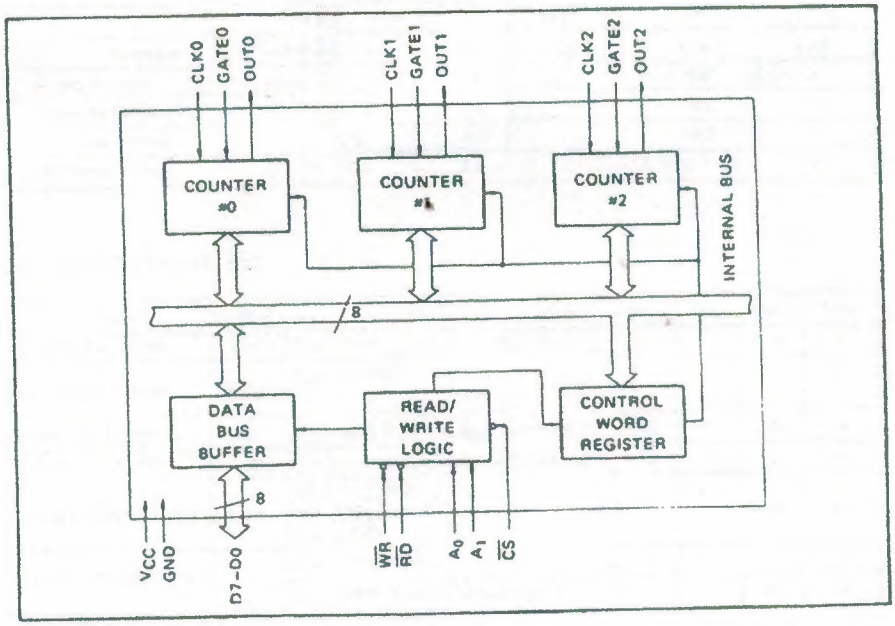


Fig 25.Counter stractur

APPENDIX 82C53

PIN CONFIGURATION

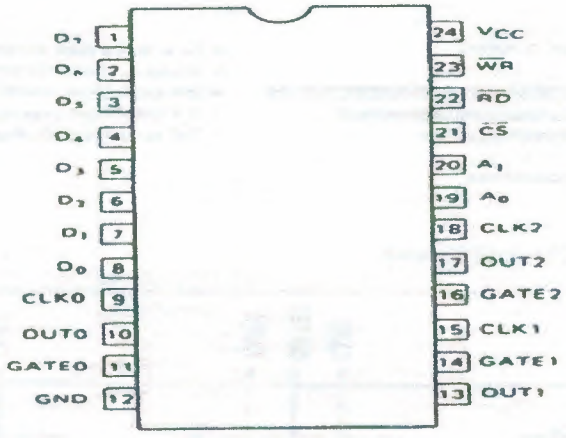


Fig 26.Pin configuration

APPENDIX 82C53

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits			Unit
			MSM82C53 5RS MSM82C53 2RS	MSM82C53 5GS MSM82C53 2GS	MSM82C53 5JS MSM82C53-2JS	
Ssupply Voltage	V_{CC}	Respect to GND	− 0.5 to + 7			V
Input Voltage	V_{IN}		− 0.5 to V_{CC} + 0.5			V
Output Voltage	V_{OUT}		− 0.5 to V_{CC} + 0.5			V
Storage Temperature	T_{stg}		− 55 to + 150			°C
Power Dissipation	P_D	Ta = 25°C	0.9	0.7	0.9	W

OPERATING RANGES

Parameter	Symbol	Limits	Conditions	Unit
Supply Voltage	V_{CC}	3 to 6	V_{IL} = 0.2V, V_{IH} = V_{CC} − 0.2V, operating frequency 2.6 MHz	V
Operating Temperature	T_{OP}	−40 to +85		°C

RECOMMENDED OPERATING CONDITIONS

Parameter	Symbol	Min.	Typ.	Max.	Unit
Supply Voltage	V_{CC}	4.5	5	5.5	V
Operating Temperature	T_{OP}	−40	+25	+85	°C
"L" Input Voltage	V_{IL}	−0.3		+0.8	V
"H" Input Voltage	V_{IH}	2.2		V_{CC} + 0.3	V

DC CHARACTERISTICS

Parameter	Symbol	Conditions		Min.	Typ.	Max.	Unit
"L" Output Voltage	V_{OL}	I_{OL} = 4mA	V_{CC}=4.5V to 5.5V Ta=−40°C to +85°C			0.45	V
"H" Output Voltage	V_{OH}	I_{OH} = −1mA		3.7			V
Input Leak Current	I_{LI}	$0 \le V_{IN} \le V_{CC}$		−10		10	μA
Output Leak Current	I_{LO}	$0 \le V_{OUT} \le V_{CC}$		−10		10	μA
Standby Supply Current	I_{CCS}	$\overline{CS} \ge V_{CC}$ −0.2V $V_{IH} \ge V_{CC}$ − 0.2V $V_{IL} \le 0.2V$				100	μA
Operating Supply Current	I_{CC}	t_{CLK} = 200 ns C_L = 0pF				5	mA
		t_{CLK} = 125 ns C_L = 0pF	MSM82C53-2			8	mA

Fig 27.Operating ranges,recommended operating conditions ,dc characteristics

APPENDIX 82C53

AC CHARACTERISTICS

(V_{CC} = 4.5V ~ 5.5V, Ta = -40 ~ +85° C)

Parameter	Symbol	MSM82C53-5 Min	MSM82C53-5 Max	MSM82C53-2 Min	MSM82C53-2 Max	Unit	Conditions
Address Set-up Time before reading	TAR	30		30		ns	C_L = 150pF
Address Hold Time after reading	TRA	0		0		ns	Read cycle
Read Pulse Width	TRR	150		150		ns	
Read Recovery Time	TRVR	200		200		ns	
Address Set-up Time before writing	TAW	0		0		ns	
Address Hold Time after writing	TWA	30		20		ns	
Write Pulse Width	TWW	150		150		ns	Write cycle
Data Input Set-up Time before writing	TDW	100		100		ns	
Data Input Hold Time after writing	TWD	30		20		ns	
Write Recovery time	TRVW	200		200		ns	
Clock Cycle Time	TCLK	200	D.C.	125	D.C.	ns	
Clock "H" Pulse Width	TPWH	60		60		ns	
Clock "L" Pulse Width	TPWL	60		60		ns	Clock and gate timing
"H" Gate Pulse Width	TGW	50		50		ns	
"L" Gate Pulse Width	TGL	50		50		ns	
Gate Input Set-up Time before clock	TGS	50		50		ns	
Gate Input Hold Time after clock	TGH	50		50		ns	
Output Delay Time after reading	TRD		120		120	ns	
Output Floating Delay Time after reading	TDF	5	90	5	90	ns	
Output Delay Time after gate	TODG		120		120	ns	Delay time
Output Delay Time after clock	TOD		150		150	ns	
Output Delay Time after address	TAD		180		180	ns	

Note: Timing measured at V_L = 0.8V and V_H = 2.2V for both inputs and outputs.

TIME CHART

Write Timing

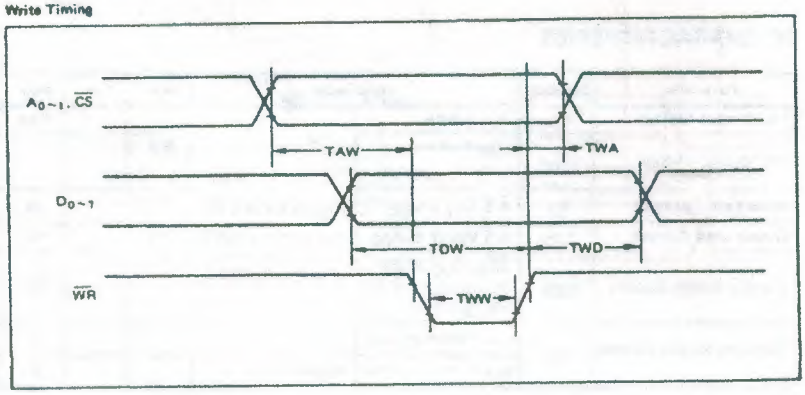
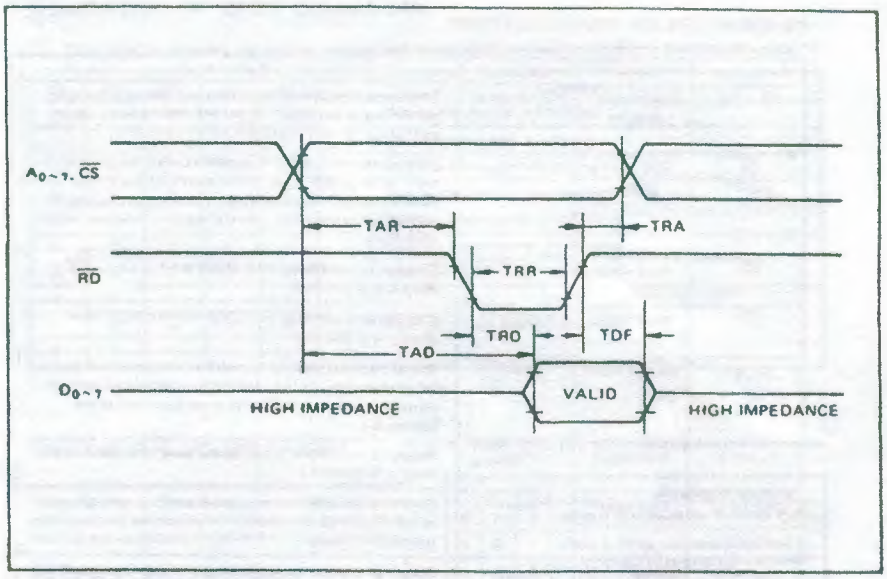


Fig 28.AC characteristics ,Time chart

APPENDIX 82C53

Read Timing



Clock & Gate Timing

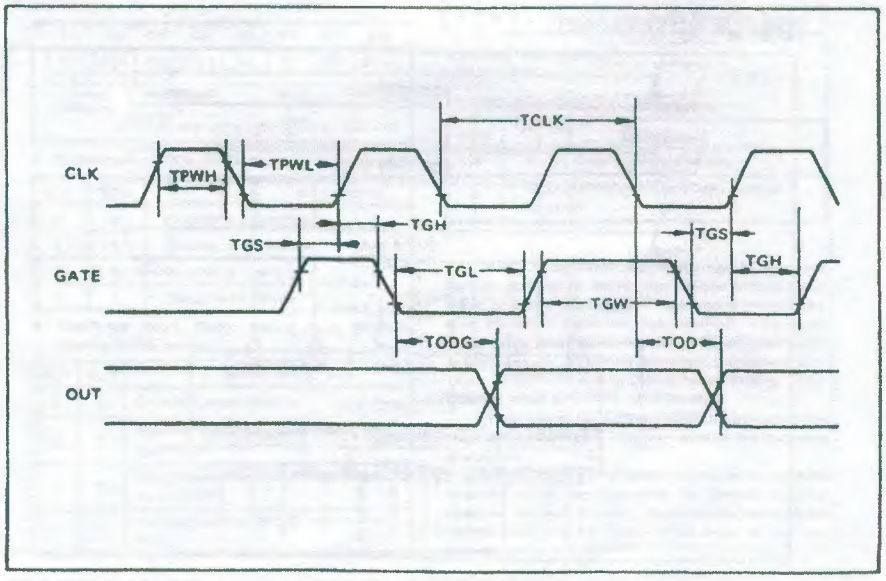


Fig 29.Read timing

APPENDIX 82C53

DESCRIPTION OF PIN FUNCTIONS

Pin Symbol	Name	Input/output	Function
D7 ~ D0	Bidirectional data bus	Input/output	Three state 8-bit bidirectional data bus used when writing control words and count values, and reading count values upon reception of $\overline{WR}$ and $\overline{RD}$ signals from CPU.
$\overline{CS}$	Chip select input	Input	Data transfer with the CPU is enabled when this pin is at low level. When at high level, the data bus (ID_0 thru D_7) is switched to high impedance state where neither writing nor reading can be executed. Internal registers, however, remain unchanged.
$\overline{RD}$	Read input	Input	Data can be transferred from MSM82C53 to CPU when this pin is at low level.
$\overline{WR}$	Write input	Input	Data can be transferred from CPU to MSM82C53 when this pin is at low level.
A0, A1	Address input	Input	One of the three internal counters or the control word register is selected by A0/A1 combination. These two pins are normally connected to the two lower order bits of the address bus.
CLK0~2	Clock input	Input	Supply of three clock signals to the three counters incorporated in MSM82C53.
GATE0~2	Gate input	Input	Control of starting, interruption, and restarting of counting in the three respective counters in accordance to the set control word contents.
OUT0~2	Counter output	Output	Output of counter output waveform in accordance with the set mode and count value.

SYSTEM INTERFACING

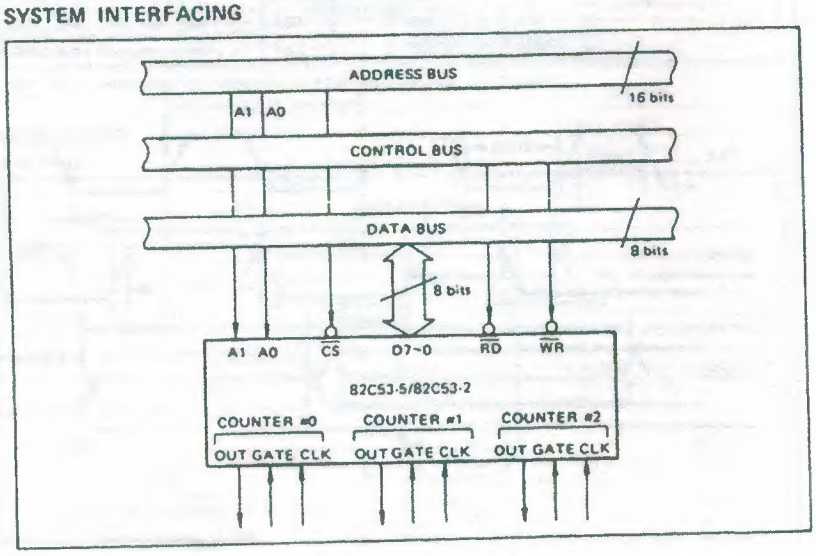


Fig 30.System interfacing

APPENDIX 82C53

DESCRIPTION OF BASIC OPERATIONS

Data transfers between the internal registers and the external data bus is outlined in the following table.

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A1	A0	Function
0	1	0	0	0	Data bus to counter # 0 Writing
0	1	0	0	1	Data bus to counter # 1 Writing
0	1	0	1	0	Data bus to counter # 2 Writing
0	1	0	1	1	Data bus to control word register Writing
0	0	1	0	0	Data bus from counter # 0 Reading
0	0	1	0	1	Data bus from counter # 1 Reading
0	0	1	1	0	Data bus from counter # 2 Reading
0	0	1	1	1	Data bus in high impedance status
1	x	x	x	x	
0	1	1	x	x	

x denotes "not specified".

DESCRIPTION OF OPERATION

82C53 functions are selected by a control word from the CPU. In the required program sequence, the control word setting is followed by the count value setting and execution of the desired timer operation.

Control Word and Count Value Program

Each counter operation mode is set by control word programming. The control word format is outlined below.

D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	RL1	RL0	M2	M1	M0	BCD
Select Counter		Read/Load		Mode			BCD

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

- **Select Counter (SC0, SC1):** Selection of set counter

SC1	SC0	Set Contents
0	0	Counter # 0 selection
0	1	Counter # 1 selection
1	0	Counter # 2 selection
1	1	Illegal combination

- **Read/Load (RL1, RL0):** Count value Reading/Loading format setting

RL1	RL0	Set Contents
0	0	Counter Latch operation
0	1	Reading/Loading of Least Significant byte (LSB)
1	0	Reading/Loading of Most Significant byte (MSB)
1	1	Reading/Loading of LSB followed by MSB

- **Mode (M2, M1, M0):** Operation waveform mode setting

M2	M1	M0	Set Contents
0	0	0	Mode 0 (Interrupt on Terminal Count)
0	0	1	Mode 1 (Programmable One-Shot)
x	1	0	Mode 2 (Rate Generator)
x	1	1	Mode 3 (Square Wave Generator)
1	0	0	Mode 4 (Software Triggered Strobe)
1	0	1	Mode 5 (Hardware Triggered Strobe)

x denotes "not specified".

- **BCD:** Operation count mode setting

BCD	Set Contents
0	Binary Count (16-bits Binary)
1	BCD Count (4-decades Binary Coded Decimal)

After setting Read/Load, Mode, and BCD in each counter as outlined above, next set the desired count value. (In some Modes, counting is started immediately after the count value has been written). This count value setting must conform with the Read/Load format set in advance. Note that the internal counters are reset to 0000H during control word setting. The counter value (0000H) can,t be read.

If the two bytes (LSB and MSB) are written at this stage (RL0 and RL1 = 1,1), take note of the following precaution.

Although the count values may be set in the three counters in any sequence after the control word has been set in each counter, count values must be set consecutively in the LSB – MSB order in any one counter.

Fig 31.Discription of operation

change ("H" to "L" or "L" to "H") in the next counter output to be executed. The counting operation at the gate input is done the same as in mode 2.

- **Mode 4 (software trigger strobe)**
  The counter output is switched to "H" level by the mode setting. Counting is started in the same way as described for mode 0. A single "L" pulse equivalent to one clock width is generated at the counter output when the terminal count is reached.
  This mode differs from 2 in that the "L" level output appears one clock earlier in mode 2, and that pulses are not repeated in mode 4. Counting is stopped when the gate input is switched to "L" level, and restarted from the set count value when switched back to "H" level.

- **Mode 5 (hardware trigger strobe)**
  The counter output is switched to "H" level by the mode setting. Counting is started, and the gate input used, in the same way as in mode 1.
  The counter output is identical to the mode 4 output .
  The various roles of the gate input signals in the above modes are summarized in the following table.

Mode \ Gate	"L" Level Falling Edge	Rising Edge	"H" Level
0	Counting not possible		Counting possible
1		(1) Start of counting (2) Retriggering	
2	(1) Counting not possible (2) Counter output forced to "H" level	Start of counting	Counting possible
3	(1) Counting not possible (2) Counter output forced to "H" level	Start of counting	Counting possible
4	Counting not possible		Counting possible
5		(1) Start of counting (2) Retriggering	

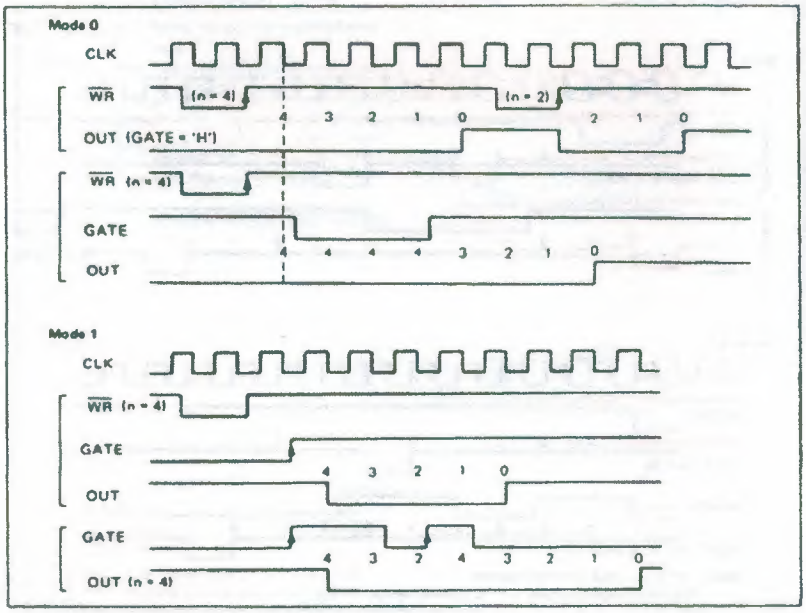


Fig 32 .

APPENDIX 82C53

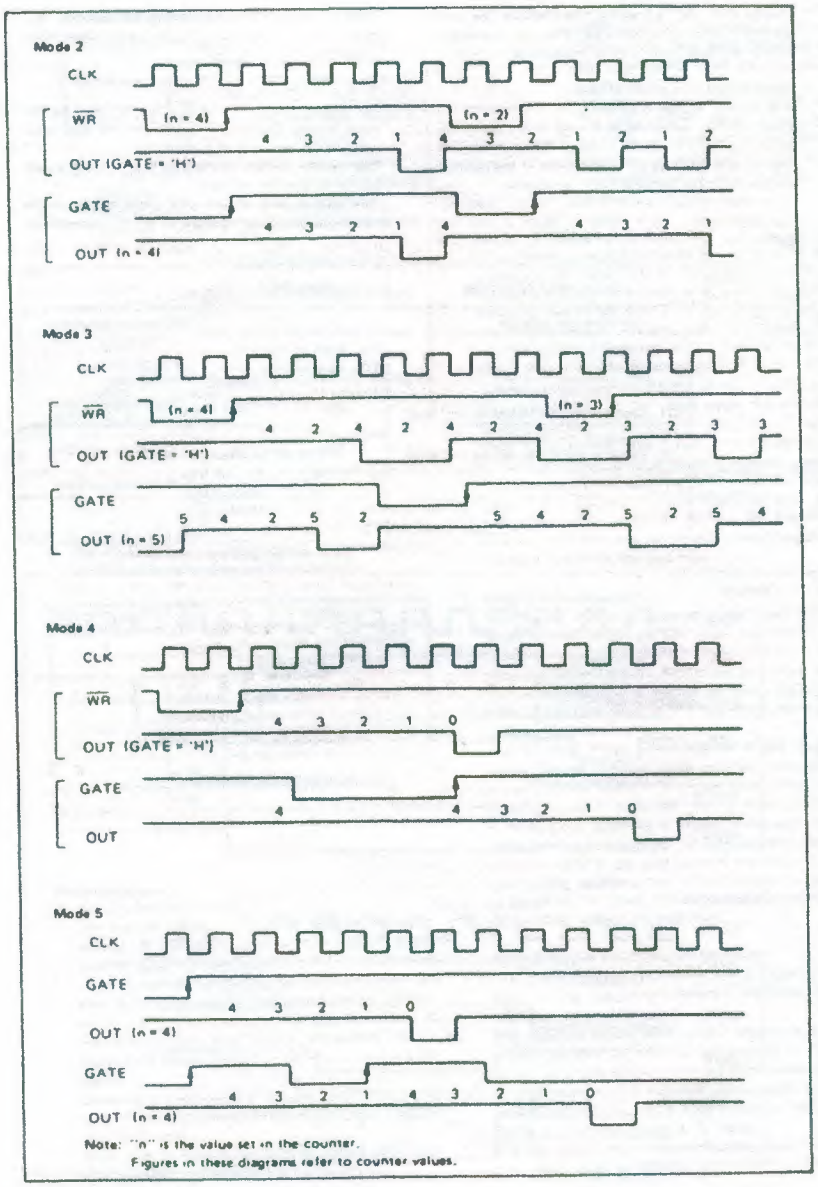


Fig 33.

APPENDIX 82C53

Reading of Counter Values

All 82C53 counting is down-counting, the counting being in steps of 2 in mode 3. Counter values can be read during counting by (1) direct reading, and (2) counter latching ("read on the fly").

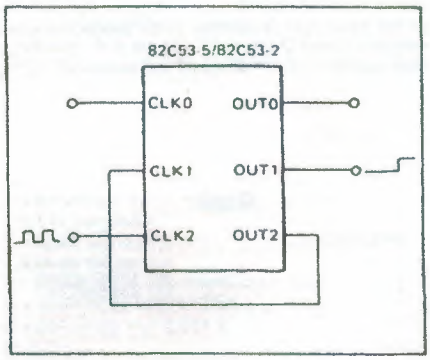
- **Direct reading**
  Counter values can be read by direct reading operations.
  Since the counter value read according to the timing of the $\overline{RD}$ and CLK signals is not guaranteed, it is necessary to stop the counting by a gate input signal, or to interrupt the clock input temporarily by an external circuit to ensure that the counter value is correctly read.

- **Counter latching**
  In this method, the counter value is latched by writing a counter latch command, thereby enabling a stable value to be read without effecting the counting in any way at all. An example of a counter latching program is given below.
  Counter latching executed for counter #1 (Read/Load 2-byte setting)

```
MVI A  0 1 0 0 x x x x
                └── Denotes counter latching
OUT n3 ──┬── Write in control word address (n3)
         └── The counter value at this point is latched
IN n1  ──┬── Reading of the LSB of the counter value latched from counter #1.
         └── n1: Counter #1 address
MOV B, A
IN n1  ──── Reading of MSB from counter #1.
MOV C, A
```

Example of Practical Application

- 82C53 used as a 32-bit counter.



Use counter #1 and counter #2

Counter #1: mode 0, upper order 16-bit counter value

Counter #2: mode 2, lower order 16-bit counter value

This setting enables counting up to a maximum of 2^{33}.

Fig 34.

APPENDIX 82C55

CMOS PROGRAMMABLE PERIPHERAL INTERFACE

GENERAL DESCRIPTION

The MSM82C55A is a programmable universal I/O interface device which operates as high speed and on low power consumption due to 3 μ silicon gate CMOS technology. It is the best fit as an I/O port in a system which employs the 8-bit parallel processing MSM80C85A CPU. This device has 24-bit I/O pins equivalent to three 8-bit I/O ports and all inputs/outputs are TTL interface compatible.

FEATURES

- High speed and low power consumption due to 3 μ silicon gate CMOS technology
- 3 V to 6 V single power supply
- Full static operation
- Programmable 24-bit I/O ports
- Bidirectional bus operation (Port A)
- Bit set/reset function (Port C)
- TTL compatible
- 40-pin DIP (MSM82C55A-5RS/MSM82C55A-2RS)
- 44-pin flat package (MSM82C55A-5GS/MSM82C55A-2GS)
- 44-pin PLCC (MSM82C55A-2JS)
- Compatible with 8255A-5

CIRCUIT CONFIGURATION

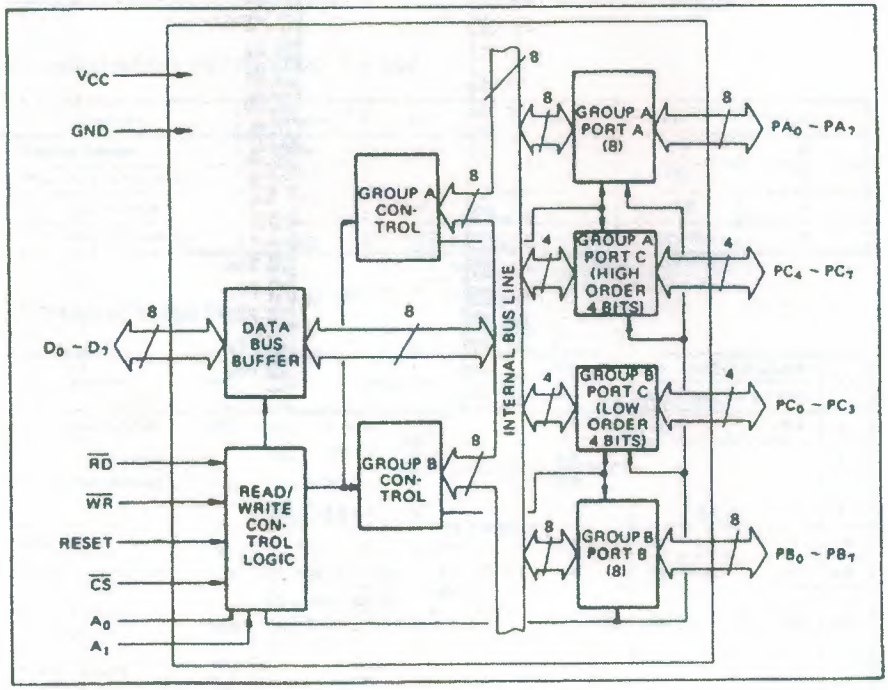


Fig 35.

APPENDIX 82C55

PIN CONFIGURATION

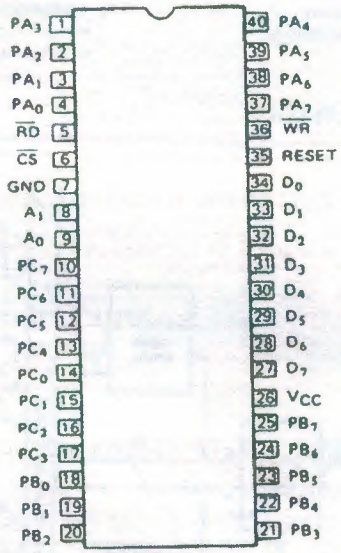


Fig 36.

APPENDIX 82C55

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits			Unit
			MSM82C55A 5RS MSM82C55A 2RS	MSM82C55A 5GS MSM82C55A 2GS	MSM82C55A 5JS	
Ssupply Voltage	V_{CC}	Ta = 25°C with respect to GND	−0.5 to +7			V
Input Voltage	V_{IN}		−0.5 to V_{CC} + 0.5			V
Output Voltage	V_{OUT}		−0.5 to V_{CC} + 0.5			V
Storage Temperature	T_{stg}	–	−55 to +150			°C
Power Dissipation	P_D	Ta = 25°C	1.0	0.7	1.0	W

OPERATING RANGE

Parameter	Symbol	Limits	Unit
Supply Voltage	V_{CC}	3 to 6	V
Operating Temperature	T_{OP}	−40 to 85	°C

RECOMMENDED OPERATING RANGE

Parameter	Symbol	Min.	Typ.	Max.	Unit
Supply Voltage	V_{CC}	4.5	5	5.5	V
Operating Temperature	T_{OP}	−40	+25	+85	°C
"L" Input Voltage	V_{IL}	−0.3		+0.8	V
"H" Input Voltage	V_{IH}	2.2		V_{CC} +0.3	V

DC CHARACTERISTICS

Parameter	Symbol	Conditions		MSM82C55A-5			MSM82C55A-2			Unit
				Min.	Typ.	Max.	Min.	Typ.	Max.	
"L" Output Voltage	V_{OL}	I_{OL} = 2.5 mA				0.45			0.4	V
"H" Output Voltage	V_{OH}	I_{OH} = −400 μA		2.4						V
		I_{OH} = −40 μA		4.2						V
		I_{OH} = −2.5 mA		–			3.7			V
Input Leak Current	I_{LI}	$0 \leq V_{IN} \leq V_{CC}$	V_{CC} = 4.5V to 5.5V Ta = −40°C to +85°C (C_L = 0pF)	−10		10	−1		1	μA
Output Leak Current	I_{LO}	$0 \leq V_{OUT} \leq V_{CC}$		−10		10	−10		10	μA
Supply Current (standby)	I_{CCS}	$\overline{CS} \geq V_{CC}$ −0.2V $V_{IH} \geq V_{CC}$ −0.2V $V_{IL} \leq$ 0.2V			0.1	100		0.1	10	μA
Average Supply Current (active)	I_{CC}	[illegible] 82C55A-5 3MHz/CPU [illegible] 82C55A-2 8MHz/CPU [illegible]				5			8	mA

Fig 37.

APPENDIX 82C55

AC CHARACTERISTICS

(V_{CC} = 4.5 to 5.5V, Ta = -40 to +80°C)

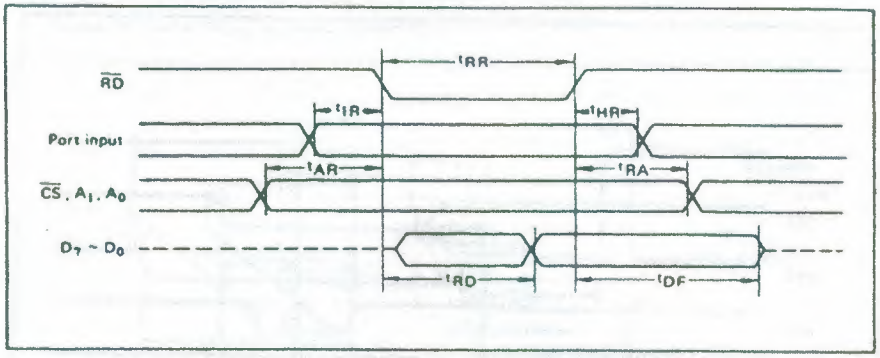
Parameter	Symbol	MSM82C55A-5 Min	MSM82C55A-5 Max.	MSM82C55A-2 Min	MSM82C55A-2 Max.	Unit	Remarks
Setup Time of address to the falling edge of $\overline{RD}$	t_{AR}	20		20		ns	
Hold Time of address to the rising edge of $\overline{RD}$	t_{RA}	20		0		ns	
$\overline{RD}$ Pulse Width	t_{RR}	300		100		ns	
Delay Time from the falling edge of $\overline{RD}$ to the output of defined data	t_{RD}		200		120	ns	
Delay Time from the rising edge of $\overline{RD}$ to the floating of data bus	t_{DF}	10	100	10	75	ns	
Time from the rising edge of $\overline{RD}$ or $\overline{WR}$ to the next falling edge of $\overline{RD}$ or $\overline{WR}$	t_{RV}	850		200		ns	
Setup Time of address before the falling edge of $\overline{WR}$	t_{AW}	0		0		ns	
Hold Time of address after the rising edge of $\overline{WR}$	t_{WA}	30		20		ns	
$\overline{WR}$ Pulse Width	t_{WW}	300		150		ns	
Setup Time of bus data before the rising edge of $\overline{WR}$	t_{DW}	100		50		ns	
Holt Time of bus data after the rising edge of $\overline{WR}$	t_{WD}	40		30		ns	
Delay Time from the rising edge of $\overline{WR}$ to the output of defined data	t_{WB}		350		200	ns	
Setup Time of port data before the falling edge of $\overline{RD}$	t_{IR}	20		20		ns	
Hold Time of port data after the rising edge of $\overline{RD}$	t_{HR}	20		10		ns	
$\overline{ACK}$ Pulse Width	t_{AK}	300		100		ns	
$\overline{STB}$ Pulse Width	t_{ST}	300		100		ns	Load 150pF
Setup Time of port data before the rising edge of $\overline{STB}$	t_{PS}	20		20		ns	
Hold Time of port data after the rising edge of $\overline{STB}$	t_{PH}	180		50		ns	
Delay Time from the falling edge of $\overline{ACK}$ to the output of defined data	t_{AD}		300		150	ns	
Delay Time from the rising edge of $\overline{ACK}$ to the floating of port (Port A in mode 2)	t_{KD}	20	250	20	250	ns	
Delay Time from the rising edge of WR to the falling edge of $\overline{OBF}$	t_{WOB}		650		150	ns	
Delay Time from the falling edge of ACK to the rising edge of $\overline{OBF}$	t_{AOB}		350		150	ns	
Delay Time from the falling edge of STB to the rising edge of IBF	t_{SIB}		300		150	ns	
Delay Time from the rising edge of $\overline{RD}$ to the falling edge of IBF	t_{RIB}		300		150	ns	
Delay Time from the falling edge of $\overline{RD}$ to the falling edge of INTR	t_{RIT}		400		200	ns	
Delay Time from the rising edge of $\overline{STB}$ to the rising edge of INTR	t_{SIT}		300		150	ns	
Delay Time from the rising edge of $\overline{ACK}$ to the rising edge of INTR	t_{AIT}		350		150	ns	
Delay Time from the falling edge of $\overline{WR}$ to the falling edge of INTR	t_{WIT}		850		250	ns	

Note: Timing is measured at V_L = 0.8 V and V_H = 2.2 V for both input and outputs

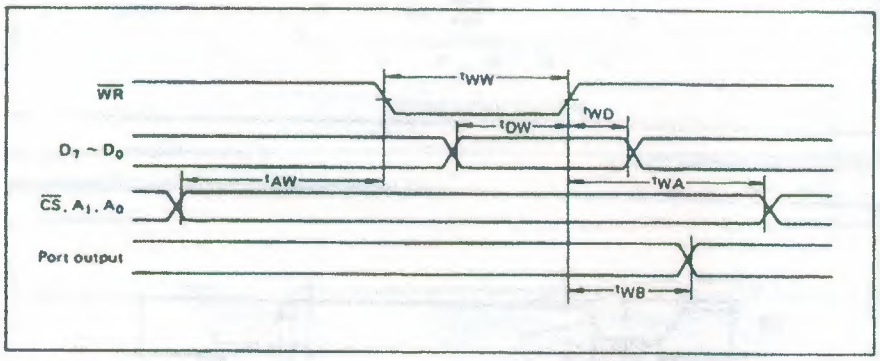
Fig 38.

APPENDIX 82C55

Basic Input Operation (Mode 0)



Basic Output Operation (Mode 0)



Strobe Input Operation (Mode 1)

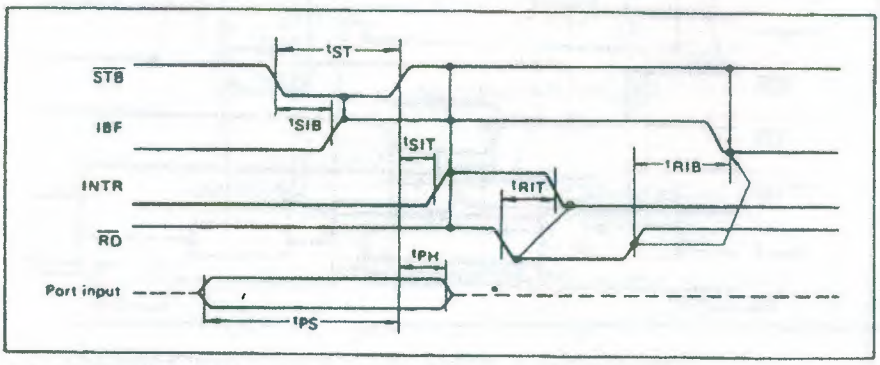
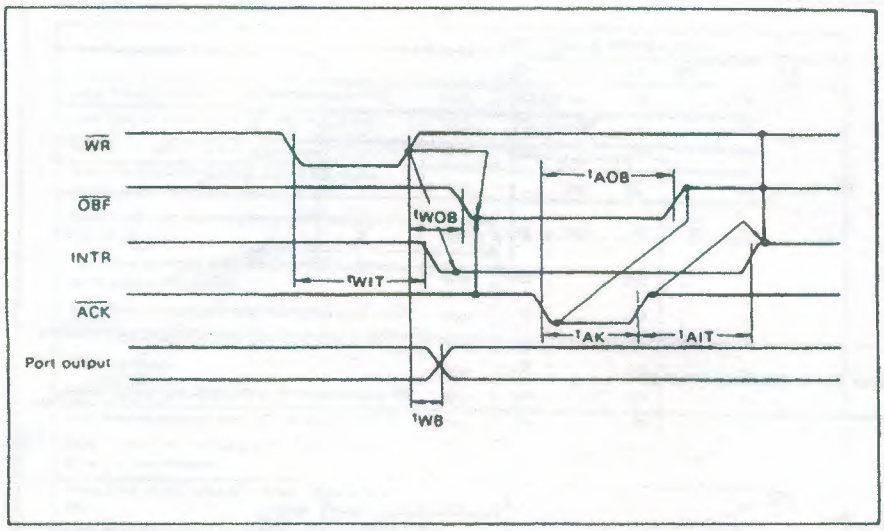


Fig 39.

APPENDIX 82C55

Strobe Output Operation (Mode 1)



Bidirectional Bus Operation (Mode 2)

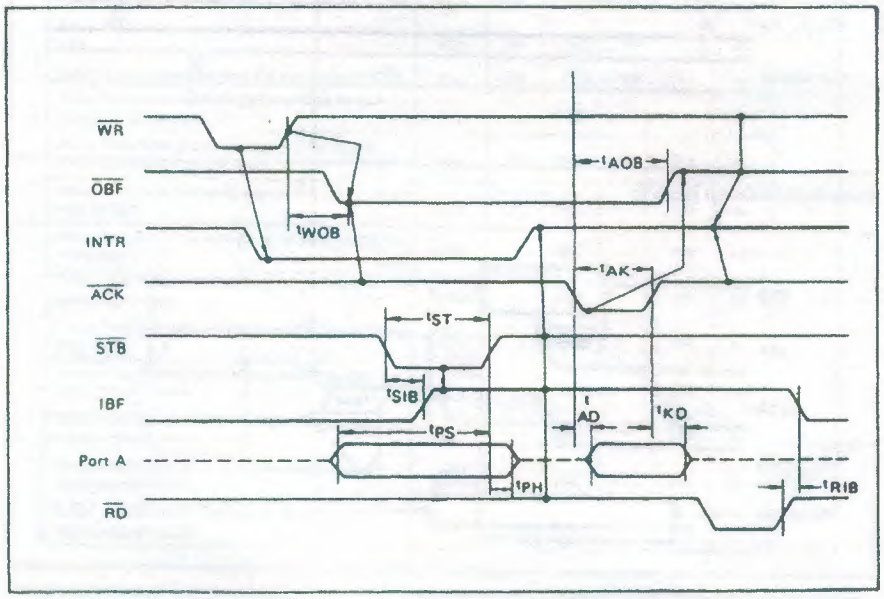
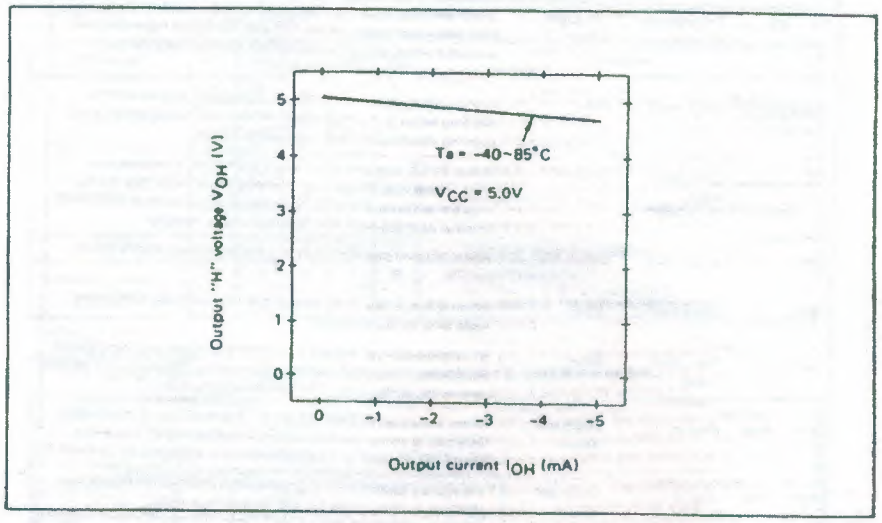


Fig 40.

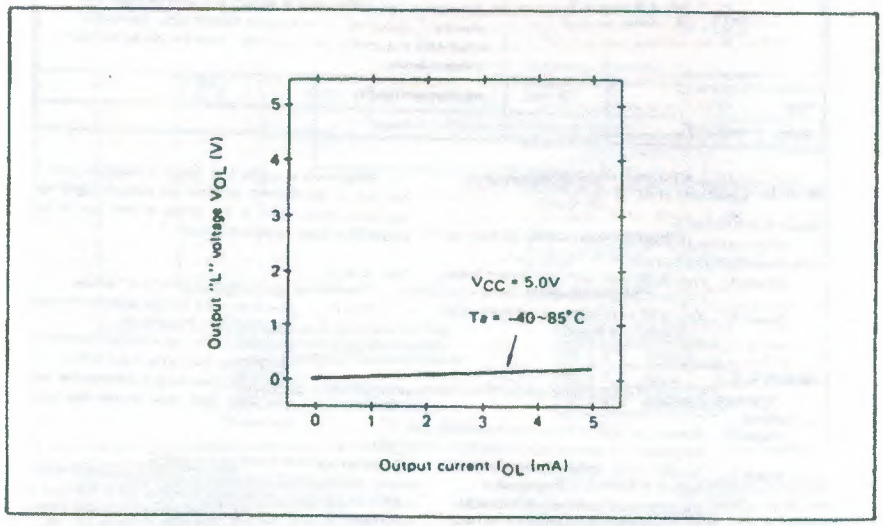
APPENDIX 82C55

OUTPUT CHARACTERISTICS (REFERENCE VALUE)

1 Output "H" Voltage (V_{OH}) vs. Output Current (I_{OH})



2 Output "L" Voltage (V_{OL}) vs. Output Current (I_{OL})



Note: The direction of flowing into the device is taken as positive for the output current.

Fig 41.

APPENDIX 82C55

FUNCTIONAL DESCRIPTION OF PIN

Pin No	Item	Input/Output	Function
D7 - D0	Bidirectional data bus	Input and output	These are three state 8-bit bidirectional buses used to write and read data upon receipt of the $\overline{WR}$ and $\overline{RD}$ signals from CPU and also used when control words and bit set/reset data are transferred from CPU to MSM82C55A.
RESET	Reset input	Input	This signal is used to reset the control register and all internal registers when it is in high level. At this time, ports are all made into the input mode (high impedance status).
$\overline{CS}$	Chip select input	Input	When the $\overline{CS}$ is in low level, data transmission is enabled with CPU. When it is in high level, the data bus is made into the high impedance status where no write nor read operation is performed. Internal registers hold their previous status, however.
$\overline{RD}$	Read input	Input	When $\overline{RD}$ is in low level, data is transferred from MSM82C55A to CPU.
$\overline{WR}$	Write input	Input	When $\overline{WR}$ is in low level, data or control words are transferred from CPU to MSM82C55A.
A0, A1	Port select input (address)	Input	By combination of A0 and A1, either one is selected from among port A, port B, port C, and control register. These pins are usually connected to low-order 2 bits of the address bus.
PA7 - PA0	Port A	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word. Especially, port A can be used as a bidirectional port when it is set to mode 2.
PB7 - PB0	Port B	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word.
PC7 - PC0	Port C	Input and output	These are universal 8-bit I/O ports. The direction of inputs/outputs can be determined by writing a control word as 2 ports with 4 bits each. When port A or port B is used in mode 1 or mode 2 (port A only), they become control pins. Especially when port C is used as an output port, each bit can be set/reset independently.
VCC			+5 V power supply.
GND			GND

BASIC FUNCTIONAL DESCRIPTION

Group A and Group B

When setting a mode to a port having 24 bits, set it by dividing it into two groups of 12 bits each.

- Group A: Port A (8 bits) and high order 4 bits of port C (PC7 ~ PC4)
- Group B: Port B (8 bits) and low order 4 bits of port C (PC3 ~ PC0)

Mode 0, 1, 2

There are 3 types of modes to be set by grouping as follows.

- Mode 0: Basic input operation/output operation (Available for both groups A and B)
- Mode 1: Strobe input operation/output operation (Available for both groups A and B)
- Mode 2: Bidirectional bus operation (Available for group A only)

When used in mode 1 or mode 2, however, port C has bits to be defined as ports for control signal for operation ports (port A for group A and port B for group B) of their respective groups.

Port A, B, C

The internal structure of 3 ports is as follows:

- Port A: One 8-bit data output latch/buffer and one 8-bit data input latch
- Port B: One 8-bit data input/output latch/buffer and one 8-bit data input buffer
- Port C: One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input)

Single bit set/reset function for port C

When port C is defined as an output port, it is possible to set (to turn to high level) or reset (to turn to low level) any one of 8 bits individually without affecting other bits.

Fig. 42.

APPENDIX 82C55

OPERATIONAL DESCRIPTION

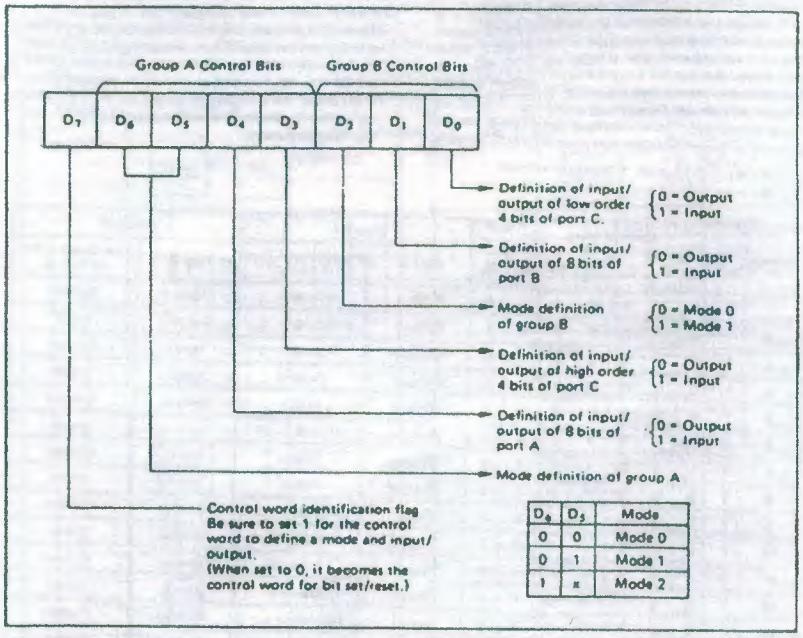
Control Logic

Operations by addresses and control signals, e.g., read and write, etc are as shown in the table below:

Operation	A1	A0	$\overline{CS}$	$\overline{WR}$	$\overline{RD}$	Operation
Input	0	0	0	1	0	Port A → Data Bus
	0	1	0	1	0	Port B → Data Bus
	1	0	0	1	0	Port C → Data Bus
Output	0	0	0	0	1	Data Bus → Port A
	0	1	0	0	1	Data Bus → Port B
	1	0	0	0	1	Data Bus → Port C
Control	1	1	0	0	1	Data Bus → Control Register
Others	1	1	0	1	0	Illegal Condition
	x	x	1	x	x	Data bus is in the high impedance status

Setting of Control Word

The control register is composed of 7-bit latch circuit and 1-bit flag as shown below.



Precaution for mode selection

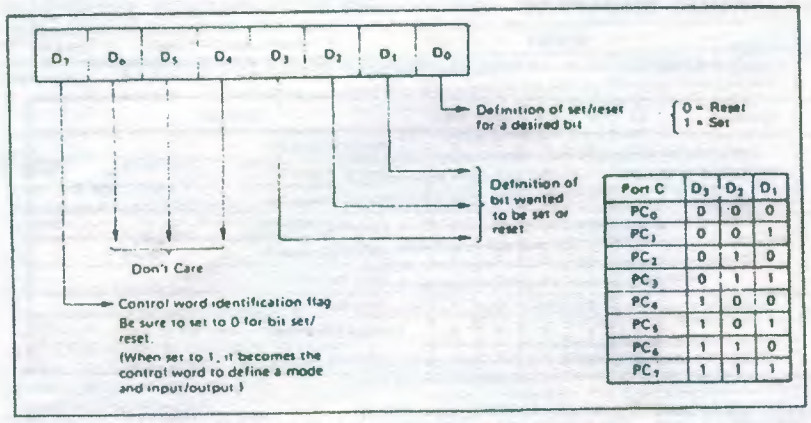
The output registers for ports A and C are cleared to ϕ each time data is written in the command register and the mode is changed, but the port B state is undefined.

Bit Set/Reset Function

When port C is defined as output port, it is possible to set (set output to 1) or reset (set output to 0) any one of 8 bits without affecting other bits as shown next page.

Fig 43.

APPENDIX 82C55



Port C	D_3	D_2	D_1
PC_0	0	0	0
PC_1	0	0	1
PC_2	0	1	0
PC_3	0	1	1
PC_4	1	0	0
PC_5	1	0	1
PC_6	1	1	0
PC_7	1	1	1

Interrupt Control Function

When the MSM82C55A is used in mode 1 or mode 2, the interrupt signal for the CPU is provided. The interrupt request signal is output from port C. When the internal flip-flop INTE is set beforehand at this time, the desired interrupt request signal is output. When it is reset beforehand, however, the interrupt request signal is not output. The set/reset of the internal flip-flop is made by the bit set/reset operation for port C virtually.

Bit set → INTE is set → Interrupt allowed
Bit reset → INTE is reset → Interrupt inhibited

Operational Description by Mode

1. **Mode 0 (Basic input/output operation)**

Mode 0 makes the MSM82C55A operate as a basic input port or output port. No control signals such as interrupt request, etc. are required in this mode. All 24 bits can be used as two 8-bit ports and two 4-bit ports. Sixteen combinations are then possible for inputs/outputs. The inputs are not latched, but the outputs are.

Type	Control Word								Group A		Group B	
	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Port A	High Order 4 Bits of Port C	Port B	Low Order 4 Bits of Port C
1	1	0	0	0	0	0	0	0	Output	Output	Output	Output
2	1	0	0	0	0	0	0	1	Output	Output	Output	Input
3	1	0	0	0	0	0	1	0	Output	Output	Input	Output
4	1	0	0	0	0	0	1	1	Output	Output	Input	Input
5	1	0	0	0	1	0	0	0	Output	Input	Output	Output
6	1	0	0	0	1	0	0	1	Output	Input	Output	Input
7	1	0	0	0	1	0	1	0	Output	Input	Input	Output
8	1	0	0	0	1	0	1	1	Output	Input	Input	Input
9	1	0	0	1	0	0	0	0	Input	Output	Output	Output
10	1	0	0	1	0	0	0	1	Input	Output	Output	Input
11	1	0	0	1	0	0	1	0	Input	Output	Input	Output
12	1	0	0	1	0	0	1	1	Input	Output	Input	Input
13	1	0	0	1	1	0	0	0	Input	Input	Output	Output
14	1	0	0	1	1	0	0	1	Input	Input	Output	Input
15	1	0	0	1	1	0	1	0	Input	Input	Input	Output
16	1	0	0	1	1	0	1	1	Input	Input	Input	Input

Note: When used in mode 0 for both groups A and B

Fig 44.

APPENDIX 82C55

2. Mode 1 (Strobe input/output operation)

In mode 1, the strobe, interrupt and other control signals are used when input/output operations are made from a specified port. This mode is available for both groups A and B. In group A at this time, port A is used as the data line and port C as the control signal.

Following is a description of the input operation in mode 1.

$\overline{STB}$ (Strobe input)

- When this signal is low level, the data output from terminal to port is fetched into the internal latch of the port. This can be made independent from the CPU, and the data is not output to the data bus until the RD signal arrives from the CPU.

IBF (Input buffer full flag output)

- This is the response signal for the $\overline{STB}$. This signal when turned to high level indicates that data is fetched into the input latch. This signal turns to high level at the falling edge of $\overline{STB}$ and to low level at the rising edge of $\overline{RD}$.

INTR (Interrupt request output)

- This is the interrupt request signal for the CPU of the data fetched into the input latch. It is indicated by high level only when the internal INTE flip-flop is set. This signal turns to high level at the rising edge of the $\overline{STB}$ (IBF = 1 at this time) and low level at the falling edge of the $\overline{RD}$ when the INTE is set.

$INTE_A$ of group A is set when the bit for PC_4 is set, while $INTE_B$ of group B is set when the bit for PC_2 is set.

Following is a description of the output operation of mode 1.

$\overline{OBF}$ (Output buffer full flag output)

- This signal when turned to low level indicates that data is written to the specified port upon receipt of the $\overline{WR}$ signal from the CPU. This signal turns to low level at the rising edge of the WR and high level at the falling edge of the ACK.

$\overline{ACK}$ (Acknowledge input)

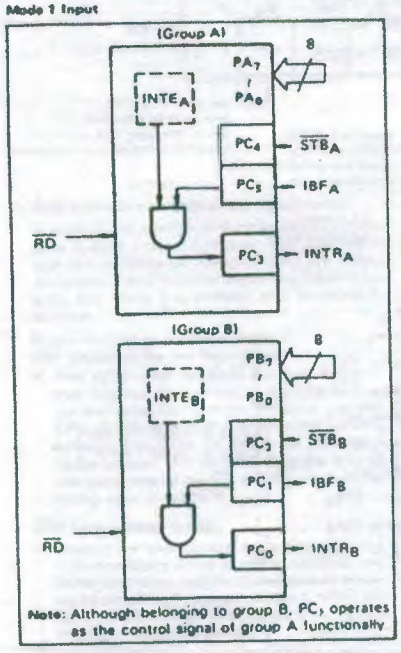
- This signal when turned to low level indicates that the terminal has received data.

INTR (Interrupt request output)

- This is the signal used to interrupt the CPU when a terminal receives data from the CPU via the MSMB2C55A-5. It indicates the occurrence of the interrupt in high level only when the internal INTE flip-flop is set. This signal turns to high level at the rising edge of the $\overline{ACK}$ (OBF = 1 at this time) and low level at the falling edge of $\overline{WR}$ when the $INTE_B$ is set.

$INTE_A$ of group A is set when the bit for PC_6 is set, while $INTE_B$ of group B is set when the bit for PC_2 is set.

Mode 1 Input



Note: Although belonging to group B, PC_7 operates as the control signal of group A functionally

Mode 1 output

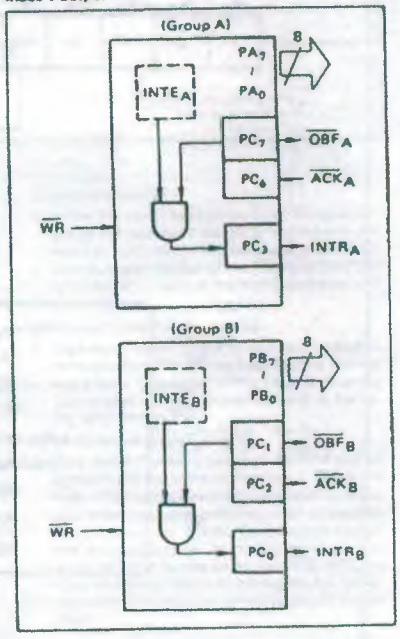


Fig 45.

APPENDIX 82C55

Port C Function Allocation in Mode 1

Combination of Input Output / Port C	Group A Input Group B Input	Group A Input Group B Output	Group A: Output Group B: Input	Group A: Output Group B: Output
PC_0	$INTR_B$	$INTR_B$	$INTR_B$	$INTR_B$
PC_1	IBF_B	$\overline{OBF}_B$	IBF_B	$\overline{OBF}_B$
PC_2	$\overline{STB}_B$	$\overline{ACK}_B$	$\overline{STB}_B$	$\overline{ACK}_B$
PC_3	$INTR_A$	$INTR_A$	$INTR_A$	$INTR_A$
PC_4	$\overline{STB}_A$	$\overline{STB}_A$	I/O	I/O
PC_5	IBF_A	IBF_A	I/O	I/O
PC_6	I/O	I/O	$\overline{ACK}_A$	$\overline{ACK}_A$
PC_7	I/O	I/O	$\overline{OBF}_A$	$\overline{OBF}_A$

Note: I/O is a bit not used as the control signal, but it is available as a port of mode 0

Examples of the relation between the control words and pins when used in mode 1 is shown below:

(a) When group A is mode 1 output and group B is mode 1 input.

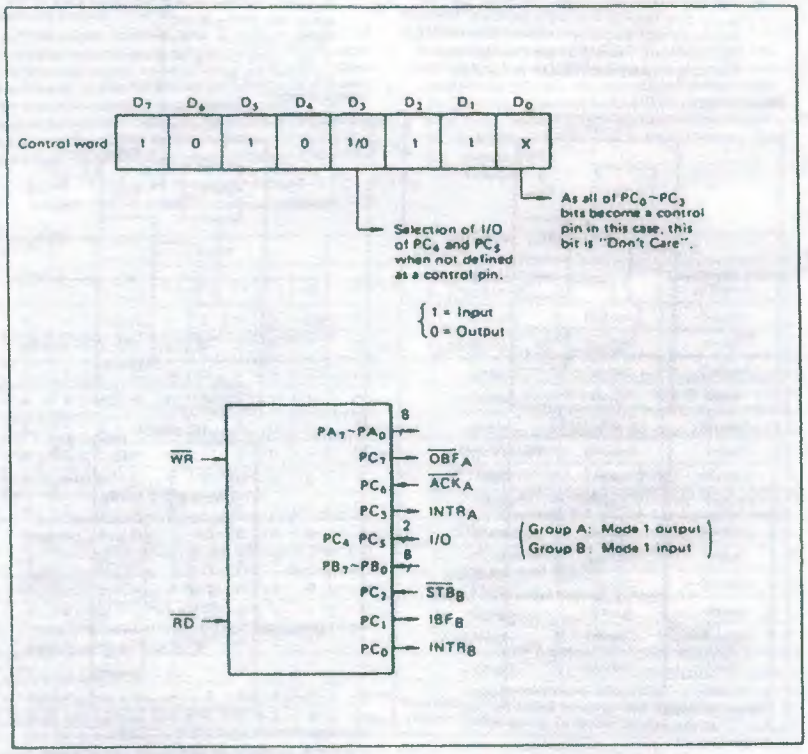
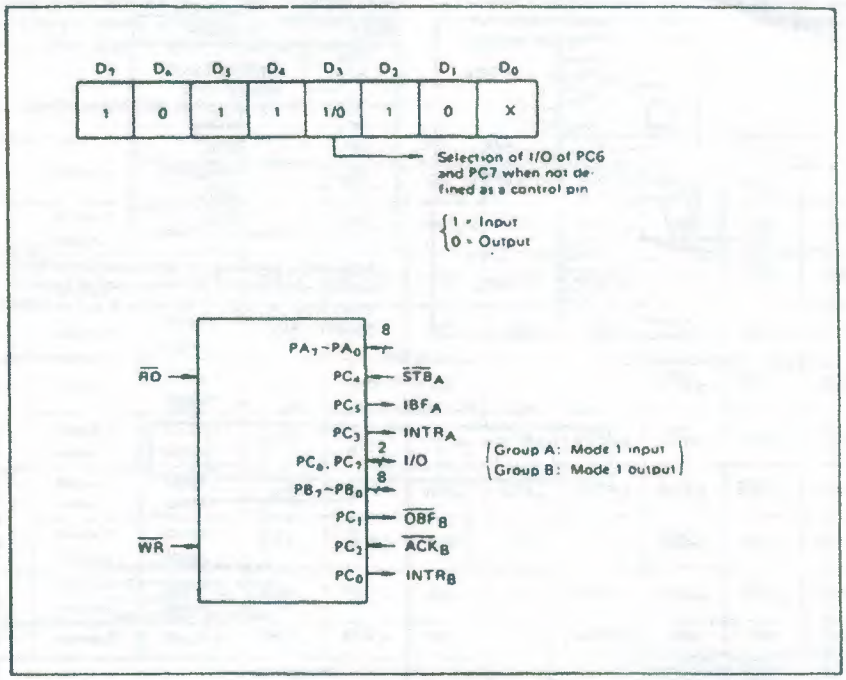


Fig 46.

(b) When group A is mode 1 input and group B is mode 1 output



3. Mode 2 (Strobe bidirectional bus I/O operation)

In mode 2, it is possible to transfer data in 2 directions through a single 8-bit port. This operation is akin to a combination between input and output operations. Port C waits for the control signal in this case, too. Mode 2 is available only for group A, however.

Next, a description is made on mode 2.

$\overline{OBF}$ (Output buffer full flag output)

- This signal when turned to low level indicates that data has been written to the internal output latch upon receipt of the $\overline{WR}$ signal from the CPU. At this time, port A is still in the high impedance status and the data is not yet output to the outside. This signal turns to low level at the rising edge of the $\overline{WR}$ and high level at the falling edge of the $\overline{ACK}$.

$\overline{ACK}$ (Acknowledge input)

- When a low level signal is input to this pin, the high impedance status of port A is cleared, the buffer is enabled, and the data written to the internal output latch is output to port A. When the input returns to high level, port A is made into the high impedance status.

$\overline{STB}$ (Strobe input)

- When this signal turns to low level, the data output to the port from the pin is fetched into the internal input latch. The data is output to the data bus upon receipt of the RD signal from the CPU, but it remains in the high impedance status until then.

IBF (Input buffer full flag output)

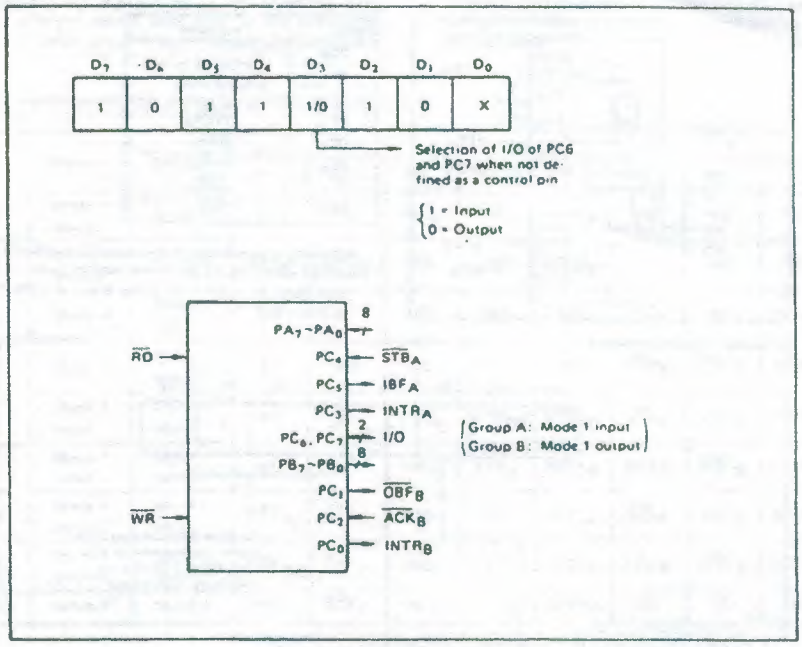
- This signal when turned to high level indicates that data from the pin has been fetched into the input latch. This signal turns to high level at the falling edge of the $\overline{STB}$ and low level at the rising edge of the $\overline{RD}$.

INTR (Interrupt request output)

- This signal is used to interrupt the CPU and its operation in the same as in mode 1. There are two INTE flip-flops internally available for input and output to select either interrupt of input or output operation. The INTE1 is used to control the interrupt request for output operation and it can be reset by the bit set for PC6. INTE2 is used to control the interrupt request for the input operation and it can be set by the bit set for PC4.

Fig 47.

(b) When group A is mode 1 input and group B is mode 1 output



3. Mode 2 (Strobe bidirectional bus I/O operation)

In mode 2, it is possible to transfer data in 2 directions through a single 8-bit port. This operation is akin to a combination between input and output operations. Port C wants for the control signal in this case, too. Mode 2 is available only for group A, however.

Next, a description is made on mode 2.

$\overline{OBF}$ (Output buffer full flag output)

- This signal when turned to low level indicates that data has been written to the internal output latch upon receipt of the $\overline{WR}$ signal from the CPU. At this time, port A is still in the high impedance status and the data is not yet output to the outside. This signal turns to low level at the rising edge of the $\overline{WR}$ and high level at the falling edge of the $\overline{ACK}$.

$\overline{ACK}$ (Acknowledge input)

- When a low level signal is input to this pin, the high impedance status of port A is cleared, the buffer is enabled, and the data written to the internal output latch is output to port A. When the input returns to high level, port A is made into the high impedance status.

$\overline{STB}$ (Strobe input)

- When this signal turns to low level, the data output to the port from the pin is fetched into the internal input latch. The data is output to the data bus upon receipt of the RD signal from the CPU, but it remains in the high impedance status until then.

IBF (Input buffer full flag output)

- This signal when turned to high level indicates that data from the pin has been fetched into the input latch. This signal turns to high level at the falling edge of the $\overline{STB}$ and low level at the rising edge of the $\overline{RD}$.

INTR (Interrupt request output)

- This signal is used to interrupt the CPU and its operation in the same as in mode 1. There are two INTE flip-flops internally available for input and output to select either interrupt of input or output operation. The INTE1 is used to control the interrupt request for output operation and it can be reset by the bit set for PC6. INTE2 is used to control the interrupt request for the input operation and it can be set by the bit set for PC4.

Fig 48.

APPENDIX 82C55

4. When Group A is Different in Mode from Group B

Group A and group B can be used by setting them in different modes each other at the same time. When either group is set to mode1 or mode 2, it is possible to set the one not defined as a control pin in port C to both input and output as a port which operates in mode 0 at the 3rd and 0th bits of the control word.

(Mode combinations that define no control bit at port C)

	Group A	Group B	Port C							
			PC_7	PC_6	PC_5	PC_4	PC_3	PC_2	PC_1	PC_0
1	Mode 1 input	Mode 0	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	I/O	I/O	I/O
2	Mode 0 output	Mode 0	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	I/O	I/O	I/O
3	Mode 0	Mode 1 input	I/O	I/O	I/O	I/O	I/O	$\overline{STB}_B$	IBF_B	$INTR_B$
4	Mode 0	Mode 1 output	I/O	I/O	I/O	I/O	I/O	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
5	Mode 1 input	Mode 1 input	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	$\overline{STB}_B$	IBF_B	$INTR_B$
6	Mode 1 input	Mode 1 output	I/O	I/O	IBF_A	$\overline{STB}_A$	$INTR_A$	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
7	Mode 1 output	Mode 1 input	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	$\overline{STB}_B$	IBF_B	$INTR_B$
8	Mode 1 output	Mode 1 output	$\overline{OBF}_A$	$\overline{ACK}_A$	I/O	I/O	$INTR_A$	$\overline{ACK}_B$	$\overline{OBF}_B$	$INTR_B$
9	Mode 2	Mode 0	$\overline{OBF}_A$	$\overline{ACK}_A$	IBF_A	$\overline{STB}_A$	$INTR_A$	I/O	I/O	I/O
			Controlled at the 3rd bit (D3) of the control word				Controlled at the 0th bit (D0) of the control word			

When the I/O bit is set to input in this case, it is possible to access data by the normal port C read operation.
When set to output, PC7 – PC4 bits can be accessed by the bit set/reset function only. Meanwhile, 3 bits from PC2 to PC0 can be accessed by normal write operation.

The bit set/reset function can be used for all of PC3 ~ PC0 bits. Note that the status of port C varies according to the combination of modes like this.

Fig 49.

APPENDIX 82C55

5. **Port C Status Read**
When port C is used for the control signal, that is, in either mode 1 or mode 2, each control signal and bus status signal can be read out by reading the content of port C.
The status read out is as follows:

	Group A	Group B	Status read on the data bus							
			D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
1	Mode 1 input	Mode 0	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	I/O	I/O	I/O
2	Mode 1 output	Mode 0	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	I/O	I/O	I/O
3	Mode 0	Mode 1 input	I/O	I/O	I/O	I/O	I/O	$INTE_B$	IBF_B	$INTR_B$
4	Mode 0	Mode 1 output	I/O	I/O	I/O	I/O	I/O	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
5	Mode 1 input	Mode 1 input	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
6	Mode 1 input	Mode 1 output	I/O	I/O	IBF_A	$INTE_A$	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
7	Mode 1 output	Mode 1 input	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
8	Mode 1 output	Mode 1 output	$\overline{OBF}_A$	$INTE_A$	I/O	I/O	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$
9	Mode 2	Mode 0	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	I/O	I/O	I/O
10	Mode 2	Mode 1 input	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	$INTE_B$	IBF_B	$INTR_B$
11	Mode 2	Mode 1 output	$\overline{OBF}_A$	$INTE_1$	IBF_A	$INTE_2$	$INTR_A$	$INTE_B$	$\overline{OBF}_B$	$INTR_B$

6. **Reset of MSM82C55A**
Be sure to keep the RESET signal at power ON in the high level at least for 50 μs. Subsequently, it becomes the input mode at a high level pulse above 500 ns.

Note:
After a write command is executed to the command register, the internal latch is cleared in PORTA PORTC. For instance, 00H is output at the beginning of a write command when the output port is assigned. However, if PORTB is not cleared at this time, PORTB is unstable. In other words, PORTB only outputs ineffective data (unstable value according to the device) during the period from after a write command is executed till the first data is written to PORTB.

Fig 50.

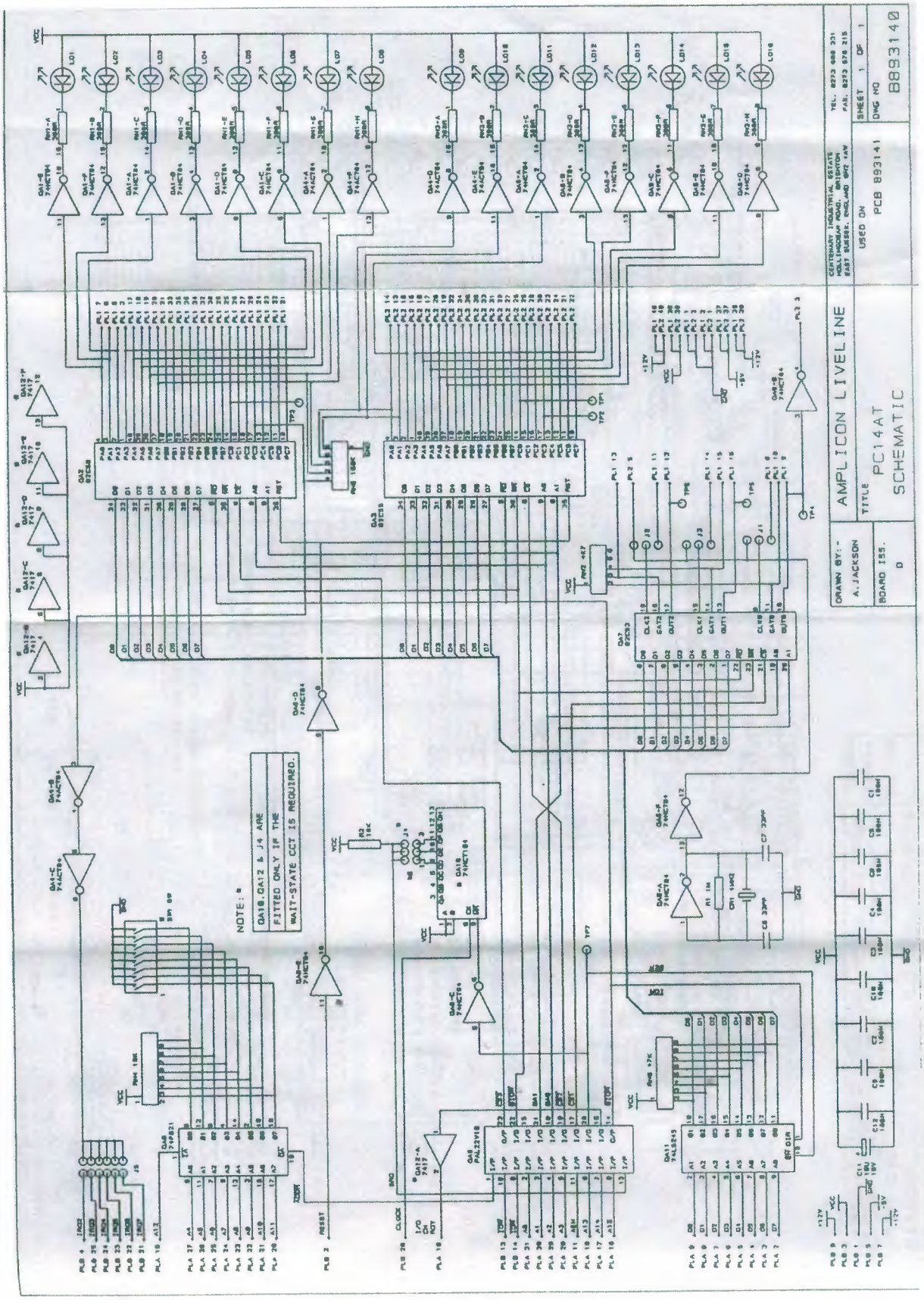

AMPLICON LIVELINE
PC14AT
SCHEMATIC
PCB 893141
889314Ø
A.JACKSON
NOTE: 8
GA18,GA12 & J4 ARE
FITTED ONLY IF THE
WAIT-STATE CCT IS REQUIRED.

Fig 51.

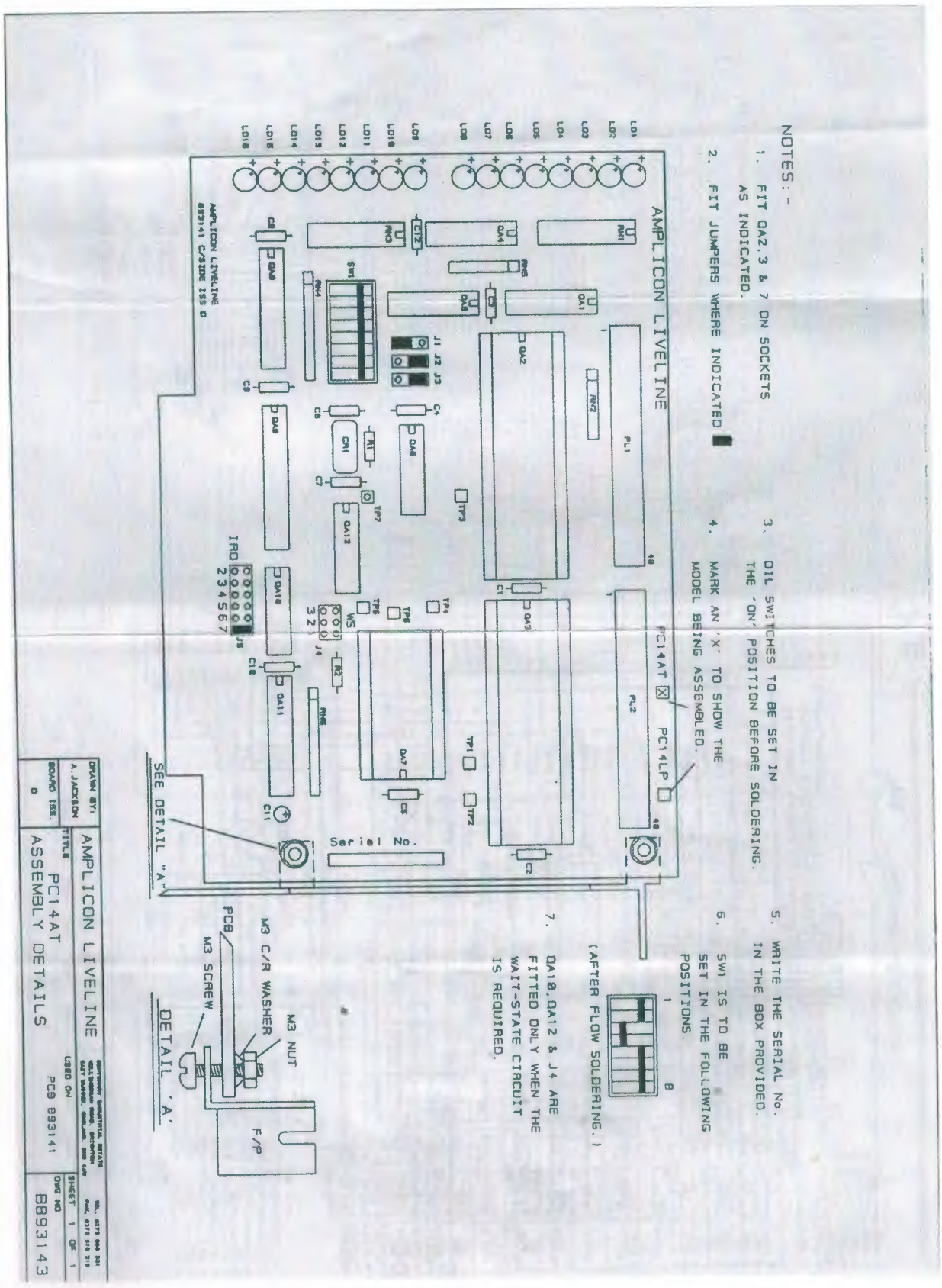

NOTES:-
1. FIT QA2,3 & 7 ON SOCKETS AS INDICATED.
2. FIT JUMPERS WHERE INDICATED
3. DIL SWITCHES TO BE SET IN THE 'ON' POSITION BEFORE SOLDERING.
4. MARK AN 'X' TO SHOW THE MODEL BEING ASSEMBLED.
5. WRITE THE SERIAL No. IN THE BOX PROVIDED.
6. SW1 IS TO BE SET IN THE FOLLOWING POSITIONS.
(AFTER FLOW SOLDERING.)
7. QA10, QA12 & J4 ARE FITTED ONLY WHEN THE WAIT-STATE CIRCUIT IS REQUIRED.
AMPLICON LIVELINE
PC14AT
PC14LP
Serial No.
SEE DETAIL "A"
DETAIL 'A'
PCB
M3 SCREW
M3 C/R WASHER
M3 NUT
F/P
DRAWN BY: A.JACKSON
BOARD ISS. D
AMPLICON LIVELINE
TITLE PC14AT ASSEMBLY DETAILS
USED ON PCB 893141
SHEET 1 OF 1
DWG NO 8893143

Fig 52.

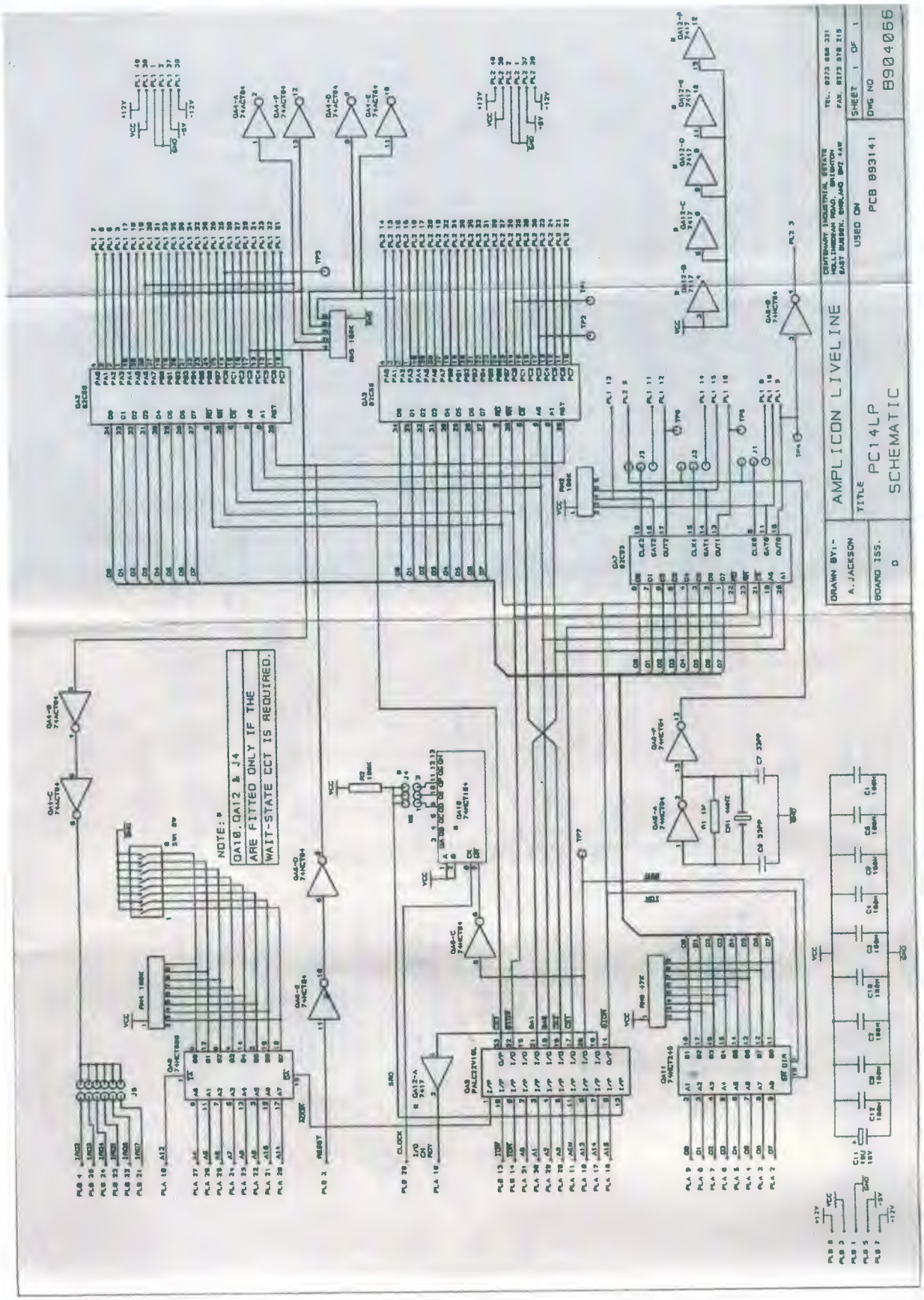

AMPLICON LIVELINE
PC14LP
SCHEMATIC
PCB 893141
8904066
NOTE: # QA10, QA12 & J4 ARE FITTED ONLY IF THE WAIT-STATE CCT IS REQUIRED.

Fig 53.

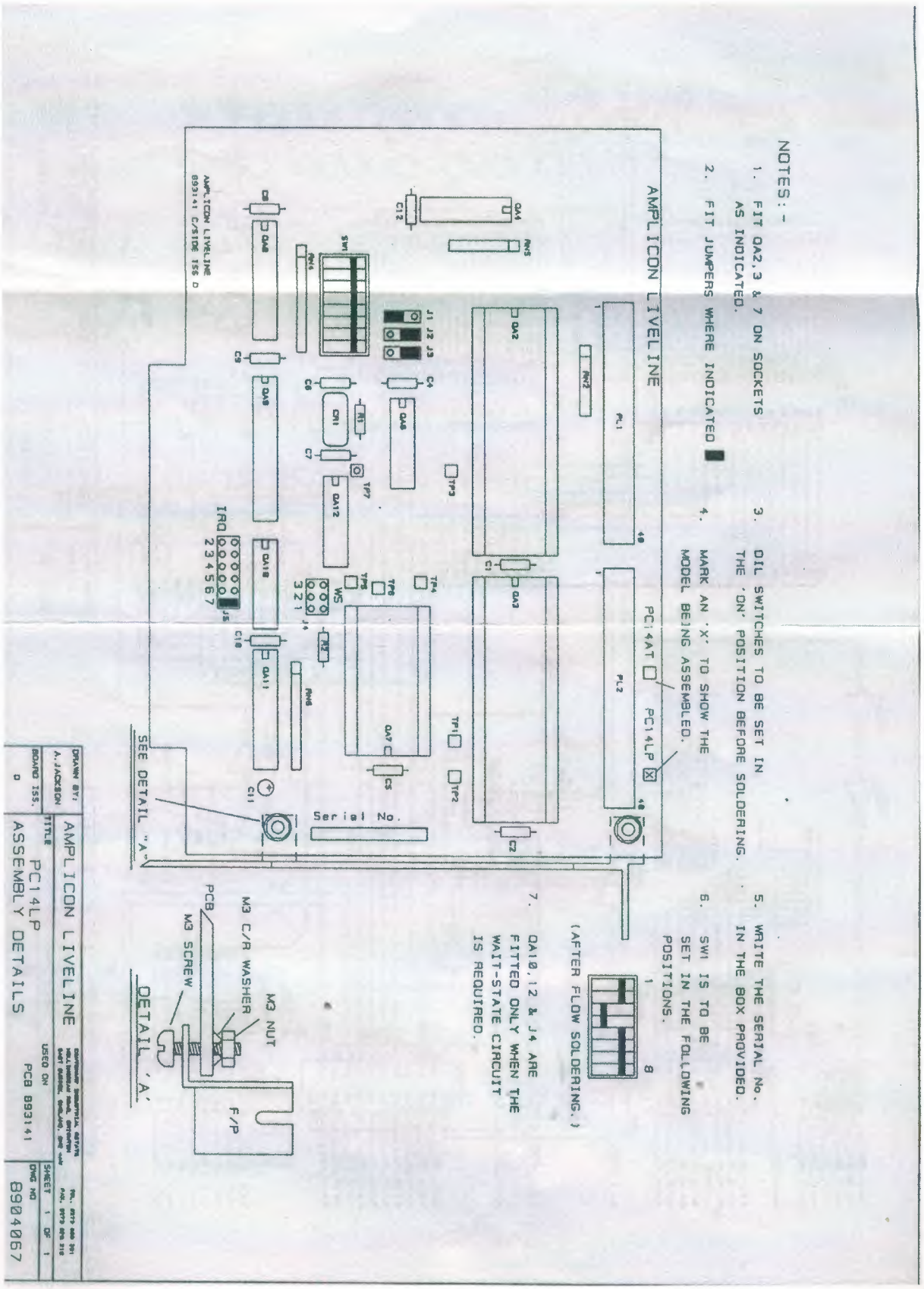

NOTES:-
1. FIT QA2,3 & 7 ON SOCKETS AS INDICATED.
2. FIT JUMPERS WHERE INDICATED ■
3. DIL SWITCHES TO BE SET IN THE 'ON' POSITION BEFORE SOLDERING.
4. MARK AN 'X' TO SHOW THE MODEL BEING ASSEMBLED.
5. WRITE THE SERIAL No. IN THE BOX PROVIDED.
6. SW1 IS TO BE SET IN THE FOLLOWING POSITIONS.
(AFTER FLOW SOLDERING.)
7. QA10,12 & J4 ARE FITTED ONLY WHEN THE WAIT-STATE CIRCUIT IS REQUIRED.
AMPLICON LIVELINE
PC14AT
PC14LP
Serial No.
SEE DETAIL "A"
DETAIL 'A'
M3 NUT
M3 C/R WASHER
M3 SCREW
PCB
F/P
AMPLICON LIVELINE
PC14LP
ASSEMBLY DETAILS
DWG NO
8904067
PCB 893141

Fig 54.