MICRO COMPUTER WITH ADD ON UTILITIES

PROJECT REPORT

SUBMITTED BY

K. BALAJI

S. GAYATHRI

V. JAYAKUMAR

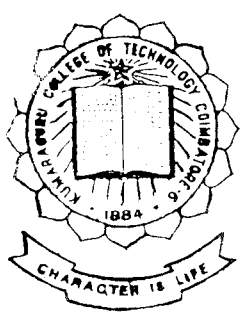
N. SIVASUBRAMANIAM

UNDER THE GUIDANCE OF

MISS. K. KAVITHA RAJALAKSHMI
DEPARTMENT OF ELECTRONICS &
COMMUNICATION ENGG.
KCT.

MR. B. SUGUMAR
SENIOR HARDWARE
CONSULTANT
C.M.C. LTD.

IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE OF **BACHELOR OF ENGINEERING**
IN ELECTRONICS & COMMUNICATION ENGINEERING
OF THE BHARATHIAR UNIVERSITY



DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006

1991 - 1992

CERTIFICATE

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006

PROJECT WORK

Register Number 8827D0173, 177, 181, 193.

This is to certify that the Project entitled

MICRO-COMPUTER WITH ADD ON UTILITIES

has been submitted by

Mr/Ms. K. BALAJI, S. GAYATHIRI, V. JAYAKUMAR, N. SIVASUBRAMANIAM.

In partial fulfilment of the requirements for the award of the degree of **Bachelor of Engineering in Electronics And Communication Engineering** of the Bharathiar University - 641 046

During the Academic Year 1991 - 92

Guide 30.3.92

Head of the Department

Submitted for the Viva-Voce examination held on.....

Internal Examiner

External Examiner

CONTENTS

SYNOPSIS

ACKNOWLEDGEMENT

INTRODUCTION

CHAPTER I KEYBOARD DESCRIPTION

1.1. PARALLEL INPUT / OUTPUT

1.2 TYPES OF INTERFACING DEVICES

1.3 PERIPHERAL I/O INSTRUCTIONS

1.4 OUT INSTRUCTION (8085)

1.5 IN INSTRUCTION

1.6 DEVICE SELECTION AND DATA TRANSFER

1.7 THE INPUT BOARD

1.8 KEY BOARDS

1.9 KEY DEBOUNCE

1.10 KEY DEBOUNCE USING SOFTWARE

1.11 THE WORKING OF THE I/O PORT CIRCUIT

1.12 CONSTRUCTION

1.13 TROUBLE - SHOOTING

CHAPTER II INPUT OUTPUT BOARD

2.1 OUTPUT PORTS

2.2 CIRCUIT DIAGRAM OF THE OUTPUT PORTS

2.3 SEVEN SEGMENT LED

CHAPTER III THE MAIN BOARD

3.1 MICROPROCESSOR ARCHITECTURE AND ITS OPERATIONS

3.2 INTERNAL DATA OPERATIONS AND THE 8085 REGISTERS

3.3 PERIPHERAL OR EXTERNALLY INITIATED OPERATIONS

3.4 RANDOM ACCESS MEMORY

3.5 CIRCUIT DESCRIPTION

CHAPTER IV EPROM PROGRAMMING BOARD

4.1 INTRODUCTION

4.2 PROGRAMMING USING THE 8085 KIT

4.3 CRICUIT DESCRIPTION

4.4 SOFTWARE DESCRIPTION

4.5 8255A PROGRAMMABLE PERIPHERAL INTERFACE

4.6 ERASING THE EPROM

CHAPTER V ADC & DAC INTERFACE

5.1 INTRODUCTION

5.2 TYPES OF ADC CIRCUITS

5.3 FUNCTIONAL DESCRIPTION

5.4 ADC 0804

5.5 CIRCUIT DESCRIPTION

5.6 DAC CIRCUIT

CHAPTER VI AUDIO CASSETTE INTERFACE

6.1 INTRODUCTION

6.2 UART

6.3 SIMPLE CASSETTE INTERFACE

6.4 INTERFACE BOARD DETAILS

6.5 CASSETTE RECORDER DETAILS

6.6 INTERFACE CIRCUIT DETAILS

6.7 CIRCUIT FOR RECORDING INTO CASSETTE

6.8 CIRCUIT FOR RECOVERY OF DATA FROM CASSETTE

6.9 USING CASSETTE INTERFACE

6.10 DESCRIPTION OF THE PROGRAM

6.11 STORE OF THE PROGRAM

CHAPTER VII KEY-NOTES

7.1 SUBROUTINE LOCATIONS

7.2 EPROM PROGRAMMING ROUTINE

CONCLUSION

SOFTWARE

APPENDIX - I

APPENDIX - II

BIBLIOGRAPHY

DEDICATED TO OUR

BELOVED PARENTS

AND

TEACHERS

SYNOPSIS

This project titled µC-WAU [MICROCOMPUTER WITH ADD-ON UTILITIES] deals with the various applications of the newfound versatile electronic chip, microprocessor, which surpasses human capability in performing repetitrous operations with an ease.

This project deals with the design, fabrication and performance of the complete EPROM programming cum copier including the add on utilities that are incorporated with this. We have demonstrated few add-on utilities.

This project has I/P, O/P, memory and logical units similar to the computer and it can be rightly called as microcomputer with add-on utilities.

ACKNOWLEDGEMENT

We are grateful to our respected Principal MAJOR. T.S.RAMAMURTHY who encouraged and motivated us all the way to proceed through and complete this project.

We cordially thank our beloved Head of the Department PROF. K.PALANISWAMI, for being our source of inspiration and for his valuable ideas which helped us meliorate this project

We express our gratitude from the bottom of our heart to our internal guide Miss. K.KAVITHA RAJALAKSHMI, who guided us through the right path of completing this project sucessfully with her timely conceptions.

We are greatly indebted to Mr. B.SUGUMAR, Senior Hardware Consultant, CMC, Trivandram, who guided us externally in spite of his heavy schedules, without whom this project wouldn't be a reality. Special thanks to you Sir !

We express our whole hearted thanks to our lab technicians for their timely services.

We are thankful to all the Staff Members and student friends of all the department who gave valuable suggestions with great care and love at various junctures.

INTRODUCTION

With the field of Electronics extending into never ending new frontiers at an incredible speed and microprocessor setting new trends everyday this project has micrprocessor as its core of the heart.

This project named µC-WAO [MICROCOMPUTER - WITH ADD-ON UTILITIES] uses Intel's versatile microprocessor 8085 with 2716 [EPROM], 6116 [RAM], 8212 [I/O port cum latch] with simple AND, OR, NAND, NOT and NOR gates to achieve the task.

Intel's 8085 has a simple set of software instructions which could be easily understood. In addition, much of the software programs for experimental use and for general purpose industrial control applications can be written early with 8085 instruction set. Depending on one's interest, the kits capability can be easily extended. This performs all the arithmetic operations, can store, can copy and so it can be rightly called as a Micro Computer.

The Add-on utilities which are included in this project are

[MICRO-COMPUTER WITH ADD-ON UTILITIES

1) Audio Cassette Interface and

2) ADC and DAC.

Audio cassette interface enables the user to make use an audio tape as a temporary storage medium which is helpful in case of lengthier programs or programs which are still on the roads of development. This interfacing is achieved with CD 4051 [Analog Mux], CD 4093 [Schmitt NAND] used with an envelop detector and a simple software routine.

The second Add-on utility is the ADC. This has numerous applications wherever an analog signal is to be converted to digital signal. Then utility uses 0804 chip with a simple software. This also includes DAC using 0800 chip.

The project being the basic block of larger computer systems, can be extended to perfrom any task, with sufficient modifications.

CHAPTER I

KEYBOARD DESCRIPTION

The microprocessor receives a series of instructions which are stored in memory. The EPROM memory stores some important instructions necessary for the user to load and read data into the other memory, the RAM. Inorder to make the microprocessor do some specific work, it must have atleast one input port and one output port.

Let us now look into a, microcomputer, or even a simple calculator for that matter, made by using a microprocessor. Its input port is the Keyboard, which consists of a set of keys that can be used to load data or execute commands such as +, -, x, clear etc. The display of a calculator is the external output port. But when we have an eight digit decimal display, it may use in practice atleast two 8 - bit binary output ports. Thus a simple calculator, which can be called a special purpose microprocessor unit, houses two output and one input ports.

The larger the number of input and output ports, the greater is the versatility of the system. The 8085 has provision for 256 input and output ports each. These are provided and accessed by an 8-bit address, using its eight address lines called A0 to A7.

1.1 PARALLEL INPUT / OUTPUT :-

In the parallel I/O data mode, the 8085/8085A accepts eight bits of data on its data bus from peripherals such as switches, hex keyboards and A/D converters. Similarly, it sends out eight bits of data on its data bus to output devices such as LED's, Seven Segment LED's and D/A converters. Each I/O device is assigned a binary address, called a device address or port number, through an appropriate interfacing logic circuit. When the microprocessor executes a data transfer data instruction for an I/O device, it sends the appropriate address on the address bus, sends the control signal, enables the device and transfers data. When these I/O tasks are accomplished by means of Input/Output (IN/OUT) instructions, the process is called peripheral I/O.

When I/O tasks are accomplished by means of memory-related data transfer functions (LDA, STA etc), the process is called memory mapped I/O.

1.2 TYPES OF INTERFACING DEVICES :-

Interfacing devices may be classified into two categories:

1. General Purpose Peripherals

2. Special Purpose Peripherals
   (also known as dedicated function peripherals)

We use the term peripheral for these devices as well as for the other I/O devices, such as printers and floppy drives that would be interfaced to the uP through them.

General purpose peripherals are devices that perform a specific task but may be used for interfacing a variety of I/O devices to the uP. Examples of general purpose peripherals are ;

a) An I/O port

b) Programmable Peripheral Interface (PPI), also known as Peripheral Interface Adapter (PPA)

c) Programmable Interrupt Controller

d) Programmable DMA controller

e) Programmable Communications Interface

f) Programmable Internal Timer.

Special Purpose peripherals are devices that may be used for interfacing a uP to a specific type of I/O device. There are peripherals much more complex and therefore relatively more expensive than the general purpose peripherals. Some of the special purpose peripherals are,

a) Programmable CRT controller

b) Programmable Floppy Disk Controller

c) Programmable Hard disk Controller

d) Programmable Keyboard and Display Interface.

The above description describes the different types of interfacing devices. Interfacing circuit for an I/O device is determined primarily by the instructions to be used for data transfer. An I/O device can be interfaced with the 8085/8085A microprocessor either as a peripheral I/O or as memory mapped I/O. In the peripheral, the instructions IN/OUT are used for data transfer, and the device is identified by an 8-bit address.

1.3 PERIPHERAL I/O INSTRUCTIONS :-

The 8085/8085A microprocessor has two instructions for data transfer between the processor and the I/O devices. The instruction IN (code DB) inputs data from an input device (such as keyboard) into the accumulator, and the instruction OUT (code D3) sends the contents of the accumualtor to an output device such as LED display. These are 2-byte instructions with the second byte specifing the address or the port number of an I/O device. For example, the OUT instruction is typically written as follows.

Memory	Machine	Mnemonics	Comments	Address	Code
2050	D3	OUT 01h	Output accumulator contents to the port with device address 01h.		
2051	01				

If the output port with the address 01h is designed as an LED display, the instruction OUT will display the contents of the accumulator at the port. The second byte of this OUT instruction can be any of the 256 combination of eight bits, from 00h to FFh. Therefore, the 8085/8085A can communicate with 256 different output ports with device addresses ranging from 00h to FFh. Similarly, the instruction IN can be used to accept data from 256 different input ports.

1.4 OUT INSTRUCTION (8085) :-

In the first machine cycle, M1 the 8085 place the high order memory address 20h on A15-A8 and the low order address 50h on AD7-AD0. At the same time, ALE goes high and IO/$\overline{M}$ goes low. The ALE signal indicates the availability of the address on the AD7-AD0 and it can be used to demultiplex

the bus. The IO/$\overline{M}$, being low, indicates that it is a memory related operation. At T2, the microprocessor sends the $\overline{RD}$ control signal which is combined with IO/$\overline{M}$ to generate the $\overline{MEMR}$ signal, and the processor fetches the instruction code D3 using the data bus. When the 8085 decodes the machine code D3, it finds out the instruction is a 2 byte instruction an and that it must read the second byte.

In the second machine cycle, M2 (Memory read), the 8085 places the next address, 2051h, on the address bus and gets the device address 01h via the data bus.

In the third machine cycle, M3 (I/O Write), the 8085 places the device address 01h on the low order(AD7-AD0) as well as the high order (A15-A8) address bus. The IO/$\overline{M}$ signal goes high to indicate that it is an I/O operation. At T2, the accumulator contents are placed on the data bus, followed by the control signal $\overline{WR}$. By ANDing the IO/$\overline{M}$ and $\overline{WR}$ signals, the $\overline{IOW}$ signal can be generated to enable an output device.

1.5 IN INSTRUCTION :-

When the microprocessor executes the instruction IN to accept data from an input port, events similar to the execution of the OUT instruction takes place. During the machine cycle M1, the address of input port is placed on the address bus, the control signal IOR is sent to enable the port and data are transferred from the input device (such as keyboard) to the accumulator via the data bus.

1.6 DEVICE SELECTION AND DATA TRANSFER :-

In general, peripherals are connected in parallel on the data and address buses. To select an appropriate peripheral the device address on the address bus and the control signal during the M3 cycle can be used as follows:

1. Decode the address bus to generate a unique pulse corresponding to the device address on the bus. This is called the device address pulse.

2. Combine (AND) the device address pulse with the control signal to generate a device select pulse that is generated only when both signals are asserted.

3. Use the device select pulse to activate the interfacing device (I/O port)

The various interfacing units that we have used are:

i. Input / Output ports
   ie., Keyboard and display unit

ii. EPROM programmer and copier

iii. RAM-memory board (upto 10K)

iv. Ports and memory expandable board

v. A/D and DAC (mulitchannel board) etc.,

The project helps us to built any electronic control circuitaries for ant applications from simplest I/O intergface to robotics and various A-I units. Let us explain each item quoted above one by one in a simpler way. The project includes few interfacing units. The report describes about each unit assuming to some of the following.

What is its necessary for such interfacing device.

Explain about the hardware.

Explanation about the software and various control signals.

Typical troubleshooting problems:

What improvement that can be done over the existing.

Let us start with keyboard interfacing.

1.7 THE INPUT BOARD :-

The input data which consists of sets of 8-bits binary information, can pertain to the software instructions or to number of data. For inputting data in the form of numbers, a keyboard consisting of a set of small push to close keys or switches are necessary. In microprocessor terminology, numbers are not dealt within the decimal way but in binary form.

1.8 KEYBOARDS :-

Sometimes small keyboards with only a fewkeys (similar to touch-one telephone dialing keyboards) are used in industrial applications. These small assemblies of keys are generally called keypads.

When an operator depress a key, electric signals must be generated which will unable the device (or other device) to determine which keys was depressed. This is called encoding. The encoding process is dependent on the mechanism used to make the induvidual keys in the keyboard.

The most desired method for encoding is based on the use of keyboard swutches, which contain a switch similar to the push button switch used in many electric devices. When the plunger is depressed, the contacts of the switch in the housing are closed, and the two terminals at the output are effectively connected. When the plunger is up, the switch in the housing is open, and the terminals are not electrically connected.

It is a good idea to load the values on the output line into a flip-flop register before the device reads the outputs. This has the advantage of storing the values until the device can read them, particularly if the keyboard operator raises the key before the device can respond.

In which a flip-flop is used on each output, and a strobe is generated to load the flip-flops, by using a delayed inverted pulse signal generated whenever anyone of the output lines from the encoder goes high. The delay is inserted to compensate the signal slieuing, where signals arrive at the output lines at different times because of differing delays through the wires in the system. The delay must be adequate to accomadate the largest delays that may occur. Also, the length of the strobe pulse should be short compared to the shortest line a key might be depressed. (A delay of 1ms and a pulse of 1ms would be reasonable).

Here we are assuming that the switch contact do not bounce, as is the case with some switch. If the contacts do bounce, the output signals must be "smoothed" and various circuits are available.

The basic division of keyboard is (1) the electromechanical keyboard, which includes the switch type just explained and (2) the solid state keyboard.

There are several basic mechanisms for solid-state keyboards. Capacitor type have mechanisms which vary the capacity of a capacitor when a key is depressed. These are low-cost keys, often used in keypads and other cost-conscious keyboards. Hall-effect keyboards are more expensive, but have long life and good key feel, as do ferrite core and photo optic keyboards. Each of the basic mechanisms has different problems with regard to encoding the keyprinters output into a coded form usable by a computer.

IC packages for encoding are made by several manufacturers and can include smoothing or debouncing for contacts and sometimes key rollover protection, which protects against two keys being depressed at the same time.

1.9 KEY DEBOUNCE :-

It is necessary that the bouncing of the key should not be read as an input. The key bounce can be eliminated from input data by the key-debounce technique, using either hardware or software.

1.10 KEY DEBOUNCE USING SOFTWARE :-

In the software technique, when a key closure is found, the microprocessor waits for atleast 10ms before it accepts the key as an input.

The uP begins by reading the input port and checks whether the previous key has been released. This will eliminate the problem that would otherwise occur if someone were to press a key and hold it for a long time. Once the key is released, it is debounced by waiting 10ms. The program reads the keyboard and checks for a key closure. If a key closure is found, it is debounced again by waiting 10ms and the binary code corresponding to the key pressed found either by the table look-up procedure or by setting up a counter.

Our project uses a hexadecimal keyboard with discrete IC encoding process. It uses no matrix type or, programmable I/O interface. Our circuit is simpler and easy. Let us see how it works.

The keys or switches in the keyboard, when pressed, will indicate the corresponding data word for that key to appear at the 8-lines entrance of the inout port. For this a decoding circuit is necessary. For instance, if the key numbered 3 is pressed, the word must be decoded as 0000 0011. For the key A, the word should be 0000 1010.

The keyboard can be hexadecimal or an octal type. Industry standard is now using hex format. Hence a hex keyboard, which needs, 16 keys, is recommended for the kit. Sixteen push switches and one - shift - key like extra push switch (command key) are required. Five more switches are also mounted on the keyboard and these are to be connected to the microprocessor board later. One end of all the (16+1) switches are connected together to Earth terminal. The other ends, which consists of 16 wires and one extra wire from the command key are to be wired by a flat ribbon cable to the circuit board.

The 16 keys are for selecting the hex characters 0 to F. The command key is not to be pressed while introducing these hex characters. If the command key is pressed together with any one of the keys 0 to 5, then it inputs the word that is interrepted (by the EPROM software on the microprocessor board) as one of several command functions such as "data store and increment memory", "set high address", "set low address", go to execute the program" etc.,

These commands are necessary to load the program from the keyboard by keying the instruction words one-by-one, checking it as it gets stored in RAM memory and finally asking the microprocessor chip to execute them either in full or part. Of course, the results of whatever program that is executed will show up on the display via the output ports.

1.11 THE WORKING OF THE INPUT/OUTPUT PORT CIRCUIT :-

The decoder consists of eight - input NAND gate ICs (7430). For example, gate A has 8 input lines connected to the keys, 1-3-5-7-9-B-D-F. A look at the hexadecimal table will show that all these numbers give the last bit as '1'. Hence the output of the NAND gate becomes hicg whenever any one of these keys is pressed.

Like wise, the remaining 8-input NAND gates B, C and D decode the second, third and fourth bits of the hexadecimal 4 - bit word. The keyboard generates one hexadecimal word. It might be recalled that the data word is an 8 bit word and the hex word from the keyboard gives only a 4 - bit word. So to get the 8 - bit word, say pertaining to an instruction we must use the keyboard two lines. By making use of a suitable shifting instruction in the microprocessor software program, the 8-bit word will be generated as two "nibbles".

The four outputs from gates A,B, C and D are led to the entrance of the input port IC 74126. In addition, the four invertor gates of 7404 and the "0" key are all given to the inputs of the NAND gate E. So when anyone of the outputs A, B, C and D go high, or if the "0" key is pressed, the gate E has its output going high. This is connected to the other 74126 IC of the input port to be output as the bit D7 during the Read pulse.

Further, if the command key (CK) is pressed, the bit D6 will be low, otherwise it remains high. Thus when the D6 data bit is high, the 4-bit word D0-D3 that is input is considered as a hex number. When D6 is low, ie., when command key is pressed, the 4 - bit word D0-D3 is considered to be representing any of the command word instructions.

The bit D7 will always go high if any one of the 16 Keys is pressed. Thus it indicates to the microprocessor whether at all a key has been pressed. Such an information is sometimes called as a "Strobe" bit.

Since the remaining bits D4 and D5, are not used, they are left free. These bits will be read as 1's since their inputs (to the 74126) are left free.

The two 74126 IC's are used to create an 8-bit input port. The 74126 is only a buffer IC. Each of the four buffers in a 74126 has a control input line, which enables the buffer to transfer the bit at the input to the output when the control lines goes high momemtarily. Thus, these control lines of all the buffers in these two IC's are connected

together and brought out for giving the read pulse from the microprocessor.

The buffers isolate the microprocessor from the input lines except during a reading operation. This read pulse must be a positive going pulse for the 74126 IC's. In fact, this pulse would be generated by the circuitry in the output port using the $\overline{I/OR}$ pulse from the microprocessor, together with the address lines, using suitable gates.

The input port in this kit is given the address 02. Thus, the instruction IN 02 would cause the data from the keyboard to be read. Any other, say IN 05 instruction, would choose only an address-5 port (if such a port is actually constructed and wired-up), and it would read from that port then.

Additional input ports can be constructed by the user using similar 74126 or 74LS126 IC pairs, and interconnected to decoded keyboards, DIP switches, analog to digital converters etc. An anlog to digital converter is useful to convert any analog signal such as voltages into number for subsequent processing in the microprocessor by a suitable program.

1.12 CONSTRUCTION :-

The keyboard is arranged on a flat Hylam sheet, using a pattern of 5x4 keys arranged in 4 coloumns and 5 rows, with a spacing of 2 cm in between them. The command key (CK) is fitted at the bottom to the right and the reset key (to go to the microprocessor board) to the bottom left. One end of all the 16 number keys and CK key go to ground. The other ends are connected using a multi-wire flat ribbon cable to the PCB. The four extra keys are needed for Interrupt functions, and they are used with the microprocessor board.

This PCB can be easily wired using either a general purpose IC type of printed board by point-to-point linking sockets are not necessary for the TTL ICs used in this board but it is advisable to provide sockets for all.

1.13 TROUBLE SHOOTING :-

A simple LED probe is needed for testing. This can be made by connecting a small 330 ohm resistor (¼ W) in series with a simple LED. The 5V supply is connected to the board and to the probe. Press any one key and see if the entrance points of the 74126 get the data, according to the key pressed. The LED probe will be lit if it is a "0" and will not be lit for a "1".

An alternative probe using 7404 invertor can be used if one wants to get the indication of LED lit with logic "1" and not lit when logic is "0". Then ground the read pulse point of the input port. The data will not appear at the data bus points for the key pressed, because the 74126 ICs are then in the open-circut state or the so called "Tri-state" condition.

CHAPTER II

THE INPUT OUTPUT BOARD

After considering the circuit diagram of the keyboard and the Input board, let us now study the circuit diagram of the output port.

2.1 THE OUTPUT PORTS:-

It is possible to have 256 output ports using the 8-bit address lines A0-A7 from the microprocessor. Usually, such a large number of ports for Input/Output are not needed for simple microprocessor applications.

We shall have three output ports in the unit described for construction of these, two will be used for the display that indicates data and address using the monitor program. The third port, which is extra is meant for experiments using the kit.

So far the inside details of the program have not been discussed sufficiently. We shall do this step by step, in a practical way.

We have already discussed that every 8-bit data which goes into or out of the microprocessor passes via the eight lines of the data bus. This bus has a bi-directional character because it passes data into or out of a microprocessor. The data is passed as per instruction whether it is an input or read instruction or an output or write instruction. Note that "Input and Output" refer to the ports while "Read and Write" refer to the memory. Both ports and memory are external to the microprocessor, and the data bus links of of them.

The address lines of the microprocessor enable the proper memory address to the decoded, and hence the data can be written into or read from that particular memory location. For example, the instructions

LDA 16 04

cause the accumulator to be loaded with the contents of the memory whose where address is 0416 (hex) or 0000, 0100, 0001, 0000 (16 bit binary address) A15..A0.

Thus, a memory of upto 64 K Words (1K=1024) can be

accessed. Such a memory size is really needed if one builds a microcomputer of some power using 8085. Simpler systems either for a microcomputer or industrial control, or other uses, can do with lesser memory. For quite simple and devoted applications, ordinarily 2K work memory is satisfactory.

2.2 CIRCUIT DIAGRAM OF THE OUTPUT PORTS :-

An output port latches (Catches and stores) the data bus word the moment an OUT instruction is executed by the microprocessor. For this purpose, the $\overline{IO/W}$ line (input or output write) coming from the microprocessor is used in conjunction with the address of the port to generate a pulse that causes D-flip-flops of a latch to catch and store the data word from the data bus.

The data on the microprocessor data bus keeps on changing as it executes instruction after instruction at a fast pace. As it executes, say the OUT 01 instruction, the relevant data appears on the data bus and the address 1 appears as high on the address bus. The output port 1 must be wired to get a strobe pulse when the address line 1 and also when the $\overline{I/OW}$ pulse are active.

Likewise the OUT 02 instruction will make address bit A1 high along with the active $\overline{I/OW}$ low signal. These two must together give a strobe pulse to the port 2 latch to catch and store the data. If we donot have a port 2 wired to the system, then the data will just appear on the data bus while executing the out 02 instruction but it will be lost, as it cannot be latched or stored.

We have to decode the address lines for port selection. In our kit, as mentioned earlier, there are basically three ports and one may add some more ports later. We are not however, going to use the full provision of 256 ports in any case. Hence we need not use all the address bits. So we may decode only the bits A7,A2,A1 and A0 and forget about the bit A3-A6. We therefore use only 4 active address bits here for decoding and selecting the I/O ports, the remaining are left undecoded.

With A7 low, we can have a combination of eight ports using the three (A0, A1 and A2) lines. Thus, with A7 low and A0=0, A1=1, A2=0 we get port 2. with A7 low and A0=1, A1=1, A2=0 we get port 3 and so on. With A7 high, we can get

eight more possible address for accessing another set of eight ports. For instance, the EPROM programmer board, uses the A7=high address combination.

The output ports used in the hardwiring of the kit uses only A7 low combination. The A7 high combination is left free for the user to expand the kit by adding more ports by an exactly similar circuit like this, but with A7 high instead of low. An inverter from the A7 line to feed pin no.12 of the 7442 is to be added then.

Referring to the circuit of the output port, the 7442 is used for decoding the address lines A7, A0, A1 and A2. It has ten outputs, and one of these goes low for each selected address. For example, when A7-A2-A1-A0 are 0010 (address 2), pin no.3 of the 7442 goes low. This going low information is combined with $\overline{I/OW}$ low going pulse from the microprocessor in one NOR gate (a quarter of 7402) whose output goes momentarily high. This high going pulse is connected to the clock output of the four D-flip-flops connected in the 7475 ICs. A 7475 contains such latches in it and so two 7475's are needed to latch the eight bits of the 8-bit data bus. The data bus lines are connected to the

D-inputs of the 7475's flip-flops. Six such 7475's are needed for the three output ports on the kit.

The port addresses selected are 1, 2 and 4 for these ports wired in the kit. Port 1, is used for wiring to the digit cathodes of the eight of digits of the multi-digit LED display. Of course, only six of these digits are used, because we have to provide two gaps in between.

Hi Add		Low Add	Data
0 4	-	0 0	C 2

The flip-flops of the 7475s used in port 1 sink the currents from the cathodes of these digits through eight BC 147B transitors. Port 2 is used for wiring to the seven segments (and decimal point) of each LED display, ie., the anodes. For example, if port 1 has a latch information as

0 0 0 0 0 0 0 1

and port 2 has the information

0 0 0 0 1 1 0 0
g f e d c b a .

(a to g denote the segments of LED).

then the display will be blank for all but the last digit

and it will show a 1 as below (segments b and c alone are lit).

- - - - - - - 1

By using software in our monitor program (kept stored in the EPROM at 0000 to 01FF), we make repeated (one be one) display of the relevant information in all the eight digits. Due to persistence of vision, however, we find the display's all eight digits to be glowing continously. This requires a scanning program that sequentially presents the bits one by one to the digits and does it repeatedly at a rate which shows no flicker. Such a display can be called a "Software Scanned Display".

For converting the binary data into the hexadecimal format also we make use of software in the monitor. A table stores the information as to which segments should glow for each hex digit. For example, for number 1, segments b and c may glow. And for A, all but segment d may glow, and so on. By having a table the data (of 16 rons) the data "nibble" can be converted into hex form, before outputting it through 2, that causes the LED segments to be lit accordingly.

Output port 4 is used for lighting the eight LED's

current limit resistors R1 are used for each LED. The $\overline{Q}$ outputs of the 7475 are used in this case so that the LEDs may glow for a '1' bit and be dark for a '0' bit seven current limiting resistors (18 to 220 ohm, ¼ W each) are used for the segements of the multi digit display.

2.3 SEVEN SEGMENT LED :-

A seven segment LED consists of seven light emitting diode segments and one segment for the decimal point. To display a number, the necessary segments are lit by sending an appropriate signal for current flow through diodes. For example, to display an 8, all the segments must be lit.

Seven segment LED's are available in two types: Common Cathode and Common Anode. Current flow in these diodes should be limited to 20 milli amps.

The seven segments A through G, are usually connected to data lines D0 through D6 respectively. The binary code required to display a digit is determined by the type of the seven segment LED, the connections of the data

lines, and the logic required to light the segment. For example to display digit 7, at the LED the requirements are as follows.

1. It is a common anode seven segment LED and a logic '0' is required to turn on the segment.

2. To display digit seven, segement A, B and C should be turned on.

3. The binary code should be,

Data lines	D7	D6	D5	D4	D3	D2	D1	D0	
Bits	X	1	1	1	1	0	0	0	= 78 + 1
Segments	NC	G	F	E	D	C	B	A	

The code for each digit can be determined by examining the connections of th data lines to the segements and logic requirements.

To display ASCII characters the following code words are used.

? A6 , A EE , B F8 , C 72 , D BC , E F2 , F E2 ,

G 7A , H EA , I 08 , J 3C , K EC , L 70 , N A8 ,

O B8 , P E6 , Q B9 , R A0 , S DA , T F0 , U 7C ,

V 38 , X A4 , Y DC , Z B6 , - 80 , Ø 00 ,

0 7E , 1 0C , 2 B6 , 3 9E , 4 CC , 5 DA , 6 FA ,

7 0E , 8 FE , 9 CE .

CHAPTER III

THE MAIN BOARD :-

The microprocessor cum memory board can now be assembled and used along with them. In this main board, we have the microprocessor and its associated IC's and the memories-the EPROM and RAM.

A Glass Epoxy type printed circuit board is necessary for the main board. Full size PCB layout has been drawn so that readers may follow the same conveniently. Since a double sided PCB layout (which is common in microprocessor kits) has not been employed, a good number of jumper connections are required to be make, as indicated on the component side diagram shown. Sockets are to used for all the IC's, including the 74-Series IC's to avoid trouble later while checking up.

After inserting the sockets, the jumpers shown should be wired with thin hook-up wires. Then the power supply pins of the IC's should be connected to the +5V and Ground.

The 4 MHz or 3.5 MHz crystal must be soldered

between pins 1 and 2 of the microprocessor IC 8085. The crystal must be handled with care. Overheating its pins or dropping on the floor will surely damage the crystal

A 0.02 mfd ceramic capacitor must be connected on the PCB between +5V and Ground pins. It may be necessary to connect a 22pf ceramic capacitor between pin 2 and Ground of the 8085.

The ICs used for this board are all quite expensive. They should be handled with care while inserting into their sockets, after completing the board wiring.

3.1 MICROPROCESSOR ARCHITECTURE AND ITS OPERATIONS :-

The microprocessor is a programmable logic device designed with registers, flip-flops and timing elements. The microprocessor has a set of instructions designed internally, to manipulate data and commnunicate with peripherals. This process of data manipulation and communication is determined by the logic design of the microprocessor called the architechture.

All the various functions performed by the microprocessor can be classified in three general categories.

1. Microprocessor initiated operations.
2. Internal data operations.
3. Peripheral initiated operations.

The microprocessor functions listed above are explained here in relation to 8085 MPU.

MICROPROCESSOR INITIATED OPERATIONS :-

The MPU performs primarily four operations:

a. Memory Read : Reads data from memory
b. Memory Write : Writes data into memory
c. I/O Read : Accepts data from input devices
d. I/O Write : Sends data to output devices

The 8085 MPU performs these functions using three sets of communication line called buses; the address bus, the data bus and the control bus.

ADDRESS BUS :-

The address bus is a group of sixteen lines generally identified as A0 to A15. The address bus is unidirectional; bits flow in one direction - from the MPU to peripheral devices. The MPU uses the address bus to perform the first function identifying a peripheral or a memory location.

In a computer system, each peripheral or memory location is identified with a binary number, called an address and the address bus is used to carry a 16-bit address. The 8085 MPU with its sixteen address lines is capable of addressing 2 16 = 65536 (generally known as 64K) memory locations. Most 8-bit microprocessor have sixteen address lines.

DATA BUS :-

The data bus is a group of eight lines used for data flow. These lines are bidirectional - data flow in both directions between the MPU and peripheral devices. Transferriing the data is performed bu the MPU using the data bus.

The eight data lines enable the MPU to manipulate 8-bit data ranging from 00 to FF (2 8 = 256 numbers). The largest number that can appear on the data bus is 1111 1111. The data bus determines the word length and register size of a microprocessor. Thus the 8085 microprocessor is called an 8 bit microprocessor.

CONTROL BUS :-

The control bus is comprised of various single lines that carry synchronization signals. It provides the timing signals. These are induvidual lines that provide a pulse to indicate an MPU operation. The MPU generates specific control signals for every operation it performs. These signals are used to identify a device type with which the MPU tends to communicate.

3.2 INTERNAL DATA OPERATIONS AND THE 8085 REGISTERS :-

The internal architecture of the 8085 microprocessor determines how and what operations can be performed with the data. These operations are,

1. Store 8-bit data
2. Perform arithmetic and logic operations
3. Test for conditions
4. Sequence the execution of instructions
5. Store data temporairily during exection in the defined R/W memory locations called the stack.

To perform these operations, the microprocessor requires registers, an arithmetic logic unit and control logic and internal buses.

REGISTERS :-

The 8085 has six general purpose registers to perform the first operation, that is to store 8-bit data during a program execution. These registers are identified as B, C, D, E, H and L. They can be combined as register pairs, BC , DE and HL - to perform some 16-bit operations.

These registers are programmable, meaning that a programmer can use them to load or transfer data from the registers by using instructions. For example, the instruction MOV B, C transfers data from register C to register B.

ACCUMULATOR :-

The accumulator is a 8-bit register that is part of the Arithmetic Logic Unit. This register is used to store 8-bit data and to perform arithmetic and logical operations. The result of an operation is stored in the accumulator. The accumulator is also identified as register A.

FLAGS :-

The ALU includes five-flops that are set or reset according to data conditions in the accumulator and other registers. It is used to perform the testing for data conditions.

For example, after an addition of two numbers, if the sum in the accumulator is larger than eight bits, the flip-flop is used to indicate a carry, called the carry flag is set to one. When an arithmetic operation results in zero, the flip-flop called the zero-flag (Z) is set to one. The 8085 has five flags to indicate five different types of data conditions. They are called Zero (Z), Carry (CY), Sign (S), Parity (P) and Auxillary Carry (AC) flags. The most commonly used flags are Zero and Carry.

PROGRAM COUNTER :-

This 16-bit register deals with the fourth operation, sequencing the execution of instructions. This register is a memory pointer. Memory location have 16-bit address, and that is why this is a 16-bit register. The microprocessor uses the register to sequence the execution of instructions. The function of the program counter is to point to the memory address from which the next byte is to be fetched.

STACK POINTER :-

The stack pointer is also a 16-bit register used as a memory pointer, initially, it will be called the stack pointer register to emphasize that it is a register. It points to a memory location in R/W memory, called the stack. The begining of the stack is defined by loading a 16-bit address in the stack pointer.

3.3 PERIPHERAL OR EXTERNALLY INITIATED OPERATIONS :-

External devices (or Signals) can initiate the following operations;

RESET:- When the reset is activated, all internal operations are suspended and the program counter is cleared. Now the program execution can again begin at the zero memory address.

INTERRUPT :- The microprocessor can be interrupted from the normal execution of instructions and as fed to execute some other instructions called serive routine. The microprocessor resumes its operation after completing the serive routine.

READY :- The 8085 has a pin called READY. If the signal at this READY pin is low, the microprocessor enters into a wait state.

HOLD:- When the HOLD pins is activated by an external signal, the microprocessor relinquishes control of buses and allows the external peripheral to use them.

MEMORY :- Memory is an essential component of a microcomputer system. It stores binary instructions and data for the microprocessor. The 8085 microcomputer has two types of memory R/W M (Read / Write Memory) and EPROM (Erasable Progammable Read Only Memory).

The R/W memory is made of registers, and each register has a group of flip-flops that stores bit of information.

The second type of memory, the ROM, stores information permanently in the form of diodes; a group of diodes can be viewed as a register. The MPU can only Read information from the ROM; it cannot write into this memory.

8085 employes EPROM in which the information stored in semipermanent. All the information can be erased by exposing the memory to ultraviolet light through a quartz window installed on the chip.

INPUT / OUTPUT :-

The remaining components of the microcomputer system are Input / Output devices. The MPU communications with the "Outside World" through such devices. The MPU accepts binary data as input from devices such as keyboard or floppy diskettes and sends data to output devices such as LED's or printers. The two methods by which the MPU identifies and communications with the I/O devices are peripheral I/O and Memory-Mapped I/O.

In peripheral to direct I/O the MPU uses eight address lines to send the address of an I/O device. And in case of Memory Mapped I/O the MPU uses sixteen address lines to identify the I/O device.

3.4 RANDOM ACCESS MEMORY:-

Read / Write Memory is popularly known as Random Access Memory. This memory is volatile, meaning that when the power is turned off, all the contents are destroyed. Two types of R/W memories are available, static and dynamic.

Static memory is made up of flip-flops and stores a bit as a voltage. Dynamic memory is made up of MOS transistor gates, and it stores a bit as a charge.

The eight address lines A7-A0 are connected directly to the address lines on the memory chip to identify 256 memory locations. The address lines A15 to A8 are used to select the memory chip through a 3 to 8 decoder and logic gates. Identifying the memory map is a two step process. The first step is to recognize the logic levels required on the address lines A15-A8 to select the memory chip. The second step is to examine the possible logic level combinations that can be assumed by the address lines A7-A0.

When R/W M chip is selected, the logic levels on the address lines A15-A8 should be as follows.

A15	A14	A13	A12	A11	A10	A9	A8	
0	0	0	0	0	1	1	1	= 07h

The memory chip has eight address lines that can assume 256 different combinations from 00h to FFh. Therefore, the memory map of the chip ranges from 0700h to 07FFh.

To read from and write into this memory, one control signal $\overline{MEMW}$ is necessary,. When the $\overline{MEMW}$ is high, data can be read just by selecting the chip. The 8085 places the address of the memory location into which it intends to write on the address bus at T1, and causes the IO/$\overline{M}$ signal to go low at the same time. At period T2, it places the data on the data bus and sends the $\overline{WR}$ signal. During T2 and T3, the memory location is identified and the data are written into the location. The memory write cycle is in many ways similar to memory read cycle.

3.5 CIRCUIT DESCRIPTION :-

The 8085 IC is heart of the circuit. It is the LSI microprocessor chip, and needs only single +5V supply for its working. It has a built-in timing oscillator and works by connecting a crystal between its pin Nos 1 and 2. The frequency upto which it can work is generally 6 MHz, but we are using a 4 MHz crystal (or even an inexpensive (TV) 3.7MHz crystal used for clock ICs). Even though the system, in this way runs at a lesser speed in one sense, we can use slower speed memory ICs, like 6116 RAM and the 2716 EPROM.

The 40 pins of the 8085 are for address lines, data lines, serial input and output, interrupt pins, Hold and Hold Acknowledge. Resetting input and output as well as status signals and control signals for accessing the memory ICs and Input/Output ports.

The lines AD0 to AD7 carry both address and data information together on a time sharing basis. The moment during which address information is present on the lines is synchronously given by the pulse coming from pin No.30 - the Address Latch Enable pin (shown as ALE in circuit diagram). So by latching this information at this instant on an 8-bit latch consisting of 8-D flip-flops the address information is continously available on the eight outputs of the flip-flops. THe 8212 IC is used for this purpose. The inputs to this are the lines AD0 to AD7. The latched address information A0 to A7 comes out as eight lines from the 8212. The AD0 to AD7 are now useful as data lines D0 to D7 which go to the data bus.

The address lines A8 to A15 are coming continously from the pins 21,22,23,24,25,26,27 and 28. We are not going to use all the address lines but confine to only 16K of memory

for the basic kit and its possible expansion. (A 1116 address lines can deal with 64 K of memory). So the address pins A15 to A14 are not used.

The main board also contains the essential EPROMs and RAM, each of 2K. The EPROM is a 2716, and it must have been pre-programmed with data representing the monitor program. The monitor program must be programmed into a fresh 2716 and then only inserted into the socket of the board. An EPROM programmer should be available for this purpose. The program listing which represents the monitor software is given later, and this must be programmed into the first 512 locations of its memory (i.e., 0000 to 01FF). The remaining locations are not used, and need not be programmed with any data; they have FF in the unprogrammed state. They can be used for other utility programs subsequently.

The purpose of the monitor program is to enable the keyboard to work. The data and programs are loaded using the program only.

The control output pins of the 8085 are IO/$\overline{M}$ (pin34), $\overline{RD}$ (pin32), $\overline{WR}$ (pin31) and other control output pins S0 and S1 (pins 29 and 33 respectively) are not used in the kit. The IO/$\overline{M}$ pin tells whether at any moment, the data refers to a port (ie input or output) or to a memory location. If this pin voltage is high, it is the former, if it is low, then it means the latter. The $\overline{RD}$ and $\overline{WR}$ signals are low while some data is being read or is output to either a port or a memory location.

These three signals are seperated for convenience into the following four signals.

$\overline{IO/R}$ - the signal which goes low for the moment any port is being addressed for a "reading from port" operation.

$\overline{IO/W}$ - the signal which goes low for the moment any port is being addressed for outputting a data word from the microprocessor's accumulator.

$\overline{MR}$ - the signal which goes low for the moment any memory location is being addressed for reading the data stored in the memory location, and

$\overline{MW}$ - the signal which goes low for the moment any memory location (8-bit) is being addressed for storing the data from the microprocessor's internal register into that location.

For separating the above signals from the I/OM, RD and $\overline{WR}$ signals, the two ICs 7402 and 7404 are used.

The 74LS138 is used for selecting the EPROM or RAM, depending on the address. We assign the address coding to these in the following way. The lines A0 to A7 represent 256 words. A8 and A9 allow upto 1K; A10 gives upto 2K.

A12	A11	Comments
0	0	First 2K
0	1	Second 2K
1	0	Third 2K
1	1	Fourth 2K

The additional line A13 can then be used to select two such sets of 8K memories. Here A13 is used to select four more 2K memory groups. Thus totally eight 2K memory groups are divided by IC 74LS138, whose eight outputs Y0 to Y7 are used, one for each 2K memory chip. The chip can be either a 2716 EPROM or a 6116 (2K) RAM. Provision for three 2716 EPROM's is made on the PCB.

2716 I First 2K (Y0)

2716 II Second 2K (Y1)

2716 III Third 2K (Y2)

The rest of the outputs, Y3-Y7 brought out to the edge connector.

So, the address decoding is done with the 3-line to 8-line decoder 74LS138. The three lines A11, A12 and A13 are seperated into eight select - outputs, each connectable to one memory chip. There are three chip - enable input pins also in this 71LS138 IC. These three are connected to A15, A14 and IO/$\overline{M}$ signals from the 8085. The 74LS138 is enabled only for memory access signal (not for I/O) and for A15 low, A14 low.

Note that C3 gets $\overline{A14}$, $\overline{CS2}$ gets A15 and hence the 74LS138 is enabled if both A14 and A15 are low.

A15	A14	A13	A12	A11	A10	A9	A8	Address page
0	0	0	0	0	X	X	X	Y0 00 to 07
0	0	0	0	1	X	X	X	Y1 08 to 0F
0	0	0	1	0	X	X	X	Y2 10 to 17
0	0	1	1	1	X	X	X	Y7 38 to 3F

Thus the Y0 pin selects EPROM 1 at address range 0000 to 07FFF; Y1 selects EPROM 2 at address ranging from 0800 to 0FFF; Y2 pin selects the 6116 RAM chip at address range 1000 to 17FF. Y3 to Y7 are brought out to the edge connector. The other pins of the 8085 which are brought out via the edge connector are:

TRAP, RST 5.5, RST 6.5, RST 7.5 (interrupt pins) and Reset out pin.

The four interrupt pins are to be normally kept low (ie., '0' level). When they are made high ('1' level), any program that is running in the microprocessor is interrupted. Then the program commences from the locations 000 044 (octal add) or 000 055 or 000 065 or 000 075

respectively. These four pins are brought via the edge connector and they must be connected to the four switches provided on the keyboard. When any of one of these four swithces is pressed, the program goes to the concerned location and begins executing the program stored from that location onwards. These locations 000 044 etc, there are written already in the 2716 only JUMP instructions which cause the program to jump to the following address:

023	044	023	054	023	064	023	074	(octal words)
13	24	13	2C	13	34	13	3C	(in hex)

We can write any needed program starting from these RAM locations, and after attending to this interrupted job, return to the main program by writing a return instruction at the end of the interrupt service routine. The interrupt pins are very useful. These are called "Hardware Interrupts".

The Reset output pin proves useful if, later any others 8085 family of ICs are added. AT present, it is not used. The "Int" and "Hold" pins of the 8085 are permanently kept low and these pins are not brought out.

The "Int pin" makes provision for the additional interrupts of a different mode. Since we already, have four interrupts, this Int pin is not brought out. Also, using this Int pin for providing interrupt programs is not quite easy.

The Hold pin, if kept high will keep the address and data buses of the 8085 in the high impedence or floating state. No data or address information can go out or get in. So, the bus lines are freed from the processor. These lines cam then be used for externally connected high or low signals so that the memory ICs connected to the bus lines can be accessed for storage or read out of data. This is called Direct Memory Accessing. We do not require this facility and so the Hold pin is keyt low by a jumper to ground and is not brought out of the board either.

The SID and SOD pins are used for inputting a 1 bit data into the 8085 and outputting a 1-bit data from the 8085. These lines are vey useful. Normally, a data word can get into or come out of a microprocessor only via an input or an output port. But where only a single - bit data is to be input or output, the SID and SOD lines are used, in conjunction with RIM and SIM instructions of the 8085 respectively.

CHAPTER IV

EPROM PROGRAMMING BOARD

4.1 INTRODUCTION :-

The 2716 (or Texas instruments TMS2516) is a third generation Erasable and Programmable Read Only Memory chip. It has 16,384 bits of programming sites on the metal oxide FET's integrated into it. It has a 24 pin IC, approximately 6 cm by 2.5 cm in size.On the centre of this IC, at the top, there is a small 0.7cm X 0.7cm quartz glass window. Through this window, ultraviolet can be shone onto the chip to erase completly all the bits.

The byte access time of the normal 2716 chip is 450 ns, and a faster 350 ns is also available. Since the chip can store 2K bytes of information, it is normally more. The 2716 is simple to program, it does not need high voltage pulses or multiple power supplies for programming. It is possible to program the chip easily on - borad the system, without the need for a special EPROM Programming machine.

The method of programming a 2716 is as follows:

1. Give the address information to the address pins A0 to A10.

2. Give the data bytes onto the data pins after making the chip select pin high.

3. Apply 25 V to pin No.21

4. Apply a single 50-millisecond pulse to the programming pin. It will be high going pulse for this duration.

5. Disconnect the 25 V supply from pin 21, and connect 5 V supply to it.

6. Remove the data input.

7. Make the chip select pin low now, and you get the stored data out.

4.2 PROGRAMMING, USING THE 8085 KIT

Data to be programmed and can be stored on the RAM of the kit and transferred to the EPROM sequentially. The steps 1 to 4 mentioned earlier will have to be repeated for all the bytes to be programmed. And because of the 50 millisec pulse required for each byte, a total time of a few minutes only is required. It is advisable to program the EPROM 2716 page by page (A page here means one block of 256 bytes). Note that the address lines of the 2716 go from A0 to A10. Of these the lines A8, A9 & A10 are high order address lines. These three can vary from 000 to 111, ie., totally there are eight pages starting from 000 page and ending with 111 page. The remaining address lines A0 to A7 give 256 locations for each such page. We may choose the 0th page (page 000) first, program all its 256 locations, then choose the first page (page 001) and program its 256 bytes and so on. This is convenient, even though all the 2K bytes could be programmed together. By programming page data (256 bytes only) into the RAM for programming the particular page.

It is also advisable to program from different locations of RAM running to various pages or to program few bytes alone in a EPROM. The above is achieved only through software developments.

In order to program, the data should be given to the data pins and the data should stay there on these pins during the period of programming. So an output port which can latch and keep the data is required. But for reading the EPROM after programming, the data pins of the IC act as output points, and they should be needed from the microprocessor via an input port. Thus the data pins should be both on an output port and also to an input port.

For situations like this, the device called Programmable Input Output port (also called Peripheral Interface Adapter - PIA) is a useful one. The Intel 8255 also available from other sources, is therefore used here for the EPROM programming board which we are describing here. This 8255 has actually three ports - A, B and C. We call them simply "Ports" because each one can be configured to be an

input or an output port at any particular time. Thus by making the ports A as input, we can read the EPROM, and by making it as the output port, we can program the EPROM without changing any connections.

4.3 CIRCUIT DESCRIPTION

In our EPROM Programmer we have two sockets for programming two EPROMS simultaneously. We use live drivers 74ls245 and 74ls244 for data lines and address lines so that the current capabality for programming two chips are achieved. 74ls245 is a bi-directional data bus driver and 74ls244 is a uni-directional address line driver. The directional select of 74ls245 pin 1 is shorted with PCI (Pin 15) chip select signal of 8255. During programming the EPROM PCI is high which makes 74ls245 to pass data from pin 2 to 18 and respectively for all pins. Thus data from 8255 is output to EPROM chip. When the EPROM make reading then PCI is low and data transfered from EPROM to 8255.

Port A is assigned for data lines

Port B is assigned for EPROM's low address lines

Port C is assigned for EPROM's page address

While programming Vpp is connected to a well stabilized 25v power supply with current capabality of 500 ma. The power supply should be very accurate as EPROM chip may get damaged due to over voltages even by 0.5 to 1volt. Reset pin of 8255 is connected to reset pin of 8085. Whenever 8085 is active low therefore it is passed through an inverter. The chip select pin is activated low only when A1 and A7 are high. When 8255 is I/O mapped then the address bit have to be greater than C0 as

C0 selects 8255 and its A port register

C1 selects B port register

C2 selects C port register

C3 selects control register

Thus OUT CO means data is outout through A port to Eprom.

4.4 SOFTWARE DESCRIPTION

Usually EPROMs are programmed page wise. In our kit it is possible to program the EPROM location wise. The start of location of programming EPROM can be any thing. The data is fetched from RAM location or from EPROMs in location No 0000 to 07FF or 0800 to 1FFF or 2000 to 27FF. Any number of bytes with in 2K can be programmed. In the programming routine initially some locations of RAM are utilised for storing initial address of EPROM which has to be programmed , starting and ending address of location from where data is fetched and also contains number of bytes to be programmed. Please refer key-notes for more information.

Initially data and address are placed in respective lines and status or low order portC line i placed with programming pulse and chip select signal for programming each location it takes 50 ms. Then a time delay for 50 ms is called . After 50 ms the programming pulse is reset by output on port C with previous control word.

While reading the EPROM remember that Vpp is connected to VCC and PCI is made low in software. Now datas are passed to 8095 from EPROM. This display shows the address is moved to HL pair. The keyboard subroutine is called which prints the contents of H,L & C registers. When you press 'C' key alone HL pair is incremented displaying next data when pressing 'D' key alone HL pair is decremented displaying previous data.

In verification program the contents of programmed EPROM is verified with RAM and EPROM from where the copy is made. If there is any wrong in data then the program displays 'ERROR' else it compares whole thing and displays 'CORRECT', if it is. For this also certain RAM locations are used for selecting address i/p's as in programming routine.

Blank check is the additional program which verifies whether the EPROM is programmed or not. Only EPROM's whose contents are FF can be programmed otherwise error in data programming occures. When such thing happens entire EPROM has

to be exposed to Ultera voilet light of 2500 A which is waste of time . To avoid such situations are EPROM's are subjected to blank check before they are programmed. If there is any mismatch the program displays 'ERROR'.

4.5 THE 8255A PROGRAMMABLE PERIPHERAL INTERFACE

The 8255A is a widely used, programmable, paralled I/O device. It can be programmed to transfer data under variour conditions, from simple I/O to interrupt I/O. It is flexible, versatile, and economical (when multiple I/O ports are required), but some what complex. It is an important general purpose I/O device that can be used with almost any microprocessor.

The 8255A has 24 I/O pins that can be grouped primarily in two 8-bit parallel ports A and B, with the remaining eight bits as port C. The eight bits of port C can be used as induvidual bits or be grouped in two 4 bit ports as C upper (Cu) and C lower (Cl). This device is like three 8212s with many more additional features. The functions of these ports are defined by writing a control word in the control register.

The functions of the 8255A, classified according to two modes is as shown below. The Bit Set / Reset (BSR) mode and the I/O mode.

The BSR mode is used to set or reset the bits in port C. The I/O mode is further divided into three modes. Mode 0, Mode 1 and Mode 2. In Mode 0, all the ports function as simple I/O ports. Model 1 is a handshake mode where by ports A and/or B use bits from port C as handshake signals. In the handshake mode, two type of I/O data transfer can be implemented. Status check and interrupt. In Mode 2, Port A can be setup for bidirectional data transfer using handshake signals from port C, and Port B can be setup either in Mode 0 or Mode 1. In our case we use 8255 in Mode 0 operation alone.

BLOCK DIAGRAM OF 8255A :-

The block diagram in Fig shows two 8 bit ports (A&B), two 4-bit ports (Cu and Cl), the data bus buffer and control logic. Figure shows a simplified but expanded version of the internal structure, including a control register. This block diagram includes all the elements of a programmable device. Port C performs functions similar to that of the status register in addition to providing handshake signals.

CONTROL LOGIC :-

The control section has 6 lines. Their functions and connections are as follows:

$\overline{RD}$ (READ) : This control signal enables the read operation. When the signal is low, the MPU reads data from a selected I/O port of the 8255A.

$\overline{WR}$ (WRITE) : This control signal enables the write operation. When the signal goes low, the MPU writes data to a selected I/O port or the control register.

RESET (Reset) : This is an active high signal, it clears the control register and sets all ports in the input mode.

$\overline{CS}$, A0 and A1 : These are device select signals. $\overline{CS}$ is connected to a decoded address and A0 and A1 are generally connected to MPU address lines A0 and A1 respectively.

The $\overline{CS}$ signal is the master chip select, and A0 and A1 specify one of the I/O ports or the control register as given below:

$\overline{CS}$	A0	A1	SELECTED
0	0	0	Port A
0	0	1	Port B
0	1	0	Port C
0	1	1	Control Register
1	X	X	8255A is not selected

As an example, the port addresses are determined by the $\overline{CS}$, A0 and A1 lines. The $\overline{CS}$ line goes low when A7=1 and A6 through A2 are at logic 0. When these signals are combined with A0 and A1, the port addresses range from 80H to 83H, as shown in Figure.

CONTROL WORD :-

Figure shows a register called control register. The contents of the register, called the control word, specify an I/O function for each port. This register can be accessed to write a control word when A0 and A1 are at logic 1, as mentioned previously. The register is not accessible for a read operation.

Bit D7 of the control register specifies either the I/O function on the Bit Set/Reset function as classified. If bit D7=1, bits D6-D0 determine I/O functions in various modes, as shown in figure. If bit D7=0, port C operates in Bit Set/Reset mode. The BSR control word does not affect the functions of port A, B and C.

To communicate with peripherals through the 8255A, three steps are necessary.

1. Determine the addresses of ports A, B and C and of the control register according to the chip select logic and address line A0 and A1.

2. Write a control word in the control register.

3. Write I/O instructions to communicate with peripherals through ports A, B and C.

MODE 0, SIMPLE INPUT OR OUTPUT :-

In this mode port A and B can be viewed as equivalent to two 8212's and port C as equivalent of two 4 bit 8212's. Each port (or half port in case of C) can be programmed to function as simply an input port or an output port. The input/output features in mode 0 are as follows.

1. Outputs are latched.

2. Inputs are not latched.

3. Ports do not have handshake or interrupt capability.

4.6 ERASING THE EPROM:-

The EPROM and PROM are read only user programmable memories that can be reprogrammed a number of times. If a mistake occurs in the data programmed the whole EPROM should be erased and reprogrammed. There are two types:

1. The UV-Erasable PROM and
2. Electrically erasable PROM.

A typical EPROM is erased by exposing it to hand (high frequency) ultra violet light for five to ten minutes. Thus returning the contents of all memory cells to zero by discharging them.

An EPROM package has a characteristic aspect. The seal on top of the chip is not opaque. It is a quartz window that allows the ultra-violet light through. Once zeroed, the EPROM can be programmed with a special PROM programmer selected locations within the EPROM can then be programmed and within a few minutes a bit pattern can be installed in the EPROM. Then the component can be inserted in the application board. If errors are detected or changes are desired, the EPROM can be plugged and reprogrammed within minutes. This process can be repeated many times.

Several technologies are used to implement EPROM's. The " floating - gate " accumulated is one of the best used. A charge is accumulated in the silicon gate "floating" above the silicon substrate but isolated from it by a silicon - dioxide layer. The charge in induced in the silicon gate by trains on pulses. Once programmed, an EPROM is expected to retain its charge for 10 years with only 30% 100s of charge. Erasure of the charge is a accomplished with hard ultra-voilet light. The photons hitting the floating silicon gate displace electrons from shallow energy levels and cause them to migrate to the silicon substrate where their charge is neutralised, the corresponding bit reverts to zero.

EPROM's are also available that are erasable by electricity.

CHAPTER V

ADC & DAC INTERFACE

5.1 INTRODUCTION :

Digital to Analog and Analog to Digital conversion form two very important aspects of digital data processing. Digital to analog conversion involves translation of digital information in to equivalent analog information. As an example, the O/P of a digital system might be changed to analog form for the purpose of pen recorder. Similiarly, an analog signal might be received for servo motors which drives the cursor arms of a plotter. In this respect, a D/A converter (DAC) is some times considered a decoding device.

The process of changing an analog signal to an equivalent digital signal is accomplished by the use of an A/D converter. For example, an A/D converter is used to change the analog output signals from a transducer into equivalent digital signals. These signals would be in a form suitable for entry into a digital system. An A/D converter is often referred to as an encoding device since it is used to encode signals for entry into a digital system.

5.2 METHODS OF ADC CONVERSION

There are many methods of A/D conversion they are

1. Simultaneous conversion or Flash converter,which is faster of all ADCs. It consists of many conparators with increasing order of reference voltages applied to other. The O/P of the comparators depends on how large the analog signal is from reference voltage. Example,:

Simultaneously A/D conversion table:

	comparator o/ps		
Input voltage	C1	C2	C3
+0 TO + V/4	L	L	L
+V/4 TO + V/2	H	L	L
+V/2 TO +3V/3	H	H	H
+3V/4 TO + V	H	H	H

Number of comparators required will be 2 n-1 n-no

2. COUNTER METHOD : By this method higher resolution could be obtained using minimum number of comparators as it is costlier in simultaneous conversions. This is possible by variable reference voltages. But the time required for this type of conversion is pretty large and depends on number of bits required.

3. SLOPE TYPE A/D CONVERTER : Here a ramp is generated and compared with analog signal by this time a clockof certain frequency is passed to the O/P logic which counts the clock pulses. Here the circuit is very complex.

4. SUCCESSIVE APPROXIMATION : This is the simplest and most accurate and speedy converter, but less speed than Flash converter.

The figure shows a Successive approximation type of A/D converter. The heart of the circuit is an 8 bit Successive approximation resistor (SAR), whose O/P is applied to an 8 bit D/A converter. The analog O/P Va of the D/A converter is then compared to an analog signal Vin by the comparator. The O/P of the comparator is a serial data input to the SAR. The SAR then adjust its digital output (8bits) until it is equivalent to analog input Vin. The 8 bit latch at the end of conversion holds on to the resultant digital data output. The circuit works as follows. At the start of a conversion cycle, the SAR is reset by holding the start(S) signal high on the first clock pulse LOW - TO - HIGH transition, the most siginificant output bit Q7, which is compared with the analog input Vin, if the comparator output is

low the digital Analog output isHIGH, the D/A output Vin and the SAR will keep the MSB Q7 set. In any case, on the next clock pulse LOW - TO - HIGH transition the SAR will set the next MSB Q6. Depending on the output of the comparator, the SAR will then either keep or reset the bit Q6. This process is continued until the SAR tries all the bits. As soon as the LSB Qo is tried, The SAR forces the conversion complete (CC) signal HIGH to indicate that the parallel output lines contain valid data. The CC signal in turn enables the latch and digital data appears at the output of the latch. Digital data are also available serially as the SAR determines each bit to cycle the converter or continuously the CC signal may be connected to start conversion input. The advantage of the successive approximation A/D converter is in its high speed and excellent resolution. for example, 8 bit successive approximation A/D converter (0804) requires only 8 clock pulse.

5.3 ADC 0804

ADC 0804 is a CMOS 8-bit successive approximation A/D convertor which use a modified potentiometers ladders, and are designed to operate with 8085 control bus via a 3-state outputs. These converters appear to the procesor as memory locationn or I/O ports and hence no interfacing logic is required.

The differential analog voltage input has good common mode rejection and permits offsetting the analog-Zero-Input voltage value. In addition, the voltage reference input can be adjusted to allow encoding any smaller analog voltage span to the full 8-bits of resolution.

5.4 FUNCTIONAL DESCRIPTION :

A functional diagram of the ADC 0804 of A/D converter is shown in fig. The device operates on the successive approximation principle. Analog switches are closed sequentially by successive approximation logic until the analog differential input voltage Vin(+) - Vin(-) matches a voltage derived from a tapped resistor string across the reference

voltage. The most significant bit is tested first and after 8 comparisions (64 clock cycles), an 8 bit binary code (1111 1111 = full-scale) is transferred to an Output latch.

The normal operation proceeds as follows. On the high-to-low transition of the $\overline{WR}$ input, the internal SAR latches and the shift-register stages are reset, and the $\overline{INTR}$ output will be set high. As long as the $\overline{CS}$ input and $\overline{WR}$ input remain low, the A/D will remain in a reset state. Conversion will start from 1 to 8 clock periods after at least one of these inputs makes a low-to-high transition. After the requiste number of clock pulses to complete the conversion, the $\overline{INTR}$ pin will make a high-to-low transition. This can be used to interrupt a processor, or otherwise signal the availability of a new conversion. A $\overline{RD}$ operation (with $\overline{CS}$ low) will clear the $\overline{INTR}$ line high again. The device may be operated in the free-running mode by connecting $\overline{INTR}$ to the $\overline{WR}$ input with $\overline{CS}$ = 0. To ensure start-up under all posible conditions, an external $\overline{WR}$ pulse is required during the first power-up cycle. A conversion-in-process can be interrupted by using a second start command.

DIGITAL DETAILS :

The converter is started by having $\overline{CS}$ and $\overline{WR}$ simultaneously low. This sets the start flip-flop (F/F) and the resulting "1" level resets the 8 bit shift register, resets the interrupt (INTR) F/F and inputs a "1" to the Q output of DFF1, which is at the input end of the 8 bit shift register. Internal clock signals then transfer this "1" to the Q output of DFF1. The AND gate, G1, combines this "1" output with a clock signal to provide a reset signal to the start F/F. If the set signal is no longer present (either $\overline{CS}$ or $\overline{WR}$ is a "1"), the start F/F is reset and the 8 bit shift register then can have the "1" clocked in, which starts the conversion process. if the set signal were to still be present, this reset pulse would have no effect (both outputs of the start F/F would be at a "1" level) and the 8 bit shift register would continued to be held in the reset mode. This allows for asynchronous or wide $\overline{CS}$ and $\overline{WR}$ signals.

After the "1" is clocked through the 8 bit shift register (which complets the SAR operation) it appears as the input to DFF2. As soon as this "1" is output from the shift register, the AND gate, G2, causes the new digital word to

transfer to the 3 state output latches. When DFF2 is subsequently clocked, the $\overline{Q}$ output makes a high-to-low transition which causes the INTR F/F to set. An inverting buffer then supplies the $\overline{INTR}$ output signal.

When data is to be read, the combination of both $\overline{CS}$ amd $\overline{WR}$ being low will cause the INTR F/F to be reset and the 3 state output latches will be enabled to proveide the 8 bit digital outputs.

DIGITAL CONTROL INPUTS :

The digital control inputs ($\overline{CS}$, $\overline{RD}$ and $\overline{WR}$) meet standard TTL logic voltage levels. These signals are essentially equivalent to the standard A/D start and output enable control signals, and are active low to allow an easy interface to microprocessor control busses. For non-microprocessor based applications, The $\overline{CS}$ input (pin 1) can be grounded and the standard A/D start function obtained by an active low pulse at the $\overline{WR}$ input (pin 3). The output enable function is acheived by an active low pulse at the $\overline{RD}$ input (pin 2).

ANALOG OPERATION :

The analog comparisions are performed by a capacitive charge summing circuit. Three capacitors (with precise ratioed values) shares a common node with the input to an auto-zero comparator. The input capacitor is switched between Vin(+) and Vin(-), while two ratioed reference capacitors are switched between taps on the reference voltage divider string. The net charge corresponds to the weighted difference between the input and the current total value set by the successive approximation register. A correction is made to offset the comparision by 1/2 LSB.

ANALOG DIFFERENTIAL VOLTAGE INPUTS AND COMMON-MODE REJECTION :

This A/D gains considerable applications flexibility from the analog differential voltage input. The Vin(-) input (pin 7) can be used to automatically subtract & fixed voltage value from the input reading (tare correction). This is also useful in 4mA - 20mA current loop conversion. In addition, common mode noise can be reduced by use of the differential input.

The time intervel between sampling Vin(+) and Vin(-) is 41/2 clock periods. The maximum error voltage due to this slight time difference between the input voltage samples is given by:

$$Ve(max) = (Vp)(2-fcm)(4.5/fCLK)$$

where:

Ve is the error voltage due to sampling delay

Vp is the peak value of the common mode voltage

fcm is the common mode frequency

For example, with a 60 Hz common-mode frequency. fcm, and a 640 kHz A/D clock, fCLK, keeping this error to 1/4 LSB (app 5mV) would allow a common-mode voltage, Vp, given by:

$$Vp = \frac{[\ Ve(MAX)\ (fCLK)]}{(2\ \ fcm)\ (4.5)}$$

or

$$Vp = \frac{(5x10\ 3)\ (640x10\ 3)}{(6.28)\ (60)\ (4.5)} == 1.9V$$

The allowed range of analog input voltage usually places more severe restrictions on input common-mode voltage levels than this.

An analog input volage with a reduced span and a relatively large zero offset can be easily handled by making use of the differential input.

ANALOG INPUT CURRENT :

The internal switching action causes displacement currents to flow at the analog inputs. The voltage on the on-chip capacitance to ground is switched through the analog differential input voltage, resulting in proportional currents entering the VIN(+) input and leaving the VIN(-) input. These current trancient occures at the leading edge of the internal clocks. they rapidly decay and do not inherently cause errors as the on-chip comparator is strobed at the end of the clock period.

INPUT BYPASS CAPACITORS :

Bypass capacitors at the input will average these charges and cause a DC current flow through the output resistances of the analog signal sources. This charge pumping action is worse for continuous conversion with the VIN(+) input

voltage at full-scale. For a 640kHz clock frequency with the VIN(+) input at 5V, this DC current is at a maximum of approximately 5microAmps. Therefore, bypass capacitors should not be used at the analog inputs or the VREF/2 pin for high resistance sources (1kohm). If input bypass capacitors are necessary for noise filtering and high source resistance is desirable to minimise capacitor size, the effects of the voltage drop accross this input resistance, due to the average value of the input current, can be compensated by a full-scale adjustment while the given source resistor and input bypass capacitor are both in place. This is possible because the average value of the input current is a precise linear function of the differential input voltage at a constant conversion rate.

INPUT SOURCE RESISTANCE :

Large values of source resistance where an input bypass capacitor is not used, will not cause errors since the input current settle on prior to the comparision time. If a low-pass filter is required in the system, use a low-value series resistor (1 Kohm) for a passive RC section or add an op

amp RC active low-pass filter. For low-source-resistance applications, (1 Kohm),and a 0.1 mfd bypass capacitor at the inputs will minimize EMI due to the series lead induction of a long wire. A 100 ohm series resistor can be used to isolate this capacitor (both the R and C are placed outside the feedback loop) from the output of an op amp, if used.

STRAY PICKUP :

The leads to the analog inputs (pin6 and 7) should be kept as short as possible to minimize stray signal pickup (EMI). Both EMI and undesired digital-clock coupling to these inputs can cause system errors. The source resistance for these input should, in general, be kept below 5Kohm. larger values of source resistance can cause undesired signal pickup. Input bypass capacitors, placed from the analog input ground, will eliminate this pickup but can create analog scale errors as these capacitor average the trancient input switching currents of the A/D . This scale error depends on both a large source resistance and use of an input bypass capacitor. This error can be compensated by a full-scale adjustment of the A/D with the source resistance and input bypass capacitor in place , and the desired conversion rate.

REFERENCE VOLTAGE SPAN ADJUST :

For maximum application flexibility, these A/Ds have brrn designed to accomodate a 5V, 2.5V or an adjust voltage reference. This has been acheived in the design of the IC as shown in fig.

Notice that the reference voltage for the IC is either 1/2 of the voltage which is applied to the V+ supply pin, or is equal to the voltage which is externally forced at the VREF/2 pin. This allows for a pseudo-ratiomatric voltage reference using, for the V+ supply, a 5V reference voltage. Alternatively, a voltage less than 2.5V can be applied to the VREF/2 input. The internal gain to the VREF/2 input is 2 to allow this factor of 2 reduction in the reference voltage.

Such an adjust reference voltage can accommodate a reduced span or dynamic voltage range of the analog input voltage. If the analog input voltage were to range from 0.5V to 3.5V, instead of 0V to 5V, the span would be 3V. With 0.5V applied o the VIN(-) pin to observe the offset, the reference voltage can be made equal to 1/2 of the 3V span or 1.5V. The A/D will now will encode the VIN(+) signal from 0.5V to 3.5V

with the 0.5V input corresponding to zero and the 3.5V with corresponding to full-scale. The full 8 bits of resolution are therefor applied over this reduced analog input voltage range. The requist connection are shown in fig. For expanded scale inputs, the circuits of fig. can be used.

REFERENCE ACCURACY REQUIREMENT :

The converter can be operated in a pseudo-ratiomatric mode or an absolute mode. In ratiomatric converter applications, the magnitude of the reference voltage is a factor in both the output of the source transducer and output of the A/D converter and therefore cancels out in the final digital output code. In absolute conversion applications, both the initial value and the temperature stability of the reference voltage are important accuracy factor in the operation of the A/D converter. For VREF/2 voltages of 2.5V nominal value, initial errors of +- 10 mV will cause conversion errors of +- 1 LSB due to the gain of 2 of the VREF/2 input. In reduced span applications, the initial value of the stability of thr VREF/2 input voltage become even more important. For example, if the span is reduced to 2.5V, the analog input LSB

voltage is correspondingly reduced from 20mV (5V span) to 10mV and 1 LSB at the VREF/2 input becomes 5mV. As can be seen, this reduces the allowed initials tolerance of the reference voltage and requires correspondingly less absolute change with temperature variations. Note that spans smaller than 2.5V place even tighter requirements on the initial accuracy and stability of the reference source.

In general, the reference voltage will require an initial adjustment. Errors due to an improper value of reference voltage appears as full-scale errors in the A/D transfer function. IC voltage regulators may be used for reference if the ambient temperature changes are not excessive.

ZERO ERROR :

The zero of the A/D does not require adjustment. If the minimum analog input voltage value, VIN(MIN), is not grounded, a zero offset can be done. the converter can be made to output OOOO OOOO digital code for this minimum input voltage by biasing the A/D VIN(-) input at this VIN(MIN) value. This utilizes the differntial mode operation of the A/D.

The zero error of the A/D converter relates to the location of the first riser of the transfer function and can be measured by grounding the VIN(-) input and applying a small magnitude poditive voltage to the VIN(+) input. Zero error is the difference between the actual DC input voltage which is necessary to just cause an output digital code transition from 0000 0000 to 0000 0001 and the ideal 1/2 LSB value (1/2 LSB = 9.8mV VREF/2 = 2.5V).

FULL - SCALE ADJUST:

The full-scale adjustment can be made by applying a differential input voltage which is 1 1/2 LSB down fromthe desired analog full-scale voltage range and then adjusting the magnitude of the VREF/2 input (pin 9) for a digital output code which is just changing from 1111 1110 to 1111 1111. When offsetting the zero and used a span-adjusted VREF/2 voltage, the full-scale adjustment is made by inputting VMIN to the VIN(+) input which is given by:

$$vin(+)FSADJ == vmax - 1.5\ [(vmax - vmin)/256]$$

Where :

VMAX = the high end of the analog input range

and

VMIN = the low end (the offset zero) of the analog range. (both are ground referenced)

CLOCKING OPTION :

The clock for the A/D can be derived from an external source such as the CPU clock or an RC network can be added to provide self-clocking. The CLK IN (pin 4) makes use of a schmitt trigger.

Heavy capacitive or DC loading of the CLocK R pin should be avoided as this will disturb normal converter operation. Loads less than 50pF, such as driving up to 7 A/D converter clock inputs from a single CLK R pin of 1 converter, are allowed. For larger clock line loading, a CMOS or low power TTL buffer or PNP input logic should be used to minimize the loading on the CLK R pin (do not use a standard TTL buffer).

RESTART DURING A CONVERSION :

If the A/D is started ($\overline{CS}$ and $\overline{WR}$ go low and return high) during a conversion, the converter is reset and a new conversion is started. The output data latch is not updated if the conversion in progress is not completed. The data from the previous conversion remains in this latch.

CONTINUOUS CONVERSIONS :

In this application, the $\overline{CS}$ input is grounded and the $\overline{WR}$ input is tied to the $\overline{INTR}$ output. This $\overline{WR}$ and $\overline{INTR}$ node should be momentairly forced to logic low following a power-up cycle to insure circuit operation.

DRIVING THE DATA BUS :

This CMOS A/D, like MOS microprocessor and memories, will require a bus driver when the total capacitance of the data bus gets larger. Other circuitary, which is tied to the data bus, will add to the total capacitive loading, even in 3 state (high-impedance mode). Backplane bussing also greatly adds to the stray capacitance of the data bus.

There are some alternatives available to the designer to handle this problem. Basically, the capacitive loading of the data bus slows down the response time, even though DC specifications are still met. For systems operating with a relatively slow CPU clock frequency, more time is available in which to establish proper logic levels on the bus and therefore higher capacitive loads can be drivwn.

Finally, if time is short and capacitive loading is high, external bus driver must be used. These can be 3-state buffers (low power Schottky is recommended, such as the 74ls240 series) or special higher-drive-current products which are designed as bus drivers. High-current bipolar bus drivers with PNP inputs are recommended.

POWER SUPPLIES :

Noise spikes on the V+ supply line can cause conversion errors as the comparator will respond to this noise. A low-inductance tantalum capacitor should be used close to the converter V+ pin, and values of 1 mfd of greater are recommended. If an unregulated voltage is available in the system, a seperate 5V voltage regulator for the converter (and another analog circuitry) will greatly reduce digital noise on

the V+ supply. An ICL7663 can be used to regulate such a supply from an input as low as 5.2V.

5.5 CIRCUIT DESCRIPTION :-

ADC 0804 is multiplexed for 8 channels using 4051 and analog multiplexer. 4051 an analog multiplexer selects one of many analog input channels to the output common bus. The selection depends on digital data A4, A5. A6 entered at 9, 10, 11th pin of 4051 pin 3 is the output. VEE is used for giving negative inputs. The potential difference between VDD and VEE should not be greater than -12V to +5V. As we are not using negative voltages we can make VEE to ground and VDD as +5V. The chip select pin $\overline{E}$ is an active low signal. This is made always ground. The channel selection is brought from A4, A5, A6 pins of main board. The analog I/P signal is always a dc signal for our case. When the signals are repetative ac signal the negative region is clipped off and positive alone is give to ADC. In the software, initially the interrupt, are enabled after the conversion process is over $\overline{INTR}$ signal occurs, 8085 sents $\overline{CS}$ signal by activating any one of the port and $\overline{RD}$ signal which reads the ADC's O/P after conversion to

start the next conversion cycle $\overline{CS}$ and $\overline{WR}$ signal lines are simultaneously made low. The channels are selected from A4, A5, A6 which is moved to register and data out of ADC is moved to other register. The keyboard subtractive is called to display the data. The ADC 0804 can also be memory-mapped by connecting the chip-select signal to any address decoding combination. As we are using only 16K, we can take 64th K memory group to map the ADC chip. The high address lines (8-lines) are connected to a NAND gate when high address happens to be FF the ADC chip is selected otherwise not.

5.6 DAC CIRCUIT :-

In addition to ADC which can multiplex upto 8 channels we hence use ADC channel also for demonstration. We have included ADC channel with I/O mapped configuration. But using any no of port select signals it is possible to select many DAC channels using multiplexing or by using as many DAC-0800 chips. 7475 is used to latch and enable DAC selection, ie., it is only, which maps DAC to I/O mapped configuration. The I/O of DAC is smoothened and level shifted using the operational amplifier usually 741 or LF351 type.

CHAPTER VI

AUDIO CASSETTE INTERFACE

6.1 INTRODUCTION:-

This is one of our add-on utility. This has an effective use when working with the lengthier up programs. When working with lengthier programs, if we have to switch off at the end of the day and restart on the next day, we've to load the program again which is a tedious process. We can think of loading the program into EPROM. But when the program is still under development this notion goes down. Under these conditions the audio cassette interface which we've developed and incorporated to our kit proves to be ideal. This interface enables us to store the programs or data on to the tape and take back into to RAM on the next day. Any ordinary cassette and tape recorder can be used for this purpose. The program can be of any type. It can be some music compositions using the music program or it can be some programs to do some calculations, process an analogue signal with some industrial control programs. In all there cases, temporary storage with audio cassette recorder is very ideal.

6.2 UART - (Universal Asynchronous Receiver - Transmitter)

The UART is a single chip LSI device which implements the asynchronous parallel - serial transmission and series - parallel conversions. These chips have provisions to include a parity bit by themselves and also prefix and suffix the start and stop bits. A clock frequency should be given as input to the device from crystal TTL oscillator which is used for clocking out or clocking in the bits. Chips can work at one or more baud rates by using internal dividers for the clock frequency to divide it. Chips IM 6403, TMS 6012, A7-31015 etc are common economical UART chips.

The 8085 has an associated chip called 8251 which is compatible with an 8080, 8085 or any other 8-bit up. Baud rate choice to X1,X16,X64 is possible. This chip can be connected directly to the data bus. It incorporates all transmitter receiver parts of UART inclusive of a pair of I/P & O/P ports to interface directly to the data bus. It indicates whether a transmission process is going on, whether received word is fully assembled and ready to be read and similar information to the microprocessor. It creates error

flags if any errors are detected. It can insert one or more stop bits and it can choose an odd or even parity. It is controllable from the microprocessor by enterring a certain code word which it underastands and thus sets its internal flip-flops accordingly. The O/P pin is used to enter control information or data information with a logical 1 or logical 0 repectively. This is usually connected to the address A0 line. The 8251 can operate in a synchronous mode also.

The use of 8251 requires some software to set its mode of working bits, parity, baud rate etc and for receiving the data properly. Such chips are called communication interface chips. They represent good technology but are rather expensive and difficult to use.

6.3 SIMPLE CASSETTE INTERFACE :-

We can manage cassette interface which is a very much simpler way by using the microprocessor itself for the parallel in serial output register and serial in parallel out functions by employing software to do this. Provided the

transmission is at a sufficiently slow rate; 110 baud rate is really very slow. We can also check for error using the software. Thus parity bits can be avoided. But we have to transmit the words one after other in the same manner as UART including a start bit and two or more start bits. While receiving the bits, we must use the software to strip off the start and stop bits and pack them up as a data and store them in RAM even as the bits are coming out from the cassette recorder.

6.4 INTERFACE BOARD DETAILS :-

INTERFACE STANDARD

While the UART or 8251 can transmit the bits of a word as a logical 1 or a 0 there bits cannot be fed directly to a cassette recorder. The mike input of a cassette recorder can only receive an audio frequency signal. The logical 1's and 0's should therefore be converted into suitable audio tones. We can use two seperate frequencies 1's and 0's. The following standard is used.

1. Logical 1's is recorded as 10 KHz

2. Logical 0's is recorded as 0 hz

3. A word consists of one start bit, 8 data bits and two or more stop bits.

4. The interval between characters consist of an unspecified amount of time 10Khz.

5. Data is sent with the least significant D0 bit first; D7 bit as last.

6. Meaningful data must be recorded within 30 seconds of tape following the clear leader.

The above format which we use in our interfacing unit deviates slightly from the Kansans City standard from the fact that it uses 10KHz and 0 frequence for logic 0 where as in Kansans city standard we use instead of 0 frequence for logic 0. But it needs extra circuitry. So we've deviated from the KCS.

Coming to the interface board details it has two inputs from the kit. One is from the Q4 output coming from port 4 D0 bit on the I/O board's edge connector pin No.16. The other is SID pin of the 8085 IC coming from the main board edge connector pin No.24.

The ouput cables connect from the interface board to the sockets in the cassette recorder. One goes to the mike input socket and the other to the earphone socket.

6.5 CASSETTE RECORDER DETAILS :-

Any cassette recorder may be used for achieving this task. The tape speed however should be constant and the cassette used must be good, free from 'drop outs'.

When a cassette recorder is started usually about 10 seconds are required for the tape speed to become steady. So, during this initial period a 10KHz continous note should be recorded. This may be done for atleast 25 to 30 seconds to give a clear leader of 30 seconds. Thereafter, the program for outputting the RAM data can be executed thereby storing

the RAM contents on the tape.

6.6 INTERFACE CIRCUIT DETAILS :-

Audio cassette interface board incorporates two circuits which enables us to record the datas coming out of the D0 bit port and into audio cassette and again detect the recorded datas and send it to microprocessor. The circuits can be divided into two main parts.

1. One while converts the logic 1 and logic 0 datas from the port 4 D0 bit to the corresponding frequencies which are in the audio range, so that they can be recorded in the audio cassette.

2. Second circuit which detects the frequencies from the audio cassette and converts to the corresponding bit level which is sent to the microprocessor as the data.

6.7 CIRCUIT FOR RECORDING INTO CASSETTE :-

The main aim of this circuit is to give 10KHz

frequency for a 1 bit level for that predetermined duration of the 1 bit level and 0Hz for the logic 0 level. Here we use an analog multiplexer IC CD4051 which has the cmos logic. It has 16 pins. It is an 8 bit analog multiplexer which low enable. The oscillator which oscillates at a frequency of 10KHz is connected to any one of the channel which is selected using the pins 9,10 & 11. The O/P is available at the O/P pin 3 when the chip is enabled. The O/P of the port 4 D0 bit is inverted using IC 7404 and given to the pin 6 which is the chip enable pin. As a result whenever a logic 1 appears at the D0 bit of port 4 then it is inverted and a low signal is applied to the chip enable pin. So the chip is selected and the 10 KHz frequency is available at the O/P pin No.3. This O/P of the chip is given to the mike I/P of the cassette recorder which enables the recording of the 1 & 0 datas in the form of 10 KHz and 0Hz.

6.8 CIRCUIT FOR RECOVERY OF DATA FROM CASSETTE :-

This circuit is used to detect the datas which are available in the form of 10KHz and 0Hz signals and to convert them to logic 1 and logic 0 respectively so that they can be

identified as datas by the microprocessor. This is a simple circuit which employes an envelop detector. This involves 2 resistors, 2 diodes and a transistor preferrably a switching transistor with a schmitt NAND IC CD4093 which is used for wave shaping purpose. This O/P is then given to the SID pin of the microprocessor. Thereafter the software program takes care of the received data.

6.9 USING THE CASSETTE INTERFACE :-

The cassette software program that follows stores one page (256 bytes) of RAM. Any page may be stored but the page number must be first entered on 18E0. If more than a page is to be stored one has to repeat the process, entering the next page number in 18E0.

If the RAM contents represent a certain program then while receiving it back it should be loaded only on that page (because 8085 programs contain Jump type of instructions and so a program is not relocateable). If however, the RAM

contents are only data (such as a music composition), then can be stored into any other page.

So while using the software program, the following information is necessary.

18E0 should contain page no of RAM (10,11,12,13 etc)
18E1 should contain data/program information (omitting bit 6)
Data = 04 (ASCII word for D)
Program = 10 (ASCII word for P)
The cassette interface software program is written with the starting address 0500.

The software consists of two parts.

1. Program for storing onto tape.
2. Reading from tape program.

Before starting to store on tape a 30 sec leader of 10 KHz tone bust be recorded for which it is necessary to output a bit 1 on D0 of port 4 for this time. This can be easily achieved by a software program.

We've to execute the program by getting down to the correct starting address. Then after pressing both record

and play buttons on the cassette recorder we've to wait for 30 seconds and execute the store program (18E0 and 18E1 to be loaded with page no. and data/program information).

We can check if the data is transferred or not by noting the port 4 LED's which flikers while the data is moving. This flickering continues for about 25 second and then the display returns to monitor program. We can stop the recorder now and rewind it. A page of data has got stored on tape. For reading back to memory late;

1. 18E0 should contain page no of RAM to which to store if it were data (otherwise no need)
2. Press play button
3. Wait 25 seconds and then execute reading program ( )
4. The LED on interface board would flicker
5. If there be any error, port 4 LED's indicate EE/ The Stop try to reload.

6. If there is no error all the 256 bytes of data are written into the RAM on the particular page and display returns to monitor program. Page no. is flashed on to the port 4 LED's to indicate to the user which page has now been loaded from tape. Stop the recorder and proceed to use the data read.

6.10 DESCRIPTION OF THE PROGRAM :-

The software program can either be stored in the RAM or incase of permanent storage it can be kept in EPROM and used any time.

6.11 STORE PROGRAM :-

First it priks up from 05E0 the page no which has to store on to the tape. It sets H-register to the value. The L-register starts with zero. The code word (which says if it is a program or data) in 05E0, is priked up and is first output, generally called the punch subroutine. Then the high address (or page no.) is ouput as the second word, calling the punch subroutine once more. Thereafter, the contents of that page are output one after another. After one page of 256

bytes are punched out, the check code (formed by adding all the respective bits and generating odd parity sum word) is punched. There after the program returns to monitor via the RST-0 instruction.

The punch subroutine adds the data word into the C-register, thereby creating a sum of all words in that register. Thereafter it outputs a start bit, a zero, waits for 9ms via the time subroutine. Then it outputs the data word serially, using the shift instruction, giving a bit time delay between each. After 8 bits a 1 as output for stop.

CHAPTER VII

KEY - NOTES

In this chapter we deal with the memory organisatation and various special functions in the project.

We use total to 16K memory of which first 6K is occupied by EPROM and next 10K by RAM. EPROM is 2716 type with 24 pin dual-in-line package. 2716 is a CMOS type 2K EPROM. The monitor program, EPROM programming routine, EPROM reading, qutomatic verification and blank check, ADC routine are located in a single EPROM chip. Still more than half the EPROM are left blank and can be used for other interfacing routines.

The memory allocations are :-

EPROM I	--	0000	TO	07FF
EPROM II	--	0800	TO	0FFF
EPROM III	--	1000	TO	17FF
RAM I	--	1800	TO	1FFF
RAM II	--	2000	TO	27FF
RAM III	--	2800	TO	27FF
RAM IV	--	3000	TO	37FF
RAM V	--	3800	TO	3FFF

Still memory can be expanded to 64K. Presently we use 16K only. We can observe from the main board of the EPROM that a single 74ls138 address decoder is used for this purpose. In that pins 1,2,3 are the inputs where we give A11,A12,A13 address lines. For selecting first 8 chips i.e. upto 16K only A11,A12,A13 lines have to be enabled but A14 and A15 are always low. therefore A15 as input to active low chip select signal at pin 4. I/oM is given to chip select signal 2. For mrmory operation (M) is low and chip select 2 is activated. as we select one of first 16K. A15 is also low. The effective chip selection is CS = CS1 + CS2 of 74ls138. Therefore this 74ls138 is selected. The next chip select is active high signal CS3 where A14 is inverted and given to pin 6. A14 is also low for selecting first 16K. For selecting next 16K i.e. from 16K to 32K CS1 AND CS2 are given the same i/p but CS3 is given A14 as input instead of inverting it as in first 74ls138. Thus the second decoder is capable of selecting upto 32K.

For similar selection upto 48K i.e. from 32K to 48K then A15 is inverted and given to pin 4 and A14 is also inverted and given to pin 6 for selection of 48K to 64K pin 4 has A15 as input and pin 6 has A14 as input.

SUBROUTINE LOCATIONS

Some important Subroutines addresses are given as below.

1. When reset key is pressed the software routine starts from 0000.

2. Software interrupts are as follows :

   a. RST 1 JUMPS TO 0040

   b. RST 2 JUMPS TO 003B

   c. RST 3 JUMPS TO 3B08

   d. RST 4 JUMPS TO 3B28

   e. RST 5 JUMPS TO 3B48

   f. RST 6 JUMPS TO 3B88

   g. RST 7 JUMPS TO 3BC8

3. Hard ware interrupts are :

   a. RST 5.5 JUMPS TO 3B68

   b. RST 6.5 JUMPS TO 3BA8

   c. RST 7.5 JUMPS TO 3BE8

   d. TRAP JUMPS TO 0050

Break point program routines are from 0040 to 0048

Single step & register Examine routines are starting from 0050 to 008B

Single step store routine from 008C

Time delay 25ms routine from 00BF to 00D1

Keyboard subroutine from 0150

Special key examine routine from 0100

Look up table from 01F0

RAM storage area to store instruction code at break point.

3B00 B register

3B01 C register

3B02 D register

3B03 E register

3B04 H register

3B05 L register

3B06 A register

3B07 Flag register

Stack is from 3BFF TO 3FFF - 1K

When A key is pressed during register examination is

I	press	B	C	D
II	press	E	H	L
III	press	A	F	-

EPROM PROGRAMMING ROUTINE :

1. Programming routine Starting point 0220 TO 02EF
2. EPROM reading routine 02F0 TO 032E
3. Verification program 032F TO 03A6
4. Error display of wrong locations for verification program 03B0 TO 03C0
5. End display routine 03C6 TO 03DE
6. Error display routine 03E0 TO 03F7
7. Blank check routine 0400 TO 0435
8. Correct dispaly

Different status while execution and necessary directions for executions of program :

In EPROM Programming routine :

1. Write the following into different RAM locations before executing programming routine.

2. Location 3AF0 contains programming EPROM's lower address to where programming is to be done.

3. Location 3AF1 contains page number of EPROM to where programming is to done.

4. Location 3AF2 contains lower address from where the I byte of instruction is fetched for programming.

5. Location 3AF3 contains higher address from where the I byte of instruction is fetched for programming.

6. Location 3AF4 contains the lower address from where the last byte of instruction is fetched for programming.

7. Location 3AF5 contains the higher address from where the last byte of instruction is fetched for programming.

8. Location 3AF6 and 3AF7 contains number of bytes to be programmed.

9. Location 3AF8 contains the control word used for programming, verification and reading.

The above locations are common for both programming, verification and reading of EPROM. When the locations are wrongly given as input i.e. when 'to' address is given in the place of 'from' location then the carry byte is set which then shows ERROR message. When both I bytes location and last bytes location are same then the dispaly shows ERROR message.

The error message means the display shows

' ERROR '

If all programming is done correctly then the dispaly shows ' CORRECT '. In reading subroutine same RAM location contents as above are initially done. After doing it execute from location 02F0 for reading EPROM subroutine. The reading EPROM has to be placed in EPROM board socket.

After executing the program press 'C' key to continue to observe the next location and press 'D' key to observe the previous location, press '0' key to stop the routine which is indicated by displaying 'S' at the I display.

In verifing program same RAM location contents as above are initially loaded. If there is any logic error in loading, the display will show Error message. After verification with the original, if there is any error then alternate LEDs will blink until you press any key to continue the program verification.

We have seperate program for blank check. i.e. program checks whether the particular EPROM is already programmed or not. If the EPROM is not already programmed then it has in all locations FF as its contents. If at any location it varies from FF then that particular location may not be programmed. So in order to avoid such circumstances usually all EPROMs have to be checked for presence of FF in all locations.

The display shows Error message if any location content differs from FF or else it displays Correct message telling that the EPROM is verified to be correct for programming. ADCs and Audio Cassette Interface and other interfacing units are given seperately in a chapter.

CONCLUSION

This microprocessor based system is designed, constructed and its performances are found to be successful.

This system is versatile in the sense it is simple, economical but powerful. This is achieved by the effective use of the powerful instruction set of the chip Intel's 8085.

Though we've incorporated two Add-on utilities to this system, considerable number of them can still be added which requires on slight changes in the hardware and the suitable software.

This system can be utilised for number of industrial process control applications like temperature control, robot arm movement (pick and place applications) etc.,

S.No.	ADDRESS	MNEMONIC	OPCODE	COMMENTS
INTERRUPT AND STARTUP LOCATIONS				
	0000	MVI A	3E	Enable interrupts
	0001	F8	F8	
	0002	SIM	30	
	0003	EI	FB	
	0004	JMP	C3	Jump to keyboard scan routine
	0005	00	00	
	0006	01	01	
	0007	NULL	00	
RST-1	0008	JMP	C3	Jump to software interrupt RST-1
	0009	40	40	
	000A	00	00	
	000B	FF		
	000C	FF		
	000D	FF		
	000E	FF		
	000F	FF		
RST-2	0010	JMP	C3	Jump to software interrupt RST-2
	0011	00	00	
	0012	3B	3B	
	0013	FF	00	
	0014	FF	FF	
	0015	FF	FF	
	0016	FF	FF	
	0017	FF	FF	

RST-3	0018	JMP	C3	Jump to RST-3 routine
	0019	08	08	
	001A	3B	3B	
	001B	FF	OO	
	001C	FF	FF	
	001D	FF	FF	
	001E	FF	FF	
	001F	FF	FF	
RST-4	0020	JMP	C3	Jump to RST-4 Routine
	0021	28	28	
	0022	3B	3B	
	0023	FF	FF	
TRAP	0024	JMP	C3	Jump to hardware interrupt trap
	0025	50	50	
	0026	00	00	
	0027	FF	FF	
RST-5	0028	JMP	C3	Jump to software interrupt RST-5
	0029	48	48	
	002A	3B	3B	
	002B	FF	FF	
RST-5.5	002C	JMP	C3	Jump to hardware interrupt 5.5
	002D	68	68	
	002E	3B	3B	
	002F	FF	FF	
RST-6	0030	JMP	C3	Jump to software interrupt RST-6

	0031	88	88	
	0032	3B	3B	
	0033	FF	FF	
RST-6-5	0034	JMP	C3	Jump to hardware interrupt 6-5
	0035	A8	A8	
	0036	3B	3B	
	0037	FF	FF	
RST-7	0038	JMP	C3	Jump to hardware interrupt routine RST 7.5
	OO3D	E8	E8	
	003E	3B	3B	
	003F	FF	FF	
	0040	XTHL	E3	Exchange HL with top of stock
	0041	DCX H	2B	
	0042	LDA	3A	Load the data from previously stored area
	0043	00		
	0044	38		
	0045	MOV M,A	77	
	0046	XTHL	E3	
	0047	OUT	D3	Output to single step by interrupting trap
	0048	03	03	
	0049	OUT	D3	Twice
	004A	03	03	
	004B	RET	C9	
	004C	FF		

	004D	FF		
	004E	FF		
	004F	FF		

SINGLE STEP & REGISTER EXAMINE.

	0050	CALL	CD	This stores register contents into RAM locations
	0051	8C	8C	
	0052	00	00	
	0053	XTHL	E3	Exchanges stock with HL
	0054	PUSH PSW	F5	
	0055	MOV C,M	4E	Take the contents of that instruction so that the next
	0056	LXI D	11	Instruction can be displayed
	0057	00	00	De pair points to store area
	0058	3B	3B	
P	0059	CALL	CD	Call keyboard routine
	005A	50	50	
	005B	01	01	
AA	005C	CPI	FE	Compare if break key is pressed
	005D	BKEY	4C	
	005E	JNC	C2	If not check if register examine key is pressed
	005F	A1	6C	
	0060	00	00	
	0061	POP PSW	F1	If a key, restore registers and cause interrupt BYH trap
	0062	XTHL	E3	
	0063	CALL	CD	Go to register storing routine
	0064	REG	DF	

	0065	RESTORE	00	
	0066	OUT	D3	True output instructions to cause the interrupt via trap
	0067	03	03	
	0068	OUT	D3	
	0069	03	03	
	006A	RET	C9	
	006B	FF	FF	
A1:	006C	CPI	FE	Press a key for register observation
	006D	4A	4A	
	006E	JNZ	C2	If not check for go key after single step
	006F	R	D4	
	0070	00	00	
	0071	MOV A,E	7B	
	0072	CPI	FE	Check for last location 3B07
	0073	07	07	
	0074	JNZ	C2	If not zero goto A2
	0075	AZ	7A	
	0076	00	00	
	0077	LXI D	11	If last location replace E to OO
	0078	00	00	
	0079	3B	3B	
A2	007A	PUSH H	45	
	007B	LDAX D	1A	Get the value and move to H
	007C	MOV H,A	67	
	007D	INX D	13	
	007E	LDAX D	1A	

007F	MOV L,A	6F	Get the next value and move to L
0080	INX D	13	
0081	LDAX D	1A	
0082	MOV C,A	4F	Get the next store value and
0083	INX D	13	
0084	CALL	CD	As the keyboard display always displays H,L, and C contents press any key to display
0085	KEY BD	50	
0086	01	01	
0087	POP H	E1	
0088	JMP	C3	Go to check what key has been pressed
0089	PT.A4	5C	
008A	00	00	
008B	NOP	00	

STORE SUBROUTINE

008C	PUSH H	E5	
008D	PUSH PSW	F5	
008E	LXI H	21	
008F	00	00	
0090	00	00	
0091	DAD SP	39	AL contains stack address
0092	MOV A,M	7E	
0093	STA	32	
0094	FLAG	07	Flag is stored in 3b07
0095	3B	3B	

0096	INX H	23	Next
0097	MOV A.M	7E	
0098	STA	32	Storing accumulator in 3B06
0099	ACC	06	
009A	3B	3B	
009B	POP PSW	F1	
009C	POP H	E1	
009D	PUSH PSW	F5	
009E	MOV A,L	7D	
009F	STA	32	Next store C register in 3B05
00A0	L-REG	05	
00A1	3B	3B	
00A2	MOV A,H	7C	H in 3BO4
00A3	STA	32	
00A4	H-REG	04	
00A5	3B	3B	
00A6	MOV A,E	7B	
00A7	STA	32	E in 3BO3
00A8	H-REG	03	
00A9	3B	3B	
00AA	MOV A,D	7A	
00AB	STA	32	D-register in 3B02
00AC	D-REG	02	
00AD	3B	3B	
00AE	MOV A,C	79	Move C to A
00AF	STA	32	and store in 3BO1
00BO	C-REG	01	

	00B1	3B	3B	
	00B2	MOV A,B	78	Move B to A store in 3BOO
	00B3	STA	32	
	00B4	B-REG	00	
	00B5	3b	3B	
	00B6	POP PSW	F1	
	00B7	RET	C9	

DELAY - I ROUTINE:

	00B8	PUSH B	C5
	00B9	MOI B	06
	00BA	80	80
AA:	00BB	DCR B	05
	00BC	JMP	C3
	00BD	PT A3	CD
	00BE	00	00

TIME DELAY

	00BF	PUSH PSW	F5
	00C0	PUSH D	D5
	00C1	LXI D	11
	00C2	30	30
	00C3	09	09
	00C4	DCX D	46
	00C5	MOV A,D	7A
	00C6	ORA E	B3
	00C7	JNZ	C2

	00C8	C4	C4
	00C9	00	00
	00CA	POP D	D1
	00CB	POP PSW	F1
	00CC	RET	C9
AB:	00CD	JNZ	C2
	00CE	PT AA	BB
	00CF	00	00
	00D0	POP B	C1
	00D1	RET	C9
	00D2	FF	FF
	00D3	FF	FF

GO KEYING AFTER SINGLE STEP:

R	00D4	CPI	FE
	00D5	GO KEY	43
	00D6	JNZ	C2
	00D7	PT P	59
	00D8	00	00
	00D9	POP PSW	F1
	00DA	XTHL	E3
	00DB	CALL	CD
	00DC	REG-RESTORE	DF
	00DD	NOP	00
	00DE	RET	C9

RESGISTER RESTORING

00DF	PUSH PSW	F5
00EO	LDA	3A
00E1	00	00
00E2	3B	3B
00E3	MOV B,A	47
00E4	LDA	3A
00E5	01	01
00E6	3B	3B
00E7	MOV C,A	4F
00E8	LDA	3A
00E9	02	02
00EA	3B	3B
00EB	MOV D,A	57
00EC	LDA	3A
00ED	03	03
00EE	3B	3B
00EF	MOV E,A	5F
FO	LDA	3A
F1	04	04
F2	3B	3B
F3	MOV H,A	67
F4	LDA	3A
00F5	05	05
00F6	3B	3B
00F7	MOV L,A	6F
00F8	POP PSW	F1

	00F9	RET	C9	
	00FA	FF	FF	
	00FB	FF	FF	
	00FC	FF	FF	
	00FD	FF	FF	
	00FE	FF	FF	
	00FF	FF	FF	

HIGH ADDRESS 01

	0100	LX ISP	31	Initialise stack pointer to end of stack 3BFF
	0101	FF	FF	
	0102	3B	3B	
	0103	LXI H	21	Initialise HL pointer to start location of RAM
	0104	00	00	
	0105	18	18	
SCAN2	0106	MOV C,M	4E	To move the contents of pointed by HL to C register
SCAN1	0107	CALL	CD	Goto keyboard sub routine 0150
	0108	KEY BD	50	
	0109	01	01	
	010A	CPI	FE	Compare immediately to check whether CK key is pressed or not
	010B	40	40	
	010C	JNC	D2	If CK-key is pressed goto control key check routine
	01OD	PT-D	1C	
	010E	01	01	

	010F	MOV B,A	47	Else move the accumulator tob and store it
	0110	MOV A,C	79	Move the contents of C-register to accumulator pointed by HL pointer
	0111	RAL	17	Rotate 4 items left to remove the left most digit
	0112	RAL	17	Of the old data
	0113	RAL	17	
	0114	RAL	17	
	0115	ANI	E6	And immediate with FO to remove lower by the datas
	0116	FO	FO	
	0117	ORA B	BO	Or the a with B (Old content of a reg) Now a register have new LHS digist = Old R.H.S. digit
	0118	MOV C,A	4F	Move this to C' register for display
	0119	JMP	C3	
	011A	SCAN 1	07	
	011B	01	01	
D:	011C	CPI	FE	Compare whether the A register was 42' to check, for low key
	011D	LOWKEY	42	
	011E	JNZ	C2	If not jump to next check
	011F	PT.E.	25	
	0120	01	01	
	0121	MOV L,C	69	If yes alter the contents of L register with recently typed low address value
	0122	JMP	C3	
	0123	SCAN2	06	

	0124	01	01	
E:	0125	CPI	FE	To compare A register for high address key
	0126	HIGH KEY	41	
	0127	JNZ	C2	If not jump to next check
	0128	PT.F	2E	
	0129	01	01	
	012A	MOV H,C	61	If yes alter the contents of high address and display it
	012B	JMP	C3	
	012C	SCAN2	06	
	012D	01	01	
F:	012E	CPI	FE	To compare for decrementing address pointer key
	012F	D KEY	44	
	0130	JNZ	C2	If not UMP to next check
	0131	PT.P	37	
	0132	01	01	
	0133	DC XH	2B	If yes decrement C pointer and display its contents
	0134	JMP SCAN2	C3	
	0135	06	06	
	0136	01	01	
P:	0137	CPI	FE	To compare for next on incrementing key
	0138	SKEY	47	
	0139	JNZ	C2	If not jump to next check
	013A	PT.G	41	
	013B	01	01	

	013C	MOV M,C	71	If yes store the contents of C register into location pointed by HL pair
	013D	INX H	23	Increment H-L pair to next address
	013E	JMP	C3	Display the cremented address on the display unit
	013F	SCAN2	06	
	0140	01	01	
G	0141	CPI	FE	To compare whether go key is pressed or not
	0142	GOKEY	43	
	0143	JNZ	C2	If not jump to next check of break point key check
	0144	PT.Q	01	
	0145	02	02	
	0146	MVI A	BE	If go key is sensed then F B is sent to port 2 and OO is sent to port 1. This with E' in all display digits the programme is run from location pointed by HL pair
	0147	F6	F6	
	0148	OUT	D3	
	0149	02	02	
	014A	MVI A	BE	
	014B	F6	F6	
	014C	OUT	D3	
	014D	01	01	
	014E	PCHL	E9	To exeate users programme
	014F	NOP	OO	

KEY BOARD ROUTINE:

	0150	1N	DB	The key board is real through port 2
	0151	02	02	
	0152	ORA A	B7	Ored with itself to oset the flag
	0153	JM	FA	If any key is pressed M&B is set and M'flag is also set. Untill the key is realised it is looped back to read input port data
	0154	KBD	50	
	0155	SUBR	01	
	0156	CALL	CD	
	0157	TIME	BF	
	0158	DELAY	00	
LOOP1	0159	CALL	CD	Call the sub routine to display the incoming data
	015A	SOFT	70	
	015B	DISPLAY	01	
	015C	IN	DB	Read the keyboard and or the A register with itself. If AM key is pressed sigh bit will be set after the OR operation. If no key is pressed allow the keyboard debounce delay time and the N start reading the actual data from the keyboard
	015D	02	02	
	015E	ORA A	B7	
	015F	JP	F2	
	0160	LOOP 1	59	
	0161	01	01	
	0162	CALL	CD	

0163	TIME	BF	
0164	DELAY	00	
0165	IN	DB	
0166	02	02	
0167	ORA A	B7	
0168	JP	F2	The second jump is to avoid spuri our data
0169	LOOP	59	
016A	01	01	
016B	XRI	EE	X or the A' register with CO then and it with 4F. Because of this D4, D5, D7, are removed (made sero) and D6 is inverted. Note as per the hardware when CK is not pressed D6 will be high and it is zero when pressed and zero when not pressed. Therefore this software inversion is required
016C	C0	C0	
016D	ANI	E6	
016E	4F	4F	
016F	RET	C9	

SOFT WARE DISPLAY ROUTINE

0170	PUSH B	C5	Move B register and flag register into stock
0171	PUSH PSW	F5	
0172	MOV A,H	7C	Move the H' register contents to accumulator
0173	ANI	E	And immediate with FO to mark
0174	FO	FO	Lower nibble data
0175	RRC	OF	Right shift and move highly nibble to lower side and now higher nibbles are O'S
0176	RRC	OF	
0177	RRC	OF	
0178	RRC	OF	
0179	MOV E,A	5F	Same it in 'E' register
017A	MVI A	3E	Select digit 1 (left most) for displaying

017B	7F	7F	
017C	OUT	D3	
017D	01	01	
017E	CALL	CD	Call digit display routine to display the left most and the successive digits by repetitive call.
017F	SEG	C3	
0180	DISPLAY	01	
0181	MOV A,H	7C	After displaying I digit display II digit from the II nibble data or H register
0182	ANI	E6	Now mask higher nibble data.
0183	OF	OF	
0184	MOV E,A	5F	Store it in E register
0185	MVI A	3E	Enable II digit display
0186	BF	BF	
0187	OUT	B3	
0188	01	01	
0189	CALL	CD	Call again segment display routine
018A	SEG	C3	
018B	DISPLAY	01	
018C	MOV A,L	7D	Similarly display C register which contains lowr byte of the address displaying respectively in 4th and 5th digit without displaying in 3 digit
018D	ANI	E6	
018E	FO	FO	
018F	RRC	OF	
0190	RRC	OF	
0191	RRC	OF	

0192	RRC	OF	
0193	MOV E,A	5F	
0194	MVI A	1E	
0195	EF	EF	
0196	OUT	D3	
0197	01	01	
0198	CALL	CD	
0199	SEG	C3	
019	display	01	
019B	MOV A,L	7D	
019C	ANI	E6	
019D	OF	OF	
019E	MOV E,A	5F	
019F	MVI A	3E	
01A0	F7	F7	
01A1	OUT	D3	
01A2	01	01	
01A3	CALL	CD	
01A4	SEG	C3	
01A5	DISPLAY	01	
01A6	MOV A,C	79	Similar to address display data content also in 7th and 8th digit.
01A7	ANI	E6	
01A8	FO	FO	
01A9	RRC	OF	
01AA	RRC	OF	
01AB	RRC	OF	

01AC	RRC	OF	
01AD	MOV E,A	5F	
01AE	MVI A	3E	
01AF	FD	FD	
01B0	OUT	D3	
01B1	01	01	
01B2	CALL	CD	
01B3	SEG	C3	
01B4	DISPLAY	01	
01B5	MOV A,C	79	
01B6	AN1	E6	
01B7	OF	OF	
01B8	MOV E,A	5F	
01B9	MVI A	3E	
01BA	FE	FE	
01BB	OUT	D3	
01BC	01	01	
01BD	CALL	CD	
01BE	SEG	C3	
01BF	DISPLAY	01	
01C0	POP PSW	F1	Output previously pushed flag and B register contents
01C1	POP B	C1	
01C2	RET	C9	Return to scan routine

SEGMENT DISPLAY SUBROUTINE

01C3	MVI D	16	Move D register with high address byte of 100 K - up - table of segement display

	01C4	01	01	
	01C5	MVI A	3E	Mask accumulates lower nibble
	01C6	FO	FO	
	01C7	ORA E	P3	OR E with A to display 7 segment code, as in corresponding position specified by 'E' register in look-up table.
	01C8	MOV E,A	5F	
	01C9	LDAX D	1A	Store that in E register again and load data to accumulate from the look-up table.
	01CA	OUT	D3	Output though port 2 of segment display
	01CB	02	02	
	01CC	JMP	C3	
	01CD	PT H.	DB	
	01CE	01	01	
	01CF	RET	C9	

DELAY 1 SUBROUTINE

	01D0	PUST PSW	F5
	01D1	PUSH D	D5
	01D2	MVI E	1E
	01D3	0E	0E
LOOP2	01D4	DCR E	1D
	01D5	JNZ	C2
	01D6	LOOP	D4
	01D7	02	02
	01D8	POP D	D1
	01D9	PDP PSW	F1

01DA	RET	C9	
01DB	CALL	CD	
01DC	DELAY-1	D0	
01DD	01	01	
01DE	XRA	AF	Xoring 'A' with 'A' results in clearing 'A' register. When that is output on port '2' the digit is exitinguished (ie) each digit is displayed for sometime and then switched off.
01DF	OUT	D3	
01E0	02	02	
01E1	RET	C9	
01E2	FF		
01E3	FF		
01E4	FF		
01E5	FF		
01E6	FF		
01E7	FF		
01E8	FF		
01E9	FF		
01EA	FF		
01EB	FF		
01EC	FF		
01ED	FF		
01EE	FF		
01EF	FF		

LOOK UP TABLE FOR SEGMENTS

01F0	0	7E
01F1	1	OC
01F2	2	B6
01F3	3	9E
01F4	4	CC
01F5	5	DA
01F6	6	FA
01F7	7	0E
01F8	8	FE
01F9	9	CE
01FA	A	EE
01FB	B	F8
01FC	C	72
01FD	D	BC
01FE	E	F6
01FF	F	E2

SINGLE STEP AND BREAKPOINT

Q:	0201	CPI	FE	Compare the key pressed with single step key
	0202	CKEY	4C	
	0203	JNZ	C2	If not check for break point key
	0204	PT R.	OE	
	0205	02	02	
	0206	MVI A	3E	If yes set the Flip-Flop for TRAP interrupt
	0207	FF	FF	

	0208	OUT	D3	
	0209	03	03	
	020A	SUB A	97	
	020B	OUT	D3	
	020C	03	03	
	020D	PCHL	E9	To program step
R-	020E	CPI	FE	Is break point key pressed
	020F	B-KEY	48	
	0210	JNZ	C2	If not go to scan routine for scanning the next key pressed.
	0211	SCAN 2	06	
	0212	01	01	
	0213	MOV A,M	7E	If break-point key is pressed more the 'HL' pointed data to accumulator and store int in 3800
	0214	STA	32	
	0215	00	00	
	0216	38	38	
	0217	MVI A	BE	Replace that memory content with Restar - 1 software interrupt command so that the program is read and executed till CF is occuring and how it jumps to break point routine 0040
	0218	CF	CF	
	0219	MOV M,A	77	
	021A	JMPSCAN2	C3	
	021B	06	06	
	021C	01	01	
	021D	FF		

021E	FF		
021F	FF		
0220	CALL	CD	
0221	32	32	
0222	02	02	
0223	LDA	3A	
0224	F8	F8	
0225	3A	3A	
0226	ANI	36	
0227	F0	F0	
0228	STA	32	
0229	F8	F8	
022A	3A	3A	
022B	JMP	C3	
022C	C6	C6	
022D	03	03	
022E	HLT	76	
022F	FF		
0230	FF		
0231	FF		
0232	STC	37	Set carry flag
0233	CMC	3F	Reset it is zero
0234	Lda	3a	Load the accumulator the inital low byte starting address of copy program.
0235	F2	F2	
0236	3A	3A	
0237	MOV E,A	5F	Save it in E

0238	LDA	3A	Load the accumulator with final low byte address of copy program.
0239	F4	F4	
023A	3A	4A	
023B	SUB E	93	
023C	STA	32	Store the difference in 3AF6
023D	F6	F6	
023E	3A	3A	
023F	MVI B	06	Reset B register
0240	00		
0241	JNC	D2	If carry is set move 01 to b else it is 00
0242	PT.A.	46	
0243	02	02	
0244	MVI B	06	
0245	01	01	
0246	LDA	3A	Load accumulator with initial high order address of copy program
0247	F3	F3	
0248	3A	3A	
0249	MOV D,A	57	Store it in D
024A	LDA	3A	Load accumulator with final high order address of copy program
024B	F5	F5	
024C	3A	3A	
024D	SBB D	9A	Subtract with carry
024E	STA	32	Store the result in 3AF7
024F	F7	F7	

0250	3A	3A	
0251	JC	DA	if the final address is less than initial address then display error
0252	PT.E RR.	EO	
0253	03	03	
0254	JNZ	C2	Check if difference between initial and final address are same if so display error
0255	PT.B	5D	
0256	02	02	
0257	MOV A,B	78	
0258	ANI	E6	
0259	01	01	
025A	JZ	CA	
025B	PT-ERR	EO	
025C	03	03	
025D	LDA	3A	If is not containing any error move the lower byte of starting location of copy program to L
025E	F2	F2	
025F	3A	3A	
0260	MOV D,A	6F	
0261	LDA	3A	Load higher byte to H register
0262	F3	F3	
0263	3A	3A	
0264	MOV H,A	67	
0265	LDA	3A	Load EPROM's starting address to C
0266	FO	FO	

0267	3A	3A	
0268	MOV C,A	4F	
0269	MVI A	3E	Make A,B,C, ports of 8255 as output by entering 80 into control register
026A	80	80	
026B	OUT	D3	
026C	C3	C3	
026D	LDA	3A	Move into accumulator with higher byte as page No. of EPROM's starting location.
026E	F1	F1	
026F	MOV B,A	3A	
0270	MOV B,A	47	Store it in B
0271	RAL	17	Rotate accumulator 4 times left
0272	RAL	17	
0273	RAL	17	
0274	RAL	17	
0275	ORI	F6	Set the control word for enabling the EPROM for programming
0276	02	02	
0277	ANI	E6	
0278	F2	F2	
0279	STA	32	
0280	NOP	00	
0281	INX H	23	Increment HL pointer and also increment EPROM pointing address
0282	INX B	03	
0283	LDA	3A	

0284	F8	F8	Load accumulator with control word
0285	3A	3A	
0286	ANI	E6	
0287	FO	FO	
0288	STC	37	
0289	CMC	3F	
028A	RAR	IF	Reset the control word the original as before
028B	RAR	IF	
028C	RAR	IF	
028D	RAR	IF	
028E	CMP B	B8	Compare it with B if there is any change in higher byte change the control word accordingly for programming next page
028F	JNC	D2	
0290	PT.D	9E	
0291	02	02	
0292	MOV A,B	78	Average the control word according to new page and store it in 3A F8
0293	RAL	17	
0294	RAL	17	
0295	RAL	17	
0296	RAL	17	
0297	ORI	F6	
0298	02	02	
0299	ANI	E6	
029A	F2	F2	
029B	STA	32	

029C	F8	F8	
029D	3A	3A	
029E	LDA	3A	
029F	F7	F7	
02A0	3A	3A	
02A1	MOV D,A	57	Move 0 with no. of pages to programmed
02A2	LDA	3A	
02A3	F6	F6	
02A4	3A	3A	
02A5	MOV E,A	5F	
02A6	DCX D	1B	decrement D-E
02A7	MOV A,D	7A	
02A8	STA	32	Store it in 3AF7
02A9	F7	F7]	
02AA	3A	3A	
02AB	MOV A,E	7B	
02AC	STA	32	Store E in 3AF6
02AD	F6	F6	
02AE	3A	3A	
02AF	MVI A	3E	
02BO	FF	FF	
02B1	ANA E	A3	Check for zero bytes remaining for programming
02B2	JNZ	C2	
02B3	PT.C	7C	If no zero go to next byte programming
02B4	02	02	
02B5	MIV A	3E	

02B6	FF	FF	
02B7	ANAD	A2	
O2B8	JNZ	C2	If not zero goto next byte programming
02B9	PT.C	7C	
02BA	02	02	
02BB	RET	C9	
02BC	FF		
02BD	FF		
02BE	FF		
02BF	FF		
02C0	LDA	3A	
02C1	F8	F8	
02C2	3A	3A	
02C3	OUT	D3	Output the control word, address and data though C, B and A port
02C4	C2	C2	
02C5	MOV A,D	7A	
02C6	OUT	D3	
02C7	CO	CO	
02C8	MOV A,C	79	
02C9	OUT	D3	
02CA	C1	C1	
02CB	LDA	3A	
02CC	F8	F8	
02CD	3A	3A	
02CE	ORI	F6	Set program pulse for 50 ms
02CF	01	01	

02DO	OUT	D3	
02D1	C2	C2	
02D2	CALL	CD	
02D3	TIME	EO	
02D4	DELAY 50 MS	O2	
02D5	LDA	3A	
02D6	F8	F8	
02D7	3A	3A	
02D8	OUT	D3	Reset program pulse
02D9	C2	C2	
02DA	RET	C9	
02DB	FF		
02DC	FF		
02DD	FF		
02DE	FF		
02DF	FF		
02E0	PUSH D	D5	
02E1	LXI D	11	
02E2	FF	FF	
02E3	11	11	
02E4	DCX D	1B	
02E5	MOV A,D	7A	
02E6	ORA E	B3	
02E7	JNZ	C2	
02E8	PT.E	E4	
02E9	02	02	
02EA	MVI A	3E	

	02EB	01	01	
	02EC	OUT	D3	
	02ED	04	04	
	02EE	POP D	D1	
	02EF	RET	C9	
READING				
	02FO	LDA	3A	
	02F1	FO	FO	
	02F2	3A	3A	
	02F3	MOV L,A	6F	Set the starting low byte address of EPROM to be read into L.
	02F4	MVI A	3E	
	02F5	90	90	
	02F6	OUT	D3	Output control word into control register of 8255
	02F7	C3	C3	
	02F8	LDA	3A	
	02F9	F1	F1	
	02FA	3A	3A	
	02FB	MOV H,A	67	Move starting high byte address of EPROM to be read into H register
A	02FC	MOV A,L	7D	
	02FD	OUT	D3	Place the high and low address though C and B port of 8255
	02FE	C1	C1	
	02FF	MOV A,H	7c	
	0300	OUT	D3	
	0301	C2	C2	

	0302	IN	DB	Input the data thorugh A port.
	0303	CO	CO	
	0304	MOV C,A	4F	Move it to C register
	0305	CALL	CD	Go to keyboard display routine to display the data
	0306	KEY	50	
	0307	SUB-ROUTINE	01	
	0308	CPI	FE	To continue displaying next data press C and to decrement address press D
	0309	OC	OC	
	030A	JNC	D2	
	030B	PT	OE	
	030C	G	03	
	030D	INX H	23	
G	030E	CPI	FE	
	030F	OD	OD	
	0310	JNC	D2	
	0311	PT	14	
	0312	H	03	
	0313	DCX H	2B	
	0314	CPI	FE	
	0315	OO	OO	
	0316	JNZ	C2	
	0317	PT	26	
	0318	I	03	
	0319	MVI A	3E	
	031A	DA	DA	
	031B	OUT	D3	

031C	02	02
031D	MVIA	3E
031E	7F	7F
031F	OUT	D3
0320	01	01
0321	HLT	76
0322	FF	
0323	FF	
0324	FF	
0325	FF	
0326	JMP	
0327	PT	FC
0328	F	02
0329	FF	
032A	FF	
032B	FF	
032C	FF	
032D	FF	
032E	FF	

VERIFICATION PROGRAMME

032F	STC	37
0330	CMC	3F
0331	LDA	3A
0332	F2	F2
0333	3A	3A
0334	MOV L,A	6F
0335	MOV E,A	5F

0336	LDA	3A	Set the initial contents one register and RAM locatio same as programming routine
0337	F4	F4	
0338	3A	3A	
0339	SUB E	93	
033A	MOV E,A	5F	
033B	STA	32	
033C	F6	F6	
033D	3A	3A	
033E	MVI B	00	
033F	NOP	00	
0340	JNZ	C2	
0341	PT	45	
0342	J	03	
0343	MVI B	06	
0344	01	01	
0345	LDA	3A	
0346	F3	F3	
0347	3A	3A	
0348	MOV D,A	57	
0349	MOV H,A	67	
0350	LDA	3A	
0351	F5	F5	
0352	3A	3A	
0353	SBB D	9A	
0354	STA	32	
0355	F7	F7	

```
          0356      3A          3A
          0357      MOV D,A     57
          0358      3C          DA
          0359      PT          EO
          035A      ERROR       03
          035B      JNZ         C2
          035C      PT          64
          035D      K           03
          035E      MOV A,B     78
          035F      ANI         E6
          0360      01          01
          0361      JZ          CA
          0362      PT          EO
          0363      ERROR       03
K         0364      MVI A       3E
          0365      90          90
          0366      OUT         D3
          0367      C3          C3
          0368      LDA         3A
          0369      F1          F1
          036A      3A          3A
          036B      MOV B,A     47
          036C      RAL         17
          036D      RAL         17
          036E      RAL         17
          036F      RAL         17
          0370      ANI         E6
          0371      FO          FO
```

0372	STA	32	
0373	F8	F8	
0374	3A	3A	
0375	LDA	3A	
0376	FO	F2	
0377	3A	3A	
0378	MOV C,A	4f	
0379	MOV A,C	79	
037A	OUT	D3	
037B	C1	C1	
037C	LDA	3A	
037D	F8	F8	
037E	3A	3A	
037F	OUT	D3	
0380	C2	C2	
0381	IN	DB	
0382	CO	CO	
0383	CMP M	BE	Read the EPROM data and compare that with the memory data
0384	CNZ	C4	If not same display 'error' else check for next routine
0385	PT	BO	
0386	ERROR	03	
0387	MVI A	3E	
0388	FF	FF	
0389	OUT	D3	
038A	04	04	
038B	DCX D	1B	

	038C	MOV A,E	7B
	038D	ANI	E6
	038E	FF	FF
	038F	JNZ	C2
	0390	PT	98
	0391	M	03
	0392	MOV A,D	7A
	0393	ANI	E6
	0394	FF	FF
	0395	JZ	CA
	0396	PT	C6
	0397	END	03
M	0398	INX H	23
	0399	INX B	03
	039A	MOV A,B	78
	039B	RAL	17
	039C	RAL	17
	039D	RAL	17
	039E	RAL	17
	039F	ANI	E6
	03AO	FO	FO
	03A1	STA	32
	03A2	F8	F8
	03A3	3A	3A
	03A4	JMP	C3
	03A5	PT	79
	03A6	L	03

03A7	FF	
03A8	FF	
03A9	FF	
03AA	FF	
03AB	FF	
04AC	FF	
03AD	FF	
03AE	FF	
03AF	FF	

ERROR DISPLAY OF WRONG LOCATION

03BO	PUSH H	E5
03B1	PUSH B	C5
03B2	PUSH D	D5
03B3	MOV L,C	69
03B4	MOV C,A	4F
03B5	MOV H,B	60
03B6	MVI A	3E
03B7	AA	AA
03B8	OUT	D3
03B9	04	04
03BA	CALL	CD
03BB	KBD	50
03BC	DISPLAY	01
03BD	POP D	D1
03BE	POP B	C1
03BF	POP H	E1

03CO	RET		
03C1	FF		
03C2	FF		
03C3	FF		
03C4	FF		
03C5	FF		
03C6	MVI A	3E	To display 'END
03C7	F6	F6	
03C8	OUT	D3	
03C9	02	02	
03CA	MVI A	3E	
03CB	7FO	7F	
03CC	OUT	D3	
03CD	01	01	
03CC	OUT	D3	
03CD	01	01	
03CE	MVI A	3E	
03CF	46	46	
03DO	02	02	
03D1	02	02	
03D2	MVI A	3E	
03D3	BF	BF	
03D4	OUT	D3	
03D5	01	01	
03D6	MVI A	3E	
03D7	BC	BC	
03D8	OUT	D3	
03D9	02	02	

```
03DA    MVI A    3E
03DB    DF       DF
03DC    OUT      D3
03DD    01       01
03DE    HLT      76
03DF    FF
```

ERROR DISPLAY

```
03EO    MVI A    3E
03E1    F6       F6
03E2    OUT      D3
03E3    02       02
03E4    MVI A    3E
03E5    7F       7F
03E6    OUT      D3
03E7    01       01
03E8    MVI A,   3E
03E9    8C       8C
03EA    OUT      D3
03EB    02       02
03EC    MVI A    3E
03ED    BF       BF
03EE    OUT      D3
03EF    01       01
03FO    MVI A    3E
03F1    8C       8C
03F2    OUT      D3
03F3    02       02
```

03F4	MVI A	3E
03F5	DF	DF
03F6	OUT	D3
03F7	01	01
03F8	MVI A	3E
03F9	BC	BC
03FA	OUT	D3
03FB	02	02
03FC	MVI A	3E
03FD	EF	EF
03FE	OUT	D3
03FF	01	01
0400	MVI A	3E
0401	2C	8C
0402	OUT	D3
0403	02	02
0404	MVI,A	3E
0405	F7	F7
0406	OUT	D3
0407	01	01
0408	HLT	76
0409	FF	
040A	FF	
040B	FF	
040C	FF	
040D	FF	
040E	FF	

	040F	FF		
BLANK CHECK				
	0410	STC	37	To check the EPROM whether are locations contain FF then only the EPROM can be programmed
	0411	CMC	3F	
	0412	MVI L	2E	
	0413	00	OO	
	0414	MVI H	26	
	0415	00	00	
	0416	MVI,D	16	
	0417	07	07	
	0418	MVI E	1E	
	0419	FF	FF	
	041A	MVI A	3E	
	041B	90	90	
	041C	OUT	D3	
	041D	C3	C3	
M	041E	MOV A,H	7C	
	041F	RAL	17	
	0420	RAL	17	
	0421	RAL	17	
	0422	RAL	17	
	0423	ANI	E6	
	0424	FO	FO	
	0425	MOV B,A	47	
	0426	NOP	00	
	0427	OUT	D3	

0428	C2	C2
0429	MOV A,C	7D
042A	OUT	D3
042B	C1	C1
042C	IN	DB
042D	IN	C0
042E	CPI	FE
042F	FF	FF
0430	JNZ	C2
0431	PT	EO
0432	ERROR	03
0433	INX H	23
0434	NOP	00
0435	DCX D	1B
0436	MOV A,E	7B
0437	ANI	E6
0438	FF	FF
0439	NOP	OO
043A	JNZ	C2
043B	PT	1E
043C	M	04
043D	MOV A,D	7A
043E	ANI	E6
043F	FF	
0340	JNZ	C2
0341	PT	1E
0342	M	05
0343	JMP	C3

0344	PT	50	
0345	CORRECT	04	
0346	HLT	76	
0347	FF		
0348	FF		
0349	FF		
034A	FF		
034B	FF		
034C	FF		
034D	FF		
044E	FF		
044F	FF		

CORRECT DISPLAY ROUTINE

0450	MVI A	3E	
0451	7F	7F	To display correct
0452	OUT	D3	
0453	01	01	
0454	MVI A	3E	
0455	72	72	
0456	OUT	D3	
0457	02	02	
0458	MVI A	3E	
0459	BF	BF	
045A	OUT	D3	
045B	01	01	
045C	MVI A	3E	

045D	BC	BC
045E	OUT	D3
045F	02	02
0460	MVI A	3E
0461	DF	DF
0462	OUT	D3
0463	01	01
0464	MVI A	3E
0465	-	
0466	OUT	D3
0467	02	02
0468	MVI A	3E
0469	EF	EF
046A	OUT	D3
046B	01	01
046C	MVI A	3E
046D	8C	8C
046E	OUT	D3
046F	02	02
0470	MVI A	3E
0471	F7	F7
0472	OUT	D3
0473	01	01
0474	MVI A	3E
0475	F6	F6
0476	OUT	D3
0477	02	02
0478	MVI A	3E

0479	FB	FB	
047A	OUT	D3	
047B	01	01	
047C	MVI A	3E	
047D	72	72	
047E	OUT	D3	
047F	02	02	
0480	MVI A	3E	
0481	FD	FD	
0482	OUT	D3	
0483	01	01	
0484	MVI A	3E	
0485	8C	8C	
0486	OUT	D3	
0487	02	02	
0488	HLT	76	
0489	FF	FF	
048A	FF	FF	
048B	FF	FF	

CASSETTE SOFTWARE PROGRAM - STORE PROGRAM

0500	LDA	3A	
0501	EO	EO	Takes address from 18E0
0502	18	18	
0503	MOV H,A	67	and moves into H register
0504	SUB A	97	Clears Accumulator
0505	MOV L,A	6F	Clears L = 0
0506	MOV C,A	4F	C = 0

0507	LDA	3A	Picks up information whether data or program (04 or 10) from 18 E1
0508	EI	EI	
0509	18	18	
050A	CALL	CD	Calls Punching subroutine and outs this information to tape
050B	PNCH	2A	
050C	10	05	
050D	MOV A,H	7C	Takes the High Address information and outputs on to tape
050E	CALL	CD	
050F	PNCH	2A	
0510	10	05	
0511 A	MOV A,M	7E	Takes the data stored from memory pointed by H, L and outs to tape using Punch subroutine
0512	CALL	CD	
0513	PNCH	2A	
0514	10	10	
0515	INX H	23	Increments address
0516	LDA	3A	
0517	EO	EO	Takes the address page no from 18 E0
0518	18	18	
0519	INR A	3C	Increments it
051A	CMP H	BC	Compares to see if one full page has been output
051B	JNZ	C2	If not, goes to Point A to repeat
051C	A	11	

	051D	10	05	
	051EB	MOV A,C	79	If one full page has been ouptut to tape. it takes the parity sum code and complements it, increments by 1.
	051F	CMA	2F	
	0520	INR A	3C	
	0521	CALL	CD	Punches on to tape this value.
	0522	PNCH	2A	
	0523	10	05	
	0524	RSTO	C7	Returns to Monitor using RST-0 (C7)
	ER			Error Flag subroutine
	0525	MVI A	3E	
	0526	EE	EE	Makes Accumulator = EE
	0527	OUT	D3	
	0528	04	04	and flashes EE on Port 4 LEDs to indicate Error and Halts.
	0529	HLT	76	
PNCH	052A	PUSH PSW	F5	Punch subroutine ouptuts one word serially via D0 bit of Port 4.
	052B	ADD C	81	Adds the new word into C Register to form the sum of words
	052C	MOV C,A	4F	
	052D	POP PSW	F1	
	052E	PUSH B	C5	Saves Registers
	052F	MVI B	06	B = 12 for 1 start bit + 8 bits of word + 3 stop bits
	0530	0C	0C	B is a counter
	0531	ORA A	B7	Clears carry bit

	0532	RAL	17	Do bit made now zero
	0533	OUT	D3	Zero is output to mark a start bit
	0534	04	04	
	0535BI	CALL	CD	
	0536	TIME	5A	Wait for 9 ms time delay
	0537	10	05	
	0538	RAR	1F	Rotate bit into Do position
	0539	STC	37	Set carry bit
	053A	OUT	D3	Output again the Do bit
	053B	04	04	
	053C	DCR B	05	Decrement bit counter by 1
	053D	JNZ	C2	If bit counter has a value 0, ignore this instruction.
	053E	B1	35	If all bits have not yet been transmitted, go back to B1.
	053F	10	05	
	0540	POP B	C1	Restore Registers and
	0541	RET	C9	Return to called program
TIME	055A	MVI D	16	One bit time delay
	055B	4A	4A	
	055CH	MVI E	IE	Enters here from 10 D5 for 1/2 bit time delay
	055D	10	10	
	055EP	DCR E	1D	
	055F	JNZ	C2	
	0560	P	5E	
	0561	10	05	
	0562	DCR D	15	
	0563	JNZ	C2	

0564	H	5C	
0565	10	05	
0566	RET	C9	

READING THE STORED PROGRAM FROM THE TAPE

056A	SUBA	97	Clear Accumulator, C,E and L Registers
056B	MOV C,A	4F	
056C	MOV E,A	5F	
056D	MOV L,A	6F	
056E	CALL	CD	Call Reading subroutine which reads one word at a time
056F	READ	A0	
0570	10	05	
0571	CPI	FE	
0572	10	10	Compare the first word with 10, to find out if the tape data is a program
0573	JZ	CA	
0574	X	7F	Jump to X if so
0575	10	05	
0576	CPI	FE	If not, compare the first word with 04 which means data
0577	04	04	

LEARN TO USE MICROPROCESSOR

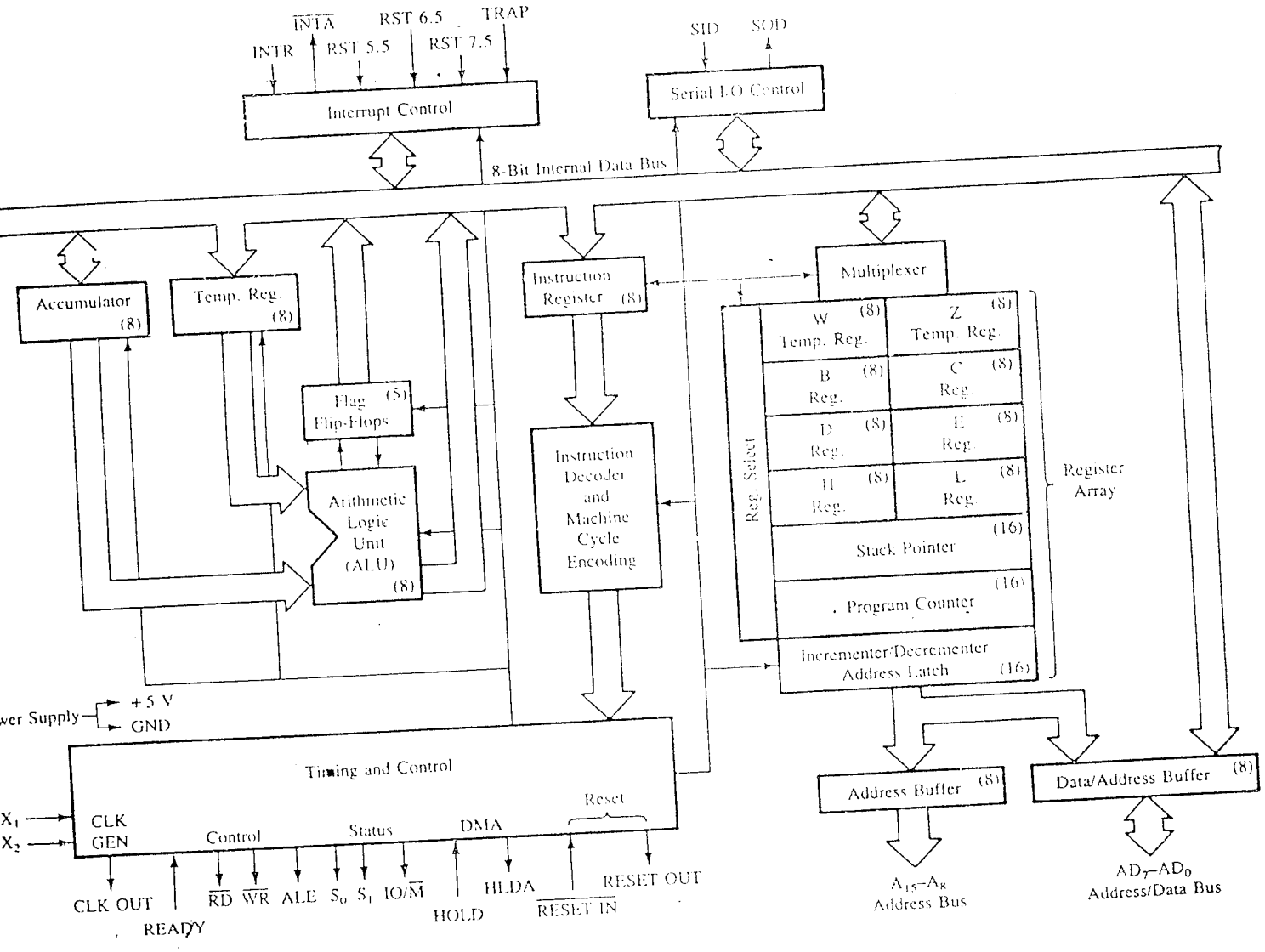
0578	JZ	CA	
0579	S	86	Jump to S if so
057A	10	05	If not,

057B	CALL	CD	Call error subr. to indicate error if the first word is neither 10 or 04
057C	ER	25	
057D	10	05	
057E	RSTO	C7	and return to monitor
057FX	CALL	CD	Now call the Read routine
0580	READ	AO	
0581	10	05	
0582	MOV H,A	67	Take the high address
0583	JMP	C3	
0584	A2	8D	
0585	10	05	
0586S	CALL	CD	If data, calll Read program
0587	READ	A0	
0588	10	05	
0589	LDA	3A	but load the high address with what is in 10 E0
058A	EO	EO	
058B	10	18	
058C	MOV H,A	67	
058DA2	CALL	CD	
058E	READ	A0	Read the 256 bytes of data one after another
058F	10	05	
0590	MOV M,A	77	
0591	INX H	23	Compare if 256 bytes have been loaded
0592	DCR E	1D	
0593	JNZ	C2	

	0594	A2	8D	Jump back to A2 if not over
	0595	10	05	
	0596	CALL	CD	If over, Read the last data word which is a check sum
	0597	READ	AO	
	0598	10	05	
	0599	JNZ	C2	This check sum must be zero. Check if it is so, if not zero,
	059A	ER	25	Then call Error display
	059B	10	05	
	059C	MOV A,H	7C	If checksum is O.K output High address page no. on Port 4 LEDs so that one knows into which page the tape data has got loaded.
	059D	OUT	D3	
	059E	04	04	
	059F	RET-O	C7	Then return to Monitor program
READ				
	05AO	CALL	CD	Read Subroutine
	05A1	SIN	A8	Calls Serial input program
	05A2	10	05	
	05A3	MOV B,A	47	
	05A4	ADD C	81	Forms sum of data words in C
	05A5	MOV C,A	4F	
	05A6	MOV A,B	78	
	05A7	RET	C9	Returns
	05A8SIN	PUSH D	D5	Serial input Program inputs a word from tape
	05A9	PUSH B	C5	Save B-C. D-E Reg. pairs
	05AA	LXI B	01	

05AB	00	00	C = 0
05AC	08	08	B=8 for bit counting
05ADINP	RIM	20	Read via SID the tape bit
05AE	RAL	17	Rotate bit left into carry
05AF	JC	DA	If no start bit found, go back to INP
05B0	INP	AD	
05B1	10	10	
05B2	CALL	CD	Wait half bit time
05B3	TIM E	D5	
05B4	10	5	
05B5	RIM	20	Read again
05B6	RAL	17	Rotate into carry
05B7	JC	DA	
04B8	INP	AD	If start bit not even now go back to 1nP.
05B9	10	05	
05BAS	CALL	CD	Wait 1 bit time
05BB	TIME	5A	
05BC	10	5	
05BD	RIM	20	Read
05BE	RLC	07	Rotate bit
05BF	ANI	E6	Pick up only the L.S.B.
05C0	01	01	
05C1	ADD C	81	Add to C Register which assembles the bits
05C2	RRC	0F	Rotate
05C3	MOV C,A	4F	and store in C Register
05C4	DCR B	05	Decrement bit counter

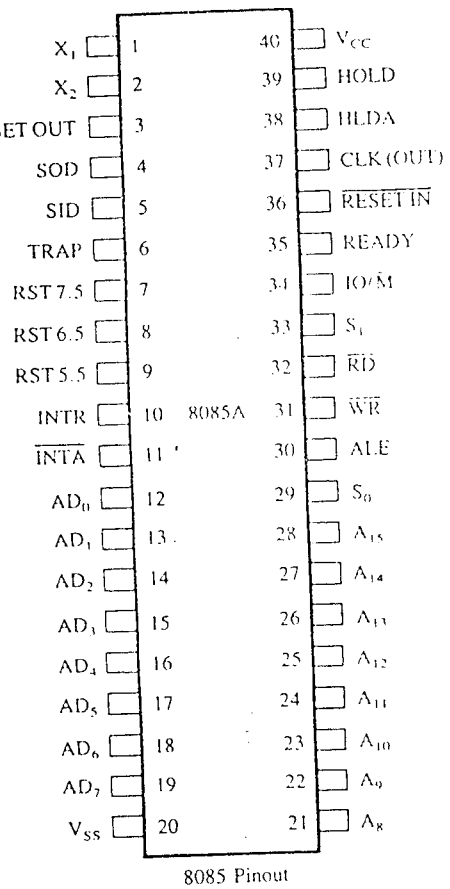
05C5	JNZ	C2	
05C6	S	BA	All bits over?
05C7	10	05	
05C8	CALL	CD	Wait for one bit time to allow for 1st stop bit
05C9	TIME	5A	
05CA	10	05	
05CB	CALL	CD	Again wait for one bit time-II stop bit
05CC	TIME	5A	
05CD	10	05	
05CE	CALL	CD	Again wait for one bit time-III stop bit
05CF	TIME	5A	
05DO	10	05	
05D1	MOV A,C	79	Move assembled word into Accumulator
05D2	POP B	C1	Restore Registers
05D3	POP F	D1	
05D4	RET	C9	Return to READ program
05D5TIMH	MVI D	16	TIMHALF is a subroutine to get 1/2 bit time delay
05D6	25	25	Move into D a value 25 which is half of 4A
05D7	JMP	C3	
05D8	PT.H	5C	Now jump to 10 5C which is Pt. H.
05D9	10	05	
05DA	RET	C9	Return after 1/2 bit delay.



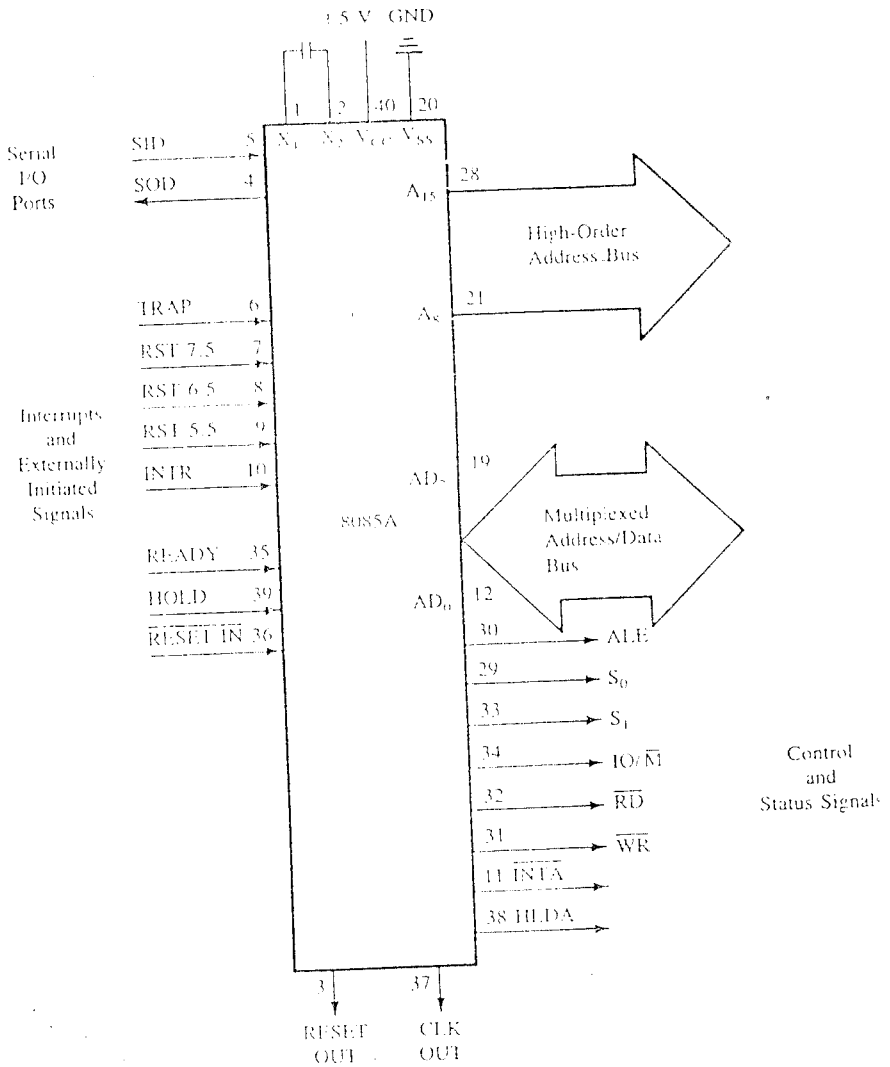
...he 8085A Microprocessor: Functional Block Diagram

NOTE: The 8085A microprocessor is commonly known as the 8085.

SOURCE: Intel Corporation, *MCS—80/85 Family User's Manual* (Santa Clara, Calif.: Author, 1979), p. 6-1.

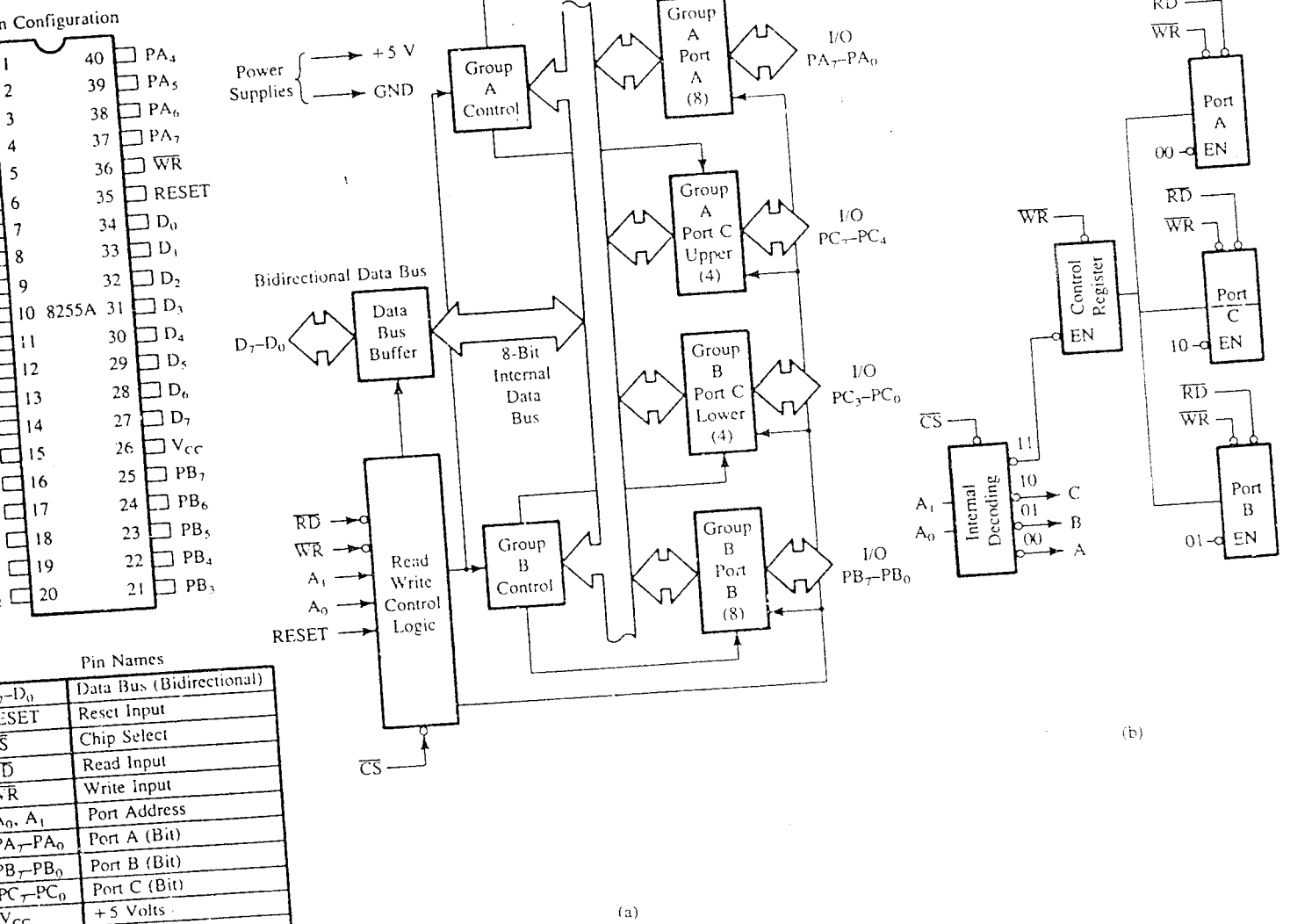


8085 Pinout



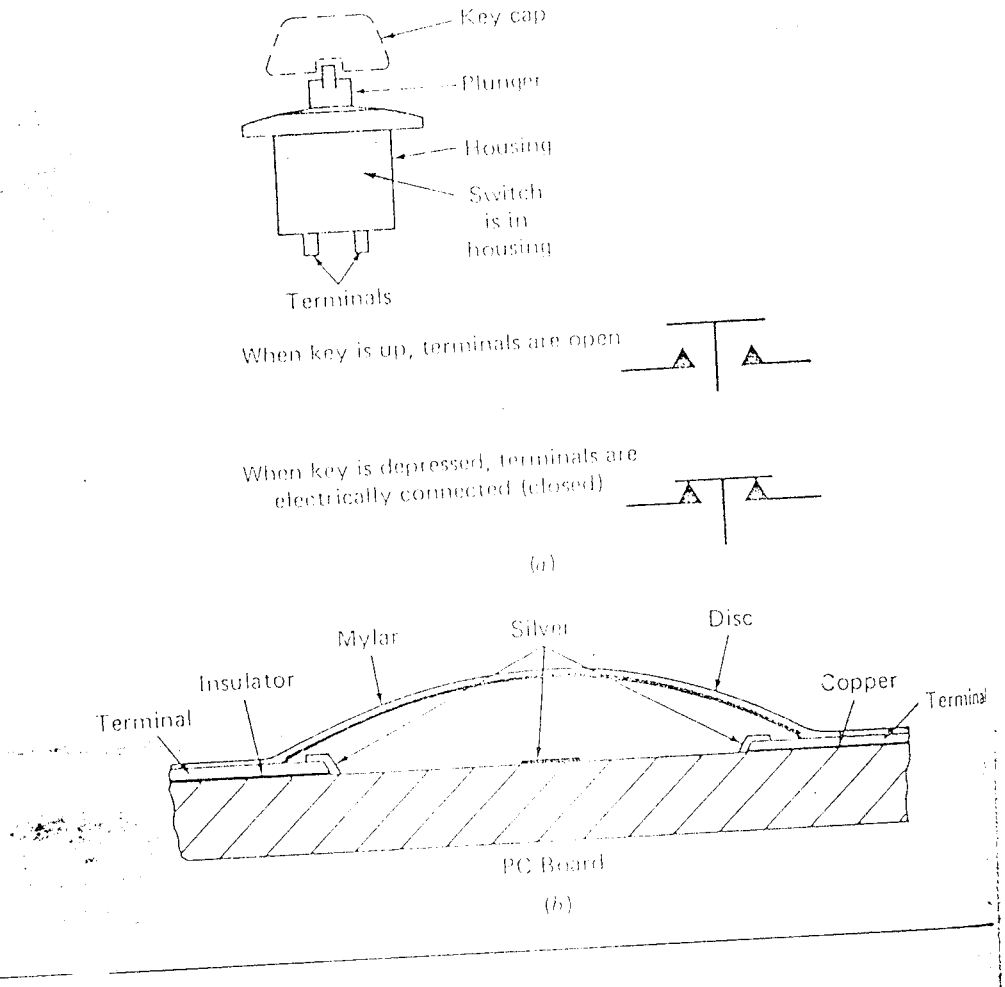
e 8085 Microprocessor Pinout and Signals

TE: The 8085A is commonly known as the 8085.
URCE (Pinout): Intel Corporation, *MCS — 80/85 Family User's Manual* (Santa Clara, Calif.: Author, 1979), p. 6-2.

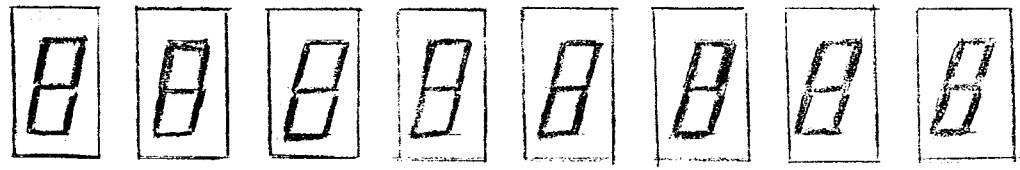


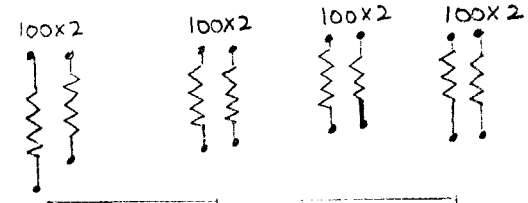
8255A Block Diagram (a) and an Expanded Version of the Control Logic and I/O Ports (b)

SOURCE: A: Intel Corporation, *MCS—80/85 Family User's Manual* (Santa Clara, Calif.: Author, 1979), p. 6-162.



(*a*) Keyboard switch.
(*b*) Keypad switch.





7475

7475

T1 T2 T3 T4 T5 T6 T7 T8

8xBC148

15kx2

15kx4

15k

7475

15k

7475

8xLED's

270x8

7430

7404

7402

7430

7430

7430

7430

74126

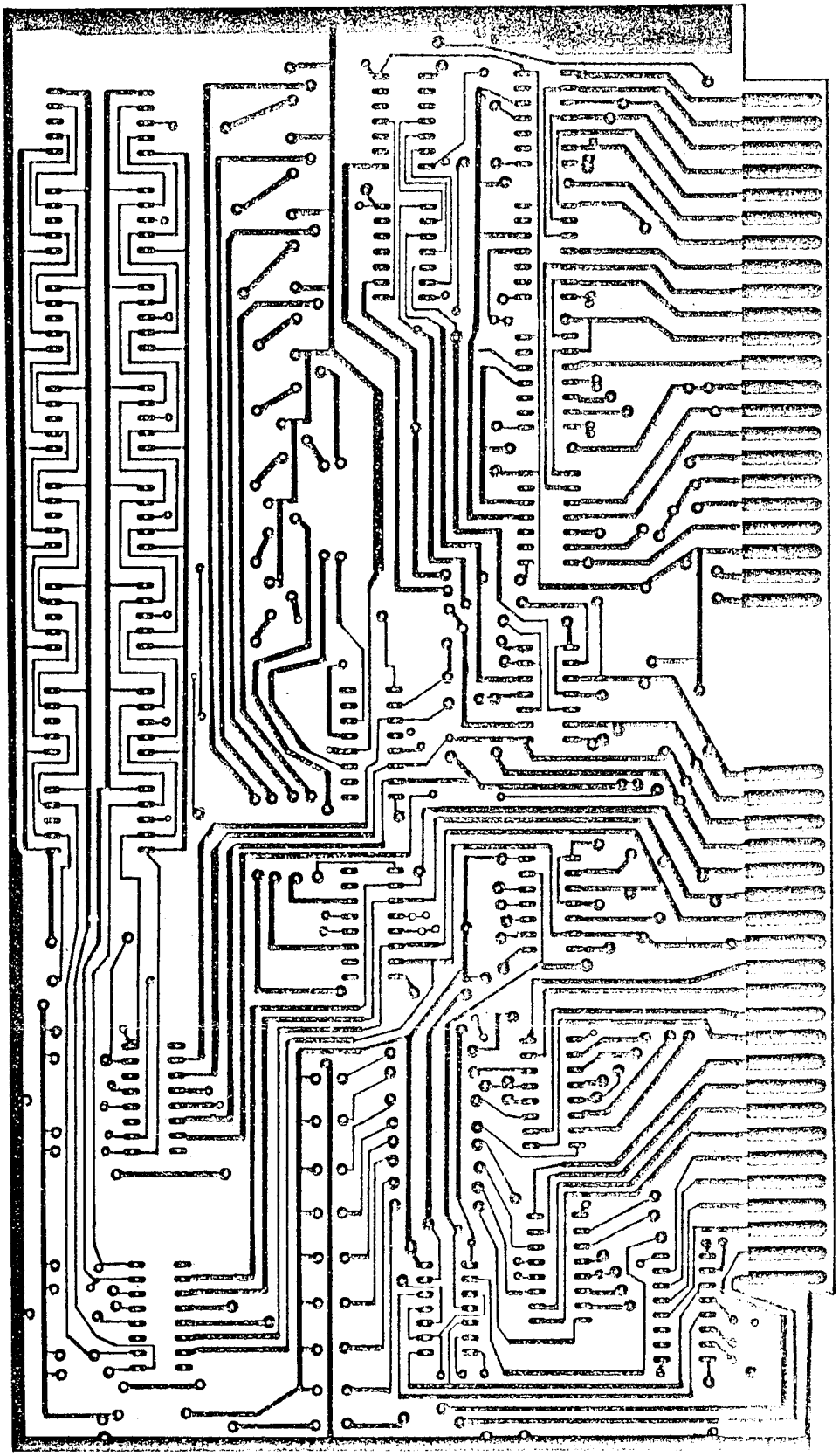
74126

7475

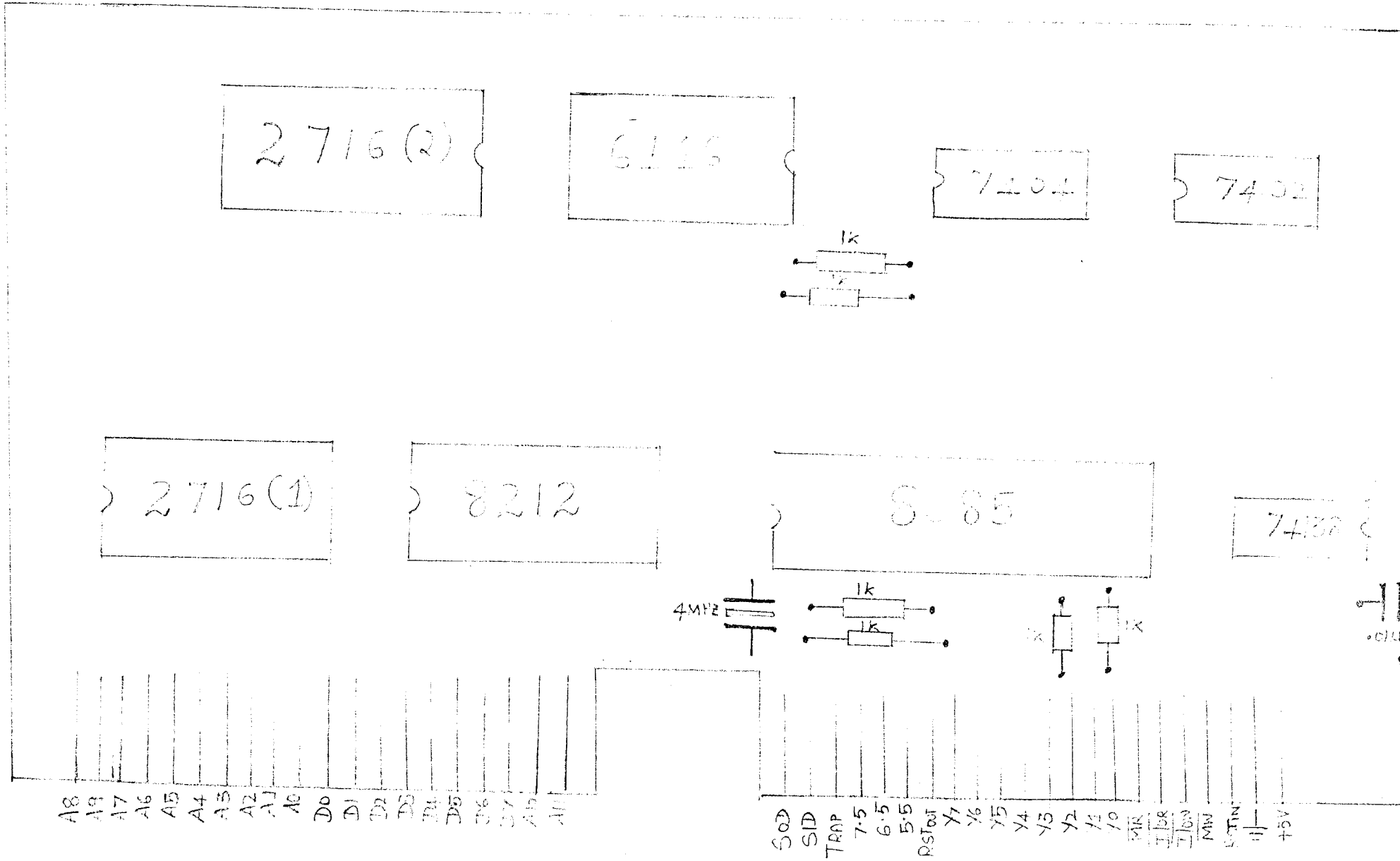
7475

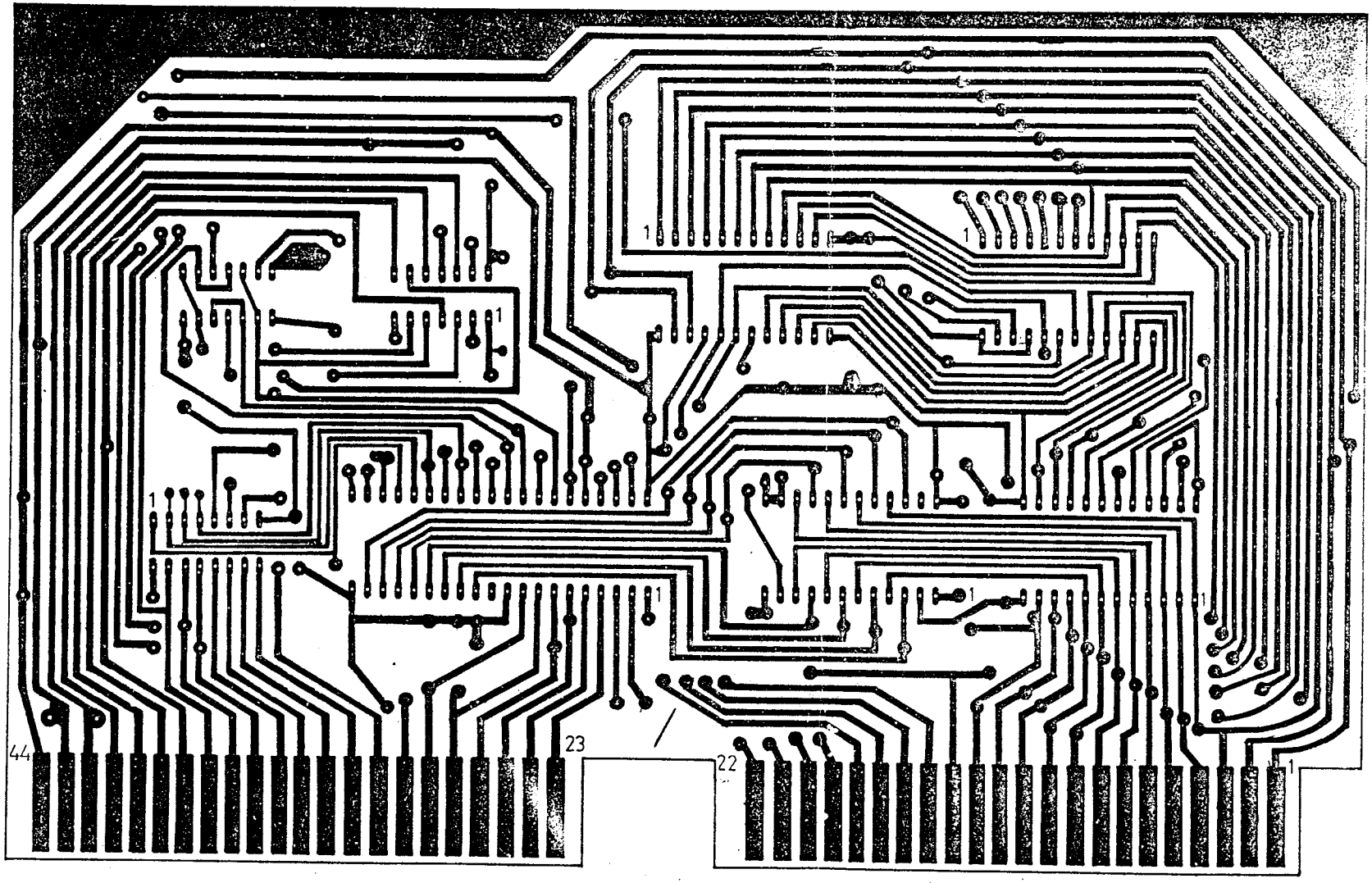
15k

7442

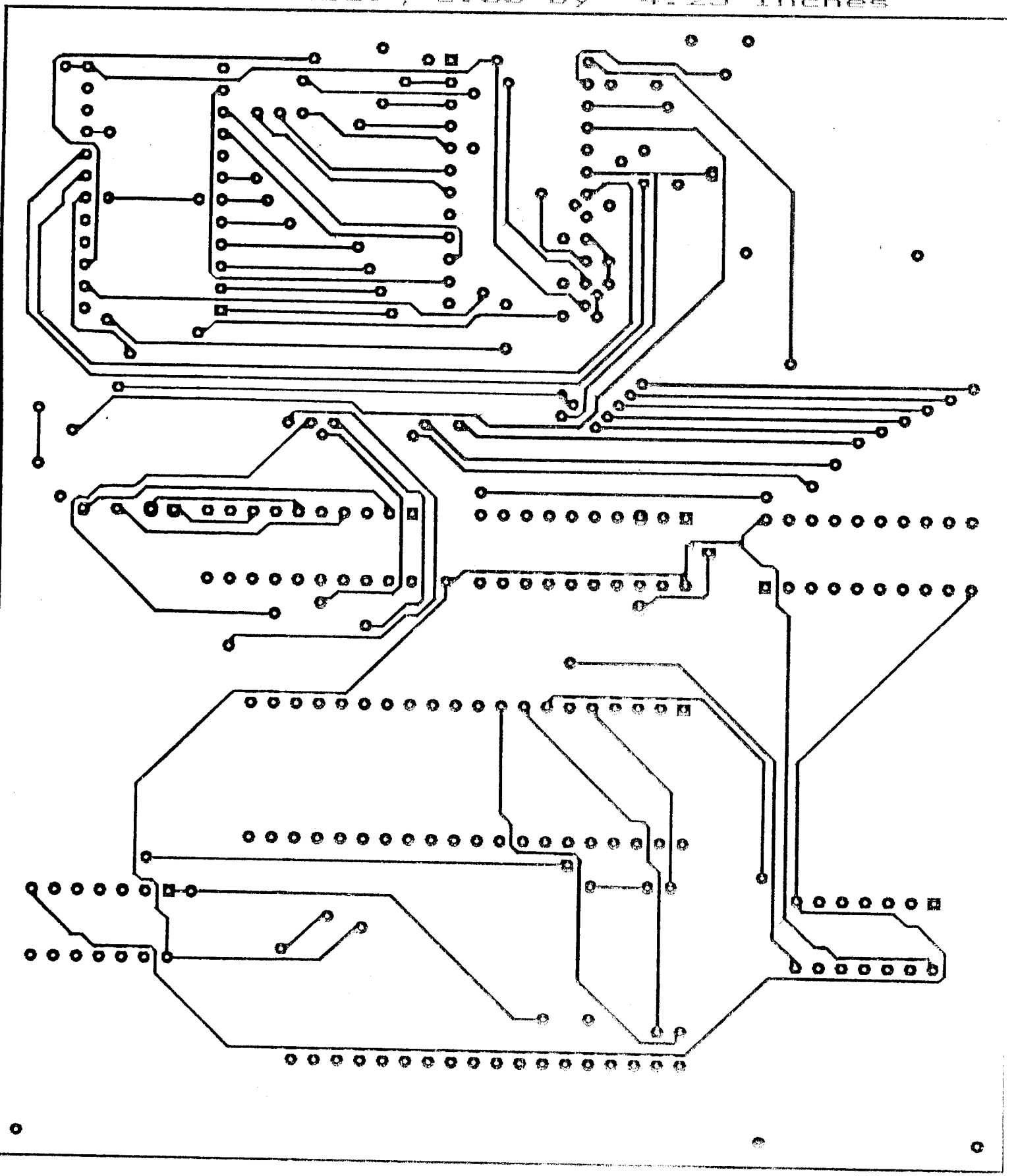


LAY OUT OF MAIN MICROPROCESSOR BOARD

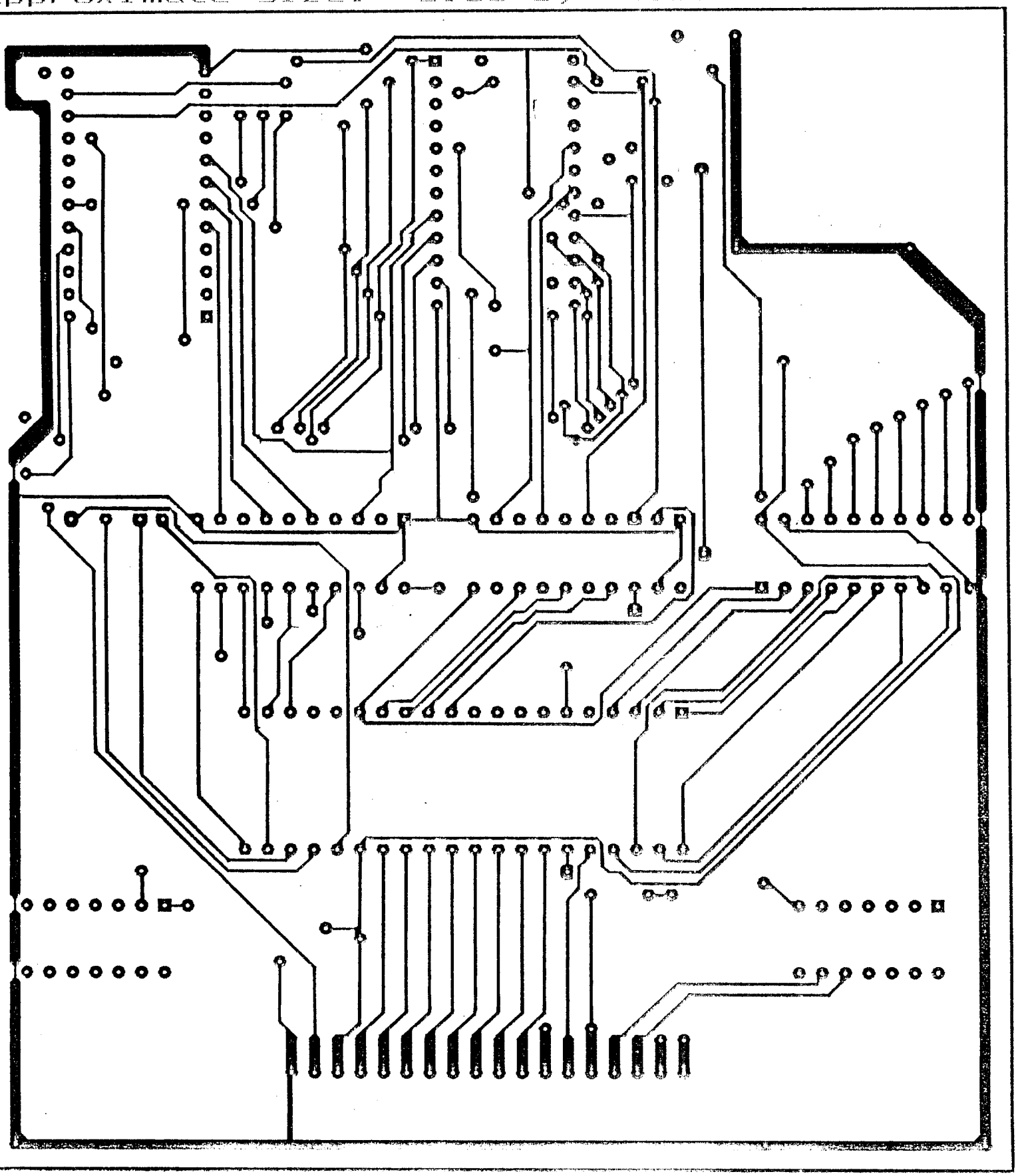



1
1
1
1
1
1
44
23
22

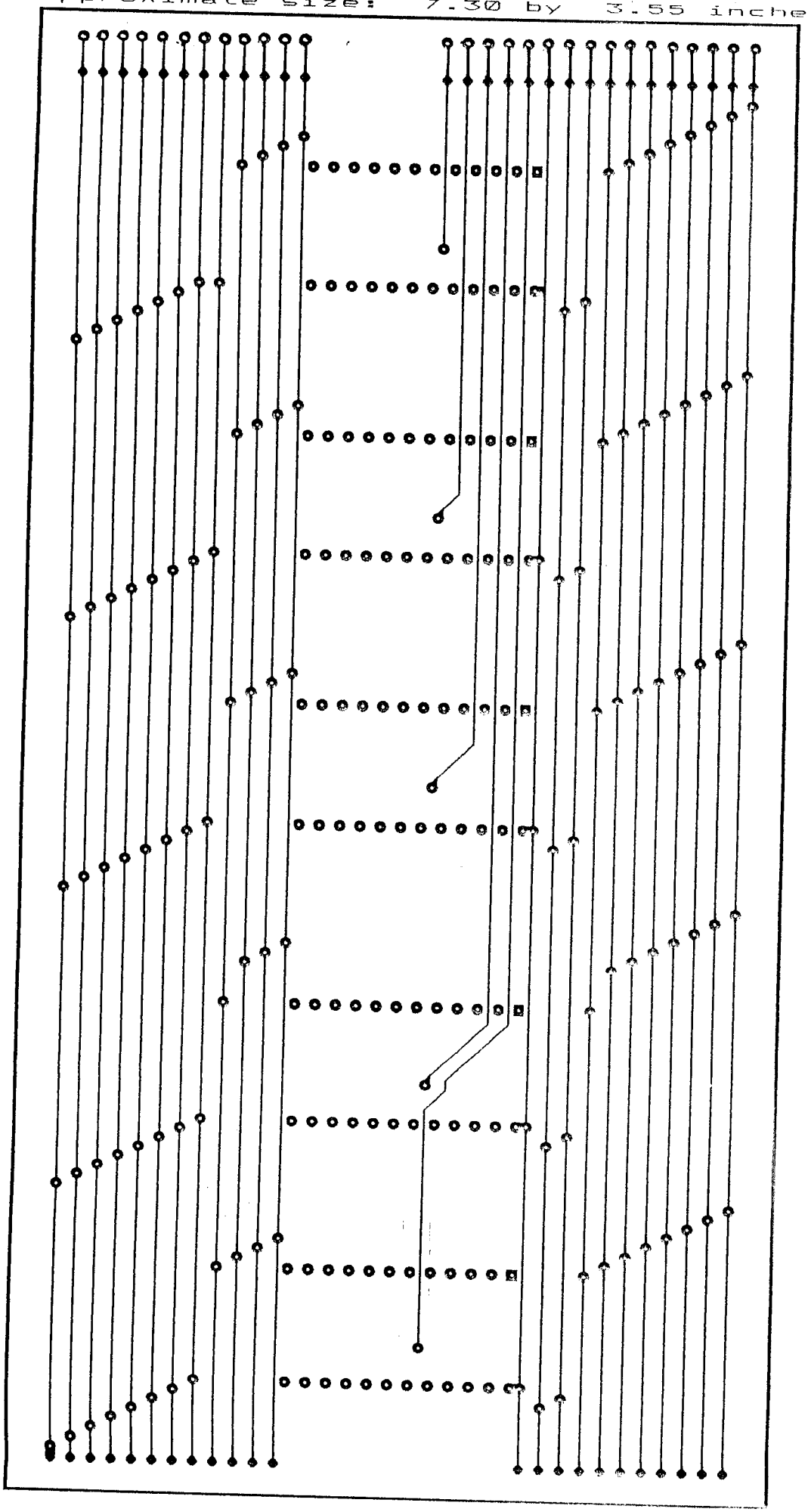
2X artwork 13 Feb 1992 17:15:19
a:jai
v1.2 r2 holes: 308 component side
approximate size: 5.00 by 4.25 inches



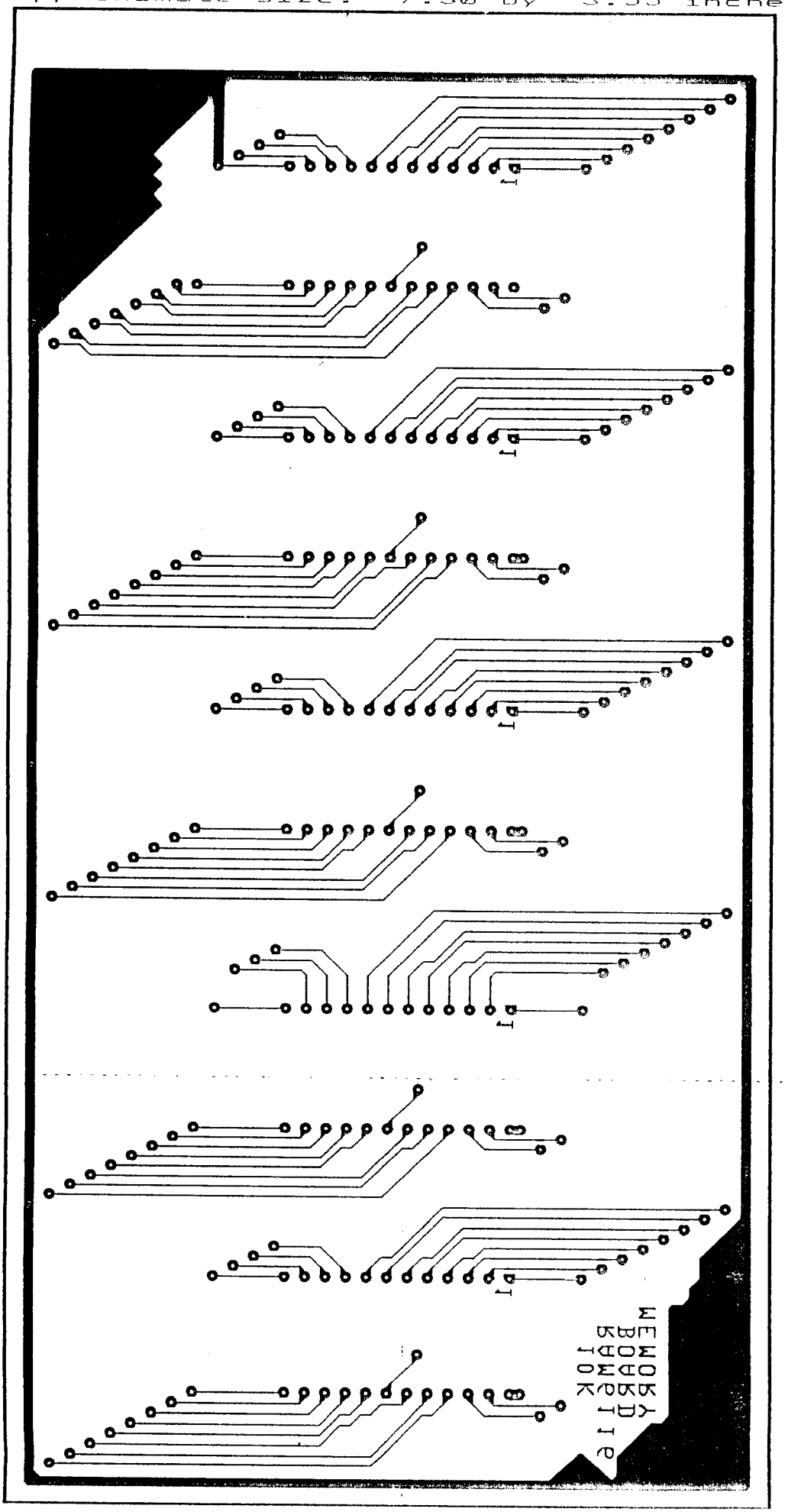
EX artwork 13 Feb 1992 17:03:54
: jai
1.2 r2 holes: 308 solder side
pproximate size: 5.00 by 4.25 inches



jk1:
v1.2 r2 holes: 239 component side
approximate size: 7.30 by 3.55 inches



2 16:42:00 14 Feb 199
jk1
v1.2 r2 holes: 239 solder side
approximate size: 7.30 by 3.55 inches



ADC0802-ADC0804

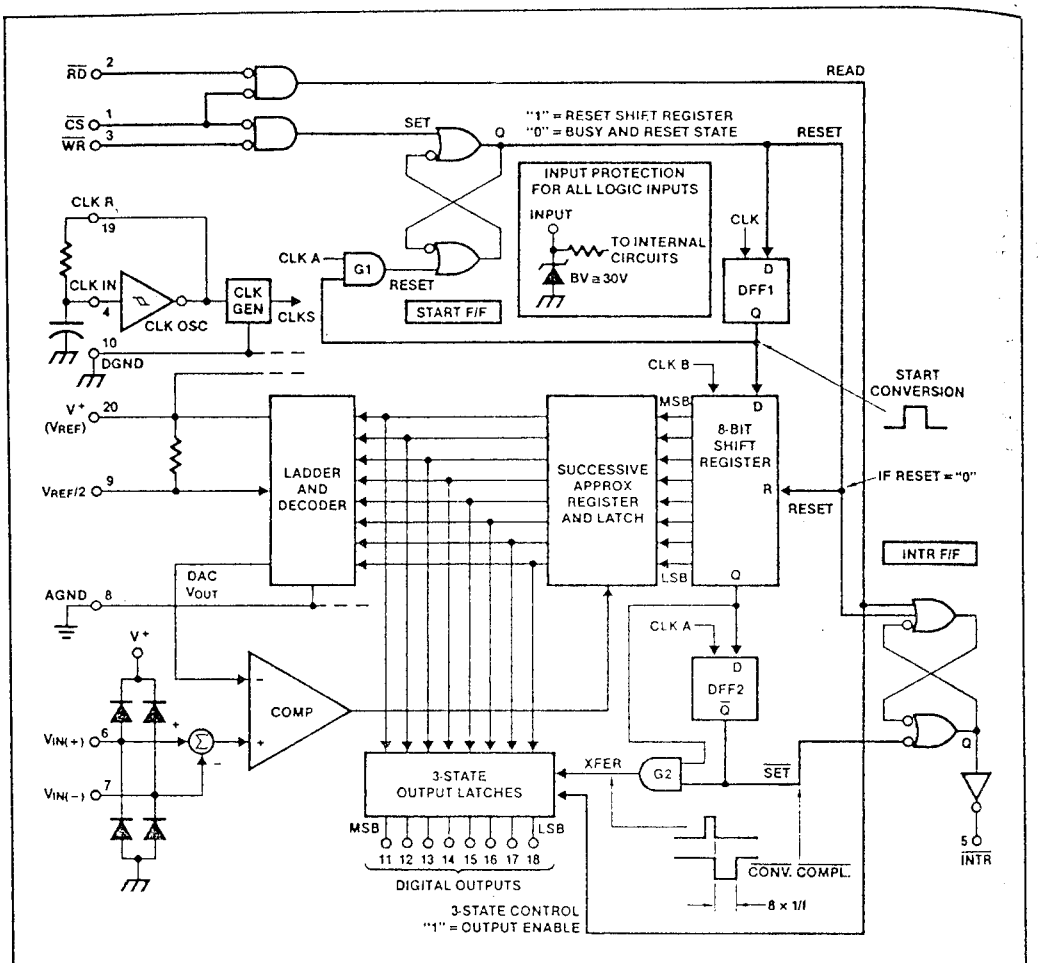
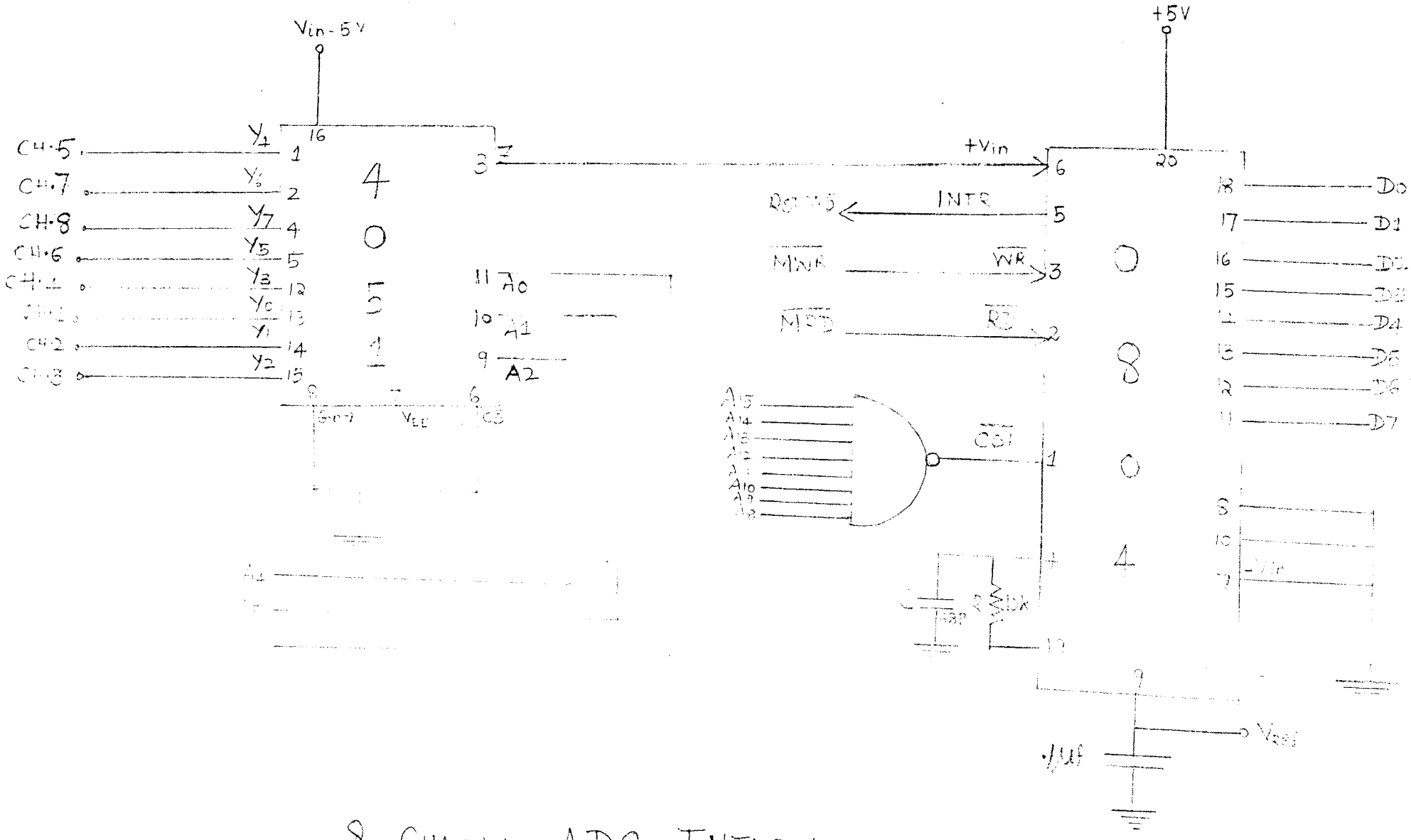
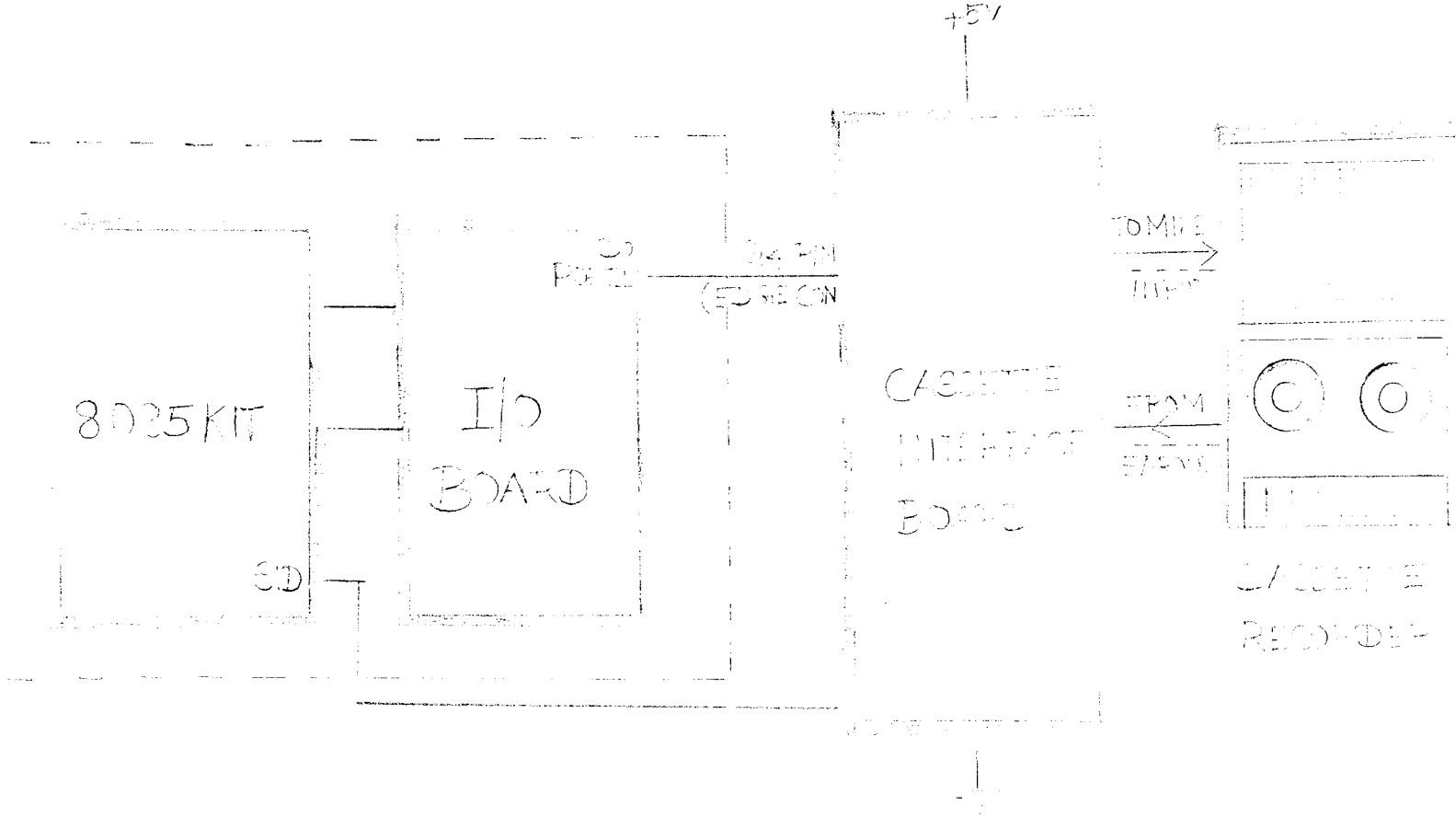


Figure 3: Functional Diagram of ADC0802–ADC0804

NOTE: All typical values have been characterized but are not tested.



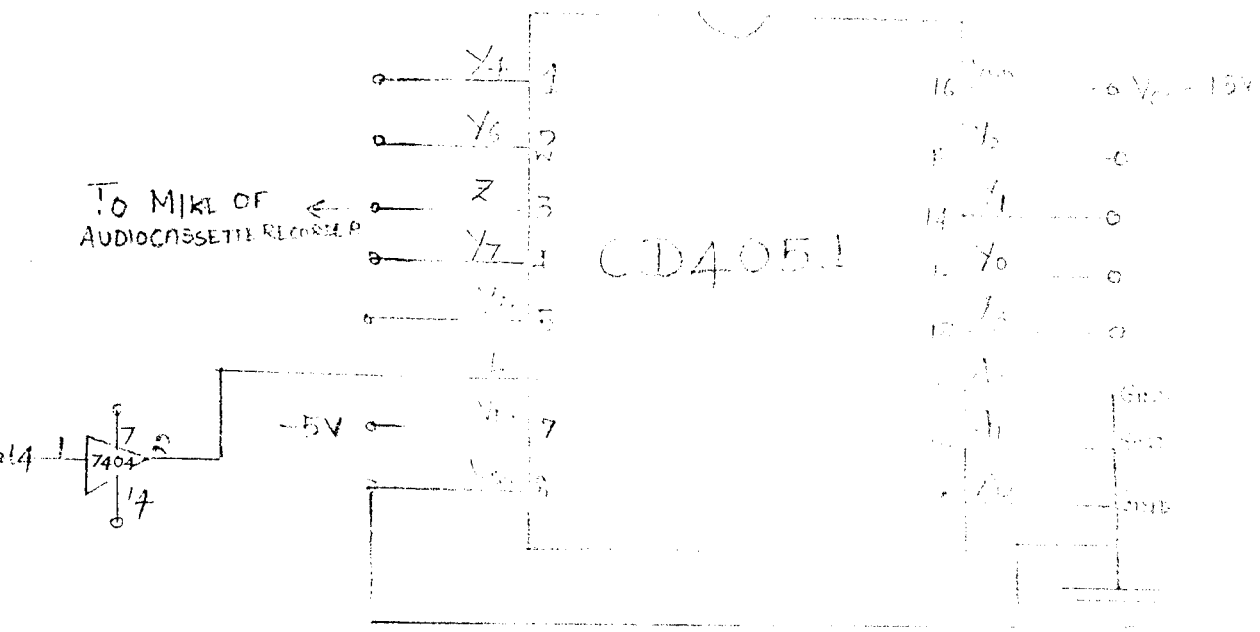
8-CHANNEL ADC INTERFACE



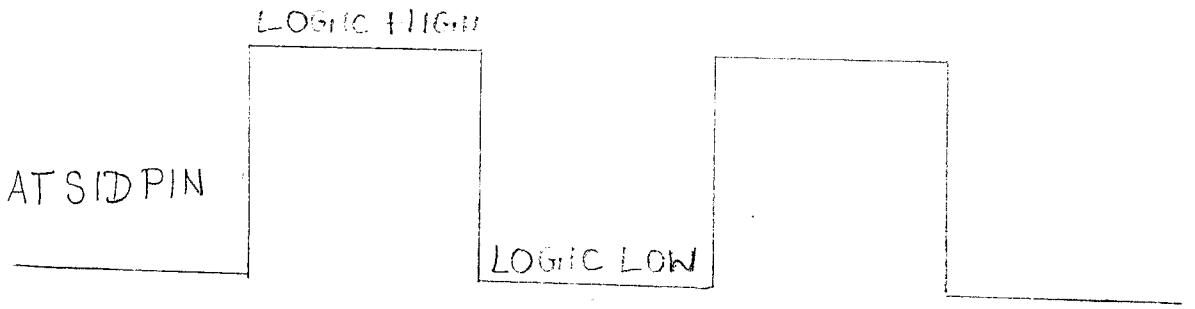
SCHEMATIC OF CASSETTE INTERFACE TO THE KIT

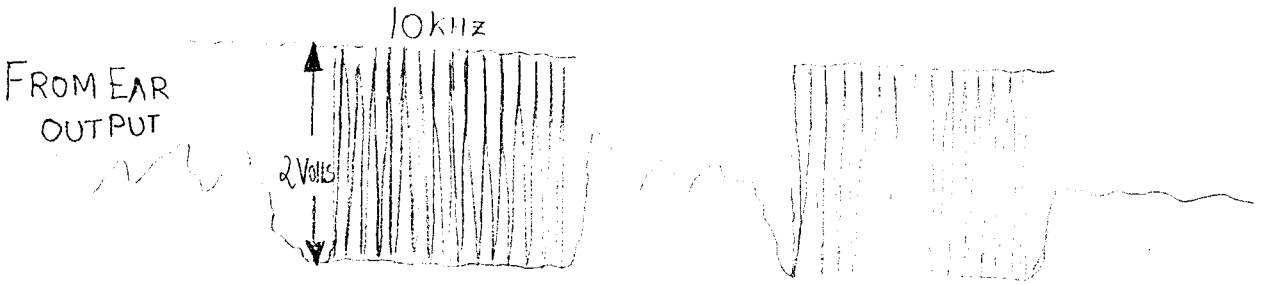
AND A CASSETTE RECORDER

Interface Circuit For Recording Into Audio Cassette



Interface Circuit For Recovery Of Data From Cassette Recorder





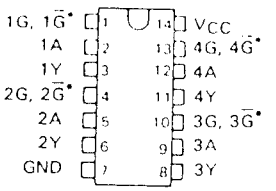
EARPHONE OUTPUT VOLTAGE WAVEFORM

TYPES SN54125, SN54126, SN54LS125A, SN54LS126A, SN74125, SN74126, SN74LS125A, SN74LS126A QUADRUPLE BUS BUFFERS WITH 3-STATE OUTPUTS

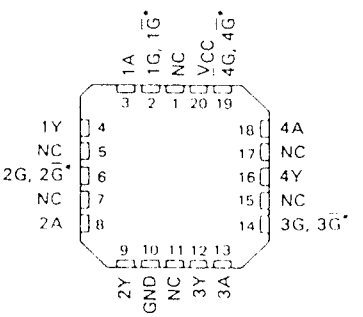
REVISED DECEMBER 1983

- **Quad Bus Buffers**
- **3-State Outputs**
- **Separate Control for Each Channel**

SN54125, SN54126, SN54LS125A, SN54LS126A . . . J OR W PACKAGE
SN74125, SN74126 . . . J OR N PACKAGE
SN74LS125A, SN74LS126A . . . D, J OR N PACKAGE
(TOP VIEW)



SN54LS125A, SN54LS126A . . . FK PACKAGE
SN74LS125A, SN74LS126A . . . FN PACKAGE
(TOP VIEW)



*$\overline{G}$ on '125, 'LS125; G on 126, 'LS126

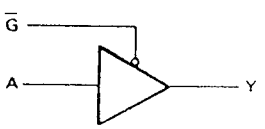
NC — No internal connection

scription

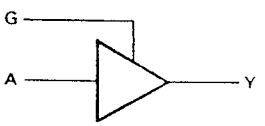
These bus buffers feature three-state outputs that, when enabled, have the low impedence characteristics of a TTL output with additional drive capability at high logic levels to permit driving heavily loaded bus lines without external pull-up resistors, when disabled, both output transistors are turned off presenting a high-impedance state to the bus so the output will act neither as a significant load nor as a driver. The '125 and 'LS125A outputs are disabled when $\overline{G}$ is high. The '126 and 'LS126A outputs are disabled when G is low.

ic diagram (each gate)

'125, 'LS125A



'126, LS126A



itive logic Y = A

solute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1) . . . . . 7 V
Input voltage: '125, '126 . . . . . 5.5 V
'LS125A, 'LS126A . . . . . 7 V
Operating free-air temperature range: SN54' . . . . . −55°C to 125°C
SN74' . . . . . 0°C to 70°C
Storage temperature range . . . . . −65°C to 150°C

TE 1: Voltage values are with respect to network ground terminal

recommended operating conditions

	SN54125, SN54126			SN74125, SN74126			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC} Supply voltage	4.5	5	5.5	4.75	5	5.25	V
V_{IH} High-level input voltage	2			2			V
V_{IL} Low-level input voltage			0.8			0.8	V
I_{OH} High-level output current			−2			−5.2	mA
I_{OL} Low-level output current			16			16	mA
T_A Operating free-air temperature	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†		SN54125, SN54126			SN74125, SN74126			UNIT
			MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IK}	V_{CC} = MIN, I_I = −12 mA				−1.5			−1.5	V
V_{OH}	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V	I_{OH} = −2 V	2.4	3.3					V
		I_{OH} = −5.2 V				2.4	3.1		
V_{OL}	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OL} = 16 mA				0.4			0.4	V
I_{OZ}	V_{CC} = MAX, V_{IH} = 2 V, V_{IL} = 0.8 V	V_O = 2.4 V			40			40	μA
		V_O = 0.4 V			−40			−40	
I_I	V_{CC} = MAX, V_I = 5.5 V				1			1	mA
I_{IH}	V_{CC} = MAX, V_I = 2.4 V				40			40	μA
I_{IL}	V_{CC} = MAX, V_I = 0.4 V				−1.6			−1.6	mA
I_{OS}§	V_{CC} = MAX		−30		−70	−28		−70	mA
I_{CC}	V_{CC} = MAX, (see Note 2)	'125		32	54		32	54	mA
		'126		36	62		36	62	

† For condition shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡ All typical values are at V_{CC} = 5 V, T_A = 25°C.
§ Not more than one output should be shorted at a time.
NOTE 2: Data inputs 0 V, output control = 4.5 V for '125 and 0 V for '126.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see note 3)

PARAMETER	TEST CONDITIONS		SN54/74125			SN54/74126			UNIT
			MIN	TYP	MAX	MIN	TYP	MAX	
t_{PLH}	R_L = 400 Ω,	C_L = 50 pF		8	13		8	13	ns
t_{PHL}				12	18		12	18	ns
t_{PZH}				11	17		11	18	ns
t_{PZL}				16	25		16	25	ns
t_{PHZ}	R_L = 400 Ω,	C_L = 5 pF		5	8		10	16	ns
t_{PLZ}				7	12		12	18	ns

NOTE 3: See General Information Section for load circuits and voltage waveforms.

Absolute Maximum Ratings

Supply Voltage	±18V or 36V
Power Dissipation (Note 1)	500 mW
Reference Input Differential Voltage (V14 to V15)	V^- to V^+
Reference Input Common-Mode Range (V14, V15)	V^- to V^+
Reference Input Current	5 mA
Digital Inputs	V^- to V^- plus 36V
Analog Current Outputs	*Figure 24*
Storage Temperature	-65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	300°C

Operating Conditions

	MIN	MAX	UNITS
Temperature (T_A)			
DAC0802L	-55	+125	°C
DAC0800L	-55	+125	°C
DAC0800LC	0	+70	°C
DAC0801LC	0	+70	°C
DAC0802LC	0	+70	°C

Electrical Characteristics (V_S = ±15V, I_{REF} = 2 mA, $T_{MIN} \le T_A \le T_{MAX}$ unless otherwise specified. Output characteristics refer to both I_{OUT} and $\overline{I_{OUT}}$.)

PARAMETER	CONDITIONS	DAC0802L/ DAC0802LC			DAC0800L/ DAC0800LC			DAC0801LC			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
Resolution		8	8	8	8	8	8	8	8	8	Bits
Monotonicity		8	8	8	8	8	8	8	8	8	Bits
Nonlinearity				±0.1			±0.19			±0.39	%FS
Settling Time	To ±1/2 LSB, All Bits Switched "ON" or "OFF", T_A = 25°C		100	135					100	150	ns
	DAC0800L					100	135				ns
	DAC0800LC					100	150				ns
t_{PHL} Propagation Delay	T_A = 25°C										
Each Bit			35	60		35	60		35	60	ns
All Bits Switched			35	60		35	60		35	60	ns
Full Scale Tempco			±10	±50		±10	±50		±10	±80	ppm/°C
Output Voltage Compliance	Full Scale Current Change < 1/2 LSB, R_{OUT} > 20 MΩ Typ	-10		18	10		18	10		18	V
Full Scale Current	V_{REF} = 10.000V, R14 = 5.000 kΩ R15 = 5.000 kΩ, T_A = 25°C	1.984	1.992	2.000	1.94	1.99	2.04	1.94	1.99	2.04	mA
Full Scale Symmetry	$I_{FS4} - I_{FS2}$		±0.5	±4.0		±1	±8.0		±2	±16	µA
Zero Scale Current			0.1	1.0		0.2	2.0		0.2	4.0	µA
Output Current Range	V^- = -5V	0	2.0	2.1	0	2.0	2.1	0	2.0	2.1	mA
	V^- = -8V to -18V	0	2.0	4.2	0	2.0	4.2	0	2.0	4.2	mA
Logic Input Levels											
Logic "0"	V_{LC} = 0V			0.8			0.8			0.8	V
Logic "1"		2.0			2.0			2.0			V
Logic Input Current	V_{LC} = 0V										
Logic "0"	-10V ≤ V_{IN} ≤ +0.8V		-2.0	-10		-2.0	-10		-2.0	-10	µA
Logic "1"	2V ≤ V_{IN} ≤ +18V		0.002	10		0.002	10		0.002	10	µA
Logic Input Swing	V^- = -15V	-10		18	-10		18	-10		18	V
Logic Threshold Range	V_S = +15V	10		13.5	10		13.5	10		13.5	V
Reference Bias Current			-1.0	-3.0		-1.0	-3.0		-1.0	-3.0	µA
Reference Input Slew Rate	*(Figure 24)*	4.0	8.0		4.0	8.0		4.0	8.0		mA/µs
Power Supply Sensitivity	4.5V ≤ V^+ ≤ 18V		0.0001	0.01		0.0001	0.01		0.0001	0.01	%/%
	-4.5V ≤ V^- ≤ -18V		0.0001	0.01		0.0001	0.01		0.0001	0.01	%/%
	I_{REF} = 1 mA										
Power Supply Current	V_S = ±5V, I_{REF} = 1 mA										
			2.3	3.8		2.3	3.8		2.3	3.8	mA
			4.3	5.8		4.3	5.8		4.3	5.8	mA
	V_S = 5V, -15V, I_{REF} = 2 mA										
			2.4	3.8		2.4	3.8		2.4	3.8	mA
			6.4	7.8		6.4	7.8		6.4	7.8	mA
	V_S = ±15V, I_{REF} = 2 mA										
			2.5	3.8		2.5	3.8		2.5	3.8	mA
			6.5	7.8		6.5	7.8		6.5	7.8	mA
Power Dissipation	±5V, I_{REF} = 1 mA		33	48		33	48		33	48	mW
	5V, -15V, I_{REF} = 2 mA		108	136		108	136		108	136	mW
	±15V, I_{REF} = 2 mA		135	174		135	174		135	174	mW

Note 1: The maximum junction temperature of the DAC0800, DAC0801 and DAC0802 is 125°C. For operating at elevated temperatures, devices in the dual-in-line J or D package must be derated based on a thermal resistance of 100°C/W, junction to ambient, 175°C/W for the molded dual-

recommended operating conditions

		SN54LS125A SN54LS126A			SN74LS125A SN74LS126A			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage	4.5	5	5.5	4.75	5	5.25	V
V_{IH}	High-level input voltage	2			2			V
V_{IL}	Low-level input voltage			0.7			0.8	V
I_{OH}	High-level output current			−1			−2.6	mA
I_{OL}	Low-level output current			12			24	mA
T_A	Operating free-air temperature	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†			SN54LS125A SN54LS126A			SN74LS125A SN74LS126A			UNIT
				MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IK}	V_{CC} = MIN, I_I = −18 mA					−1.5			−1.5	V
V_{OH}	V_{CC} = MIN, V_{IH} = 2 V	V_{IL} = 0.7 V,	I_{OH} = −1 mA	2.4						V
		V_{IL} = 0.8 V,	I_{OH} = −2.6 mA				2.4			
V_{OL}	V_{CC} = MIN, V_{IH} = 2 V	V_{IL} = 0.7 V,	I_{OL} = 12 mA		0.25	0.4				V
		V_{IL} = 0.8 V,	I_{OL} = 12 mA					0.25	0.4	
		V_{IL} = 0.8 V,	I_{OL} = 24 mA					0.35	0.5	
I_{OZ}	V_{CC} = MAX, V_{IH} = 2 V	V_{IL} = 0.7 V	V_O = 2.4 V			20				μA
			V_O = 0.4 V			−20				
		V_{IL} = 0.8 V	V_O = 2.4 V						20	
			V_O = 0.4 V						−20	
I_I	V_{CC} = MAX,	V_I = 7 V				0.1			0.1	mA
I_{IH}	V_{CC} = MAX,	V_I = 2.7 V				20			20	μA
I_{IL}	V_{CC} = MAX, V_I = 0.4 V	'LS125A-$\overline{G}$ inputs				−0.2			−0.2	mA
		'LS125A-A inputs, 'LS126A-All inputs				−0.4			−0.4	mA
I_{OS}§	V_{CC} = MAX			−40		−225	−40		−225	mA
I_{CC}	V_{CC} = MAX, (see Note 2)	'LS125A			11	20		11	20	mA
		'LS126A			12	22		12	22	

†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡All typical values are at V_{CC} = 5 V, T_A = 25°C.
§Not more than one output should be shorted at a time, and duration of the short circuit should not exceed one second.
NOTE 2: Data inputs = 0 V, Output controls = 4.5 V for 'LS125A and 0 V for 'LS126A.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see note 3)

PARAMETER	TEST CONDITIONS		SN54/74LS125A			SN54/74LS126A			UNIT
			MIN	TYP	MAX	MIN	TYP	MAX	
t_{PLH}	R_L = 667 Ω,	C_L = 45 pF		9	15		9	15	ns
t_{PHL}				7	18		8	18	ns
t_{PZH}				12	20		16	25	ns
t_{PZL}				15	25		21	35	ns
t_{PHZ}	R_L = 667 Ω,	C_L = 5 pF			20			25	ns
t_{PLZ}					20			25	ns

NOTE 3: See General Information Section for load circuits and voltage waveforms.

TYPES SN54LS138, SN54S138, SN74LS138, SN74S138
3-LINE TO 8-LINE DECODERS/DEMULTIPLEXERS

DECEMBER 1972 - REVISED APRIL 1985

- **Designed Specifically for High-Speed:**
  Memory Decoders
  Data Transmission Systems
- **3 Enable Inputs to Simplify Cascading and/or Data Reception**
- **Schottky-Clamped for High Performance**

description

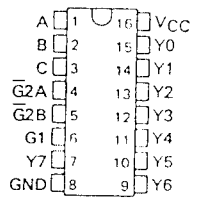
These Schottky-clamped TTL MSI circuits are designed to be used in high-performance memory decoding or data-routing applications requiring very short propagation delay times. In high-performance memory systems these decoders can be used to minimize the effects of system decoding. When employed with high-speed memories utilizing a fast enable circuit the delay times of these decoders and the enable time of the memory are usually less than the typical access time of the memory. This means that the effective system delay introduced by the Schottky-clamped system decoder is negligible.

The 'LS138 and 'S138 decode one of eight lines dependent on the conditions at the three binary select inputs and the three enable inputs. Two active-low and one active-high enable inputs reduce the need for external gates or inverters when expanding. A 24-line decoder can be implemented without external inverters and a 32-line decoder requires only one inverter. An enable input can be used as a data input for demultiplexing applications.

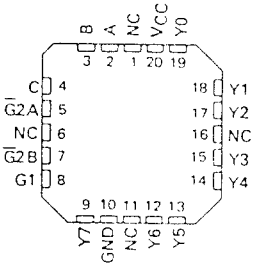
All of these decoder/demultiplexers feature fully buffered inputs, each of which represents only one normalized load to its driving circuit. All inputs are clamped with high-performance Schottky diodes to suppress line-ringing and to simplify system design.

The SN54LS138 and SN54S138 are characterized for operation over the full military temperature range of −55°C to 125°C. The SN74LS138 and SN74S138 are characterized for operation from 0°C to 70°C.

SN54LS138, SN54S138 . . . J OR W PACKAGE
SN74LS138, SN74S138 . . . D, J OR N PACKAGE
(TOP VIEW)

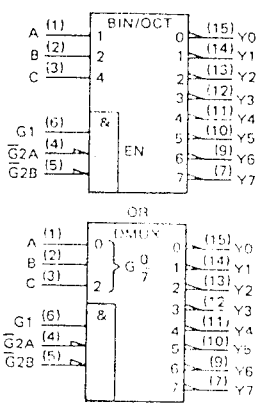


SN54LS138, SN54S138 . . . FK PACKAGE
SN74LS138, SN74S138 . . . FN PACKAGE
(TOP VIEW)



NC – No internal connection

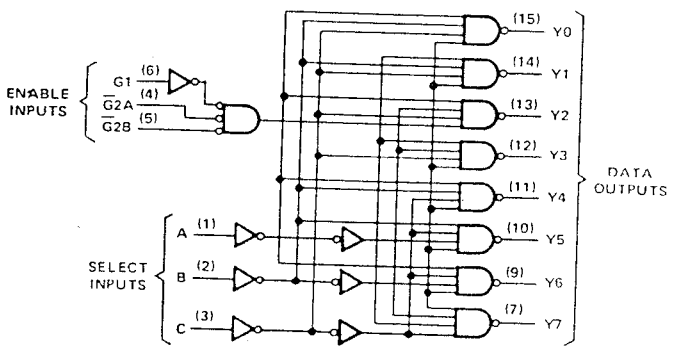
logic symbols



Pin numbers shown on logic notation are for D, J or N packages.

logic diagram and function table

'LS138, 'S138



Pin numbers shown on logic notation are for D, J or N packages.

'LS138, 'S138
FUNCTION TABLE

INPUTS					OUTPUTS							
ENABLE		SELECT										
G1	$\overline{G2}$*	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

*$\overline{G2} = \overline{G2A} + \overline{G2B}$

H = high level, L = low level, X = irrelevant

TEXAS INSTRUMENTS
POST OFFICE BOX 225012 • DALLAS, TEXAS 75265

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1) . . . . . 7 V
Input voltage . . . . . 7 V
Operating free-air temperature range: SN54LS138 . . . . . −55°C to 125°C
SN74LS138 . . . . . 0°C to 70°C
Storage temperature range . . . . . −65°C to 150°C

NOTE 1: Voltage values are with respect to network ground terminal.

recommended operating conditions

		SN54LS138			SN74LS138			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage	4.5	5	5.5	4.75	5	5.25	V
V_{IH}	High-level input voltage	2			2			V
V_{IL}	Low-level input voltage			0.7			0.8	V
I_{OH}	High-level output current			−0.4			−0.4	mA
I_{OL}	Low-level output current			4			8	mA
T_A	Operating free-air temperature	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS †		SN54LS138			SN74LS138			UNIT
			MIN	TYP‡	MAX	MIN	TYP‡	MAX	
V_{IK}	V_{CC} = MIN, I_I = −18 mA				−1.5			−1.5	V
V_{OH}	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = MAX, I_{OH} = −0.4 mA		2.5	3.4		2.7	3.4		V
V_{OL}	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = MAX	I_{OL} = 4 mA		0.25	0.4		0.25	0.4	V
		I_{OL} = 8 mA					0.35	0.5	
I_I	V_{CC} = MAX, V_I = 7 V				0.1			0.1	mA
I_{IH}	V_{CC} = MAX, V_I = 2.7 V				20			20	μA
I_{IL}	V_{CC} = MAX, V_I = 0.4 V	Enable			−0.4			−0.4	mA
		A, B, C			−0.2			−0.2	
I_{OS} §	V_{CC} = MAX		−20		−100	−20		−100	mA
I_{CC}	V_{CC} = MAX, Outputs enabled and open			6.3	10		6.3	10	mA

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡ All typical values are at V_{CC} = 5 V, T_A = 25°C.
§ Not more than one output should be shorted at a time.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER¶	FROM (INPUT)	TO (OUTPUT)	LEVELS OF DELAY	TEST CONDITIONS	SN54LS138 SN74LS138			UNIT
					MIN	TYP	MAX	
t_{PLH}	Binary Select	Any	2	R_L = 2 kΩ, C_L = 15 pF, See Note 2		11	20	ns
t_{PHL}						18	41	ns
t_{PLH}			3			21	27	ns
t_{PHL}						20	39	ns
t_{PLH}	Enable	Any	2			12	18	ns
t_{PHL}						20	32	ns
t_{PLH}			3			14	26	ns
t_{PHL}						13	38	ns

¶ t_{PLH} = propagation delay time, low-to-high-level output; t_{PHL} = propagation delay time, high-to-low-level output.
NOTE 2: See General Information Section for load circuits and voltage waveforms.

TEXAS INSTRUMENTS
POST OFFICE BOX 225012 • DALLAS, TEXAS 75265

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1) . . . . . 7 V
Input voltage . . . . . 5.5 V
Operating free-air temperature range: SN54S138 . . . . . −55°C to 125°C
SN74S138 . . . . . 0°C to 70°C
Storage temperature range . . . . . −65°C to 150°C

NOTE 1: Voltage values are with respect to network ground terminal.

recommended operating conditions

		SN54S138			SN74S138			UNIT
		MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage	4.5	5	5.5	4.75	5	5.25	V
V_{IH}	High-level input voltage	2			2			V
V_{IL}	Low-level input voltage			0.8			0.8	V
I_{OH}	High-level output current			−1			−1	mA
I_{OL}	Low-level output current			20			20	mA
T_A	Operating free-air temperature	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER	TEST CONDITIONS†		SN54S138 SN74S138			UNIT
			MIN	TYP‡	MAX	
V_{IK}	V_{CC} = MIN, I_I = 18 mA				−1.2	V
V_{OH}	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OH} = −1 mA	SN54S'	2.5	3.4		V
		SN74S'	2.7	3.4		
V_{OL}	V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OL} = 20 mA				0.5	V
I_I	V_{CC} = MAX, V_I = 5.5 V				1	mA
I_{IH}	V_{CC} = MAX, V_I = 2.7 V				50	μA
I_{IL}	V_{CC} = MAX, V_I = 0.5 V				−2	mA
I_{OS}§	V_{CC} = MAX		−40		−100	mA
I_{CC}	V_{CC} = MAX, Outputs enabled and open	SN54S'		60	74	mA
		SN74S'		75	90	

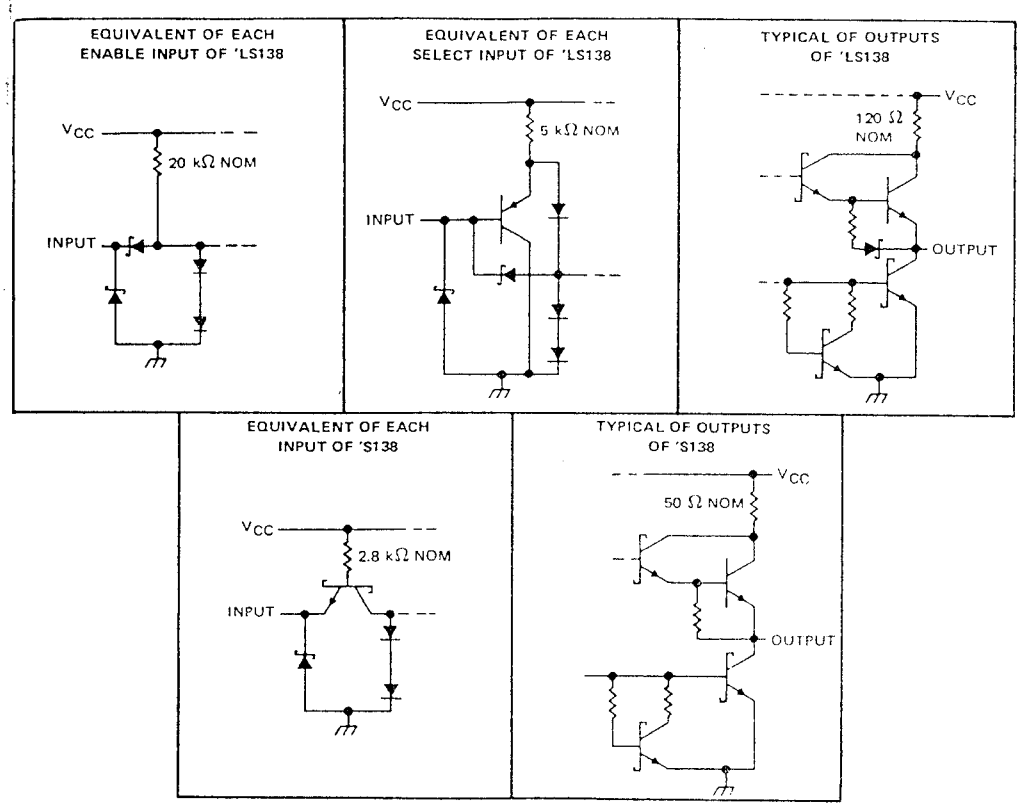
†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡All typical values are at V_{CC} = 5 V, T_A = 25°C.
§Not more than one output should be shorted at a time, and duration of the short circuit test should not exceed one second.

switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER¶	FROM (INPUT)	TO (OUTPUT)	LEVELS OF DELAY	TEST CONDITIONS	SN54S138 SN74S138		UNIT
					MIN	TYP MAX	
t_{PLH}	Binary Select	Any	2	R_L = 280 Ω, C_L = 15 pF, See Note 2		4.5 7	ns
t_{PHL}						7 10.5	
t_{PLH}			3			7.5 12	ns
t_{PHL}						8 12	
t_{PLH}	Enable	Any	2			5 8	ns
t_{PHL}						7 11	
t_{PLH}			3			7 11	ns
t_{PHL}						7 11	

¶ t_{PLH} = propagation delay time, low-to-high-level output
t_{PHL} = propagation delay time, high-to-low-level output
NOTE 2: See General Information Section for load circuits and voltage waveforms.

schematics of inputs and outputs



Signetics

Logic Products

7442, LS42
Decoders

BCD-To-Decimal Decoder (1-of-10)
Product Specification

FEATURES

- **Mutually exclusive outputs**
- **1-of-8 demultiplexing ability**
- **Outputs disabled for input codes above nine**

DESCRIPTION

The '42 decoder accepts four active HIGH BCD inputs and provides 10 mutually exclusive active LOW outputs, as shown by logic symbol or diagram. The active LOW outputs facilitate addressing other MSI units with active LOW input enables.

The logic design of the '42 ensures that all outputs are HIGH when binary codes greater than nine are applied to the inputs.

The most significant input, A_3, produces a useful inhibit function when the '42 is used as a 1-of-8 decoder. The A_3 input can also be used as the Data input in an 8-output demultiplexer application.

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
7442	15ns	28mA
74LS42	18ns	7mA

ORDERING CODE

PACKAGES	COMMERCIAL RANGE V_{CC} = 5V ±5%; T_A = 0°C to +70°C
Plastic DIP	N7442N, N74LS42N
Plastic SO	N74LS54D

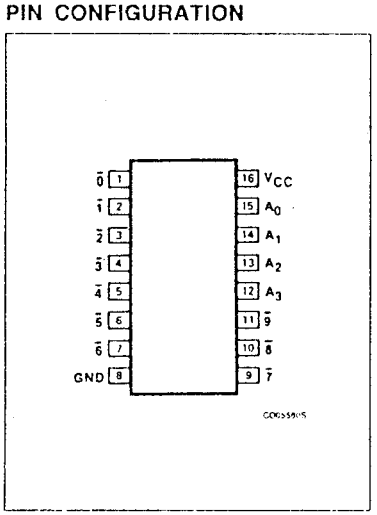
NOTE:
For information regarding devices processed to Military Specifications, see the Signetics Military Products Data Manual.

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

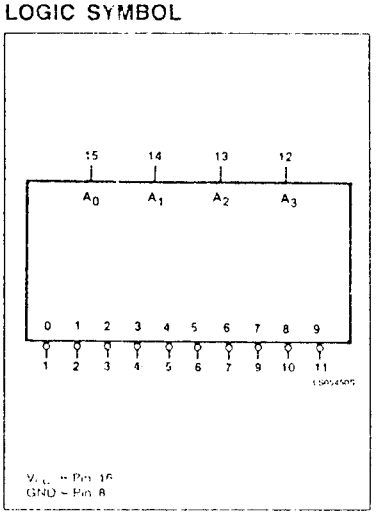
PINS	DESCRIPTION	74	74LS
$A_0 - A_3$	Inputs	1ul	1LSul
$\bar{0} - \bar{9}$	Outputs	10ul	10LSul

NOTE:
Where a 74 unit load (ul) is understood to be 40μA I_{IH} and −1.6mA I_{IL} and a 74LS unit load (LSul) is 20μA I_{IH} and −0.4mA I_{IL}.

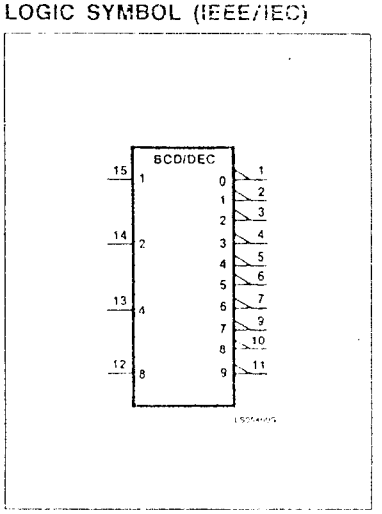
PIN CONFIGURATION



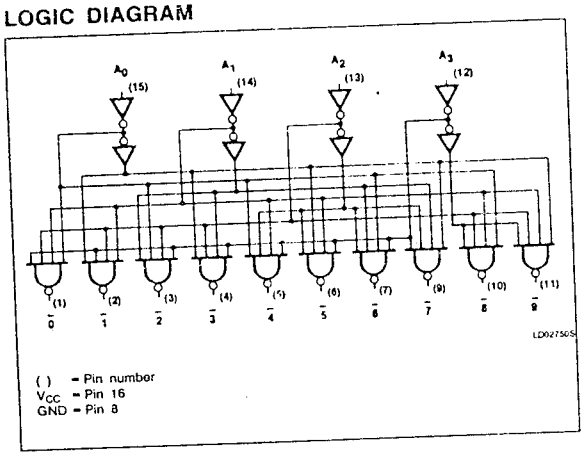
LOGIC SYMBOL



LOGIC SYMBOL (IEEE/IEC)



LOGIC DIAGRAM



FUNCTION TABLE

A_3	A_2	A_1	A_0	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$	$\overline{4}$	$\overline{5}$	$\overline{6}$	$\overline{7}$	$\overline{8}$	$\overline{9}$
L	L	L	L	L	H	H	H	H	H	H	H	H	H
L	L	L	H	H	L	H	H	H	H	H	H	H	H
L	L	H	L	H	H	L	H	H	H	H	H	H	H
L	L	H	H	H	H	H	L	H	H	H	H	H	H
L	H	L	L	H	H	H	H	L	H	H	H	H	H
L	H	L	H	H	H	H	H	H	L	H	H	H	H
L	H	H	L	H	H	H	H	H	H	L	H	H	H
L	H	H	H	H	H	H	H	H	H	H	L	H	H
H	L	L	L	H	H	H	H	H	H	H	H	L	H
H	L	L	H	H	H	H	H	H	H	H	H	H	L
H	L	H	L	H	H	H	H	H	H	H	H	H	H
H	L	H	H	H	H	H	H	H	H	H	H	H	H
H	H	L	L	H	H	H	H	H	H	H	H	H	H
H	H	L	H	H	H	H	H	H	H	H	H	H	H
H	H	H	L	H	H	H	H	H	H	H	H	H	H
H	H	H	H	H	H	H	H	H	H	H	H	H	H

H = HIGH voltage levels
L = LOW voltage levels

ABSOLUTE MAXIMUM RATINGS (Over operating free-air temperature range unless otherwise noted.)

	PARAMETER	74	74LS	UNIT
V_{CC}	Supply voltage	7.0	7.0	V
V_{IN}	Input voltage	−0.5 to +5.5	−0.5 to +7.0	V
I_{IN}	Input current	−30 to +5	−30 to +1	mA
V_{OUT}	Voltage applied to output in HIGH output state	−0.5 to +V_{CC}	−0.5 to +V_{CC}	V
T_A	Operating free-air temperature range	0 to 70		°C

RECOMMENDED OPERATING CONDITIONS

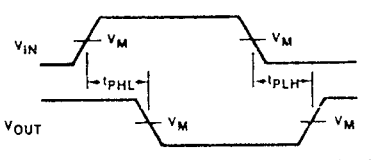
	PARAMETER	74			74LS			UNIT
		Min	Nom	Max	Min	Nom	Max	
V_{CC}	Supply voltage	4.75	5.0	5.25	4.75	5.0	5.25	V
V_{IH}	HIGH-level input voltage	2.0			2.0			V
V_{IL}	LOW-level input voltage			+0.8			+0.8	V
I_{IK}	Input clamp current			−12			−18	mA
I_{OH}	HIGH-level output current			−800			−400	µA
I_{OL}	LOW-level output current			16			8	mA
T_A	Operating free-air temperature	0		70	0		70	°C

DC ELECTRICAL CHARACTERISTICS (Over recommended operating free-air temperature range unless otherwise noted.)

PARAMETER		TEST CONDITIONS[1]		7442			74LS42			UNIT
				Min	Typ[2]	Max	Min	Typ[2]	Max	
V_{OH}	HIGH-level output voltage	V_{CC} = MIN, V_{IH} = MIN, V_{IL} = MAX, I_{OH} = MAX		2.4	3.4		2.7	3.4		V
V_{OL}	LOW-level output voltage	V_{CC} = MIN, V_{IH} = MIN, V_{IL} = MAX	I_{OL} = MAX		0.2	0.4		0.35	0.5	V
			I_{OL} = 4mA (74LS)					0.25	0.4	V
V_{IK}	Input clamp voltage	V_{CC} = MIN, I_I = I_{IK}				−1.5			−1.5	V
I_I	Input current at maximum input voltage	V_{CC} = MAX	V_I = 5.5V			1.0				mA
			V_I = 7.0V						0.1	mA
I_{IH}	HIGH-level input current	V_{CC} = MAX	V_I = 2.4V			40				μA
			V_I = 2.7V						20	μA
I_{IL}	LOW-level input current	V_{CC} = MAX, V_I = 0.4V				−1.6			−0.4	mA
I_{OS}	Short-circuit output current[3]	V_{CC} = MAX		−18		−55	−20		−100	mA
I_{CC}	Supply current[4] (total)	V_{CC} = MAX			28	56		7	13	mA

NOTES:
1. For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions for the applicable type.
2. All typical values are at V_{CC} = 5V, T_A = 25°C.
3. I_{OS} is tested with V_{OUT} = +0.5V and V_{CC} = V_{CC} MAX + 0.5V. Not more than one output should be shorted at a time and duration of the short circuit should not exceed one second.
4. I_{CC} is measured with all outputs open and all inputs grounded.

AC WAVEFORM



V_M = 1.3V for 74LS; V_M = 1.5V for all other TTL families.

Waveform 1. Waveform For Inverting Outputs

AC ELECTRICAL CHARACTERISTICS T_A = 25°C, V_{CC} = 5.0V

PARAMETER		TEST CONDITIONS	74		74LS		UNIT
			C_L = 15pF, R_L = 409Ω		C_L = 15pF, R_L = 2kΩ		
			Min	Max	Min	Max	
t_{PLH}	Propagation delay	Waveform 1		30		30	ns
t_{PHL}	Address to output	3 logic levels		30		30	
t_{PLH}	Propagation delay	Waveform 1		25		25	ns
t_{PHL}	Address to output	2 logic levels		25		25	

TYPES SN5475, SN5477, SN54L75, SN54L77, SN54LS75, SN54LS77, SN7475, SN74LS75
4-BIT BISTABLE LATCHES

MARCH 1974—REVISED DECEMBER 1983

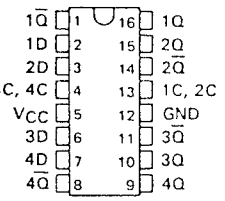
logic

FUNCTION TABLE
(each latch)

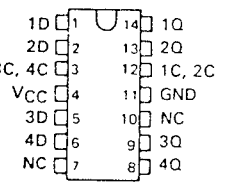
INPUTS		OUTPUTS	
D	C	Q	$\overline{Q}$
L	H	L	H
H	H	H	L
X	L	Q_0	$\overline{Q}_0$

H = high level, L = low level, X = irrelevant
Q_0 = the level of Q before the high-to-low transition of G

SN5475, SN54LS75 . . . J OR W PACKAGE
SN54L75 . . . J PACKAGE
SN7475 . . . J OR N PACKAGE
SN74LS75 . . . D, J OR N PACKAGE
(TOP VIEW)



SN5477, SN54LS77 . . . W PACKAGE
SN54L77 . . . T PACKAGE
(TOP VIEW)



NC – No internal connection

FOR CHIP CARRIER INFORMATION,
CONTACT THE FACTORY

description

These latches are ideally suited for use as temporary storage for binary information between processing units and input/output or indicator units. Information present at a data (D) input is transferred to the Q output when the enable (C) is high and the Q output will follow the data input as long as the enable remains high. When the enable goes low, the information (that was present at the data input at the time the transition occurred) is retained at the Q output until the enable is permitted to go high.

The '75, 'L75, and 'LS75 feature complementary Q and $\overline{Q}$ outputs from a 4-bit latch, and are available in various 16-pin packages. For higher component density applications, the '77, 'L77, and 'LS77 4-bit latches are available in 14-pin flat packages.

These circuits are completely compatible with all popular TTL families. All inputs are diode-clamped to minimize transmission-line effects and simplify system design. Series 54, 54L, and 54LS devices are characterized for operation over the fully military temperature range of −55°C to 125°C; Series 74, and 74LS devices are characterized for operation from 0°C to 70°C.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1) . . . . . 7 V
Input voltage: '75, 'L75, '77, 'L77 . . . . . 5.5 V
'LS75, 'LS77 . . . . . 7 V
Interemitter voltage (see Note 2) . . . . . 5.5 V
Operating free-air temperature range: SN54' . . . . . −55°C to 125°C
SN74' . . . . . 0°C to 70°C
Storage temperature range . . . . . −65°C to 150°C

NOTES: 1. Voltage values are with respect to network ground terminal.
2. This is the voltage between two emitters of a multiple-emitter input transistor and is not applicable to the 'LS75 and 'LS77.

TEXAS INSTRUMENTS
POST OFFICE BOX 225012 • DALLAS, TEXAS 75265

recommended operating conditions

	SN5475, SN5477			SN7475			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}	4.5	5	5.5	4.75	5	5.25	V
High-level output current, I_{OH}			−400			−400	μA
Low-level output current, I_{OL}			16			16	mA
Width of enabling pulse, t_w	20			20			ns
Setup time, t_{su}	20			20			ns
Hold time, t_h	5			5			ns
Operating free-air temperature, T_A	−55		125	0		70	°C

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER			TEST CONDITIONS†		MIN	TYP‡	MAX	UNIT
V_{IH}	High-level input voltage				2			V
V_{IL}	Low-level input voltage						0.8	V
V_{IK}	Input clamp voltage		V_{CC} = MIN, I_I = −12 mA				−1.5	V
V_{OH}	High-level output voltage		V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OH} = −400 μA		2.4	3.4		V
V_{OL}	Low-level output voltage		V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OL} = 16 mA			0.2	0.4	V
I_I	Input current at maximum input voltage		V_{CC} = MAX, V_I = 5.5 V				1	mA
I_{IH}	High-level input current	D input	V_{CC} = MAX, V_I = 2.4 V				80	μA
		C input					160	
I_{IL}	Low-level input current	D input	V_{CC} = MAX, V_I = 0.4 V				−3.2	mA
		C input					−6.4	
I_{OS}	Short-circuit output current§		V_{CC} = MAX	SN54'	−20		−57	mA
				SN74'	−18		−57	
I_{CC}	Supply current		V_{CC} = MAX, See Note 3	SN54'		32	46	mA
				SN74'		32	53	

†For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.
‡All typical values are at V_{CC} = 5 V, T_A = 25°C.
§Not more than one output should be shorted at a time.
NOTE 3: I_{CC} is tested with all inputs grounded and all outputs open.

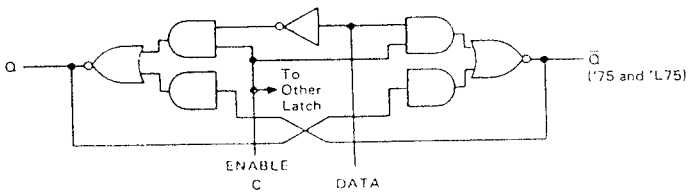
switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{PLH}	D	Q	C_L = 15 pF, R_L = 400 Ω, See Figure 1		16	30	ns
t_{PHL}					14	25	
t_{PLH}¶	D	$\bar{Q}$			24	40	ns
t_{PHL}¶					7	15	
t_{PLH}	C	Q			16	30	ns
t_{PHL}					7	15	
t_{PLH}¶	C	$\bar{Q}$			16	30	ns
t_{PHL}¶					7	15	

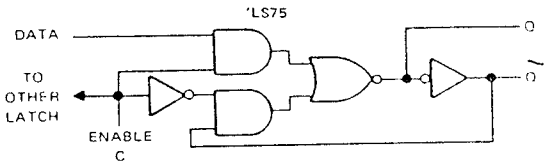
t_{PLH} = propagation delay time, low-to-high-level output
t_{PHL} = propagation delay time, high-to-low-level output
¶These parameters are not applicable for the SN5477.

logic diagrams (each latch)

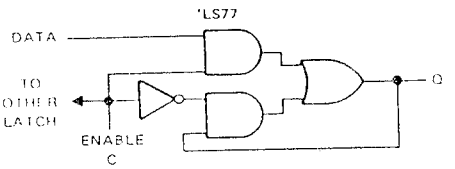
'75, '77, 'L75, 'L77



'LS75



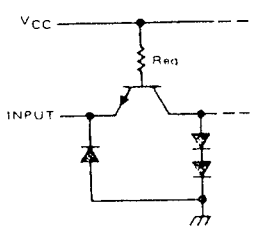
'LS77



schematics of inputs and outputs

'75, '77

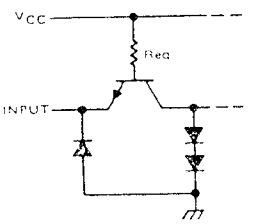
EQUIVALENT OF EACH INPUT



Data R_{eq} = 2 kΩ NOM
Enable R_{eq} = 1 kΩ NOM

'L75, 'L77

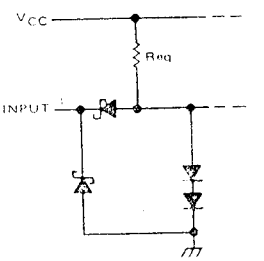
EQUIVALENT OF EACH INPUT



Data R_{eq} = 4 kΩ NOM
Enable R_{eq} = 2 kΩ NOM

'LS75, 'LS77

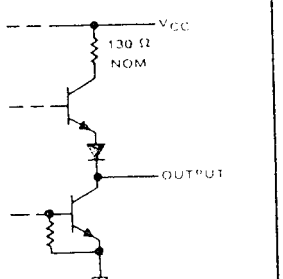
EQUIVALENT OF EACH INPUT



Data R_{eq} = 17 kΩ
Enable R_{eq} = 4.2 kΩ

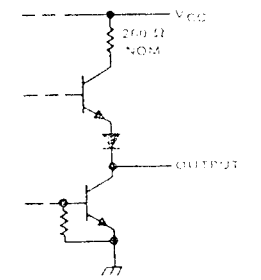
'75, '77

TYPICAL OF ALL OUTPUTS



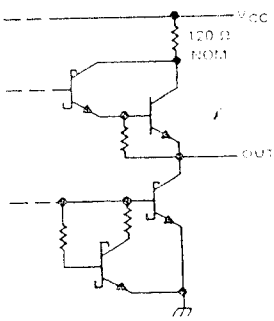
'L75, 'L77

TYPICAL OF ALL OUTPUTS



'LS75, 'LS77

TYPICAL OF ALL OUTPUTS



TEXAS INSTRUMENTS
POST OFFICE BOX 225012 • DALLAS, TEXAS 75265

National Semiconductor

DM54ALS244A/DM74ALS244A Octal TRI-STATE® Bus Driver

General Description

This octal TRI-STATE bus driver is designed to provide the designer with flexibility in implementing a bus interface with memory, microprocessor, or communication systems. The output TRI-STATE gating control is organized into two separate groups of four buffers, and both control inputs enable the respective outputs when set logic low. The TRI-STATE circuitry contains a feature that maintains the buffer outputs in TRI-STATE (high impedance state) during power supply ramp-up or ramp-down. This eliminates bus glitching problems that arise during power-up and power-down.

The -1 versions of the DM74ALS devices are identical to their standard versions except that the recommended maximum I_{OL} is increased to 48 mA. There are no -1 versions of the DM54ALS devices.

Features

- Advanced low power oxide-isolated ion-implanted Schottky TTL process
- Functional and pin compatible with the DM54/74LS counterpart
- Improved switching performance with less power dissipation compared with the DM54/74LS counterpart
- Switching response specified into 500Ω and 50 pF load
- Switching response specifications guaranteed over full temperature and V_{CC} supply range
- PNP input design reduces input loading
- Low level drive current:
  54ALS = 12 mA, 74ALS = 24 mA, 74ALS-1 = 48 mA

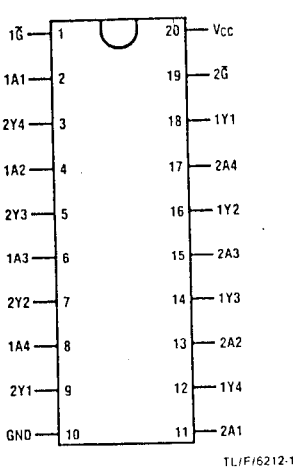
Absolute Maximum Ratings (Note 1)

Supply Voltage, V_{CC}	7V
Input Voltage	7V
Output Voltage	5.5V
Storage Temperature Range	−65°C to +150°C

Note 1: The "Absolute Maximum Ratings" are those values beyond which the safety of the device can not be guaranteed. The device should not be operated at these limits. The parametric values defined in the "Electrical Characteristics" table are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Connection Diagram

Dual-In-Line Package



TL/F/6212-1

TOP VIEW

DM54ALS244A (J) DM74ALS244A (J, N)

Function Table

Enable Input $1\bar{G}$ or $2\bar{G}$	Data Buffer Outputs
L	Active
H	TRI-STATE

Recommended Operating Conditions

Symbol	Parameter	DM54ALS244A			DM74ALS244A			Units
		Min	Typ	Max	Min	Typ	Max	
V_{CC}	Supply Voltage	4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High Level Input Voltage	2			2			V
V_{IL}	Low Level Input Voltage			0.8			0.8	V
I_{OH}	High Level Output Current			−12			−15	mA
I_{OL}	Low Level Output Current			12			24	mA
	DM74ALS244A-1			—			48	mA
T_A	Operating Free Air Temperature	−55		125	0		70	°C

Electrical Characteristics over recommended operating free air temperature range (unless otherwise specified)

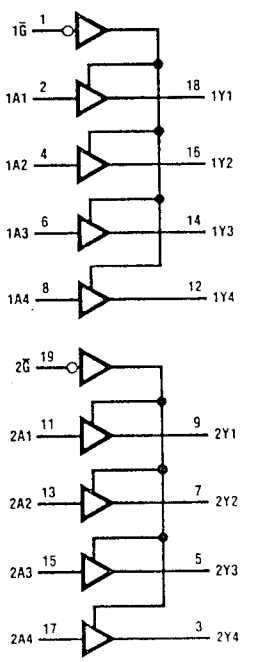
Symbol	Parameter	Conditions		DM54ALS244A			DM74ALS244A			Units
				Min	Typ	Max	Min	Typ	Max	
V_{IK}	Input Clamp Voltage	V_{CC} = 4.5V, I_I = −18 mA				−1.5			−1.5	V
V_{OH}	High Level Output	V_{CC} = 4.5V to 5.5V	I_{OH} = −0.4 mA	V_{CC} − 2			V_{CC} − 2			V
		V_{CC} = 4.5V	I_{OH} = −3 mA	2.4			2.4			V
			I_{OH} = Max	2			2			V
V_{OL}	Low Level Output Voltage	V_{CC} = 4.5V, I_{OL} = 54ALS (Max)			0.25	0.4		0.25	0.4	V
		I_{OL} = 74ALS (Max)			—	—		0.35	0.5	V
I_I	Input Current at Max Input Voltage	V_{CC} = 5.5V, V_I = 7V				0.1			0.1	mA
I_{IH}	High Level Input Current	V_{CC} = 5.5V, V_I = 2.7V				20			20	μA
I_{IL}	Low Level Input Current	V_{CC} = 5.5V, V_{IL} = 0.4V				−0.1			−0.1	mA
I_O	Output Drive Current	V_{CC} = 5.5V, V_O = 2.25V		−30		−112	−30		−112	mA
I_{OZH}	High Level TRI-STATE Output Current	V_{CC} = 5.5V, V_O = 2.7V				20			20	μA
I_{OZL}	Low Level TRI-STATE Output Current	V_{CC} = 5.5V, V_O = 0.4V				−20			−20	μA
I_{CC}	Supply Current	V_{CC} = 5.5V, Outputs High			9	15		9	15	mA
		Outputs Low			15	24		15	24	mA
		Outputs TRI-STATE			17	27		17	27	mA

Switching Characteristics over recommended operating free air temperature range (Note 1).

Parameter	From (Input)	To (Output)	Conditions	54ALS244A			74ALS244A			Units
				Min	Typ	Max	Min	Typ	Max	
t_{PLH}	A	Y	V_{CC} = 4.5V to 5.5V, C_L = 50 pF, R1 = 500Ω, R2 = 500Ω, T_A = Min to Max	3		13	3		10	ns
t_{PHL}				3		13	3		10	ns
t_{PZH}	$\overline{G}$	Y		7		25	7		20	ns
t_{PZL}				7		25	7		20	ns
t_{PHZ}	$\overline{G}$	Y		2		12	2		10	ns
t_{PLZ}				3		18	3		13	ns

Note 1: See Section 1 for test waveforms and output load.

Logic Diagram



TL/F/6212-2

National Semiconductor

DM54ALS245A/DM74ALS245A Octal TRI-STATE® Bus Transceivers

General Description

This advanced low power Schottky device contains 8 pairs of TRI-STATE logic elements configured as octal bus transceivers. These circuits are designed for use in memory, microprocessor systems and in asynchronous bidirectional data buses. Two way communication between buses is controlled by the (DIR) input. Data transmits either from the A bus to the B bus or from the B bus to the A bus. Both the driver and receiver outputs can be disabled via the ($\overline{G}$) enable input which causes outputs to enter the high impedance mode so that the buses are effectively isolated.

Features

- Advanced oxide-isolated, ion implanted Schottky TTL process
- Non-inverting logic output
- Glitch free bus during power up and down
- TRI-STATE outputs independently controlled on A and B buses
- Low output impedance to drive terminated transmission lines to 133Ω
- Switching response specified into 500Ω 50 pF
- Specified to interface with CMOS at $V_{OH} = V_{CC} - 2V$
- PNP inputs to reduce input loading
- Switching specifications guaranteed over full temperature and V_{CC} range

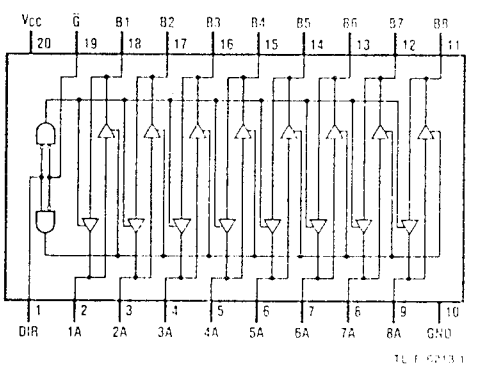
Absolute Maximum Ratings (Note 1)

Supply Voltage, V_{CC}	7V
Input Voltage	
Control Inputs	7V
I/O Ports	5.5V
Operating Free Air Temperature Range	
DM54ALS	−55°C to 125°C
DM74ALS	0°C to 70°C
Storage Temperature Range	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	300°C

Note 1: The "Absolute Maximum Ratings" are those values beyond which the safety of the device can not be guaranteed. The device should not be operated at these limits. The parametric values defined in the "Electrical Characteristics" table are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

Connection Diagram

Dual-In-Line Package



DM54ALS245A (J) DM74ALS245A (J, N)

Function Table

Control Inputs		Operation
$\overline{G}$	DIR	
L	L	B Data to A Bus
L	H	A Data to B Bus
H	X	HI-Z

Recommended Operating Conditions

Symbol	Parameter	DM54ALS245A			DM74ALS245A			Units
		Min	Typ	Max	Min	Typ	Max	
V_{CC}	Supply Voltage	4.5	5	5.5	4.5	5	5.5	V
V_{IH}	High Level Input Voltage	2			2			V
V_{IL}	Low Level Input Voltage			0.8			0.8	V
I_{OH}	High Level Output Current			−12			−15	mA
I_{OL}	Low Level Output Current			12			24	mA
	DM74ALS245A-1 Option Only						48	mA

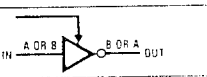
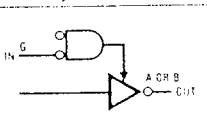
Electrical Characteristics over recommended operating free air temperature range

All typical values are measured at $V_{CC} = 5V$, $T_A = 25°C$.

Symbol	Parameter	Conditions		Min	Typ	Max	Units
V_{IK}	Input Clamp Voltage	$V_{CC} = 4.5V$, $I_{IN} = -18$ mA				−1.5	V
V_{OH}	High Level Output Voltage	$V_{CC} = 4.5V$, $I_{OH} = -3$ mA		2.4	3.2		V
		$V_{CC} = 4.5V$, $I_{OH} = $ Max		2	2.3		V
		$I_{OH} = -0.4$ mA, $V_{CC} = 4.5V$ to 5.5V		$V_{CC} - 2$			V
V_{OL}	Low Level Output Voltage	$V_{CC} = 4.5V$	54/74ALS $I_{OL} = 12$ mA		0.25	0.4	V
			74ALS $I_{OL} = 24$ mA		0.35	0.5	V
			74ALS-1 $I_{OL} = 48$ mA			0.5	V
I_I	Input Current at Max Input Voltage	$V_{CC} = 5.5V$	$V_{IN} = 7V$ Control Inputs			0.1	mA
			$V_{IN} = 5.5V$ A or B Ports			0.1	
I_{IH}	High Level Input Current	$V_{CC} = 5.5V$, $V_{IN} = 2.7V$				20	μA
I_{IL}	Low Level Input Current	$V_{CC} = 5.5V$, $V_{IN} = 0.4V$				−0.1	mA
I_O	Output Drive Current	$V_{CC} = 5.5V$, $V_{OUT} = 2.25V$		−30		−112	mA
I_{CC}	54ALS245A Supply Current	$V_{CC} = 5.5V$	Outputs High		30	48	mA
			Outputs Low		38	60	mA
			TRI-STATE		38	63	mA
I_{SC}	74ALS245A Supply Current	$V_{CC} = 5.5V$	Outputs High		30	45	mA
			Outputs Low		36	55	mA
			TRI-STATE		38	58	mA

Switching Characteristics over recommended operating free air temperature range (Notes 1 and 2)

All typical values are measured at $V_{CC} = 5V$, $T_A = 25°C$.

Symbol	Parameter (Propagation Delay Time)	Circuit Configuration	DM54ALS245A			DM74ALS245A			Units
			Min	Typ	Max	Min	Typ	Max	
t_{PLH}	High-to-Low Level Output		3		15	3		10	ns
t_{PHL}	High-to-Low Level Output		3		13	3		10	ns
t_{PZL}	Output Enable to Low Level		5		25	5		20	ns
t_{PZH}	Output Enable to High Level		5		25	5		20	ns
t_{PLZ}	Output Disable from Low Level		4		18	4		15	ns
t_{PHZ}	Output Disable from High Level		2		12	2		10	ns

Note 1: See Section 1 for test waveforms and output load.

Note 2: Switching characteristic conditions are $V_{CC} = 4.5V$ to $5.5V$, $R_L = 500\Omega$, $C_L = 50$ pF.

2732A
32K (4K x 8) PRODUCTION AND UV ERASABLE PROMS

- 200 ns (2732A-2) Maximum Access Time ... HMOS*-E Technology
- Compatible with High-Speed Microcontrollers and Microprocessors ... Zero WAIT State
- Two Line Control
- 10% V_{CC} Tolerance Available
- Low Current Requirement
  - −100 mA Active
  - −35 mA Standby
- Int$_e$ligent Identifier™ Mode
  - −Automatic Programming Operation
- Industry Standard Pinout ... JEDEC Approved 24 Pin Ceramic and Plastic Package

  (See Packaging Spec. Order # 221369)

The Intel 2732A is a 5V-only, 32,768-bit ultraviolet erasable (cerdip) Electrically Programmable Read-Only Memory (EPROM). The standard 2732A access time is 250 ns with speed selection (2732A-2) available at 200 ns. The access time is compatible with high performance microprocessors such as the 8 MHz iAPX 186. In these systems, the 2732A allows the microprocessor to operate without the addition of WAIT states.

The 2732A is currently available in two different package types. Cerdip packages provide flexibility in prototyping and R & D environments where reprogrammability is required. Plastic DIP EPROMs provide optimum cost effectiveness in production environments. Inventoried in the unprogrammed state, the P2732A is programmed quickly and efficiently when the need to change code arises. Costs incurred for new ROM masks or obsoleted ROM inventories are avoided. The tight package dimensional controls, inherent non-erasability, and high reliability of the P2732A make it the ideal component for these production applications.

An important 2732A feature is Output Enable ($\overline{OE}$) which is separate from the Chip Enable ($\overline{CE}$) control. The $\overline{OE}$ control eliminates bus contention in microprocessor systems. The $\overline{CE}$ is used by the 2732A to place it in a standby mode ($\overline{CE}$ = V_{IH}) which reduces power consumption without increasing access time. The standby mode reduces the current requirement by 65%; the maximum active current is reduced from 100 mA to a standby current of 35 mA.

*HMOS is a patented process of Intel Corporation.

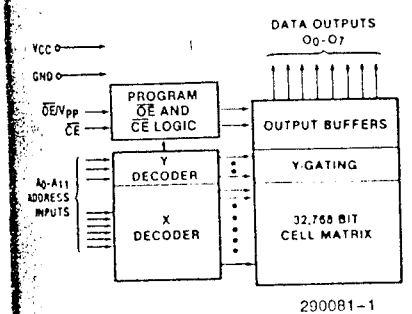


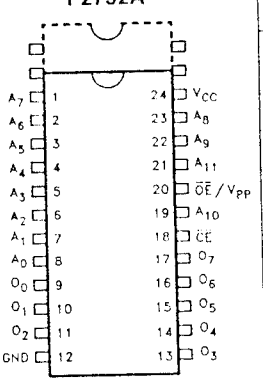
Figure 1. Block Diagram

Pin Names

A_0-A_{11}	Addresses
$\overline{CE}$	Chip Enable
$\overline{OE}$/V_{PP}	Output Enable/V_{PP}
O_0-O_7	Outputs

27128 27128A	2764 2764A 27C64 87C64	2716
V_{PP}	V_{PP}	
A_{12}	A_{12}	
A_7	A_7	A_7
A_6	A_6	A_6
A_5	A_5	A_5
A_4	A_4	A_4
A_3	A_3	A_3
A_2	A_2	A_2
A_1	A_1	A_1
A_0	A_0	A_0
O_0	O_0	O_0
O_1	O_1	O_1
O_2	O_2	O_2
GND	GND	GND

2732A
P2732A



2716	2764 2764A 27C64 87C64	27128 27128A
	V_{CC} $\overline{PGM}$	V_{CC} $\overline{PGM}$
V_{CC}	N.C.	A_{13}
A_8	A_8	A_8
A_9	A_9	A_9
V_{PP}	A_{11}	A_{11}
$\overline{OE}$	$\overline{OE}$	$\overline{OE}$
A_{10}	A_{10}	A_{10}
$\overline{CE}$	$\overline{CE}$ ALE/CE	$\overline{CE}$
O_7	O_7	O_7
O_6	O_6	O_6
O_5	O_5	O_5
O_4	O_4	O_4
O_3	O_3	O_3

NOTE:
Intel "Universal Site" compatible EPROM configurations are shown in the blocks adjacent to the 2732A pins.

Figure 2. Cerdip/Plastic DIP Pin Configuration

Order Number: 290081-002

EXTENDED TEMPERATURE (EXPRESS) EPROMs

The Intel EXPRESS EPROM family is a series of electrically programmable read only memories which have received additional processing to enhance product characteristics. EXPRESS processing is available for several densities of EPROM, allowing the choice of appropriate memory size to match system applications. EXPRESS EPROM products are available with 168 ±8 hour, 125°C dynamic burn-in using Intel's standard bias configuration. This process exceeds or meets most industry specifications of burn-in. The standard EXPRESS EPROM operating temperature range is 0°C to 70°C. Extended operating temperature range (−40°C to +85°C) EXPRESS products are available. Like all Intel EPROMs, the EXPRESS EPROM family is inspected to 0.1% electrical AQL. This may allow the user to reduce or eliminate incoming inspection testing.

READ OPERATION

D.C. CHARACTERISTICS

Electrical Parameters of EXPRESS EPROM products are identical to standard EPROM parameters except for:

Symbol	Parameter	TD2732A LD2732A		Test Conditions
		Min	Max	
I_{SB}	V_{CC} Standby Current (mA)		45	$\overline{CE} = V_{IH}$, $\overline{OE} = V_{IL}$
I_{CC1}(1)	V_{CC} Active Current (mA)		150	$\overline{OE} = \overline{CE} = V_{IL}$
	V_{CC} Active Current at High Temperature (mA)		125	$\overline{OE} = \overline{CE} = V_{IL}$, $V_{PP} = V_{CC}$, $T_{Ambient} = 85°C$

NOTE:
1. Maximum current value is with outputs O_0 to O_7 unloaded.

EXPRESS EPROM PRODUCT FAMILY

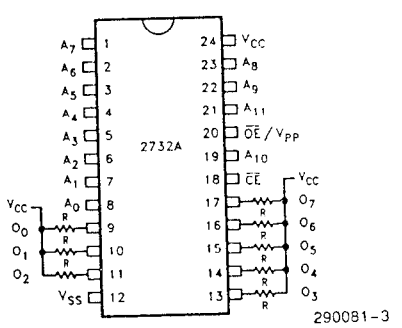
PRODUCT DEFINITONS

Type	Operating Temperature	Burn-In 125°C (hr)
Q	0°C to +70°C	168 ±8
T	−40°C to +85°C	None
L	−40°C to +85°C	168 ±8

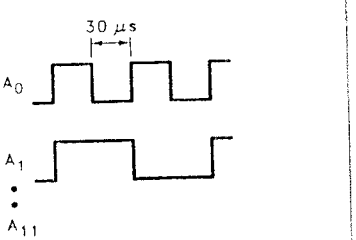
EXPRESS OPTIONS

2732A Versions

Speed Versions	Packaging Options	
	Cerdip	Plastic
−2	Q	
STD	Q, T, L	
−3	Q	
−4	Q, T, L	
−20	Q	
−25	Q, T, L	
−30	Q	
−45	Q, T, L	



$\overline{OE}/V_{PP}$ = +5V, R = 1KΩ, V_{CC} = +5V
V_{SS} = GND, $\overline{CE}$ = GND



Binary Sequence from A_0 to A_{11}

Burn-In Bias and Timing Diagrams

ABSOLUTE MAXIMUM RATINGS*

Operating Temp. During Read . . . . . 0°C to +70°C
Temperature Under Bias . . . . . −10°C to +80°C
Storage Temperature . . . . . −65°C to +125°C
All Input or Output Voltages with Respect to Ground . . . . . −0.3V to +6V
Voltage on A9 with Respect to Ground . . . . . −0.3V to +13.5V
V_{PP} Supply Voltage with Respect to Ground During Programming . . . . . −0.3V to +22V
V_{CC} Supply Voltage with Respect to Ground . . . . . −0.3V to +7.0V

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

READ OPERATION

D.C. CHARACTERISTICS $0°C \le T_A \le +70°C$

Symbol	Parameter	Limits			Units	Conditions
		Min	Typ(3)	Max		
I_{LI}	Input Load Current			10	μA	$V_{IN} = 5.5V$
I_{LO}	Output Leakage Current			10	μA	$V_{OUT} = 5.5V$
I_{SB}(2)	V_{CC} Current (Standby)			35	mA	$\overline{CE} = V_{IH}$, $\overline{OE} = V_{IL}$
I_{CC1}(2)	V_{CC} Current (Active)			100	mA	$\overline{OE} = \overline{CE} = V_{IL}$
V_{IL}	Input Low Voltage	−0.1		0.8	V	
V_{IH}	Input High Voltage	2.0		$V_{CC}+1$	V	
V_{OL}	Output Low Voltage			0.45	V	$I_{OL} = 2.1$ mA
V_{OH}	Output High Voltage	2.4			V	$I_{OH} = -400$ μA

A.C. CHARACTERISTICS $0°C \le T_A \le 70°C$

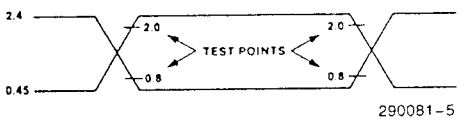
Versions	$V_{CC} \pm 5\%$	2732A-2 P2732A-2		2732A P2732A		2732A-3 P2732A-3		2732A-4 P2732A-4		Units	Test Conditions
	$V_{CC} \pm 10\%$	2732A-20		2732A-25		2732A-30		2732A-45			
Symbol	Parameter	Min	Max	Min	Max	Min	Max	Min	Max		
t_{ACC}	Address to Output Delay		200		250		300		450	ns	$\overline{CE} = \overline{OE} = V_{IL}$
t_{CE}	$\overline{CE}$ to Output Delay		200		250		300		450	ns	$\overline{OE} = V_{IL}$
t_{OE}	$\overline{OE}/V_{PP}$ to Output Delay		70		100		150		150	ns	$\overline{CE} = V_{IL}$
t_{DF}(4)	$\overline{OE}/V_{PP}$ High to Output Float	0	60	0	60	0	130	0	130	ns	$\overline{CE} = V_{IL}$
t_{OH}	Output Hold from Addresses, $\overline{CE}$ or $\overline{OE}/V_{PP}$, Whichever Occurred First	0		0		0		0		ns	$\overline{CE} = \overline{OE} = V_{IL}$

NOTES:
1. V_{CC} must be applied simultaneously or before $\overline{OE}/V_{PP}$ and removed simultaneously or after $\overline{OE}/V_{PP}$.
2. The maximum current value is with outputs O_0 to O_7 unloaded.
3. Typical values are for $T_A = 25°C$ and nominal supply voltages.
4. This parameter is only sampled and is not 100% tested. Output Float is defined as the point where data is no longer driven—see timing diagram.

CAPACITANCE [2] T_A = 25°C, f = 1 MHz

Symbol	Parameter	Typ	Max	Unit	Conditions
C_{IN1}	Input Capacitance Except $\overline{OE}/V_{PP}$	4	6	pF	V_{IN} = 0V
C_{IN2}	$\overline{OE}/V_{PP}$ Input Capacitance		20	pF	V_{IN} = 0V
C_{OUT}	Output Capacitance	8	12	pF	V_{OUT} = 0V

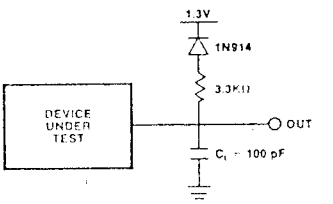
A.C. TESTING INPUT/OUTPUT WAVEFORM



290081-5

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

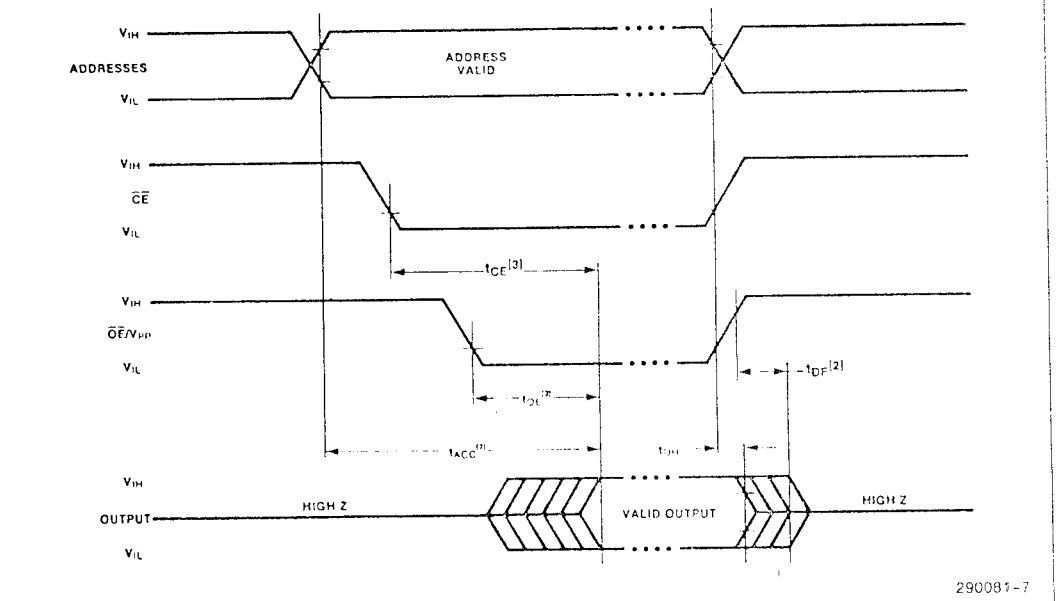
A.C. TESTING LOAD CIRCUIT



290081-6

C_L = 100 pF
C_L Includes Jig Capacitance

A.C. WAVEFORMS



290081-7

NOTES:
1. Typical values are for T_A = 25°C and nominal supply voltages.
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 timing diagram.
3. $\overline{OE}/V_{PP}$ may be delayed up to $t_{ACC} - t_{OE}$ after the falling edge of $\overline{CE}$ without impacting t_{CE}.

PLASTIC EPROM APPLICATIONS

Intel's P2732A is the result of a multi-year effort to make EPROMs more cost effective for production applications. The benefits of a plastic package enable the P2732A to be used for high volume production with lower profile boards and easier production assembly (no cover over UV transparent windows).

The reliability of plastic EPROMs is equivalent to traditional CERDIP packaging. The plastic is rugged and durable making it optimal for auto insertion and auto handling equipment. Design and testing ensures device programmability, data integrity, and impermeability to moisture.

Intel's Plastic EPROMs are designed for total compatibility with their CERDIP packaged predecessors. This encompasses quality, reliability, and programming. All Intel Plastic EPROMs have passed Intel's strict process and product reliability qualifications.

DEVICE OPERATION

The modes of operation of the 2732A are listed in Table 1. A single 5V power supply is required in the read mode. All inputs are TTL levels except for $\overline{OE}/V_{PP}$ during programming and 12V on A_9 for the int$_e$ligent Identifier™ mode. In the program mode the $\overline{OE}/V_{PP}$ input is pulsed from a TTL level to 21V.

Table 1. Mode Selection

Pins / Mode	$\overline{CE}$	$\overline{OE}/V_{PP}$	A_9	A_0	V_{CC}	Outputs
Read/Program Verify	V_{IL}	V_{IL}	X	X	V_{CC}	D_{OUT}
Output Disable	V_{IL}	V_{IH}	X	X	V_{CC}	High Z
Standby	V_{IH}	X	X	X	V_{CC}	High Z
Program	V_{IL}	V_{PP}	X	X	V_{CC}	D_{IN}
Program Inhibit	V_{IH}	V_{PP}	X	X	V_{CC}	High Z
Int$_e$ligent Identifier(3)						
—Manufacturer	V_{IL}	V_{IL}	V_H	V_{IL}	V_{CC}	89H
—Device	V_{IL}	V_{IL}	V_H	V_{IH}	V_{CC}	01H

NOTES:
1. X can be V_{IH} or V_{IL}.
2. V_H = 12V ±0.5V.
3. A_1–A_8, A_{10}, A_{11} = V_{IL}.

Read Mode

The 2732A has two control functions, both of which must be logically active in order to obtain data at the outputs. Chip Enable ($\overline{CE}$) is the power control and should be used for device selection. Output Enable ($\overline{OE}/V_{PP}$) is the output control and should be used to gate data from the output pins, independent of device selection. Assuming that addresses are stable, address access time (t_{ACC}) is equal to the delay from $\overline{CE}$ to output (t_{CE}). Data is available at the outputs after the falling edge of $\overline{OE}/V_{PP}$, assuming that $\overline{CE}$ has been low and addresses have been stable for at least $t_{ACC}-t_{OE}$.

Standby Mode

EPROMs can be placed in a standby mode which reduces the maximum active current of the device by applying a TTL-high signal to the $\overline{CE}$ input. When in standby mode, the outputs are in a high impedance state, independent of the $\overline{OE}/V_{PP}$ input.

Two Line Output Control

Because EPROMs are usually used in larger memory arrays, Intel has provided two control lines which accommodate this multiple memory connection. The two control lines allow for:

a) The lowest possible memory power dissipation, and

b) complete assurance that output bus contention will not occur.

To use these two control lines most efficiently, $\overline{CE}$ should be decoded and used as the primary device selecting function, while $\overline{OE}/V_{PP}$ should be made a common connection to all devices in the array and connected to the $\overline{READ}$ line from the system control bus. This assures that all deselected memory devices are in their low power standby mode and that the output pins are active only when data is desired from a particular memory device.

SYSTEM CONSIDERATION

The power switching characteristics of EPROMs require careful decoupling of the devices. The supply current, I_{CC}, has three segments that are of interest to the system designer—the standby current level, the active current level, and the transient current peaks that are produced by the falling and rising edges of Chip Enable. The magnitude of these transient current peaks is dependent on the output capacitive and inductive loading of the device. The associated transient voltage peaks can be suppressed by complying with Intel's two-line control and by use of properly selected decoupling capacitors. It is recommended that a 0.1 μF ceramic capacitor be used on every device between V_{CC} and GND. This should be a high frequency capacitor of low inherent inductance and should be placed as close to the device as possible. In addition, a 4.7 μF bulk electrolytic capacitor should be used between V_{CC} and GND for

every eight devices. The bulk capacitor should be located near where the power supply is connected to the array. The purpose of the bulk capacitor is to overcome the voltage droop caused by the inductive effects of PC board traces.

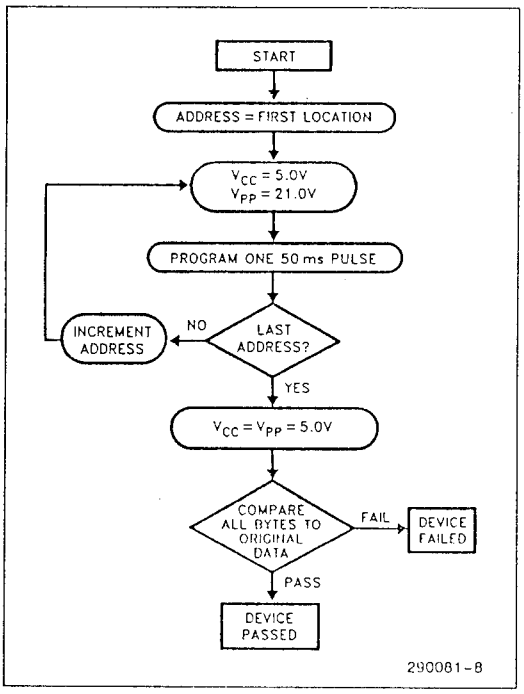


Figure 3. Standard Programming Flowchart

PROGRAMMING MODES

CAUTION: Exceeding 22V on $\overline{OE}/V_{PP}$ will permanently damage the device.

Initially, and after each erasure (cerdip EPROMs), all bits of the EPROM are in the "1" state. Data is introduced by selectively programming "0s" into the bit locations. Although only "0s" will be programmed, both "1s" and "0s" can be present in the data word. The only way to change a "0" to a "1" in cerdip EPROMs is by ultraviolet light erasure.

The device is in the programming mode when the $\overline{OE}/V_{PP}$ input is at 21V. It is required that a 0.1 μF capacitor be placed across $\overline{OE}/V_{PP}$ and ground to suppress spurious voltage transients which may damage the device. The data to be programmed is applied 8 bits in parallel to the data output pins. The levels required for the address and data inputs are TTL.

When the address and data are stable, a 20 ms 50 ms typical) active low, TTL program pulse is applied to the $\overline{CE}$ input. A program pulse must be applied at each address location to be programmed (see Figure 3). Any location can be programmed at any time—either individually, sequentially, or at random. The program pulse has a maximum width of 55 ms. The EPROM must not be programmed with a DC signal applied to the $\overline{CE}$ input.

Programming of multiple 2732As in parallel with the same data can be easily accomplished due to the simplicity of the programming requirements. Like inputs of the paralleled 2732As may be connected together when they are programmed with the same data. A low level TTL pulse applied to the $\overline{CE}$ input programs the paralleled 2732As.

Program Inhibit

Programming of multiple EPROMs in parallel with different data is easily accomplished by using the Program Inhibit mode. A high level $\overline{CE}$ input inhibits the other EPROMs from being programmed. Except for $\overline{CE}$, all like inputs (including $\overline{OE}/V_{PP}$) of the parallel EPROMs may be common. A TTL low level pulse applied to the $\overline{CE}$ input with $\overline{OE}/V_{PP}$ at 21V will program that selected device.

Program Verify

A verify (Read) should be performed on the programmed bits to determine that they have been correctly programmed. The verify is performed with $\overline{OE}/V_{PP}$ and $\overline{CE}$ at V_{IL}. Data should be verified t_{DV} after the falling edge of $\overline{CE}$.

int$_e$ligent Identifier™ Mode

The int$_e$ligent Identifier Mode allows the reading out of a binary code from an EPROM that will identify its manufacturer and type. This mode is intended for use by programming equipment for the purpose of automatically matching the device to be programmed with its corresponding programming algorithm. This mode is functional in the 25°C ±5°C ambient temperature range that is required when programming the device.

To activate this mode, the programming equipment must force 11.5V to 12.5V on address line A9 of the EPROM. Two identifier bytes may then be sequenced from the device outputs by toggling address line A0 from V_{IL} to V_{IH}. All other address lines must be held at V_{IL} during the int$_e$ligent Identifier Mode.

Byte 0 (A0 = V_{IL}) represents the manufacturer code and byte 1 (A0 = V_{IH}) the device identifier code. These two identifier bytes are given in Table 1.

INTEL EPROM PROGRAMMING SUPPORT TOOLS

Intel offers a full line of EPROM Programmers providing state-of-the-art programming for all Intel programmable devices. The modular architecture of Intel's EPROM programmers allows you to add new support as it becomes available, with very low cost add-ons. For example, even the earliest users of the iUP-FAST 27/K module may take advantage of Intel's new Quick-Pulse Programming™ Algorithm, the fastest in the industry.

Intel EPROM programmers may be controlled from a host computer using Intel's PROM Programming software (iPPS). iPPS makes programming easy for a growing list of industry standard hosts, including the IBM PC, XT, AT, and PCDOS compatibles, Intellec Development Systems, Intel's iPDS Personal Development System, and the Intel Network Development System (iNDS-II). Stand-alone operation is also available, including device previewing, editing, programming, and download of programming data from any source over an RS232C port.

For further details consult the EPROM Programming section of the Development Systems Handbook.

ERASURE CHARACTERISTICS (FOR CERDIP EPROMS)

The erasure characteristics are such that erasure begins to occur upon exposure to light with wavelengths shorter than aproximately 4000 Angstroms (Å). It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000–4000Å range. Data shows that constant exposure to room level fluorescent lighting could erase the EPROM in approximately 3 years, while it would take approximately 1 week to cause erasure when exposed to direct sunlight. If the device is to be exposed to these types of lighting conditions for extended periods of time, opaque labels should be placed over the window to prevent unintentional erasure.

The recommended erasure procedure is exposure to shortwave ultraviolet light which has a wavelength of 2537 Angstroms (Å). The integrated dose (i.e., UV intensity × exposure time) for erasure should be a minimum of 15 Wsec/cm^2. The erasure time with this dosage is approximately 15 to 20 minutes using an ultraviolet lamp with a 12000 μW/cm^2 power rating. The EPROM should be placed within 1 inch of the lamp tubes during erasure. The maximum integrated dose an EPROM can be exposed to without damage is 7258 Wsec/cm^2 (1 week @ 12000 μW/cm^2). Exposure of the device to high intensity UV light for longer periods may cause permanent damage.

PROGRAMMING

D.C. PROGRAMMING CHARACTERISTICS

T_A = 25°C ±5°C, V_{CC} = 5V ±5%, V_{PP} = 21V ±0.5V

Symbol	Parameter	Limits			Units	Test Conditions (Note 1)
		Min	Typ(3)	Max		
I_{LI}	Input Current (All Inputs)			10	µA	$V_{IN} = V_{IL}$ or V_{IH}
V_{IL}	Input Low Level (All Inputs)	−0.1		0.8	V	
V_{IH}	Input High Level (All Inputs Except $\overline{OE}/V_{PP}$)	2.0		V_{CC} + 1	V	
V_{OL}	Output Low Voltage During Verify			0.45	V	I_{OL} = 2.1 mA
V_{OH}	Output High Voltage During Verify	2.4			V	I_{OH} = −400 µA
I_{CC2}(4)	V_{CC} Supply Current (Program and Verify)		85	100	mA	
I_{PP2}(4)	V_{PP} Supply Current (Program)			30	mA	$\overline{CE} = V_{IL}$, $\overline{OE}/V_{PP} = V_{PP}$
V_{ID}	A9 intelligent Identifier Voltage	11.5		12.5	V	

A.C. PROGRAMMING CHARACTERISTICS

T_A = 25°C ±5°C, V_{CC} = 5V ±5%, V_{PP} = 21V ±0.5V

Symbol	Parameter	Limits			Units	Test Conditions* (Note 1)
		Min	Typ(3)	Max		
t_{AS}	Address Setup Time	2			µs	
t_{OES}	$\overline{OE}/V_{PP}$ 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}	$\overline{OE}/V_{PP}$ High to Output Not Driven	0		130	ns	(Note 2)
t_{PW}	$\overline{CE}$ Pulse Width During Programming	20	50	55	ms	
t_{OEH}	$\overline{OE}/V_{PP}$ Hold Time	2			µs	
t_{DV}	Data Valid from $\overline{CE}$			1	µs	$\overline{CE} = V_{IL}$, $\overline{OE}/V_{PP} = V_{IL}$
t_{VR}	V_{PP} Recovery Time	2			µs	
t_{PRT}	$\overline{OE}/V_{PP}$ Pulse Rise Time During Programming	50			ns	

NOTES:
1. V_{CC} must be applied simultaneously or before $\overline{OE}/V_{PP}$ and removed simultaneously or after $\overline{OE}/V_{PP}$.
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 timing diagram.
3. Typical values are for T_A = 25°C and nominal supply voltages.
4. The maximum current value is with outputs O_0 to O_7 unloaded.

*A.C. TEST CONDITIONS

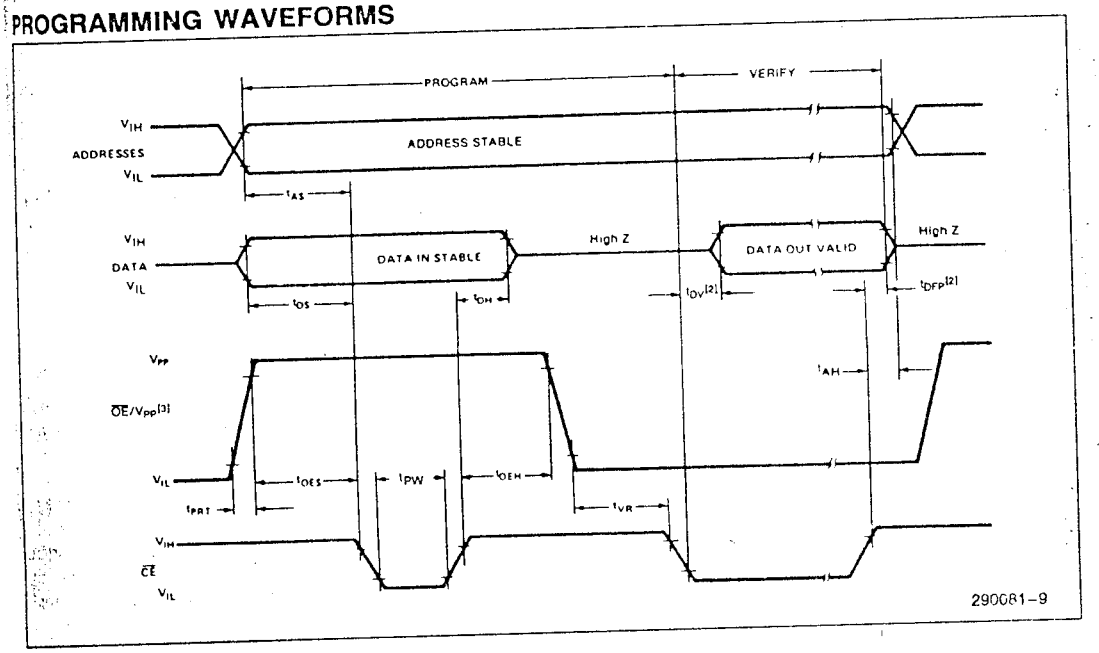
Input Rise and Fall Time (10% to 90%) ..... ≤20 ns

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

PROGRAMMING WAVEFORMS



NOTES:
1. The input timing reference level is 0.8V for a V_{IL} and 2V for a V_{IH}.
2. t_{DV} and t_{DFP} are characteristics of the device but must be accommodated by the programmer.
3. When programming the 2732A, a 0.1μF capacitor is required across $\overline{OE}/V_{PP}$ and ground to suppress spurious voltage transients which can damage the device.

8-CHANNEL ANALOGUE MULTIPLEXER/DEMULTIPLEXER

The HEF4051B is an 8-channel analogue multiplexer/demultiplexer with three address inputs (A_0 to A_2), an active LOW enable input ($\overline{E}$), eight independent inputs/outputs (Y_0 to Y_7) and a common input/output (Z).

The device contains eight bidirectional analogue switches, each with one side connected to an independent input/output (Y_0 to Y_7) and the other side connected to a common input/output (Z).

With $\overline{E}$ LOW, one of the eight switches is selected (low impedance ON-state) by A_0 to A_2. With $\overline{E}$ HIGH, all switches are in the high impedance OFF-state, independent of A_0 to A_2.

V_{DD} and V_{SS} are the supply voltage connections for the digital control inputs (A_0 to A_2, and $\overline{E}$). The V_{DD} to V_{SS} range is 3 to 15 V. The analogue inputs/outputs (Y_0 to Y_7, and Z) can swing between V_{DD} as a positive limit and V_{EE} as a negative limit. $V_{DD}-V_{EE}$ may not exceed 15 V.

For operation as a digital multiplexer/demultiplexer, V_{EE} is connected to V_{SS} (typically ground).

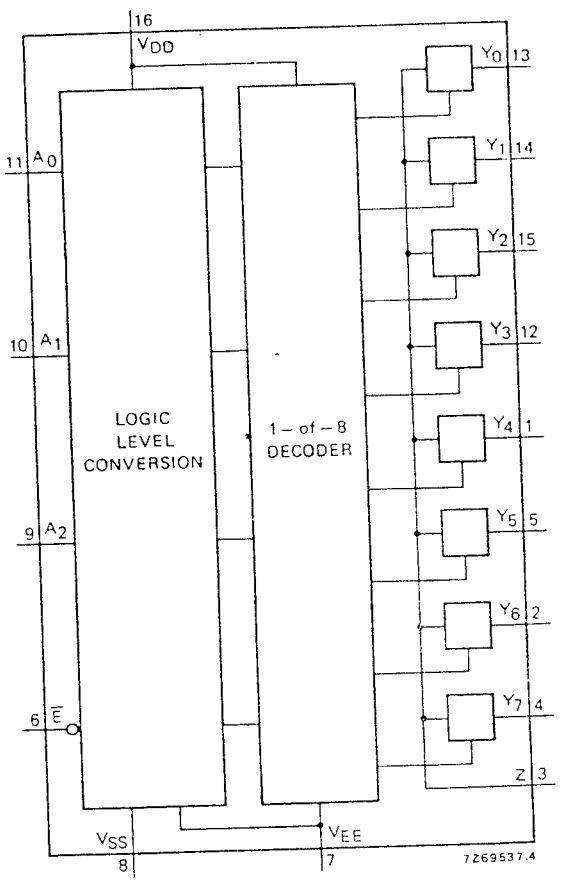


Fig. 1 Functional diagram.

FAMILY DATA

I_{DD} LIMITS category MSI

see Family Specifications

Products approved to CECC 90 104-039.

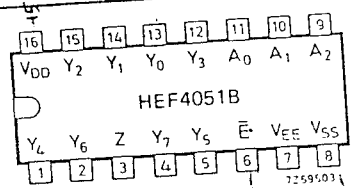


Fig. 2 Pinning diagram.

HEF4051BP : 16-lead DIL; plastic (SOT-38Z).
HEF4051BD: 16-lead DIL; ceramic (cerdip) (SOT-74).
HEF4051BT : 16-lead mini-pack; plastic (SO-16; SOT-109A).

PINNING

Y_0 to Y_7	independent inputs/outputs
A_0 to A_2	address inputs
$\bar{E}$	enable input (active LOW)
Z	common input/output

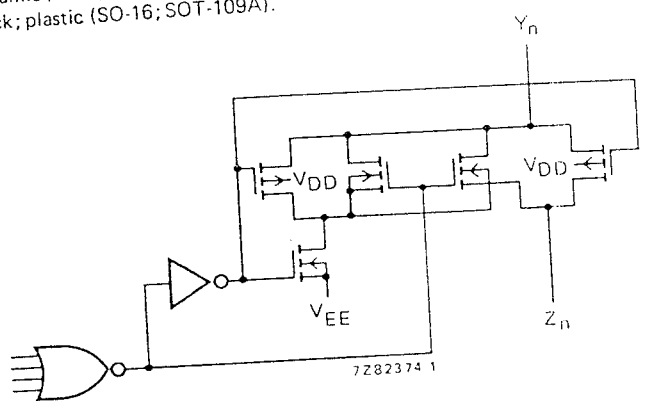


Fig. 3 Schematic diagram (one switch).

FUNCTION TABLE

inputs				channel ON
$\bar{E}$	A_2	A_1	A_0	
L	L	L	L	Y_0–Z
L	L	L	H	Y_1–Z
L	L	H	L	Y_2–Z
L	L	H	H	Y_3–Z
L	H	L	L	Y_4–Z
L	H	L	H	Y_5–Z
L	H	H	L	Y_6–Z
L	H	H	H	Y_7–Z
H	X	X	X	none

H = HIGH state (the more positive voltage)
L = LOW state (the less positive voltage)
X = state is immaterial

RATINGS

Limiting values in accordance with the Absolute Maximum System (IEC 134)

Supply voltage (with reference to V_{DD}) V_{EE} −18 to + 0,5 V

NOTE

To avoid drawing V_{DD} current out of terminal Z, when switch current flows into terminals Y, the voltage drop across the bidirectional switch must not exceed 0,4 V. If the switch current flows into terminal Z, no V_{DD} current will flow out of terminals Y, in this case there is no limit for the voltage drop across the switch, but the voltages at Y and Z may not exceed V_{DD} or V_{EE}.

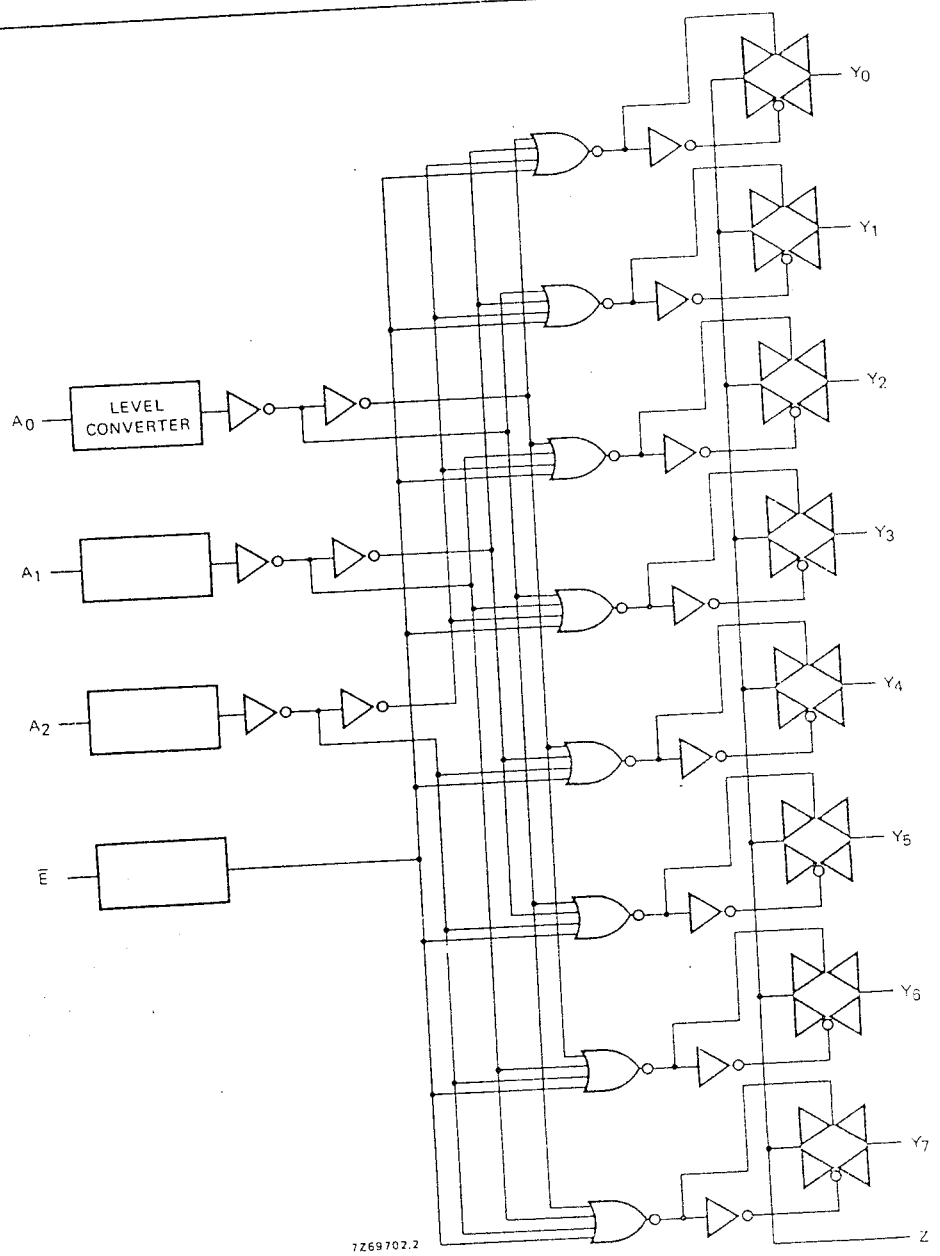


Fig. 4 Logic diagram.

D.C. CHARACTERISTICS

T_{amb} = 25 °C

	$V_{DD}-V_{EE}$ V	symbol	typ.	max.		conditions
ON resistance	5	R_{ON}	350	2500	Ω	V_{is} = 0 to $V_{DD}-V_{EE}$ see Fig. 6
	10		80	245	Ω	
	15		60	175	Ω	
ON resistance	5	R_{ON}	115	340	Ω	V_{is} = 0 see Fig. 6
	10		50	160	Ω	
	15		40	115	Ω	
ON resistance	5	R_{ON}	120	365	Ω	V_{is} = $V_{DD}-V_{EE}$ see Fig. 6
	10		65	200	Ω	
	15		50	155	Ω	
'Δ' ON resistance between any two channels	5	ΔR_{ON}	25	–	Ω	V_{is} = 0 to $V_{DD}-V_{EE}$ see Fig. 6
	10		10	–	Ω	
	15		5	–	Ω	
OFF-state leakage current, all channels OFF	5	I_{OZZ}	–	–	nA	$\overline{E}$ at V_{DD} V_{SS} = V_{EE}
	10		–	–	nA	
	15		–	1000	nA	
OFF-state leakage current, any channel	5	I_{OZY}	–	–	nA	$\overline{E}$ at V_{SS} V_{SS} = V_{EE}
	10		–	–	nA	
	15		–	200	nA	

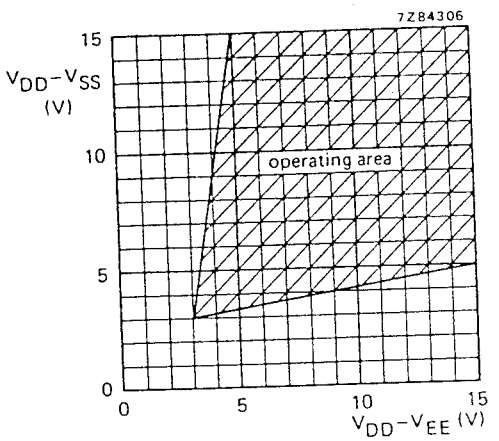


Fig. 5 Operating area as a function of the supply voltages.

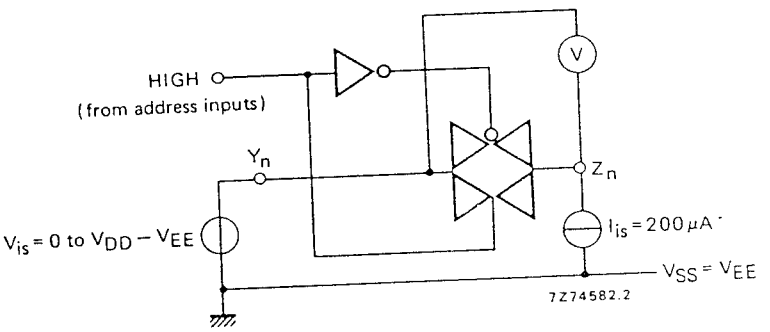


Fig. 6 Test set-up for measuring R_{ON}.

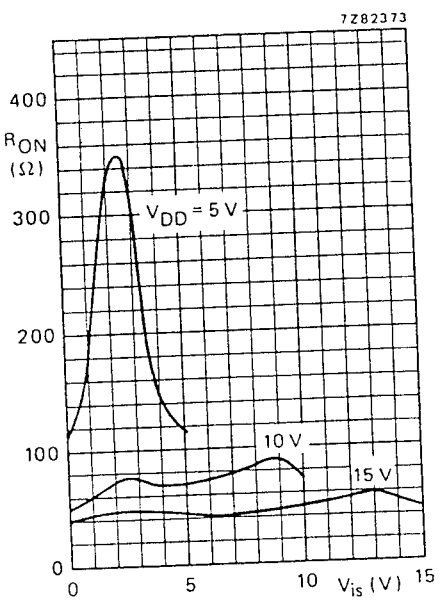


Fig. 7 Typical R_{ON} as a function of input voltage.
I_{is} = 200 μA
$V_{SS} = V_{EE}$ = 0 V

A.C. CHARACTERISTICS

$V_{EE} = V_{SS} = 0$ V; $T_{amb} = 25$ °C; input transition times ⩽ 20 ns

	V_{DD} V	typical formula for P (μW)	where
Dynamic power dissipation per package (P)	5 10 15	$1\,000\ f_i + \Sigma(f_o C_L) \times V_{DD}^2$ $5\,500\ f_i + \Sigma(f_o C_L) \times V_{DD}^2$ $15\,000\ f_i + \Sigma(f_o C_L) \times V_{DD}^2$	f_i = input freq. (MHz) f_o = output freq. (MHz) C_L = load capacitance (pF) $\Sigma(f_o C_L)$ = sum of outputs V_{DD} = supply voltage (V)

A.C. CHARACTERISTICS

$V_{EE} = V_{SS} = 0$ V; $T_{amb} = 25$ °C; input transition times ⩽ 20 ns

	V_{DD} V	symbol	typ.	max.		
Propagation delays $V_{is} \rightarrow V_{os}$ HIGH to LOW	5	t_{PHL}	15	30	ns	note 1
	10		5	10	ns	
	15		5	10	ns	
LOW to HIGH	5	t_{PLH}	15	30	ns	note 1
	10		5	10	ns	
	15		5	10	ns	
$A_n \rightarrow V_{os}$ HIGH to LOW	5	t_{PHL}	150	300	ns	note 2
	10		60	120	ns	
	15		45	90	ns	
LOW to HIGH	5	t_{PLH}	150	300	ns	note 2
	10		65	130	ns	
	15		45	90	ns	
Output disable times $\overline{E} \rightarrow V_{os}$ HIGH	5	t_{PHZ}	120	240	ns	note 3
	10		90	180	ns	
	15		85	170	ns	
LOW	5	t_{PLZ}	145	290	ns	note 3
	10		120	240	ns	
	15		115	230	ns	
Output enable times $\overline{E} \rightarrow V_{os}$ HIGH	5	t_{PZH}	140	280	ns	note 3
	10		55	110	ns	
	15		40	80	ns	
LOW	5	t_{PZL}	140	280	ns	note 3
	10		55	110	ns	
	15		40	80	ns	

A.C. CHARACTERISTICS

$V_{EE} = V_{SS} = 0$ V; $T_{amb} = 25$ °C; input transition times ⩽ 20 ns

	V_{DD} V	symbol	typ.	max.		
Distortion, sine-wave response	5		0,25		%	note 4
	10		0,04		%	
	15		0,04		%	
Crosstalk between any two channels	5		–		MHz	note 5
	10		1		MHz	
	15		–		MHz	
Crosstalk; enable or address input to output	5		–		mV	note 6
	10		50		mV	
	15		–		mV	
OFF-state feed-through	5		–		MHz	note 7
	10		1		MHz	
	15		–		MHz	
ON-state frequency response	5		13		MHz	note 8
	10		40		MHz	
	15		70		MHz	

NOTES

V_{is} is the input voltage at a Y or Z terminal, whichever is assigned as input.
V_{os} is the output voltage at a Y or Z terminal, whichever is assigned as output.

1. $R_L = 10$ kΩ to V_{EE}; $C_L = 50$ pF to V_{EE}; $\overline{E} = V_{SS}$; $V_{is} = V_{DD}$ (square-wave); see Fig. 8.
2. $R_L = 10$ kΩ; $C_L = 50$ pF to V_{EE}; $\overline{E} = V_{SS}$; $A_n = V_{DD}$ (square-wave); $V_{is} = V_{DD}$ and R_L to V_{EE} for t_{PLH}; $V_{is} = V_{EE}$ and R_L to V_{DD} for t_{PHL}; see Fig. 8.
3. $R_L = 10$ kΩ; $C_L = 50$ pF to V_{EE}; $\overline{E} = V_{DD}$ (square-wave);
   $V_{is} = V_{DD}$ and R_L to V_{EE} for t_{PHZ} and t_{PZH};
   $V_{is} = V_{EE}$ and R_L to V_{DD} for t_{PLZ} and t_{PZL}; see Fig. 8.
4. $R_L = 10$ kΩ; $C_L = 15$ pF; channel ON; $V_{is} = ½\ V_{DD(p\text{-}p)}$ (sine-wave, symmetrical about $½\ V_{DD}$); $f_{is} = 1$ kHz; see Fig. 9.
5. $R_L = 1$ kΩ; $V_{is} = ½\ V_{DD(p\text{-}p)}$ (sine-wave, symmetrical about $½\ V_{DD}$);
   $20 \log \frac{V_{os}}{V_{is}} = -50$ dB; see Fig. 10.
6. $R_L = 10$ kΩ to V_{EE}; $C_L = 15$ pF to V_{EE}; $\overline{E}$ or $A_n = V_{DD}$ (square-wave); crosstalk is $|V_{os}|$ (peak value); see Fig. 8.
7. $R_L = 1$ kΩ; $C_L = 5$ pF; channel OFF; $V_{is} = ½\ V_{DD(p\text{-}p)}$ (sine-wave, symmetrical about $½\ V_{DD}$);
   $20 \log \frac{V_{os}}{V_{is}} = -50$ dB; see Fig. 9.
8. $R_L = 1$ kΩ; $C_L = 5$ pF; channel ON; $V_{is} = ½\ V_{DD(p\text{-}p)}$ (sine-wave, symmetrical about $½\ V_{DD}$);
   $20 \log \frac{V_{os}}{V_{is}} = -3$ dB; see Fig. 9.

ADC0802 - ADC0804
8-Bit μP-Compatible A/D Converters

INTERSIL

GENERAL DESCRIPTION

The ADC0802 family are CMOS 8-bit successive approximation A/D converters which use a modified potentiometric ladder, and are designed to operate with the 8080A control bus via three-state outputs. These converters appear to the processor as memory locations or I/O ports, and hence no interfacing logic is required.

The differential analog voltage input has good common-mode-rejection, and permits offsetting the analog zero-input-voltage value. In addition, the voltage reference input can be adjusted to allow encoding any smaller analog voltage span to the full 8 bits of resolution.

FEATURES

- 80C48 and 80C80/85 Bus Compatible — No Interfacing Logic Required
- Conversion Time < 100μs
- Easy Interface to Most Microprocessors
- Will Operate In a "Stand Alone" Mode
- Differential Analog Voltage Inputs
- Works With Bandgap Voltage References
- TTL Compatible Inputs and Outputs
- On-Chip Clock Generator
- 0V to 5V Analog Voltage Input Range (Single +5V Supply)
- No Zero-Adjust Required

ORDERING INFORMATION

Part Number	Error	Temperature Range	Package
ADC0802LCN	± ½ bit no adjust	0°C to +70°C	20 pin Plastic DIP
ADC0802LCD	± ¾ bit no adjust	−40°C to +85°C	20 pin CERDIP
ADC0802LD	± 1 bit no adjust	−55°C to +125°C	20 pin CERDIP
ADC0803LCN	± ½ bit adjusted full-scale	0°C to +70°C	20 pin Plastic DIP
ADC0803LCD	± ¾ bit adjusted full-scale	−40°C to +85°C	20 pin CERDIP
ADC0803LD	± 1 bit adjusted full-scale	−55°C to +125°C	20 pin CERDIP
ADC0804LCN	± 1 bit no adjust	0°C to +70°C	20 pin Plastic DIP
ADC0804LCD	± 1 bit no adjust	−40°C to +85°C	20 pin CERDIP

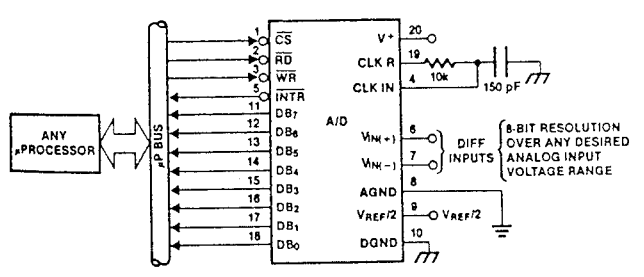
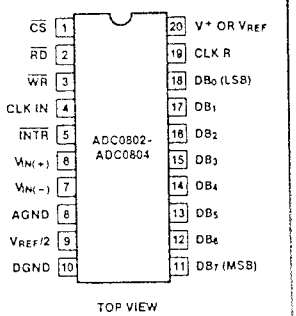


Figure 1: Typical Application



(Outline dwg. CD, CN)
Figure 2: Pin Configuration

NOTE: All typical values have been characterized but are not tested.

ADC0802-ADC0804

ABSOLUTE MAXIMUM RATINGS

Supply Voltage 6.5V
Voltage at Any Input ..... −0.3V to (V^+ +0.3V)
Storage Temperature Range ..... −65°C to +150°C
Package Dissipation at T_A = +25°C 875mW
Lead Temperature (Soldering, 10sec) ..... 300°C

OPERATING RATINGS

Temperature Range
ADC0802/03LD ..... −55°C to +125°C
ADC0802/03/04LCD ..... −40°C to +85°C
ADC0802/03/04LCN ..... 0°C to +70°C
Supply Voltage Range ..... 4.5V to 6.3V

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

ELECTRICAL CHARACTERISTICS (Notes 1 and 7)

Converter Specifications: $V^+ = 5V$, $V_{REF}/2 = 2.500V$, $T_A = +25°C$ and $f_{CLK} = 640kHz$ unless otherwise stated.

Parameter	Test Conditions	Min	Typ	Max	Unit
ADC0802: Total Unadjusted Error	Completely Unadjusted			±½	LSB
ADC0803: Total Adjusted Error	With Full Scale Adjust			±½	LSB
ADC0804: Total Unadjusted Error	Completely Unadjusted			±1	LSB
$V_{REF}/2$ Input Resistance	Input Resistance at Pin 9	1.0	1.3		kΩ
Analog Input Voltage Range	(Note 2)	GND−0.05		V^+ +0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range		±1/16	±⅛	LSB
Power Supply Sensitivity	$V^+ = 5V$ ±10% Over Allowed Input Voltage Range		±1/16	±⅛	LSB

Converter Specifications: $V^+ = 5V$, $V_{REF}/2 = 2.500V$, $0°C \le T_A \le +70°C$ and $f_{CLK} = 640$ kHz unless otherwise stated.

Parameter	Test Conditions	Min	Typ	Max	Unit
ADC0802: Total Unadjusted Error	Completely Unadjusted			±½	LSB
ADC0803: Total Adjusted Error	With Full Scale Adjust			±½	LSB
ADC0804: Total Unadjusted Error	Completely Unadjusted			±1	LSB
$V_{REF}/2$ Input Resistance	Input Resistance at Pin 9	1.0	1.3		kΩ
Analog Input Voltage Range	(Note 2)	GND−0.05		V^+ +0.05	V
DC Common-Mode Rejection	Over Analog Input Voltage Range		±1/16	±⅛	LSB
Power Supply Sensitivity	$V^+ = 5V$ ±10% Over Allowed Input Voltage Range		±1/16	±¼	LSB

NOTE: All typical values have been characterized but are not tested.

National Semiconductor

DAC0800, DAC0801, DAC0802 8-Bit Digital-to-Analog Converters

General Description

The DAC0800 series are monolithic 8-bit high-speed current-output digital-to-analog converters (DAC) featuring typical settling times of 100 ns. When used as a multiplying DAC, monotonic performance over a 40 to 1 reference current range is possible. The DAC0800 series also features high compliance complementary current outputs to allow differential output voltages of 20 Vp-p with simple resistor loads as shown in *Figure 1*. The reference-to-full-scale current matching of better than ±1 LSB eliminates the need for full-scale trims in most applications while the nonlinearities of better than ±0.1% over temperature minimizes system error accumulations.

The noise immune inputs of the DAC0800 series will accept TTL levels with the logic threshold pin, V_{LC}, pin 1 grounded. Simple adjustments of the V_{LC} potential allow direct interface to all logic families. The performance and characteristics of the device are essentially unchanged over the full ± 4.5V to ±18V power supply range; power dissipation is only 33 mW with ± 5V supplies and is independent of the logic input states.

The DAC0800, DAC0802, DAC0800C, DAC0801C and DAC0802C are a direct replacement for the DAC-08, DAC-08A, DAC-08C, DAC-08E and DAC-08H, respectively.

Features

- Fast settling output current — 100 ns
- Full scale error — ±1 LSB
- Nonlinearity over temperature — ±0.1%
- Full scale current drift — ±10 ppm/°C
- High output compliance — −10V to +18V
- Complementary current outputs
- Interface directly with TTL, CMOS, PMOS and others
- 2 quadrant wide range multiplying capability
- Wide power supply range — ±4.5V to ±18V
- Low power consumption — 33 mW at ±5V
- Low cost

Typical Applications

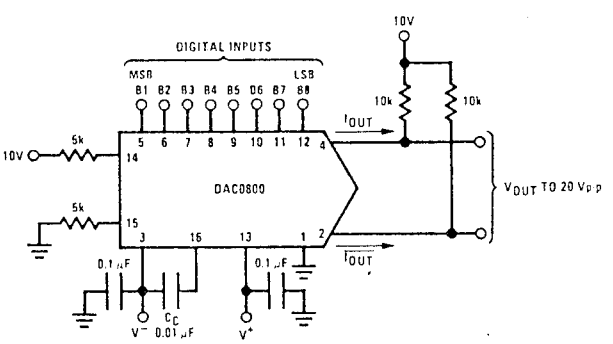
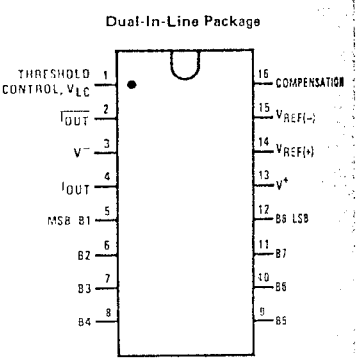


FIGURE 1. ±20 Vp-p Output Digital-to-Analog Converter

Connection Diagram

Dual-In-Line Package



TOP VIEW

Ordering Information

NON LINEARITY	TEMPERATURE RANGE	ORDER NUMBERS*					
		D PACKAGE (D16C)		J PACKAGE (J16A)		N PACKAGE (N16A)	
±0.1% FS	−55°C ≤ T_A ≤ +125°C	DAC0802LD	DAC-08AQ				
±0.1% FS	0°C ≤ T_A ≤ +70°C			DAC0802LCJ	DAC-08HQ	DAC0802LCN	DAC-08HP
±0.19% FS	−55°C ≤ T_A ≤ +125°C	DAC0800LD	DAC-08Q				
±0.19% FS	0°C ≤ T_A ≤ +70°C			DAC0800LCJ	DAC-08EQ	DAC0800LCN	DAC-08EP
±0.39% FS	0°C ≤ T_A ≤ +70°C			DAC0801LCJ	DAC-08CQ	DAC0801LCN	DAC-08CP

*Note. Devices may be ordered by using either order number.

	Mnemonic	Op code (hex)	Description
A	ADD *A*	87	Add *A* to *A* (double *A*)
	ADD *B*	80	Add *B* to *A*
	ADD *C*	81	Add *C* to *A*
	ADD *D*	82	Add *D* to *A*
	ADD *E*	83	Add *E* to *A*
	ADD *H*	84	Add *H* to *A*
	ADD *L*	85	Add *L* to *A*
	ADD *M*	86	Add memory LOC (*H* & *L*) to *A*
	ADI *v*	C6	Add immediate data *v* to *A*
	ADC *A*	8F	Add *A* to *A* with carry (double *A* with carry)
	ADC *B*	88	Add *B* to *A* with carry
	ADC *C*	89	Add *C* to *A* with carry
	ADC *D*	8A	Add *D* to *A* with carry
	ADC *E*	8B	Add *E* to *A* with carry
	ADC *H*	8C	Add *H* to *A* with carry
	ADC *L*	8D	Add *L* to *A* with carry
	ADC *M*	8E	Add memory LOC (*H* & *L*) to *A* with carry
	ACI *v*	CE	Add immediate data *v* to *A* with carry
	ANA *A*	A7	Test *A* and clear carry
	ANA *B*	A0	AND *B* with *A*
	ANA *C*	A1	AND *C* with *A*
	ANA *D*	A2	AND *D* with *A*
	ANA *E*	A3	AND *E* with *A*
	ANA *H*	A4	AND *H* with *A*
	ANA *L*	A5	AND *L* with *A*
	ANA *M*	A6	AND memory LOC (*H* & *L*) with *A*
	ANI *v*	E6	AND immediate data *v* with *A*
C	CALL *aa*	CD	Call subroutine at address *aa*
	CZ *aa*	CC	If zero, CALL at address *aa*
	CNZ *aa*	C4	If not zero, CALL at address *aa*
	CP *aa*	F4	If plus, CALL at address *aa*
	CM *aa*	FC	If minus, CALL at address *aa*
	CC *aa*	DD	If carry, CALL at address *aa*
	CNC *aa*	D4	If no carry, CALL at address *aa*
	CPE *aa*	EC	If even parity, CALL at address *aa*
	CPO *aa*	E4	If odd parity, CALL at address *aa*
	CMA	2F	Complement *A*
	CMC	3F	Complement carry
	CMP *A*	BF	Set zero flag
	CMP *B*	B8	Compare *A* with *B*
	CMP *C*	B9	Compare *A* with *C*
	CMP *D*	BA	Compare *A* with *D*
	CMP *E*	BB	Compare *A* with *E*
	CMP *H*	BC	Compare *A* with *H*
	CMP *L*	BD	Compare *A* with *L*
	CMP *M*	BE	Compare *A* with memory LOC (*H* & *L*)
	CPI *v*	FE	Compare *A* with immediate data *v*

Fig. 9-1 Instruction set summary for the 8080/8085 microprocessor

	Mnemonic	Op code (hex)	Description
D	DAA	27	Decimal adjust A
	DAD *B* DAD *D* DAD *H* DAD *SP*	09 19 29 39	Add *B* & *C* to *H* & *L* Add *D* & *E* to *H* & *L* Add *H* & *L* to *H* & *L* (double *H* & *L*) Add *SP* to *H* & *L*
	DCR *A* DCR *B* DCR *C* DCR *D* DCR *E* DCR *H* DCR *L*	3D 05 0D 15 1D 25 2D	Decrement *A* Decrement *B* Decrement *C* Decrement *D* Decrement *E* Decrement *H* Decrement *L*
	DCR *M*	35	Decrement memory LOC (*H* & *L*)
	DCX *B* DCX *D* DCX *H* DCX *SP*	0B 1B 2B 3B	Decrement *B* & *C* Decrement *D* & *E* Decrement *H* & *L* Decrement *SP*
	DI	F3	Disable interrupts
E	EI	FB	Enable interrupts
H	HLT	76	Halt until interrupt
I	IN *v*	DB	Input from device *v*
	INR *A* INR *B* INR *C* INR *D* INR *E* INR *H* INR *L*	3C 04 0C 14 1C 24 2C	Increment *A* Increment *B* Increment *C* Increment *D* Increment *E* Increment *H* Increment *L*
	INR *M*	34	Increment memory LOC (*H* & *L*)
	INX *B* INX *D* INX *H* INX *SP*	03 13 23 33	Increment *B* & *C* Increment *D* & *E* Increment *H* & *L* Increment *SP*
J	JMP *aa*	C3	Jump to address *aa*
	JZ *aa* JNZ *aa* JP *aa* JM *aa* JC *aa* JNC *aa* JPE *aa* JPO *aa*	CA C2 F2 FA DA D2 EA E2	If zero JMP to address *aa* If not zero JMP to address *aa* If plus JMP to address *aa* If minus JMP to address *aa* If carry JMP to address *aa* If no carry JMP to address *aa* If even parity JMP to address *aa* If odd parity JMP to address *aa*
L	LDA *aa*	3A	Load *A* from address *aa*
	LDAX *B* LDAX *D*	0A 1A	Load *A* from memory LOC (*B* & *C*) Load *A* from memory LOC (*D* & *E*)
	LHLD *aa*	2A	Load *H* & *L* from address *aa*

Mnemonic	Op code (hex)	Description
LXI *B,vv*	01	Load *B* & *C* with immediate data *vv*
LXI *D,vv*	11	Load *D* & *E* with immediate data *vv*
LXI *H,vv*	21	Load *H* & *L* with immediate data *vv*
LXI *SP,vv*	31	Load *SP* with immediate data *vv*
MOV *A,B*	78	Move *B* to *A*
MOV *A,C*	79	Move *C* to *A*
MOV *A,D*	7A	Move *D* to *A*
MOV *A,E*	7B	Move *E* to *A*
MOV *A,H*	7C	Move *H* to *A*
MOV *A,L*	7D	Move *L* to *A*
MOV *A,M*	7E	Move memory LOC (*H* & *L*) to *A*
MOV *B,A*	47	Move *A* to *B*
MOV *B,C*	41	Move *C* to *B*
MOV *B,D*	42	Move *D* to *B*
MOV *B,E*	43	Move *E* to *B*
MOV *B,H*	44	Move *H* to *B*
MOV *B,L*	45	Move *L* to *B*
MOV *B,M*	46	Move memory LOC (*H* & *L*) to *B*
MOV *C,A*	4F	Move *A* to *C*
MOV *C,B*	48	Move *B* to *C*
MOV *C,D*	4A	Move *D* to *C*
MOV *C,E*	4B	Move *E* to *C*
MOV *C,H*	4C	Move *H* to *C*
MOV *C,L*	4D	Move *L* to *C*
MOV *C,M*	4E	Move memory LOC (*H* & *L*) to *C*
MOV *D,A*	57	Move *A* to *D*
MOV *D,B*	50	Move *B* to *D*
MOV *D,C*	51	Move *C* to *D*
MOV *D,E*	53	Move *E* to *D*
MOV *D,H*	54	Move *H* to *D*
MOV *D,L*	55	Move *L* to *D*
MOV *D,M*	56	Move memory LOC (*H* & *L*) to *D*
MOV *E,A*	5F	Move *A* to *E*
MOV *E,B*	58	Move *B* to *E*
MOV *E,C*	59	Move *C* to *E*
MOV *E,D*	5A	Move *D* to *E*
MOV *E,H*	5C	Move *H* to *E*
MOV *E,L*	5D	Move *L* to *E*
MOV *E,M*	5E	Move memory LOC (*H* & *L*) to *E*
MOV *H,A*	67	Move *A* to *H*
MOV *H,B*	60	Move *B* to *H*
MOV *H,C*	61	Move *C* to *H*
MOV *H,D*	62	Move *D* to *H*
MOV *H,E*	63	Move *E* to *H*
MOV *H,L*	65	Move *L* to *H*
MOV *H,M*	66	Move memory LOC (*H* & *L*) to *H*
MOV *L,A*	6F	Move *A* to *L*
MOV *L,B*	68	Move *B* to *L*
MOV *L,C*	69	Move *C* to *L*
MOV *L,D*	6A	Move *D* to *L*
MOV *L,E*	6B	Move *E* to *L*
MOV *L,H*	6C	Move *H* to *L*

Fig. 9-1 Instruction set summary for the 8080/8085 microprocessor (*cont.*)

	Mnemonic	Op code (hex)	Description
	MOV *L,M*	6E	Move memory LOC (*H* & *L*) to *L*
M	MOV *M,A*	77	Move *A* to memory LOC (*H* & *L*)
	MOV *M,B*	70	Move *B* to memory LOC (*H* & *L*)
	MOV *M,C*	71	Move *C* to memory LOC (*H* & *L*)
	MOV *M,D*	72	Move *D* to memory LOC (*H* & *L*)
	MOV *M,E*	73	Move *E* to memory LOC (*H* & *L*)
	MOV *M,H*	74	Move *H* to memory LOC (*H* & *L*)
	MOV *M,L*	75	Move *L* to memory LOC (*H* & *L*)
	MVI *A,v*	3E	Move immediate data *v* to *A*
	MVI *B,v*	06	Move immediate data *v* to *B*
	MVI *C,v*	0E	Move immediate data *v* to *C*
	MVI *D,v*	16	Move immediate data *v* to *D*
	MVI *E,v*	1E	Move immediate data *v* to *E*
	MVI *H,v*	26	Move immediate data *v* to *H*
	MVI *L,v*	2E	Move immediate data *v* to *L*
	MVI *M,v*	36	Move immediate data *v* to memory LOC (*H* & *L*)
N	NOP	00	No operation
O	ORA *A*	B7	Test *A* and clear carry
	ORA *B*	B0	OR *B* with *A*
	ORA *C*	B1	OR *C* with *A*
	ORA *D*	B2	OR *D* with *A*
	ORA *E*	B3	OR *E* with *A*
	ORA *H*	B4	OR *H* with *A*
	ORA *L*	B5	OR *L* with *A*
	ORA *M*	B6	OR memory LOC (*H* & *L*) with *A*
	ORI *v*	F6	OR immediate data *v* with *A*
	OUT *p*	D3	Output *A* to device *v*
P	PCHL	E9	Jump to memory LOC contained in (*H* & *L*)
	POP *B*	C1	Pop *B* & *C* from stack
	POP *D*	D1	Pop *D* & *E* from stack
	POP *H*	E1	Pop *H* & *L* from stack
	POP *PSW*	F1	Pop *A* and flags from stack
	PUSH *B*	C5	Push *B* & *C* onto stack
	PUSH *D*	D5	Push *D* & *E* onto stack
	PUSH *H*	E5	Push *H* & *L* onto stack
	PUSH *PSW*	F5	Push *A* and flags onto stack
R	RAL	17	Rotate *CY* + *A* left
	RAR	1F	Rotate *CY* + *A* right
	RLC	07	Rotate *A* left and into carry
	RRC	0F	Rotate *A* right and into carry
	RIM	20	Read interrupt mask (8085 only)
	RET	C9	Return from subroutine
	RZ	C8	If zero, return from subroutine
	RNZ	C0	If not zero, return from subroutine
	RP	F0	If plus, return from subroutine
	RM	F8	If minus, return from subtoutine
	RC	D8	If carry, return from subroutine
	RNC	D0	If no carry, return from subroutine
	RPE	E8	If even parity, return from subroutine
	RPO	E0	If odd parity, return from subroutine

Fig. 9-1 Instruction set summary for the 8080/8085 microprocessor (*cont.*)

	Mnemonic	Op code (hex)	Description
R	RST 0	C7	Restart subroutine at address 00H
	RST 1	CF	Restart subroutine at address 08H
	RST 2	D7	Restart subroutine at address 10H
	RST 3	DF	Restart subroutine at address 18H
	RST 4	E7	Restart subroutine at address 20H
	RST 5	EF	Restart subroutine at address 28H
	RST 6	F7	Restart subroutine at address 30H
	RST 7	FF	Restart subroutine at address 38H
S	SIM	30	Set interrupt mask (8085 only)
	SPHL	F9	Load *SP* from *H* & *L*
	SHLD *aa*	22	Store *H* & *L* at memory LOC *aa*
	STA *aa*	32	Store *A* at memory LOC *aa*
	STAX *B* STAX *D*	02 12	Store *A* at memory LOC (*B* & *C*) Store *A* at memory LOC (*D* & *E*)
	STC	37	Set carry flag
	SUB *A* SUB *B* SUB *C* SUB *D* SUB *E* SUB *H* SUB *L*	97 90 91 92 93 94 95	Clear *A* Subtract *B* from *A* Subtract *C* from *A* Subtract *D* from *A* Subtract *E* from *A* Subtract *H* from *A* Subtract *L* from *A*
	SUB *M*	96	Subtract contents of memory LOC (*H* & *L*) from *A*
	SUI *v*	D6	Subtract immediate data *v* from *A*
	SBB *A* SBB *B* SBB *C* SBB *D* SBB *E* SBB *H* SBB *L*	9F 98 99 9A 9B 9C 9D	Set *A* to minus carry Subtract *B* from *A* with borrow Subtract *C* from *A* with borrow Subtract *D* from *A* with borrow Subtract *E* from *A* with borrow Subtract *H* from *A* with borrow Subtract *L* from *A* with borrow
	SBB *M*	9E	Subtract memory LOC (*H* & *L*) from *A* with borrow
	SBI *v*	DE	Subtract immediate data *v* from *A* with borrow
X	XCHG	EB	Exchange *D* & *E* with *H* & *L*
	XTHL	E3	Exchange top of stack with *H* & *L*
	XRA *A* XRA *B* XRA *C* XRA *D* XRA *E* XRA *H* XRA *L*	AF A8 A9 AA AB AC AD	Clear *A* Exclusive OR *B* with *A* Exclusive OR *C* with *A* Exclusive OR *D* with *A* Exclusive OR *E* with *A* Exclusive OR *H* with *A* Exclusive OR *L* with *A*
	XRA *M*	AE	Exclusive OR memory LOC (*H* & *L*) with *A*
	XRI *v*	EE	Exclusive OR immediate data *v* with *A*

Fig. 9-1 Instruction set summary for the 8080/8085 microprocessor (*cont.*)

BIBLIOGRAPHY

1. DIGITAL COMPUTER FUNDAMENTALS - VI EDITION
   - **THOMAS C. BARTEE**

2. DIGITAL PRINCIPLES AND APPLICATIONS
   - **ALBERT PAUL MALVINO**
   - **DONALD P. LEACH**

3. MICROPROCESSOR ARCHIETECTURE, PROGRAMMING, AND APPLICATIONS WITH 8085/8080A
   - **RAMESH S. GAONKAR**

4. 8080A - 8085 ASSEMBLY LANGUAGE PROGRAMMING
   - **LANCE A. LEVENTHAL**

5. INTEL'S MCS '85 USER MANUAL

6. LEARN TO USE MICROPROCESSOR - II EDITION
   - **K. PADMANABHAN**

7. MICROPROCESSOR AND ITS INTERFACING DEVICES
   - **D. HALL**

8. INTRODUCTION TO MICROPROCESSORS - III EDITION
   - **ADITYA P. MATHUR**