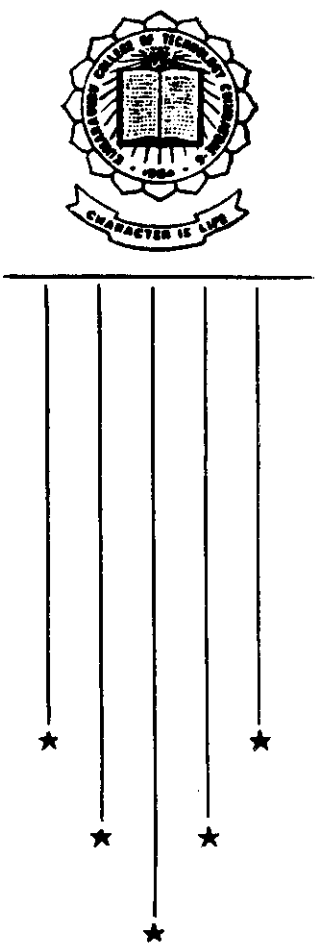
Microcontroller Based Analog Signal Tester

PROJECT REPORT



Submitted by

E. MALAR

S. SUBATHRA

N. SATHEESH KUMAR

R. BABU

Guided by

Mrs. N. KALAIARASI, B.E.,

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF

BACHELOR OF ENGINEERING IN ELECTRICAL & ELECTRONICS ENGINEERING

OF THE BHARATHIAR UNIVERSITY

1997 - 98

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

Kumaraguru College of Technology

COIMBATORE - 641 006.

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE -641 006

CERTIFICATE

This is to certify that the report entitled

Microcontroller Based Analog Signal Tester

has been submitted by

Mr./Ms. _____

In partial fulfilment for the award of Bachelor of Engineering in Electrical and Electronics Engineering Branch of Bharathiar University, Coimbatore - 641 046. during the academic year 1997-'98

4/5/98

Guide

Dr. K. A. PALANISWAMY, B.E.(M.Sc.(Engg.)

Professor and Head

Department of Electrical and Electronics Engineering

Kumaraguru College of Technology

Coimbatore 641 006

Head of the Department

Date : 2/4/98

Certified that the candidate was examined by us in the project work Viva-Voce Examination held on _____ and the University Register Number was _____.

Internal Examiner

External Examiner

PREMIER

CERTIFICATE

This is to certify that the following students of **KUMARAGURU COLLEGE OF TECHNOLOGY**, Coimbatore, of Electrical & Electronics Engineering

E. Malar

S. Subathra

N. Satheesh kumar

R. Babu

had undertaken their project entitled "Microcontroller Based Analog Signal Tester", from September '97 to March '98 at our industry and have successfully completed it.

Their performance during the period was found to be good. We wish them all success.

Place : Coimbatore

Date : 24.03.98

A.G. Raghunath
Project Manager

Industry Seal

Premier Polytronics Limited
304, TRICHY ROAD
SINGANALLUR POST
COIMBATORE-641 005.

premier polytronics ltd.
304, Trichy Road • Singanallur • Coimbatore - 641 005. India.
Tel : 0422 - 573548 • Telex : 0855-8466 PPL IN • Grams : PREMTEX • Fax : 0422 - 573651.
096 TNGST RC No. 1820282 CST RC No. 289178 Dated : 23.6.82.
Regd. Office 185, A.T.D. Street, Race Course, Coimbatore - 641 018.

ACKNOWLEDGEMENT

We record our sincere thanks and profound gratitude to our beloved lecturer Mrs.N.KALAIARASI B.E., M.I.S.T.E., who helped and inspired us right from the time of inception to completion of the project.

We are elated to record our heartiest indebtedness to our Professor and Head of the Department of Electrical and Electronics Engineering Dr.K.A.PALANISWAMY M.Sc., (Engg.), Ph.D., M.I.S.T.E., C.Eng.(I), F.I.E., for his valuable help and encouragement during this project.

We have great pleasure in expressing our gratitude to the Principal Dr.S.SUBRAMANIAN M.Sc.,(Engg.), Ph.D., S.M.IEEE.,for his valuable encouragement.

We acknowledge with gratitude, M/s PREMIER POLYTRONICS LTD., Coimbatore, for having sponsored our project and providing all the facilities to do the project.

We are thankful to Mr.A.G.RAGHUNATH, PROJECT MANAGER - R&D, Premier Polytronics Ltd., for motivating us to do the project and helping us a lot with constant encouragement.

We are also indebted to Mr.M.DEVENDRAN. ASSITANT ENGINEER - R&D, Premier Polytronics Ltd., for his immense help.

We also wish to acknowledge our sincere thanks to all the **staff members** of the ELECTRICAL AND ELECTRONICS ENGINEERING DEPARTMENT for their valuable help.

We are very much thankful to our friends for having helped us in many ways during the course of the project.

SYNOPSIS

Quality management and market leadership are the primary motto of current industrial scenario. Digital technology has paved way for developing methods to produce elegant products by which quality assurance of products can be guaranteed.

This method of quality assurance is extended to reduce time span and fatigue of workers, as time and energy are two non - renewable resources of mankind.

This project elucidates one such test- jig for the YARN QUALITY MONITOR. Yarn Quality Monitor is a textile machinery which continuously monitors the quality of yarn. This jig is designed and fabricated for L1 module of this Yarn Quality Monitor which collects the data regarding the quality of yarn and test to achieve the required quality.

Our ANALOG SIGNAL TESTER being Microcontroller based uses 80C32 Microcontroller (ROMless Version) and has an efficient software which test this PCB at a high speed.

Thus the commercial Yarn Quality Monitor in large numbers can be easily tested using this Analog Signal Tester.

CONTENTS

CHAPTER -1

1.INTRODUCTION

ANALOG SIGNAL TESTER is basically a testing instrument. This is used to test the PCB of the YARN QUALITY MONITOR. The PCB to be tested has eight similar channels. Hence a testing equipment designed for one channel can be used for all eight modules. The modules to be tested is selected using the corresponding latch.

This system consist of electronic circuits like keyboard display interface, decoder, latch etc., microcontroller and a keyboard display card. Through the keyboard the input parameters such as voltage, gain and the channel to be tested are fed in. The keyboard display interface feeds the data to the digital card through 8032 microcontroller.

The digital card which is used for serial communication receives the data, selects the channel to be tested and sets the voltage and gain correspondingly. The output signal from the PCB is verified and the result is displayed accordingly.

1.1INTRODUCTION TO YARN QUALITY MONITOR

The process of yarn making consists of converting a stock of fibre into yarn. During conversion forming a long strand of aligned fibres is called as SLIVER. Sliver is twisted to form yarn.

This Yarn Quality Monitor is placed in the stage of conversion of sliver to yarn. This is used to monitor the quality of the yarn.

Yarn Quality Monitor consists of the following four modules.

1. L0 Module
2. L1 Module
3. L2 Module
4. L3 Module

L0 MODULE :

L0 Module is a sensor module. It consists of capacitive sensors to sense the quality of the yarn.

L1 MODULE :

L1 Module does the data collection and processing. It collects the output of the L0 Module as data and processes it.

L2 MODULE :

L2 Module displays the processed output of L1 Module. The quality of the yarn is displayed at various instants.

L3 MODULE :

L3 Module is used to interface the yarn quality monitor to personal computer. A personal computer can control 15 such yarn quality monitors.

Out of this four Modules the complete processing of the data is done by L1 Module. Hence this plays a vital role in determining the quality of the yarn. Failure of any components in this Module leads to drastic errors. To eradicate this problem the performance of this Module has to be tested initially.

1.2 L1 MODULE DESCRIPTION :

This module has eight similar channels of analog circuits. A single channel can be divided in to the following three stages.

1. Offset Null Stage
2. Gain Stage
3. Clipper Stage

OFFSET NULL STAGE :

It consists of an opamp operating in differential mode. In differential mode the output of opamp is the difference of the two input signals. The inputs to the opamp are from a sensor and a DAC.

In any opamp the output offset voltage is the voltage at the output with no input signals applied. If this is not nullified, this gets added up with the actual output.

To nullify the output offset voltage following procedure is adopted.

- The output of the opamp is read with the input points grounded i.e., the input from the sensor and DAC is grounded.
- The DAC is set to the value of the output obtained. By this the accumulation of error voltage is eliminated.

GAIN STAGE :

This stage amplifies the output signal of the offset null stage. As the input from the sensor is in the range of millivolts, any change in the yarn quality cannot be determined accurately.Therefore this weak signal should be amplified appropriately. Gain for which the signals should be can be varied by the user, through microcontroller and latches.

CLIPPER STAGE :

Microcontroller can read only the signals of voltage ranging from 0 - 5V. Any signal beyond this limit is excluded. To avoid this, any signal which is below 0V is grounded through diode D1 and the signals above 5V is directed to Vref through diode D2. Thus only the signals from 0 - 5V is allowed to enter microcontroller. The circuit diagram of the L1 module is shown in figure1.1.

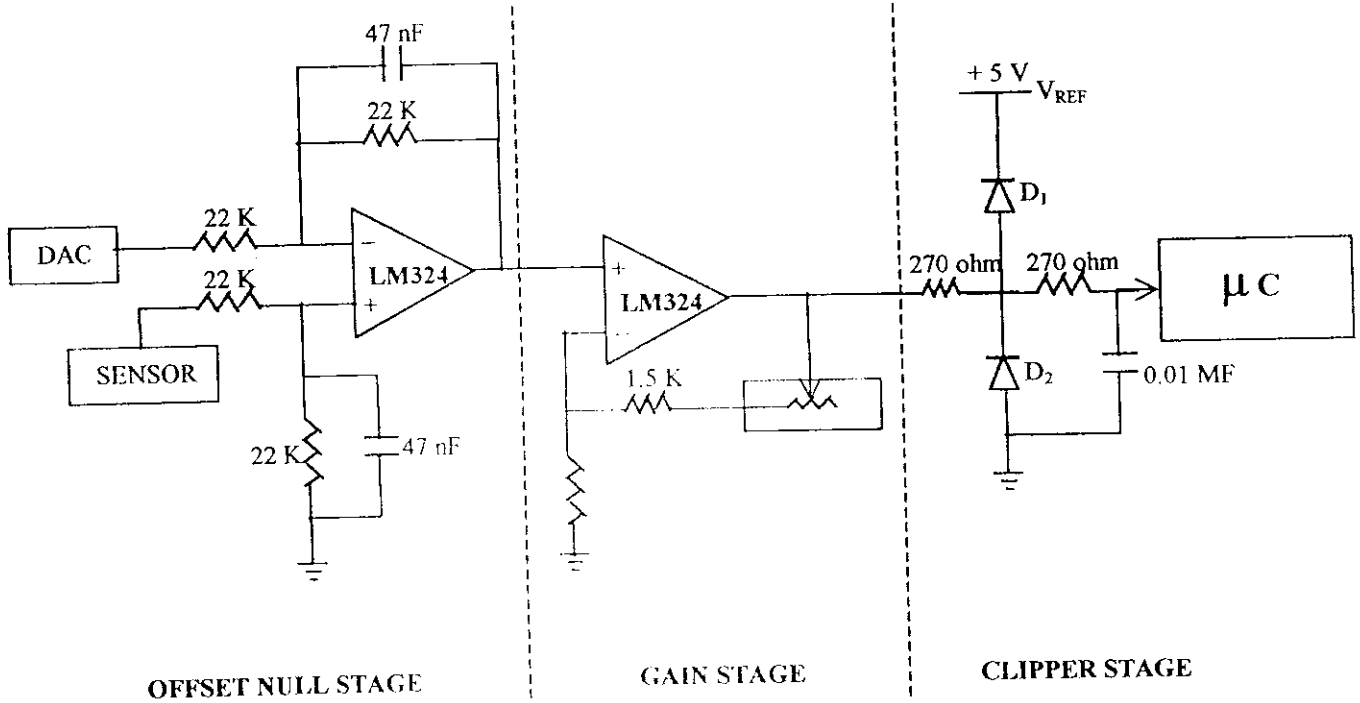


FIG 1.1 BLOCK DIAGRAM OF L1 MODULE

CHAPTER - 2

HARDWARE DESCRIPTION

2.1 THE 8279 PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE:

It is a keyboard display interface. It is used to interface the keyboard and seven segment display to 8032. The keyboard segment of 8279 can be connected to a 16-contact key matrix through scan and return lines. The channel to be tested, voltage and the gain are fed in through the keyboard in the following format.

C	X1	G	X2	V	X3

C : Channel

X1 : Numbers from 0 - 7 (Channel number)

G : Gain

X2 : Numerical value of gain

V : Voltage

X3 : Numerical value of voltage.

This data is stored in the FIFO RAM of 8279 to be read by 8032. The output data to be displayed from microcontroller is stored in display RAM. This is then displayed in the six character, seven segment display.

2.2 DECODER [74LS138] :

This is a 3-8 bit decoder. This is employed in interfacing the Keyboard Display interface (8279) with 80C32. The decoder is a logic circuit that identifies each combination of the signals present at its input. The 74LS138 has three input lines and eight active low output lines. It requires three enable inputs. Two are active low and one is active high. All three enable lines should be activated so that the device can function as a decoder. The address lines A13, A14, A15 of the microcontroller are the input lines to the decoder1. The memory map of RAM ranges from 0000H to 7FFFH. For various combinations of A13, A14 and A15 the RAM is selected. The decoder circuit diagram is shown in fig 2.3.

Decoder2 is used to select 8279.The address lines A9, A10, A11 are used as input lines. The memory map of 8279 ranges from E400H to E5FFH . The output of the decoder is connected to the chip select pins of RAM and 8279. When chip select is low, the output lines go into the high impedance state . Then the 8279 or RAM is activated .

2.3 MICROCONTROLLER [80C32] :

The purpose of this chip is to execute the users program instruction by executing a series of its microinstructions there by controlling data transfer and all functions from computed results.

The microcontroller has two 16 bit timers. The timers are versatile and can be used for A/D conversion, pulse width measurement and frequency counting. They can also act as counters. Signal pulses from transducers like phototransistors could be conventionally counted. The UART in 8032 enables it to transmit and receive serial data from other devices like CRT terminal, computer and printer. Two interrupts are allowed from external sources. There is a free I/O port available in 8032 in addition to three other ports. But 8032 uses these ports for external memory since it has no in-built EPROM or ROM. Nevertheless it saves a lot of hardware. Moreover any bit of a port can be read by the software or toggled or processed.

The microcontroller serves as a bit processor, because it has instructions to deal with a bit and thus can do Boolean logic functions more conveniently. The 8032 has a low power mode in which its internal RAM and registers can be retained with low battery power.

2.4 LATCH [74LS573] :

The latch is employed to demultiplex the address/data lines. During transfer of data from memory to the controller, the higher order address remains on the bus. However the lower order address bits will be lost after the first T-state. Hence the need for address to be latched and used for identifying the memory address. When the ALE goes high, the

latch is transparent i.e., the output changes according to the input. When ALE is low, the lower order address is latched until the next ALE and the output of the latch represents the lower order address bus (A0-A7) after the latching operation .

2.5 EPROM [27512] :

Erasable programmable read only memory is a type of memory which can be reprogrammed again and again simply by erasing it subjecting it to ultraviolet rays.

Programming the EPROM:

The programming tool used for loading the EPROM is PC-UPROG.UPROG system is designed with 40 pin socket that supports most devices. PC-UPROG is a convenient programming tool for all kinds of programmable devices. PC-UPROG uses the popular IBM and IBM compatible desk top computer as its host system for data and algorithm storage while actual programming is under control of the hardware pulse generators and D/A converter of the PC-UPROG system. PC-UPROG is very user friendly, no special knowledge of programmable devices are required. All the operations are menu driven and displayed in clear, easy to understand English text.

Erasing the EPROM:

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4000 Angstroms. Since sunlight and fluorescent have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent light) could cause inadvertent erasure. If an application subjects the device of this type of exposure, it is suggested that an opaque label be placed over the window.

The recommended erasure procedure is exposure to ultraviolet light (at 2537Angstroms) to an integrated dose of atleast 15WSecs/sq.cm. Exposing the EPROM to an ultraviolet lamp of 12,000 mW/sq.cm rating for 20 to 30mins, at a distance of about 1 inch, should be sufficient. Erasure leaves the array in all 1's state.

2.6 STATIC RAM [MCM 62256 / 63256] :

It is a read-write memory i.e., a static RAM. A static RAM is essentially a matrix of flipflops. Therefore one can write a new data word in RAM location at any time by applying the word to the flipflop data inputs. Here the EPROM and RAM are employed as external memory to the controller.

2.7 VOLTAGE COMPARATOR [7705] :

Voltage comparator is used to protect the controller from over voltage and under voltage levels of the supply voltage. The comparator monitors the supply voltage and compares it with a reference. If the voltage drops or rises above V_{ref}, then a hardware reset will be initiated.

2.8 WATCH DOG TIMER :

Watch dog timer is a missing pulse detector. It provides a means to recover from a software upset . When the watch dog timer is enabled, it will initiate a hardware reset. So the software should clear it every 64K state times. The clearing pulses are monitored. If the pulses are missed until the gets overflowed, it pulls down the reset pin resetting the 8032 and all other devices tied to the reset line.

2.9 DIGITAL CARD:

This card consists of 80C196 microcontroller, DACs, latches, RAMs and EPROMs. The latches used here to set gain for the channel to be tested. The DACs are used to convert the digital input from 8032 to analog signal. RAMs and EPROMs are used as external memories for 80C196. 80C196 is operated in mode 1,(i.e) asynchronous mode. This digital card serves as serial communication port between 8032 card and the PCB to be tested.

2.10 POWER SUPPLY :

The power supply for all IC's is given from the +5V of the regulated supply. These IC's can tolerate the fluctuations within certain limits .These limits are fixed by the voltage monitoring IC. The power supply diagram is shown in fig 2.2.

2.11 Logic Diagram Description :

The circuit designed is to interface the keyboard and display unit with that of the digital card. The operation of interfacing is Microcontroller based. 80C32 Microcontroller is used for this purpose.

80C32 has only 128 bytes of data memory on chip and it has no program memory. But 80C32 can address 64 KB of RAM and 64 KB of program memory (EPROM) externally. The program to be executed is loaded in 27512 EPROM, as it has 64 KB of memory. For temporary data storage we need 32 KB of RAM. This is done by 63256 RAM.

8279 keyboard display interface is used to interface keyboard display card with that of 80C32 Microcontroller. 74LS573 3-8 8 bit decoders are used to select 63256 (RAM) and 8279 chips. The active low chip select line to the EPROM is grounded to make it available always.

The data and address lines are shared by different chips. The lower order address lines and data lines are multiplexed. For this purpose a latch is used. An octal bidirectional transceiver or buffer (74LS245) is used as temporary storage of data. Since bidirectional it can transfer data from Microcontroller to EPROM or RAM or vice versa.

Chip 1232 is used to reset the hardware externally. Supply to all the chips is provided using a 7705 voltage comparator. This provides 5 V of supply. Connector pins are provided to interface this board with keyboard display card and digital card.

The keyboard used here is 4 X 4 matrix keyboard. Six characters can be displayed using seven segment display. The PCB for this circuit is designed and fabricated. The logic diagram is shown in fig.2.1.

2.12 TESTING PROCEDURE :

The PCB to be tested is interfaced with the digital card and 80C32 microcontroller board. The channel to be selected ,the testing voltage and the gain are fed as inputs through the keyboard display card which is directly linked to the 80C32 board. The 8279 keyboard display interface scans the keyed in data and stores in FIFO RAM temporarily. This data is transferred to 80C32 microcontroller which then passes this

data to digital card through serial communication port. The entire program to perform this process is loaded in the EPROM. RAM is used for temporary storage of datas during processing.

The digital board receives the input data through the serial I/O port. Then the digital card selects the particular channel and sets the testing voltage and gain . The gain settings are done through the gain latches in the digital board. The DAC feeds the voltage to be set to the non-inverting terminal of the OP-AMP in the PCB which is to be tested.

The output voltage from the PCB is fed to the digital card. This analog signal is converted into digital by the inbuilt analog to digital converter in 80C196 microcontroller. This microcontroller checks whether the output is corresponding to the input. Based on the performance it then transmits the result to the 80C32 microcontroller through serial I/O port. The data received is in the form of ASCII code which is then converted into display code by the microcontroller. The output whether the channel works or not is displayed in the keyboard display card.

Hence a particular channel in the PCB can be tested as per above procedure. The next channel can be selected and tested successively.

The general block diagram is shown in fig.2.4.

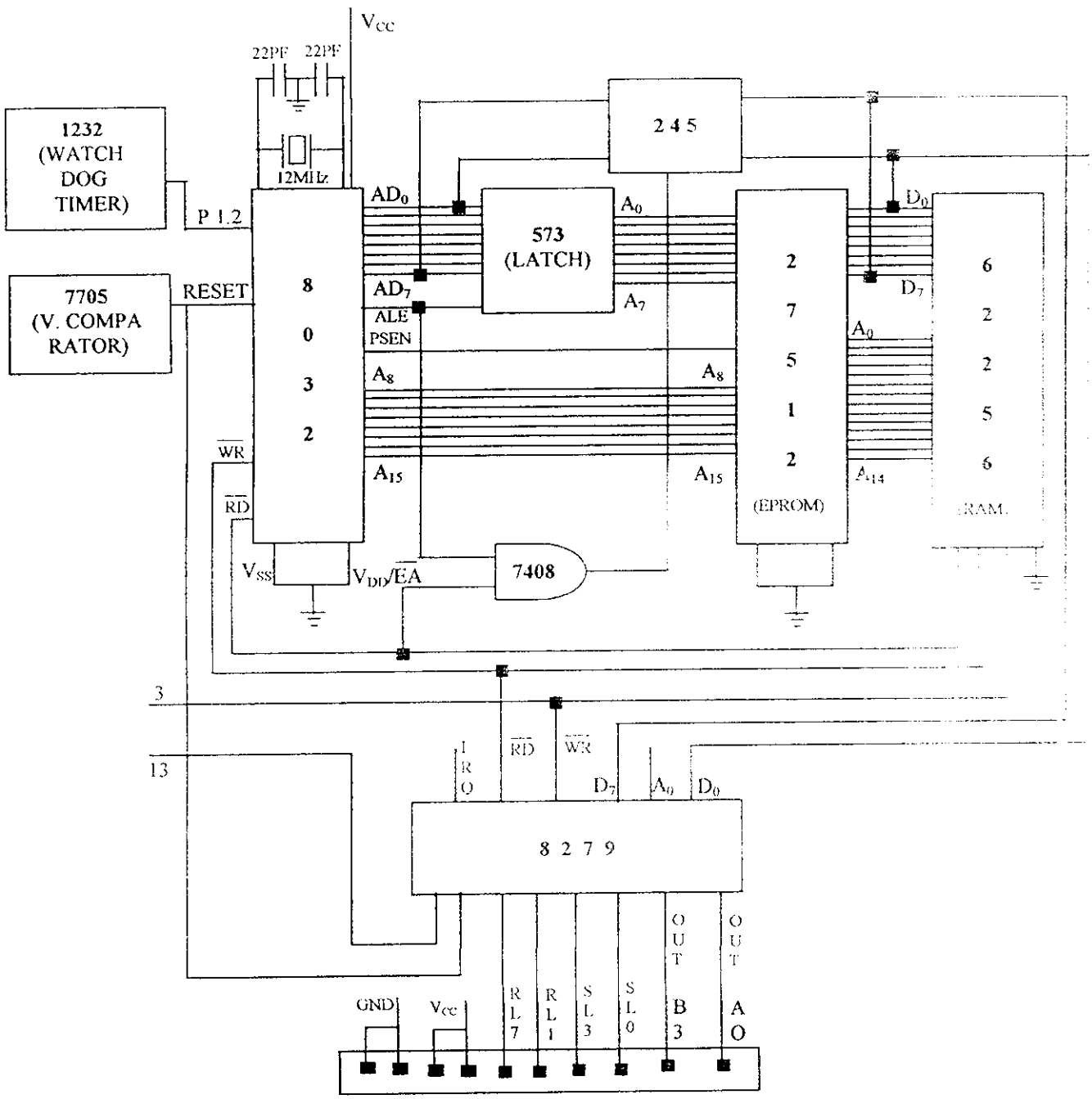


FIG 2.1 **LOGIC DIAGRAM**

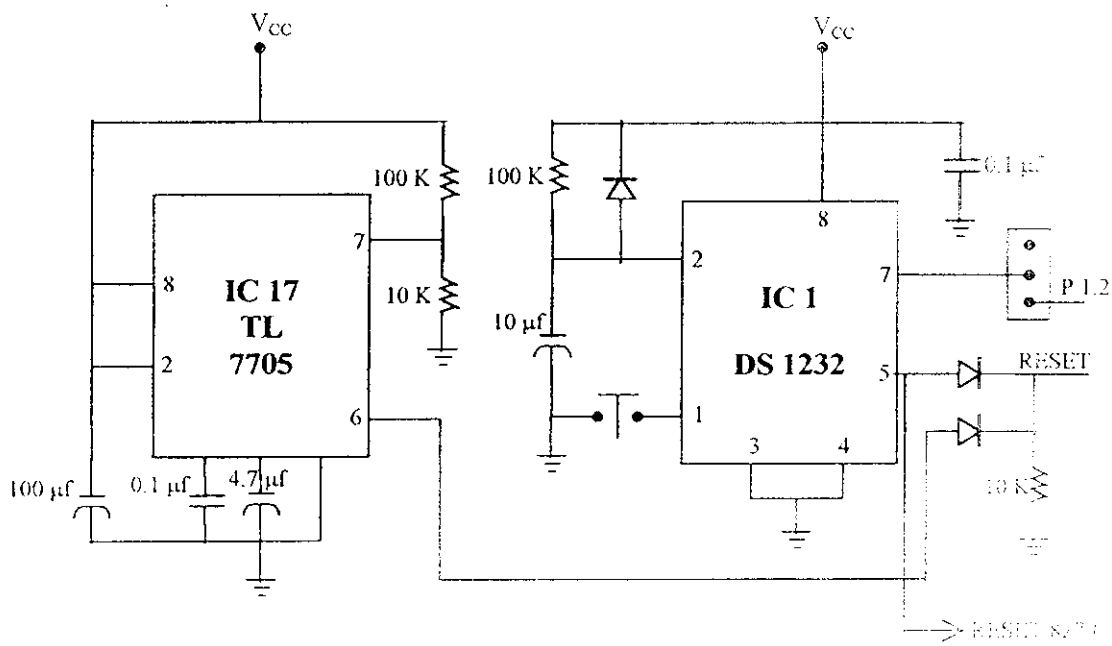


FIG 2.2 **POWER CIRCUIT DIAGRAM**

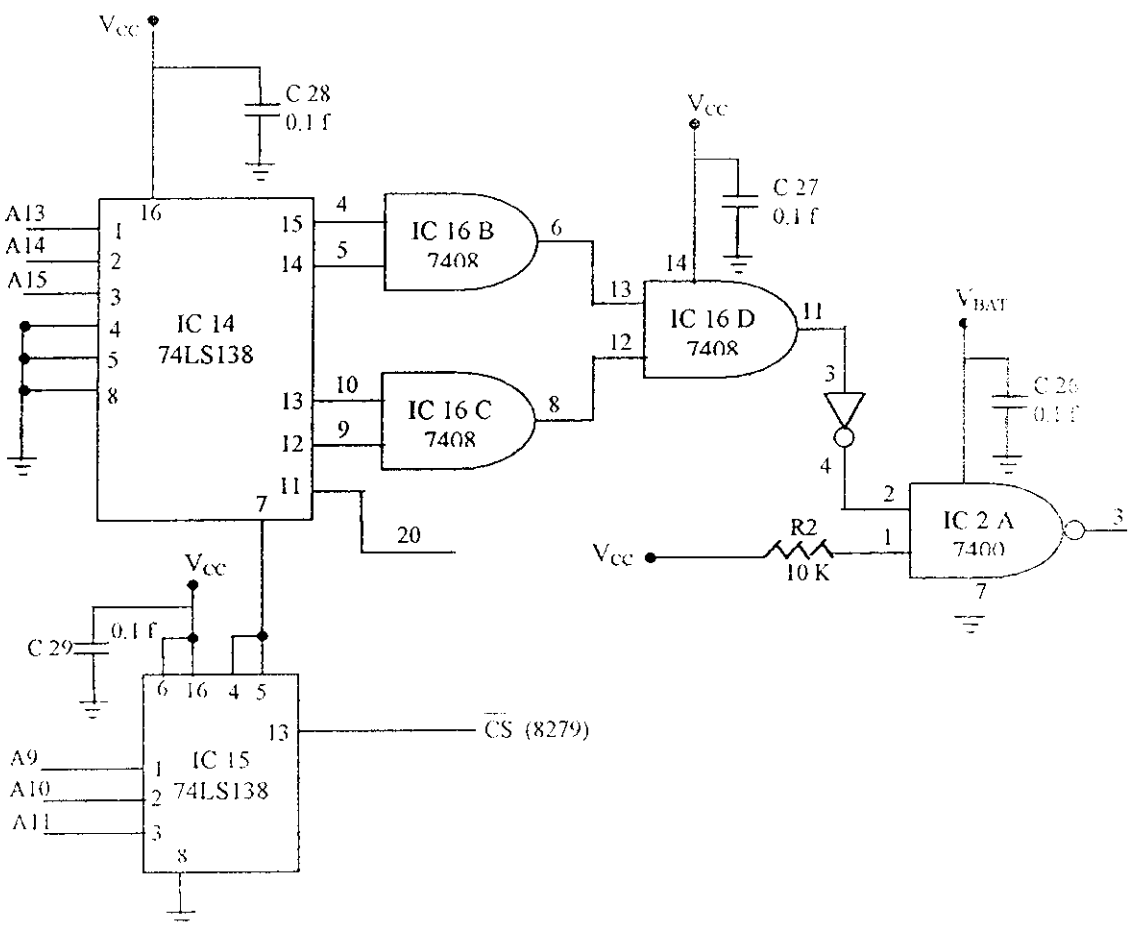


FIG 2.3 **DECODER CIRCUIT DIAGRAM**

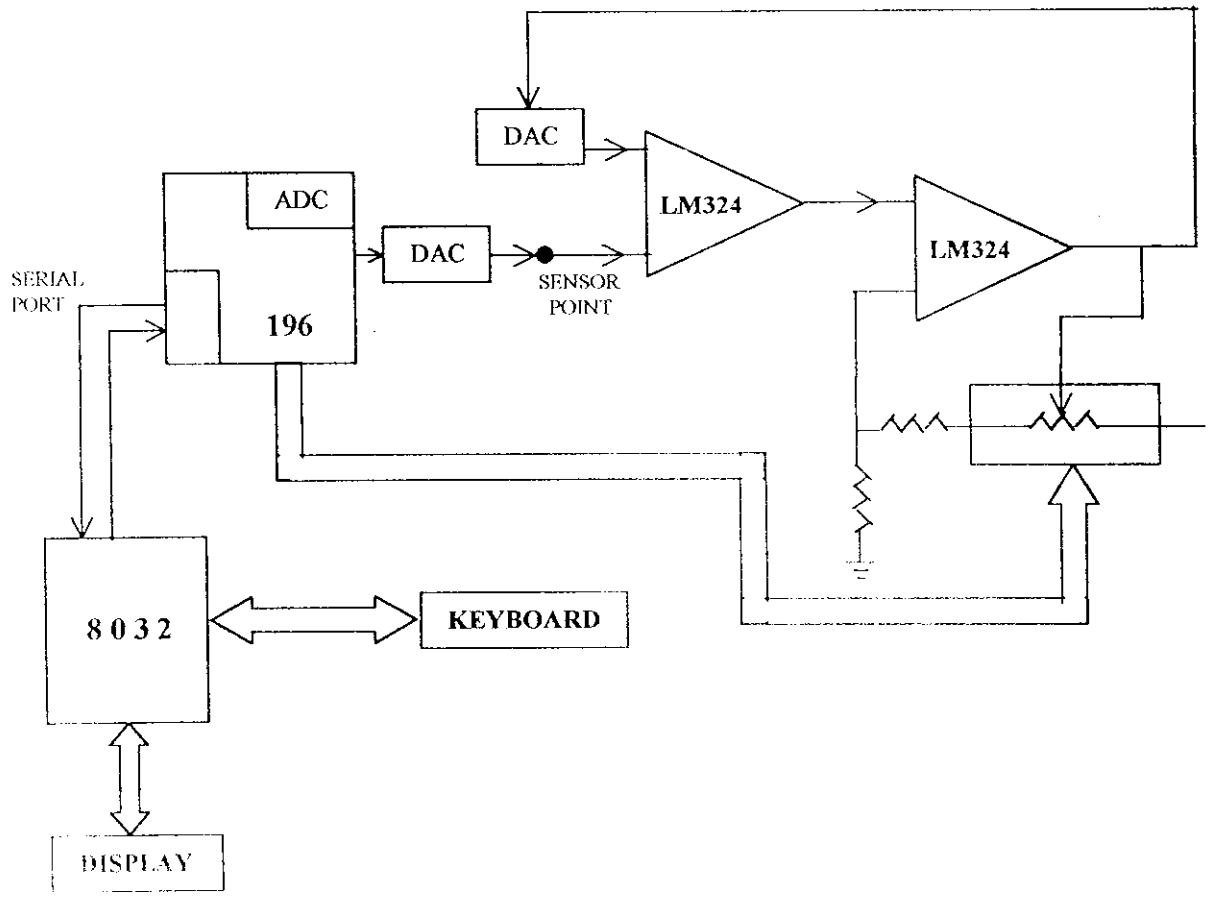


FIG 2.4 **GENERAL BLOCK DIAGRAM**

CHAPTER - 3

8032 DESCRIPTION

3.1 INTRODUCTION :

SUPPLY — Single 5V operation with power down mode for saving internal RAM memory .

INTERNAL MEMORY — 128 bytes of register memory on chip, four banks of eight registers have special functions , two registers have additional indexing capability in each bank .

MEMORY — 64 KB program memory , switchable in 4K banks .
64 KB data memory , fully addressable .

CLOCK TIME — 1 microsecond instruction cycle time with 12MHZ crystal. Two external interrupts, two level priority, level or edge triggerable, two timer interrupts , one UART interrupt .

PORTS — One 8-bit I/O port , bit addressable , one UART port high speed programmable , two counter inputs could

be used for gating timers or counters . Two multimode 16bit (8bit) timers/counters.

ALU — Binary or decimal arithmetic, Boolean bit for control logic programs, hardware 8-bit multiply and divide in 4 microsecs, parity computation, overflow detection.

The block diagram is shown in the fig. 3.1

3.2 HARDWARE FEATURES OF 8032 CPU :

REGISTERS:

Registers are small memories within the CPU. Registers are one part of control section. Like memory storage units, they consist of binary cells and have address that distinguish them. The number of registers is very small however, data may be saved in a register until the program requires it. The registers are distinguished as follows:

ACCUMULATOR :

The accumulator is a eight bit register that is a part of the arithmetic logic unit (ALU). This register is the focus of all accumulator instructions that include arithmetic, logic, store and I/O instructions.

The result of an operation is also stored in the accumulator which is also identified as register A.

PROGRAM COUNTER (PC):

The register is a memory pointer. Memory locations have sixteen bit address. The microcontroller uses this register to sequence the execution of instructions. The microcontroller uses this 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 has to be fetched. When a byte is being fetched the program counter is incremented by one point to the next memory location. The 8032 program counter is 16 - bits wide and can access upto 64KB of program or data memory.

FLAG REGISTER :

The processor flags are combined in one status word (PSW)

D7	D6	D5	D4	D3	D2	D1	D0
CY	AC	FO	RS1	RS0	OV	X	P

CY : Carry flag

AC : Auxiliary carry flag

F0 : Indicator

RS1 : Register bank

RS0 : Register bank

OV : Overflow flag

X : Don't care

P : Parity flag

STACK POINTER (SP):

The stack pointer is also a 16-bit register used as a memory pointer. It points to a memory location in the R/W memory, called the stack which is defined by loading a sixteen bit address in the stack pointer. The stack is also within the RAM only. The stack pointer points to the location where the stack operations has to be done. Stack fills upward from there. Reset defaults the stack pointer to 08, pushing and popping of register / memory location can be done with one byte of instruction.

SERIAL DATA BUFFER :

The serial data buffer is actually two separate registers. When data is moved to SBUF, it goes to the transmit buffer where it is held for serial transmission (Moving a byte to SBUF is what initiates transmission). When data is moved from SBUF, it comes from the receive buffer. During serial reception the incoming bits are clocked into a separate shift register. When reception of a frame is complete and if various other conditions are satisfied, received data bits are transferred from the shift register to the receive buffer. The shift register is then ready to commence reception of a second frame, while the frame already received awaits servicing.

CONTROL AND STATUS REGISTERS:

Special Function Registers IP(Interrupt Priority) IE(Interrupt Enable), TMOD(Timer / Counter Mode), TCON(Timer / Counter Control), SCON(Serial Control) and PCON(Power Control) contain control and status bits for the interrupt system, the timers and the serial port.

STACK :

Data is pushed in to the stack or popped out of the stack on the LIFO basis. The stack can be located anywhere in the RAM memory. The stack pointer maintains the address of the last byte entered into the stack and is decremented everytime data is pushed into the stack and incremented each time data is popped out of the stack. The stack pointer is always incremented or decremented by two bytes.

ARITHMETIC LOGIC UNIT :

The ALU consists of the accumulator, the flag register and some temporary register that are accessible to the user. Arithmetic, logic and rotate operations are performed by the ALU.

INTERNAL TIMERS / COUNTERS:

There are two timers, timer1 and timer0. Actually, two 16-bit registers are used, which incremented one by one, either by a tick of the timer clock or by a count pulse at pins T0 and T1. If a 12MHz crystal is

used, each tick is one microsecond. Thus a timer register can fill up to 65536 clock ticks. Longer time delays can be arranged by using the timer repeatedly in a software loop. As a counter, the same could accumulate pulses at frequencies from DC to 500MHz, with 16-bit precision. The timer could be operated in four modes.

3.3 INTERRUPTS :

The 8032 provides five interrupt sources, each of which can be programmed to one of the two priority levels. The five interrupt sources are listed below.

Source	Description
INT O	External request from P3.2 pin.
Timer 0	Overflow request Timer 0 activates interrupt request flag TF0
INT 1	External request from P3.3 pin.
Timer 1	Overflow from Timer 1 activates interrupt request flag TF1.
Serial Port	Completion of transmission or reception of one serial frame activates request flag TI or RI respectively.

Each source can be individually enabled or disabled by setting or clearing a bit in special function register IE.

IE :

MSB							LSB
EA	X	X	ES	ET1	EX1	ET0	EX0

EA : Enable all.

ES : Enable serial port control bit.

ET0, ET1 : Enable timer 0 and timer 1 control bit

EX0, EX1 : Enable external interrupts 0 or 1

Since there is more than one interrupt, priority is given to them. Set or clear five bits of the IP register to make the priority high or low respectively.

IP :

MSB							LSB
—	—	—	PS	PT1	PX1	PT0	PX0

PS : Serial port priority control bit.

PT0, PT1 : Timer interrupts

PX0, PX1 : External interrupts

Each interrupt has a vector address to which the program branches. In the monitor program, long jump instructions are given to take the interrupt service routines at the addresses given below :

Source	Address
External Interrupt	0003H
Timer 0 Overflow	000BH
External Interrupt 1	0013H
Timer 1 Overflow	001BH
Serial Port	0023H

Execution proceeds from that address until the RET1 instruction is encountered.

3.4 PORT DETAILS:

PORT 0:

Port 0 is a 8 bit open drain bi-directional I/O port. Port 0 pins that have 1's written to them float, and in that state can be used as high impedance inputs. Port 0 is also the multiplexed low order address and data bus during access to external program and data memory. Port 0 also receives the code bytes during programming the EPROM parts, and outputs the code bytes during program verification of the ROM and EPROM parts. External pull-ups are required during program verification.

PORT 1:

Port 1 is an 8-bit bi-directional I/O port with external pull-ups. The port 1 output can sink /source 4 LS TTL inputs. Port 1 pins have 1's written to them are pulled high by the external pull-ups and that state can be used as inputs. As inputs, port 1 pins that are externally being pulled low will source current because of the internal pull-ups.

PORT 2 :

Port 2 is an 8-bit bi-directional I/O port with internal pull-up. The port 2 output buffers can sink/source 4 LS TTL inputs. Port 2 pins that have 1's written to them are pulled high by the internal pull-ups and in that state can be used as inputs. Port 2 also receives the higher order address bits during programming of the EPROM parts and during program verification of the EPROM parts. Port 2 emits the higher order address byte during access to external data memory that use 16-bit address.

PORT 3:

Port 3 is a 8-bit bi-directional I/O port with internal pull-up. The port 3 output buffers can sink /source 4 LS TTL inputs. Port 3 pins that have 1's written to them are pulled high by the internal pull-ups, and in that state it can be used as inputs. Port 3 pins that are externally being pulled low will source current because of the pull-ups. Port 3 also serves the functions of various special features of the 8032 as listed below

PORT PIN	ALTERNATIVE FUNCTION
P 3.0	RXD (Serial input port)
P 3.1	TXD (Serial output port)
P 3.2	INT0 (External interrupt 0)
P 3.3	INT1 (External interrupt 1)
P 3.4	T0 (Timer 0 external input)
P 3.5	T1 (Timer 1 external input)
P 3.6	WR (External data memory write strobe)
P 3.7	RD (External data mmeory read strobe)

3.5 SERIAL INTERFACE:

The serial port is full duplex, meaning it can transmit and receive simultaneously. It is also receive buffered, meaning it can commence reception of a second byte before a previously received byte has been read from the receive register. The serial port registers are both accessed at Special Function Register serial buffer (SBUF). A write to SBUF loads the transmit register and a read access a physically separate receive register.

The serial port can operate in 4 modes as follows:

MODE 0: Serial data enters and exits through RXD. TXD outputs the shift clock. 8-bits are transmitted / received : 8 data bits (LSB first). The baud rate is fixed at 1/12 the oscillator frequency.

MODE 1: 10 bits are transmitted (through TXD) or received (through RXD): a start bit (0), 8 data bits (LSB first), and a stop bit(1). On receive, the stop bit goes into RB8 in Special Function Register SCON. The baud rate is variable.

MODE 2 : 11 bits are transmitted (through TXD) or received (through RXD): a start bit (0), 8 data bits (LSB first), a programmable 9^{th} data bit, and a stop bit (1). On transmit, the 9^{th} data bit (TB8) can be assigned the value of 0 or 1. With nominal software overhead, TB8 can be made a parity bit. On receive, the 9^{th} data bit goes into RB8 in Special Function Register SCON, and the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 the oscillator frequency.

MODE 3 : 11 bits are transmitted (through TXD) or received (through RXD):a start bit (0), 8 data bits (LSB first), a programmable 9^{th} data bit, and a stop bit (1).

SERIAL PORT CONTROL REGISTER :

Special Function Register SCON is used to define the operating modes and control certain functions of the serial port. It also receives the 9th data bit (RB8), and contains the transmit and receive interrupt flags (TI and RI). The register is as shown below:

7	6	5	4	3	2	1	0
SM0	SM1	SM2	REN	TB8	RB8	T1	R1

Where SM0 and SM1 specify the port mode, as follows:

SM1	SM1	Mode	Description	Baud rate
0	0	0	shift register	f osc. / 12
0	1	1	8-bit UART	variable
1	0	2	9-bit UART	f osc. / 64 or f osc. / 32
1	1	3	9-bit UART	variable

- SM2 enables the multiprocessor communication feature in modes 2 and 3.
- REN enables serial reception.

- TB8 is the 9^{th} data bit that will be transmitted in modes 2 and 3.
- RB8 in modes 2 and 3, is the 9^{th} data bit that was received.
- TI is transmit interrupt flag.
- RI is receive interrupt flag.

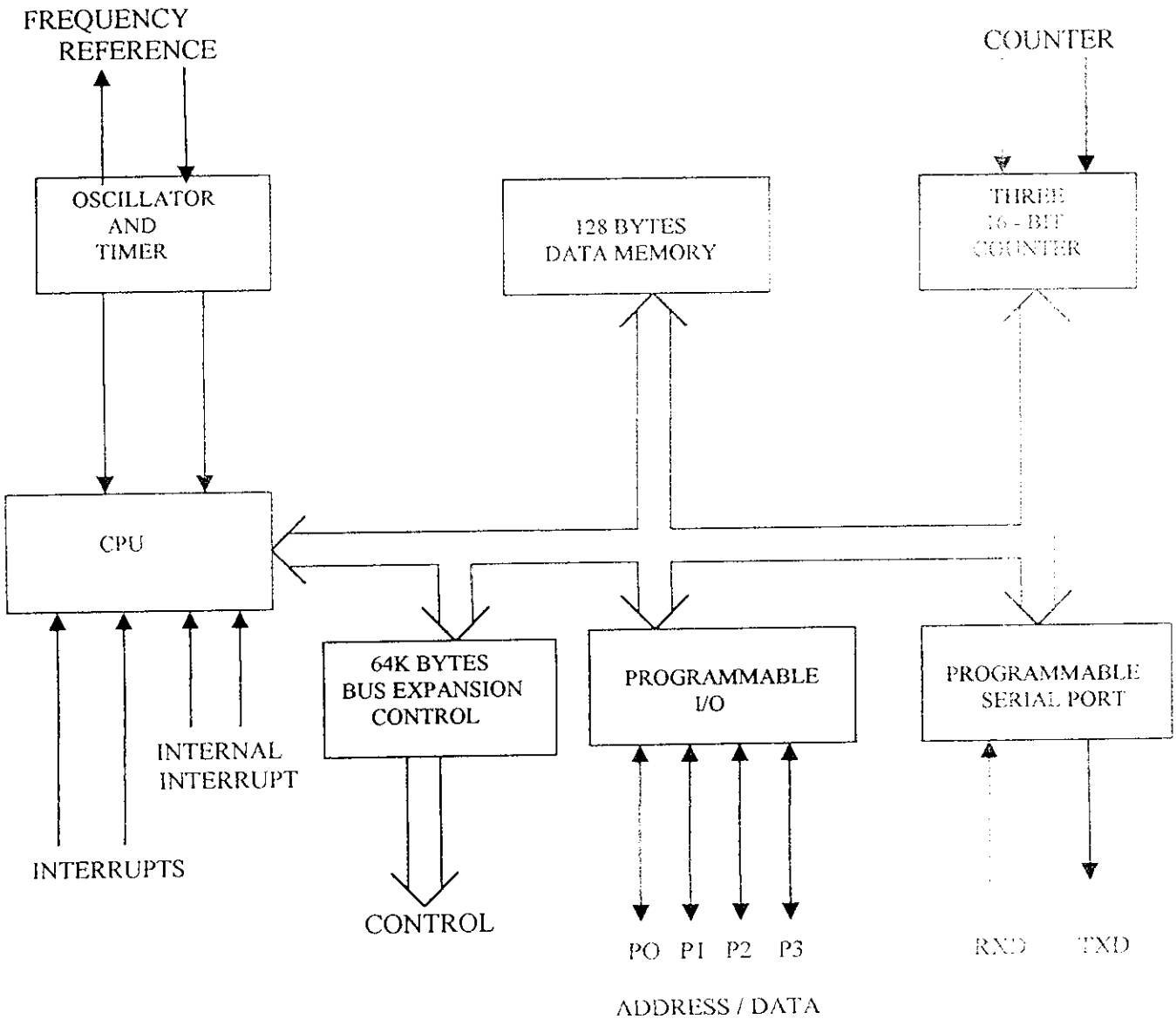


FIG 3.1 INTERNAL ARCHITECTURE OF 8032

CHAPTER - 4

DIGITAL CARD DESCRIPTION

4.1 INTRODUCTION :

The Digital card used for serial communication is previously used by the industry for the same purpose. The main part of the digital card is a 80C196 microcontroller supported by two EPROMs (odd and even), two RAMs and gain latches.

4.2 80C196 DESCRIPTION :

The 80C196 is a 16-bit microcontroller. There are five peripherals on the 80C196KB: The Pulse Width Modulated output (PWM), timers, high speed I/O units, serial port and A/D converter. Out of the five peripherals serial port and A/D converter are used here.

SUPPLY — Single 5V operation with power down mode for saving internal RAM memory.

MEMORY SPACE — 64K Bytes of memory space is available for program or data memory. Locations 0000H through 00FFH and 1FFFH through 2080H are used for special purposes. All other locations can be used for either

program or data storage or for memory mapped peripherals.

INTERNAL TIMING Internal operation of the 80C196 KB is based on the crystal or external oscillator frequency divided by 2 .Every 2 oscillator periods is referred to as a state time, the basic time measurement for all 80C196 KB. With a 12MHZ oscillator the state time is 167 ns .

PORTS Five 8-bit I/O ports . Port0 is an input port also used as analog input for A/D convertor . Port1 and Port2 are quasi-bidirectional, input and output ports . Port3 and Port4 are open drain bidirectional ports .

RALU Most calculations performed by 80C196 takes place in RALU. The RALU contains a 17-bit ALU , the program status word (PSW) , the program counter (PC) , a loop counter and three temporary registers . All of these registers are 16-bit or 17-bit wide . Some of the registers have the ability to perform simple operations to off-load the ALU .

The internal architecture of 80C196 microcontroller is shown in Fig .4.1

SYSTEM BUS:

There are several modes of system bus operation on the 80C196KB.The standard bus mode uses a 16-bit multiplexed address/data bus. Other bus modes include an 8-bit mode and a mode in which the bus size can dynamically be switched between 8- bits and 16-bits.

PROGRAM STATUS WORD:

PSW is a collection of Boolean flag which retain information concerning the state of the user's program. There are 2 bytes in the PSW: the actual status word and the low byte of the interrupt mask. PSW can be saved in the system stack with a single instruction PUSH F and restore in a POP F instruction. Only the interrupt section of the PSW can be accessed directly. There is no SFR for the PSW status bits.

MSB LSB

Z	N	V	VT	C	X	I	ST

Z Zero flag

N Negative flag

V	Overflow flag
VT	Overflow trap flag
C	Carry flag
X	Reserved
I	Interrupt disable bit
ST	Set to indicate that during a right shift a 1 has been shifted first into the carry flag.

INTERRUPTS:

80C196 microcontroller supports 28 sources of interrupts. These sources are gathered into 15 vectors plus special vectors NMI,TRAP and UNIMPLEMENTED opcodes.

NMI :

The external NMI pin generates an unmaskable interrupt for implementation of critical interrupt routines.

TRAP:

It is useful in the development of custom software debuggers or generation of software interrupts. The TRAP instruction prevents the acknowledgement of interrupts until after execution of the next instruction.

UNIMPLEMENTED OPCODE:

It generates an interrupt when unimplemented opcodes are executed. This provides software recovery from random execution during hardware and software failures.

4.3 SERIAL PORT:

Serial port on 80C196KB has one synchronous and three asynchronous modes.

Serial Port Modes :

Mode 0: Mode 0 is a synchronous mode which is commonly used for shift register based I/O expansion. In this mode the TXD pin outputs a set of 8 pulses while the RXD pin either transmits or receives data. Data is transferred 8 bits at a time with LSB first. In this mode the serial port expands the I/O capability of the 80C196 KB by simply adding shift registers.

Mode 1: Mode1 is the standard asynchronous communication mode. The data frame used in this mode consists of 10 bits; a start bit (0),8 data bits(LSB first),and a stop bit(1),if parity is enabled by setting SP_CON.2, and even parity bit is sent instead of the 8th data bit and parity is checked on reception.

Mode 2: Mode2 is the asynchronous 9^{th} bit recognition mode .This mode is commonly used with mode3 for multiprocessor communications. The transmission frame in this mode consists of 11 bits; one start bit , 9 data bit and one stop bit.9^{th} bit in data is a programmable one.

Mode 3: It is the asynchronous 9 bit mode. The data frame of this mode is same as that of mode 2. The transmission difference between mode2 and mode3 is that parity can be enabled (PEN=1) and cause the 9^{th} data bit to take even parity value . The TB 8 bit can still be used if parity is not enabled (PEN=0). In mode3 a reception always causes an interrupt regardless of the state of the 9^{th} bit. 9^{th} bit is stored if PEN=0 and can be read in bit RB8. If PEN=1 then RB8 becomes receive parity error flag.

4.4 A/D CONVERTER

Analog inputs to the 80C196KB system are handled by the A/D converter system. The system has an 8-channel multiplexer , a sample and hold, and a 10-bit successive approximation A/D converter. Conversions can performed on one of 8 channels , the input of which share with port 0 . A conversion can be done in 91 state times.

The A/D converter can cause an interrupt on completing the conversion. So by using this interrupt we can read the result in the interrupt service routine if required. Otherwise we can read the result using a polling method.

4.5 EXTERNAL MEMORY:

Using EPROMs & RAMs:

Since 80C196 is a 16-bit microcontroller it needs two 8-bit EPROMs to perform word size operations. One EPROMs is called the Odd EPROM and the other is called the Even EPROM. The 8-bits of Odd EPROM is used as MSB of a word. The 8-bits of Even EPROM and 8-bits of Odd EPROM are together parallelly fed into the microcontroller for processing. Similarly two RAMs are used, one for 8-bits (MSB) and other for rest 8-bits (LSB) storage.

4.6 GAIN LATCHES:

There are five gain latches used here. Each latch has eight output lines. For one channel five output lines are used. When the input for a particular channel is given the five output lines of the corresponding channel are latched and the gain input to the channel is set. The block diagram for the digital card is shown in figure.4.1.

4.7 DIGITAL TO ANALOG CONVERTER :

The inbuilt DAC in 80C196 can be used only for six channels and can be operated in unipolar mode alone. But we need eight output lines of DAC for eight channels. Therefore we go in for external DACs which can be operated in bipolar mode. The DACs used here are voltage output DACs.

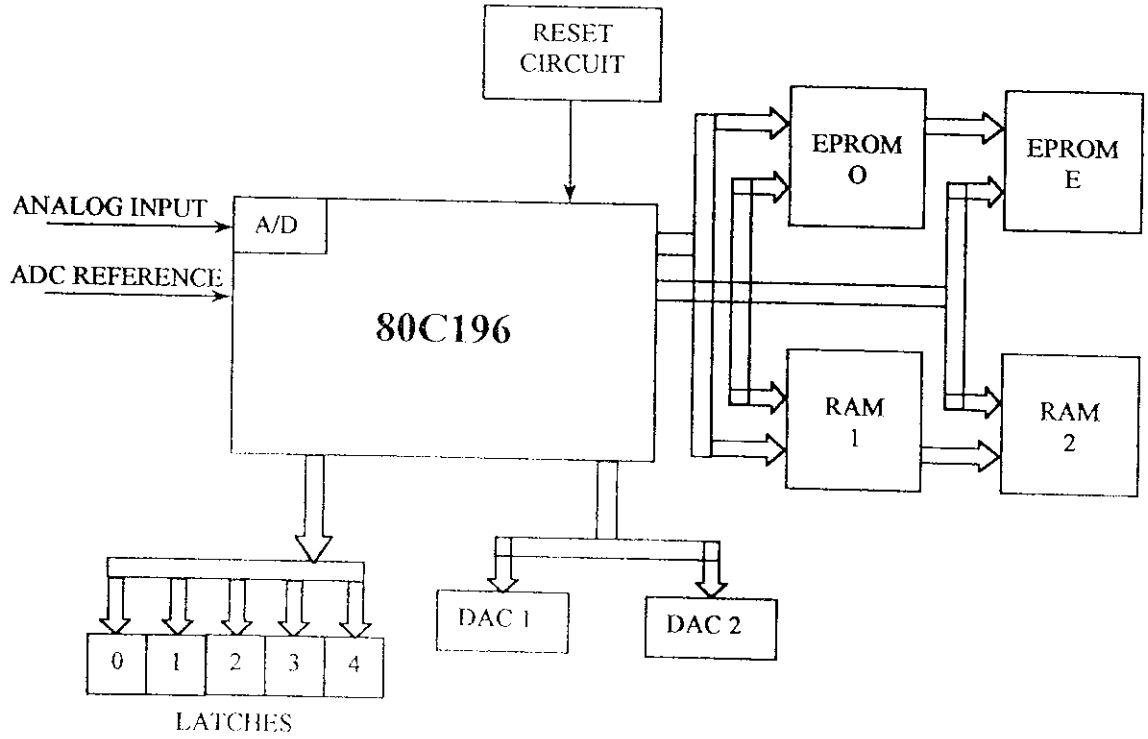


FIG 1 BLOCK DIAGRAM OF DIGITAL CARD

CHAPTER - 5

SOFTWARE

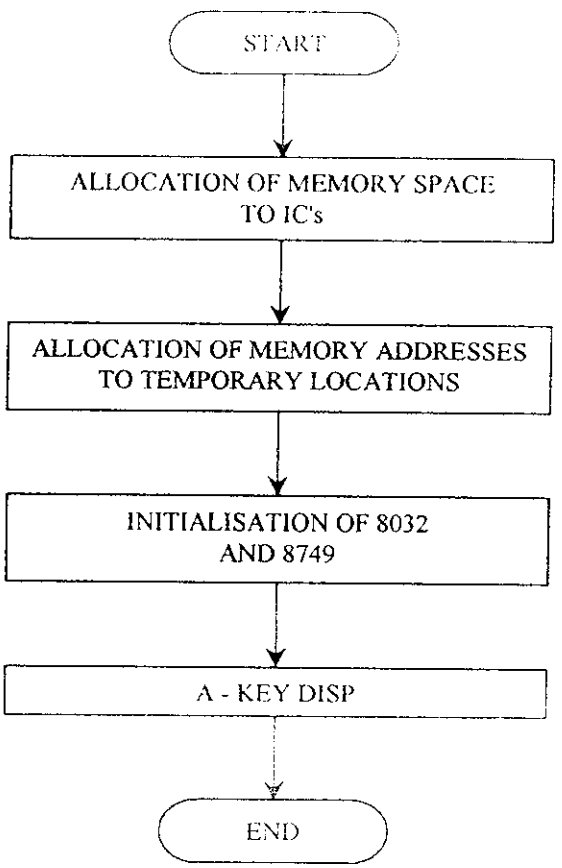
5.1 INTRODUCTION:

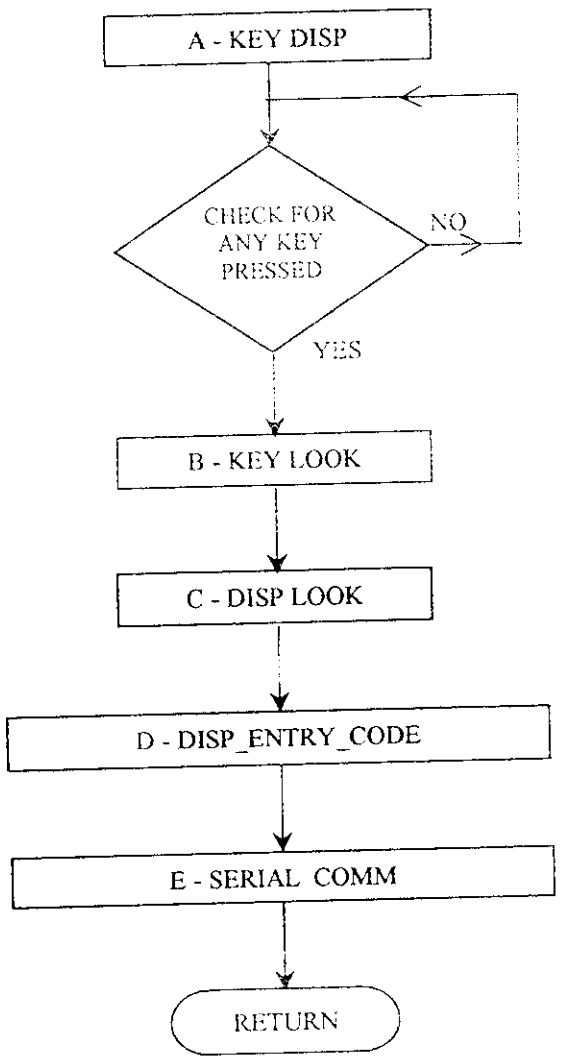
This software is essential to initiate the various processes that are necessary for testing the PCB. An efficient assembly level program is written which makes the hardware function works successfully. Program memory allocated externally to the microcontroller has a memory capacity of 64KB and it is used to store the program to be executed.

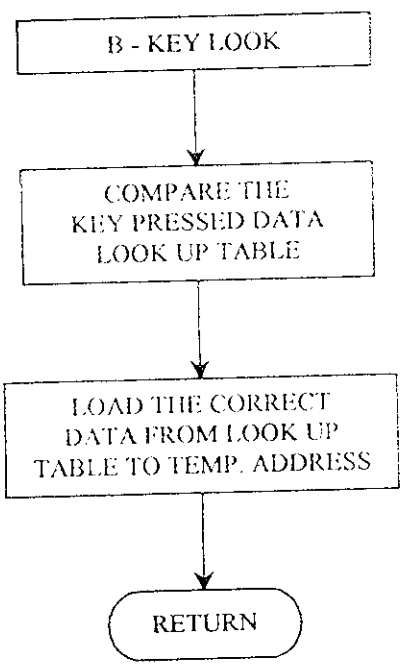
5.2 ALGORITHM:

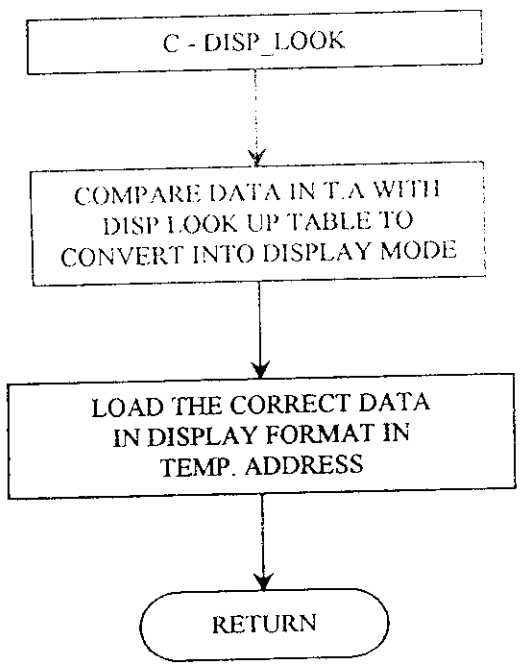
1. Memory allocation to various chips are done.
2. The 8032 Microcontroller and 8279 keyboard display interface are initialized.
3. Check whether any key is pressed using the program status word. If pressed , read the data else wait until any key is pressed.
4. The character read corresponds to the position of the key in the keyboard along with the status of the control and shift lines. Control and shift lines are always maintained high. Out of the six remaining bits, three bits indicate the row of the pressed key and three bits indicate the column. This data format is converted into original form by the keyboard lookup tables.

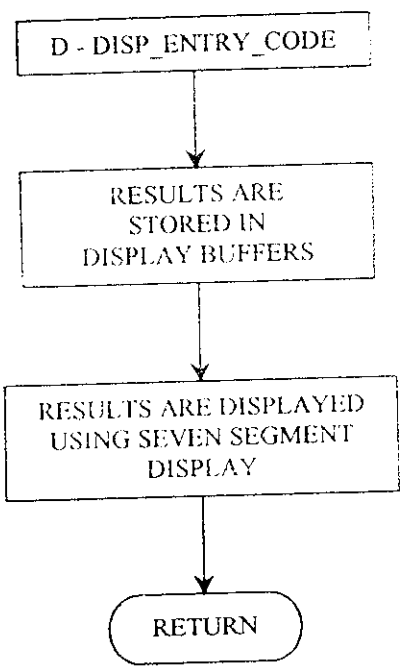
5. The data recorded is currently displayed and simultaneously transferred to the digital board through serial input/output port.
6. The datas from 80C196 Microcontroller are received after the completion of processing by the 80C32 Microcontroller.
7. The data received is converted to display format from ASCII code.
8. The result is displayed in the seven segment display as PASS or FAIL.

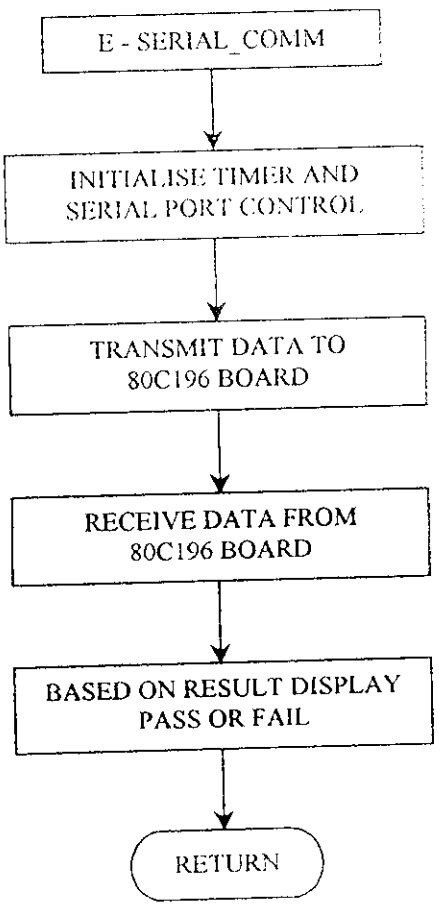

START
ALLOCATION OF MEMORY SPACE TO IC's
ALLOCATION OF MEMORY ADDRESSES TO TEMPORARY LOCATIONS
INITIALISATION OF 8032 AND 8749
A - KEY DISP
END


A - KEY DISP
CHECK FOR ANY KEY PRESSED
NO
YES
B - KEY LOOK
C - DISP LOOK
D - DISP_ENTRY_CODE
E - SERIAL COMM
RETURN


B - KEY LOOK
COMPARE THE
KEY PRESSED DATA
LOOK UP TABLE
LOAD THE CORRECT
DATA FROM LOOK UP
TABLE TO TEMP. ADDRESS
RETURN


C - DISP_LOOK
COMPARE DATA IN T.A WITH DISP LOOK UP TABLE TO CONVERT INTO DISPLAY MODE
LOAD THE CORRECT DATA IN DISPLAY FORMAT IN TEMP. ADDRESS
RETURN


D - DISP_ENTRY_CODE
RESULTS ARE STORED IN DISPLAY BUFFERS
RESULTS ARE DISPLAYED USING SEVEN SEGMENT DISPLAY
RETURN


E - SERIAL_COMM
INITIALISE TIMER AND SERIAL PORT CONTROL
TRANSMIT DATA TO 80C196 BOARD
RECEIVE DATA FROM 80C196 BOARD
BASED ON RESULT DISPLAY PASS OR FAIL
RETURN

```
;*****************************************************************
;HARDWARE SPECIFICATIONS AND DESIGN DETAILS

;*****************************************************************

;EPROM    64KB    PROGRAM MEMORY    MAPPING    00000H  TO  CFFFFH
;SRAM     32KB    EXTERNL MEMORY    MAPPING    00000H  TO  07FFFH
;NVRAM    08KB    EXTERNL MEMORY    MAPPING    08000H  TO  09FFFH
;8279     ----      --DO--          MAPPING    0E400H  TO  0E5FFH
;PORT     P1.2    DESIGNATED FOR WATCH DOG PULSE OUTPUT
                                                ;HIGHEST PRIORITY


;*****************************************************************
;INTERNAL RAM LOCATIONS FOR BYTE AND WORD VARIABLES

;*****************************************************************

OP_0            EQU     033H    ;STARTING DISPLAY
                                ;LOCATION-1st DIGIT
OP_1            EQU     034H    ;2nd DIGIT
OP_2            EQU     035H    ;3rd DIGIT
OP_3            EQU     036H    ;4th DIGIT
OP_4            EQU     037H    ;5th DIGIT
OP_5            EQU     038H    ;6th DIGIT
TMP_0           EQU     039H    ;TEMPORARY RAM
                                ;LOCATIONS
TMP_1           EQU     03AH
DBUF1_IADR      EQU     040H
DBUF2_IADR      EQU     041H
DBUF3_IADR      EQU     042H
DBUF4_IADR      EQU     043H
DBUF5_IADR      EQU     044H
TEMP0_ADDR      EQU     045H
BLUE0_ADDR      EQU     046H
BLUE1_ADDR      EQU     047H
BLUE2_ADDR      EQU     048H
BLUE3_ADDR      EQU     049H
BLUE4_ADDR      EQU     04AH
BLUE5_ADDR      EQU     04BH
BLUE6_ADDR      EQU     04CH

;*****************************************************************
; CONSTANT DATA DECLARATIONS

;*****************************************************************

BAUD_FACTOR     EQU     0F3H    ;TIMER 1 LOADING VALUE
                                ;FOR 2400 BAUD RATE
TIMER0_LO_COUNT EQU     0B0H    ;TIMER0 50 THOUSAND
                                ;MICRO SECONDS FACTOR
```

```
TIMER0_HI_COUNT     EQU         03CH    ;TIMER0 50 THOUSAND
                                        ;MICRO SECONDS FACTOR
TIMER2_LO_COUNT     EQU         030H    ;TIMER 2 FOR ONE MINUTE
                                        ;TIME COUNTER
TIMER2_HI_COUNT     EQU         03CH    ;TIMER 2 FOR ONE MINUTE
                                        ;TIME COUNTER
BLANK               EQU         0FFH
BLANK_CODE          EQU         020H    ;DATA FOR BLANKING
                                        ;THE DISPLAY
ZERO_CODE           EQU         000H
PCON                DATA        087H

;*************************************************************
;PERIPHERAL'S LOCATION DEFINITIONS

;*************************************************************

CMND_8279   EQU         0E401H  ;COMMAND & CONTRL REG. ADDRESS
DATA_8279   EQU         0E400H  ;DATA REG. ADDRESS
;*************************************************************

;&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
;COMMANDS FOR 8279;

;&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

KEY_DISP_MODE       EQU     000H
PGM_CLOCK           EQU     03AH
FIFO_RAM            EQU     050H
DISP_RAM_READ       EQU     070H
DISP_RAM_WRITE      EQU     090H
CLEAR_DISP          EQU     0DFH

;&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&

KEY_TBLADR                      EQU                 4100H

KEY_TBLCODE                     EQU                 4120H

KY_RTRNADDR                     EQU                 4151H

DISP_TBLADR                     EQU                 4150H

;^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

        ORG     00H
        LJMP    START

        ORG     1BH
```

```
        RETI

        ORG     23H
        RETI

        ORG     100H

;*****************************************************************
;starting location of the program

;*****************************************************************

START:
        LCALL   BLANK_DISPLAY
        MOV     OP_0,#0FFH
        MOV     OP_1,#0FFH
        MOV     OP_2,#0FFH
        MOV     OP_3,#0FFH
        MOV     OP_4,#0FFH
        MOV     OP_5,#0FFH
        CALL    INIT_8279
        MOV     R2,#06H
        CPL     P1.2

LOO:    CALL    KEY_DISP
        CPL     P1.2
        CJNE    R2,#00H,LOO
        MOV     R2,#06H
        MOV     OP_0,#0FFH
        MOV     OP_1,#0FFH
        MOV     OP_2,#0FFH
        CPL     P1.2
        MOV     OP_3,#0FFH
        MOV     OP_4,#0FFH
        MOV     OP_5,#0FFH
        CPL     P1.2
        JMP     LOO

;*****************************************************************
KEY_DISP:

;*****************************************************************

        CPL     P1.2
        MOV     DPTR,#CMND_8279
        MOVX    A,@DPTR
        JNB     ACC.4,RD_FIFO
        CPL     P1. 2
RTRN:
        RET
```

```
RD_FIFO:
        ANL     A,#07H
        CPL     P1.2
        JZ      RTRN
        MOV     DPTR,#CMND_8279
        MOV     A,#040H
        MOVX    @DPTR,A
        MOV     DPTR,#DATA_8279
        CPL     P1.2
        MOVX    A,@DPTR
        MOV     TEMP0_ADDR,A
        LCALL   KEY_LOOK
        CALL    DISP_LOOK
        CALL    DISP_ENTRY_CODE
        CPL     P1.2
        RET

KEY_LOOK:

;*****************************************************************
; THIS SUBROUTINE COMPARES THE FIFO DATA WITH THE LOOKUP
;TABLE AND RETURN THE DATA IN KEY_RTRNADDR LOCATION.IF THE
;VALUE NOT AVAILABLE IN LOOK_UP TABLE THEN RETURNS DATA 0FFH

;*****************************************************************
        CPL     P1.2
        MOV     DPTR,#KEY_TBLADR
DO_CMPR:
        MOV     A,#00H
        MOVC    A,@A+DPTR
        CJNE    A,TEMP0_ADDR,NEXT_CMPR
        CPL     P1.2
        MOV     A,#020H
        MOVC    A,@A+DPTR
        MOV     DPTR,#KY_RTRNADDR
        MOVX    @DPTR,A
        MOV     TEMP0_ADDR,A
        CJNE    R2,#06H,AL1
        MOV     BLUE1_ADDR,TEMP0_ADDR
        CPL     P1.2
        JMP     AL6
AL1:    CJNE    R2,#05H,AL2
        MOV     BLUE2_ADDR,TEMP0_ADDR
        JMP     AL6
        CPL     P1.2
AL2:    CJNE    R2,#04H,AL3
        MOV     BLUE3_ADDR,TEMP0_ADDR
        JMP     AL6
AL3:    CJNE    R2,#03H,AL4
        MOV     BLUE4_ADDR,TEMP0_ADDR
        CPL     P1.2
        JMP     AL6
```

```
AL4:    CJNE    R2,#02H,AL5
        MOV     BLUE5_ADDR,TEMP0_ADDR
        JMP     AL6
AL5:    CJNE    R2,#01H,AL6
        CPL     P1.2
        MOV     BLUE6_ADDR,TEMP0_ADDR
AL6:    RET

NEXT_CMPR:
        INC     DPTR
        MOV     A,#00H
        CPL     P1.2
        MOVC    A,@A+DPTR
        CJNE    A,#0FFH,DO_CMPR
        MOV     DPTR,#KY_RTRNADDR
        MOVX    @DPTR,A
        CPL     P1.2
        RET

;*****************************************************************
DISP_LOOK:

;*****************************************************************
;COMPARE THE DATA AVAILABLE IN TEMP0_ADDR WITH DISPLAY LOOK
;UP TABLE AND RETURNS THE CORRESPONDING SEGMENT DATA IN
;TEMP0_ADDR

;*****************************************************************

        CPL     P1.2
        MOV     R3,#ZERO_CODE       ;START WITH 000H FOR
                                    ;COMPARING
        MOV     DPTR,#DISP_TBLADR   ;LOOK UP TABLE STARTING
                                    ;ADDRESS

DO_LOOK:
        MOV     A,TEMP0_ADDR
        CJNE    A,#20H,NO_BLANK
        CPL     P1.2
        MOV     A,#BLANK
        MOV     TEMP0_ADDR,A
        RET

NO_BLANK:
        MOV     A,TEMP0_ADDR
        ANL     A,#3FH
        CPL     P1.2
        CLR     C
        SUBB    A,R3
        JNZ     NEXT_LOOK           ;IF NOT EQUAL TO THEN
                                    ;INCREMENT R3 AND CHECK
                                    ;FOR 009H
```

```
            MOV   A,#ZERO_CODE   ;OFFSET FOR DISP_TBLADR
                                  ;LOCATION
            MOVC  A,@A+DPTR      ;GET THE SEGMENT DATA FROM
                                  ;DISPLAY TABLE
            MOV   TEMP0_ADDR,A
            CPL   P1.2
            CJNE  R2,#005H,RT1
            MOV   TMP_0,TEMP0_ADDR
RT1:        RET
NEXT_LOOK:
            INC   DPTR                    ;NEXT DISPLAY SEGMENT
            CPL   P1.2
            INC   R3                      ;R3 = R3 + 1 UP TO 027H
            CJNE  R3,#027H,DO_LOOK        ;IF R3=0AH THEN RETURN
                                          ;BLANK IN TEMP0_ADDR
            MOV   TEMP0_ADDR,#BLANK
            RET


;*****************************************************************
DISP_ENTRY_CODE:

;*****************************************************************
;DISPLAY THE CORRESPONDING PRESSED KEY

;*****************************************************************

            CPL     P1.2
            CJNE    R2,#006H,K1
            MOV     OP_0,TEMP0_ADDR
            JMP     K6
K1:         CJNE    R2,#005H,K2
            CPL     P1.2
            MOV     OP_1,TEMP0_ADDR
            JMP     K6
K2:         CJNE    R2,#004H,K3
            MOV     OP_2,TEMP0_ADDR
            CPL     P1.2
            JMP     K6
K3:         CJNE    R2,#003H,K4
            MOV     OP_3,TEMP0_ADDR
            CPL     P1.2
            JMP     K6
K4:         CJNE    R2,#002H,K5
            MOV     OP_4,TEMP0_ADDR
            CPL     P1.2
            JMP     K6
K5:         CJNE    R2,#001H,K6
            MOV     OP_5,TEMP0_ADDR
            CPL     P1.2
```

```
K6:     LCALL   OUT_SEGMENT
        DJNZ    R2,LOOP
        LCALL   SERIAL_COMM
        CPL     P1.2
LOOP:   RET

;*****************************************************************
OUT_SEGMENT:

;*****************************************************************
; THIS SUBROUTINE DISPLAYS 5 DIGIT DATA AVAILABLE IN
; DISPLAY BUFFER ADDRESS LOCATIONS TO 8279 & INITIALISE
; WRITE DISPLAY RAM ADDRESS OF 8279

;*****************************************************************

        CPL     P1.2
        MOV     DPTR,#CMND_8279
        MOV     A,#090H
        MOVX    @DPTR,A
        MOV     DPTR,#DATA_8279
        CPL     P1.2
        MOV     A,OP_0
        MOVX    @DPTR,A
        MOV     A,OP_1
        MOVX    @DPTR,A
        MOV     A,OP_2
        CPL     P1.2
        MOVX    @DPTR,A
        MOV     A,OP_3
        MOVX    @DPTR,A
        MOV     A,OP_4
        MOVX    @DPTR,A
        CPL     P1.2
        MOV     A,OP_5
        MOVX    @DPTR,A
        RET

;*****************************************************************
ORG KEY_TBLADR

;*****************************************************************

        DB    0C3H,0C2H,0CAH,0D2H,0C1H,0C9H,0D1H ;0,1,2,3,4,5,6
        DB    0C4H,0CCH,0D4H,0CBH,0D3H,0DAH,0DBH
                                          ;7,8,9,DP,ENTER,P1,P2
        DB    0FFH                              ;BLANK
```

```
;*****************************************************************
ORG KEY_TBLCODE

;*****************************************************************

        DB    000H,001H,002H,003H,004H,005H,006H ;0,1,2,3,4,5,6
        DB    007H,008H,009H,013H,011H,017H,026H
                                                  ;7,8,9,DP,ENTER,F1,F2

;*****************************************************************
ORG DISP_TBLADR

;*****************************************************************

        DB    0C0H        ;7 SEGMENT CODE FOR '0';00
        DB    0F9H        ;7 SEGMENT CODE FOR '1';01
        DB    0A4H        ;7 SEGMENT CODE FOR '2';02
        DB    0B0H        ;7 SEGMENT CODE FOR '3';03
        DB    099H        ;7 SEGMENT CODE FOR '4';04
        DB    092H        ;7 SEGMENT CODE FOR '5';05
        DB    082H        ;7 SEGMENT CODE FOR '6';06
        DB    0F8H        ;7 SEGMENT CODE FOR '7';07
        DB    080H        ;7 SEGMENT CODE FOR '8';08
        DB    090H        ;7 SEGMENT CODE FOR '9';09
        DB    0C6H        ;7 SEGMENT CODE FOR 'C';0A
        DB    088H        ;7 SEGMENT CODE FOR 'A';0B ;ENTER
        DB    0C2H        ;7 SEGMENT CODE FOR 'G';0C
        DB    0C1H        ;7 SEGMENT CODE FOR 'V';0D
        DB    086H        ;7 SEGMENT CODE FOR 'E';0E
        DB    08EH        ;7 SEGMENT CODE FOR 'F';0F
        DB    083H        ;7 SEGMENT CODE FOR 'B';10
        DB    088H        ;7 SEGMENT CODE FOR 'A';11
        DB    0F9H        ;7 SEGMENT CODE FOR 'I';12
        DB    0C6H        ;7 SEGMENT CODE FOR 'C';13
        DB    0C7H        ;7 SEGMENT CODE FOR 'L';14
        DB    0EAH        ;7 SEGMENT CODE FOR 'M';15
        DB    0ABH        ;7 SEGMENT CODE FOR 'N';16
        DB    0C2H        ;7 SEGMENT CODE FOR 'G';17
        DB    08CH        ;7 SEGMENT CODE FOR 'P';18
        DB    0AFH        ;7 SEGMENT CODE FOR 'R';19
        DB    092H        ;7 SEGMENT CODE FOR 'S';1A
        DB    087H        ;7 SEGMENT CODE FOR 'T';1B
        DB    0C1H        ;7 SEGMENT CODE FOR 'U';1C
        DB    0A1H        ;7 SEGMENT CODE FOR 'V';1D
        DB    0E2H        ;7 SEGMENT CODE FOR 'W';1E
        DB    091H        ;7 SEGMENT CODE FOR 'Y';1F
        DB    0FFH        ;7 SEGMENT CODE FOR ' ';20 FOR SPACE
        DB    0F7H        ;7 SEGMENT CODE FOR '_';21 FOR HYPHEN
        DB    0C7H        ;7 SEGMENT CODE FOR 'L';22
        DB    0EAH        ;7 SEGMENT CODE FOR 'M';23
        DB    0ABH        ;7 SEGMENT CODE FOR 'N';24
```

```
        DB    0C2H        ;7 SEGMENT CODE FOR 'G';25
        DB    0C1H        ;7 SEGMENT CODE FOR 'V';26

;*****************************************************************
SERIAL_COMM:

;*****************************************************************

                MOV     SP,#60H
                MOV     PSW,#00H
                CPL     P1.2
                MOV     TMOD,#020H
                MOV     SCON,#050H
                MOV     PCON,#080H
                CPL     P1.2
                MOV     TL1,#0F3H
                MOV     TH1,#0F3H
                CPL     P1.2
                CLR     TI
                CLR     RI
                SETB    TR1
                SETB    EA

                CPL     P1.2
                MOV     A,BLUE1_ADDR
                ADD     A,#030H
                MOV     SBUF,A
        L1:
                CPL     P1.2
                JNB     TI,L1
                CLR     TI
                CPL     P1.2
                MOV     A,BLUE2_ADDR
                ADD     A,#030H
                MOV     SBUF,A
        L2:
                CPL     P1.2
                JNB     TI,L2
                CLR     TI
                CPL     P1.2
                MOV     A,BLUE3_ADDR
                ADD     A,#030H
                MOV     SBUF,A
        L3:
                CPL     P1.2
                JNB     TI,L3
                CLR     TI
                CPL     P1.2
                MOV     A,BLUE4_ADDR
                ADD     A,#030H
                MOV     SBUF,A
```

```
L4:
        CPL     P1.2
        JNB     TI,L4
        CLR     TI
        CPL     P1.2
        MOV     A,BLUE5_ADDR
        ADD     A,#030H
        MOV     SBUF,A
L5:
        CPL     P1.2
        JNB     TI,L5
        CLR     TI
        CPL     P1.2
        MOV     A,BLUE6_ADDR
        ADD     A,#030H
        MOV     SBUF,A

L6:     CPL     P1.2
        JNB     TI,L6
        CLR     TI

L9:     CPL     P1.2
        JNB     RI,L9
        MOV     A,SBUF
        CLR     RI
        CPL     P1.2
        CJNE    A,#031H,L10
        CALL    DISP_PASS
        RET
L10:
        CPL     P1.2
        CJNE    A,#032H,L11
        CALL    DISP_FAIL
L11:     RET

DISP_PASS:
        CPL     P1.2
        MOV     DBUF1_IADR,#08CH
        MOV     DBUF2_IADR,#088H
        MOV     DBUF3_IADR,#092H
        MOV     DBUF4_IADR,#092H
        MOV     DBUF5_IADR,TMP_0
        CPL     P1.2
        MOV     BLUE0_ADDR,#0C6H
        LCALL   SEGMENT_DISP
        RET
```

```
DISP_FAIL:
        CPL     P1.2
        MOV     DBUF1_IADR,#08EH
        MOV     DBUF2_IADR,#088H
        MOV     DBUF3_IADR,#0F9H
        MOV     DBUF4_IADR,#0C7H
        CPL     P1.2
        MOV     DBUF5_IADR,TMP_0
        MOV     BLUE0_ADDR,#0C6H
        LCALL   SEGMENT_DISP
        RET

SEGMENT_DISP:
        CPL     P1.2
        MOV     DPTR,#CMND_8279
        MOV     A,#090H
        MOVX    @DPTR,A
        CPL     P1.2
        MOV     DPTR,#DATA_8279
        MOV     A,BLUE0_ADDR
        MOVX    @DPTR,A
        MOV     A,DBUF5_IADR
        MOVX    @DPTR,A
        CPL     P1.2
        MOV     A,DBUF1_IADR
        MOVX    @DPTR,A
        MOV     A,DBUF2_IADR
        MOVX    @DPTR,A
        CPL     P1.2
        MOV     A,DBUF3_IADR
        MOVX    @DPTR,A
        MOV     A,DBUF4_IADR
        MOVX    @DPTR,A
        CPL     P1.2
        RET

;*****************************************************************
INIT_8279:

;*****************************************************************
;This is to initialise the 8279 for 8char,right entry,encoded
;key board mode

;*****************************************************************

        MOV     DPTR,#CMND_8279
        MOV     A,#000H  ;KBRD - DISP MODE
        MOVX    @DPTR,A
        CPL     P1.2
        MOV     A,#3AH          ;PRGM CLOCK
        MOVX    @DPTR,A
        MOV     A,#50H          ;FIFO RAM
```

```
                MOVX    @DPTR,A
                CPL     P1.2
                MOV     A,#70H          ;READ DISPLAY RAM ADDRESS
                MOVX    @DPTR,A
                CPL     P1.2
                MOV     A,#90H          ;WRITE DISPLAY RAM ADDRESS
                MOVX    @DPTR,A
                CPL     P1.2
                MOV     A,#0DFH         ;CLEAR MODE
                MOVX    @DPTR,A
                MOV     R7,#0FFH
        DDD   :
                CPL     P1.2
                DJNZ    R7,DDD
                MOV     R7,#06H
                MOV     DPTR,#DATA_8279 ;purposely blank the
                                        ;displays
                MOV     A,#0FFH
        DO_OUT:
                MOVX    @DPTR,A
                DJNZ    R7,DO_OUT
                RET

;*****************************************************************

;*****************************************************************
BLANK_DISPLAY:

;*****************************************************************
;THIS ROUTINE FOR BLANK ALL DIGITS

;*****************************************************************

                MOV     DPTR,#CMND_8279
                MOV     A,#090H
                MOVX    @DPTR,A
                MOV     DPTR,#DATA_8279
                MOV     A,#0FFH
                MOVX    @DPTR,A
                MOVX    @DPTR,A
                MOVX    @DPTR,A
                MOVX    @DPTR,A
                MOVX    @DPTR,A
                MOVX    @DPTR,A
                RET

;*****************************************************************
 END
```

CHAPTER - 6

CONCLUSION

In this project a microcontroller based Analog signal tester, to test a PCB of the Yarn Quality Monitor has been designed and fabricated. This Analog Signal Tester is employed in the industry for regular use. The tester is very fast and accurate in its performance. This method has considerably decreased the time span required for the entire testing sequence in to few minutes and there by conserves the precious of time.

The visual display reduces the fatigue of the worker and makes his work easy.

FURTHER DEVELOPMENT :

- As this project tests the entire channel at a time, this can be improved to test each and every component individually.
- A testing method which interfaces the PCB to be tested to the personal computer can be developed.

REFERENCES

1. D.Roy Choudhury, Shail Jain, "Linear Integrated Circuits", Wiley Eastern Ltd., New Age International Ltd., New Delhi, 1984.

2. "Intel 8-bit Embedded Controller Handbook ",1986.

3. Ramesh .S. Goankar, " Microprocessor Architecture, Programming and applications", Wiley Eastern Ltd., New Age International Ltd., New Delhi, 2nd Edition, 1987.

4. "National Semiconductors Data Acquisition Data Book",1993.

5. "Intel Memory Components Handbook", 1987.

6. Doughlas V.Hall, "Microprocessors and Interfacing Programming and Hardware", McGraw-Hill International Editions, Newyork, 1988.

8279/8279-5
PROGRAMMABLE KEYBOARD/DISPLAY INTERFACE

- Simultaneous Keyboard Display Operations
- Scanned Keyboard Mode
- Scanned Sensor Mode
- Strobed Input Entry Mode
- 8-Character Keyboard FIFO
- 2-Key Lockout or N-Key Rollover with Contact Debounce
- Dual 8- or 16-Numerical Display
- Single 16-Character Display
- Right or Left Entry 16-Byte Display RAM
- Mode Programmable from CPU
- Programmable Scan Timing
- Interrupt Output on Key Entry
- Available in EXPRESS
  - —Standard Temperature Range
  - —Extended Temperature Range

The Intel® 8279 is a general purpose programmable keyboard and display I/O interface device designed for use with Intel® microprocessors. The keyboard portion can provide a scanned interface to a 64-contact key matrix. The keyboard portion will also interface to an array of sensors or a strobed interface keyboard, such as the hall effect and ferrite variety. Key depressions can be 2-key lockout or N-key rollover. Keyboard entries are debounced and strobed in an 8-character FIFO. If more than 8 characters are entered, overrun status is set. Key entries set the interrupt output line to the CPU.

The display portion provides a scanned display interface for LED, incandescent, and other popular display technologies. Both numeric and alphanumeric segment displays may be used as well as simple indicators. The 8279 has 16X8 display RAM which can be organized into dual 16X4. The RAM can be loaded or interrogated by the CPU. Both right entry, calculator and left entry typewriter display formats are possible. Both read and write of the display RAM can be done with auto-increment of the display RAM address.

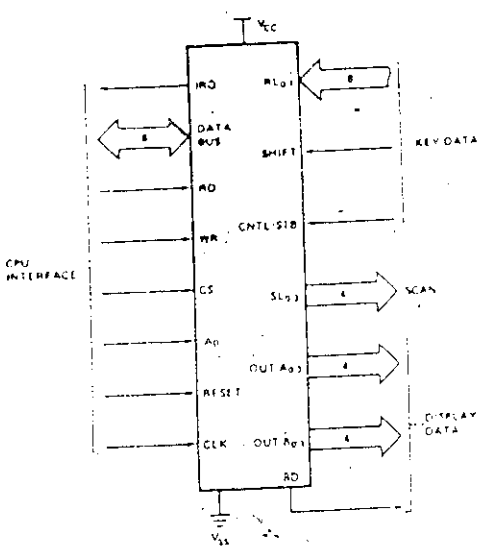


Figure 1. Logic Symbol

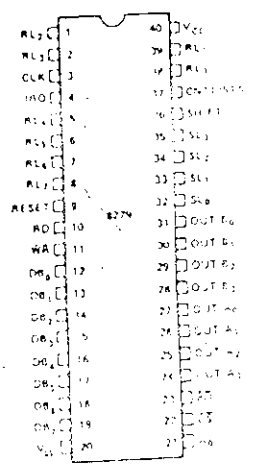


Figure 2. Pin Configuration

HARDWARE DESCRIPTION

The 8279 is packaged in a 40 pin DIP. The following is a functional description of each pin.

Table 1. Pin Descriptions

Symbol	Pin No.	Name and Function
DB_0-DB_7	8	**Bi-directional data bus:** All data and commands between the CPU and the 8279 are transmitted on these lines.
CLK	1	**Clock:** Clock from system used to generate internal timing.
RESET	1	**Reset:** A high signal on this pin resets the 8279. After being reset the 8279 is placed in the following mode: 1) 16 8-bit character display —left entry. 2) Encoded scan keyboard—2 key lockout. Along with this the program clock prescaler is set to 31.
CS	1	**Chip Select:** A low on this pin enables the interface functions to receive or transmit.
A_0	1	**Buffer Address:** A high on this line indicates the signals in or out are interpreted as a command or status. A low indicates that they are data.
$\overline{RD}$, $\overline{WR}$	2	**Input/Output Read and Write:** These signals enable the data buffers to either send data to the external bus or receive it from the external bus.
IRQ	1	**Interrupt Request:** In a keyboard mode, the interrupt line is high when there is data in the FIFO/Sensor RAM. The interrupt line goes low with each FIFO/Sensor RAM read and returns high if there is still information in the RAM. In a sensor mode, the interrupt line goes high whenever a change in a sensor is detected.
V_{SS}, V_{CC}	2	**Ground and power supply pins.**
SL_0-SL_3	4	**Scan Lines:** Scan lines which are used to scan the key switch or sensor matrix and the display digits. These lines can be either encoded (1 of 16) or decoded (1 of 4).
RL_0-RL_7	8	**Return Line:** Return line inputs which are connected to the scan lines through the keys or sensor switches. They have active internal pullups to keep them high until a switch closure pulls one low. They also serve as an 8-bit input in the Strobed Input mode.
SHIFT	1	**Shift:** The shift input status is stored along with the key position on key closure in the Scanned Keyboard modes. It has an active internal pullup to keep it high until a switch closure pulls it low.
CNTL/STB	1	**Control/Strobed Input Mode:** For keyboard modes this line is used as a control input and stored like status on a key closure. The line is also the strobe line that enters the data into the FIFO in the Strobed Input mode. (Rising Edge). It has an active internal pullup to keep it high until a switch closure pulls it low.
OUT A_0-OUT A_3 OUT B_0-OUT B_3	4 4	**Outputs:** These two ports are the outputs for the 16 x 4 display refresh registers. The data from these outputs is synchronized to the scan lines (SL_0-SL_3) for multiplexed digit displays. The two 4 bit ports may be blanked independently. These two ports may also be considered as one 8 bit port.
$\overline{BD}$	1	**Blank Display:** This output is used to blank the display during digit switching or by a display blanking command.

FUNCTIONAL DESCRIPTION

Since data input and display are an integral part of many microprocessor designs, the system designer needs an interface that can control these functions without placing a large load on the CPU. The 8279 provides this function for 8-bit microprocessors.

The 8279 has two sections: keyboard and display. The keyboard section can interface to regular typewriter style keyboards or random toggle or thumb switches. The display section drives alphanumeric displays or a bank of indicator lights. Thus the CPU is relieved from scanning the keyboard or refreshing the display.

The 8279 is designed to directly connect to the microprocessor bus. The CPU can program all operating modes for the 8279. These modes include:

Input Modes

- Scanned Keyboard — with encoded (8 x 8 key keyboard) or decoded (4 x 8 key keyboard) scan lines. A key depression generates a 6-bit encoding of key position. Position and shift and control status are stored in the FIFO. Keys are automatically debounced with 2-key lockout or N-key rollover.
- Scanned Sensor Matrix — with encoded (8 x 8 matrix switches) or decoded (4 x 8 matrix switches) scan lines. Key status (open or closed) stored in RAM addressable by CPU.
- Strobed Input — Data on return lines during control line strobe is transferred to FIFO.

Output Modes

- 8 or 16 character multiplexed displays that can be organized as dual 4-bit or single 8-bit ($B_0 = D_0$, $A_3 = D_7$).
- Right entry or left entry display formats.

Other features of the 8279 include:

- Mode programming from the CPU.
- Clock Prescaler
- Interrupt output to signal CPU when there is keyboard or sensor data available.
- An 8 byte FIFO to store keyboard information.
- 16 byte internal Display RAM for display refresh. This RAM can also be read by the CPU.

PRINCIPLES OF OPERATION

The following is a description of the major elements of the 8279 Programmable Keyboard/Display Interface device. Refer to the block diagram in Figure 3.

I/O Control and Data Buffers

The I/O control section uses the $\overline{CS}$, A_0, $\overline{RD}$ and $\overline{WR}$ lines to control data flow to and from the various internal registers and buffers. All data flow to and from the 8279 is enabled by $\overline{CS}$. The character of the information, given or desired by the CPU, is identified by A_0. A logic one means the information is a command or status. A logic zero means the information is data. $\overline{RD}$ and $\overline{WR}$ determine the direction of data flow through the Data Buffers. The Data Buffers are bi-directional buffers that connect the internal bus to the external bus. When the chip is not selected ($\overline{CS}$ = 1), the devices are in a high impedance state. The drivers input during $\overline{WR} \cdot \overline{CS}$ and output during $\overline{RD} \cdot \overline{CS}$.

Control and Timing Registers and Timing Control

These registers store the keyboard and display modes and other operating conditions programmed by the CPU. The modes are programmed by presenting the proper command on the data lines with A_0 = 1 and then sending a WR. The command is latched on the rising edge of WR.

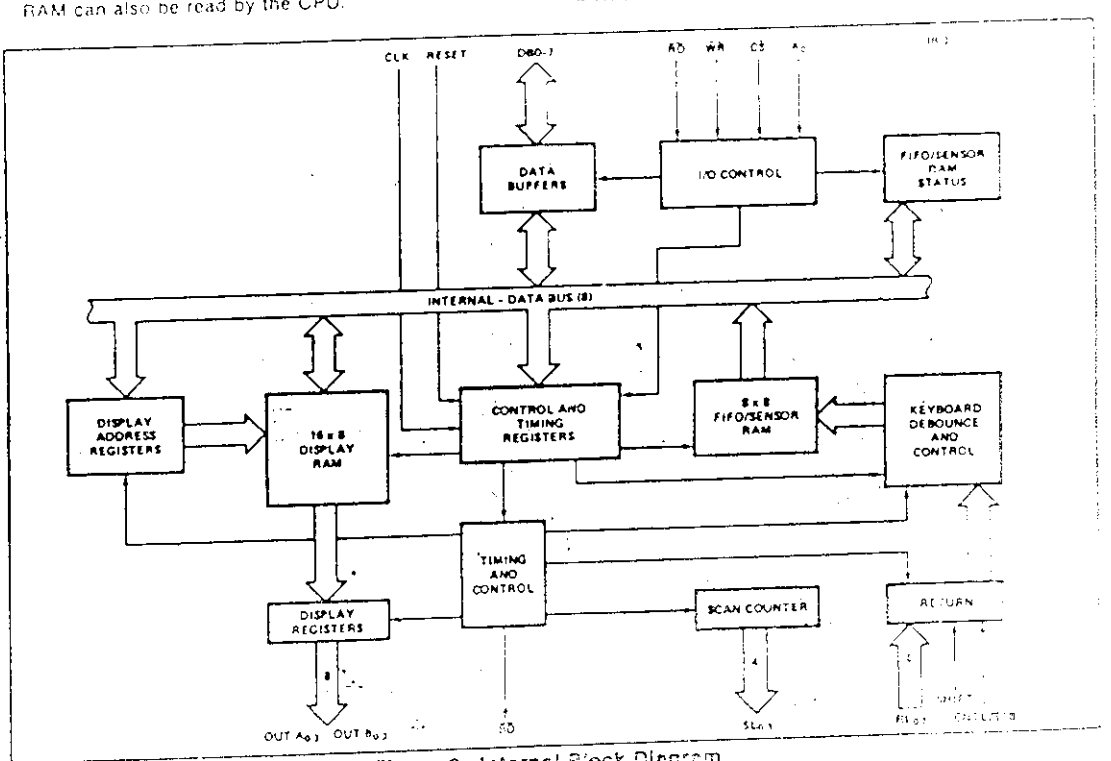


Figure 3. Internal Block Diagram

The command is then decoded and the appropriate function is set. The timing control contains the basic timing counter chain. The first counter is a ÷ N prescaler that can be programmed to yield an internal frequency of 100 kHz which gives a 5.1 ms keyboard scan time and a 10.3 ms debounce time. The other counters divide down the basic internal frequency to provide the proper key scan, row scan, keyboard matrix scan, and display scan times.

Scan Counter

The scan counter has two modes. In the encoded mode, the counter provides a binary count that must be externally decoded to provide the scan lines for the keyboard and display. In the decoded mode, the scan counter decodes the least significant 2 bits and provides a decoded 1 of 4 scan. Note than when the keyboard is in decoded scan, so is the display. This means that only the first 4 characters in the Display RAM are displayed.

In the encoded mode, the scan lines are active high outputs. In the decoded mode, the scan lines are active low outputs.

Return Buffers and Keyboard Debounce and Control

The 8 return lines are buffered and latched by the Return Buffers. In the keyboard mode, these lines are scanned, looking for key closures in that row. If the debounce circuit detects a closed switch, it waits about 10 msec to check if the switch remains closed. If it does, the address of the switch in the matrix plus the status of SHIFT and CONTROL are transferred to the FIFO. In the scanned Sensor Matrix modes, the contents of the return lines is directly transferred to the corresponding row of the Sensor RAM (FIFO) each key scan time. In Strobed Input mode, the contents of the return lines are transferred to the FIFO on the rising edge of the CNTL/STB line pulse.

FIFO/Sensor RAM and Status

This block is a dual function 8 x 8 RAM. In Keyboard or Strobed Input modes, it is a FIFO. Each new entry is written into successive RAM positions and each is then read in order of entry. FIFO status keeps track of the number of characters in the FIFO and whether it is full or empty. Too many reads or writes will be recognized as an error. The status can be read by an $\overline{RD}$ with $\overline{CS}$ low and A_0 high. The status logic also provides an IRQ signal when the FIFO is not empty. In Scanned Sensor Matrix mode, the memory is a Sensor RAM. Each row of the Sensor RAM is loaded with the status of the corresponding row of sensor in the sensor matrix. In this mode, IRQ is high if a change in a sensor is detected.

Display Address Registers and Display RAM

The Display Address Registers hold the address of the word currently being written or read by the CPU and the two 4-bit nibbles being displayed. The read/write addresses are programmed by CPU command. They also can be set to auto increment after each read or write. The Display RAM can be directly read by the CPU after the correct mode and address is set. The addresses for the A and B nibbles are automatically updated by the 8279 to match data entry by the CPU. The A and B nibbles can be entered independently or as one word, according to the mode that is set by the CPU. Data entry to the display can be set to either left or right entry. See Interface Considerations for details.

SOFTWARE OPERATION

8279 commands

The following commands program the 8279 operating modes. The commands are sent on the Data Bus with $\overline{CS}$ low and A_0 high and are loaded to the 8279 on the rising edge of $\overline{WR}$.

Keyboard/Display Mode Set

	MSB							LSB
Code:	0	0	0	D	D	K	K	K

Where DD is the Display Mode and KKK is the Keyboard Mode.

DD

0	0	8 8-bit character display — Left entry
0	1	16 8-bit character display — Left entry*
1	0	8 8-bit character display — Right entry
1	1	16 8-bit character display — Right entry

For description of right and left entry, see Interface Considerations. Note that when decoded scan is set in keyboard mode, the display is reduced to 4 characters independent of display mode set.

KKK

0	0	0	Encoded Scan Keyboard — 2 Key Lockout*
0	0	1	Decoded Scan Keyboard — 2-Key Lockout
0	1	0	Encoded Scan Keyboard — N-Key Rollover
0	1	1	Decoded Scan Keyboard — N-Key Rollover
1	0	0	Encoded Scan Sensor Matrix
1	0	1	Decoded Scan Sensor Matrix
1	1	0	Strobed Input, Encoded Display Scan
1	1	1	Strobed Input, Decoded Display Scan

Program Clock

Code:	0	0	1	P	P	P	P	P

All timing and multiplexing signals for the 8279 are generated by an internal prescaler. This prescaler divides the external clock (pin 3) by a programmable integer. Bits PPPPP determine the value of this integer which ranges from 2 to 31. Choosing a divisor that yields 100 kHz will give the specified scan and debounce times. For instance, if Pin 3 of the 8279 is being clocked by a 2 MHz signal, PPPPP should be set to 10100 to divide the clock by 20 to yield the proper 100 kHz operating frequency.

Read FIFO/Sensor RAM

Code:	0	1	0	AI	X	A	A	A

X = Don't Care

The CPU sets up the 8279 for a read of the FIFO/Sensor RAM by first writing this command. In the Scan Key-

*Default after reset.

board Mode, the Auto-Increment flag (AI) and the RAM address bits (AAA) are irrelevant. The 8279 will automatically drive the data bus for each subsequent read ($A_0 = 0$) in the same sequence in which the data first entered the FIFO. All subsequent reads will be from the FIFO until another command is issued.

In the Sensor Matrix Mode, the RAM address bits AAA select one of the 8 rows of the Sensor RAM. If the AI flag is set (AI = 1), each successive read will be from the subsequent row of the sensor RAM.

Read Display RAM

Code:

0	1	1	AI	A	A	A	A

The CPU sets up the 8279 for a read of the Display RAM by first writing this command. The address bits AAAA select one of the 16 rows of the Display RAM. If the AI flag is set (AI = 1), this row address will be incremented after each following read *or write* to the Display RAM. Since the same counter is used for both reading and writing, this command sets the next read *or write* address and the sense of the Auto-Increment mode for both operations.

Write Display RAM

Code:

1	0	0	AI	A	A	A	A

The CPU sets up the 8279 for a write to the Display RAM by first writing this command. After writing the command with $A_0 = 1$, all subsequent writes with $A_0 = 0$ will be to the Display RAM. The addressing and Auto-Increment functions are identical to those for the Read Display RAM. However, this command does not affect the source of subsequent Data Reads; the CPU will read from whichever RAM (Display or FIFO/Sensor) which was last specified. If, indeed, the Display RAM was last specified, the Write Display RAM will, nevertheless, change the next Read location.

Display Write Inhibit/Blanking

Code:

				A	B	A	B
1	0	1	X	IW	IW	BL	BL

The IW Bits can be used to mask nibble A and nibble B in applications requiring separate 4-bit display ports. By setting the IW flag (IW = 1) for one of the ports, the port becomes marked so that entries to the Display RAM from the CPU do not affect that port. Thus, if each nibble is input to a BCD decoder, the CPU may write a digit to the Display RAM without affecting the other digit being displayed. It is important to note that bit B_0 corresponds to bit D_0 on the CPU bus, and that bit A_3 corresponds to bit D_7.

If the user wishes to blank the display, the BL flags are available for each nibble. The last Clear command issued determines the code to be used as a "blank." This code defaults to all zeros after a reset. Note that both BL flags must be set to blank a display formatted with a single 8-bit port.

Clear

Code:

1	1	0	C_D	C_D	C_D	C_F	C_A

The C_D bits are available in this command to clear all rows of the Display RAM to a selectable blanking code as follows:

C_D C_D C_D	
0 X	All Zeros (X = Don't Care)
1 0	AB = Hex 20 (0010 0000)
1 1	All Ones

Enable clear display when = 1 (or by C_A = 1)

During the time the Display RAM is being cleared (~160 μs), it may not be written to. The most significant bit of the FIFO status word is set during this time. When the Display RAM becomes available again, it will automatically reset.

If the C_F bit is asserted ($C_F = 1$), the FIFO status is cleared and the interrupt output line is reset. Also, the Sensor RAM pointer is set to row 0.

C_A, the Clear All bit, has the combined effect of C_D and C_F; it uses the C_D clearing code on the Display RAM and also clears FIFO status. Furthermore, it resynchronizes the internal timing chain.

End Interrupt/Error Mode Set

Code:

1	1	1	E	X	X	X	X

X = Don't care

For the sensor matrix modes this command lowers the IRQ line and enables further writing into RAM. (The IRQ line would have been raised upon the detection of a change in a sensor value. This would have also inhibited further writing into the RAM until reset).

For the N-key rollover mode — if the E bit is programmed to "1" the chip will operate in the special Error mode. (For further details, see Interface Considerations Section.)

Status Word

The status word contains the FIFO status, error, and display unavailable signals. This word is read by the CPU when A_0 is high and $\overline{CS}$ and $\overline{RD}$ are low. See Interface Considerations for more detail on status word.

Data Read

Data is read when A_0, $\overline{CS}$ and $\overline{RD}$ are all low. The source of the data is specified by the Read FIFO or Read Display commands. The trailing edge of $\overline{RD}$ will cause the address of the RAM being read to be incremented if the Auto-Increment flag is set. FIFO reads always increment (if no error occurs) independent of AI.

Data Write

Data that is written with A_0, $\overline{CS}$ and $\overline{WR}$ low is always written to the Display RAM. The address is specified by the latest Read Display or Write Display command. Auto-incrementing on the rising edge of $\overline{WR}$ occurs if AI, set by the latest display command.

INTERFACE CONSIDERATIONS

Scanned Keyboard Mode, 2-Key Lockout

There are three possible combinations of conditions that can occur during debounce scanning. When a key is depressed, the debounce logic is set. Other depressed keys are looked for during the next two scans. If none are encountered, it is a single key depression and the key position is entered into the FIFO along with the status of CNTL and SHIFT lines. If the FIFO was empty, IRQ will be set to signal the CPU that there is an entry in the FIFO. If the FIFO was full, the key will not be entered and the error flag will be set. If another closed switch is encountered, no entry to the FIFO can occur. If all other keys are released before this one, then it will be entered to the FIFO. If this key is released before any other, it will be entirely ignored. A key is entered to the FIFO only once per depression, no matter how many keys were pressed along with it or in what order they were released. If two keys are depressed within the debounce cycle, it is a simultaneous depression. Neither key will be recognized until one key remains depressed alone. The last key will be treated as a single key depression.

Scanned Keyboard Mode, N-Key Rollover

With N-key Rollover each key depression is treated independently from all others. When a key is depressed, the debounce circuit waits 2 keyboard scans and then checks to see if the key is still down. If it is, the key is entered into the FIFO. Any number of keys can be depressed and another can be recognized and entered into the FIFO. If a simultaneous depression occurs, the keys are recognized and entered according to the order the keyboard scan found them.

Scanned Keyboard — Special Error Modes

For N-key rollover mode the user can program a special error mode. This is done by the "End Interrupt/Error Mode Set" command. The debounce cycle and key-validity check are as in normal N-key mode. If during a single debounce cycle, two keys are found depressed, this is considered a simultaneous multiple depression, and sets an error flag. This flag will prevent any further writing into the FIFO and will set interrupt (if not yet set). The error flag could be read in this mode by reading the FIFO STATUS word. (See "FIFO STATUS" for further details.) The error flag is reset by sending the normal CLEAR command with CF = 1.

Sensor Matrix Mode

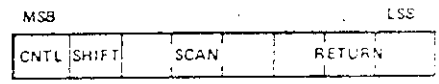
In Sensor Matrix mode, the debounce logic is inhibited. The status of the sensor switch is inputted directly to the Sensor RAM. In this way the Sensor RAM keeps an image of the state of the switches in the sensor matrix. Although debouncing is not provided, this mode has the advantage that the CPU knows how long the sensor was closed and when it was released. A keyboard mode can only indicate a validated closure. To make the software easier, the designer should functionally group the sensors by row since this is the format in which the CPU will read them.

The IRQ line goes high if any sensor value change is detected at the end of a sensor matrix scan. The IRQ line is cleared by the first data read operation if the Auto-Increment flag is set to zero, or by the End Interrupt command if the Auto-Increment flag is set to one.

Note: Multiple changes in the matrix Addressed by (SL_{0-3} = 0) may cause multiple interrupts. (SL_0 = 0 in the Decoded Mode). Reset may cause the 8279 to see multiple changes.

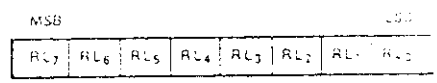
Data Format

In the Scanned Keyboard mode, the character entered into the FIFO corresponds to the position of the switch in the keyboard plus the status of the CNTL and SHIFT lines (non-inverted). CNTL is the MSB of the character and SHIFT is the next most significant bit. The next three bits are from the scan counter and indicate the row the key was found in. The last three bits are from the column counter and indicate to which return line the key was connected.

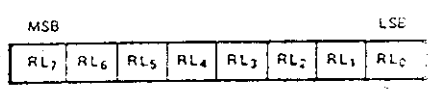


SCANNED KEYBOARD DATA FORMAT

In Sensor Matrix mode, the data on the return lines is entered directly in the row of the Sensor RAM that corresponds to the row in the matrix being scanned. Therefore, each switch position maps directly to a Sensor RAM position. The SHIFT and CNTL inputs are ignored in this mode. Note that switches are not necessarily the only thing that can be connected to the return lines in this mode. Any logic that can be triggered by the user can enter data to the return line inputs. Eight simple input ports could be tied to the return lines to read the status of 64 switches.



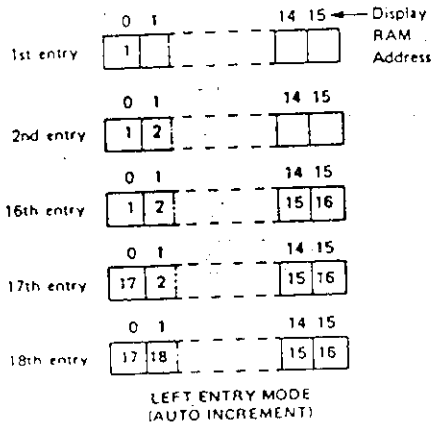
In Strobed Input mode, the data is also entered to the FIFO from the return lines. The data is entered by the rising edge of a CNTL/STB line pulse. Data can come from another encoded keyboard or simple switch matrix. The return lines can also be used as a general purpose strobed input.



Display

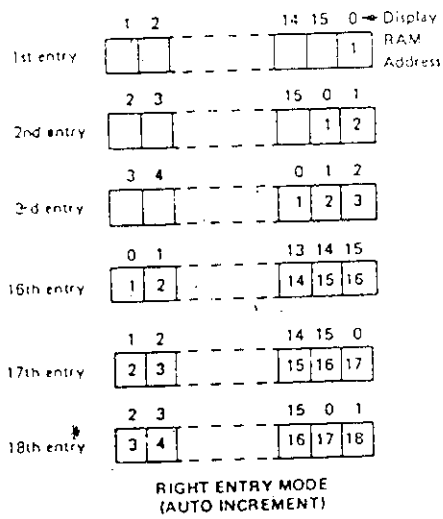
Left Entry

Left Entry mode is the simplest display format in that each display position directly corresponds to a byte (or nibble) in the Display RAM. Address 0 in the RAM is the left-most display character and address 15 (or address 7 in 8 character display) is the right most display character. Entering characters from position zero causes the display to fill from the left. The 17th (9th) character is entered back in the left most position and filling again proceeds from there.



Right Entry

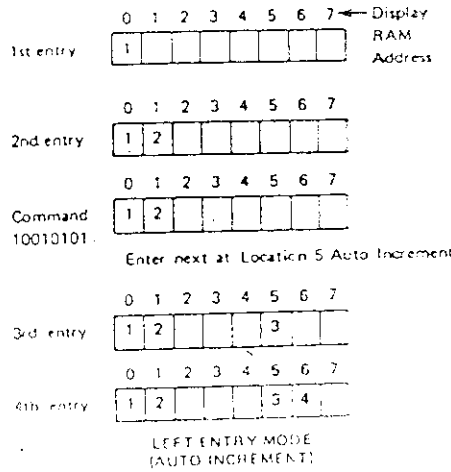
Right entry is the method used by most electronic calculators. The first entry is placed in the right most display character. The next entry is also placed in the right most character after the display is shifted left one character. The left most character is shifted off the end and is lost.



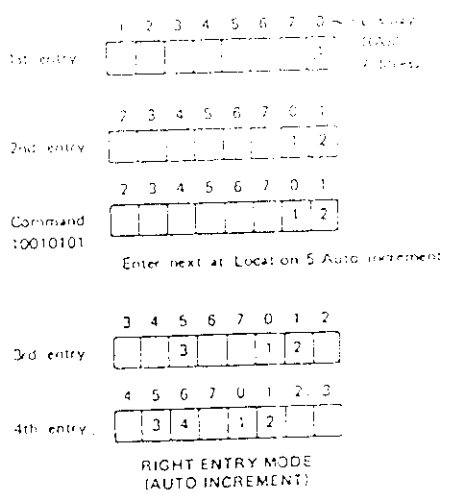
Note that now the display position and register address do not correspond. Consequently, entering a character to an arbitrary position in the Auto Increment mode may have unexpected results. Entry starting at Display RAM address 0 with sequential entry is recommended.

Auto Increment

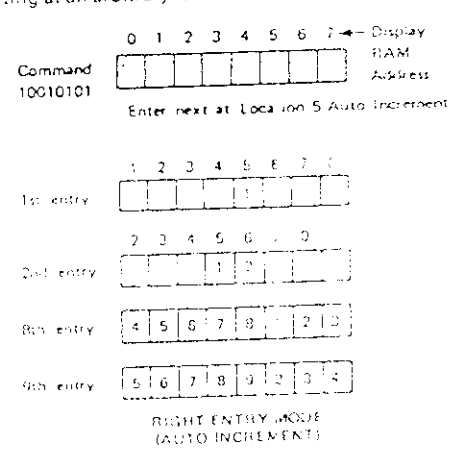
In the Left Entry mode, Auto Incrementing causes the address where the CPU will next write to be incremented by one and the character appears in the next location. With non-Auto Incrementing the entry is both to the same RAM address and display position. Entry to an arbitrary address in the Auto Increment mode has no undesirable side effects and the result is predictable.



In the Right Entry mode, Auto Incrementing and non Incrementing have the same effect as in the Left Entry except if the address sequence is interrupted.



Starting at an arbitrary location operates as shown below:



Entry appears to be from the initial entry point.

8/16 Character Display Formats

If the display mode is set to an 8 character display, the on duty-cycle is double what it would be for a 16 character display (e.g., 5.1 ms scan time for 8 characters vs. 10.3 ms for 16 characters with 100 kHz internal frequency).

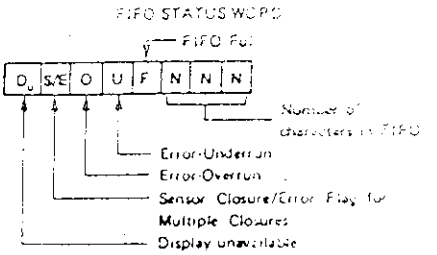
G. FIFO Status

FIFO status is used in the Keyboard and Strobed Input modes to indicate the number of characters in the FIFO and to indicate whether an error has occurred. There are two types of errors possible: overrun and underrun. Overrun occurs when the entry of another character into a full FIFO is attempted. Underrun occurs when the CPU tries to read an empty FIFO.

The FIFO status word also has a bit to indicate that the Display RAM was unavailable because a Clear Display or Clear All command had not completed its clearing operation.

In a Sensor Matrix mode, a bit is set in the FIFO status word to indicate that at least one sensor closure indication is contained in the Sensor RAM.

In Special Error Mode the S/E bit is showing the error flag and serves as an indication to whether a simultaneous multiple closure error has occurred.



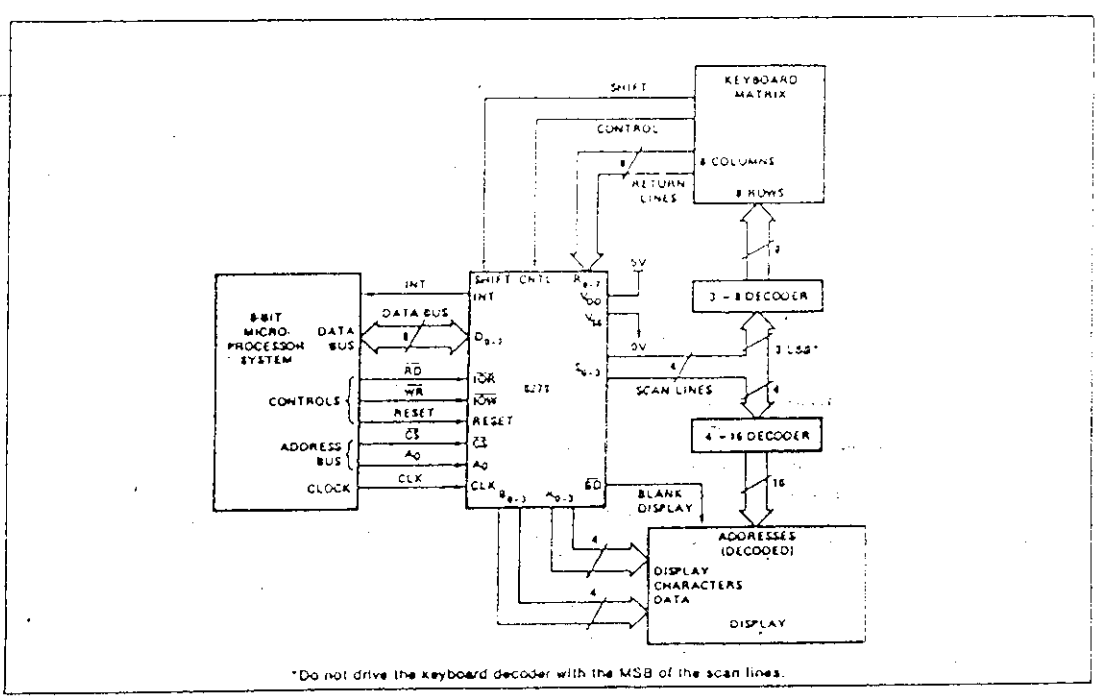


Figure 4. System Block Diagram

MCS®-51
8-BIT CONTROL-ORIENTED MICROCOMPUTERS

8031/8051
8031AH/8051AH
8032AH/8052AH
8751H/8751H-12

- **High Performance HMOS Process**
- **Internal Timers/Event Counters**
- **2-Level Interrupt Priority Structure**
- **32 I/O Lines (Four 8-Bit Ports)**
- **64K Program Memory Space**
- **Boolean Processor**
- **Bit-Addressable RAM**
- **Programmable Full Duplex Serial Channel**
- **111 Instructions (64 Single-Cycle)**
- **64K Data Memory Space**

- **Security Feature Protects EPROM Parts Against Software Piracy**

The MCS®-51 products are optimized for control applications. Byte-processing and numerical operations on small data structures are facilitated by a variety of fast addressing modes for accessing the internal RAM. The instruction set provides a convenient menu of 8-bit arithmetic instructions, including multiply and divide in structions. Extensive on-chip support is provided for one-bit variables as a separate data type, allowing direct bit manipulation and testing in control and logic systems that require Boolean processing.

Device	Internal Memory		Timers/ Event Counters	Interrupts
	Program	Data		
8052AH	8K × 8 ROM	256 × 8 RAM	3 × 16-Bit	6
8051AH	4K × 8 ROM	128 × 8 RAM	2 × 16-Bit	5
8051	4K × 8 ROM	128 × 8 RAM	2 × 16-Bit	5
8032AH	none	256 × 8 RAM	3 × 16-Bit	6
8031AH	none	128 × 8 RAM	2 × 16-Bit	5
8031	none	128 × 8 RAM	2 × 16-Bit	5
8751H	4K × 8 EPROM	128 × 8 RAM	2 × 16-Bit	5
8751H-12	4K × 8 EPROM	128 × 8 RAM	2 × 16-Bit	5

The 8751H is an EPROM version of the 8051AH; that is, the on-chip Program Memory can be electrically programmed, and can be erased by exposure to ultraviolet light. It is fully compatible with its predecessor, the 8751-8, but incorporates two new features: a Program Memory Security bit that can be used to protect the EPROM against unauthorized read-out, and a programmable baud rate modification bit (SMOD). SMOD is not present in the 8751H-12.

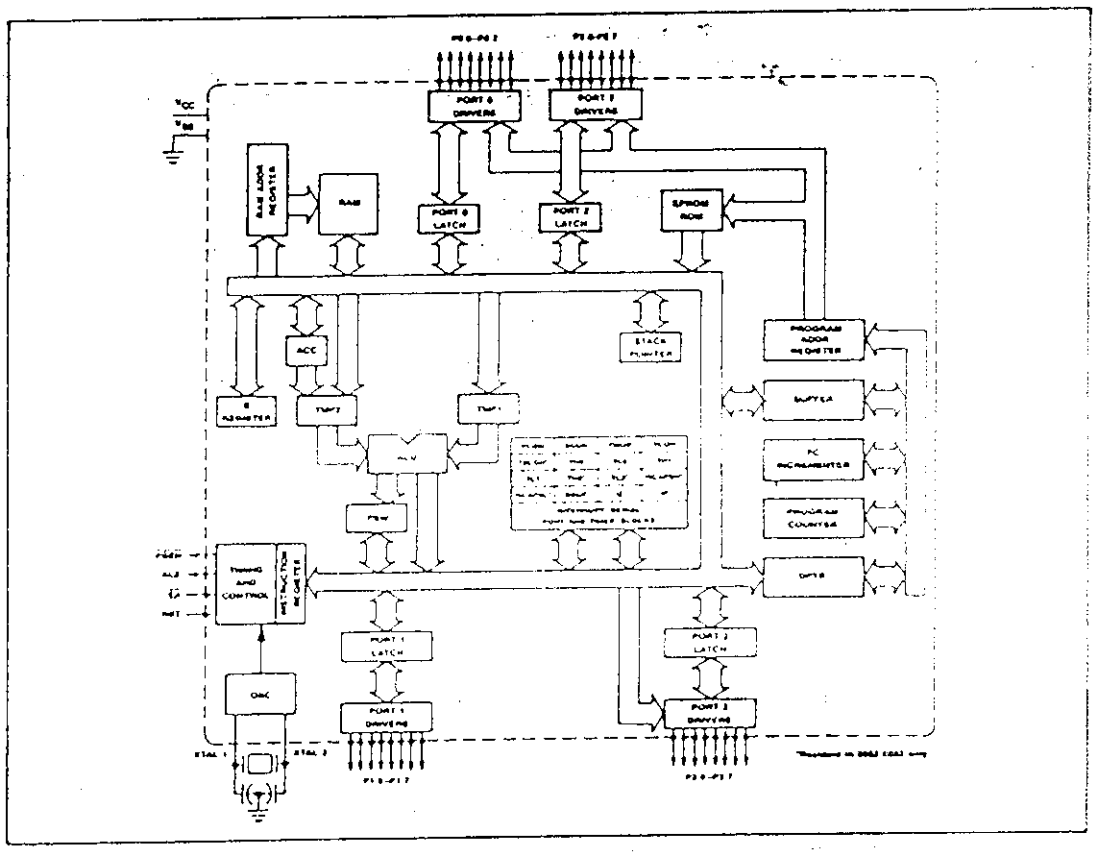


Figure 1. MCS®-51 Block Diagram

PIN DESCRIPTIONS

VCC

Supply voltage.

VSS

Circuit ground.

Port 0

Port 0 is an 8-bit open drain bidirectional I/O port. As an output port each pin can sink 8 LS TTL inputs. Port 0 pins that have 1s written to them float, and in that state can be used as high-impedance inputs.

Port 0 is also the multiplexed low-order address and data bus during accesses to external Program and Data Memory. In this application it uses strong internal pullups when emitting 1s, and can source and sink 8 LS TTL inputs.

Port 0 also receives the code bytes during programming of the EPROM parts, and outputs the code bytes during program verification of the ROM and EPROM parts. External pullups are required during program verification.

Port 1

Port 1 is an 8-bit bidirectional I/O port with internal pullups. The Port 1 output buffers can sink/source 4 LS TTL inputs. Port 1 pins that have 1s written to them are pulled high by the internal pullups, and in that state can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL}, on the data sheet) because of the internal pullups.

Port 1 also receives the low-order address bytes during programming of the EPROM parts and during program verification of the ROM and EPROM parts.

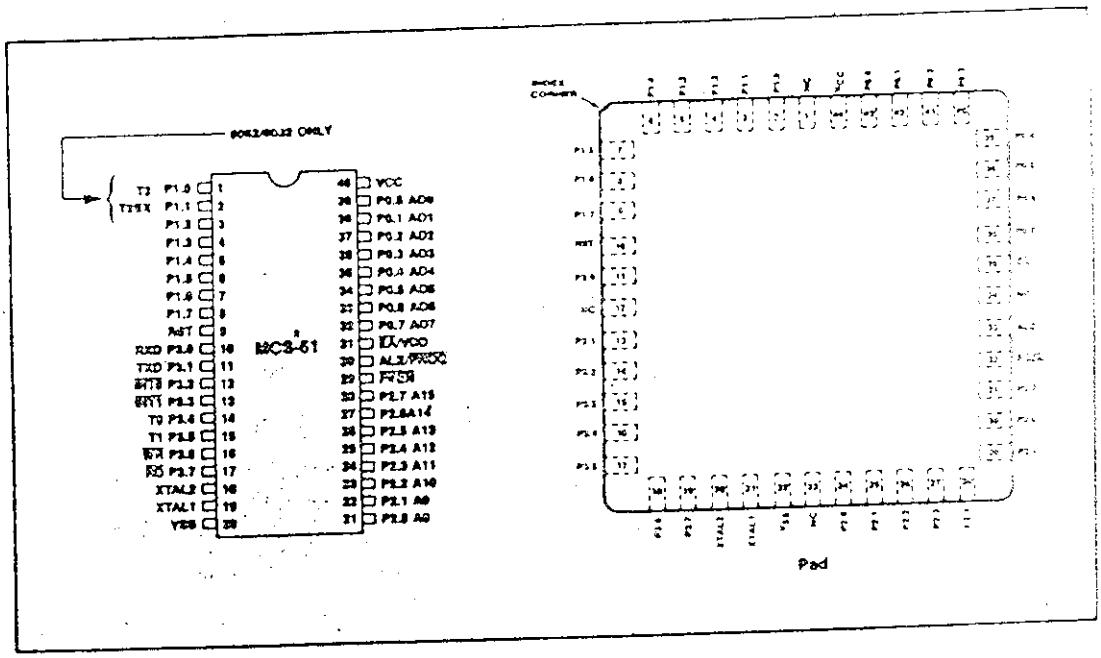


Figure 2. MCS®-51 Pin Connections

In the 8032AH and 8052AH, Port 1 pins P1.0 and P1.1 also serve the T2 and T2EX functions, respectively.

Port 2

Port 2 is an 8-bit bidirectional I/O port with internal pullups. The Port 2 output buffers can sink/source 4 LS TTL inputs. Port 2 pins that have 1s written to them are pulled high by the internal pullups, and in that state can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (IIL, on the data sheet) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external Program Memory and during accesses to external Data Memory that use 16-bit addresses (MOVX @DPTR). In this application it uses strong internal pullups when emitting 1s. During accesses to external Data Memory that use 8-bit addresses (MOVX @Ri), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits during programming of the EPROM parts and during program verification of the ROM and EPROM parts.

Port 3

Port 3 is an 8-bit bidirectional I/O port with internal pullups. The Port 3 output buffers can sink/source 4 LS TTL inputs. Port 3 pins that have 1s written to them are pulled high by the internal pullups, and in that state can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (IIL, on the data sheet) because of the pullups.

Port 3 also serves the functions of various special features of the MCS-51 Family, as listed below:

Port Pin	Alternative Function
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	$\overline{INT0}$ (external interrupt 0)
P3.3	$\overline{INT1}$ (external interrupt 1)
P3.4	T0 (Timer 0 external input)
P3.5	T1 (Timer 1 external input)
P3.6	$\overline{WR}$ (external data memory write strobe)
P3.7	$\overline{RD}$ (external data memory read strobe)

RST

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

ALE/$\overline{PROG}$

Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. ALE can drive 8 LS TTL inputs. This pin is also the program pulse input ($\overline{PROG}$) during programming of the EPROM parts.

In normal operation ALE is emitted at a constant rate of 1/6 the oscillator frequency, and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external Data Memory.

$\overline{PSEN}$

Program Store Enable is the read strobe to external Program Memory. $\overline{PSEN}$ can drive 8 LS TTL inputs.

When the device is executing code from external Program Memory, $\overline{PSEN}$ is activated twice each machine cycle, except that two $\overline{PSEN}$ activations are skipped during each access to external Data Memory.

$\overline{EA}$/VPP

External Access enable $\overline{EA}$ must be externally held low in order to enable any MCS-51 device to fetch code from external Program Memory locations 0 to 0FFFH (0 to 1FFFH, in the 8032AH and 8052AH).

Note, however, that if the Security Bit in the EPROM devices is programmed, the device will not fetch code from any location in external Program Memory.

This pin also receives the 21V programming supply voltage (VPP) during programming of the EPROM parts.

XTAL1

Input to the inverting oscillator amplifier.

XTAL2

Output from the inverting oscillator amplifier.

OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 3. Either a quartz crystal or ceramic resonator may be used. More detailed information concerning the use of the on-chip oscillator is available in Application Note AP-155, "Oscillators for Microcontrollers."

To drive the device from an external clock source, XTAL1 should be grounded, while XTAL2 is driven, as shown in Figure 4. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum high and low times specified on the Data Sheet must be observed.

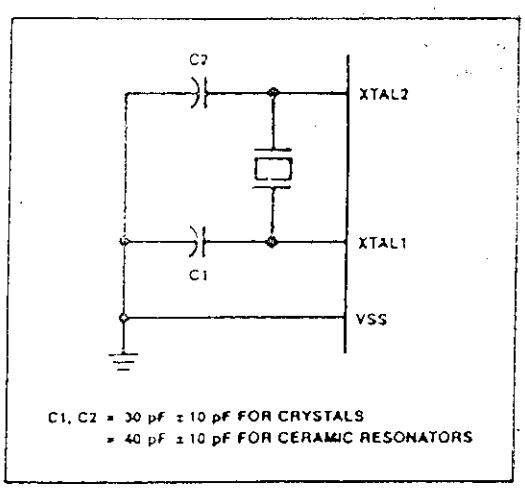


Figure 3. Oscillator Connections

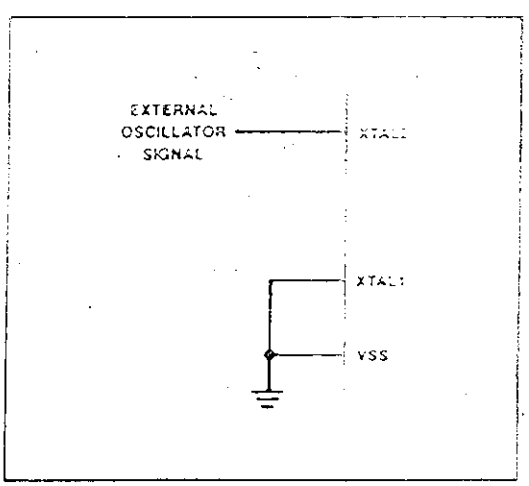


Figure 4. External Drive Configuration

MOTOROLA

MCM63256

Product Preview

256K BIT READ ONLY MEMORY

The MCM63256 is a MOS mask programmable byte-organized Read Only Memory (ROM). The MCM63256 is organized as 32,768 bytes of 8 bits and is fabricated using Motorola's high performance N-channel silicon gate technology (HMOS). This device is designed to provide maximum circuit density and reliability with highest possible performance while maintaining low power dissipation and wide operating margins and remaining fully compatible with TTL inputs and outputs.

The active level of the Chip Enable and the Output Enable, along with the memory contents, are defined by the user. The Chip Enable input deselects the output and puts the chip in a power-down mode.

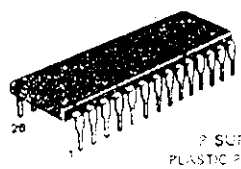
- Single ±10% +5 Volt Power Supply
- Fully Static Periphery — No Clocking Required on Chip Enable
- Automatic Power Down
- Power Dissipation
  - 100 mA Active (Maximum) (Unloaded)
  - 15 mA Standby (Maximum)
- Current Surge Suppression When Powering Up Device
- Program Layer Late in Process for Quick Turnaround Time
- 150 ns Maximum Access from Address and Chip Enable
- 28-Pin JEDEC Standard Package and Pinout

ADDITIONAL FEATURE

- Address (A14) is User Selectable for Either Pin 27 or Pin 1

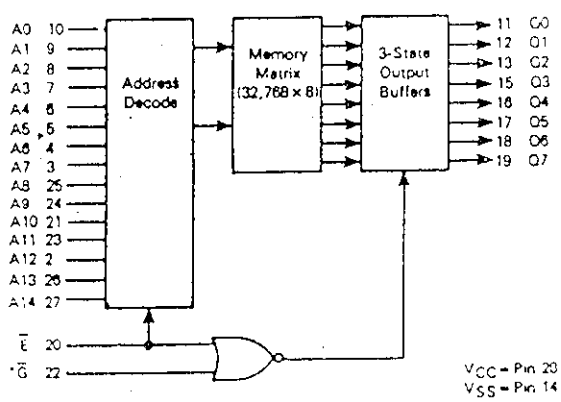
HMOS
(N-CHANNEL, SILICON GATE)

32,768 × 8 BIT
READ ONLY MEMORY



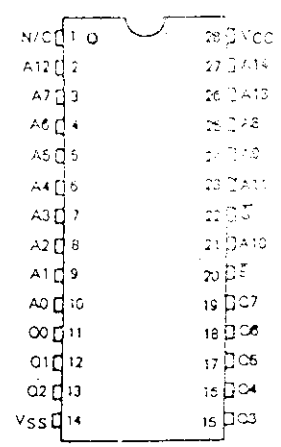
P SUFFIX
PLASTIC PACKAGE
CASE 710

BLOCK DIAGRAM



*Active level defined by the user.

PIN ASSIGNMENT



PIN NAMES	
A0-A14	Address
E	Chip Enable
G	Output Enable
O0-O7	Data Output
VCC	+5 V Power Supply
VSS	Ground

This document contains information on a product under development. Motorola reserves the right to change or discontinue this product without notice.

NP-8/0/5-82

256 K-BIT (32,768 x 8) CMOS STATIC RANDOM ACCESS MEMORY WITH DATA RETENTION AND LOW POWER

The Fujitsu MB 84256 is a 32,768-word by 8-bit static random access memory fabricated with a CMOS silicon gate process. The memory utilizes asynchronouse circuitry and may be maintained in any state for an indefinite period of time. All pins are TTL compatible, and a single +5 volts power supply is required.

The MB 84256 is ideally suited for use in microprocesser systems and other applications where fast access time and ease of use are required. All devices offer the advantages of low power dissipation, low cost and high performance.

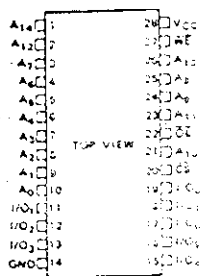
- Organization: 32,768 x 8 bits
- Fast access time: 100 ns max. (MB 84256-10/10L/10LL)
  120 ns max. (MB 84256-12/12L/12LL)
  150 ns max. (MB 84256-15/15L/15LL)
- Completely static operation: No clock required
- TTL compatible inputs/outputs
- Three-state outputs
- Single +5V power supply, ±10% tolerance
- Low power standby:
  CMOS level: 5.5 mW max. (MB 84256-10/12/15)
  0.55 mW max. (MB 84256-10L/10LL/12L/12LL/15L/15LL)
  TTL level: 16.5 mW max. (MB 84256-10/10L/10LL/12/12L/12LL/15/15L/15LL)
- Data retention: 2.0V
- Standard 28-pin DIP (600 mil) (Suffix: -P)
- Standard 28-pin Bend-type Plastic Flat Package (450 mil) (Suffix: -PF)
- Standard 32-pad LCC (Suffix: -CV)

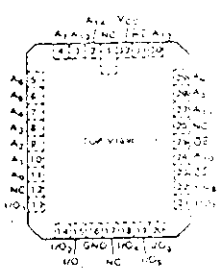
PLASTIC PACKAGE
DIP-28P-M02

PLASTIC PACKAGE
FPT-28P-M02

CERAMIC PACKAGE
LCC-32C-A02

PIN ASSIGNMENT





ABSOLUTE MAXIMUM RATINGS (see NOTE)

Rating		Symbol	Value	Unit
Supply Voltage		V_{CC}	−0.5 to +7.0	V
Input Voltage		V_{IN}	−0.5 to V_{CC}+0.5	V
Output Voltage		V_{OUT}	−0.5 to V_{CC}+0.5	V
Temperature Under Bias		T_{BIAS}	−10 to +85	°C
Storage Temperature Range	CERAMIC	T_{STG}	−65 to +150	°C
	PLASTIC		−40 to +125	

27512
512K (64K x 8) PRODUCTION AND UV ERASABLE PROM

- Software Carrier Capability
- 170 ns Maximum Access Time
- Two-Line Control
- Int$_e$ligent Identifier™ Mode
  — Automated Programming Operations
- TTL Compatible
- Low Power
  — 125 mA max. Active
  — 40 mA max. Standby
- Int$_e$ligent Programming™ Algorithm
- Available in 28-Pin Cerdip
  (See packaging spec order #231369)

The Intel 27512 is a 5V-only, 524,288-bit ultraviolet Erasable and Electrically Programmable Read Only Memory (EPROM) organized as 64K words by 8 bits. This ensures compatibility with high-performance microprocessors, such as the Intel 8 MHz iAPX 286, allowing full speed operation without the addition of performance-degrading WAIT states. The 27512 is also directly compatible with Intel's 8051 family of microcontrollers.

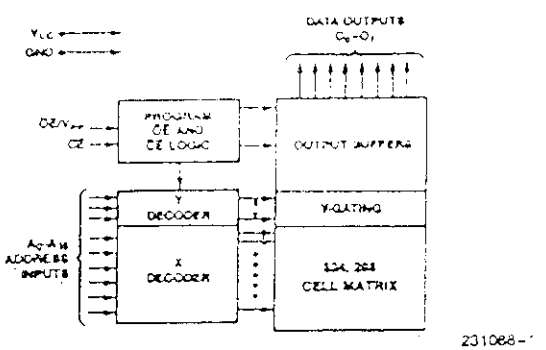
The 27512 enables implementation of new, advanced systems with firmware intensive architectures. The combination of the 27512's high-density, cost-effective EPROM storage, and new advanced microprocessors having megabyte addressing capability provides designers with opportunities to engineer user-friendly, high-reliability, high-performance systems.

The 27512's large storage capability of 64 K-bytes enables it to function as a high-density software carrier. Entire operating systems, diagnostics, high-level language programs and specialized application software can reside in a 27512 EPROM directly on a system's memory bus. This permits immediate microprocessor access and execution of software and eliminates the need for time-consuming disk accesses and downloads.

Two-line control and JEDEC-approved, 28-pin packaging are standard features of all Intel high-density EPROMs. This assures easy microprocessor interfacing and minimum design efforts when upgrading, adding, or choosing between nonvolatile memory alternatives.

The 27512 is manufactured using Intel's advanced HMOS *II-E technology.

*HMOS is a patented process of Intel Corporation.



231068-1

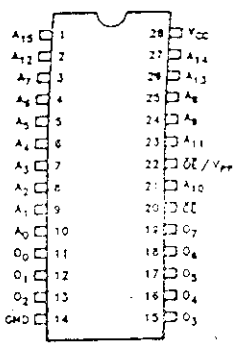
Figure 1. Block Diagram

Pin Names

A_0–A_{15}	Addresses
$\overline{CE}$	Chip Enable
$\overline{OE}/V_{PP}$	Outputs Enable/V_{PP}
O_0–O_7	Outputs
D.U.	Don't Use

27513	27256 27C256 87C256	27128 27128A	2764 2764A 27C64 87C64
D.U.	V_{PP}	V_{PP}	V_{PP}
A_{12}	A_{12}	A_{12}	A_{12}
A_7	A_7	A_7	A_7
A_6	A_6	A_6	A_6
A_5	A_5	A_5	A_5
A_4	A_4	A_4	A_4
A_3	A_3	A_3	A_3
A_2	A_2	A_2	A_2
A_1	A_1	A_1	A_1
A_0	A_0	A_0	A_0
D_0/O_0	O_0	O_0	O_0
D_1/O_1	O_1	O_1	O_1
O_2	O_2	O_2	O_2
Gnd	Gnd	Gnd	Gnd

27512



231088-2

2764 2764A 27C64 87C64	27128 27128A	27256 27C256 87C256	27513
V_{CC}	V_{CC}	V_{CC}	V_{CC}
PGM	PGM	A_{14}	$\overline{WE}$
N.C.	A_{13}	A_{13}	A_{13}
A_8	A_8	A_8	A_8
A_9	A_9	A_9	A_9
A_{11}	A_{11}	A_{11}	A_{11}
$\overline{OE}$	$\overline{OE}$	$\overline{OE}$	$\overline{OE}/V_{PP}$
A_{10}	A_{10}	A_{10}	A_{10}
$\overline{CE}$/A_{14}/$\overline{CE}$	$\overline{CE}$	$\overline{CE}$/A_{13}/$\overline{CE}$	$\overline{CE}$
O_7	O_7	O_7	O_7
O_6	O_6	O_6	O_6
O_5	O_5	O_5	O_5
O_4	O_4	O_4	O_4
O_3	O_3	O_3	O_3

Figure 2. Pin Configurations

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 hours, 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.

EXPRESS EPROM PRODUCT FAMILY

PRODUCT DEFINITIONS

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

27512 VERSIONS

Packaging Options	
Speed Versions	**Cerdip**
-170V05	Q
-200V05, -200V10	Q, L
-STD, -25, -250V10	Q, T, L
-3, -30	Q, T, L

74LS245
Transceiver

Octal Transceiver (3-State)
Product Specification

Logic Products

FEATURES

- Octal bidirectional bus interface
- 3-State buffer outputs
- PNP inputs for reduced loading
- Hysteresis on all Data inputs

DESCRIPTION

The 'LS245 is an octal transceiver featuring non-inverting 3-State bus compatible outputs in both send and receive directions. The outputs are all capable of sinking 24mA and sourcing up to 15mA, producing very good capacitive drive characteristics. The device features a Chip Enable (CE) input for easy cascading and a Send/Receive (S/R) input for direction control. All data inputs have hysteresis built in to minimize AC noise effects.

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74LS245	8ns	58mA

ORDERING CODE

PACKAGES	COMMERCIAL RANGE V_{CC} = 5V ±5%; T_A = 0°C to +70°C
Plastic DIP	N74LS245N
Plastic SOL-20	N74LS245D

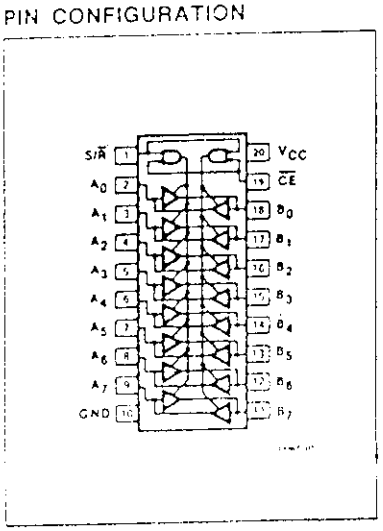
NOTE:
For information regarding devices processed to Military Specifications, see the Signetics Military Data Manual.

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

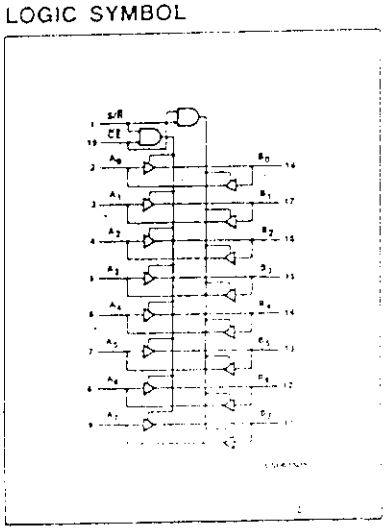
PINS	DESCRIPTION	74LS
All	Inputs	1.0/3.0
All	Outputs	150/40

NOTE:
Where a 74LS unit load (LSUL) is 20µA I_{IH} and −0.4mA I_{IL}.

PIN CONFIGURATION



LOGIC SYMBOL



LOGIC SYMBOL (IEEE/IEC)