MICROPROCESSOR BASED WAVE GENERATOR

SUBMITTED IN THE PARTIAL FULFILMENT OF THE REQUIREMENTS FOR
THE AWARD OF THE DEGREE OF BACHELOR OF ENGINEERING IN
COMPUTER TECHNOLOGY & INFORMATICS ENGINEERING
OF THE BHARATHIAR UNIVERSITY, COIMBATORE-641 006.

P-104

SUBMITTED BY

T. Nanda Kishore

K. V. Krishna Kumar

M. C. Rajender Prasad

UNDER THE GUIDANCE OF

Miss. R. NALINA, B E.

DEPARTMENT OF COMPUTER TECHNOLOGY AND INFORMATICS

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006.

1989 - 1990

DEPARTMENT OF COMPUTER TECHNOLOGY AND INFORMATICS

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006.

CERTIFICATE

This is to certify that the report entilted "MICROPROCESSOR BASED WAVE GENERATOR" has been submitted by.....

Is partial fulfilment for award of Bachelor of Engineering in the COMPUTER TECHNOLOGY & INFORMATICS Branch of the Bharathiar University, Coimbatore - 641 006. during the academic year 1989 - 1990.

GUIDE

HEAD OF DEPARTMENT

Certified that the candidate was examined by us in the project work Viva-Voce examination held on.....and the University register No. was.....

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

First of all, we wish to express our heartfelt gratitude to our beloved Principal, **Prof.R.Palanivelu** B.E., M.Sc.(Engg.), FIE, MISTIE, MISPRS, for having provided us with excellent facilities in our college to complete our project successfully.

We are greatly indebted to **Prof.P.Shanmugam** M.Sc. (Engg.), MS, MIEE, MISTIE, Head of Computer Technology and Informatics Department for his valuable suggestions, constructive criticism and timely advises.

We sincerely extend our heartfelt gratitude to our guide **Miss.R.Nalina,** B.E., for her selfless and valuable guidance throughout the project.

We sincerely wish to thank **Mr.T.Suresh,** B.E., for extending his valuable support and help in finishing our project.

We are greatly indebted to **Mr.M.Ramaswamy,** Assistant Professor, Department of Computer Technology and Informatics, for his valuable support and advice which was really helpful in finishing our project.

Finally, we wish to thank all the faculty members of Computer Techhnology and Informatics Department for their valuable help and encouragement.

- TEAM MEMBERS

CONTENTS

SYNOPSIS

This project on microprocessor based wave generator has been designed to help many control operations by way of generating waveforms. Also the software provides a wide range of selection both in choosing the type of waveform i.e. (1) Square wave, (2) Triangular, (3) Sawtooth and (4) Sinusoidal and the frequency of all these waveforms. An easy selection can be made in frequency by going through user's manual where the equivalent Hexadecimal values to be entered in a given memory location are available. Choosing the type of waveform is also similar to the above procedure. Henceforth any user should make two entries (1) for choosing the waveform type and (2) for choosing the frequency.

The software is written for the above mentioned jobs and the generation of the waveform involves the 8255 port of Microprocessor-8085 and 0800 DAC, - a digital/analog converter. The values to be input, are being fed to 8255 ports which act as input ports to the DAC 0800 via the 8085 processor.

The D/A converter now makes the conversion through the control words specified by software routine which generates the required waveform through a series of digital datas, put into an indefinite loop which continues to give an output similar to any oscillator through the analog output pins of the D/A converter.

In the pages to come, the above mentioned procedure has been explained in detail step by step throwing light to the practical work encountered in this project with adequate flow charts, for software knowledge and chip details for hardware knowledge. To impart the software clearly the whole software is grouped into a few sectors and individually attention has been paid on the explanation of using and performing it efficiently.

2 . USAGE OF MICROPROCESSOR IN WAVE GENERATION

The microprocessor is a general purpose programable logic device. The thorough understanding of the microprocessor demands the concepts and skills from two diffierent disciplines: hardware concepts from electronics and programing skills from computer science. Hardware is the physical structure of the microprocessor and programming makes it come alive. Therefore, in this project report, the contents are presented with a general approach to hardware and software in the context of 8085 microprocessor.

In most control operations it is necessary to generate different types of waveform, depending on the circumstances the control operation is subjected to. Hence it becomes necessary to generate the required waveforms with different frequencies and these varying frequencies to control the operation timings. As an advancement through feedback we can receive the information of error and changes and correspondingly change the physical parameteres of the waveform. This is where the real time operation is being enhanced with the microprocessor.

3. ABOUT 8085A/8085 MICROPROCESSOR

3.1 Introduction

The microprocessor is a programmable logic device, designed with registers, flip-flops and timing elements. The microprocessor has a set of instructions designed internally, to manipulate data and communication with peripherals. The microprocessor performs primarily four operations:

1. Memory Read : Reads data from memory
2. Memory write : Write data from memory
3. I/O Read : Accepts data from input device
4. I/O Write : Sends data to output device

All these four operations are part of the communication process between the Microprocessor and peripheral (or a memory location), the microprocessor needs to perform the following steps:

Step 1 : Identify the peripheral or memory location (with its address)

Step 2 : Transfer data

Step 3 : Provide timing or synchronization signals

The 8085 microprocessor performs these functions using three sets of communication times called buses: the address bus, the data bus, the control bus.

3.2 BLOCK DIAGRAM

The block diagram of 8085 is shown in fig.(3.1). Eventhough the drawings does not include control signals, they drive all the internal registers.

Address, Data and Control Buses

Near the top of the drawing is an 8-bit internal data bus. This carries instructions and data between the CPU register. The external buses are the ones, which are connected to other chips like memory, I/O and so forth. Near the bottom left of the drawing is the external control bus (RD, WR, ALE,...). On the bottom right are the external address and address-data buses.

The upper bits are on a separate bus always used for address bit; this upper section of the address bus is designated A15-A8. The lower 8 bits are multiplexed. This means that the eight lower bus lines are used for adddressing bits during some T states and for data bits during other T states. This is why the bus is labeled address data bus, designated AD_7 - AD_0

Why is multiplexing used in 8085? Because at the time this chip was developed, the practical limit on the number of pins was 40. The only solution was to multiplex part of the addrress bus with the data bus.

Address Bus

The address bus is a group of sixteen lines generally identified as A_0 to A_{15}. The address bus is unidirectional: bits flow in one direction-from the microprocessor to peripheral devices. [The microprocessor uses the address bus to perform the first function (Step 1)]

Data Bus

The data bus is a group of eight lines used for data flow. These lines are bidirectional-data flow in both directions between the microprocessor and peripharal devices. [The microprocessor uses the data bus to perform the second function (Step 2)].

Control Bus

The control bus is comprised of various single lines that carry synchronization signals. The term bus, in relation to the control signals are somewhat confusing.These are individual lines that provide a pulse to indicate a microprocessor operation. The microprocessor generates specific control signals for every operation it performs. These signals are used to identify a device type with which the microprocessor intends to communicate.

Accumulator

The accumulator is an 8-bit register. This register is used to store 8-bit data and to perform arithematic and logic operations. The result of an operation is stored in the accumulator. As shown, this is connected to the 8-bit internal data bus. The bidirectional arrow between the accumulator and the bus indicates a three-state connection that allows the accumulator to send or receive data.

Temporary Registers

The other input for the ALU comes from the temporary register. This 8-bit register stores the operands of the arithematic-logic operation. For instance, during an ADDC the contents of the C register are copied in the temporary register during one T state and added during another T state.

ALU and Flags

The ALU carries out the arithematic and logic operations. As shown, the contents of the accumulator and the temporary register are the inputs to the ALU.

The four different flags are: Zero, Sign, Carry and Parity. The 8085 includes a fifth flag, called the auxilliary carry flag.

Instruction Register and Decoder

During the fetch cycle, the op code of an instruction is stored in the instruction register. The op code then drives the instruction decodes and machine-cycle encoder.

Timing and Control

The timing and control includes an oscillator and a controller-sequencer. The oscillator generates the two-phase clock signals (CLK and CLK) that synchronize all registers. The controllers-sequencer also produces the control signals needed for internal and external control.

The controlled-sequencer is microprogrammed; it had a ROM that stores all the microroutines needed for executing the instruction. After each instruction is fetched and stored in the instruction register, the op code is decoded to get the starting address of the desired microroutine. As each micro instruction is read out of the control ROM, the control signals are sent to the internal and external data buses. The effect is to move data between registers, to perform arithematic-logic operations, to input or output data.

CPU Registers

The array of CPU register (B, C, D, etc.). This register array is like a small on-chip RAM with addressable

memory locations control signals select the register for a read or write operation.

Included in the register array are the stock pointer, program counter, and incrementer-decrementer. This can add, or substract, from the contents of the stock pointer or program counter.

Address Buffer and Address-Data Buffer:

At the bottom right are two buffer registers called the address buffer and the address-data buffer. The contents of the stock pointer or program counter can be loaded into the address buffer and address-data buffer. The output of these buffers then drives the external address bus. Memory and I/O chips are connected to these buses. In this way the CPU can send the address of desired data to the memory or I/O chips.

Interrupt control

Sometimes it is necessary to interrupt the execution of the main program to answer a request from an I/O device. For instance, an I/O device (Input/Output) may send an interrupt signal to the interrupt control unit to indicate that data is ready for output/input.

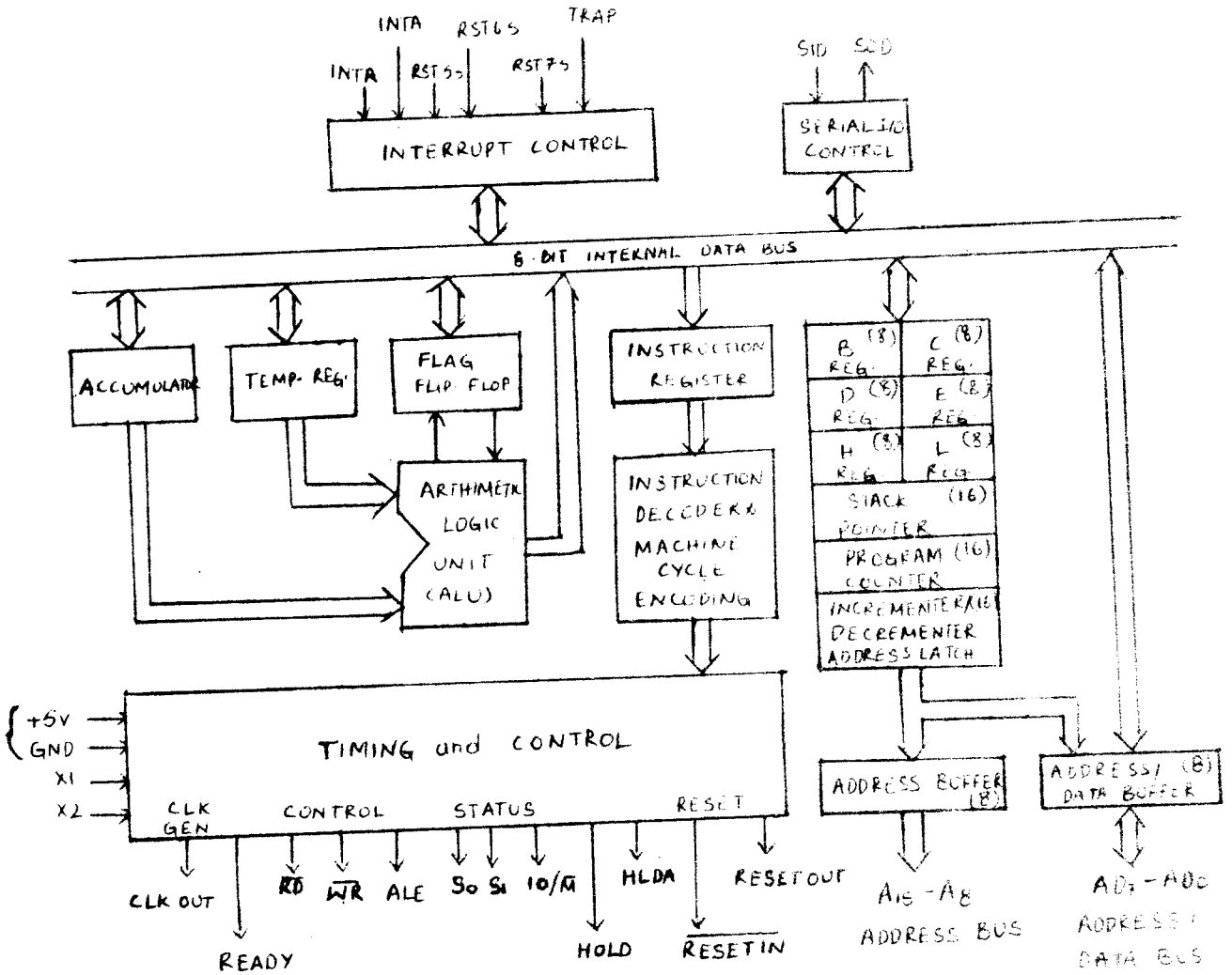


FIG. 3.4 8085 BLOCK DIAGRAM

Sometimes, I/O devices work with serial data rather than parallel. In this case, the serial data stream from an input device must be converted to 8-bit parallel data before the computer can use it.

The SID input at the upper right of fig (3.1) is where serial input data enters the 8085. The SOD output is where the serial data leaves the 8085.

3.3 PINOUT DIAGRAM

Figure (3.2) is the Pinout Diagram for the 8085. A brief description of each pin is as follows:

PIN 1 and 2

Oscillators are crystal-LC or RC-controlled. The 8085 has an on-chhip oscillators with all the required circuitry except for the crystal, LC tank or RCnetwork that controls the freqeuency. This is the purpose of pins 1 and 2; connected a crystal, LC Circuit or RC network to X1 and X2.

Pin 3

This pin carries RESET OUT signal when high, it indicates that the CPU is being reset; that is, the program counter, instruction register and so on are being reset to zero. The RESET OUT signal goes out to peripharal chips. When power in up, initially, the whole system including the 8085 and peripharal chips is reset or initialized. After the RESET OUT goes slow, the processing begins.

Pin 4 and 5

SOD stands for serial out data. This serial data comes out of Pin 4, which can be connected to a serial output device.

SID stands for serial in data. Pin 5 is the input pin for serial data.

Pin 6 to 11

These pins are ports of the interrupt control unit. The 8085 has five inputs for interrupt requests, a priority exists among the interrupt pins; some are more important than other. Inorder of their importance the five interrupt signals are designated TRAP, RST 7.5, RST 6.5, RST 5.5 and INTR of two or more interrupts go high at the same times the 8085 will service them inorder in which, as shown above. Pin 6 to 10 are input pins for the interrupt signals. Pin 11, is, however, is an out pin with a signal called the interrupt acknowledge (INTA). This particular signal is used in response to an INTR signal.

Pin 12 to 28

Pins 12 to 19 carry the lower 8 address bits or the 8 bits data. As explained above, the lower half of the address bus is multiplexed with the data bus to keep the pin count at 40. Pin 20 is labelled Vss, is the system ground. Pins 21 to 28 are the rest of the address bus.

Pins 29 and 33

Pins 29 and 33 carry output signals known as status signals. Labelled S0 and S1, these status signals (and IO/$\overline{M}$ signal) indicate whether our instruction fetch, memory read, memory write or other operation is taking place.

Pin 30

8085 Microprocessor needs one or more memory chips connected to it. Each memory chip has its own MAR, usually called an 'address latch'. The latch stores the incoming address from the address bus and address-data bus.

At what point in the machine cycle does a memory chip store the incoming address? This is where the ALE signal comes in. ALE stands for address latch enable. The ALE signal comes out of Pin 30 and goes to peripheral chips such as memory chips. The falling edge of ALE signal strobes (loads) the address on the address bus and address data bus into the MAR or address latch of the memory chips.

Pin 31, 32 and 34

These three pins function together. They are connected to memory and I/O chips. Pin 34 carries the IO/$\overline{M}$ signal. A low IO/$\overline{M}$ indicates a memory operation, and a high IO/$\overline{M}$ means that an I/O instruction is being executed. In other words, a low IO/$\overline{M}$ signal enables the memory chips, and high IO/$\overline{M}$ enables the I/O chips.

The WR and RD determine whether a write or a read is done. Since these signals are active low, a low WR means a write operation and low RD means a read operation.

Pin 35

Some peripheral devices are slow; they are unable to run at the same speed at the 8085. One way to slow down the 8085 is with READY signal.

The 8085 addressing a peripheral device as the first step in sending or receiving data from that device. If the device is not ready, it will return a low READY but to 8085. The 8085 then generates a number of T states (called WAIT states). Eventually, when the peripheral device is ready, it will send a high READY signal to 8085. Then the 8085 can complete the data transfer.

Pin 36 and 37

Pin 36 is an input carrying the RESET IN signal. This signal may come from an operator reset botton or other sources. When RESETIN is low, the CPU will reset the program counter, instruction register, and other circuits. The CPU remains in reset until the RESET IN signal goes high. Then the data processing begins. The CLK signal out of 37 is derived from the onchip oscillator. CLS is the system clock; each cyle represents one T state. The CLK signal goes to peripheral chips and synchronize their timings.

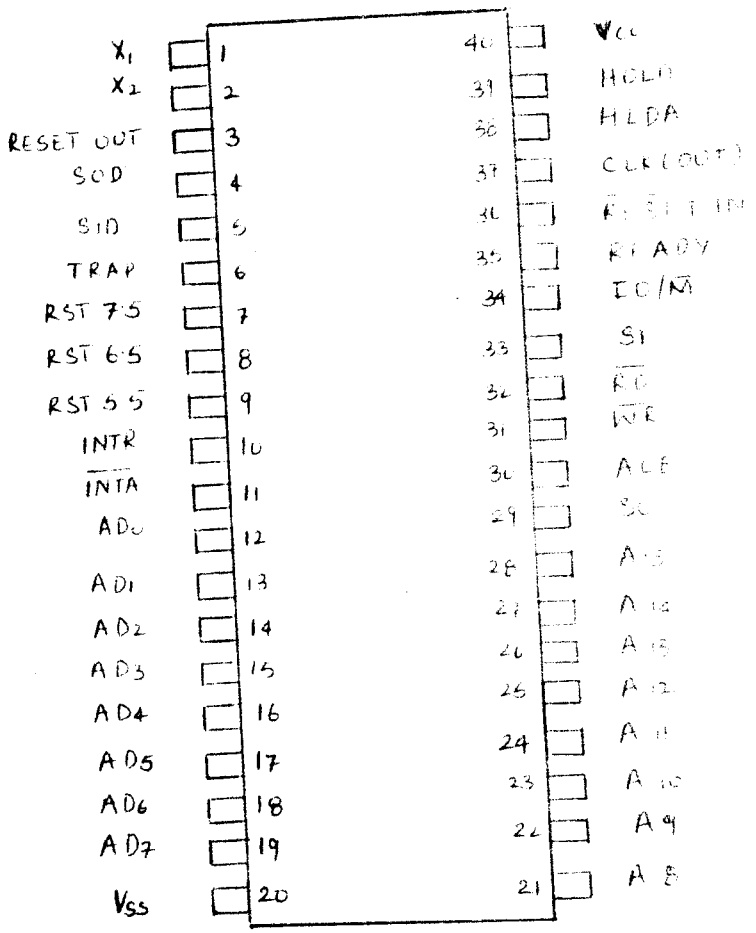


FIG. 3.3 8085 PINOUT

Pin 38 to 40

The IN instruction is the usual way to input data from peripheral devices. The accumulator is involved because it receives the input data. Similarly, the OUT instruction transfers data from the accumulator to output devices. In either case, going through accumulator slow down I/O transfers.

The solution to speeding up memory-peripheral transfers is called direct memory access (DMA). In this approach, the 8085 turns over control to the buses to a DMA controller, a chip optimized for high-speed memory transfers. The HOLD and HLDA signals (Pins 39 and 38) are used in DMA operations with the DMA approach, large amounts of data can be transfered in a short time. Pin 40 is lost pins. It connects to a source of +5V. The tolerable on the supply voltage is $\pm$5 percent. The power dissipation is less than 1.5W.

4. ABOUT DAC 0800

4.1 General Description

The DAC 0800 is a monolithic 8 bit high speed current-output digital to analog converters (DAC) featuring typical settling time of 100 ns. When used as a multiplying DAC, monotonic performance over a 40 to 1 reference current range is possible. It also features high compliance complentary current outputs to allow differential output voltages of 20 Vp.p with simple resistor loads as shown in figure.

The refernece to full scale current matching of better than ±LSB eliminates the need for full scale trims in most applications. while the nonlinearities of better than ±0.1% over temperature minimizes system error accumulations.

The noise immune inputs of the DAC0800 series will accept TTL levels with the logic threshold pin. VLC pin grounded. Simple adjustments of the VLC potential allow direct interface to all logic families. The performance and characteristics of the device are essentially unchanged over the fullscale ±4.5V to ±18V power supply range. Power description is only 33 nW with ±5V supplies and is independent of the logic input states.

4.2 Features

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

Operating condition of DAC0800 is between temperatures -55°C and +125°C. The DAC0800 is an R-2R ladder digital to analog converter. The R-2R ladder circuit is discussed in more detail in the following section.

4.3 THE R-2R LADDER D/A CONVERTER

The converter-resistor array of figure 4.1 (a) uses resistors of only two sizes R and 2R. In this figure only 4 bits are indicated, which ofcourse can be expanded to accomodate an arbitrary number of bits. It is understood from the figure that when du=1 the corresponding resistor is connected to a voltage Vt and when du=0, the resistor input is grounded.

To see most simply the relative weight at the output of the individual switches, consider the situation where we have set do=1, while d1=d2=d3=0 as shown in figure 4.1 (b). Here we find the voltage source VR in series with a resistor 2R. Applying Thevenin's theorem at AA' we find, as shown in figure 4.1 (c) that we now have a voltage source VR/2 in series with R. We repeat the application of Thevenin's theorem at BB', CC' and DD'. We find that at each such application, the voltage source is again divided by 2, while the Thevenin's equivalent output impedance remains constant at R. The final equivalent circuit for the output is shown in figure 4.1 (d). If we repeated these operations starting with S1=1, S3=S2=S0=0, we would find an equivalent circuit as in figure (d) concept that the voltage source would be $VR/2^3$ and so on for switches d2 and d3. Thus, at the output each switch contributes its proper relative binary weight. For the arrangeent in figure 4.1 (a) we then have

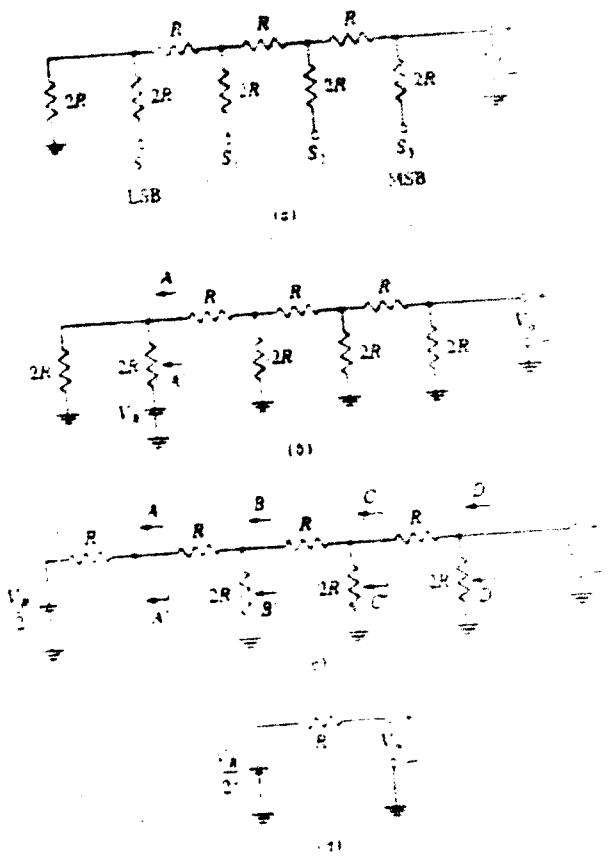


FIGURE 4-1
(a) The R-2R ladder D/A converter (b) and (c) Thevenin's theorem used to determine the output voltage V_o. (d) The final equivalent circuit as seen at the output

$$V_0 = V_R \left(\frac{S_3}{2^1} + \frac{S_2}{2^2} + \frac{S_1}{2^3} + \frac{S_0}{2^4} \right) \quad \text{--- (A)}$$

and

$$V_0 = \frac{V_R}{2^4} \left(S_3 2^3 + S_2 2^2 + S_1 2^1 + S_0 2^0 \right) \quad \text{--- (B)}$$

or more generally if there are N input digits and corresponding N switches, we have,

$$V_0 = \frac{V_R}{2^N} \left(S_{N-1} 2^{N-1} + S_{N-2} 2^{N-2} + \ldots + S_0 2^0 \right) \quad \text{--- (C)}$$

A D/A converter using an R-2R ladder and an OP-amp buffer is shown in figure 4.2. It can be verified that the output V0 is given by equation (C).

4.4 SPECIFICATIONS FOR D/A CONVERTERS

We now consider a number of the parameters which serve to describe the quality of performance of a D/A converter. These parameters are generally specified by manufacturers of converters.

Resolution

This term specifies the number of bits the converter can accommodate and correspondingly the number of output voltages (or currents). For example, a converter which can accept 8 input bits is referred to as a converter with a 8-bit resolution. The number of possible output voltages is $2^8=256$. Hence the smallest possible change in output voltage is 1/256 of the full-scale output range.

Linearity

In an ideal D/A converter equal increments in the numerical significance of the digital-input should yield equal increment in the analog output. The linearity of a converter serves as a measure of the precision with which this requirement is satisfied. Linearity is measured in the manner suggested by the figure 4.3.

THE MEASUREMENT OF LINEARITY

Here the input bit combinations with fixed interval separation, in the order of numerical significance is located along the x-axis. Along the y axis the corresponding analog output voltage for each case, as might be amounted in a physical converter is indicated by volts. If the converter were perfect, the dots would fall on a straight line. In the figure 4.3 a straight line is drawn that best fits the dots and the linearity error is indicated by E. The voltage is the the nominal analog output change corresponding to a digital input change equivalent to a change in the least significant bit (LSB).

The linearity of a converter is generally specified by comparing E to . This linearity of a commercial unit is specified as "less than ±½ LBS", meaning that IEI ½ .

The linearity of a converter depends principally on the accuracy of the resistors. It depends as well on the precision with which the voltage drops across the switches are fixed. Since both of these are temperature-dependent, linearity may be adversely affected by substantial temperature changes.

Accuracy

The accuracy of a converter is a measure of the difference between the actual analog output voltage and what the output should

be in the ideal case. Lack of linearity contributes to inaccuracy. Further limitations on accuracy are contributed by the uncertainity in the reference voltages, the amplifier gain, amplifier offset etc.

Settling time

When the digital input to a converter changes, switches open and close and abrupt voltage changes appear. Because of the inevitable stray capacitance and inductance present in the passive circuitry. The transients so initiated may persist for an appreciable time. Typically a plot of the change in output voltage as a function of time might be as shown in the figure 4.4

Settling time in a D/A

The interval that elapses from the input change to the time when the output has come close enough to its final value is called the settling time. Typically a general purpose converter might have a settling time given as "500ns to 0.2° of full scale".

Temperature sensitivity

At any fixed digital input, the analog output will vary with temperature. This sensitivity typically ranges from

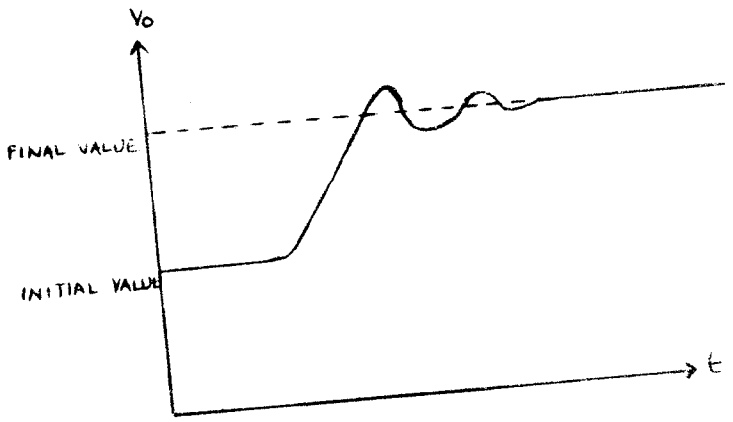


FIG. 4.3 SETTILING TIME IN D/A

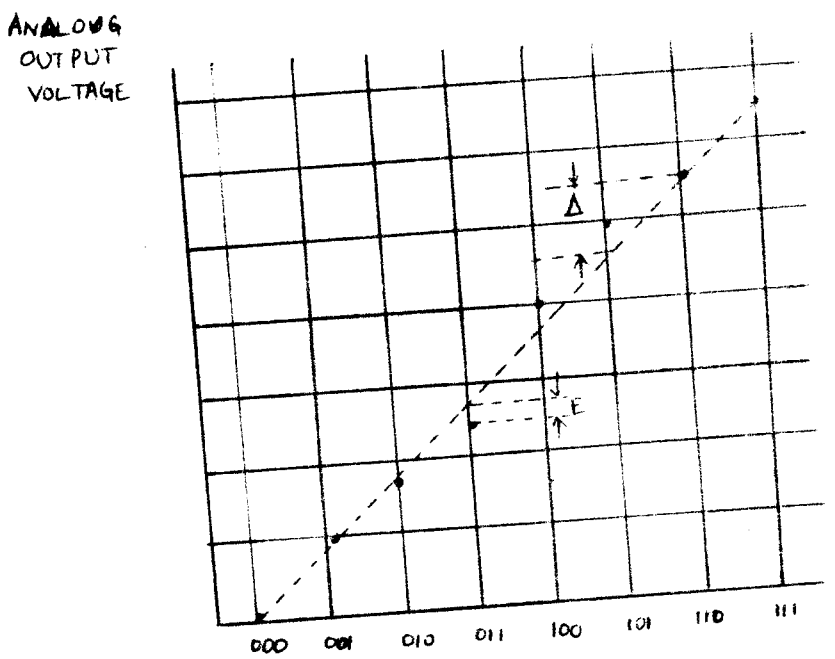


FIG. 4.4 MEASUREMENT OF LINEARITY

about ±50 ppm/°C in a general purpose converter to as low as ±1.5 ppm/°C in a high-quality unit. The overall temperature sensitivity is due to the temperature sensitivies of reference voltages, the resisters in the converter, the op-amp. and even the amplifier offset voltage.

5. CIRCUIT DESCRIPTION

In the circuit description of MICROPROCESSOR based wave generator the whole circuit can be logically divided into five parts which are as follows:-

(i) CPU Part

(ii) Memory part

(iii) I/O decoding part

(iv) DAC part

(v) Keyboard and display part.

Each of the above five sections are briefly explained as follows:-

5.1 CPU part

The CPU of this circuit of MICROPROCESSOR BASED WAVE generator, as explained in previous chapter consists of 8085 microprocessor. The pin details are shown in the fig. It has a 8-bit data which comes from Pin 12 - Pin 19 and a 16-bit address bus which comes from Pin 12-Pin 19 and Pin 21 - Pin 28 as shown in the fig. The address and data buses are connected to two D-type latches, which acts as buffers. The 74LS373 pin details can be referred from Appendix. The 74LS373 has 8 D-type flip flops which are edge-triggered. A stable data and address bits can be achieved by this connection.

Pins 1 and 2 of 8085 are connected to a crystal. Pin 3 of 8085 is connected to Pin 9 of 8279 which is a RESET OUT signal, when high, it indicates that the CPU is being reset; i.e. program counter, instruction register, are next to zero when RESET OUT (RESET) goes low, the processing begins. Pin 4 and Pin 5 are SOD and SID. Pin 6 is permanently grounded. Pin 7 is connected to Pin 4 of 8279, which is a key board/display chip.

As shown in the circuit, the Pins 12 to Pin 19 are the data bus lines and Pins 12 to Pin 19 and Pin 21 to Pin 28, total of 16 Pin lines acts as address bus. These in turn, data and address buses are passed through two 74LS373 chips which are Buffers and hence the output from these chips will be token, as input for memory and decoding port.

The pin details of these 74LS373 buffers can be referred from the Appendix.

5.2 Memory Part

In the memory part, a ROM and a EPROM are the chips to consider. IC6116 which is a 28 pin chip RAM has a 13 input address lines from the address bus which ae from A0 to A12. The output from this is connected to the data bus and the other IC 2764 which is an express EPROM has 28 Pins. The input of thirteen address lines comes from the address bus i.e. from A0 to A12. The address locations for ROM and RAM are as follows:

EPROM 0000 to 1 FFF

RAM 4000 to 5 FFF

IC 2764 is a 64K bit 5 V only memories organized as 8192 words of 8 bits. The pin details can be referred from Appendix.

5.3 Decoding Part

In the Decoding part of the circuit, this part has two decoding chips namely IC 74LS138. One for memory decoding and the other for input, output peripherals. These decoders/demultiplexers are 3x8 decoders i.e. they will have 3 inputs and select the output according to the three inputs. The three inputs for memory decoding comes from A13, A14, A15 of the address bus and chip select CSO is used for selecting ERPOM and CS2 for RAM and other pins are left with no connection. The other decoder, which is a periphoral selector has the inputs from A4, A5, A6 of the data bus. These three lines selects DAC and the keyboard/display chips respectively, which are from CS2 and CS3 respectively. The pin details of these decoders can be referred from Appendix.

5.4 DAC Part

The DAC part, which is shown in a separte part of the circuit, has one DAC chip which is DAC 0800, is a 16 pins IC, an op amp and a amplifyig circuit which has two transistors

as shown in the circuit. The inputs for DAC 0800 comes from data bus, through a 74LS374 which is a latch having 8-D-type flip-flops. The output of the buffer acts as input for DAC. The output of DAC 0800 is taken through an operational amplifier which is IC 741. The output of this is taken through a Pin 6. Hence the analog voltage of DAC 0800 can be obtained from this pin. To have a proper waveform, on the CRO, an amplifying circuit is used, which has two transistors. The collector of the first transistor acts as base for the other transistor. Hence the output of the second transistor comes from the collector of it. The pin details of 741 can be referred from Appendix.

This DAC part is shown as a separate cord with a PCB and the chips and transistor used in it. If this alone is made as a separate card, this card can be interfaced through 8255A programmable peripheral interface to 8085 microprocessor. And the software should be written in a way such that the control word must be appropriate for IC 8255. The detailed description of 8255 is explained in the following sections.

5.5 ABOUT 8255A PROGRAMMABLE PERIPHERAL INTERFACE

The 8255A is a programmable, parallel I/O device. It can be programmed to transfer data under various conditions, from simple I/O to interrupt I/O. It is flexible, versatile and economical, but somewhat complex, that can be used with almost any microprocessor.

The 8255 has 24 I/O pins that can be grounded primarily in two 8-bit parallel ports: A and B, with the remaining eight bits as Port C. The 8 bits of Port C can be used as individual bits or be grouped in two 4-bit ports: CUPPER(CU) and CLOWER (CL) as shown in fig. (a) and fig (b) shows all the functions of 8255A, classified all to two modes: the Bit Set/Reset (BSR) mode and I/O mode. The BSR Mode is used to set or reset the bits in Port C. The I/O mode is further divided into three modes: Mode 0, Mode 1 and Mode 2. In Mode 0, all ports function as simple I/O ports, Mode 1 is a handshake mode whereby ports A and/or B use bits from Port C as handshake signals. In the handshake mode, two types of I/O data transfer can be implemented: status check and interrupt. In Mode 2, Port A can be set up for bidirectional data transfer using handshake signals from Port C and Port B can be set up either in Mode 0 or Mode 1.

CONTROL LOGIC

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

RD(Read): This control signal enables the Read operation. When the signal is low, the microprocessor reads data from a selected I/O port of the 8255A.

WR(Write): This control signal enables the write operation. When the signal goes low, the microprocessor writes into a selected I/O port of the control register.

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

CS, AO and A1: These are device select signals. CS is connected to a decoded address, and A0 and A1 are generally connected to microprocessor address liner A0 and A1 respectively. The CS signal is the master chip select and A0 and A1 specify one of the I/O port or the control register as given below:

CS	A1	A0	Selected
0	0	0	Port A
0	0	1	Port B
0	1	0	Port C
0	1	1	Control Register
1	X	X	8255A is not selected

Here MODEO is used for interfacing DAC with 8085 Microprocessor.

PROGRAM DESCRIPTION

6.1 Bipolar Wave Form Generation

The control word 80H is first placed in the control register to initialize the 8255 ports. The 8255 now functions in mode 0 i.e. in simple I/O mode.

To select a particular wave the user has to place the value OA, OB, OC or OD according to the table given below in location 5000.

OA	-	Sawtooth wave
OB	-	Square wave
OC	-	Triangular wave
OD	-	Ramp

Inorder to select the required frequency, the user has to place the corresponding hexadecimal value provided in the look-up table in location 5001.

Now, when the program is executed the required wave with the required frequency is got at the output points of the digital to analog converter.

The explanation for each wave can be referred from the flow charts, which follow in preceding chapters.

6.2 UNIPOLAR WAVE FORM GENERATION

To select the unipolar wave to be generated the user has to input the following values depending on the kind of wave be needs in location 5000.

01	-	Sawtooth wave
02	-	Triangular wave
03	-	Square wave

Further the user has to input the value OE in location 5001 if he wants a negative bias unipolar wave. For positive bias any value other than OE will produce the required unipolar wave. In the location 5002, the hexadecimal value corresponding to the frequency the user needs should be input referring to the look up table.

The description of each program can be understood from the flow chart given in the following section.

The output wave is got from the output Pin P1 of the D/A converter.

Unipolar wave form generation can be done in two ways which are as follows:

(i) Positive Bias

(ii) Negative Bias

The flow charts with the program of each wave can be referred from the chapter following.

7.1 **PROGRAM TO SELECT THE BIPOLAR WAVE TO BE GENERATED**

ADDRESS	INSTR	LABLE	OPCODE	OPERAND	COMMENT
4100	3E,80	START:	MVI	A,80	Enable the 8255 Port
4102	D3, OF		OUT	OF	
4104	3A,00,50		LDA	5000	Load value in 5000 into accumulator.
4107	FE, OA		CP1	OA	
4109	CA,50,41		JZ	4150	Jump on 0 to the locations specified
410C	FE,OB		CP1	OB	
410E	CA,00,42		JZ	4200	
4111	FE,OC		CP1	OC	
4113	CA,00,43		JZ	4300	
4116	FE,OD		CP1	OD	
4118	CA,50,43		JZ	4350	
411B	C3,00,41		JMP	START:	Loop back to start.

7.2 PROGRAM TO GENERATE SAWTOOTH WAVE

ADDRESS	INSTR.	LABLE	OPCODE	OPERAND	COMMENTS
4150	3A,01,50	START:	LDA	5001	Load accumulated with value in location 5001
4153	47		MOV	B,A	
4154	3E,00		MVI	A,00	Move 00 into accumulator
4156	03,01		OUT	OC	Out to port A
4158	80	LOOP1:	ADD	B	Increment accumulator with value in B.
4159	D3,OC		OUT	OC	Out to Port A
415B	C2,58,41		JNZ	LOOP1	Jump on no zero to LOOP 1
415E	C3,50,41		JMP	START1	Loop to start of the program.

7.3 PROGRAM TO GENERATE SQUAR WAVE

ADDRESS	INSTR.	LABLE	OPCODE	OPERAND	COMMENTS
4200	3E,00	START2:	MVI	A,00	Move 00 into accumulator
4202	D3,OC		OUT	OC	Out to Port A
4204	CD,50,42		CALL	DELAY	Call delay
4207	3E,FF		MVI	A,FF	
4209	D3,OC		OUT	OC	
420B	CD,50,42		CALL	DELAY	
420E	C3,00,42		JMP	START:	Jump to start

DELAY PROGRAM

4250	3A,01,50	DELAY:	LDA	5001	Load value of 5001 into accumulator
4253	4F		MOV	C,A	Move accumulator content to C register.
4254	OD	LOOP2:	DCR	C	Decrement C register.
4255	C2,54,42		JNZ	LOOP2	Jump on no zero to Loop 2.
4258	C9		RET		Return

7.4 PROGRAM TO GENERATE TRIANGULAR WAVE

ADDRESS	INSTR	LABLE	OPCODE	OPERAND	COMMENTS
4300	3A,01,50	START3:	LDA	5001	Load value in 5001 into accumulator
4303	6F		MOV	L,A	Move it to L register.
4304	3E,00		MVI	A,00	Move 00 into accumulator
4306	D3,0C		OUT	OC	Out to port A
4308	85	LOOP3:	ADD	L	Increment Register L with accumulator
4309	D3,0C		OUT	OC	Out to Port A
430B	D2,08,43		JNC	LOOP3:	Jump on no carry to Loop 3
430E	3E,FF		MVI	A,FF	Move FF into accumulator
4310	D3,OC		OUT	OC	Out it to Port A
4312	95	LOOP4:	SUB	L	
4313	D3,OC		OUT	OC	
4315	D2,12,43		JNC	LOOP4:	Jump on no carry to Loop 4
4318	C3,00,43		JMP	START3:	Jump to start

7.5 PROGRAM TO GENERATE RAMP WAVE

ADDRESS	INSTR.	LABLE	OPCODE	OPERAND	COMMENTS
4350	3E,FF	START4:	MVI	A,FF	Move FF into accumulator
4352	D3,OC		OUT	OC	Out to Port A
4354	3E,00		MVI	A,00	Move 00 into accumulator
4356	D3,OC		OUT	OC	Out to Port A
4358	C3,50,43		JMP	START4:	Loop to Start of the program

7.6.1. LOOK-UP TABLE FOR SQUARE WAVE

HEXADECIMAL VALUE	FREQUENCY
FF	416.7 Hz
FF	454.5 Hz
DF	476.2 Hz
CF	500.0 Hz
BF	555.6 Hz
AF	588.2 Hz
9F	666.7 Hz
8F	714.2 Hz
7F	833.3 Hz
6F	909.1 Hz
5F	1.11 KHz
4F	125 KHz
3F	1.54 KHz
2F	2 KHz
1F	2.9 KHz
0F	5 KHz

7.6.1. LOOK-UP TABLE FOR SQUARE WAVE

HEXADECIMAL VALUE	FREQUENCY
FF	416.7 Hz
FF	[illegible] Hz
DF	476.2 Hz
CF	500.0 Hz
BF	555.6 Hz
AF	588.2 Hz
9F	666.7 Hz
8F	714.2 Hz
7F	833.3 Hz
6F	909.1 Hz
5F	1.11 KHz
4F	125 KHz
3F	1.54 KHz
2F	2 KHz
1F	2.9 KHz
0F	5 KHz

7.6.2 LOOK UP TABLE FOR SAWTOOTH WAVE

HEXADECIMAL VALUE	FREQUENCY
01	243.9 Hz
02	476.2 Hz
03	714.2 Hz
04	909.1 Hz
05	1.25 KHz
06	1.43 KHz
07	1.6 KHz
08	1.8 KHz
09	2.2 KHz
0A	2.5 KHz
0B	2.78 KHz
0C	2.94 KHz
OD	3.12 KHz
OE	3.3 KHz
OF	3.85 KHz

7.6.3. LOOKUP TABLE FOR TRIANGULAR WAVE

HEXADECIMAL VALUE	FREQUENCY
01	161.3 Hz
02	303 Hz
03	357 Hz
04	476.2 Hz
05	526.3 Hz
06	714.3 Hz
07	833.3 Hz
08	909.1 Hz
09	1 KHz
OA	1.11 KHz
OB	1.61 KHz
OC	1.79 KHz
OD	2 KHz
OE	2.08 KHz
OF	2.17 KHz

7.7.1 PROGRAME TO SELECT THE UNIPOLAR WAVE TO BE GENERATED

ADDRESS	INSTR.	LABLE	OPCODE	OPERAND	COMMENT
4400	3E,80	START:	MVI	A,80	Initialize ports of 8255
4402	D3,OF		OUT	OF	
4404	3A,00,50		LDA	5000	Load accumulator with value in 5000
4407	FE,01		CPI	01	Compare with 01
4409	CA,00,45		JZ	4500	Jump on zero to 4500
440C	FE,02		CPI	02	Compare with 02
440E	CA,00,46		JZ	4600	Jump on zero to 4600
4411	FE,03		CPI	03	Compare with 03
4413	CA,00,47		JZ	4700	Jump on zero to 4700
4416	C3,00,44		JMP	START:	Loop to start of program

7.2.2 UNIPOLAR POSITIVE BIAS SAWTOOTH

ADDRESS	INSTR	LABLE	OPCODE	OPERAND	COMMENTS
4503	3A,01,50		LDA	5001	Load accumulator with value in location 5001
4503	FE,OE		CPI	OE	
4505	C2,50,45		JNZ	4550	Jump to location 4550
4508	3A,02,50	DAT:	LDA	5002	Load accumulator with value 5002
450B	6F		MOV	L,A	
450C	3E,00		MVI	A,00	
450E	D3,OC		OUT	OC	Out ot port A
4510	85		ADD	L	
4511	D3,OC		OUT	OC	
4513	FE,80		CPI	80	Compare with 80
4515	C2,10,45		JNZ	4510	Jump on no carry to location 4510
4518	C2,08,45		JMP	DAT:	

7.7.3 POSITIVE BIAS UNIPOLAR SAWTOOTH WAVE

ADDRESS	INSTR	LABLE	OPCODE	OPERAND	COMMENTS
4550	3A,02,50	START:	LDA	5002	Load accumulator with value 5002
4553	6F		MOV	L,A	
4554	3E,80		MVI	A,80	
4556	D3,OC		OUT	OC	
4558	85	DAD:	ADD	L	Increment L
4559	D3,OC		OUT	OC	Out to port A
455B	FE,FF		CPI	FF	compare with FF
455D	C2,5845		JNZ	DAD:	Go to location 4558
4561	C2,50,45		JMP	START:	Jump to start

7.7.4 PROGRAM FOR TRIANGULAR UNIPOLAR WAVE NEGATIVE BIAS

ADDRESS	INSTR	LABLE	OPCODE	OPERAND	COMMENTS
4600	3A,01,50		LDA	5001	Load accumulator with value 5001
4603	FE,OE		CPI	OE	Compare with OE
4605	C2,50,46		JNZ	4650	
4608	3A,02,50	MUM:	LDA	5002	
460B	6F		MOV	L,A	
460C	3E,00		MVI	A,00	
460E	D3,OC		OUT	OC	Out to port A
4610	85	LAT:	ADD	L	
4611	D3,OC		OUT	OC	Out ot port A
4613	FE,80		CPI	80	Compare with 80
4615	C2,10,46		JNZ	LAT:	Jump to location specified
4618	3E,80		MVI	A,80	
461A	D3,OC		OUT	OC	Out to port A
461C	95	PAT:	SUB	L	
461D	D3,OC		OUT	OC	
461F	D2,46,1C		JNC	PAT:	Jump to the location 461C
4622	C3,08,46		JMP	MUM:	Jump to the location 4608

7.7.5 UNIPOLAR TRIANGULAR POSITIVE BIAS WAVE

ADDRESS	INSTR	LABEL	OPCODE	OPEREND	COMMENTS
4650	3A,02,50	START:	LDA	5002	Load accumulator with the value 5002
			MOV	L,A	
4653	6F		MVI	A,80	
4654	3E,80		OUT	OC	Out to port A
4656	D3,OC		ADD	L	
4658	85	APE:	OUT	OC	
4659	D3,OC		CPI	FF	Compare with FF
465B	FE,FF		JNZ	APE:	
465D	C2,58,46		MVI	A,FF	
4660	3E,FF		OUT	OC	Out to port A
4662	D3,OC		SUB	L	
4664	95	LIP:	OUT	OC	
4665	D3,OC		CPI	80	Compared with 80
4667	FE,80		JNZ	LIP:	
4669	C2,64,46		JMP	START:	
466C	C3,50,46				

7.7.6 UNIPOLAR NEGATIVE BIAS SQUARE WAVE

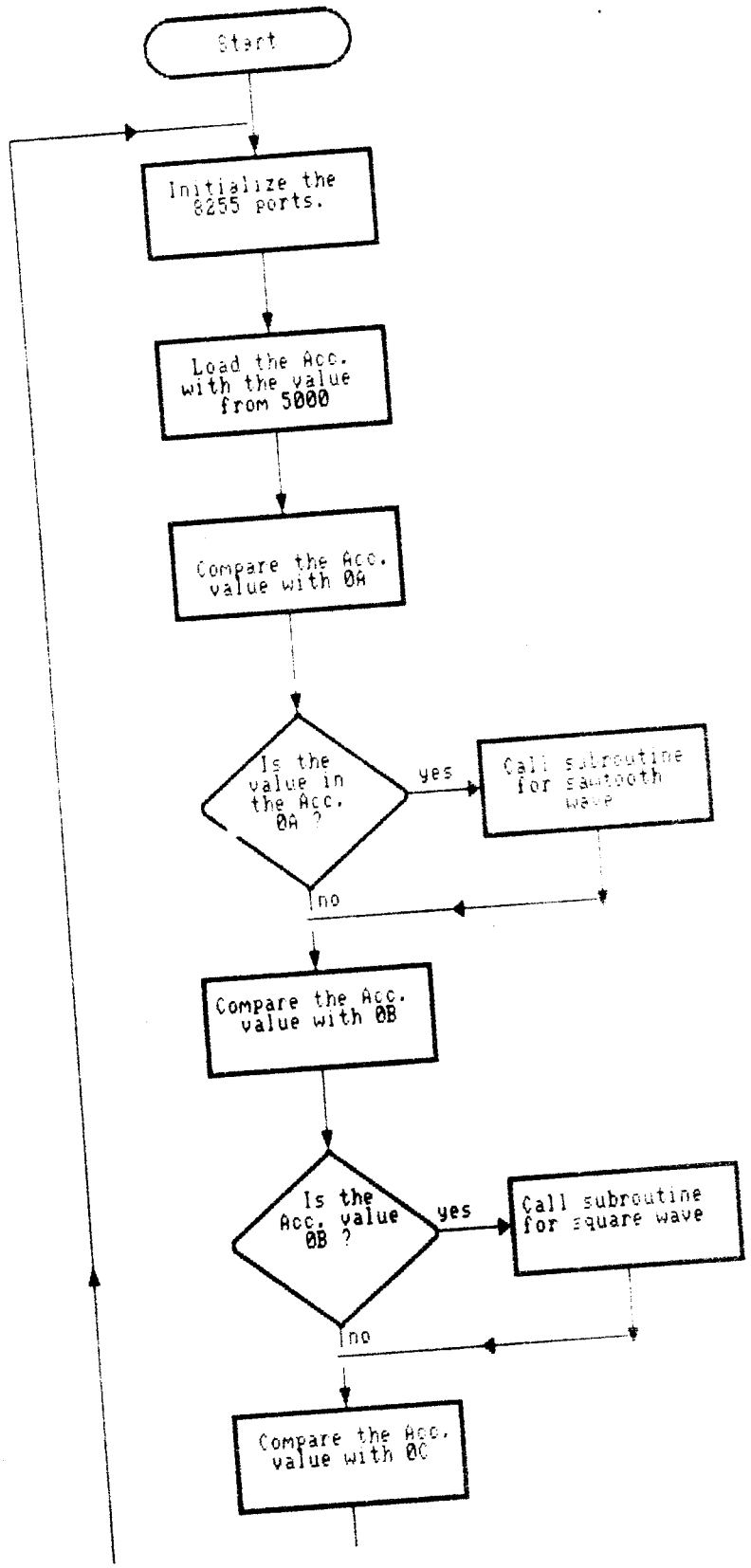
ADDRESS	INSTR.	LABLE	OPCODE	OPERAND	COMMENT
4700	3A,01,50		LDA	5001	Load accumulator with the value
4703	FE, OE		CPI	OE	Compare with OE
4705	C2,00,48		JN2	4800	Jump to specified location
4708	38,00	MID:	MVI	A,00	
470A	D3,OC		OUT	OC	
470C	CD,50,47		CALL	4750	Call Delay
470F	3E,80		MVI	A,80	
4711	D3,OC		OUT	OC	Out to Port A
4713	CD,50,47		CALL	4750	Call DELAY
4716	C3,08,47		JMP	MID:	

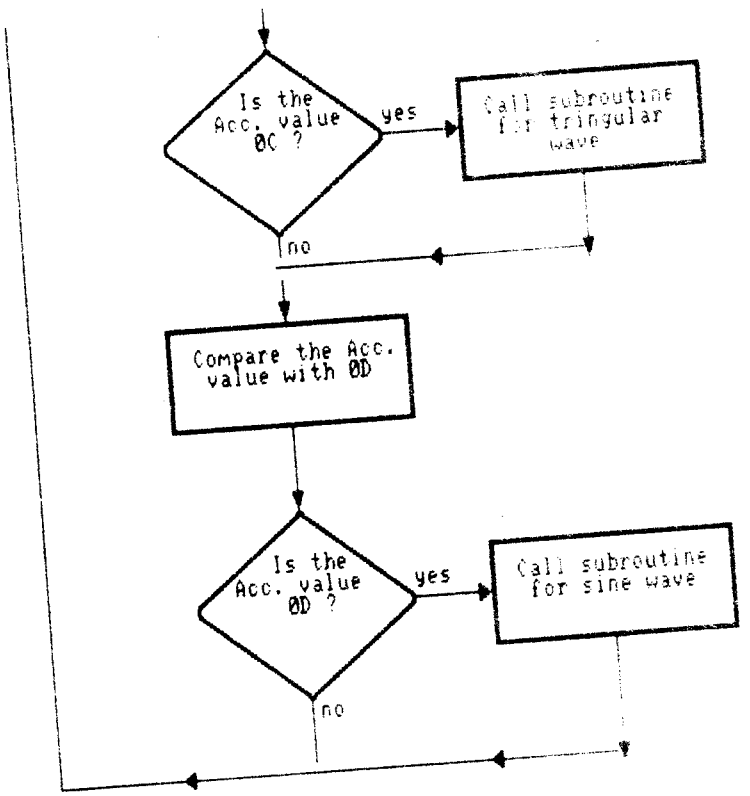
7.7.7. UNIPOLAR POSITIVE BIAS SQUARE WAVE

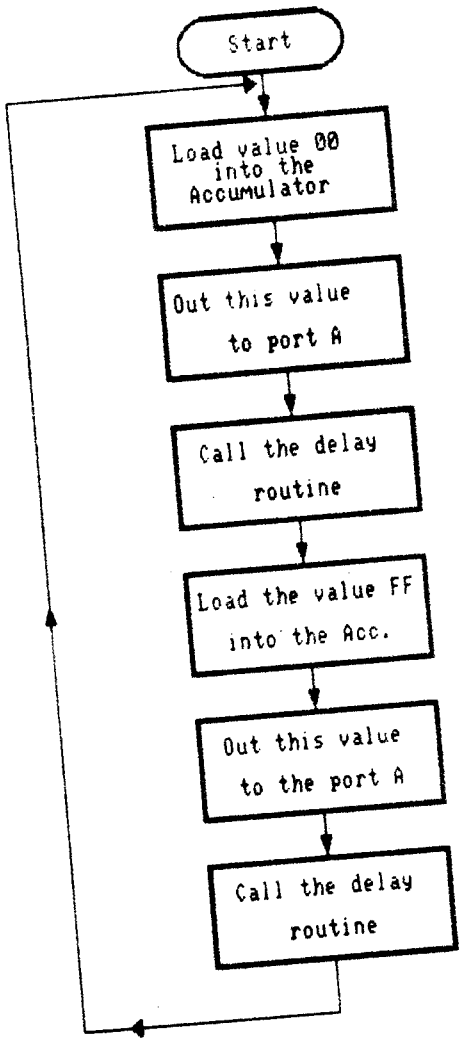
ADDRESS	INSTR.	LABLE	OPCODE	OPEREND	COMMENTS
4800	3E,80	START:	MVI	A,80	Move immediate accumulator with value 80
4802	D3,OC		OUT	OC	Out to Port A
4804	CD,50,47		CALL	4750	CALL DELAY
4807	3E,FF		MVI,A	FF	Move accumulator with value FF
4809	D3,OC		OUT	OC	Out Port A
480B	CD,50,47		CALL	4750	CALL DELAY
480E	C3,80,48		JMP	START:	

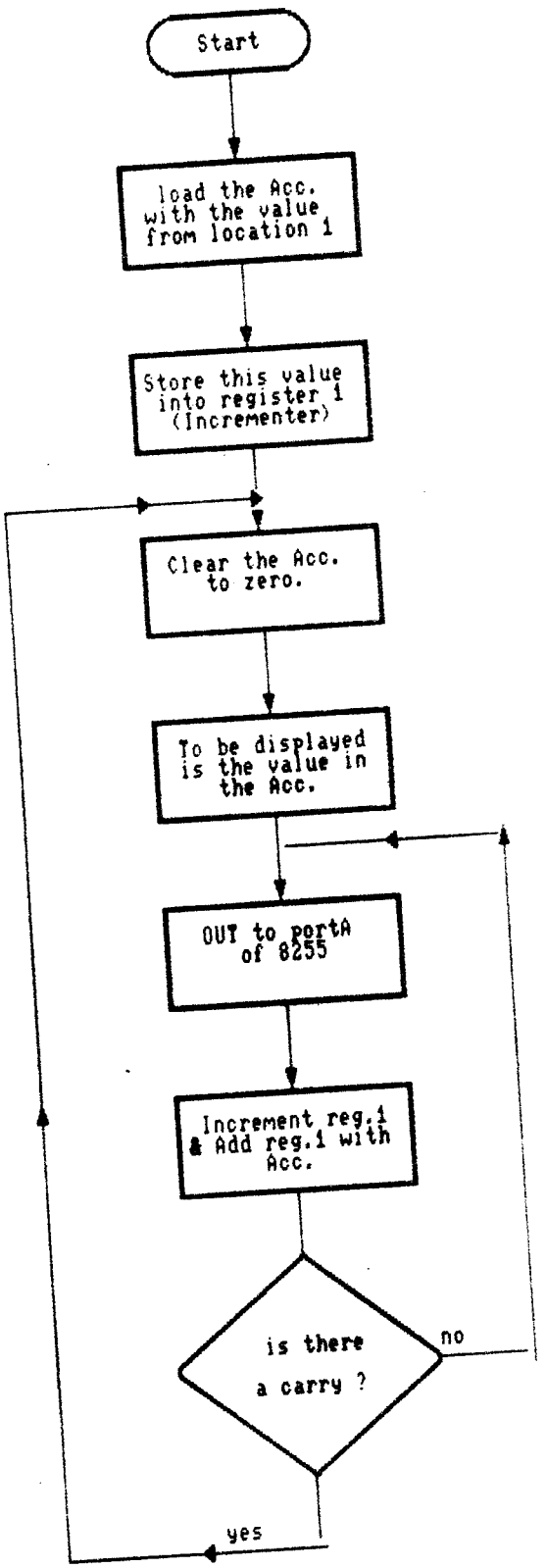
7.7.8 DELAY PROGRAM

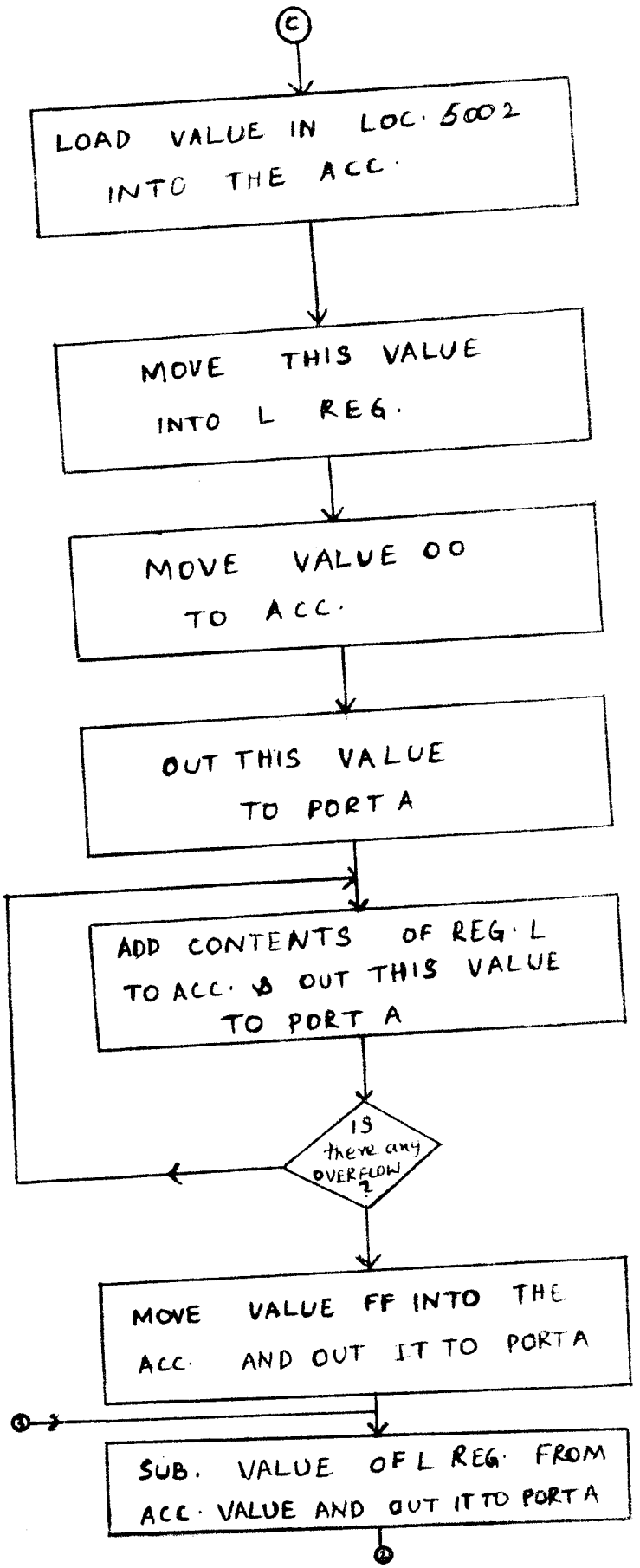
ADDRESS	INSTR.	LABLE	OPCODE	OPEREND	COMMENTS
4750	3A,02,50		LDA	5002	Load accumulator with value 5002.
4753	4F		MOV	C,A	
4754	OD	DEC:	DCR	C	Decrement register C
4755	C2,54,47		JNZ	DEC	Jump to location specified
4758	C9		RET		Return

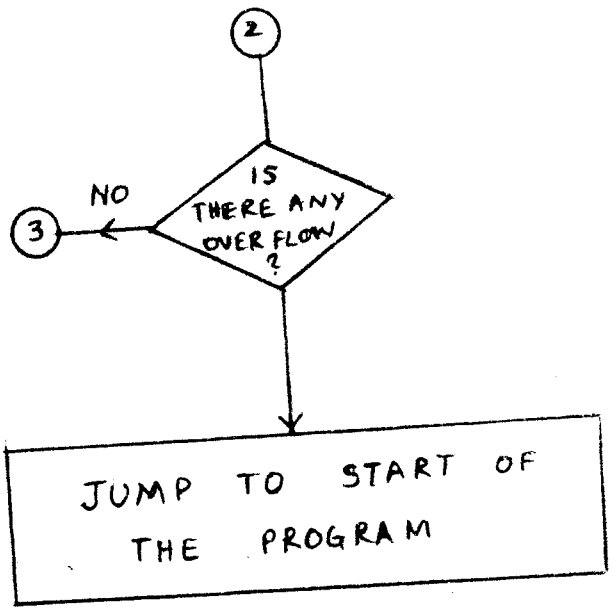

Start
Initialize the 8255 ports.
Load the Acc. with the value from 5000
Compare the Acc. value with 0A
Is the value in the Acc. 0A ?
yes
Call subroutine for sawtooth wave
no
Compare the Acc. value with 0B
Is the Acc. value 0B ?
yes
Call subroutine for square wave
no
Compare the Acc. value with 0C

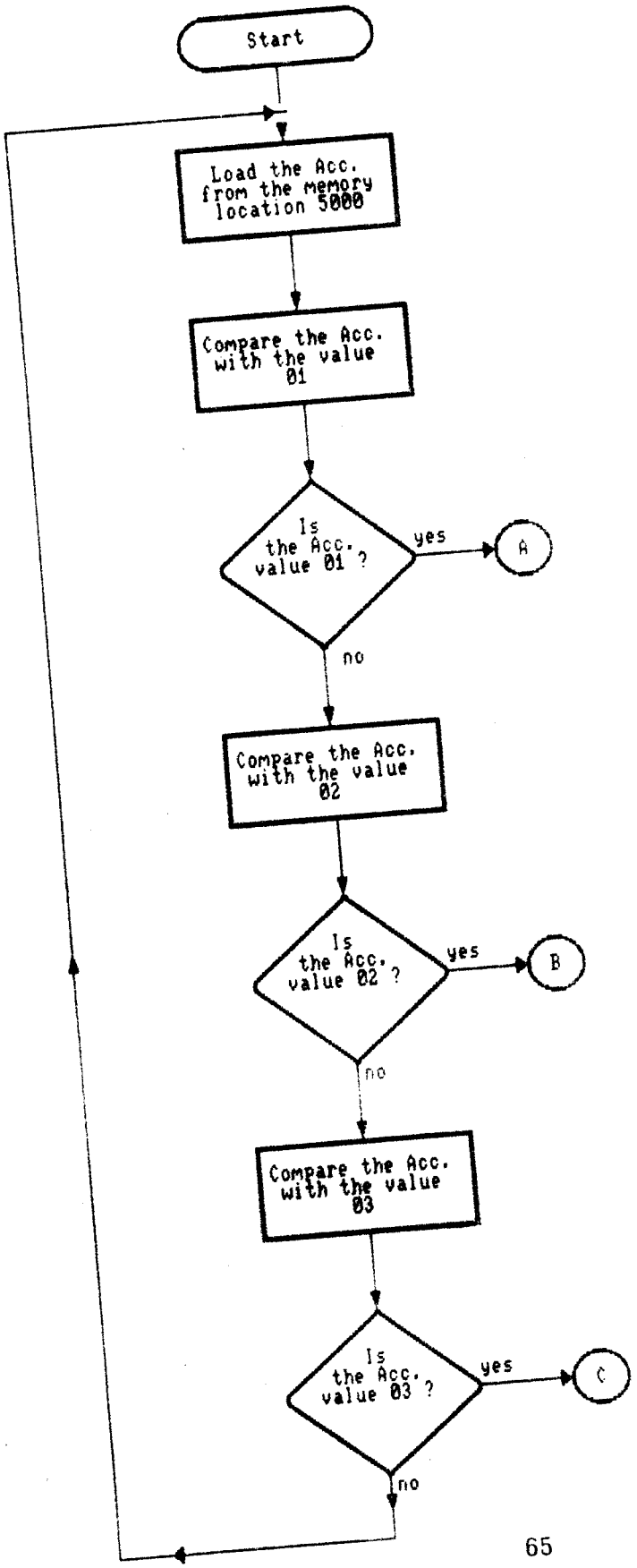

Is the Acc. value 0C ?
yes
Call subroutine for tringular wave
no
Compare the Acc. value with 0D
Is the Acc. value 0D ?
yes
Call subroutine for sine wave
no

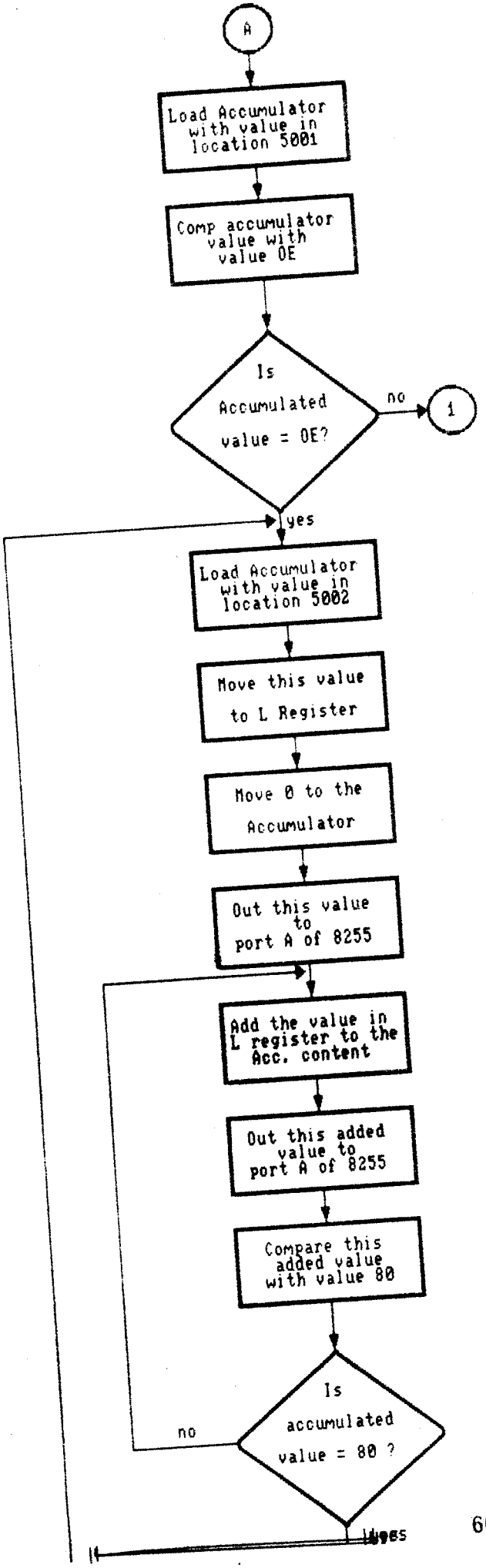

Start
Load value 00 into the Accumulator
Out this value to port A
Call the delay routine
Load the value FF into the Acc.
Out this value to the port A
Call the delay routine

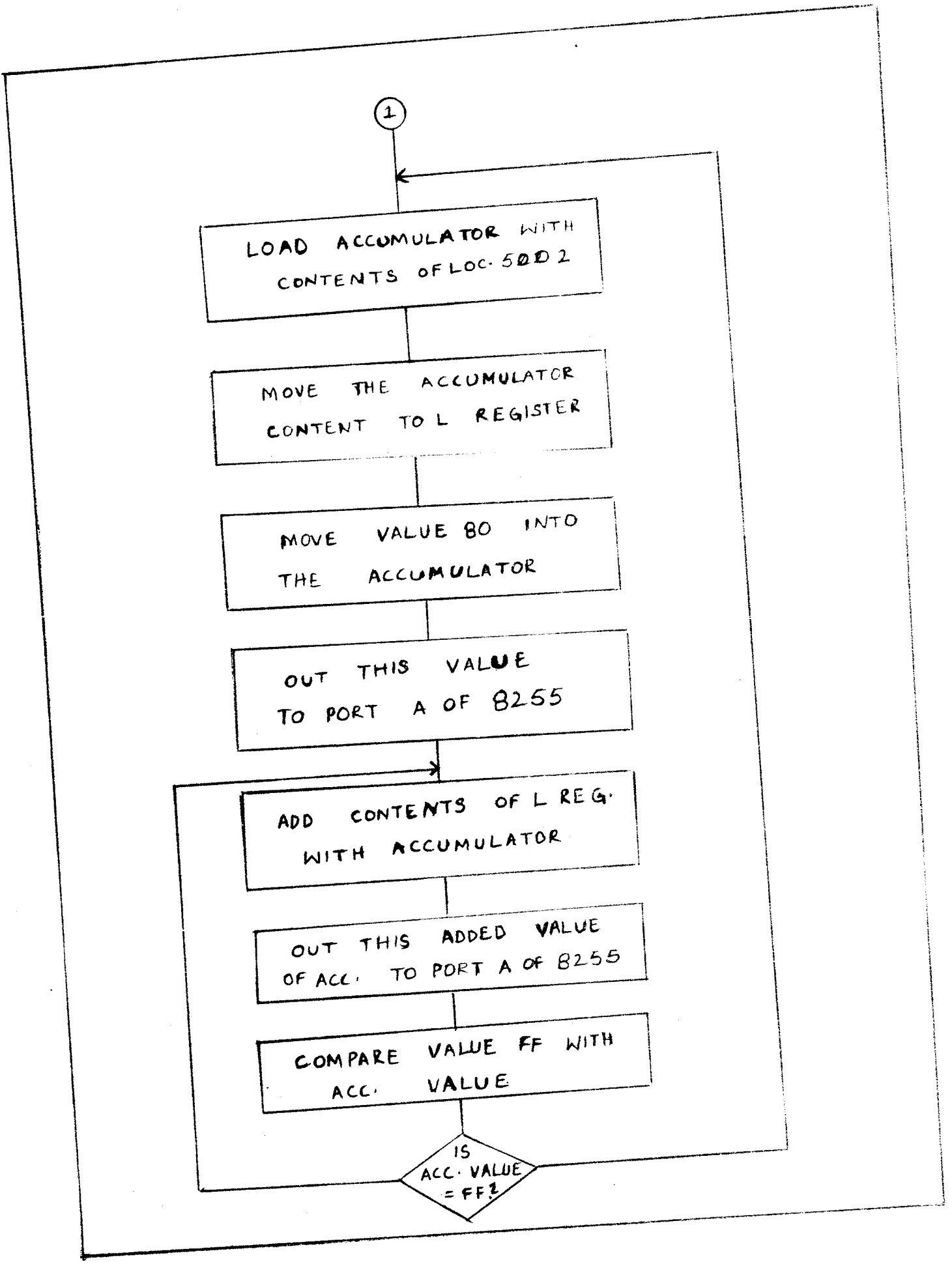

Start
load the Acc. with the value from location 1
Store this value into register i (Incrementer)
Clear the Acc. to zero.
To be displayed is the value in the Acc.
OUT to portA of 8255
Increment reg.i & Add reg.i with Acc.
is there a carry ?
no
yes

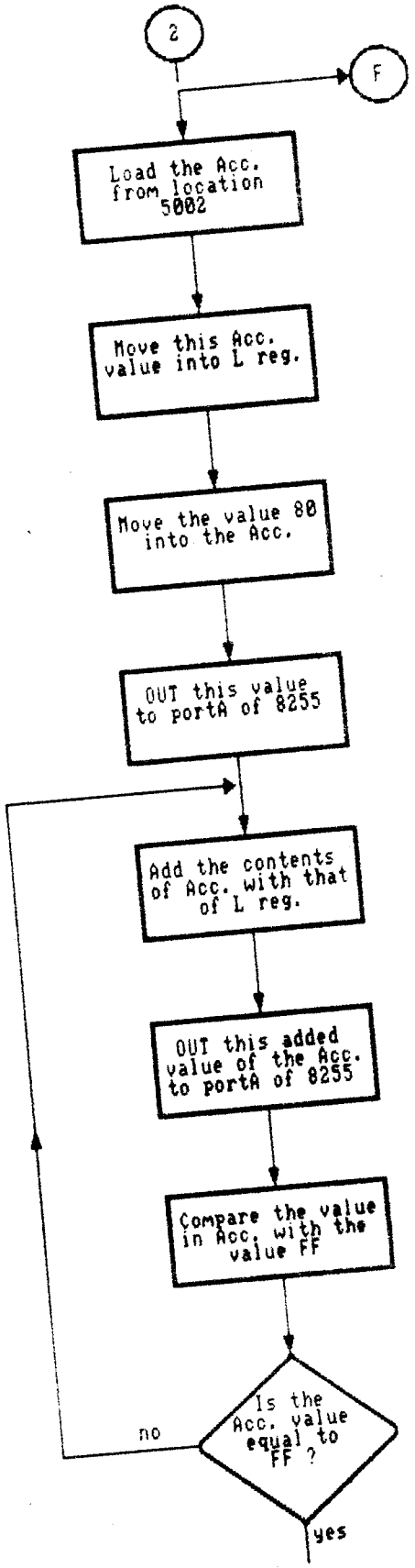

C
LOAD VALUE IN LOC. 5002 INTO THE ACC.
MOVE THIS VALUE INTO L REG.
MOVE VALUE 00 TO ACC.
OUT THIS VALUE TO PORT A
ADD CONTENTS OF REG. L TO ACC. & OUT THIS VALUE TO PORT A
IS there any OVERFLOW ?
MOVE VALUE FF INTO THE ACC. AND OUT IT TO PORT A
SUB. VALUE OF L REG. FROM ACC. VALUE AND OUT IT TO PORT A

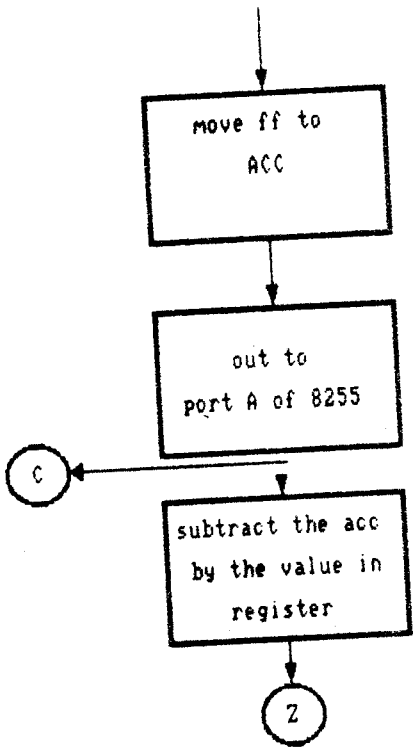

2
IS THERE ANY OVER FLOW ?
NO
3
JUMP TO START OF THE PROGRAM

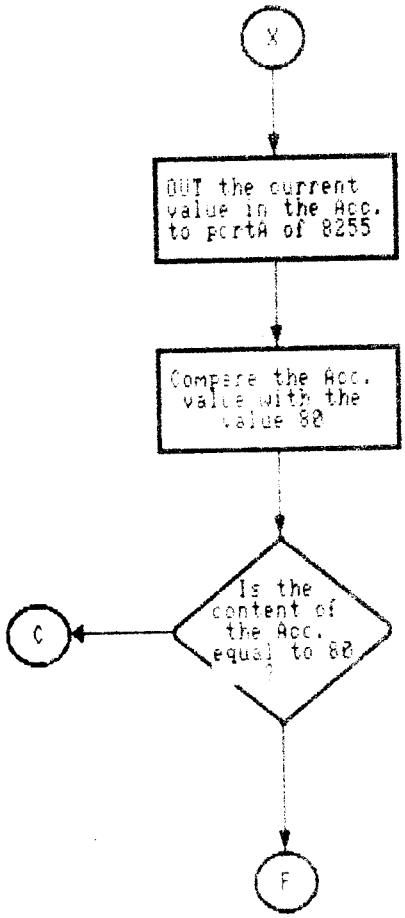

Start
Load the Acc. from the memory location 5000
Compare the Acc. with the value 01
Is the Acc. value 01 ?
yes
A
no
Compare the Acc. with the value 02
Is the Acc. value 02 ?
yes
B
no
Compare the Acc. with the value 03
Is the Acc. value 03 ?
yes
C
no

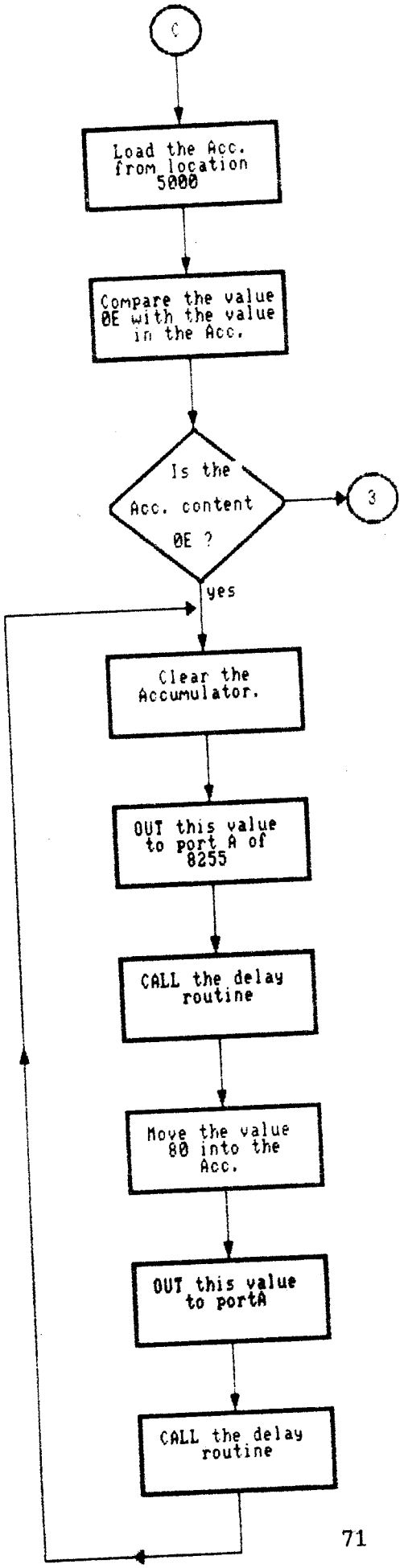

A
Load Accumulator with value in location 5001
Comp accumulator value with value 0E
Is Accumulated value = 0E?
no
1
yes
Load Accumulator with value in location 5002
Move this value to L Register
Move 0 to the Accumulator
Out this value to port A of 8255
Add the value in L register to the Acc. content
Out this added value to port A of 8255
Compare this added value with value 80
Is accumulated value = 80 ?
no
yes

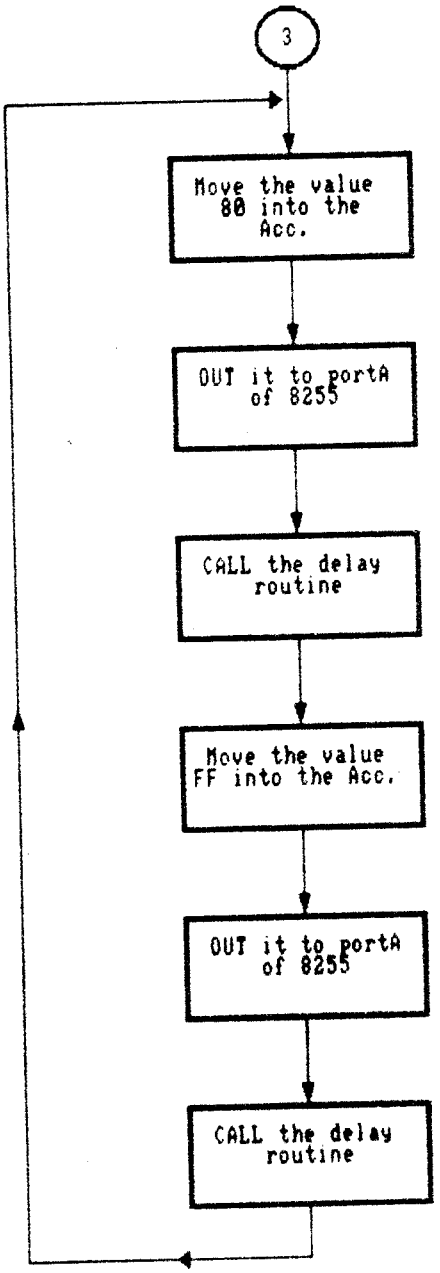

1
LOAD ACCUMULATOR WITH CONTENTS OF LOC. 5002
MOVE THE ACCUMULATOR CONTENT TO L REGISTER
MOVE VALUE 80 INTO THE ACCUMULATOR
OUT THIS VALUE TO PORT A OF 8255
ADD CONTENTS OF L REG. WITH ACCUMULATOR
OUT THIS ADDED VALUE OF ACC. TO PORT A OF 8255
COMPARE VALUE FF WITH ACC. VALUE
IS ACC. VALUE = FF?


2
F
Load the Acc. from location 5002
Move this Acc. value into L reg.
Move the value 80 into the Acc.
OUT this value to portA of 8255
Add the contents of Acc. with that of L reg.
OUT this added value of the Acc. to portA of 8255
Compare the value in Acc. with the value FF
Is the Acc. value equal to FF ?
no
yes


move ff to
ACC
out to
port A of 8255
C
subtract the acc
by the value in
register
Z


X
OUT the current value in the Acc. to portA of 8255
Compare the Acc. value with the value 80
Is the content of the Acc. equal to 80
C
F


C
Load the Acc. from location 5000
Compare the value 0E with the value in the Acc.
Is the Acc. content 0E ?
3
yes
Clear the Accumulator.
OUT this value to port A of 8255
CALL the delay routine
Move the value 80 into the Acc.
OUT this value to portA
CALL the delay routine


3
Move the value 80 into the Acc.
OUT it to portA of 8255
CALL the delay routine
Move the value FF into the Acc.
OUT it to portA of 8255
CALL the delay routine

REFERENCES

1. DIGITAL COMPUTER ELECTRONICS, AN INTRODUCTION TO MICROCOMPUTERS - MALVINO

2. MICROPROCESSOR ARCHITECTURE, PROGRAMMING AND APPLICATIONS WITH THE 8085/808A - RAMESH S. GAONKAR

3. INTRODUCTION TO MICROPROCESSORS - ADITYA P. MATHUR

4. INTRODUCTION TO MICROPROCESSORS - DOUGLAS V. HALL

5. DIGITAL INTEGRATED ELECTRONICS - TAUB AND SHILLING

9. CONCLUSION

Hence, for the software the inputs must be given, i.e. after the execution of program for comparison, the user has to specify the wave he wants, namely, Sawtooth wave, Triangular Wave, Square wave, Ramp wave, Sine wave, with codes as OAH, OBH, OCH, ODH respectively. For a sine wave there is no need to enter the above code. And the user has to specify the frequency which is given the hexcode, in a loop-up table specified for a particular range of frequency. Hence, the two inputs, first the type of wave and next the frequency in hexcode is to be entered in a specified location say 5000, so that the o/p wave form can be derived from P1, of the DAC card, which is interfaced to 8085 microprocessor through a 8255A, programmable peripheral interface. The control word for 8255A is 80H.

For generation of sine wave forms, a program starts from a specified location with a fixed input table which gives the ouput with a fixed frequency, which can be watched in CRO, through Pin P1. If the user wants to have a wave with a frequency other than the specified one say he wants to give his own look up table according to the sample one given herewith.

For a unipolar wave generator, the software starts from a specified location in the memory, after execution, the user has to specify the frequency and enter the hex code accordingly, forwhich the output can be watched through Pin P1 of DAC Card, again on the CRO screen.

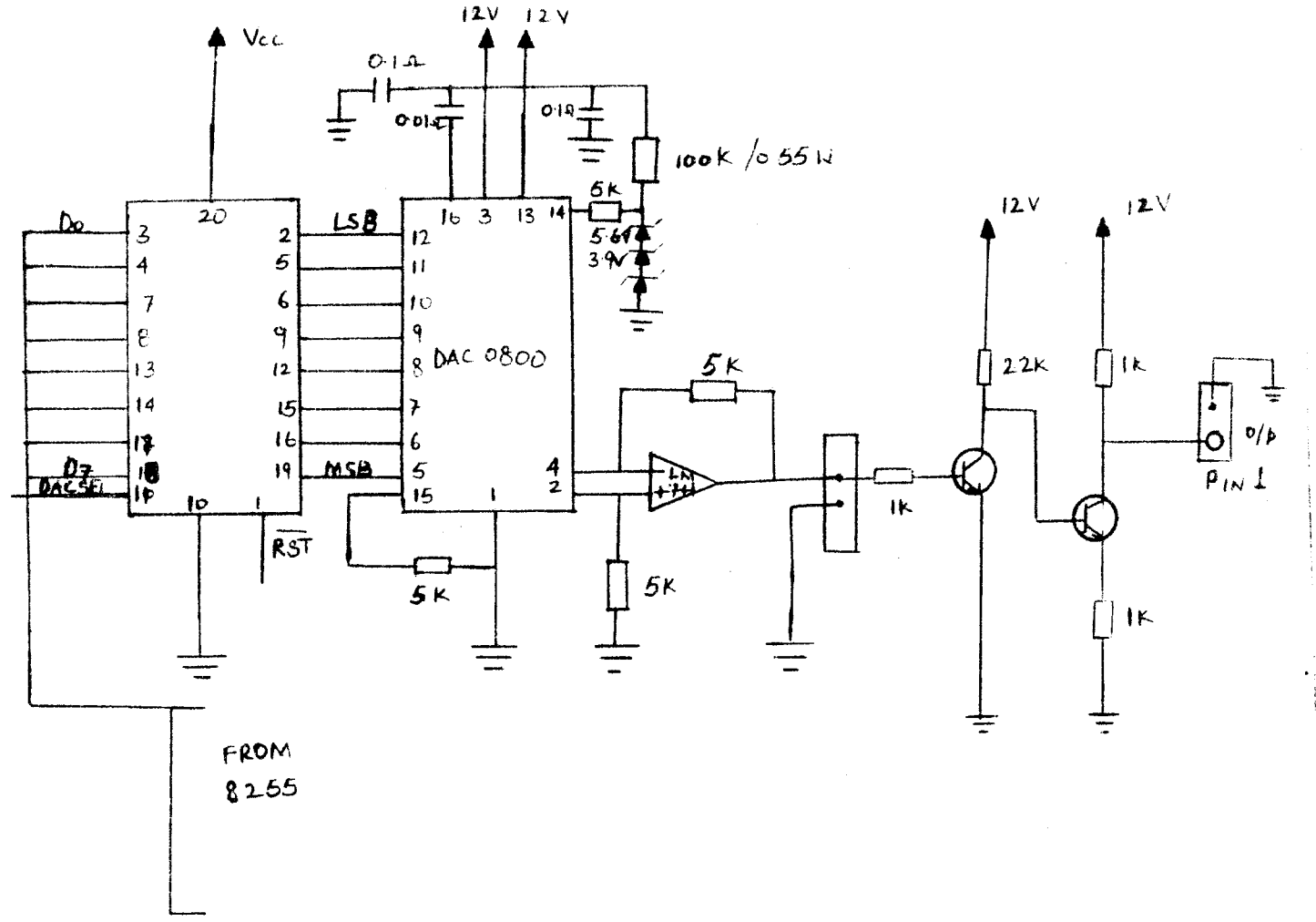
10. FURTHER EXPANSION

The hardware and software of this project can be expanded in isuch a way that it is automatic, i.e. after the ON of power supply, a program is run which resets and displays four codes namely, OA, OB, OC AND OD. When the user presses the appropriate code for the wave he wants, the Microprocessor recognizes it and asks for the required frequency. whether he wants a bipolar wave or an unipolar one. Once the user gives these three inputs the outputs can be viewed on a CRO.

* For a Ramp wave, the frequency remains a constant.

Thus, the **MICROPROCESSOR BASED WAVE GENERATOR** can be automated in a way that the user has to feed only 3 inputs and can view his required waves on the CRO.

DAC PART


Vcc
12V
12V
0.1μ
0.01μ
0.1μ
100K / 0.55W
5K
5.6V
3.9V
20
LSB
16
3
13
14
DAC 0800
12
11
10
9
8
7
6
5
15
1
4
2
MSB
D0
D7
DAC SEL
3
4
7
8
13
14
17
18
19
2
5
6
9
12
15
16
10
1
RST
5K
5K
5K
1K
22K
1K
1K
12V
12V
o/p
PIN 1
FROM
8255