

# 1 GHz FREQUENCY METER CARD FOR PERSONAL COMPUTERS

P-135

PROJECT REPORT

SUBMITTED IN PARTIAL FULFILMENT OF THE  
REQUIREMENTS FOR THE AWARD OF THE DEGREE OF  
**BACHELOR OF ENGINEERING IN**  
**ELECTRICAL AND ELECTRONICS ENGINEERING**  
OF THE BHARATHIAR UNIVERSITY  
COIMBATORE-641046

SUBMITTED BY

**Kirtivasan, G.**

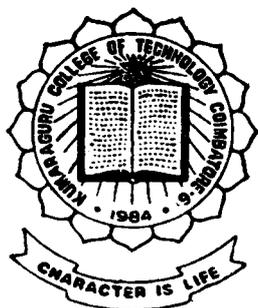
**Sivakumar, D.**

**Venkatesh Babu, K.**

UNDER THE GUIDANCE OF

**Miss B. Uma Maheswari B.E.**

**Miss G. Poongodi B. E.**



Department of Electrical and Electronics Engineering  
**KUMARAGURU COLLEGE OF TECHNOLOGY**

COIMBATORE - 641 006

1990 - 91



## ACKNOWLEDGEMENT

On the offset we would like to thank our guides **Miss. B. UMAMAHESWARI B.E.**, and **Miss. G. POONGODI B.E.**, who had been instrumental for the progress and successful completion of this project work. Without their timely advice and able guidance we could not have orchestrated such an endeavour.

We owe a massive dept to **DR. K.A. PALANISWAMY B.E., M.E., Ph.D(Roorkee), C.Eng., M.I.S.T.E., F.I.E.**, Professor and Head of the Department of Electrical and Electronics Engineering, for generously providing us with all the necessary infrastructure in carrying out this project, and motivating us to do our best.

We are greatly obliged to our Principal Prof. **R. PALANIVEL B.E. M.SC(Engg.), F.I.E, M.I.S.T.E, M.I.S.P.R.S,** who was a moving force behind the successful completion, from start to stop.

We express our profound gratitude to **MR. S. KESAVAN M.E,** and **Mr. S. RAJU B.E.**, of CCIT for being kind enough, in allowing us to have unhindered access to the computers of CCIT.

Last but not the least, we sincerely thank the teaching and non teaching staff of EEE Dept. of K.C.T for rendering their total co-operation and constructive criticisms.

## CONTENTS

Chapter	Description	Page No.
	<b>LIST OF FIGURES AND TABLES</b>	1
	<b>SYNOPSIS</b>	3
	<b>INTRODUCTION</b>	4
<b>1.</b>	<b>FREQUENCY MEASUREMENT</b>	<b>5</b>
1.1	Advantages Of Digital Frequency Measurement	5
1.2	Digital Versus Analog Frequency Measurement	6
<b>2.</b>	<b>DIGITAL FREQUENCY MEASUREMENT</b>	<b>8</b>
2.1	Basic Digital Frequency Measuring Unit	8
2.2	Clock Oscillator	8
2.3	TTL Clock	11
2.4	Counters And Dividers	12
2.5	Flip — Flops	14
2.6	JK Flip — Flop	14
2.7	Asynchronous Counters	17
2.8	Synchronous Counter	19
<b>3.</b>	<b>CIRCUIT HARDWARE OPERATION DETAILS</b>	<b>31</b>

<b>Chapter</b>	<b>Description</b>	<b>Page No.</b>
4.	<b>PCB AND COMPONENTS DETAILS</b>	37
4.1	Printed Circuit Board (P.C.B)	37
4.2	8255A Programmable Peripheral Interface	37
4.3	Monostable With Input Logic	45
4.4	74LS04	48
4.5	74LS02	49
4.6	74LS32	50
4.7	74LS30	51
4.8	IC4040	52
4.9	SN 74LS245	53
4.10	CD4066	54
4.11	MM74HC393	57
5.	<b>EXPANSION SLOTS IN A PC</b>	67
6.	<b>DESCRIPTION OF SOFTWARE</b>	71
6.1	Software Procedure For Reading Data	71
6.2	GW BASIC Program that Controls the Frequency Meter Card	77
	<b>CONCLUSION</b>	78
	<b>APPENDIX</b>	80
	<b>BIBLIOGRAPHY</b>	82

## LIST OF FIGURES AND TABLES

Fig No.	Description	Page No.
2.1	Basic Frequency measuring unit	21
2.2	Clock Wave forms	22
2.3	TTL Clock Circuit	23
2.4	(i) NAND gate latch	24
	(ii) One way to implement a JK Flip-Flop	24
	(iii) Truth table for JK flip flop	24
2.5	JK Flip - Flop Symbols	25
2.6	At $J = K = 1$	26
2.7	(i) 3 Bit Asynchronous counter	27
	(ii) Waveforms at output of asynchronous counter	28
2.8	(i) Mod-8 Parallel binary counter	29
	(ii) Truth table of synchronous counter	30
	(iii) Waveforms of synchronous counter	30
3.0	Circuit diagram of the frequency meter card for PCs	36
4.1	Printed circuit Board layout (Upper and Lower layers)	59
4.2	(i) Block diagram of 8255 PPI	61
	(ii) Pin details of 8255 PPI	62
4.3	(i) 8255 mode set control word format	63
	(ii) Port C bit set/Reset control word	63
4.4	(i) Logic diagram & truth table of IC 74121	64
	(ii) Negative triggering	65
	(iii) Positive triggering	65

	( iv ) Triggers on every pulse at B	66
	( v ) Triggers on every other pulse set B	66
5.1	Component lay out diagram for IBM PC mother board	68
5.5.2	Block diagram of circuitary on IBM PC mother board	69
5.3	Pin names and numbers for peripheral slots on IBM PC mother board	70
6.1	Flow charts of the control program of frequency meter card for PCs	
	( i ) Description of measurement principle	74
	( ii ) Commands in measurement procedure	75
	( iii ) Measurement procedure in GWBASIC	76

## SYNOPSIS

This is a frequency meter card, capable of measuring frequencies upto 1GHz. The value of the frequency thus measured is displayed on the CRT screen of a personal computer. It can accomodate only TTL compatible input signal and non TTL compatible signals should be suitably converted to TTL compatible signals andthen inputted. The other feature of this project is that it can be directly plugged into the motherboard of a PC, thus eliminating any external interfacing cord connection from the card to the PC. A software written in GWBASIC controls the operation of the hardware circuit. This software can be aptly amended and the plug in card can be used as a counter also.

## INTRODUCTION

Frequency is a parameter which is associated with almost all electronic & electrical circuits. Precise and accurate measurement of frequency is extremely important in such circuits. A wrong reading of frequency due to unreliable and inaccurate meters could lead to malfunction of the system and in cases even to system failure.

Frequency can be measured by analog and digital methods. Analog methods of measuring frequency is hardly used in industrial and scientific applications. Here, digital methods of frequency measurement is employed as it totally eliminates errors on account of human factors.

One way of measuring frequency employing digital technique would be to interface the hardware components to a P.C. where in the frequency could be read by a software and displayed on the CRT screen of the P.C.

Our project employing the above method is a cost-efficient alternative to a stand-alone digital frequency meter. Simple to build and program, this meter has high sensitivity and is capable of measuring a wide range of frequency.

## CHAPTER 1

### FREQUENCY MEASUREMENT

An analog frequency meter displays the frequency in terms of deflection of a pointer. i.e. an analog displacement or an angle corresponding to the frequency. While digital measurements gives the measurand in decimal number. The digital instruments work on the principle of quantization. The analog quantity to be measured is quantized into a number of small intervals upto many small decimal places. The objective of the digital instrument is then to determine in which portion of the subdivision the measurand can thus be identified as an integral multiple of the smallest unit called the quantum, chosen for subdivision. Thus the measuring procedure reduces to one of counting the number of quanta present in the measurand.

The counter circuit or chip is used for counting. The counter output is fed to the ports of a Programmable Peripheral Interface Chip which acts as a buffer and later is fed to a computer where we can read the output or obtain the printout at certain time intervals.

#### 1.1 Advantages of digital frequency measurement

- i) The computer screen gives the value of the frequency directly in decimal numbers and therefore errors on account of human factors like errors due to parallax and approximation are eliminated.

- ii) The readings are carried to any number of significant figures by merely positioning the decimal point.
- iii) The output being in digital form and therefore the O/P may be directly fed to memory devices like tape recorders, printers for storage.
- iv) The power requirement is less.

## 1.2 Digital versus analog frequency measurement

- 1) **Accuracy** - Digital instruments can be made very accurate unlike analog where frequency measuring instruments are rated usually within  $\pm 0.1$  per cent of full scale.
- 2) **Reaction to environment** - Analog frequency measuring instruments operate under a wide range of environments. Whereas digital instruments are relatively complex and sensitive to temperature and humidity. However no mechanical movements are involved in digital instruments and hence errors are less.
- 3) **Resolution** - This is sometimes referred to as readability below which differences can no longer be differentiated. In analog instruments the limit is one part in several hundreds. Digital instruments can be made with a resolution of one part in several thousands.

4) **Power requirement** - Digital instruments draw only negligible power, where as the analog instruments may heavily load the circuit under measurement and thus indicate an erroneous reading. Digital instruments have I/P impedances of the order of 10 Mega ohms or even higher.

5) **Cost and Portability** - Analog instruments are extremely portable and usually do not requires an outside source of supply for measurements. Analog instruments are low in cost and can be moved from one place to another with ease. On the other hand digital instruments are not easily portable and require an external source of power. However in the meter card that we have fabricated the circuit is portable and moderate in cost.

6) **Range and polarity** - This meter card incorporates automatic polarity and range indication. (Automatic Polarity selection and auto-ranging facilities) which reduces operator training, measurement error and possible damage through overloads.

7) **Free from observational errors** - As explained earlier the reading of the frequency is in decimal in the screen of the P.C. They directly indicate the frequency being measured.

Analog instruments causes parallax and observational errors.

## CHAPTER 2

### DIGITAL FREQUENCY MEASUREMENT

#### 2.1 Basic digital frequency measuring unit

Fig 2.1 shows a basic digital frequency measuring unit. The clock oscillator usually employs a quartz crystal as R C & L C oscillators are unreliable due to variation in the parametric value of resistors, inductors and capacitors because of temperature & external magnetic fields. The clock oscillator produces square wave of very high frequency (in Mega hertz). The oscillator frequency is then divided by a divider circuit. i.e the time period of the square wave is raised to a known value. Let us assume that the clock frequency is divided & the time period raised to one second. This wave is called as gate enable. Let the input signal have a frequency of 750 Hz. Then the unknown waveform will have logic 1 state 750 times in 1 second. The AND gate will then have an output of logic 1 750 times during one logic high period of gate enable. The output from AND gate is counted by a counter & the output can be obtained in the screen of a computer or in a LED display unit.

The gate need not necessarily be an AND gate. The output pulses can be counted using a NOR gate also where the logic 0 level is taken into account. The only criteria here is that the output of the gate should in some way convey the frequency of the input wave to the counter.

#### 2.2 Clock Oscillator

The heart of every digital system is the digital clock it does, indeed, provide the heart beat without which the system would cease

to function. We shall consider the characteristics of a digital clock circuit & the clock signal.

Since all logic operations in a digital system occur in synchronism with a clock, the system clock becomes the basic timing unit. The system clock must provide a periodic waveform that can be used as a synchronizing signal. The clock signal need not be a perfectly symmetrical square wave. The wave can be asymmetrical square wave. But however the main requirement is simply that the clock be perfectly periodic.

The clock defines a basic timing interval during which the logic operations must be performed. The basic timing interval is defined as a clock cycle time and is equal to one period of the clock waveform. Thus all logic elements, flip-flops, gates and so on must complete their transitions in less than one clock cycle time.

The meter card that we have fabricated uses the clock signal as a reference based on which the frequency of the input waveform is measured.

The clock levels must hold a steady value of +5v(high) & 0v(low). However in actual practice the stability of the clock is much more important than the absolute value of the voltage level. For instance it might be perfectly acceptable to have a high level of +4.8v instead of +5.0v provided it is steady, unchanging +4.8v.

The second characteristic deals with the time required for the clock to change from high to low or vice versa. The transition of the clock from low to high at point 'a' in Fig 2.2(i) as shown by a vertical line segment. This implies a time of zero, i.e the transition occurs instantaneously -it requires zero time. The same is true of the transition from high to low at point 'b' in Fig 2.2(i). Thus an ideal clock has zero transition time.

A nearly perfect clock waveform on an oscilloscope might look as in Fig 2.2(ii). At first it would seem to be two horizontal traces composed of line segments. On closer examination, the vertical segments may not appear on the oscilloscope as the oscilloscope is not capable of fast transitions.

After reducing the 'sweep time' in the oscilloscope the waveform can be expanded such that the transition times are visible as in Fig 2.2(iii). Clearly it requires some time for the waveform to transit from low to high & vice versa. The time required for transition from low to high is  $t_r$  & that from high to low is  $t_f$ .

Finally the third requirement that defines an ideal clock is its frequency stability. The frequency of the clock should be steady and unchanging over a specified period of time. Short term stability can be specified by requiring that the clock frequency (or its period) not be allowed to vary by more than a given percentage over a short period of time- say a few hours.



The meter card requires a clock chip with short term rating as it would normally be operating for a short time.

Long term rating clocks are required in electronic watches, ovens having timer facilities etc, which should work for days, months or years without losing stability.

### 2.3 TTL Clock

An excellent TTL compatible clock circuit is shown in Fig 2.3. This clock circuit is well known and is widely used. Two inverters are used to construct a two stage amplifier with an overall phase shift  $360^\circ$ . A portion of the signal at 6 is fed back by means of a crystal to 1 and the circuit oscillates at a frequency determined by the crystal. Since the feedback element is a crystal the frequency of oscillation is stable. Here's how the oscillator works.

Inverter - 1 has a 330 ohm feedback resistor ( $R_1$ ) connected from output (2) to input (1). This forms a current to voltage amplifier with a gain of  $A_1 = V_{out}/I_{in} = -R_1$ . In this case the gain is -330 V/A where the negative sign indicates a  $180^\circ$  phase shift. An increase in 1mA in  $I_m$  will cause a negative - going voltage of 1mA x 330 = 330m V at V out.

Inverter - 2 is connected exactly as in inverter 1, Its gain is  $A_2 = R_2$ . The two amplifier are then a-c coupled with a  $0.01 \mu F$

capacitor to form an amplifier that has an overall gain of  $A = A_1 \times A_2 = R_1 R_2$ . The overall gain is +ve as the phase shift is  $360^\circ$ .

A series mode crystal is used as the feedback element to return a portion of the signal at 6 to 1. The crystal acts as a series RLC circuit and at resonance it ideally appears as a low-resistance element with no phase - shift. The feedback signal must therefore be at resonance and the two inverters in conjunction with the crystal form an oscillator operating at the crystal resonant frequency.

With the overall feedback element connected the overall gain is sufficient to drive each inverter between saturation and cut off and the output signal is a periodic waveform as shown in fig 2.3. Typically the output clock signal will transition between 0 to +5v and will essentially be a square wave. The frequency of the oscillator is determined by the crystal, and values between 1 and 20 MHz are common.

Inverter 3 is used as an output buffer amplifier and is capable of driving a load of 330 ohms in parallel with 100 p.F. while still providing rise & fall times of less than 10 ns.

## **2.4 Counters & dividers**

Counters and dividers are probably one of the most useful and versatile subsystem in a digital system. A counter driven by a clock can be used to count the number of clock cycles. Since the clock pulses occur at known intervals, the counter can be used as an instrument for

measuring time and therefore period or frequency. There are basically two different types of counters - synchronous and asynchronous.

The ripple counter is simple & straight forward in operation and construction and usually requires a minimum of hardware. It does, however, have a speed limitation. Each flip-flop is triggered by the previous flip flop, and thus the counter has a cumulative settling time. Counters such as these are called serial or asynchronous.

An increase in speed of operation can be achieved by use of parallel or synchronous counter. Here, every flip-flop is triggered by the clock (in synchronism) and thus settling time is simply equal to the delay time of a single flip - flop. The increase in speed is usually obtained at the price of increased hardware.

Serial and parallel counters are used in combination to compromise between speed of operation and hardware count. Serial, parallel or combination counters can be designed such that each clock pulse advances the contents of the counter by one; it is then operating in a count-up mode. The opposite is also possible; the counter then operates in the count-down mode. Furthermore many counters can be either 'cleared' so that every flip-flop contains a zero, or preset such that the contents of the flip-flop represent any desired binary number.

## 2.5 Flip-flops

The main component of a counter & divider are the flip flops. Flip-flop and is a circuit whose output remain unchanged, once set, even if there is a change in input. Flip flop is used to store a binary number.

A counter / divider circuit used JK flip-flop. The characteristics of a JK flip and its operation is described below.

## 2.6 JK Flip-flop

JK flip flop is the ideal element to use in digital counter. Fig 2.4(ii) shows one way to build a JK flip flop. The variable J and K are called control inputs because they determine what the flip-flop does when a positive clock edge arrives. The RC circuit has a short time constant, thus converting the rectangular clock pulse into narrow spikes.

The logic block of a NAND gate latch as shown in fig 2.4(i).

When J and K are both low, both AND gates are disabled. Therefore, clock pulses have no effect. The first possibility is the initial entry in the truth table. As shown when J & K are both 0s, q retains its last value.

When J is low & Q is high, the upper gate is disabled, so there's no way to set the flip flop. The only possibility is reset when Q is high, the lower gate passes a RESET trigger as soon as the next positive clock edge arrives. This forces Q to become low ( the second entry

in the truth table). Therefore  $J = 0$  and  $K = 1$  means that the next clock edge resets the flip - flop (unless  $Q$  is already reset).

When  $J$  is high and  $K$  is low, the lower gate is disabled, so it's impossible to reset the flip - flop. But we can set the flip-flop as follows. When  $Q$  is low,  $\bar{Q}$  is high; therefore, the upper gate passes a SET trigger on the next positive clock edge. This drives  $Q$  into the high state ( the third entry in the truth table). It can be seen that  $J = 1$  and  $K = 0$  means that the next positive clock edge sets the flip flop (unless  $Q$  is already high).

When  $J$  and  $K$  are both high (this is a forbidden state with an RS flip -flop) it's possible to set or reset the flip - flop. If  $Q$  is high, the lower gate passes a RESET trigger on the next positive clock edge. On the other hand, when  $Q$  is low, the upper gate passes a SET trigger on the next positive clock edge. Either way,  $Q$  changes to the complement of the last state (see the truth table). Therefore  $J = 1$  and  $K = 1$  means the flip flop will toggle on the next positive clock edge. 'Toggle' means to switch to the opposite state.

propagation delay prevents the JK flip flop from racing (toggling more than once during a positive clock edge). Here's why. In Fig 2.4(ii) the outputs change after the positive clock edge has struck. By then, the new  $Q$  and  $\bar{Q}$  values are too late to coincide with the positive spikes driving the AND gates. For instance if  $t_p = 20$  ns, the outputs change approximately 20 ns after the leading edge of the clock. If the clock spikes are narrower than 20ns, the returning  $Q$  and  $\bar{Q}$  arrive too

late to cause false triggering.

Fig 2.5(i) shows a symbol for a JK flip flop of any design. When seen this on a schematic diagram, remember that on the next positive clock edge, (1) low J and K have no effect, (2) low J and high K produce a reset high J and low K produce a set, and high J and K result in a toggle.

We can include OR gates in the design to accommodate PRESET and CLEAR. Fig 2.5(ii) shows the symbol for a JK flip flop with PR and CLR. It is positive edge triggered.

Fig 2.5(iii) is also another commercially available JK flip flop. It is negative edge triggered and requires low PR to set it or low CLR to reset it.

At  $J = 1$  and  $K = 1$ , the flip - flop simply toggles each time the clock goes low. The waveform at Q has a period twice that of the CLK waveform. In other words the frequency of Q is only one - half that of CLK. This circuit acts as a frequency divider-the output frequency is equal to the input frequency divided by 2. Fig 2.6 gives the connection and the waveforms.

Now that we have seen the operation of a flip-flop we can now take a look at how counters and dividers use the flip flops and operate.

We shall briefly see the working of asynchronous counters and synchronous counters.

## 2.7 Asynchronous counters

A binary ripple counter can be constructed by use of clocked JK flip-flops. Fig 2.7(i) shows three master-slave, JK flip-flops connected in cascade. The system clock, a square wave, drives flip-flop A. The output of A drives B, and the output of B drives flip-flop C. All the J and K inputs are tied to  $+V_{CC}$ . This means that each flip-flop will change state (toggle) with a negative transition at its clock input.

When the output of a flip - flop is used as the clock input for the next flip-flop, we call the counter a ripple counter or asynchronous counter. The A flip - flop must change states before it can trigger the B flip - flop, and the B flip flop has to change states before it can trigger the C flip flop. The triggers move through the flip-flops like a ripple in water. Because of this, the overall propagation delay time is the sum of the individual delays. For instance if each flip - flop in this three flip - flop counter has a propagation delay time of 10 ns, the overall propagation delay time for the counter is 30 ns.

The waveforms given in Fig 2.8 (iii) show the action of the counter as the clock runs. Let's assume that the flip-flops are all initially reset to produce 0 outputs. If we consider A to be least significant bit (LSB) and C the most significant bit (MSB), we can say contents of the counter is CBA = 000.

Every time there is a negative-clock transition, flip-flop A will change states. Thus at point a on the time line, A goes high, at point b it goes back low, at C it goes back high and so on. Notice that the waveform at the output of flip-flop A is one half the clock frequency.

Since A acts as the clock for B, each time the waveform at A goes low, flip-flop B will toggle. Thus at point b on the time line, B goes high, it then goes low at point d and toggles back high again at point f. Notice that the waveform at the output of flip-flop B is one - half the frequency of A and one - fourth the clock frequency.

Since B acts as the clock for C, each time the waveform at B goes low, flip-flop C will toggle. Thus C goes high at point d on the time line and goes back low again at point h. The frequency of the waveform at C is one - half that at B, but it is only one - eighth the clock frequency.

Therefore if we consider the O/P ABC where A is the LSB and C the MSB then the output ABC will give a value which is the no. of highs counted. In other words ABC output is the output of a counter. By cascading many JK flip-flops we can count many pulses. The circuit as a divider can be studied from the waveform in fig. 2.7(ii). The frequency gets divided in each stage of JK flip-flop. The cascaded flip-flops can be used as a divider and division depends upon the no. of flip-flops used.

## 2.8 Synchronous Counters

The ripple counter is the simplest to build, but there is a limit to its highest operating frequency. Each flip-flop has a delay time. In ripple counter these delay times are additive, and the total 'settling' time for the counter is approximately the delay time times the total number of flip flops.

Furthermore, there is the possibility of glitches occurring at the output of decoding gates used with a ripple counter. Both of these problems can be overcome by the use of a synchronous or parallel counter. The main difference here is that every flip-flop is triggered in synchronism with the clock.

The construction of one type of parallel binary counter is shown in fig. 2.8 (i) along with the truth table and waveforms for the natural count sequence. Since each state corresponds to an equivalent binary number (or count), we refer to each state as a count from now on. The basic idea here is to keep the J and K inputs of each flip-flop high, such that the flip-flop will toggle with any negative clock transitions at its clock input. We then use AND gates to gate every second clock to flip-flop B, every fourth clock to flip-flop C, and so on. This logic configuration is often referred to as "Steering logic" since the clock pulse are gated or steered to each individual flip-flop.

The clock is applied directly to flip-flop A. Since the JK flip-flop used responds to a negative transition at the clock input and toggles when both the J and K inputs are high, flip-flop A will change state with each negative clock transition.

Whenever A is high, AND gate X is enabled and a clock pulse is passed through the gate to the clock input of flip-flop B. Thus B changes state with every other negative clock transition at points b, d, f and h on the time line.

Since AND gate Y is enabled and will transmit the clock to flip-flop C only when both A and B are high, flip-flop C changes state with every fourth negative clock transition at points d and h on the time line.

Examination of the waveforms and the truth table reveals that this counter progresses upward in a natural binary sequence from count 000 up to count 111, advancing one count with each negative clock transition. Fig. 2.8 (ii) illustrates the operation as a counter and fig.2.8(iii) the operation as a divider.

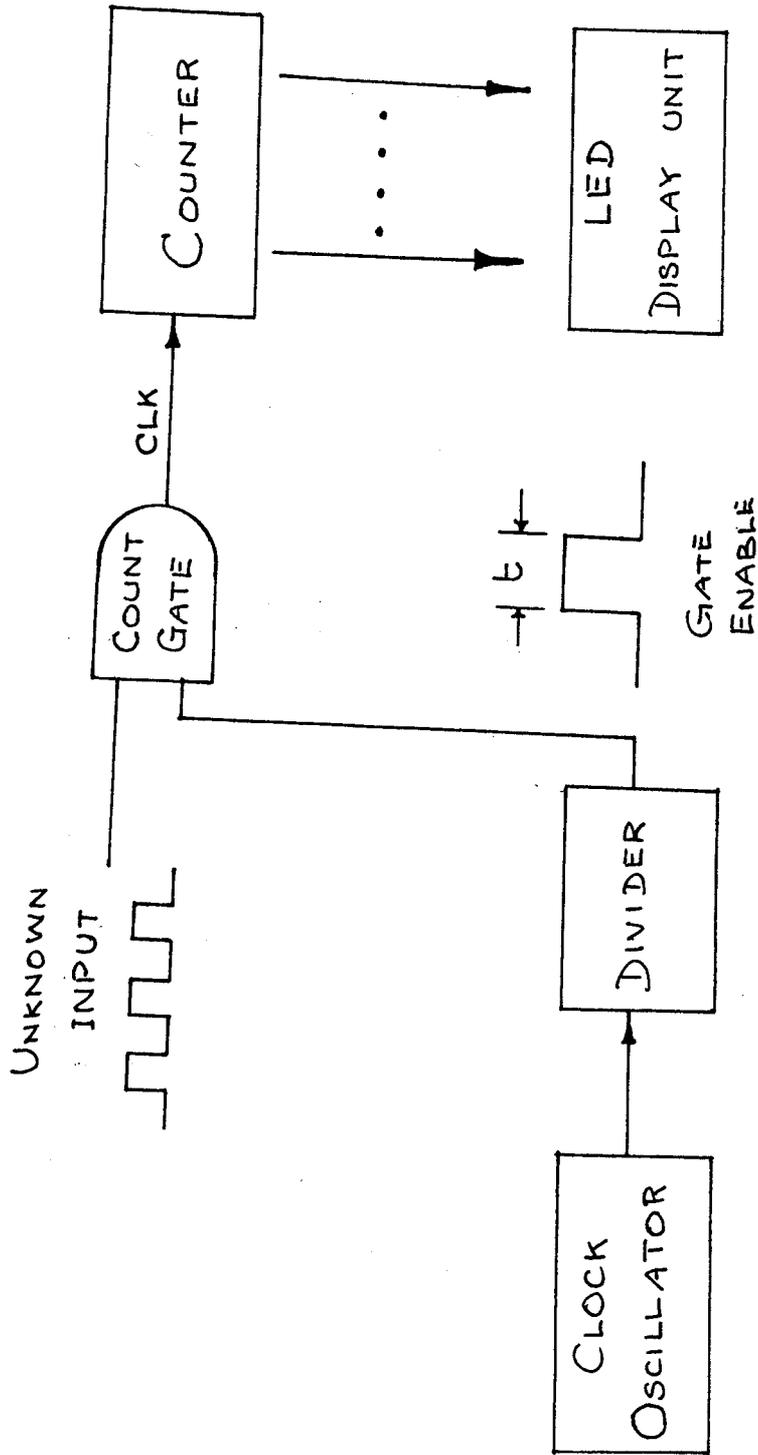
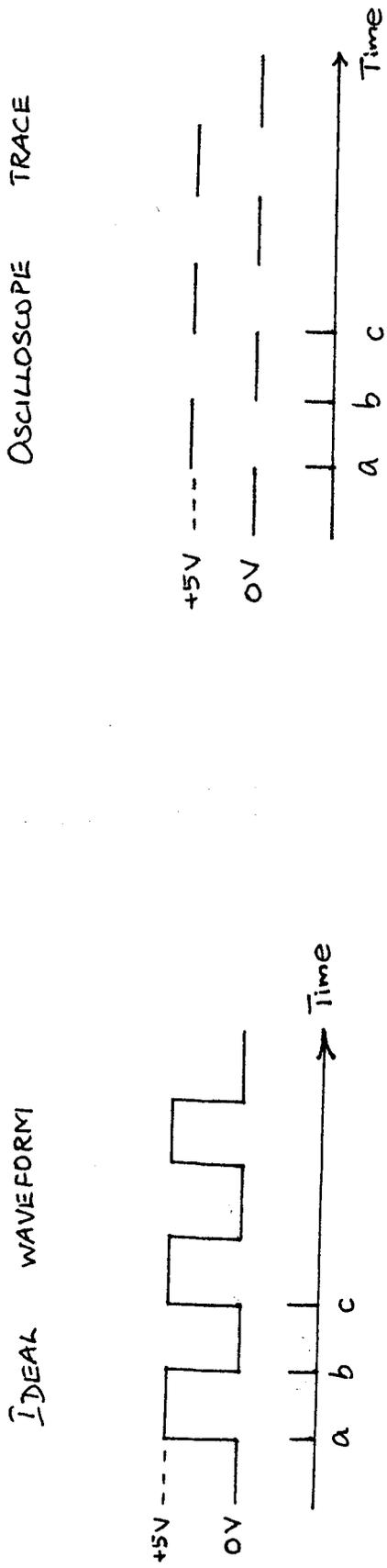


Fig 2.1 Basic Frequency Measuring Unit



EXPANDED OSCILLOSCOPE TRACE

(ii)

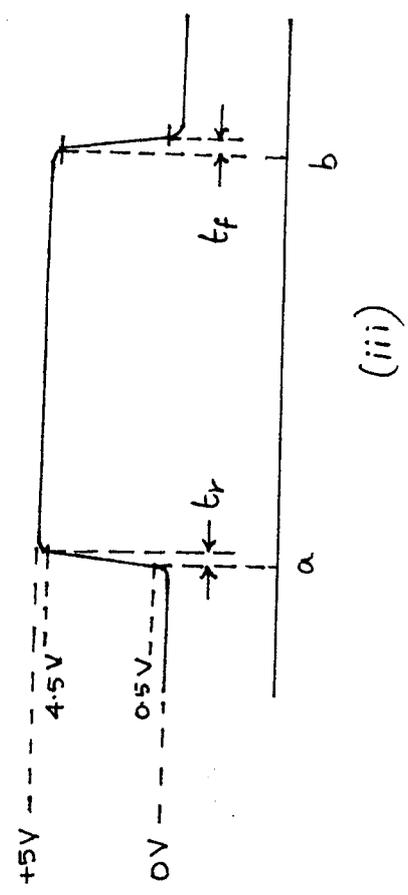


Fig 2.2 Clock Wave forms

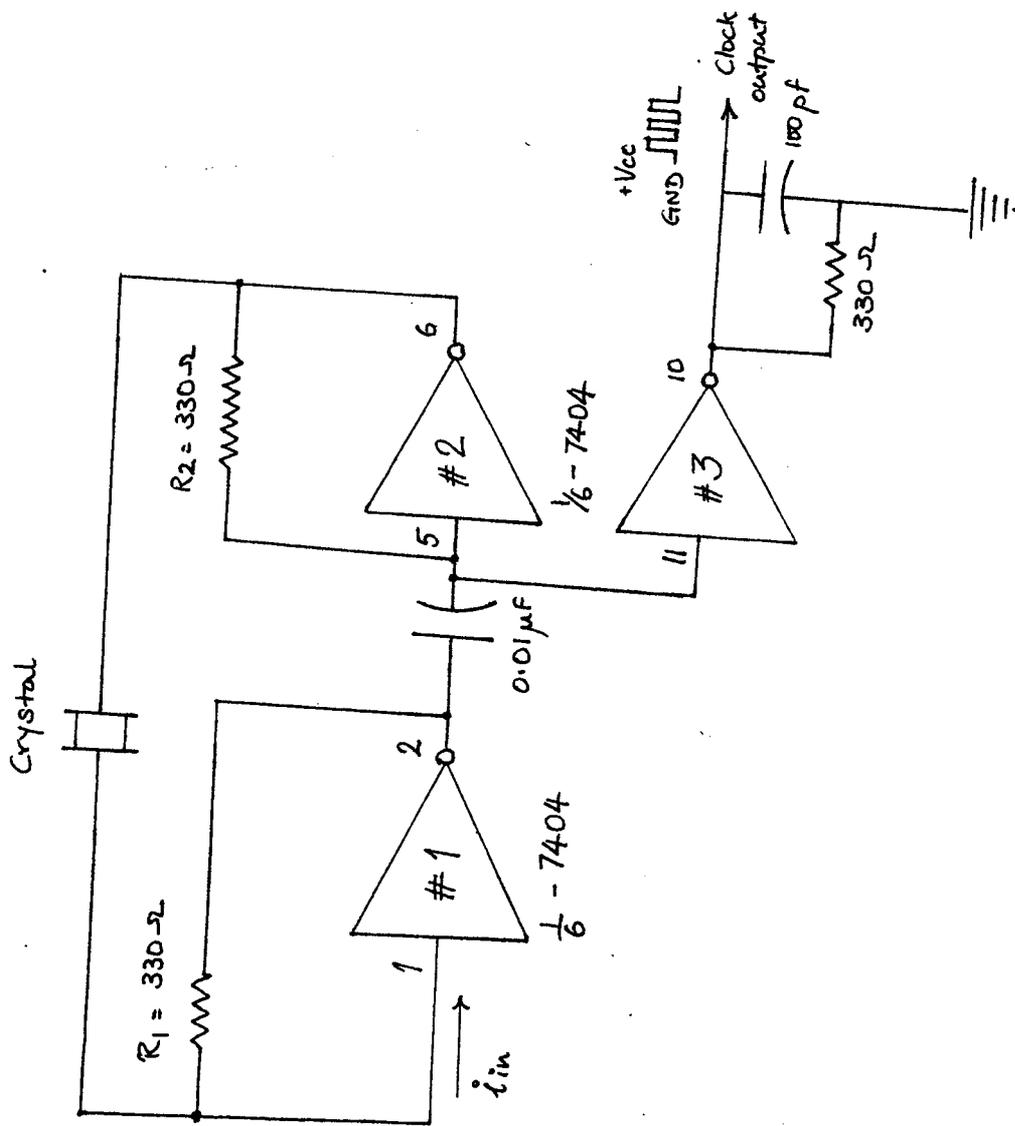


Fig 2.3 TTL Clock Circuit

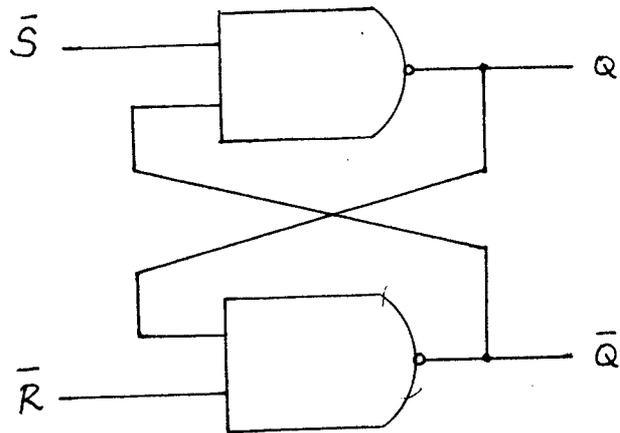


Fig 2.4(i) NAND Gate Latch

CLK	J	K	$Q_{n+1}$
x	0	0	$Q_n$ (last state)
↑	0	1	0
↑	1	0	1
↑	1	1	$\bar{Q}_n$ (toggle)

Fig 2.4(iii) Truth table for JK Flip flop

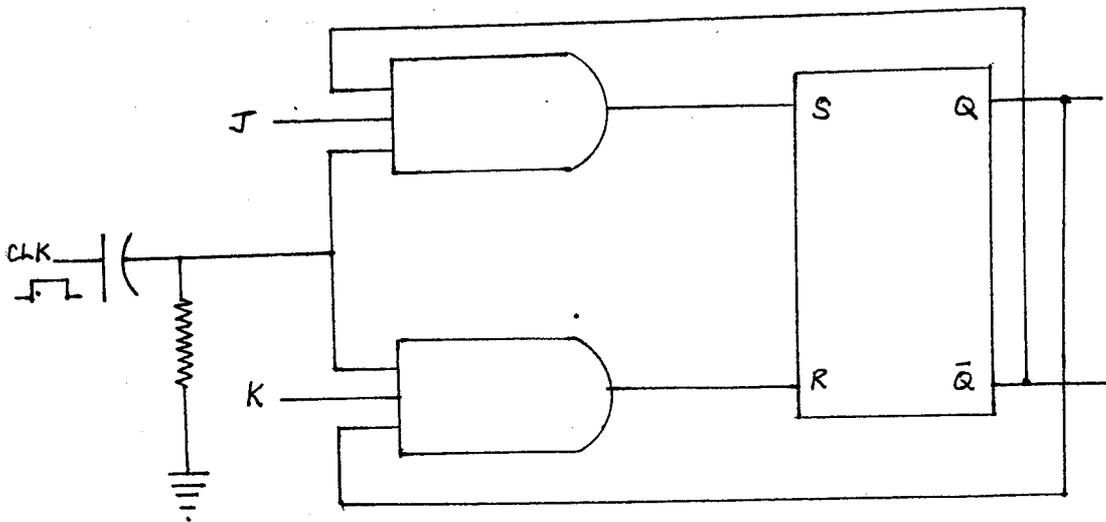
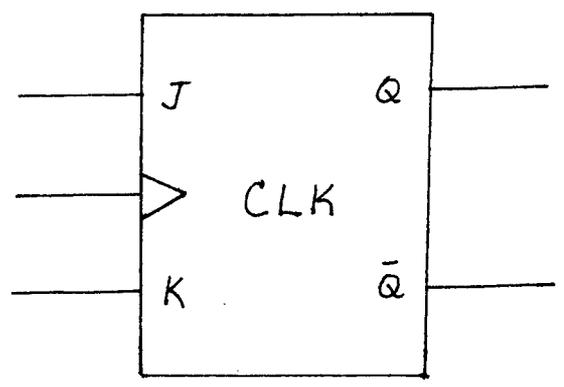


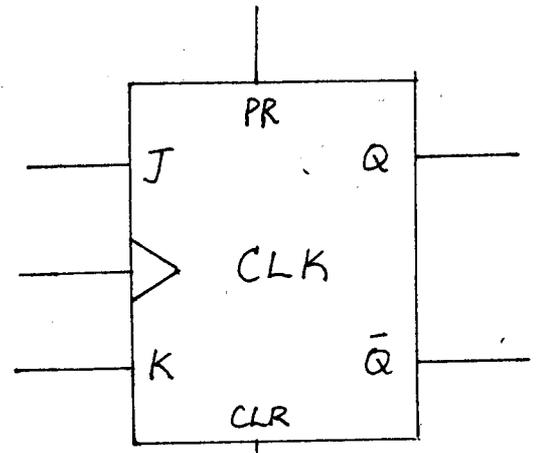
Fig 2.4(ii) One Way to Implement a JK Flip-Flop

BASIC SYMBOL



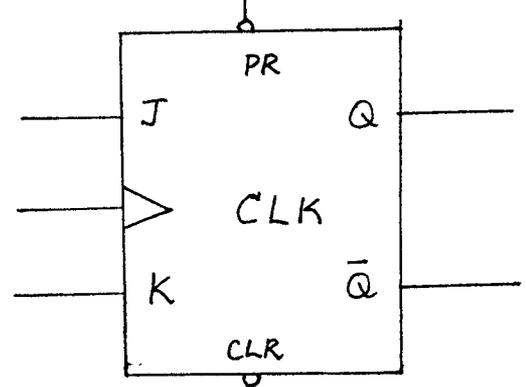
(i)

POSITIVE PRESET AND CLEAR



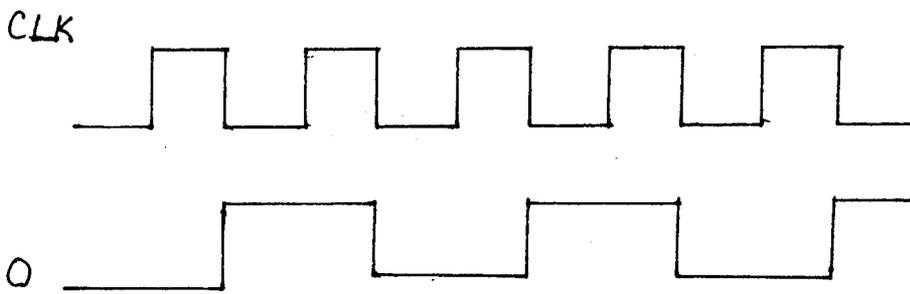
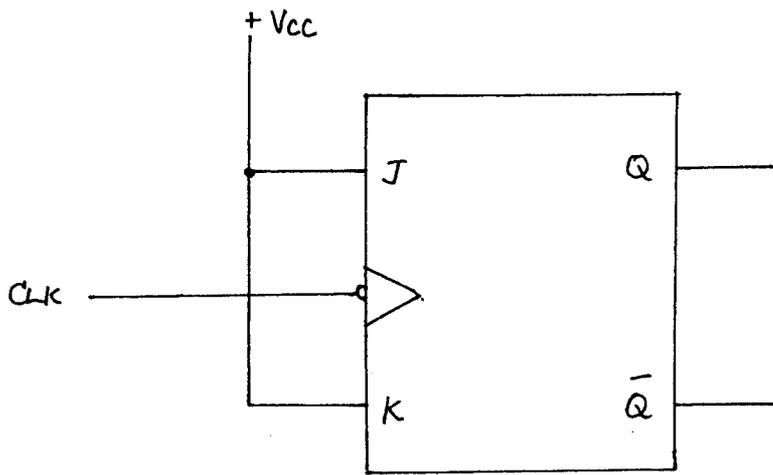
(ii)

INVERTED PRESET AND CLEAR

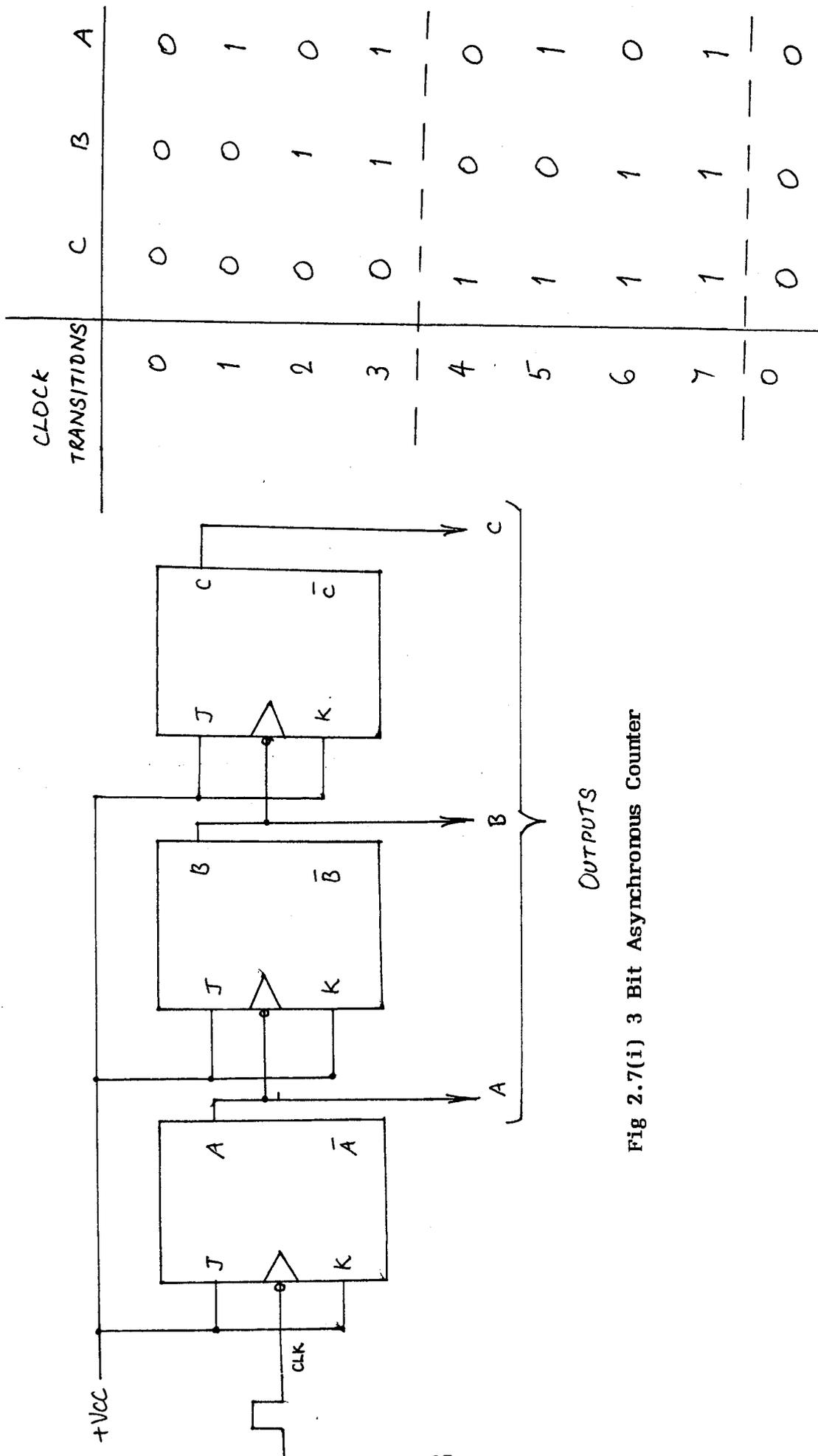


(iii)

Fig 2.5 JK Flip - Flop Symbols



**Fig 2.6 At  $J=K=1$**



OUTPUTS

Fig 2.7(i) 3 Bit Asynchronous Counter

Fig 2.7(iii) Truth Table for asynchronous counter

CLOCK TRANSITIONS	C	B	A
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
0	0	0	0

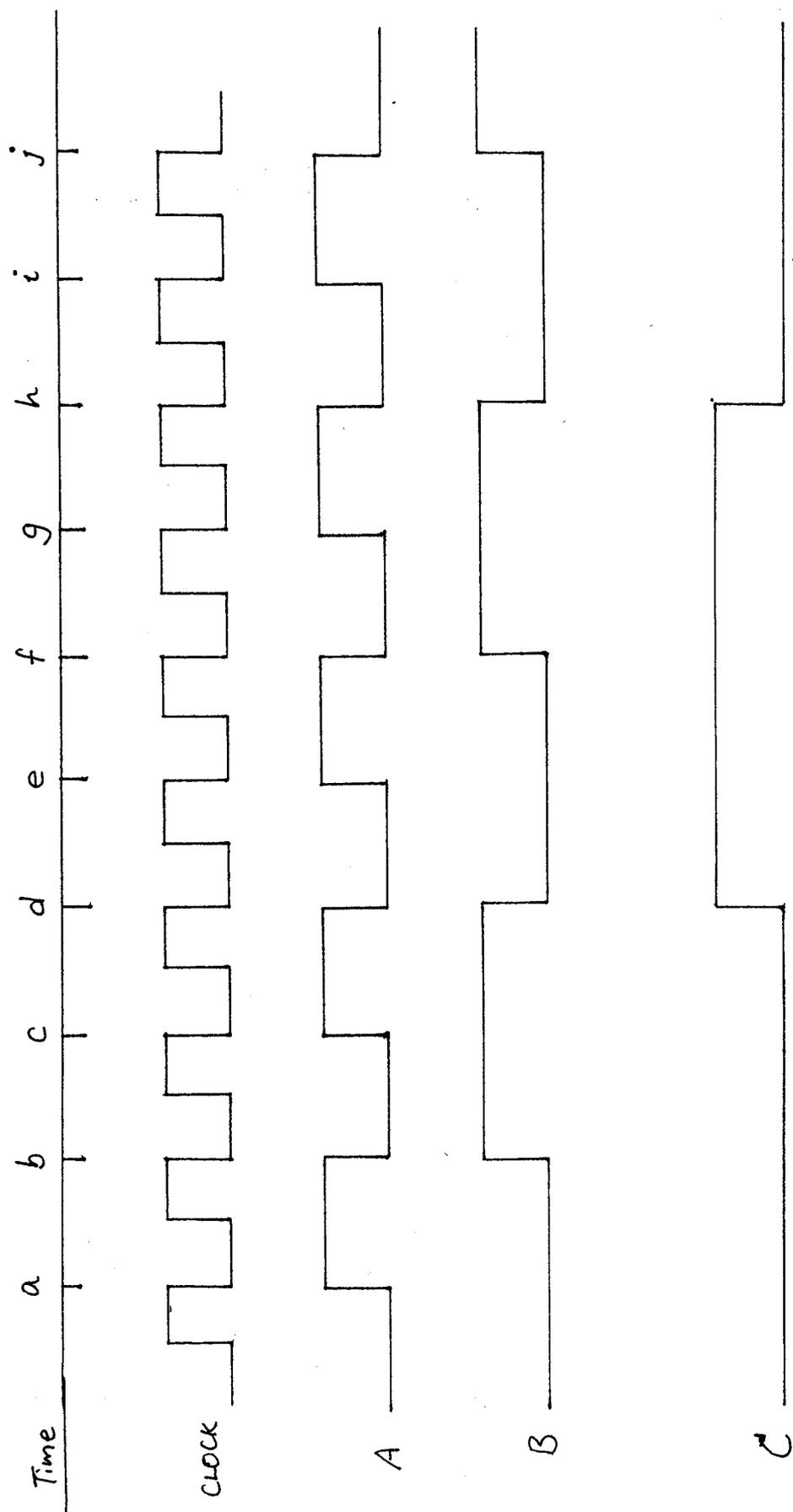


Fig 2.7(ii) Wave forms at output of asynchronous counter

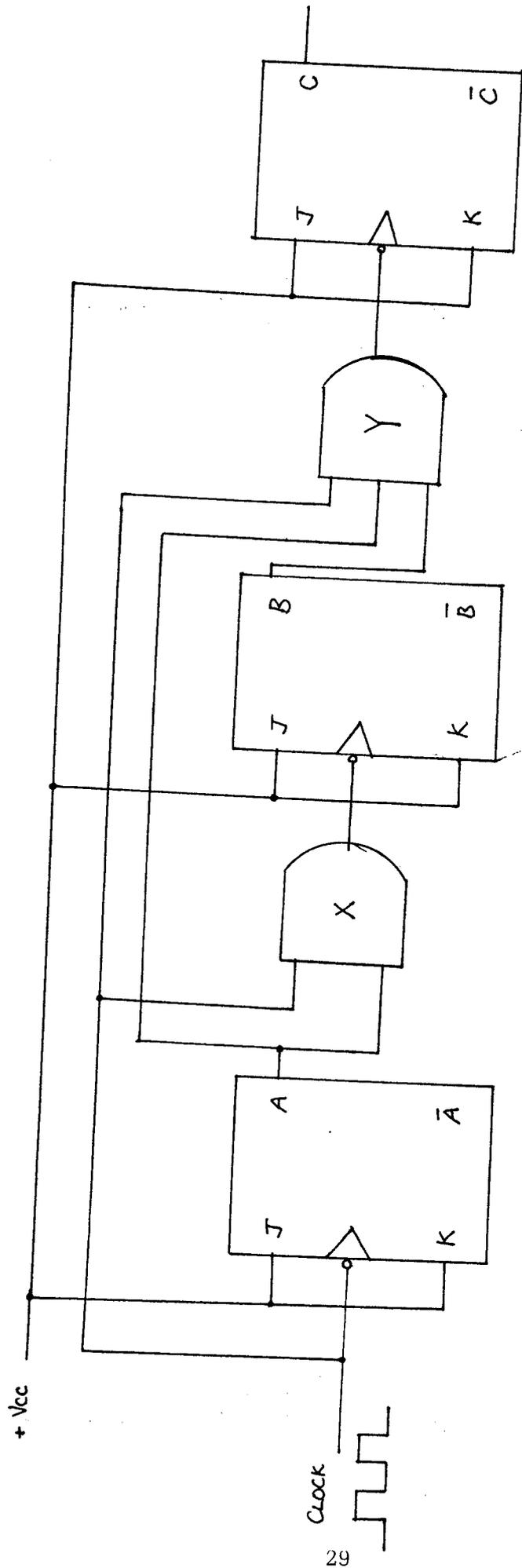
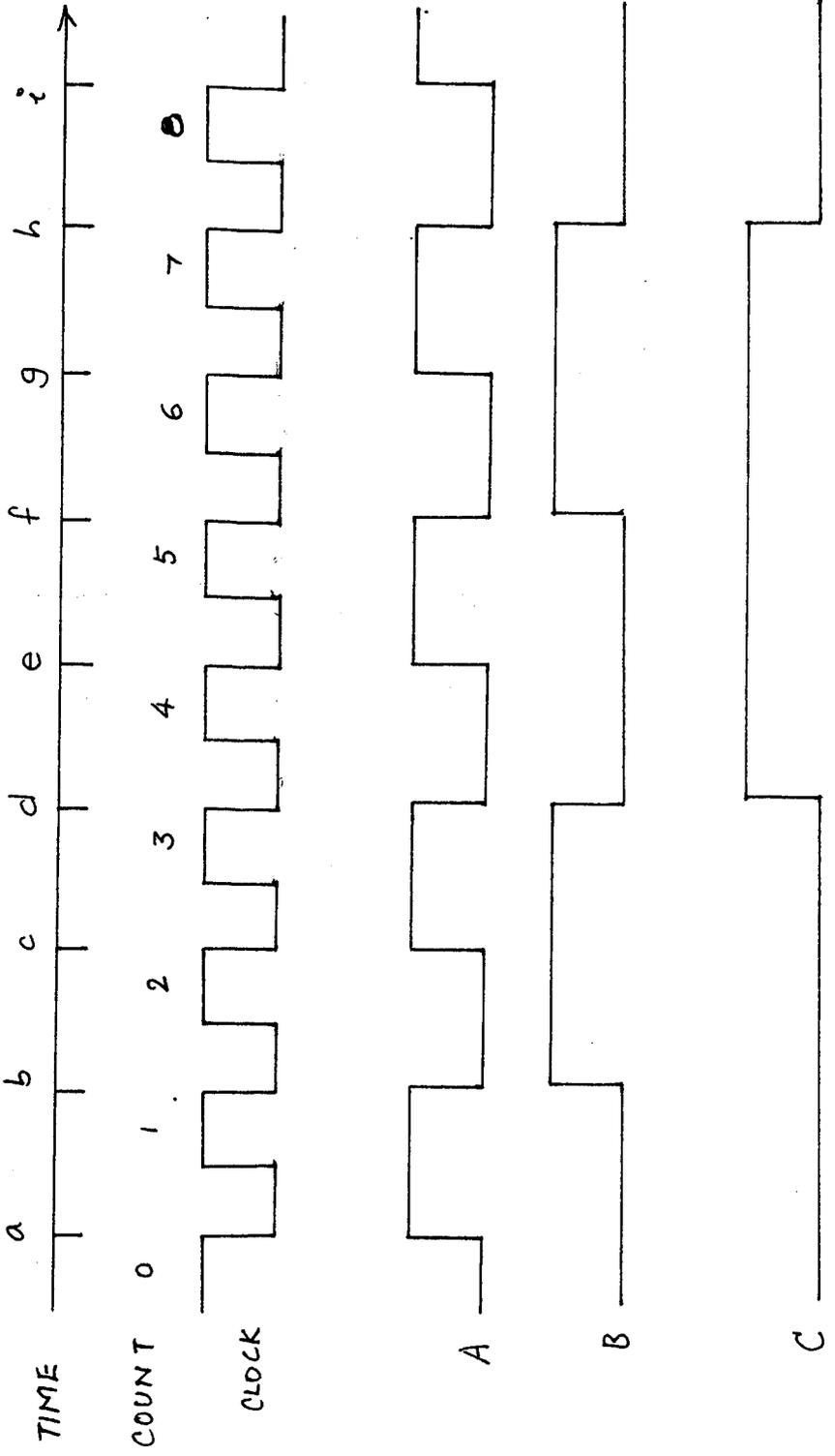


Fig 2.8(i) Mod - 8 Parallel binary counter



C	B	A	COUNT
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7
0	0	0	0

Fig 2.8(ii) Truth table of synchronous counter

Fig 2.8(iii) Wave forms at outputs of synchronous counter

## CHAPTER 3

### CIRCUIT HARDWARE OPERATION DETAILS

The signals on the expansion slots of an IBM PC or compatible machine form the starting point for the design of an extension card for this type of computer. The hardware arranges the data flow, and addressing must be laid down carefully to meet the technical requirements for the hardware to work. In practice the eight bidirectional datalines between the extension card and the motherboard must be buffered to prevent the drive capacity of the CPU and associated peripheral chips being exceeded at the risk of permanent damage to the computer. The next requirement is that the extension circuit must occupy a carefully selected address range that can be accessed by the CPU in accordance with the I/O (input/output) map defined by IBM. Hence, the address range occupied by the extension card must be unique in the computer system to avoid I/O contention problems.

The frequency meter card presented here has a type 8255 PPI (Programmable Peripheral Interface) to meet the above requirements regarding buffering and decoding. All control of counter and clock circuits on the extension card is arranged by the CPU via the three data registers and one control register contained in the 8255 PPI.

The circuit diagram of the meter card is shown in Fig. 3.0. This frequency meter card is capable of measuring frequencies of both TTL

compatible and TTL non compatible waves. The TTL compatible wave can be directly fed to the NOR gate N13 as shown in the circuit diagram. The TTL non compatible wave is fed through the socket K1. The chip U6060B converts the TTL non compatible wave to TTL compatible wave. The switch S1 can then be utilised to input either of the two waves.

Frequency measurement invariably requires a clock oscillator. The basic principle and operations of a clock oscillator has been already described. A 4MHz oscillator block controlled by an on board quartz crystal is need for our frequency meter card.

The clock pulses are fed through 3 NOR gates which change the levels of the pulses. i.e. High level 1 becomes 0 and vice versa. The circuit operates during the low level (0).

An electronic switch IC4066 inputs the pulses to 74HCT393. The input is to pin no.1. As explained, 74HCT393 is a divider circuit chip. The output is obtained at pin no. 8. Between pin no. 1 and 8 there are 8 flip flops and hence the input frequency at pin no.1 is divided and the frequency obtained at the output will be  $\frac{4 \times 10^6}{2^8} = 15625$  Hz.

The divider 74HCT393 is cascaded with IC4040 which is also a divider circuit. The input is at pin no.10, the  $\overline{\text{CLK}}$ . The output

is obtained at pin no. 1 of IC4040. The frequency on the way to pin no.1 gets divided by a factor  $2^{12}$  and the frequency now obtained at the output of IC10 pin 1 will be  $\frac{15625}{2^{12}} = 3.81$  Hz.

The IC10 is cascaded with another IC4040 chip (IC9) and output is obtained at pin 9 which divides the frequency obtained at its input by a factor  $2^1$ . Hence the frequency at the output of IC9 is  $\frac{3.81}{2} = 1.905$  Hz.

Hence the frequency which was initially at 4MHz has been divided and finally frequency of 1.905 Hz has been reached. The time period has risen from  $2.5 \times 10^{-7}$  to 0.52493 sec. The time period at pin 2 of N13 is 0.524 sec. The gate time i.e. the time during which the signal remains low is 0.262 sec.

The low level signal which remains for 0.262 sec. at pin 2 of N13 is inverted by N15 and the high level obtained at the O/P of N15 keeps the LED glowing and the electronic switch ES1 in ON position.

The other input signal to the NOR gate is the TTL compatible signal whose frequency has to be measured.

The NOR gate gives a high level during the periods when the input signal remains low. Hence for a period of 0.262 sec. the number of times

the NOR gate N13 gives high level output is dependent on the frequency of the input wave. The pulses at the output of N13 are fed to the counter circuits.

After 0.262 sec is over the signal at the input pin 2 of N13 is high. The inverter N15 gives a low output and ES1 and LED are disabled. i.e. the switch ES1 is opened and the LED stops glowing.

The output pulses at N13 pin 1 is connected to pin 1 of 74HCT393 (IC14). Now the chip 74HCT393 acts as a counter. The output of the counter is obtained from each of the 8 flip flops present in it. Hence the first 8 bit information regarding the number of pulses counted are obtained. These bits of information are fed to port B of PPI.

IC14 is cascaded to two other counter chips IC12 and IC11. IC12 measures 12 bits out of which 7 bits go to port C. As the computer would read these bits separately from the byte of information obtained at port B therefore the pulses counted at C are multiplied by  $2^8 = 256$ .

IC12 counts 5 bits for port A while IC11 the other 3 bits. The number of pulses actually counted should be multiplied by  $2^{8+7} = 2^{15} = 32768$

These multiplication factors are taken into account by the software.

Also the PPI which gets the 23 bits of information regarding the number of pulses counted is controlled by the software.

Chapter 6 describes the software which controls the PPI. The chapter also describes how the number of pulses counted can be used to calculate frequency.

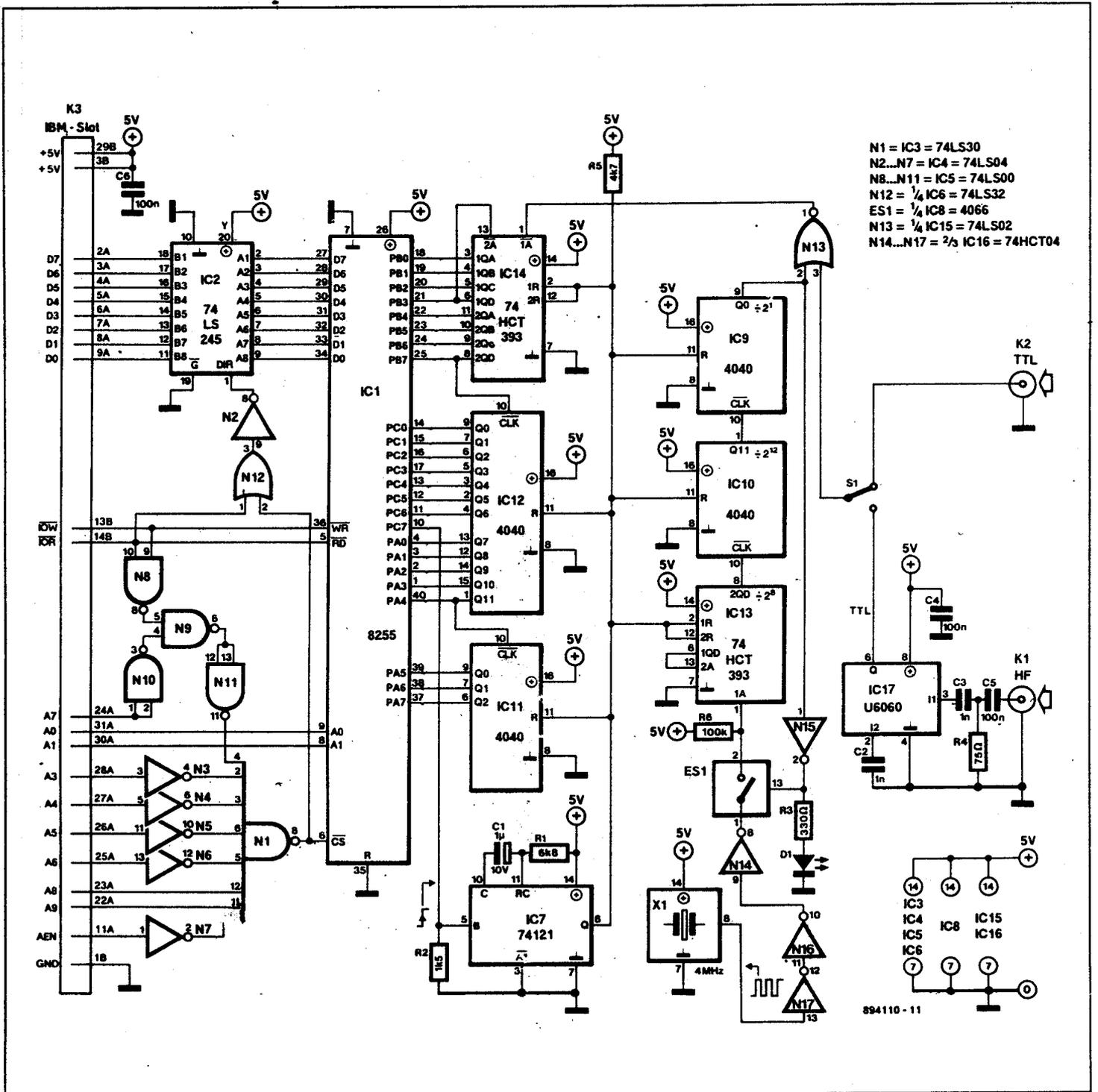


Fig 3.0 Circuit diagram of the Frequency meter card for PCs.

## CHAPTER 4

### PCB AND COMPONENTS DETAILS

The components used in the Frequency meter card are all TTL compatible. A brief description of the components employed in the circuit is given in this chapter. Let us see the details regarding the printed circuit board(PCB) before we move on to the components description.

#### 4.1. Printed circuit Board(P.C.B.)

A print-out of the PCB designed using a software SMART WORK is as shown in Fig 4.1 . The PCB is a double-sided board. The holes are plated through. The male PC slots which are connected to the female expansion slots in the PC mother board are gold plated.

#### 4.2 8255 A Programmable Peripheral Interface:

In this project we are using the programmable peripheral interfacing device 8255 A. It is a programmable, parallel I/O device. It can be programmed to transfer data under various conditions, i.e. from simple I/O to interrupt I/O. It has 24 I/O pins that can be grouped primarily in to 8 bit parallel port, A and B, with the remaining 8 bits as port C. The eight bits of port C can be used as individual bits or grouped in to two 4 bit ports, C upper(Cu) and C lower (Cl). The function of these ports are defined by writing a control word in the control register. The fig.4.2(i),(ii) shows the block diagram and pin details of 8255 PPI.

**Functions:**

The function of 8255 A is classified according to two modes. The Bit set/Reset mode (BSR) and I/O mode.

The Bit set/Reset 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 port function as simple I/O ports, mode 1 is a handshake mode where by Port A and /or B use bit from port C as handshake signals. In handshake mode, two type 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 connection are as given.

RD (Read) - This control signal enables the Read operation. when the signal is low, the MPU reads data from a selected I/O Port of the 8255 A

WR (write) - This control signal enables the write operation. When the signal goes low, the MPU writes in to a selected I/O port or control register.

RESET (Reset) - This is an active high signal. It clears the control register and set all ports in the input mode.

CS  $A_0$  and  $A_1$  : These are device select signals. CS is connected to a decoded address, and  $A_0$  and  $A_1$  are generally connected to MPU address lines  $A_0$  and  $A_1$ , respectively.

The CS signal is the master chip select and  $A_0$  and  $A_1$  specify one of  $A_2$  I/O ports or the control register as given below

CS	$A_1$	$A_0$	selected
0	0	0	Port A
0	0	1	Port B
0	1	0	Port C
0	1	1	Control register
1	x	x	8255 A is not selected

### Control Word:

Figure 4.3 (i),&(ii) shows a register called the control register. The contents of this register, called the control word, specify an I/O functions for each port. This register can accessed to write a control word when  $A_0$  and  $A_1$  are at logic 1, as mentioned previously. The register is not accessible for a read operation.

Bit  $D_7$  of the control register specifies either the I/O function or the Bit set/reset funtion as classified in fig 4.3(i),&(ii). If bit  $D_7=0$ , Port C operates in the Bit set/Reset (BSR) mode. The BSR control word does not affect the functions of Port A and B.

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

1. Determine the addresses of port A, B and C and of the control register according to the chip select logic and address lines  $A_0$  and  $A_1$
2. Write a control word in the control register.
3. Write I/O instructions to communicate with peripherals through port A, B, and C.

Various modes are given below.

#### **Mode 0 - Simple input or output:**

In this mode, Port A and B can be viewed as equivalent to two 8212, and port C as equivalent to two 4 bit 82125. Each port can be programmed to function as simply as input port or on output port. The input/output features in mode 0 are as follows.

1. Outputs are latched
2. Inputs are not latched
3. Ports do not have handshake or intempt capability

In BSR mode is concerned only with the eight bit of port C, which can be set or reset by writing an appropriate control word in the control register. A control word with bit  $D_7 = 0$  is recognised a BSR control word, and it does not alter any previously transmitted control word with bit  $D_7 = 1$ , thus the I/O operations of port A and B are not affected by BSR control word. In the BSR mode individual bits of port C can be used for applications such an on/off switch.

**Mode 0 : control word.**

The configuration of the ports is specified as follows

Port A: As an input port

Port  $C_L$ : As an output port because bit  $PC_0$  is used to start conversion.

Port  $C_U$ : As an input port to read the status at  $PC_7$

Port B : Not used

**Mode 1 Input or Output with handshake:**

In mode 1, handshake signals are exchanged between the MPU and peripherals prior to data transfer. The features of this mode include

1. Two ports (A and B) function as 8 bit I/O ports. They can be configured either as input or output ports.
2. Each port uses three lines from port C as handshake signals. The remaining two lines of port C can be used for simple I/O function.
3. Input and output data are latched
4. Interrupt logic is supported.

In the 8255 A, the specific lines from port C used for handshake signals vary according to the I/O function of a port. Therefore, input and output functions in mode 1 are discussed separately.

### **Mode 1 Input control signals:**

The control signals used for handshaking when port A and B configured as input ports. Port A uses the upper three signals;  $PC_3$ ,  $PC_4$ , and  $PC_5$ . Port B uses the lower three signals:  $PC_2$ ,  $PC_1$  and  $PC_0$ . The function of three signals are as follow.

### **STB (stroke Input):**

This signal is generated by a peripheral device to indicate that it has transmitted a byte of data.

### **IBF (Input Buffer Full):**

This signal is an acknowledgement by the 8255 A to indicate that the input latch has received the data byte. This is reset when the MPU reads the data.

### **INTR (Interrupt Request):**

This is an output signal that may be used to interrupt the MPU. This signal is generated if STB, IBF and INTE (Internal flip-flop) are all at logic 1. This is reset by the falling edge of the RD signal.

**INTE (Interrupt Enable)**

This is an internal flip flop used to enable or disable the generation of the INTR signal.

**Mode 1 : Out Put Control Words.**

The signals used when port A and B are configured as out put ports.

**OBF (Out put Buffer Full):**

This is an output signals that goes low when the MPU writes data into the output latch of the 8255A. This indicates to an output peripheral that new data are ready to be read. It goes high again after the 8255 A receives an ACK from the peripheral.

**ACK (Acknowledge):**

This is an input signal from a peripheral that must output a low when the peripheral receives the data from the 8255A port.

**INTR (Interrupt Request):**

This is an out put signal and it is set by the rising edge of the ACK signal. This signal can be used to interrupt the MPU to request the next data byte for output. The INTR is set when OBF, ACK and INTE are all one and reset by the falling edge of WR.

## **Mode 2 : Bidirectional Data Transfer:**

This mode is used primarily in applications such as data transfer between two computers or floppy disk controller interface. In this mode, port A can be configured as the bidirectional port and port B either in mode 0 or Mode 1. Port A uses five signals from port C as control signals for data transfer. The remaining three signals from port C can be used either as simple I/O or as handshake for port B.

### 4.3 IC 74121 Monostable with Input Logic

The basic monostable circuit discussed in the previous section provides an output pulse of predetermined width in response to an input trigger. Logic gates have been added to the inputs of a number of commercially available monostable circuits to facilitate the use of these circuits as general purpose delay elements. The 74121 non triggerable is such a circuit.

The logic inputs to these circuits can be used to allow triggering of the device on either a high-to-low transition or on a low-to-high transition. Whenever the value of the input logic equation changes from false to true, the circuit will trigger. Takes care to not that a transition from false to true must occur, and simply holding the input logic equation in the true state will have no effect.

The logic diagram, truth table, and typical waveform for a 74121 are given in Fig. 4.4(i). The inputs to the 74121 are  $A_1$ ,  $A_2$  and B. The trigger input to the monostable appears at the output of the AND gate. Here's how the gates work:-

1. If B is held high, a negative transition at either  $A_1$  or  $A_2$  will trigger the circuit (see fig. 4.4(ii)). This corresponds to the bottom two entries in the truth table.

2. If either  $A_1$  or  $A_2$  or both are held low, a positive transition at B will trigger the circuit (see fig. 4.4(iii)).
3. This corresponds to the top two entries in the truth table. A logic equation for the trigger input can be written as

$$T = (A_1 + A_2) \overline{BQ}$$

Note that for T to be true (high) either  $\overline{A_1}$  or  $\overline{A_2}$  must be true - that is, either  $\overline{A_1}$  or  $\overline{A_2}$  at the gate input must be low. Also since  $\overline{Q}$  is low during the timing cycle (in the quasistable state) it is not possible for a transition to occur at T during this time. The logic equation for T must be low if  $\overline{Q}$  is low. In other words, once the monostable has been triggered into its quasistable state, it must time out and switch back to its stable state before it can be triggered again. This circuit is thus nontriggerable.

The output pulse width at Q is set according to the values of the timing resistor R and capacitor C as  $t=0.69 RC$ . For instance, if  $C = 1 \mu\text{F}$  and  $R = 10 \text{ K}\Omega$ , the output pulse width will be  $t = 0.69 \times 10^4 \times 10^{-6} = 6.9 \text{ ms}$ .

Consider a 74121 monostable in fig. 4.4(i) with  $R = 1 \text{ k}\Omega$  and  $C = 10,000 \text{ pF}$ . Pins 3 and 4 are tied to GND and a series of positive pulses are applied to pin 5. The expected waveform at pin 6 is as shown in fig. 4.4 (iv) assuming the input pulses are spaced by  $10 \mu\text{s}$ , the circuit is connected such that positive pulses applied to pin 5 will trigger it. The output pulse width will be  $t = 0.69 \times 10^3 \times 10^{-8} = 6.9 \mu\text{s}$ . The monostable will trigger and time out for every input pulse appearing at B, as shown in fig 4.4.(iv).

The same condition applied in the input pulses spaced by  $5 \mu\text{s}$  causes the monostable to become non-triggerable, it will trigger once and time out for every other input pulse shown in fig. 4.4 (v).

4.4

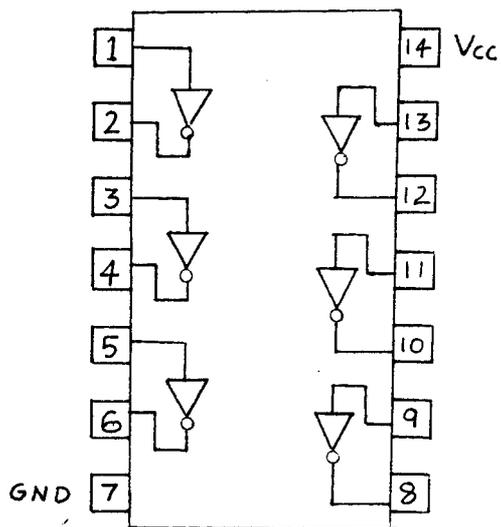
74LS04

Hex inverters

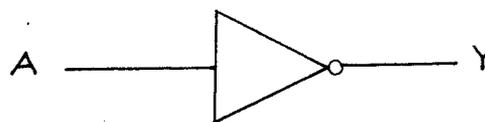
Description

These devices contain six independent inverters.

74LS04 is characterized for operation from 0°C to 70°C



Logic Diagram(each inverter)



Function Table(each inverter)

INPUTS	OUTPUT
A	Y
H	L
L	H

Positive Logic

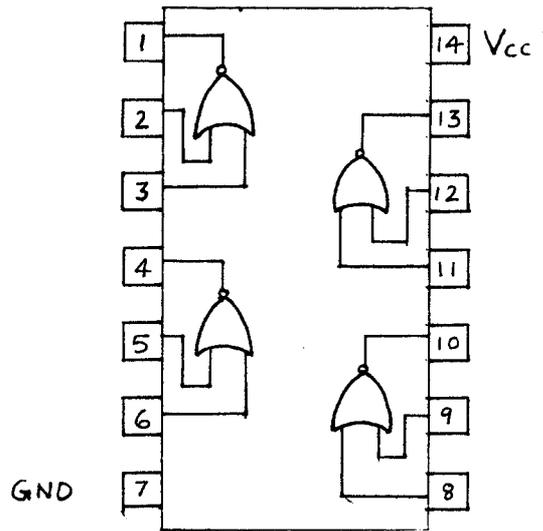
$$Y = \bar{A}$$

**Quadruple 2 Input Positive NOR Gates**

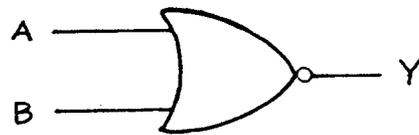
**Description**

These devices contain four independent 2-input-NOR gates.

74LS02 is characterized for operation from 0°C to 70°C.



**Logic diagram (each gate)**



**Function Table (each gate)**

INPUTS		OUTPUT Y
A	B	
H	X	L
X	H	L
L	L	H

**Positive Logic**

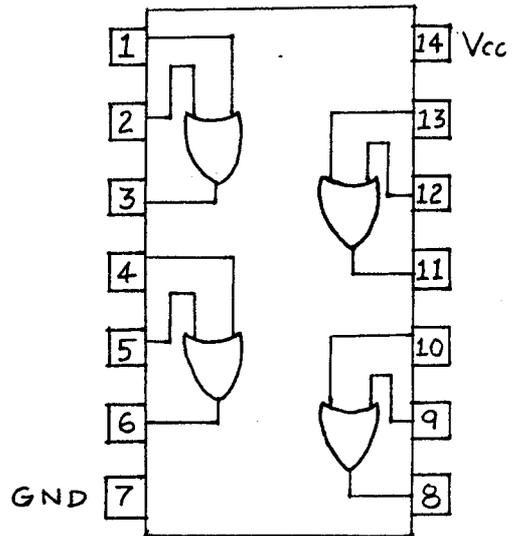
$$Y = \overline{A \cdot B} \quad (\text{or}) \quad Y = \overline{A + B}$$

**Quadruple 2 Input Positive OR Gates**

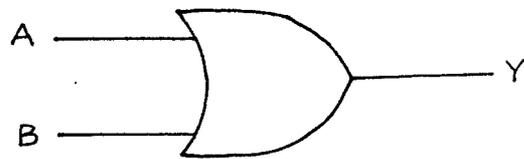
**Description**

These devices contain four independent 2 input OR gates.

74LS32 is characterized for operation from 0°C to 70°C.



**Logic Diagram(each gate)**



**Function Table(each gate)**

INPUTS		OUTPUT
A	B	Y
H	X	H
X	H	H
L	L	L

**Positive Logic**

$$Y = A + B \text{ (or) } \overline{\overline{A} \cdot \overline{B}}$$

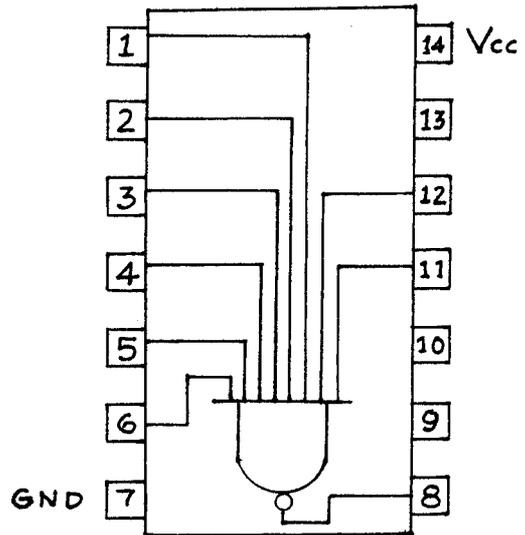
4.7

8 Input Positive NAND Gates

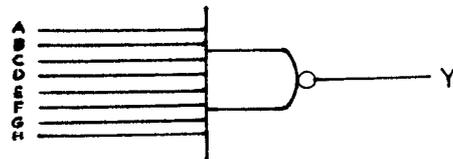
Description

These devices contain a single 8 Input NAND Gate.

74LS30 is characterized for operation from 0°C to 70°C.



Logic Diagram



Function Table

INPUTS A THRU H	OUTPUT Y
ALL INPUTS H	L
ONE OR MORE INPUTS L	H

Positive Logic

$$Y = \overline{A \cdot B \cdot C \cdot D \cdot E \cdot F \cdot G \cdot H}$$

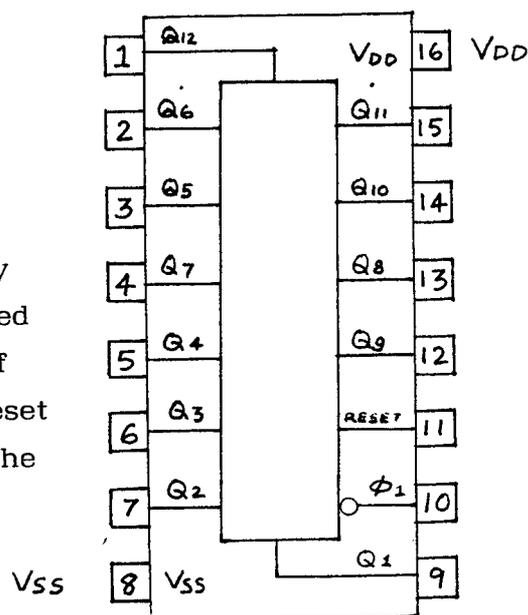
(or)

$$Y = \overline{A} + \overline{B} + \overline{C} + \overline{D} + \overline{E} + \overline{F} + \overline{G} + \overline{H}$$

## 12 Stage Ripple Carry Binary Counters

## Description

4040 is a 12 stage ripple carry binary counter. The counters are advanced one count on the negative transition of each clock pulse. The counters are reset to the zero state by a logical "1" at the reset input independent of clock.



## Features

- \* Wide supply voltage range 1.0 V to 15 V
- \* High noise immunity  $0.45 V_{DD}$  (typ.)
- \* Low power TTL compatibility fan out of 2 driving 74L or 1 driving 74LS
- \* Medium speed operation 8 MHz typ. at  $V_{DD} = 10 V$
- \* Schmitt trigger clock input

4.9

SN74LS245

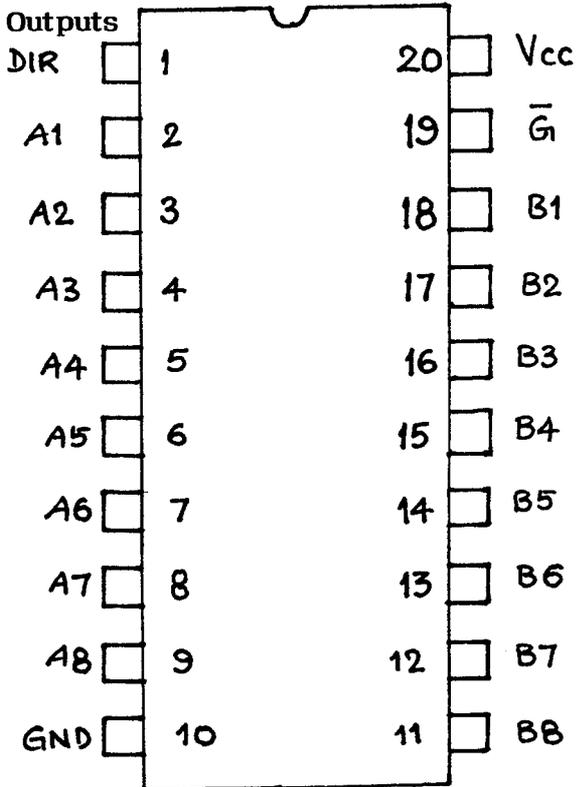
Octal Bus Transceivers with 3 State Outputs

Description

These octal bus transceivers are designed for asynchronous two way communication between data buses. The control function implementation minimizes external timing requirements.

The devices allow data transmission from the A bus to the B bus or from the B bus to the A bus depending upon the logic level at the direction control (DIR) input. The enable input (G) can be used to disable the device so that the buses are effectively isolated.

The 74LS245 is characterized for operation from 0°C to 70°C.

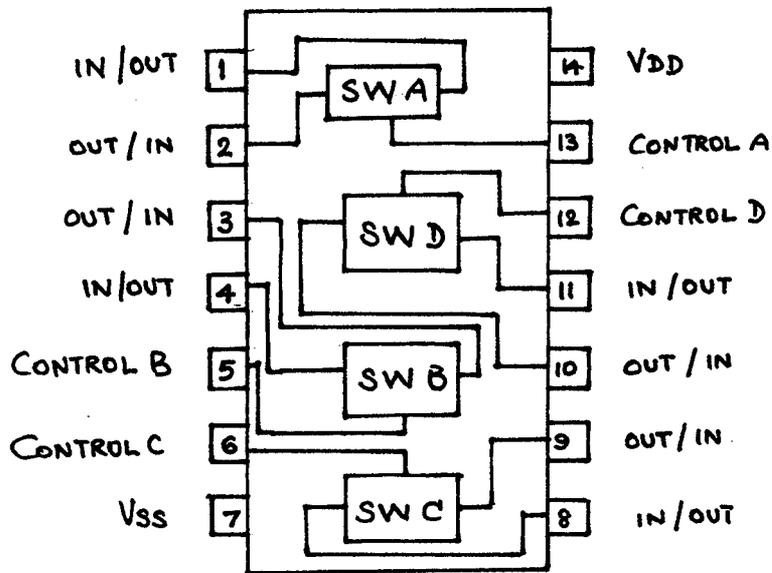
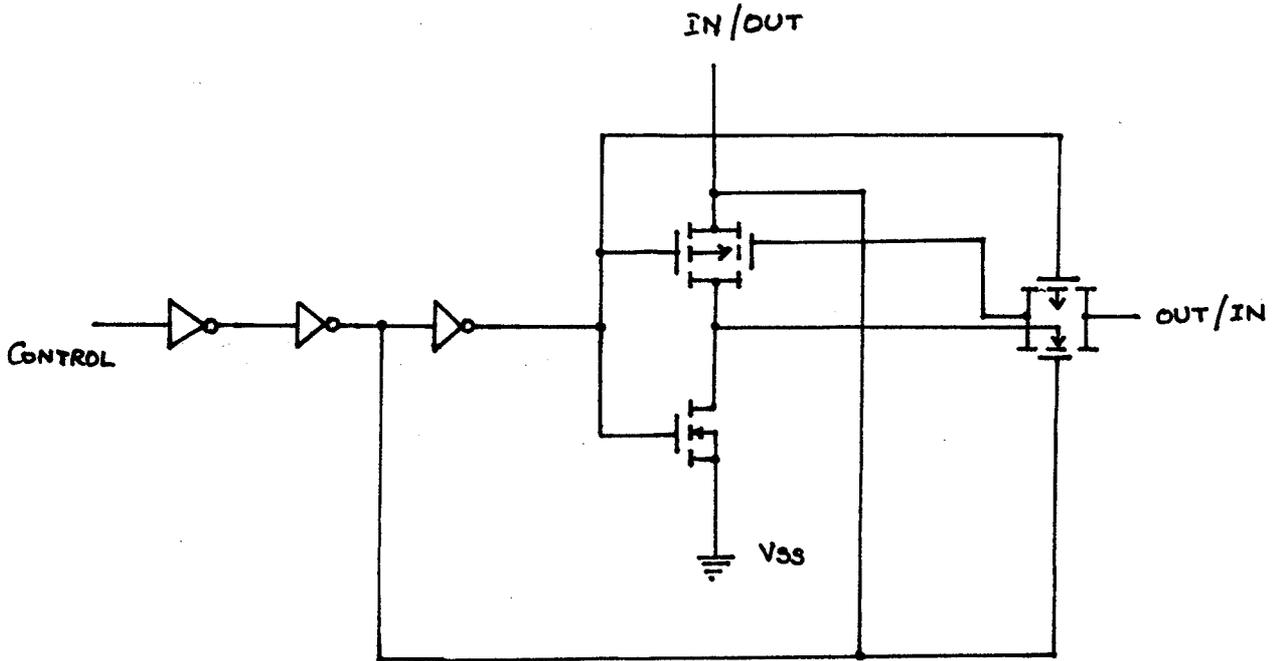


FUNCTIONAL TABLE

ENABLE $\bar{G}$	DIRECTION CONTROL DIR	OPERATION
L	L	B data to A bus
L	H	A data to B bus
H	X	Isolation

- H = HIGH LEVEL
- L = LOW LEVEL
- X = IRRELEVANT

4.10 SCHEMATIC AND CONNECTION DIAGRAMS OF CD4066



## IC 4066

### Quad Bilateral Switch

#### Description

The 4066 is a quad bilateral switch intended for the transmission or multiplexing of analog or digital signals. It is pin for pin compatible with CD4016BC, but has a much lower "ON" resistance, and "ON" resistance is relatively constant over the input signal range.

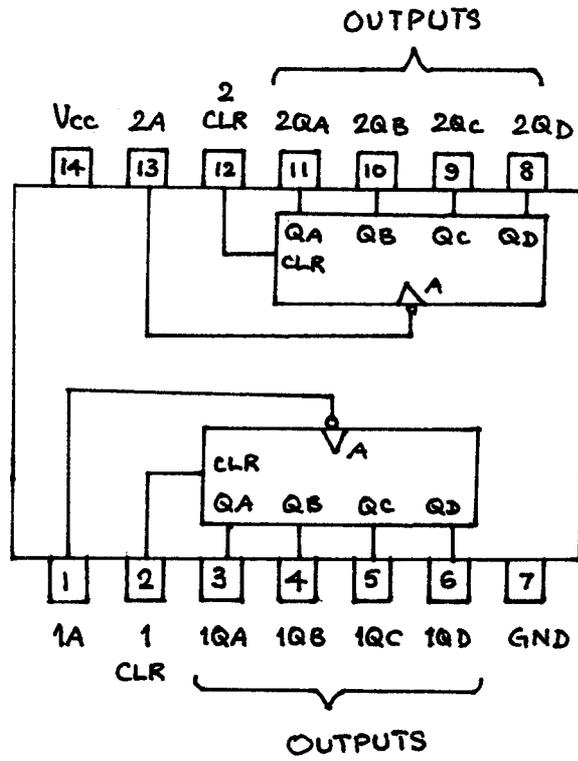
#### Features

- \* Wide supply voltage range            3V to 15V
- \* High noise immunity                0.45  $V_{DD}$  (typ.)
- \* Wide range of digital and analog  
switching                                7.5  $V_{PEAK}$
- \* "ON" resistance for 15V  
operation                                80
- \* Matched "ON" resistance over  
15V signal input                         $R_{ON} = 5$  (typ.)
- \* "ON" resistance flat over peak to peak signal range
- \* High "ON"/"OFF"                      65 dB (typ.)  
Output voltage ratio                     $f_{is} = 10$  KHZ,  $R_L = 10$  K
- \* High degree linearity                0.1 % distortion (typ.)

- High degree linearity @  $f_{is} = 1 \text{ kHz}$ ,  $V_{is} = 5V_{pp}$
- High degree linearity  $V_{DD} - V_{SS} = 10V$ ,  $R_L = 10 \text{ K}$
- \* Extremely low "OFF" switch leakage @  $V_{DD} - V_{SS} = 10V$
- \* Extremely high control input impedance
- \* Low crosstalk between switches @  $f_{is} = 0.9 \text{ MHz}$
- \* Frequency response, switch "ON" 40 MHz

4.11

# MM74 HC393



## 74HC393

### Description

These counter circuits contain independent ripple carry counters and utilize micro CMOS Technology, 3.5 micron silicon gate P well CMOS. The 74HC393 contain two 4 bit ripple carry binary counters, which can be cascaded to create a single divide-by-256 counter.

Each of the two 4 bit counters is incremented on the high to low transition (negative edge) of the clock input, and each has an independent clear input. When clear is set high all four bits of each counter are set to a low level. This enables count truncation and allows the implementation of divide by N counter configurations.

Each of the counters outputs can drive 10 low power Schottky TTL equivalent loads. These counter are functionally as well as pin equivalent to the 74LS393. All inputs are protected from damage due to static discharge by diodes to  $V_{CC}$  and ground.

### Features

- \* Typical operating frequency: 50 MHz
- \* Typical propagation delay: 13 ns (CK to  $Q_A$ )
- \* Wide operating supply voltage range: 2 6V
- \* Low input current: 1 A
- \* Low quiescent supply current: 80 A maximum (74HC Series)
- \* Fanout of 10 LS TTL loads

checkplot v1.0 r4      21 Feb 1991    15:01:15  
file: sedem            upper layer  
approx. size: 6.60 by 5.35 in.    holes: 356

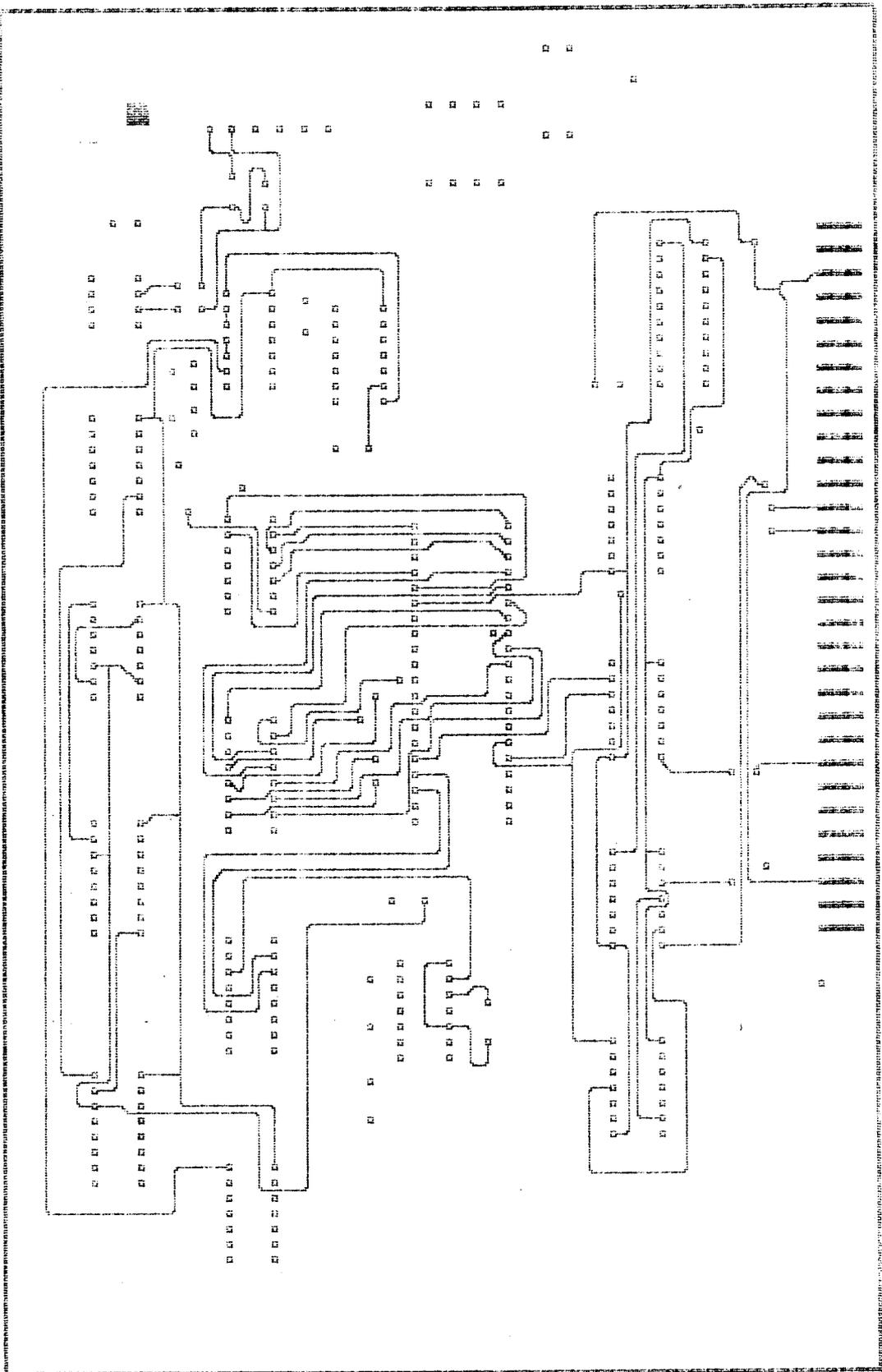


Fig 4.1 Printed Circuit Board Layout (Upper Layer)

checkplot v1.0 r4      21 Feb 1991    14:54:00  
file: sadam            lower layer  
approx. size: 8.60 by 5.35 in.    holes: 356

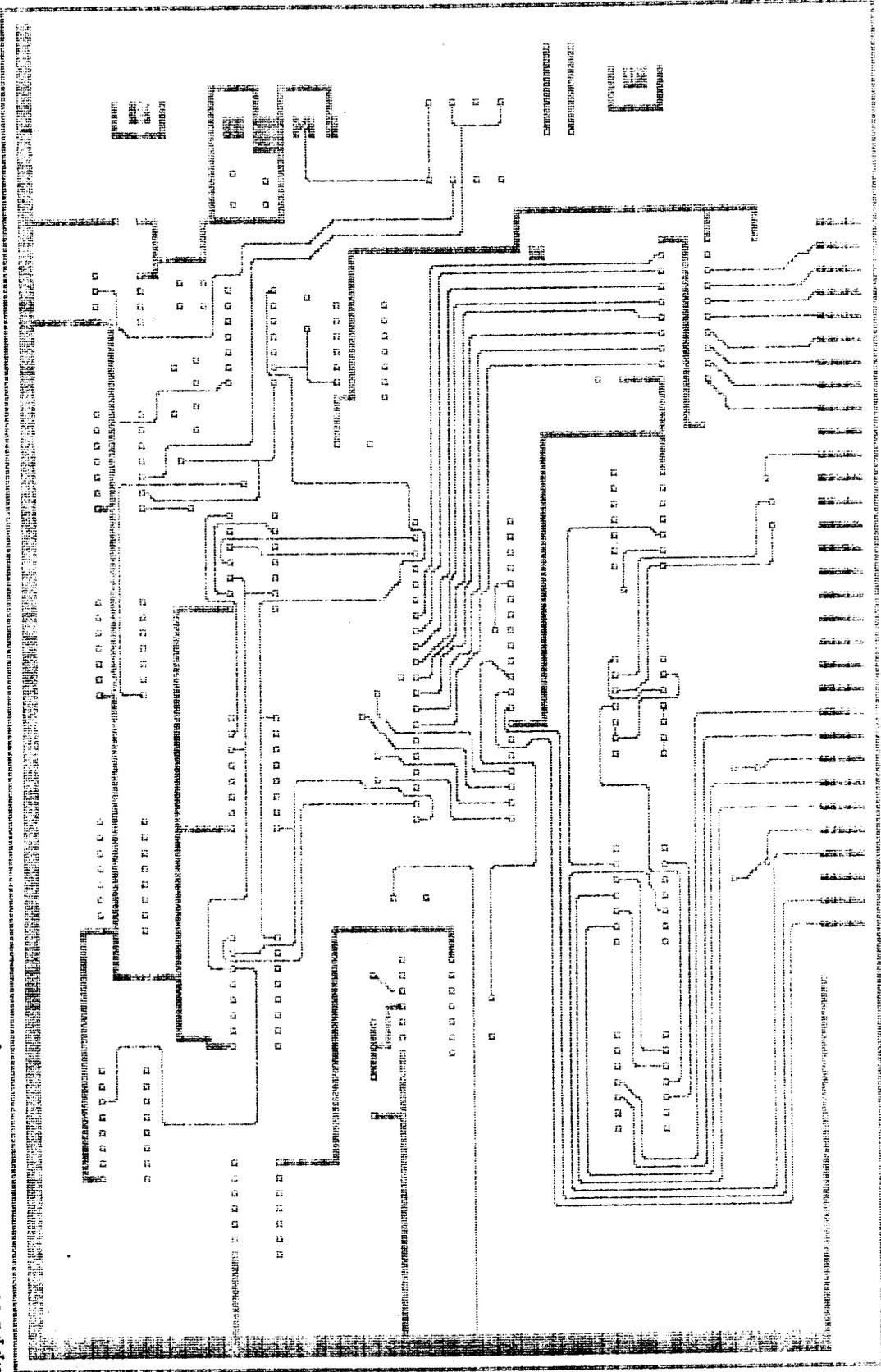
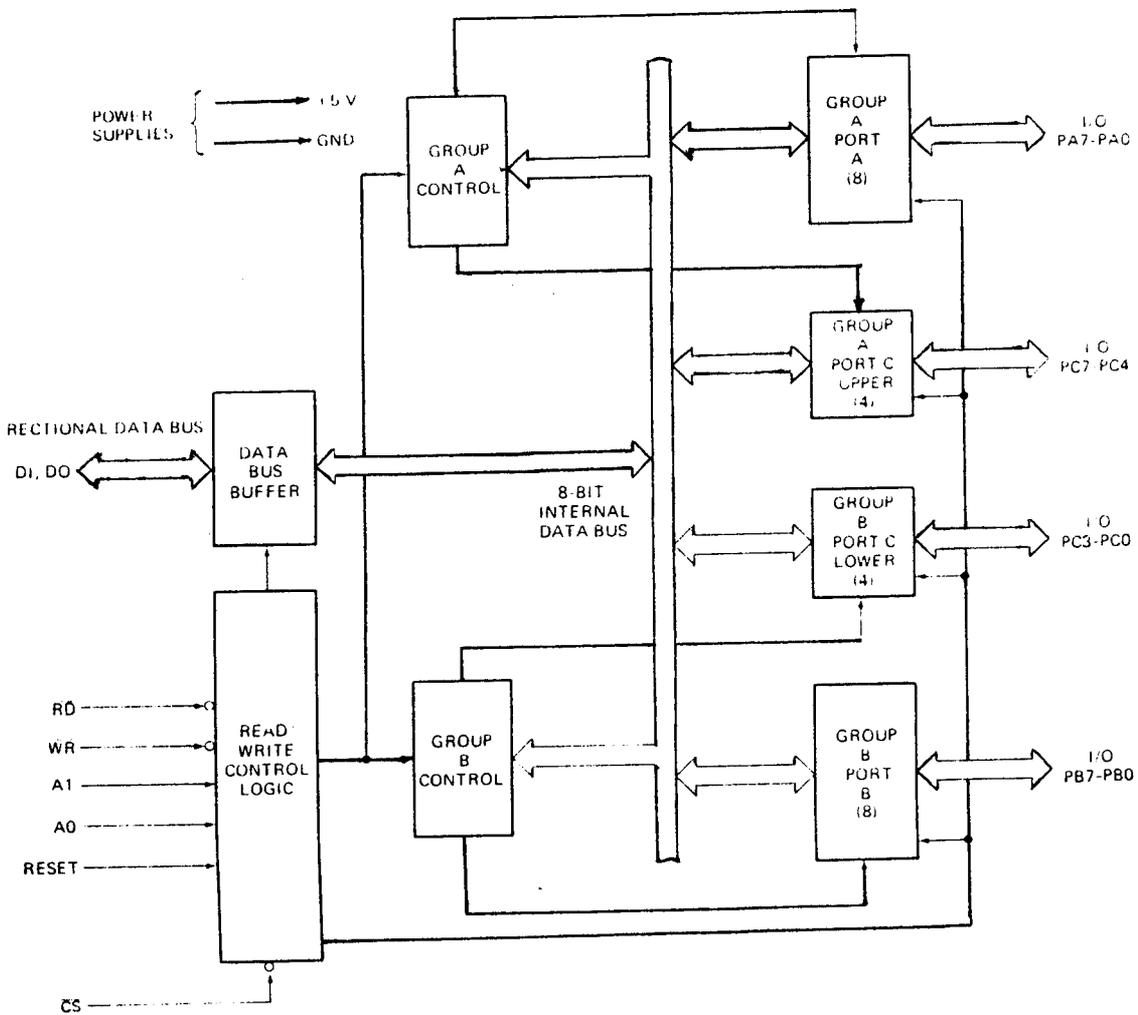
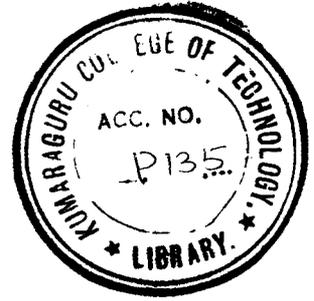


Fig 4.1 Printed Circuit Board Layout (Lower Layer)



**Fig 4.2 (i) Block diagram of 8255 PPI**

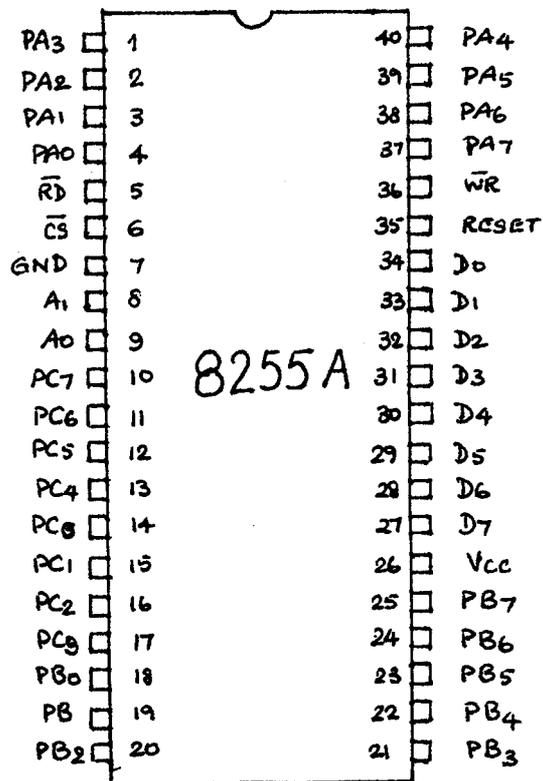
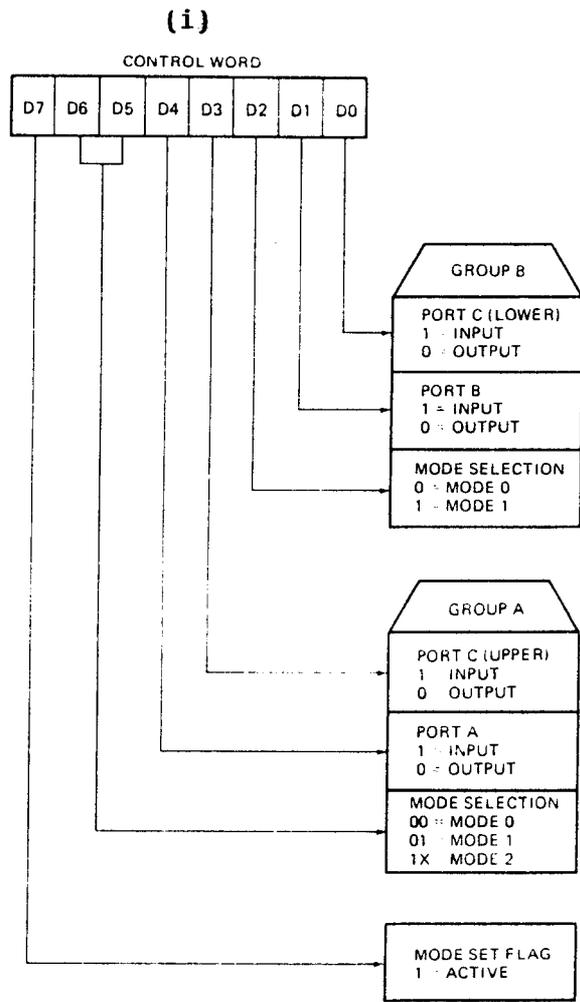
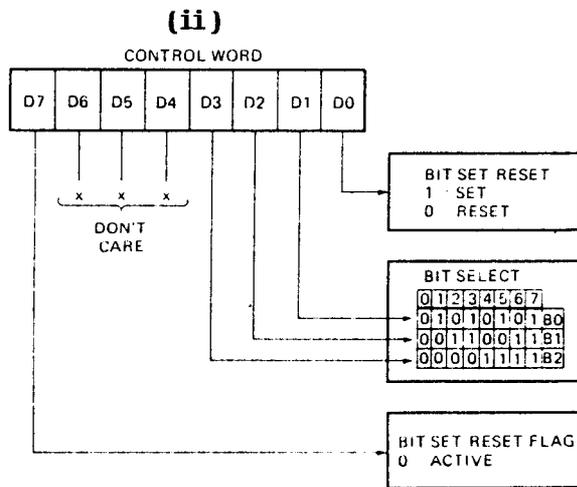


Fig 4.2(ii) Pin details of 8255 PPI



(a)



(b)

Fig 4.3 8255 control word formats

(i) Mode set control word.

(ii) Port C Bit set/Reset control word.

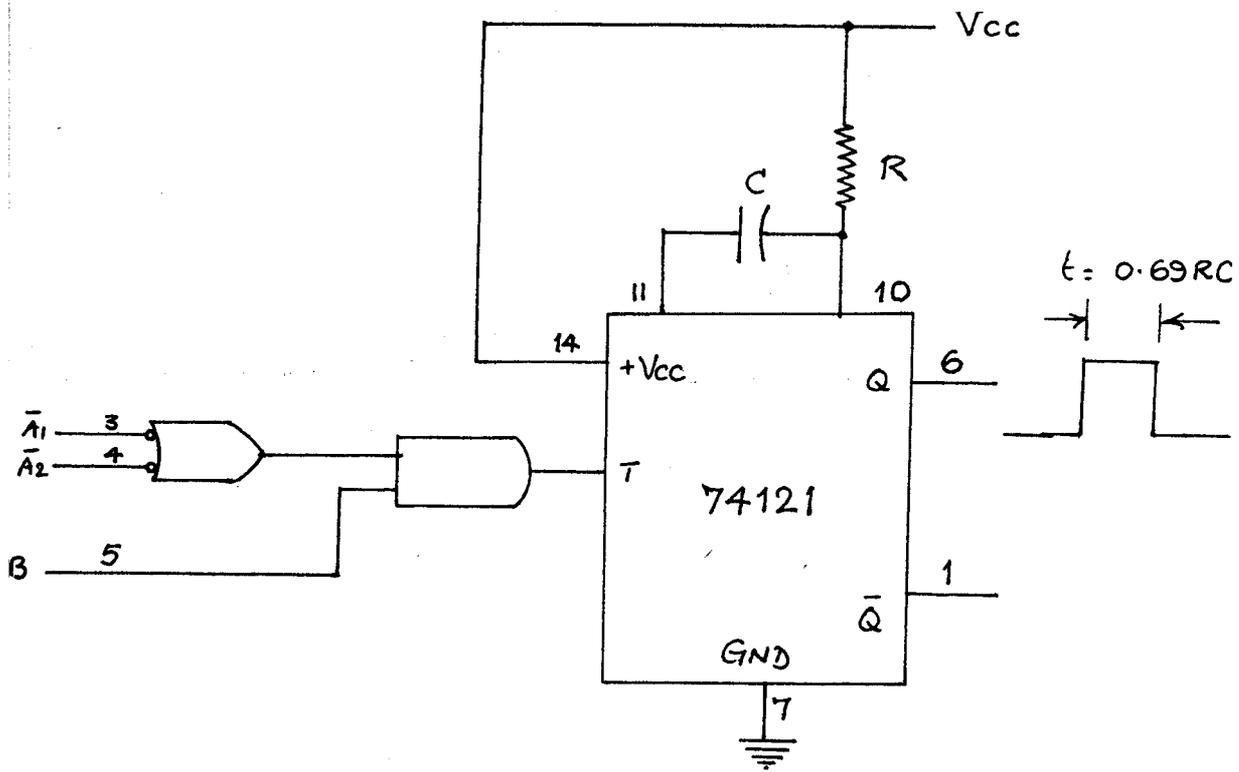


Fig 4.4(i) Logic Diagram of IC 74121

$\bar{A}_1$	$\bar{A}_2$	B	RESULT
L	X	↑	TRIGGER
X	L	↑	TRIGGER
↓	H	H	TRIGGER
H	↓	H	TRIGGER

L = LOW

H = HIGH

X = DONT CARE

↑ = LOW TO HIGH  
TRANSITION

↓ = HIGH TO LOW  
TRANSITION

Fig 4.4(i) Truth Table of IC 74121

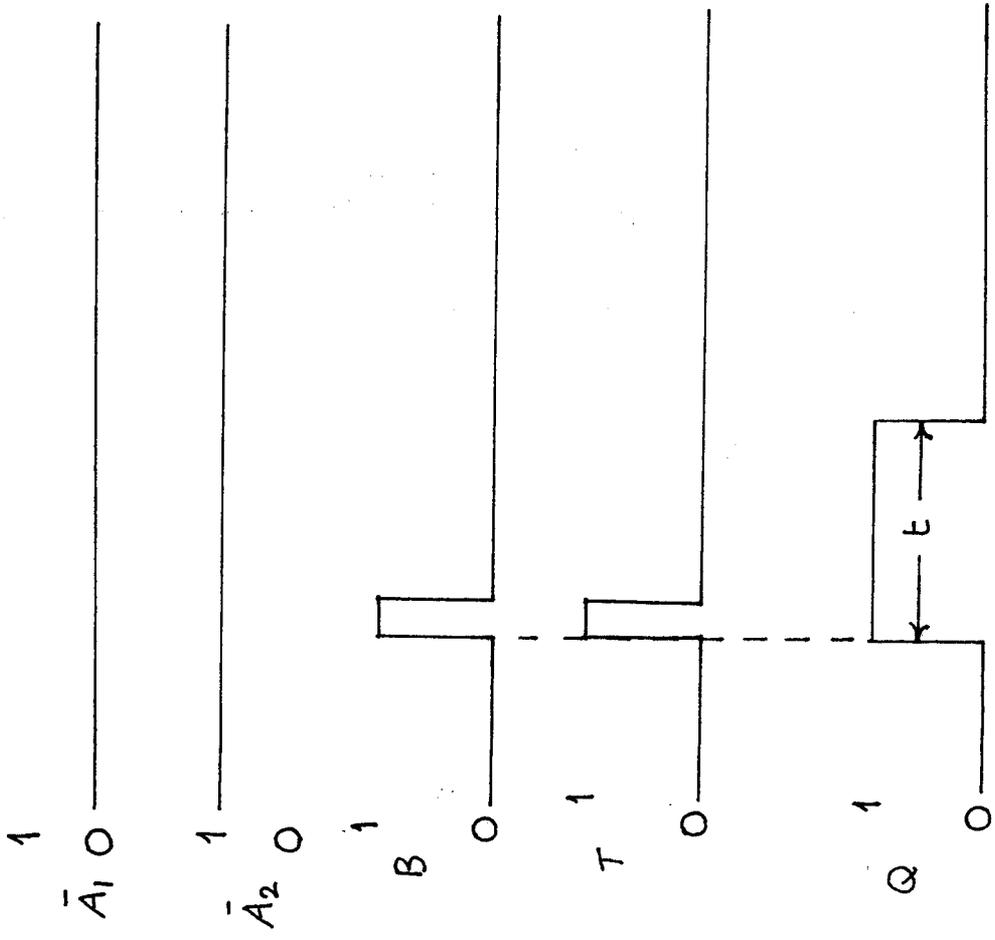


Fig 4.4(iii) Positive triggering

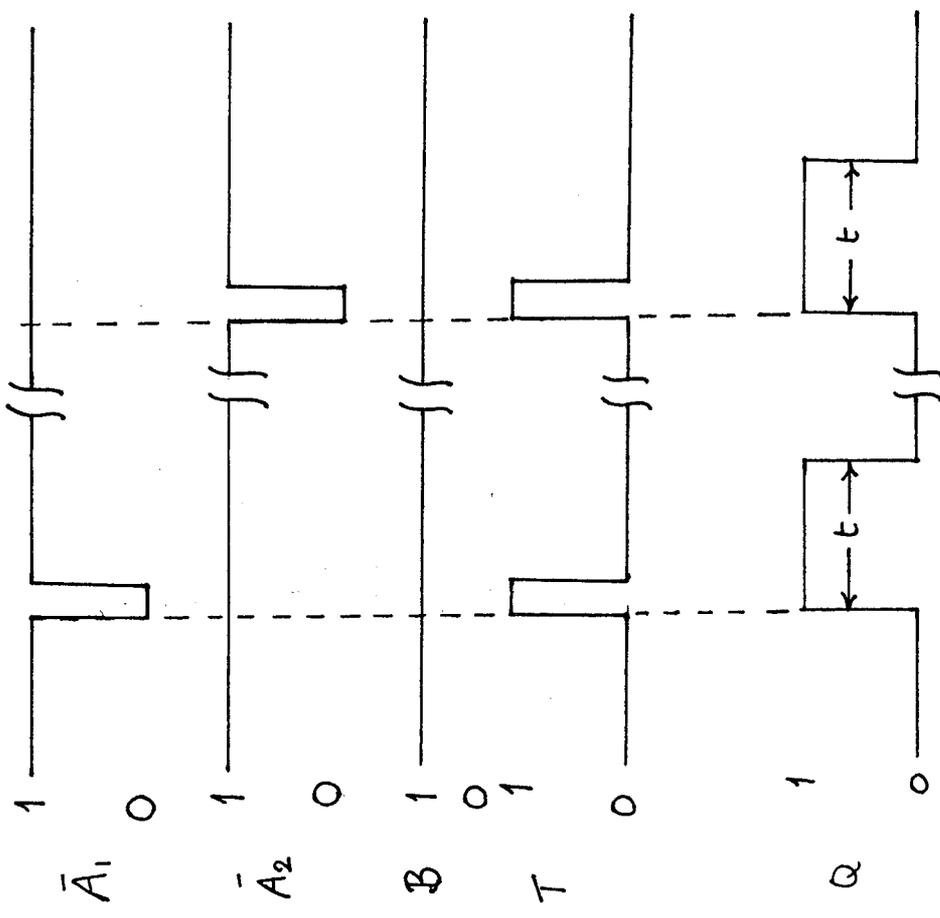


Fig 4.4(ii) Negative triggering

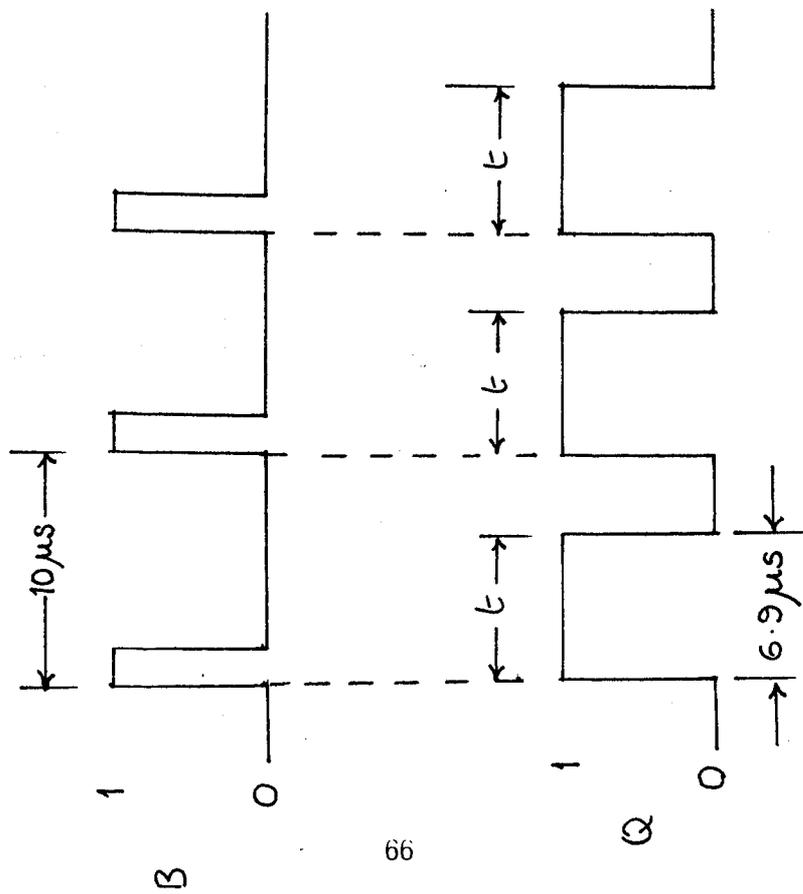


Fig 4.4(iv) Triggers on every pulse set B

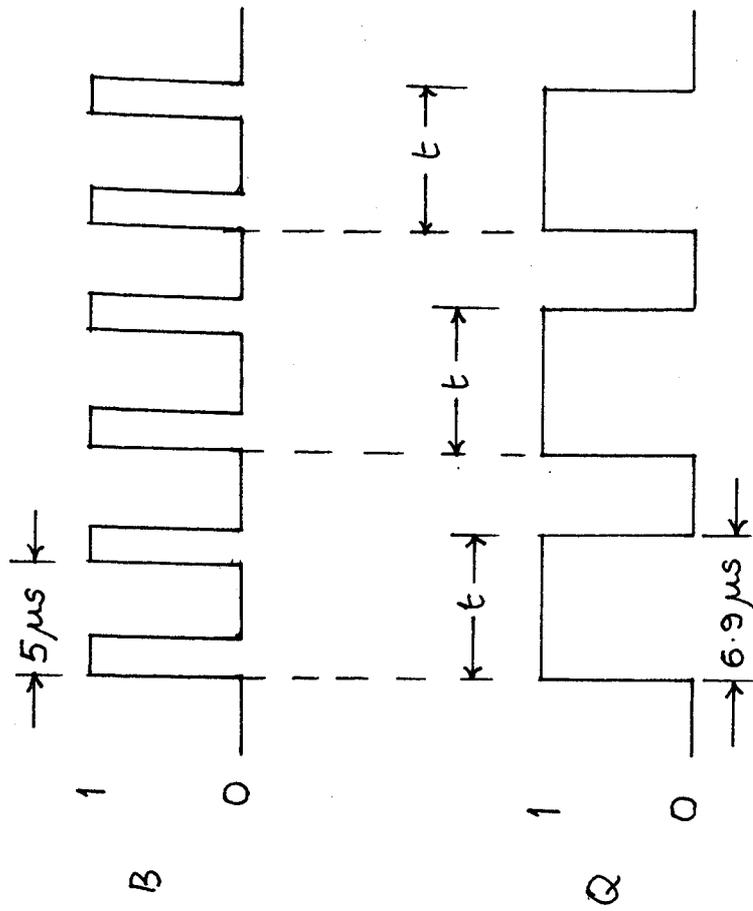


Fig 4.4(v) Trigger on every other Pulse set B

## CHAPTER 5

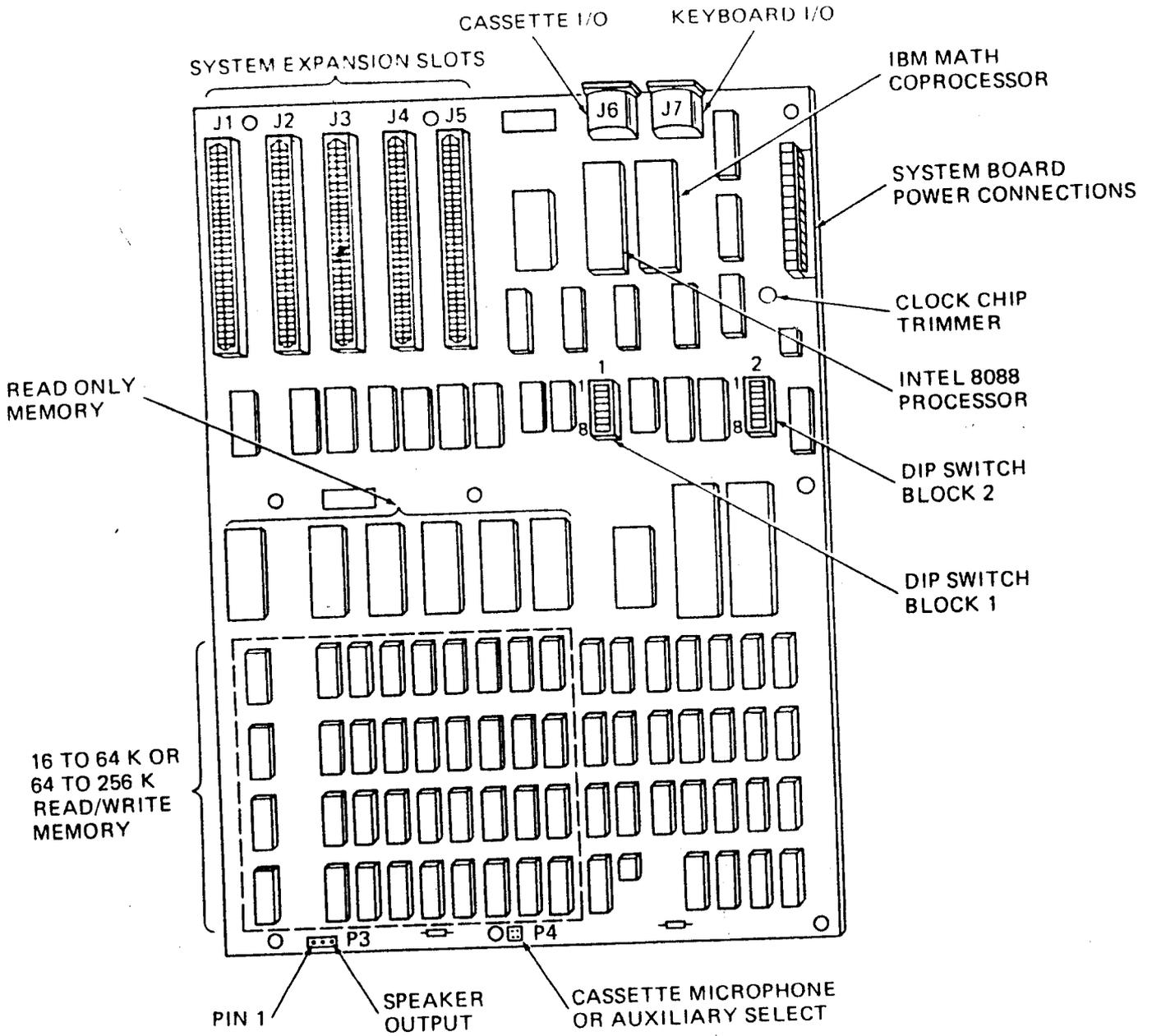
### EXPANSION SLOTS IN A PC:

The Frequency meter card is driven by software written in GW BASIC. This meter card should be connected to a IBM PC for its operation. The frequency meter card should have a direct access with the CPU of a IBMPC which then controls the PPI used in the meter card circuit by software and the required data from the PPI is obtained.

In the main microprocessor board of a IBMPC often called a motherboard there are expansion slots. These slots allow us to add the specific function we require it to do in addition to the basic CPU board. This 'open system' approach lets us easily customize the system for our application. Fig 5.1. shows the motherboard of a IBMPC. The system expansion slots are shown in the upper left hand corner of the fig.

Fig 5.2. shows the block diagram as to how the expansion slots are provided in an IBM PC motherboard. The main processor unit (8088Cpu) is connected to address buffers & data buffers. A bus controller chip 8288 is required as 8088 is operated in maximum mode. An interrupt controller (82594) is also provided. The buses from these connect to the 62-pin peripheral board connectors. The CPU can then use these buffers to communicate directly with the boards in the peripheral expansion slots.

The pin names and members for peripheral slots on IBM PC mother slots are shown in fig 5.3.



**Fig 5.1 Component lay out diagram for IBM PC mother board .**

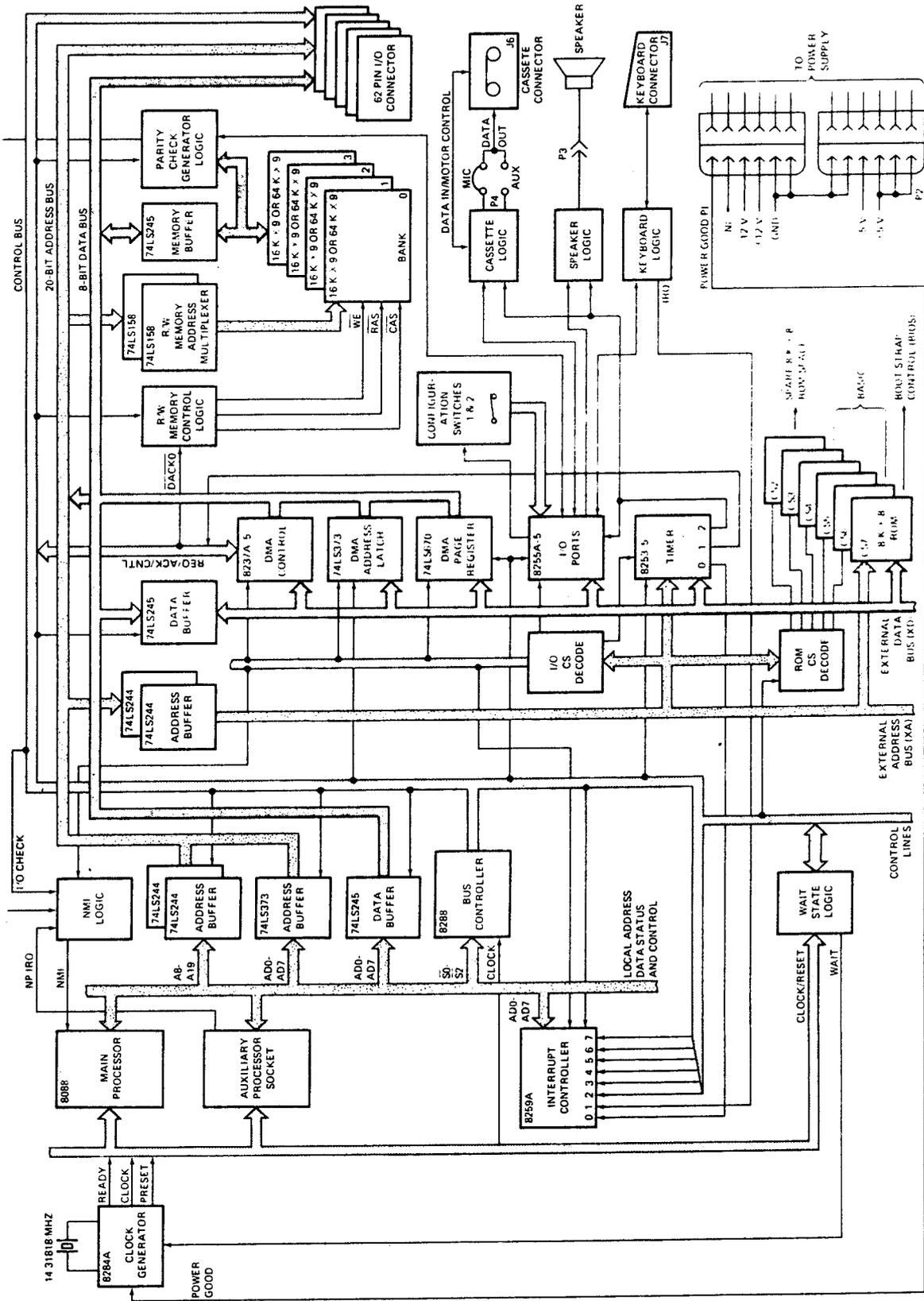


Fig 5.2 Block diagram of circuitry on IBM PC mother board.

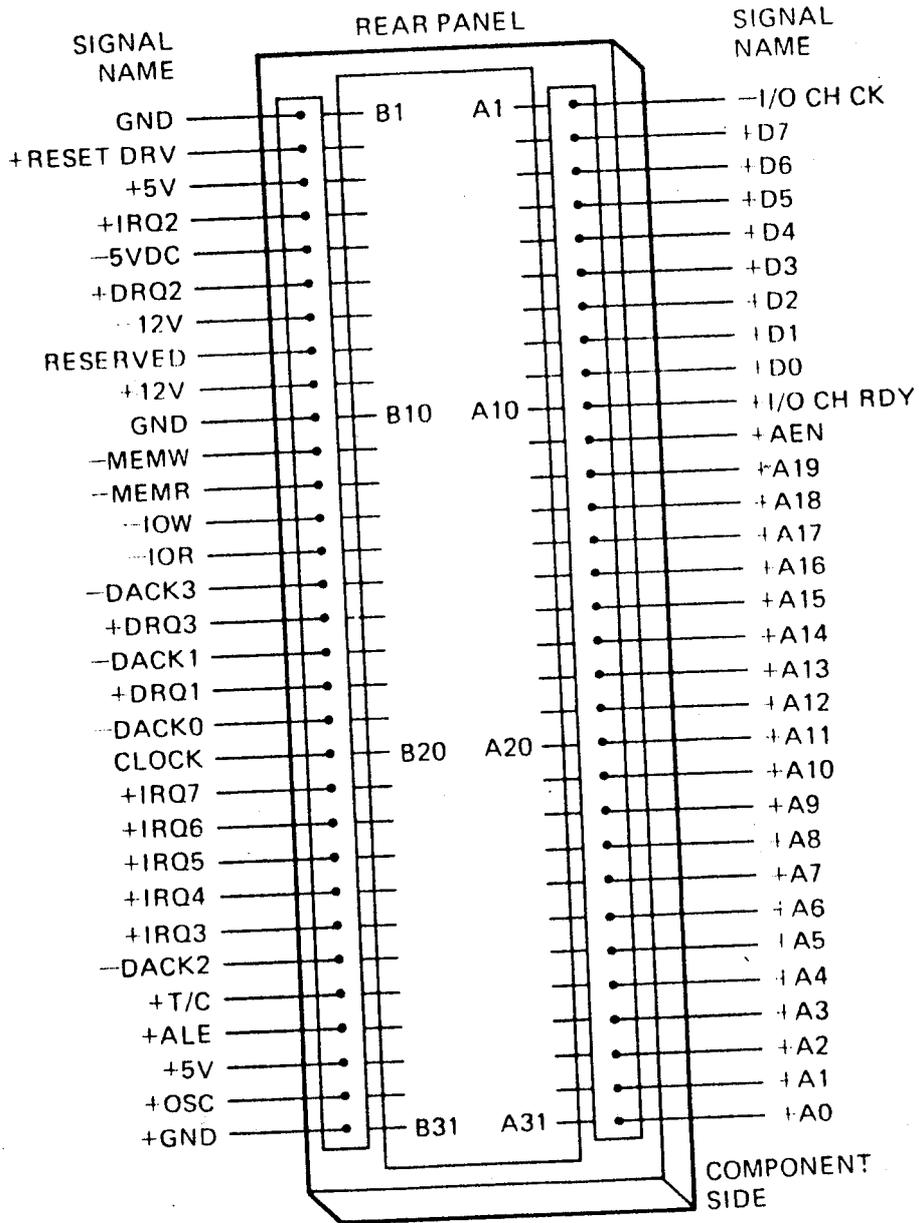


Fig 5.3 Pin names and numbers for Peripheral slots on IBM PC mother board .

## CHAPTER 6

### DESIGN OF SOFTWARE

There are three ports in a 8255 chip namely Port A, port B and port C. A part from it there is a control register. The control register is sometimes called as Port D. The decimal and hexadecimal address of the ports for 8255 AC-2 is as given below.

The data contained in port D is called as control word.

The control word Format for I/O mode is shown in Fig 6 (i),(ii),(iii)

address		register
dec.	hex.	
768	300	Port A
769	301	Port B
770	302	Port C
771	303	Port D

The computer can read the data present in these ports by simply giving in the port address. The computer can also store data in the ports by giving suitable GW BASIC command.

#### 6.1 Software procedure for reading data:

- i) The computer feeds into the control register 771dec.(303H) the value 147 dec(=93H).

The control register now contains

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
1	0	0	1	0	0	1	1

= **147 dec.**

This implies that now the port A, port B and lower part of C (lower 4bits) are ready to be read. While the upper 4 bits of C are in output mode.

ii) The computer feeds in 128Dec(=80Hex) to port C.

The port C will have

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
1	0	0	0	0	0	0	0

Then PC<sub>7</sub> bit gets high and it activates the monostable multivibrator IC7(74121), which responds by resetting all counters by means of its output signal at pin 6. A counter state of 0 results in a low level at the input of N<sub>15</sub>. Also Es1 is closed, so that the circuit is ready for a new measurement cycle.

iii) The computer writes 0 to port C which makes the port C

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
0	0	0	0	0	0	0	0

The PC<sub>7</sub> bit is brought back to the low level.

iv) Port D (control register) is written with the control word 115 dec (9BH). Port D becomes

D <sub>7</sub>	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
0	1	1	1	0	0	1	1

A 0 in D<sub>7</sub> implies it is in BSR mode & hence the bits of port C are reset.

v) Time is allowed for the port C to read-in data again.

vi) The data contained in Port A, Port B and Port C are read.

vi) The number of pulses counted will be then given by

$$\text{Count (or) Pulses} = 32768 * A + 256 * C + B$$

The factors 32768 ( $2^{15}$ ) & 256 ( $2^8$ ) come because of the reasons earlier explained.

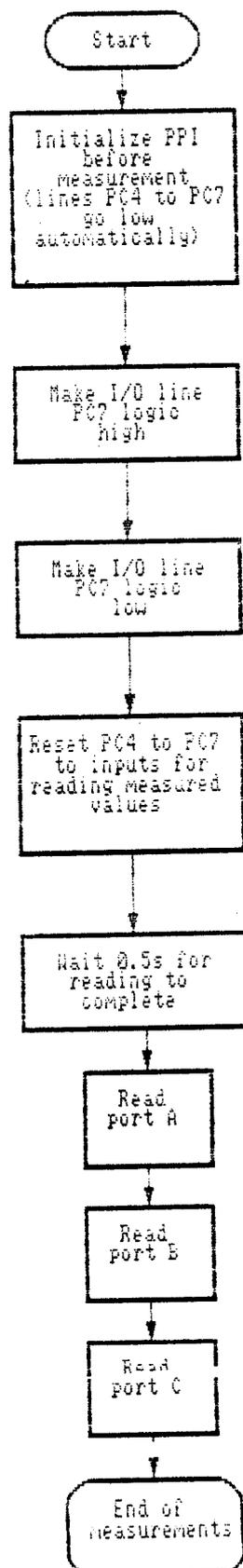
vii) The number of pulses (or) count divided by the gate time would give the frequency.

$$\text{Frequency} = \text{count/gate time}$$

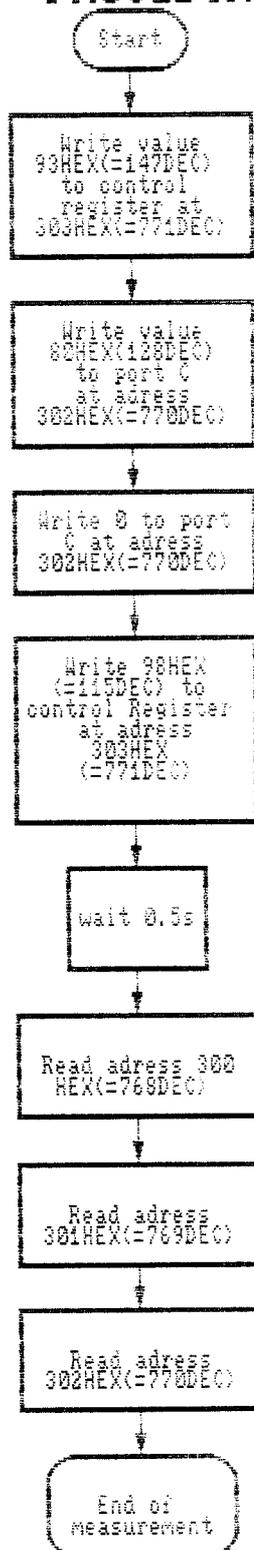
The program is simply typed in under GW BASIC, saved to disk and started with RUN command. LED D<sub>1</sub> flashes during measurements to indicate activity of the card.

Flow chart of the software that controls the frequency meter card is given in fig 6.1(i),(ii),(iii).

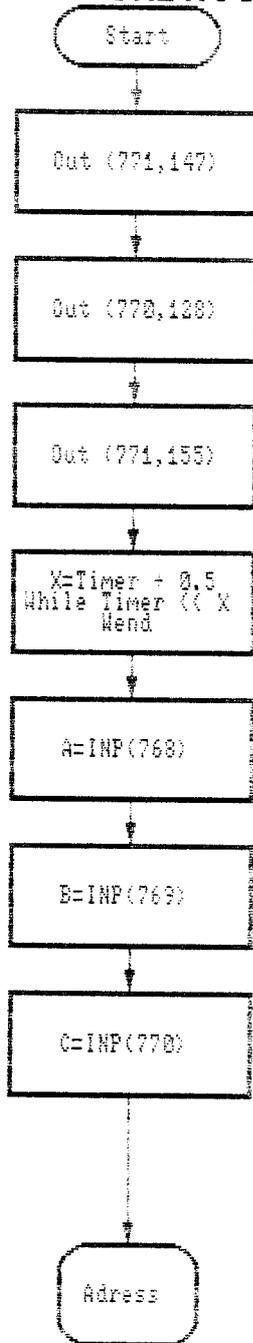
## 6.1 (i) DESCRIPTION OF MEASUREMENT PRINCIPLE



## 6.1 (ii) COMMANDS IN MEASUREMENT PROCEDURE



## 6.1(iii) MEASUREMENT PROCEDURE IN GWBASIC



## 6.2

```

5 PROGRAM LISTING OF THE GWBASIC PROGRAM THAT CONTROLS THE FREQUENCY
6 METER CARD
7
8
10 CLS: BEEP: KEY OFF
20 GATETIME=.262144: FACTOR=1: B$="Hz"
30 GOSUB 460: REM START FIRST MEASUREMENT (ALSO INITIALISES PPI 8255)
40 PRINT"
50 PRINT" *****
60 PRINT" *** PC FREQUENCY METER *** *** KIRTI SIVA VENGI ***
70 LOCATE 23,1
80 PRINT" H=> Hz K=> KHz M=> MHz R=> RANGE Q=> QUIT"
90 REM TTL
100 LOCATE 5,8: PRINT"RANGE 1: 4 Hz - 32 MHz (+/- 4 Hz) "
110 LOCATE 7,8: PRINT" SWITCH TO LOWER INPUT SOCKET (TTL INPUT) "
120 PRESCALER = 1
130 GOSUB 350 : REM MEASUREMENT
140 GOSUB 280 : REM OPERATION
150 IF A$="R" OR A$="r" THEN BEEP: GOTO 180
160 GOTO 130
170 REM HF
180 LOCATE 5,8 : PRINT "RANGE 2: 30 MHz - 1 GHz (+/- 1 KHz) "
190 LOCATE 7,8 : PRINT "SWITCH TO UPPER INPUT SOCKET (RF-INPUT) "
200 PRESCALER= 256
210 GOSUB 350 : REM MEASUREMENT
220 GOSUB 280 : REM OPERATION
230 IF A$="R" OR A$="r" THEN BEEP : GOTO 100
240 GOTO 210
250 REM LEAVE PROGRAM
260 CLS: KEY ON : END
270 REM OPERATION SUBROUTINE
280 A$= INKEY$
290 IF A$="Q" OR A$="q" THEN GOTO 260
300 IF A$="H" OR A$="h" THEN FACTOR = 1: B$="Hz"
310 IF A$="K" OR A$="k" THEN FACTOR = 1000!: B$="KHz"
320 IF A$="M" OR A$="m" THEN FACTOR = 1000000!: B$="MHz"
330 RETURN
340 REM MEASUREMENT SUBROUTINE
350 WHILE TIMER <MITV: WEND : REM WAIT FOR MEASUREMENT INTERVAL TO LAPSE
360 A=INP(768)
370 B=INP(769)
380 C=INP(770)
390 GOSUB 460:REM START NEXT MEASUREMENT
400 C=C AND &H7F: REM MASK BIT 7 ON PORT C
410 COUNT = 32768!*A+256*C+B: REM USE BYTES READ TO CALCULATE COUNTER STATE
420 F=COUNT*PRESCALER/(GATETIME*FACTOR)
430 LOCATE 13,8: PRINT"FREQUENCY IN ";B$;"=";INT (F+.5);" "
440 RETURN
450 REM START NEXT MEASUREMENT
460 OUT 771,147
470 OUT 770,128
480 OUT 770,0
490 OUT 771,155
500 MITV=TIMER +.5
510 RETURN: REM

```

## CONCLUSION

In the concluding part of our report, we would like to admit with reasonable modesty that this frequency meter card for Personal Computer was successfully tested and has proved to be a success.

This is a plug-in-card i.e., it can be directly connected with the Personal computer eliminating the usage of external interfacing cords which are cumbersome to the user motherboard. It is a cost-efficient alternative to a stand-alone frequency meter. This project is simple to build and programme with a TTL input as well as a prescaler input. The other advantage is that it has a high sensitivity, very versatile, capable of handling frequencies upto 1 GHZ. The one draw back with this kit is that some components are imported and not readily available in India.

It has one limitation that it cannot measure frequencies higher than 1 GHZ. Hence, measurement of high oscillating frequencies are not possible. This card's scope is restricted to stable and continuous signals, and widely varying oscillating signals cannot be faithfully read.

This project work can be amended to work as a counter or divider circuit by slightly altering the software language, G.W.Basic which can be easily understood. Also this card can be suitably protected to provide immunity against external field

disturbances for a more accurate and sensitive measurement. Finally, we encourage our predecessors to actively take up further developments and improvements in this project to bring about a better version of the same.

## APPENDIX

### COMPONENTS LIST :

#### Resistors:

R1 = 6K8

R2 = 1K5

R3 = 330

R4 = 75

R5 = 4KT

R6 = 100K

#### Capacitors :

C1 = 1No; 10v ; radial

C2;C3 = 1 No

C4;C5;C6 = 100n

#### Semiconductors :

D1 = LED(5mm)

IC1 = 8255AC-2

IC2 = 74LS245

IC3 = 74LS30

IC4 = 74LS04

IC5 = 74LS00

IC6 = 74LS32

IC7 = 74121

IC8 = 4066

IC9;IC10;IC11;IC12 = 4040

IC13;IC14 = 74HC7393

IC15 = 74LS02

IC16 = 74HC704

IC17 = U6060BC (Telefunken)

## BIBLIOGRAPHY

1. **Elector India, February 1990 Issue.**
  
2. **MICROPROCESSORS AND INTERFACING  
Programming and Hardware.**  
by Douglas V.Hall  
McGraw-Hill Book Company
  
3. **Digital Principles and Applications**  
by Albert Paul Malvino  
and Donald P.Leach  
McGraw Hill Book Company
  
4. **Microprocesor Architecture, Programming and Applications.**  
By Ramesh S. Gaonkar  
Wiley Eastern Limited.