Pc Based Multifunction Test Instrument

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.

BY

E. ANAND.
S. GANESH.
D. KARTHIKEYAN

UNDER THE GUIDANCE OF

Dr. K. A. PALANISWAMY.
B.E. M.Sc., (Engg.) Ph.D., M.I.S.T.E. C.Eng. (I), F.I.E

1992-'93

Department of Electrical and Electronics Engineering

Kumaraguru College of Technology

COIMBATORE - 641 006.

Certificate

Department of Electrical and Electronics Engineering

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006.

Certificate

NAME ..... ROLLNO. .....

Register No. .....

This is the bonafide record of the project work titled

"PC based Multifunction test instrument"

done by

Mr. .....

in partial fulfilment of the requirements for the award of the Degree of Bachelor of Engineering in Electrical and Electronics Engineering Branch of Bharathiar University, Coimbatore during the academic year 1992-93.

Dr. K. A. PALANISWAMY, B.E.(M.Sc.(Engg).Ph.D.
M.I.E.(I).C.Engg(I).,FIE.,
.....Professor and Head.....
Department of Electrical and Electronics Engineering,
Kumaraguru College of Technology.
Coimbatore 641 006

Guide

Dr. K. A. PALANISWAMY, B.E.(M.Sc.(Engg).Ph.D.
M.I.E.(I).C.Engg(I).,FIE.,
Professor and Head
Department of Electrical and Electronics Engineering,
Kumaraguru College of Technology.
Coimbatore 641 006

Head of the Department

Submitted for the University Examination held on .....

.....
Internal Examiner

.....
External Examiner

Acknowledgement

Acknowledgement

We would like to express our Profound appreciation and gratitude to our guide Dr. K.A. Palaniswamy B.E. M.Sc.(Engg),Ph.D., M.I.S.T.E., C.Eng(I), FIE, Professor and Head of the Department of Electrical and Electronic Engineering who has given us invaluable help and support to complete this Project success fully.

We are also indebted to Mr. A. Ramasamy M.E., Asst. Professor of the C.S.E. Department and Mr. R. Shanmugashankar B.E, of the C.S.E. Department who were immensely helpful in providing us the Computer facilities for the development of the software.

Lastly we would like to express our thanks to Major T.S. Ramamurthy, B.E., M.I.S.T.E., Principal for his encouragement.

Synopsis

SYNOPSIS

This project deals with design and construction of a pc Based Multi function test instrument. The project aims to build an instrument which integrates common measuring functions (like Multimeter functions plus other functions like capacitance, frequency and phase measurements. To enhance the flexibility and the through put of the test instrument a pc has been used.

The instrument requires minimum manual operations and has facilities like autoranging and data logging. Software has been written in Quick Basic interfaced with Assembly Language. Assembly Language, is used for actual interface to the instrument hardware. Quick Basic is used for providing mean and statistical functions plus a variety of other functions like help etc.

Contents

CONTENTS

Introduction

CHAPTER - I

INTRODUCTION

Nearly all instruments today are based on digital techniques of measurement. The basic advantages of digital instruments is the case of their operation. The readings are easy to take and does not lend itself to errors of interpretation. Moreover, the number of ranges is limited in steps of 10 (as opposed to $[10]^{1/2}$ steps used or analog instruments). The concepts of digital instrumentation has further been revolutionised by the advent of microprocessor and microcomputers.

The instrument described here is capable of measuring the following quantities

1. Voltage (dc and ac)
2. Current (dc and ac)
3. Resistance
4. Capacitance
5. Frequency and
6. Phase difference

The instrument has also got provision to log measured data and retrieve when required. The details of each section, their principle of working are explained in the following chapters. The instrument is based on an IBM/IBM compatible pc. Hence, the

flexibility of the instrument is greatly enhanced. The instrument is auotmated and except for high frequency calibration, little manual intervention is required. The use of a pc has added "Intelligence" to the instrument. We may define an intelligent instrument to be one where after measurement has been made of a variable some further processing (usually digital) is carried out to refine the data and present it to the observer. In contrast to this we have the "dumb" instruments, like the oscilloscope where the observer has to interpret the data. The advantage of an intelligent instrument over the "dumb" instrument is then obvious.

1.1 Some Important Terms and Their Meanings

The instrument is capable of providing mean and statistical functions. These functions are particularly relevant and important to low frequency measurements. Before dealing with the need for statistical functions it would be proper to define some important terms regarding measurement.

1. Measurement

It is the emperical objective assignment of number to properties of objects or events in the real world in such a way as to describe them.

Measurement is not a description of the object but a description of the properties of an object.

2. True Value

It is the value of a measured as indicated by the best available standard instrument.

Error (E) is the difference between the result of a measurement and the true value of the measured quantity.

Error is further classified as fullscale error, relative error and obsolute error.

Errors can be either random or systematic with systematic problems, because they can be, in general, qualified. For eg. systematic errors can be offset error in an operational amplifier, non-linearity error in an ADC etc.

Random errors cannot be qualified. Hence we have to call in statistics to perform the job of predicting the deviation from the actual value. (approximate prediction). Also, any measured value subjected to random errors will be inaccurate. To overcome this a large number of readings may be taken and averaged. This gives the likely mean. If an envelope drawn around all the readings taken according to the frequency of occurance then a spread of values will be seen. This envelope takes the shape of what is known as the Gaussian Probability distribution function (or a normal probability density function). Then it is possible to use the concept of standard deviation to ascribe an error band to any reading made. It is further possible to give a likelihood of any reading being in a particular part of the distribution or be within a specified error band, viz.

68.3 % of all reading lie within $\pm$ 1 S.D of $\bar{x}$

99.7 % of all reading lie within $\pm$ 3 S.D of $\bar{x}$

where S.D is the standard deviation $\bar{x}$ is the mean.

It is then, possible to predict the complement i e 5% chance of a reading being outside $\pm$ 1.96 S.D of $\bar{x}$.

Thus the shape of the curve is important. Figure 8.1 shows curve (a) has a tight error band (small S.D) whereas curve b has a very wide error band (large S.D) . The prob able error is often given for a 50% probability as lying within a range of $\pm$0.675 S.D.

The standard deviation is given by S.D = $[\bar{x}^2 - (\bar{x}^2)]^{1/2}$ where x is the mean of the square of the measured readings.

The importance of standard deviation can be thus seen. Great caution should be excercised while using mathematical tools such as statistics. They appear to have a great deal of authority care should be particularly excercised when for reasons of time or expense only a few readings are taken.

1.2.2The pc and Its Role

1.2.1 IBM pc System (PC/XT)

The block diagram of the IBM PC/XT system unit is shown in fig. 8.2. The heart of the system is the intel 8088 processor. It is operated in the maximum mode so that a co-processor (such as 8087) can be added. At 4.77 MHz clock rate the 8088 bus cycles are

four clocks of 210 ns or 840 ns. The processor is supported by a set of high function support devices providing four channels of 20 bit DMA, 3-16 bit timer-counter channels and 8-prioritised interrupt levels. Three of the four DMA channels are available on the I/O bus and support high speed data transfers. The fourth DMAchannel is programmed to refresh the system dynamic memory.

The three programmable timers/counters are used as follows. Channel 0 is used as a general-purpose timer providing a constant time base for implementing a time of day-clock. Channel is used for DMA transfer and channel 2 is used for generating sound tone for the audio speaker.

Of the eight prioritised levels of interrupt, six are bussed to the system expansion slots for use by feature cards. Two levels are used by the system board. Level is attached to the channel 0 of the timer-counter and provides periodic interrupt for time of the day clock. Level 1 is used for interrupting the processor, whenever a scan code is sent from the key board.

The system board supports both ROM and R/W memory. It has space of 64 K x 8 ROM. ROM contains the routines for power on self test, I/O drivers, dot patterns for 128 characters in graphics mode and a diskette boot strap loader.

The system also has RAM which is expandable to 640 K (a minimum of 128 K is necessary). All R/W memory is parity checked.

The system board has adapter circuits for interfacing keyboard through a 5-pin DIN connector.

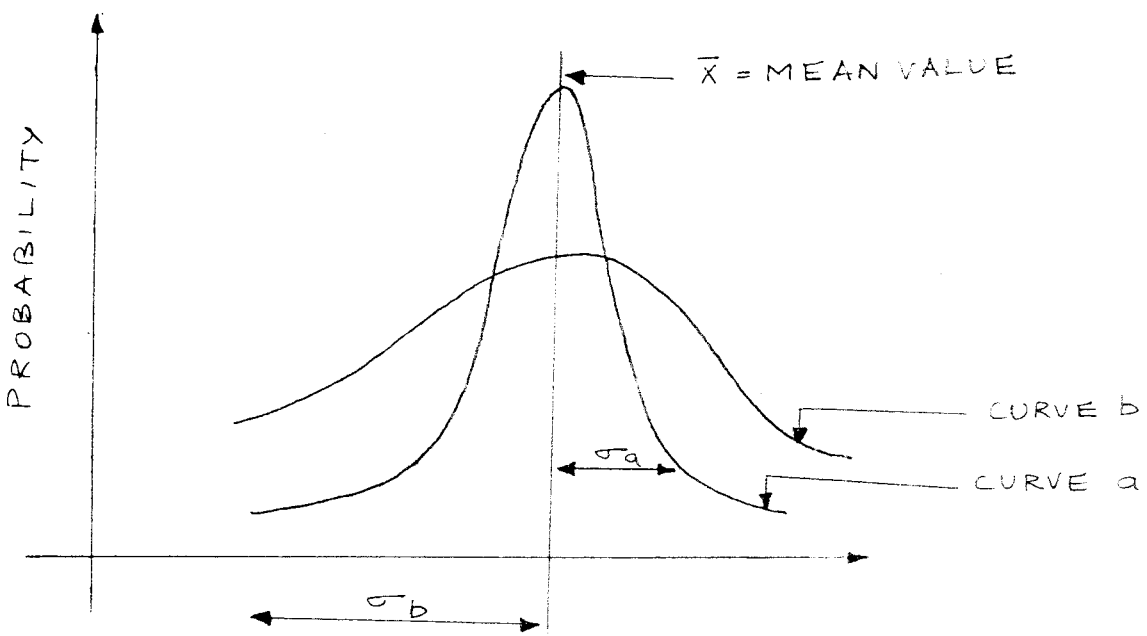


FIG- 1.1. NORMAL PDF CURVE

Operation of the instrument

CHAPTER -II

OPERATION OF THE INSTRUMENT

2.1 Block Diagram

Fig. 2.1 shows the block diagram of the test instrument. The instrument is broadly classified into two sections namely the voltage section and the frequency section. These sections are named so because the quantities voltage, current and resistance are converted to a proportional voltage and the quantities frequency, capacitance and phase are converted to a proportional frequency before they are actually measured.

The voltage section consists of programmable gain amplifiers (as the name implies, the gain of these amplifiers are selected by software control), analog multiplexors, sample hold amplifier, RMS-DC convertor and an ADC. The frequency section consists of counters, multivibrators, comparators and latches. The actual hardware is described in the next chapter. When a particular function is chosen by the user, the instrument switches to the proper function and automatically arrive at a proper range by switching and comparing with a known value. If the value is found (after comparison) to be too high, then the next lower range of gain is selected and vice versa. The range being selected, the analog data may then be converted to a digital data by the ADC.A sample and hold amplifier is used to measure dynamically varying inputs(over a smal margin)

2.2 DC Voltage measurement

The range of dc voltage that can be measured is from 0 to ± 200 volts with a resolution of 4.88 mv. When the function dc voltage measurement is selected the gain of the programmable amplifier is kept at minimum. The output digital voltage is compared with a known voltage (±1 volt). If the range is outside ±1 volt then the voltage is measured. If the range is within ±1 volt then the gain is set to the next higher value. By this process the correct range may be found out. The lowest voltage to be measured is restricted by the resolution of the ADC. The highest voltage is ±200 volts. The block diagram and the actual circuit are shown in fig. 2.2

2.3. AC voltage measurement

Until recently the RMS value of an ac signal was found out by measuring the average value and then multiplying it by the form factor. (1.1 in the case of a sinewave). This procedure works well for sinusoidal waves but the accuracy is poor for non sinusoidal ac inputs such as square waves.

To overcome this problem a true RMS-DC convertor is used. A true RMS-DC converter is a device which converts a signal AC, DC + AC) to its equivalent DC heating value. The RMS value of a wave form is a fundamental measurement of that waveform. It may be noted that any signal (irrespective of their phases) will dissipate the same amount of energy over a period in a resistor if it has the same RMS value . Also Random signals can be described only in terms of RMS value and not in terms of amplitude and phase such as

periodic signals. An interesting point to note is that the RMS value of any stationary zero mean random signal is equal to the standard deviation of the signal. Since the mean square value (RMS) provides a universal scale of measurement.

For waveforms with harmonics a RMS to DC converter can be used effectively. For example a voltage from an SCR controlled circuit. An averaging meter would read correctly only during180^{0} conducting angle. It would read 51% error in 45° conducting angle.

In the instrument the ac voltage is converted to its RMS value before being converted by the ADC to a digital value. The process of range selection is the same as in the DC voltage measurement. The range is from 0 v to 200 volts (RMS). The actual circuit is shown in fig 2.3. The output stage is shown in fig 2.4

2.4 Current Measurement

2.4.1 Low Range (500 nA to 100 mA)

The measurement of current is similar to that employed commonly. The current is passed through a know resistance. The voltage across this resistance is thenproportional to the value of the unknown current. A low offset current operational amplifier is used here since currents of the order of a few hundreds of nano amperes have to be measured. The CMOS switches are properly switched to get the appropriate voltage output. This voltage is given to the input of the ac voltage measuring section which the measures this value. The op- amps is configurated in differential input mode to give a output of-2 $I_{in}R$.

2.4.2 High Range (100 μ A to 1 Amp)

In this case the current is first dropped through a 5 Ω , 10 watt resistor. This voltage is fed to a singal ended output operational amplifier. The voltage is given to the input of the ac voltage measuring section (as in the low-range) case. Here the voltage O/P = - 5 I x R_2/R_1, R_2- Feedaback resistance R_1- Input resistance. The circuit for low range and high range current measurement are shown in fig. 2.5 and fig 2.6.

2.5 Resistance Measurement

Two accurate reference voltages are used for generating a current source. Two voltages are used to extent the range of measurement. A precision zener-voltage reference LM 385 (2.5 v) is used. A 5 volt reference is generated by converting this 2.5 volt to 5 volts by using a non-inverting operational amplifier with $R_f/ R_1 = 1$. The 5 volt is divided by 10 using a divider network and the two reference voltages are buffered. The range of resistance measurement is from 200 Ω to 20MΩ . This range is limited by the ON resistance of the CMOS switch. Even though the error introduced by the ON resistance can be taken care by software, it is safer to restrict the range to 200 Ω, as random errors are bound to creeptin the practical ON resistance by a substantial value. The 5 volt reference is used on range upto 20k and the 0.5 volt reference is used on the 200 Ω and 2 k Ω ranges.The out put voltage which is fed to the dc voltage section may be either

$$\frac{5 \times R_x}{R_1} \quad \text{or} \quad -0.5 \times \frac{R_x}{R_1}$$

where Rx is the unkown resistance and R_1 is the input switched reistance.

R_1 is a variable and can be suitably selected by comparing the voltage output. The resistance measuring circuit is shown in fig 2,7

2.6 Capacitance Measurement

2.6.1 Low Range

The low range for capacitance measurement is from 10pf to 1 F. The measurement is based on P M techniques. An Astable multivibrator produces a clock which is divided by four using tow 'D' type positive edge triggered flip flops. The output of these flilp flops serve as the trigger for a monostable multivibrator. The output pulse width (1.1 RC) is proportional to the value of the unknown capacitance. Since the value of R, the range resistance, is known, by measuring the pulse width, the value of the unknown capacitance can be found out. The circuit is shown in fig 2.8

2.6.2 High Range

In the circuit in fig 8.11a, transistor Q, the diodes, R_1 and R_3 form a constant current source for charging the capacitance under test. Another constant current source consisting of R_2 and Q_2 (andusing the diodes) provides current to R_4, R_5 and R_6 to make up a dual reference voltage. When the switch S_1 is open the capacitance is charged lineraly as shown in fig 8.11b and the 741 buffers this voltage from the relatively low input impedance of IC_2 and IC_3. IC_2 whose ouptput is normally at ground, switches to a positive output. A short time later when the voltage from the buffer reaches the reference voltage of the IC_3, the O/P o IC_3 switches from positive. Diodes D_4 and D_5 detect this condition

and when it occurs R_7 causes the junction of the diodes to go positive, producing a pulse whose width is proportional to the value of the unknown capacitance. By having a suitable calibrating capacitor accuracies of 0.1% may be obtained.

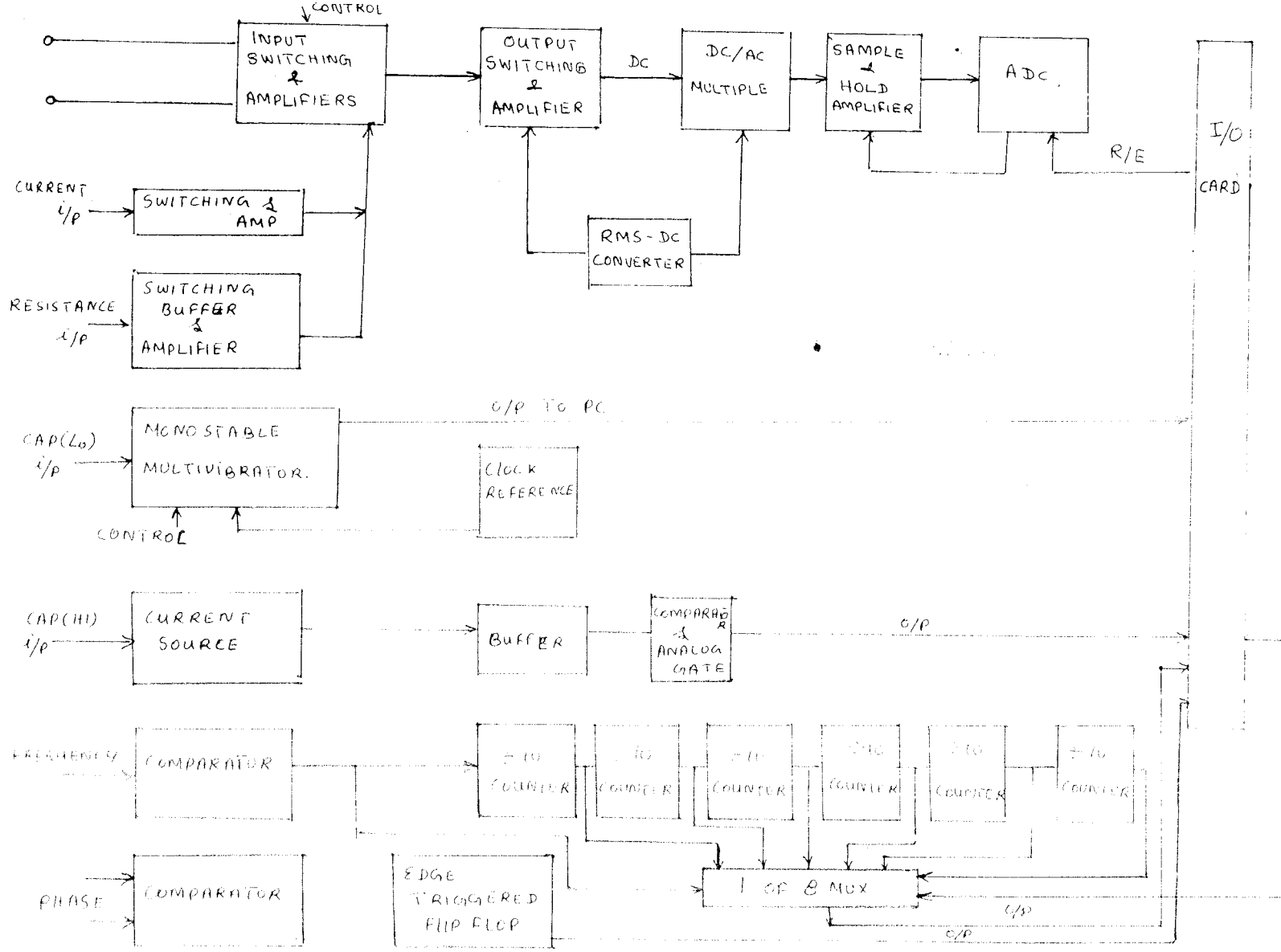
2.7 Frequency Measurement

The input signal whose frequency is to be measured is first shaped to a square wave using comparators. The output of the comparators are fed to 6 symmetrical divide by 10 counters and each of these 6 outputs plus the comparator output are fed to a digital mulitplexor. The frequency is found out by measuring the pulse width, using the pc. First the range is set at by 10^6 and if the output here is '1' for more than 1 milli second the next lower range is selected. This process is continued till the pulse width is less then 1 ms and it can then be measured by software. The frequency measurement circuit is shown in fig 2.9.

2.8 Phase Measurement

The two signals whose phase difference is to be found out are first given to comparators. The outputs of these comparators act as clocks to two dual 'D'type edge triggered flip flops. the rising edge of the output pulse is controlled by the rising of the leading signal and the trailing edge of the output pulse is controlled by the rising edge of the lagging signal. By this method a 360^0 phase difference can be found out. The data bits of the two flip flops are permanently set while the set bits are permanently grounded. The relevant circuit and the waveform are shown in fig 2.10 and fig 2.11 .

FIG 2.1 BLOCK DIAGRAM



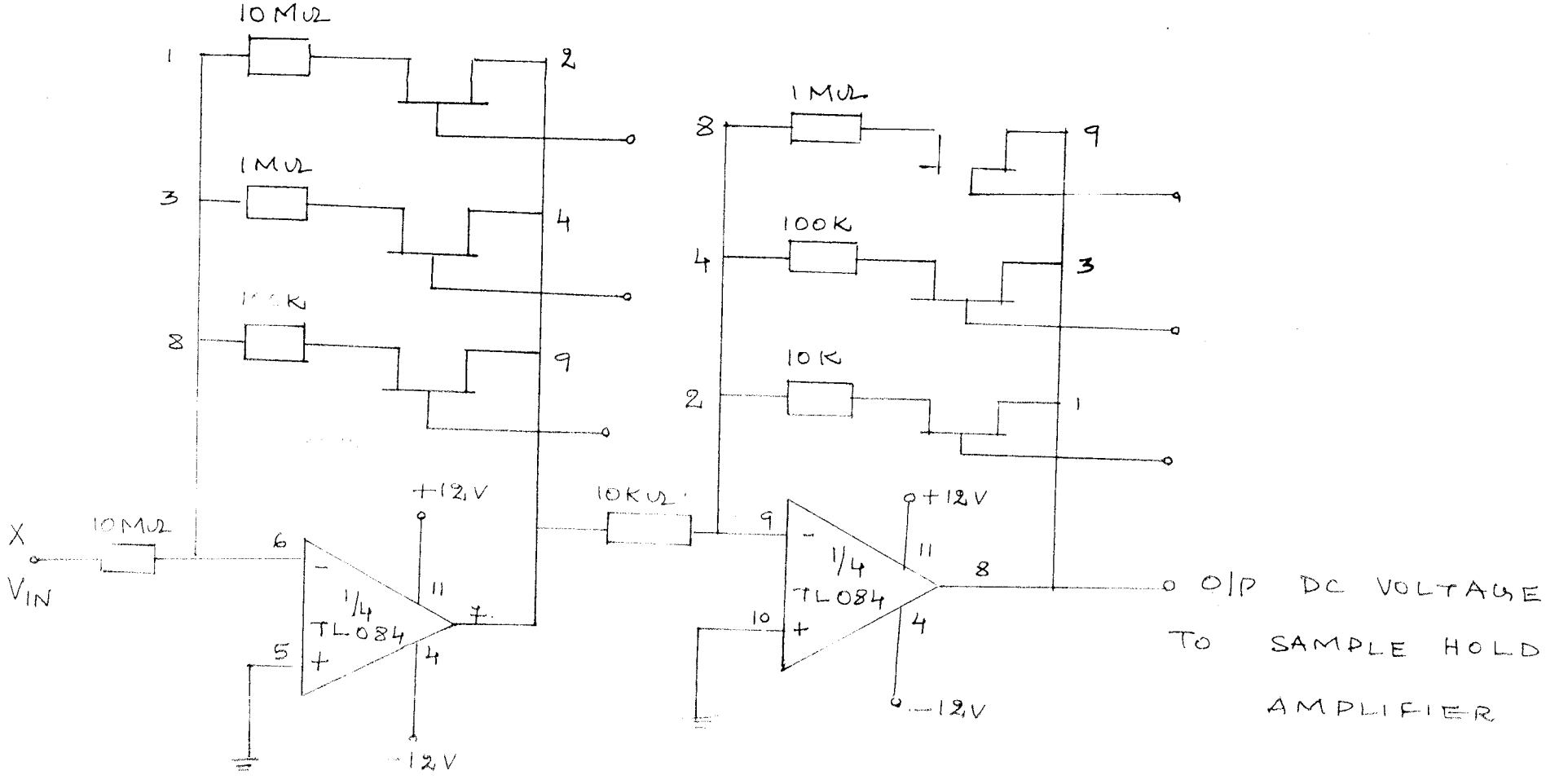


FIG - 2.2 T[illegible] VOLTAGE SECTION

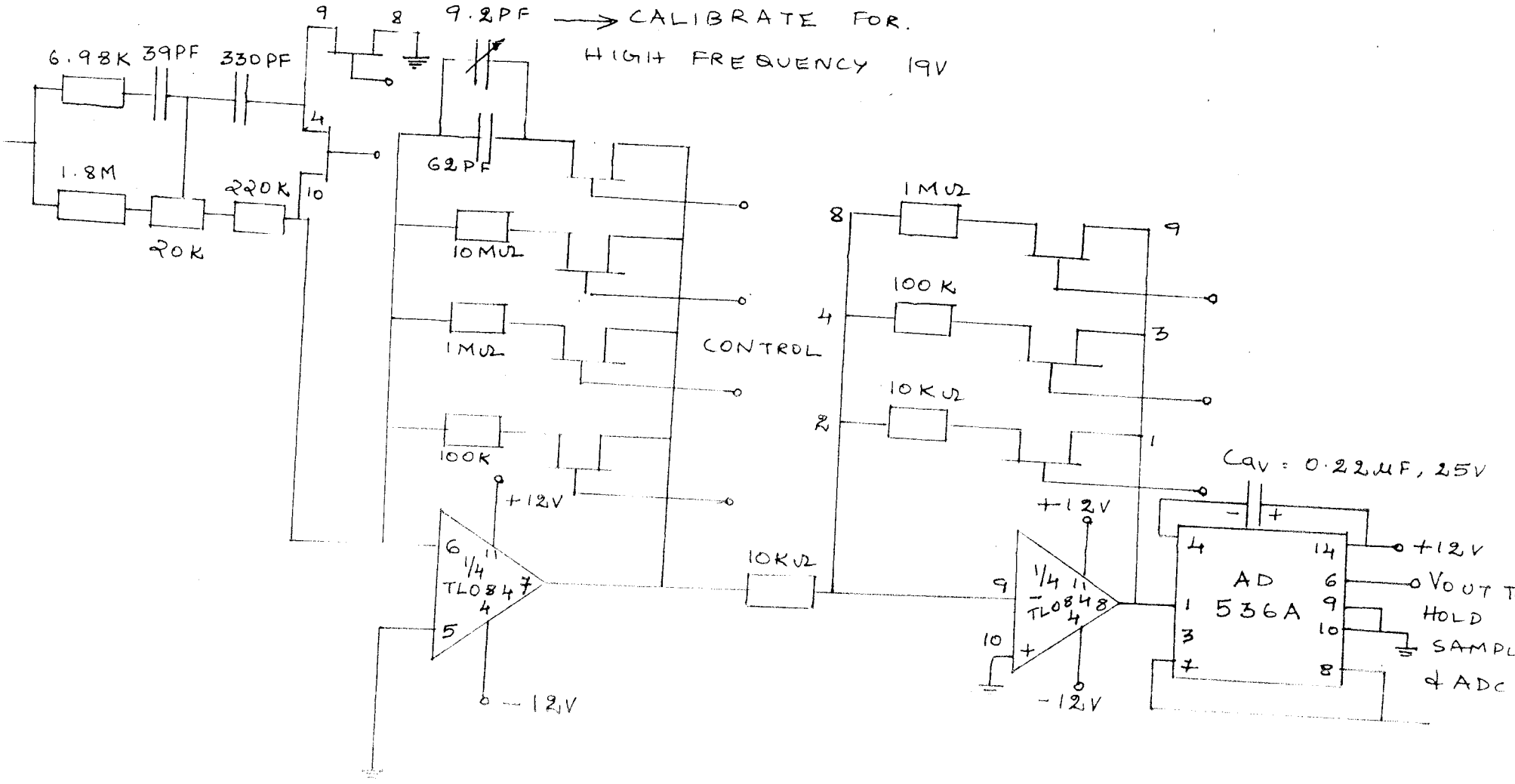


FIG - 23 AC VOLTAGE SECTION

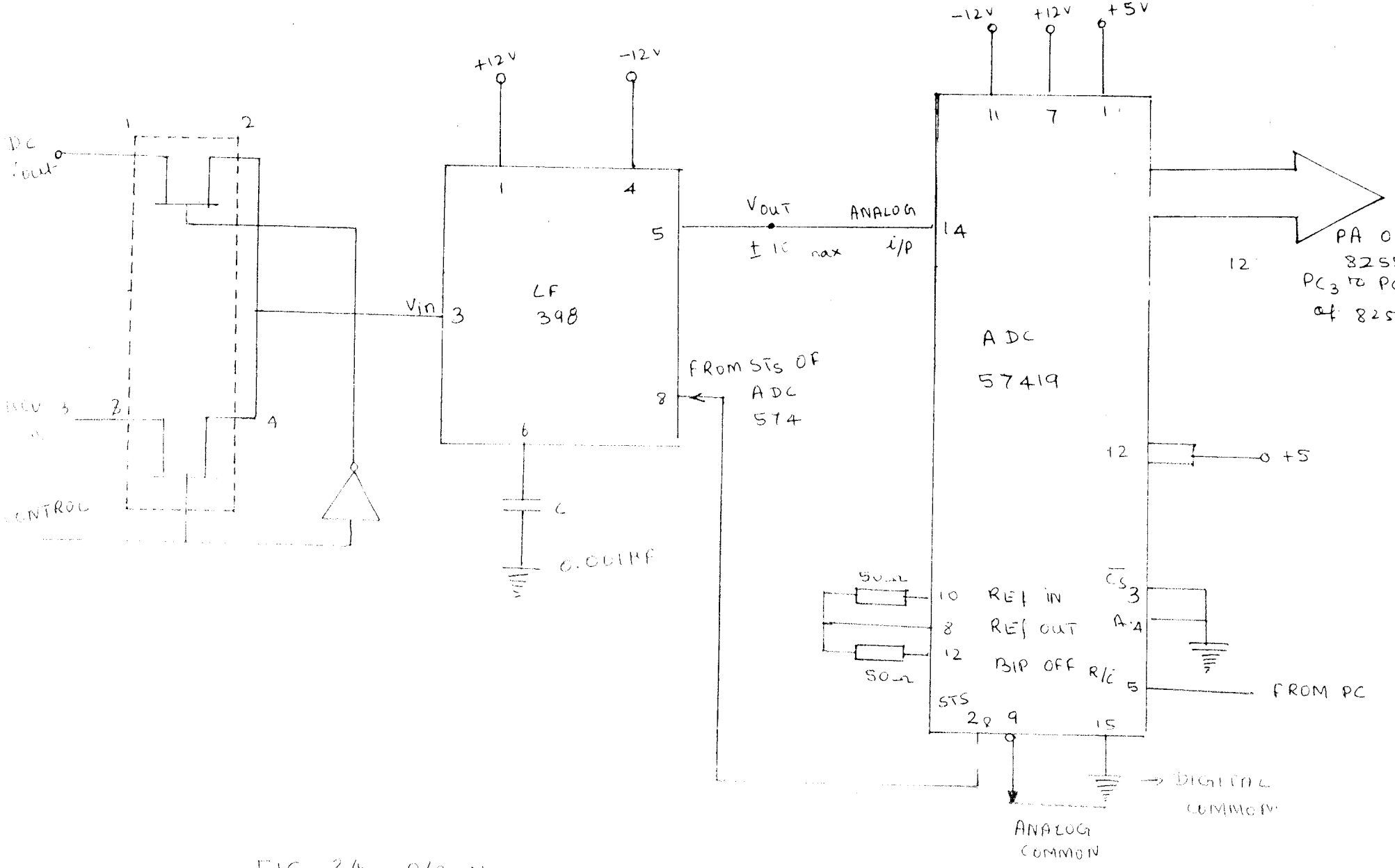


FIG. 2.4 O/P MEASURING SECTION.

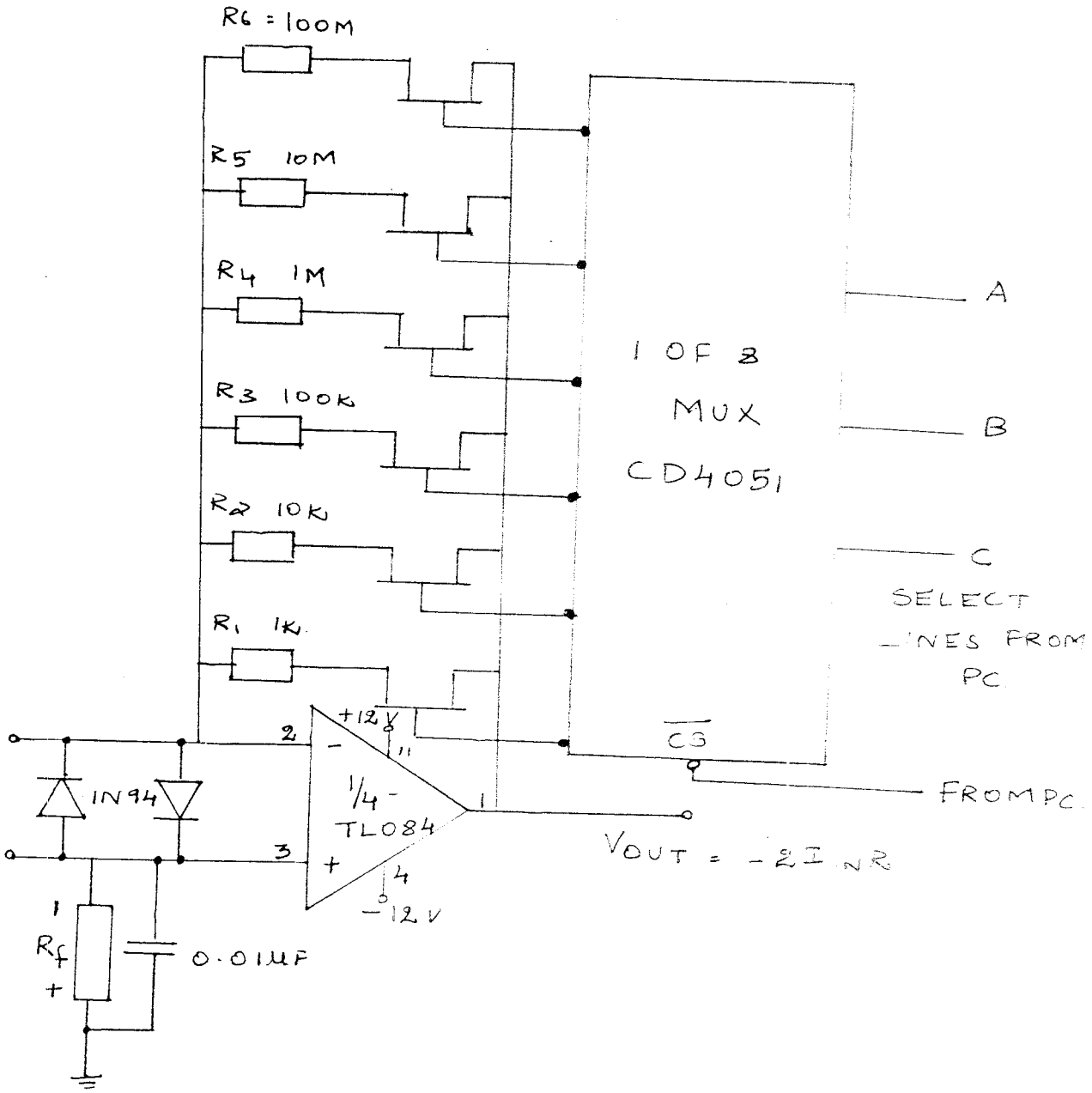


FIG - 2.3 CURRENT (LOW RANGE)

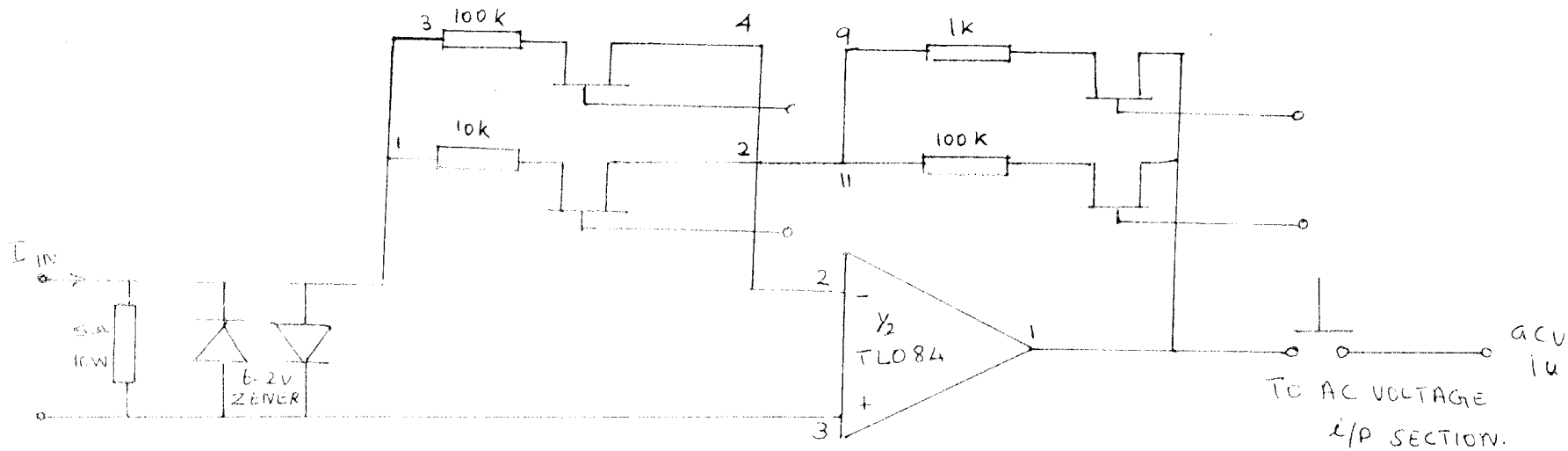
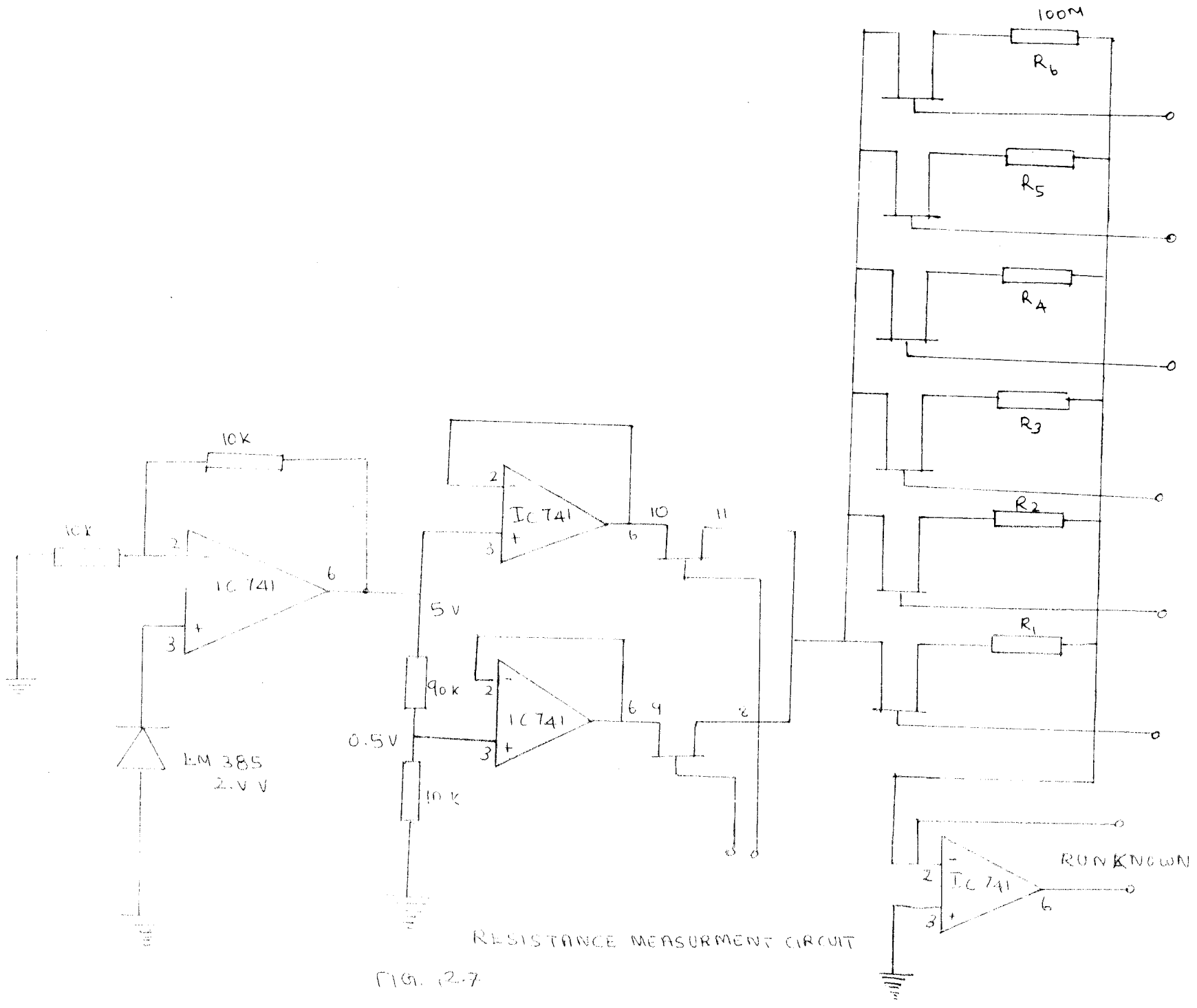


FIG 2.6 HIGH RANGE CURRENT.



RESISTANCE MEASURMENT CIRCUIT

FIG. 12.7

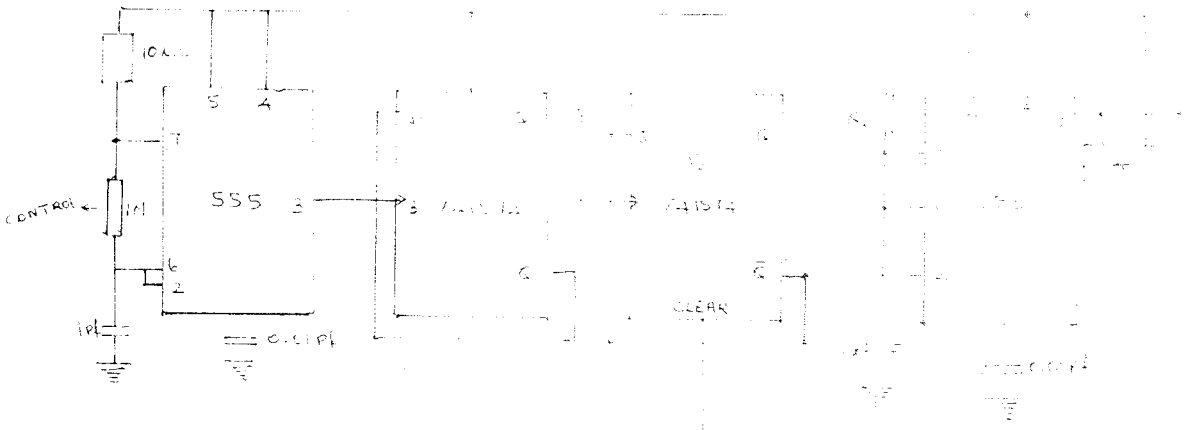


FIG 2.9 CAP LOW RANGE (10PF – 1MF)

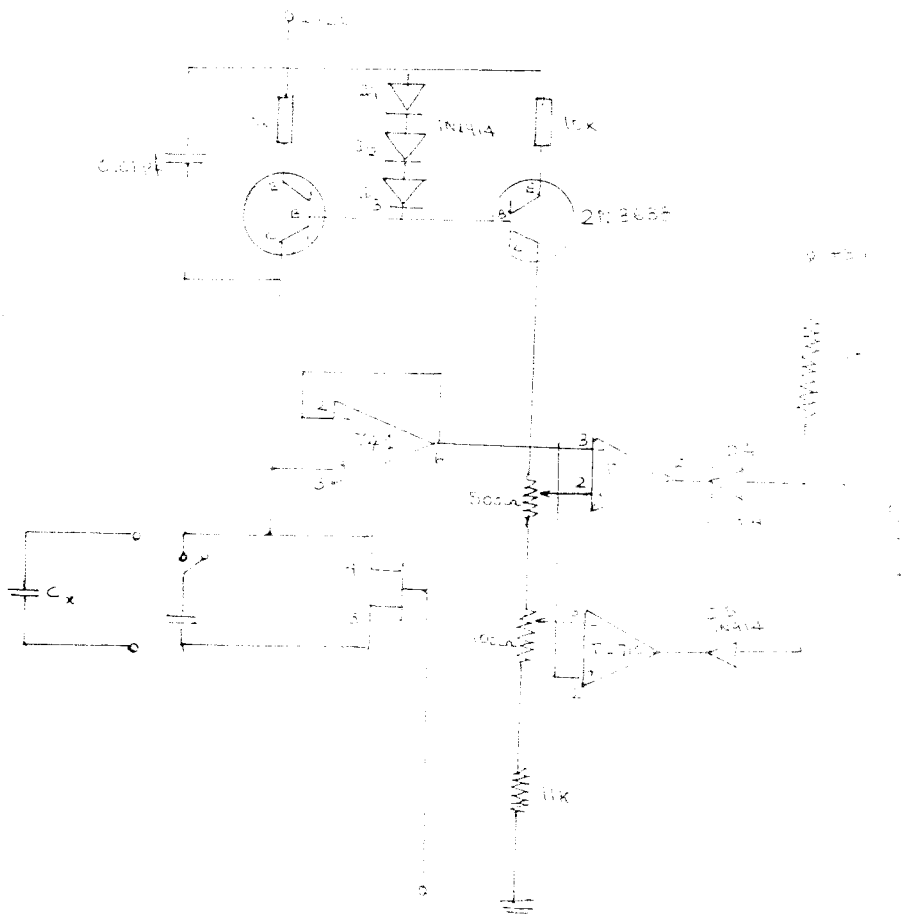


FIG 2.8a CAP HIGH RANGE (1MF – 1000MF)

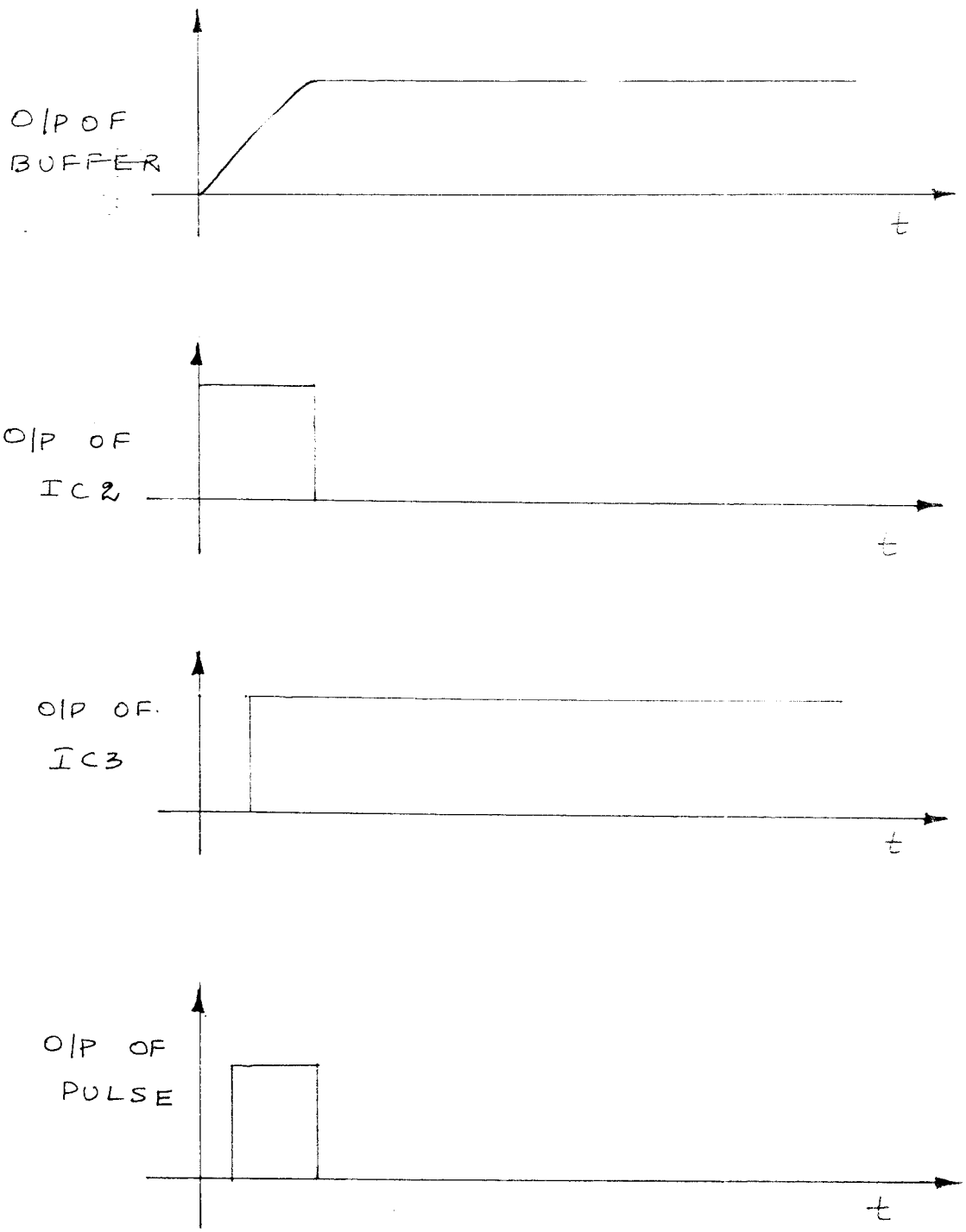
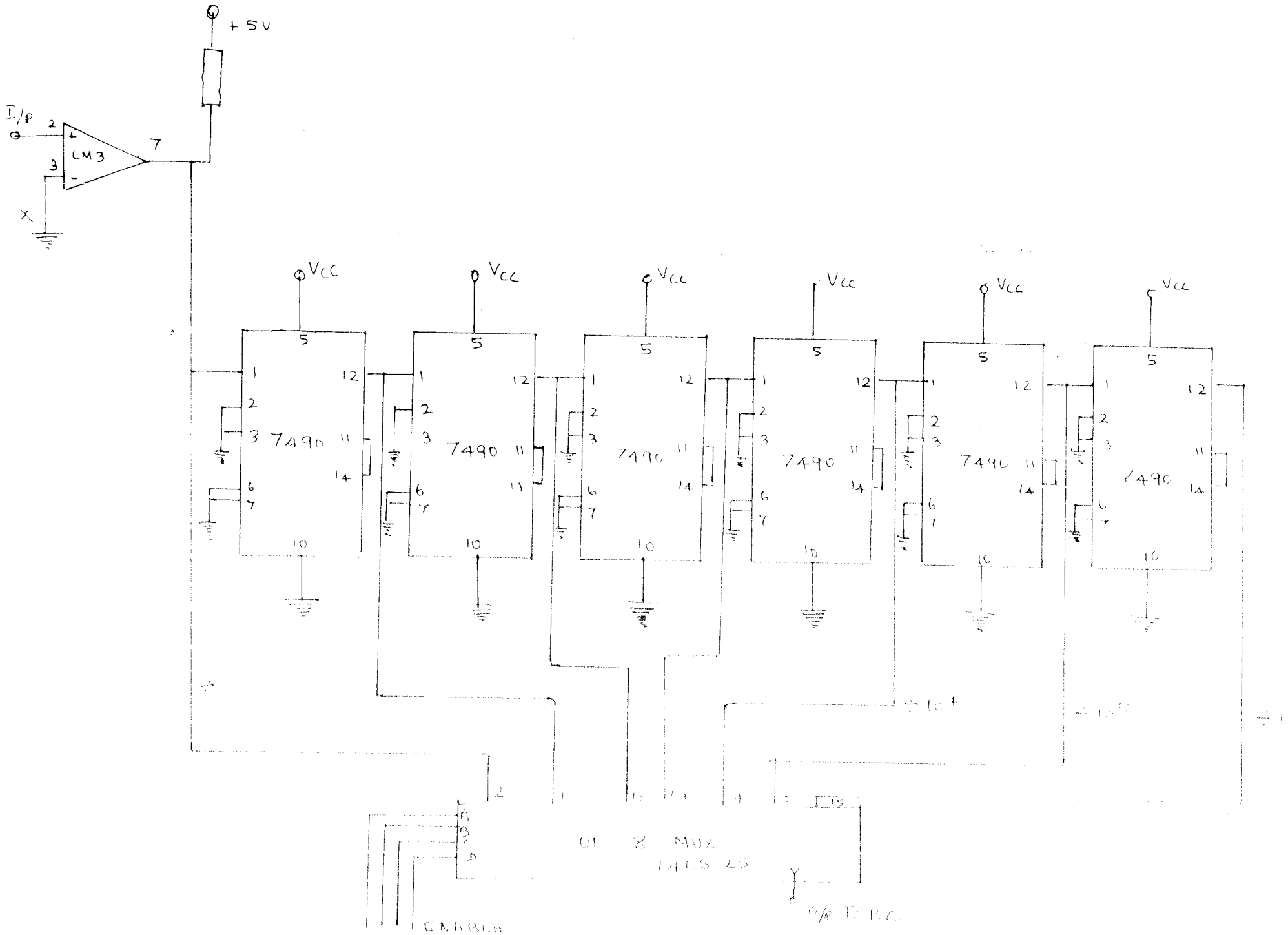


FIG - 2.8b WAVEFORMS FOR CAP HIG


+5V
I/P
LM3
X
Vcc
7490
÷1
÷10
÷100
÷1K
8 MUX
74LS 25
ENABLE
O/P

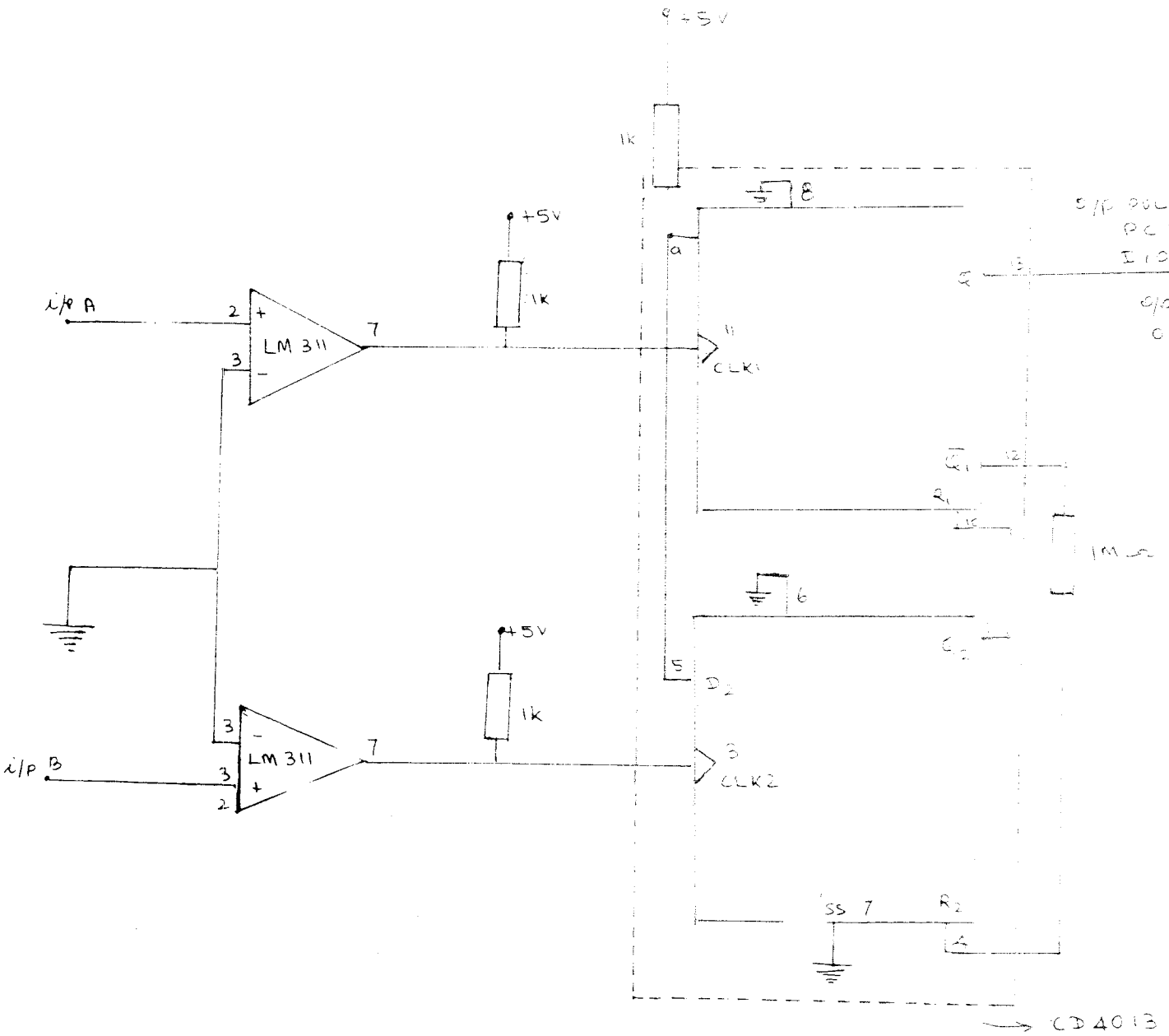


FIG 2.10 Q. PHASE MEASURING CIRCUIT.

Hardware and interfacing

CHAPTER - III

HARDWARE AND INTERFACING

3.1 Introduction of Instrument Hardware

The previous chapter has attempted to explain the basic principles of operation of the test instrument. This chapter will expalin the hardware devices used in the design and the interfacing of the instrument to the pc. The hardware used here are theones commonly used in pc based dataacquisition systems. They are (i) Programmable Gain Amplifiers (ii) Analog and Digital Multiplexers (iii) Sample and Hold Amplifier (iv) RMS-DC Converter (v) Clocked Circuits (like multivibrators) and (vi) Analog to Digital Converter. The interface to the pc is through a digital I/O card.

3.2 Programmable Gain Amplifier

Consider an Op-Amp configured in the inverting mode. The gain of the Op-Amp may be varied by varying the feed back resistance, or the input resistance or both. In our test instrument we have chosen to keep the input resistance constant at 10M and vary the feed back resistance to vary the gain. Thus, both a high input impedance and programmable gain are possible. The operational amplifier used here is the TL-084, a JFET input Op-Amp. This provides a high input impedance and alsohas very low offset currents (in the order of pA). It has four Op-Amps in ;one package and hence conserves area compared to the 741. This chipis hence well suited for measurement applications. The gain is raised by controlling the analog switches which are placed in series between the feed back resistance and the output. The analog switch used here is the CD 4016 quad switch. It can

switch both positive and negative voltages. It has a very low ON resistance of 100 and has impedanceof $10^{12}\ \Omega$ in the OFF state. The analog switch has also been used to multiplex the dc and ac outputs to the sample and hold input.

3.3 Analog Multiplexer

In both current (low range) and resistance measurements 6- ranges are used to switch any one of the 6- ranges we have chosen an analog multioplexer which is a 1 of 8 selector. Anyy one of the 8 inputs can be chosen by using the select lines ABC. For eg. the 5th line can be selected by giving 1 0 1 to ABC . While the 0th line can be selected by giving 0 0 0 to ABC. The CD 4051, is a one of 8 selector. It can select positive as well as negative voltages (upto ± 5 v). The functional block diagram of 4051 is shown in fig. 3.1

3.4 Sample and Hold Amplifiers

The principal application of sample and hold amplifiers is to maintain an ADC's input of a certain precisely known time. The characteristics of the sample and hold amplifiers are crucial to the system accuracy and the reliability of digital data especially in 12 bit data (as is the case here) and high through put rate applications.

The SHA used here is the I.F 398. It consists of a capacitor, input and output buffer amplifiers and a switch and its drive circuitry. During sample the current is connected to promote rapid charging of the capacitance. During hold, the capacitor is disconnected from its charging source and - ideally - retains the stored charge. The functional diagram of LF 398 is shown in fig 3.2

The important parameters of a SHA that are to be considered is the aperture time and aperture uncertainity. Aperture time is the time required after the hold command for the switch to open fully. Aperture uncertainity is the range of variation in aperture time.

In order to get precise timing the hold command would have to be advanced by the aperture time.

3.5 RMS-DC Converter

A very important chip used in the instrument is the RMS -DC converter. The need for RMS - DC converter has been dealt with in the previous chapter under "AC measurements". The block diagram and Standard RMS connection of AD 536 is shown in fig - 3.3

The actual computation followed by the RMS-DC converter is as shown below.

$$V_{rms} = \text{Avg} \left[\frac{V_{in}^2}{V_{rms}}\right]$$

The above equation tells us that the RMS -DC converter performs the following operations

i. First it takes the absolute value of the signal

ii. It squares the signal and

iii. It divides the feed back output using the logarithmic chracteristics of the transistor junction.

iv. It filters the result.

The resulting O/P

$$V_{rms} = [Avg (V_{in}^{2})]^{1/2}$$

is valid if the averaging time constant is sufficiently long compared to the periods of the lowest frequency ac components of the signals. The simplest form of averaging is one using a single-pole filter by connecting an external capacitance. The value of capacitance is to be chosen so that it must arrive at optimum values of both ripple content and settling time as both are inversly proportional to each other.

3.6.1 Analog to Digital Converter

The ADC used here is the AD 574 A, which is of the successive approximation type.

The device consists of two chips, one containing a precision DAC with precision voltage reference , the other certaining the comparator , successive approximation register, clock, output buffers and control circuitry, when the control section is commanded to initiate a conversion (through the R/$\overline{C}$), it enables the clock and resets all bits of the Successive Approximation Register (SAR) to zero's. Once a conversion cycle starts, it cannot be stopped and data is not available from the output buffers.The SAR, timed by the clock will then sequence through the conversion and return and end of conversion flag to the control section. The control section will then disable the clock, bring the output status flag low and enable control functions to alow data read functions to allow data read functions by external command.

During the conversion cycle the internal 12-bit current output DAC is sequenced by the SAR from the MSB to the LSB to provide an output current which accurately balances the input signal current through the 10 KΩ resistor. The comparator weighs whether the addition of each successively - weighed bit current carses the DAC current sum is less the lift bit is turned on; if more the left bit is turned of :

After testing all the bits the, SAR contains a 12 - bit code which accurately represent the input signal within ± 1/2 LSB.

3.6.2 Bipolar Operation

The connection for bipolar operations are shown in Fig. 3.4a.The range of analog voltage at the input of the ADC can be from ± 10 Volts.

3.6.3. Stand Alone Operation

Since dedicated input ports are avilable, the Ad 574 can be used in the "Stand Alone" mode. Thus full bus interface capacity is not required.

In this mode AO & $\overline{CS}$ are wired low,whereas CE & 12 /$\overline{8}$ are wired high To start the conversion, a low pulse is sent to the R /$\overline{C}$ pin. This pin should be low for a minimum of 305 ns. The stand alone mode timing diagrams are shown in Fig. respectivelly.

3.7 INTERFACING TO PC

The voltage section is interfaced to the PC through the I/O card from the ADC. The frequence section is interfaced directly to the I/O card from the resepective outputs

of measuring circuitries. The I/O card is an adapter card which can directly interface to the PC Bus. The I/O card consists of two 8255's (PPI) and a timer 8253. All I/O operations can be done through the 8255 and timer operation can be done through 8253. The 8255 is operated in mode 0 (simple I/O). All inputs are buffered and all outputs are latched. The bit patterns allocated to the different ports, ie. PA1, PA2, PA3, PA2, PB2, PC2 are shown in detail in Fig. 8.18. There is a specific bit called $\overline{V}/F$ bit. When it is zero the voltage buffers (in the form of 74LS 244- 1 and 74LS 244-2) are enabled. When the $\overline{V}/F$=1the frequency buffers are enabled. Not all the control lines are buffered. Some are directly routed to the control hardware in the instrument. The configuration of ports in the 8255-1 are as follws.

(1) PA - INPUT PORT

INTERFACED TO LSB OF ADC

(2) PB - OUTPUT PORT (CONTROL)

(3) PCU - CONTROL

(4) PCL - INPUT PORT

- INTERFACED TO 4 UPPER BITS OF ADC

The configuration of 8255 -2 is as follows.

(i) PA - OUTPUT PORT

(ii) PB - OUTPUT PORT

(iii) PC UPPER - OUTPUT PORT

(iv) PC LOWER - INPUT PORT.

The block diagram of the I/O card is shown in Fig. 8.19

3.8 ABOUT SM ART WORK

The PCB layout for the circuit was done in the CAD package, Sm ART work. The PCB is a double-sided one. The component and solder side of the PCB have been attached in the report.

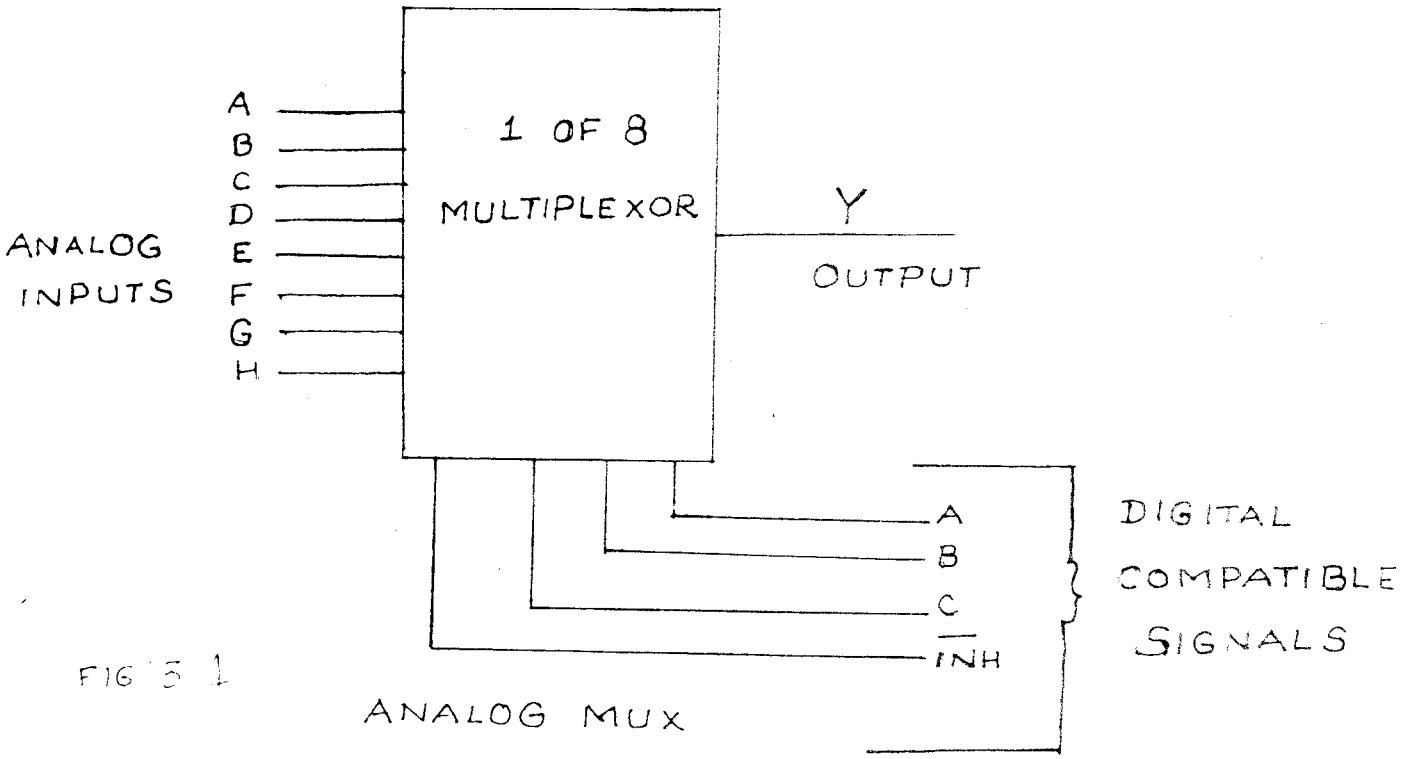


FIG 3 1

ANALOG MUX

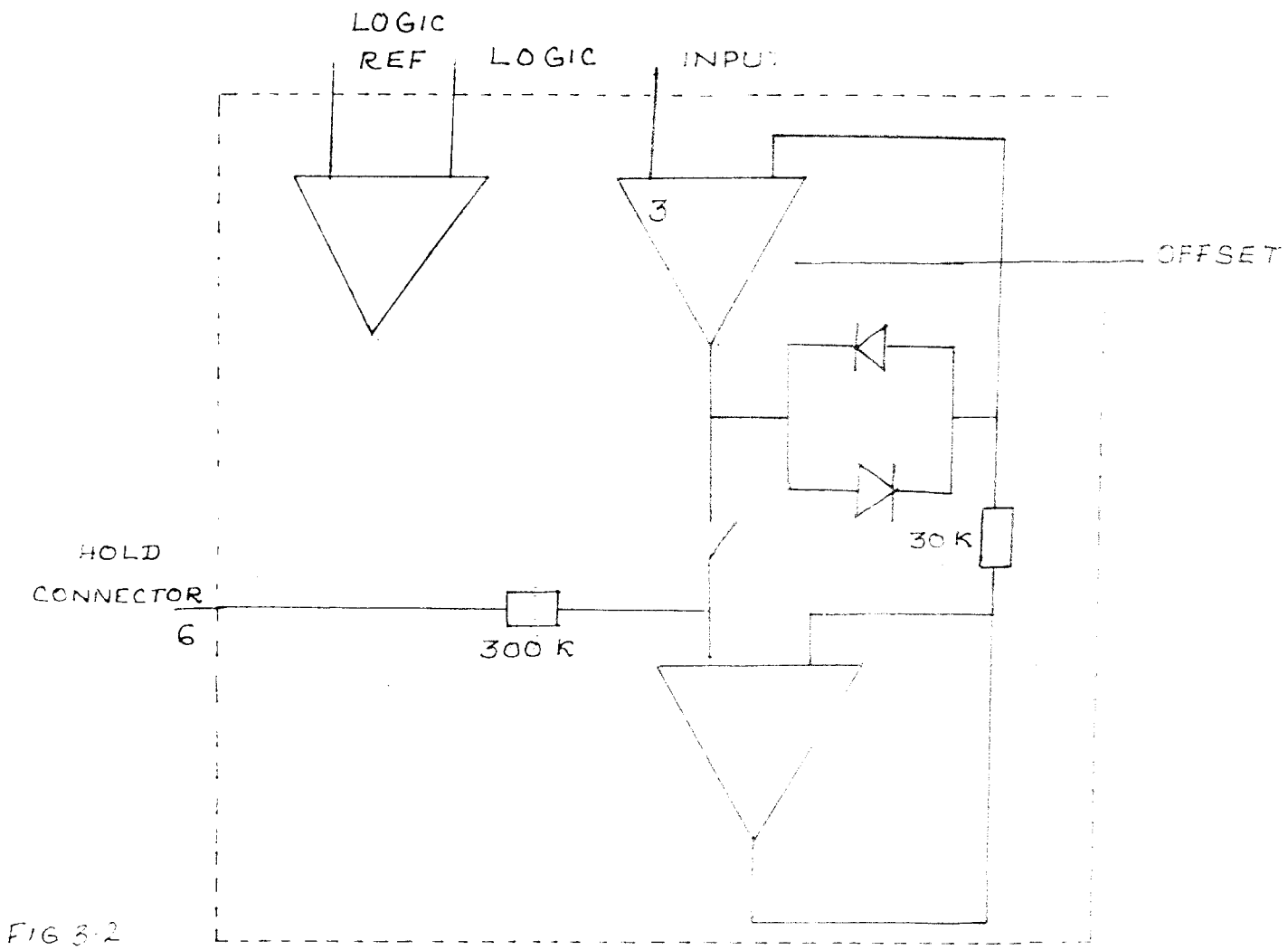


FIG 3-2

SAMPLE AND HOLD AMP BLOCK DIAGRAM

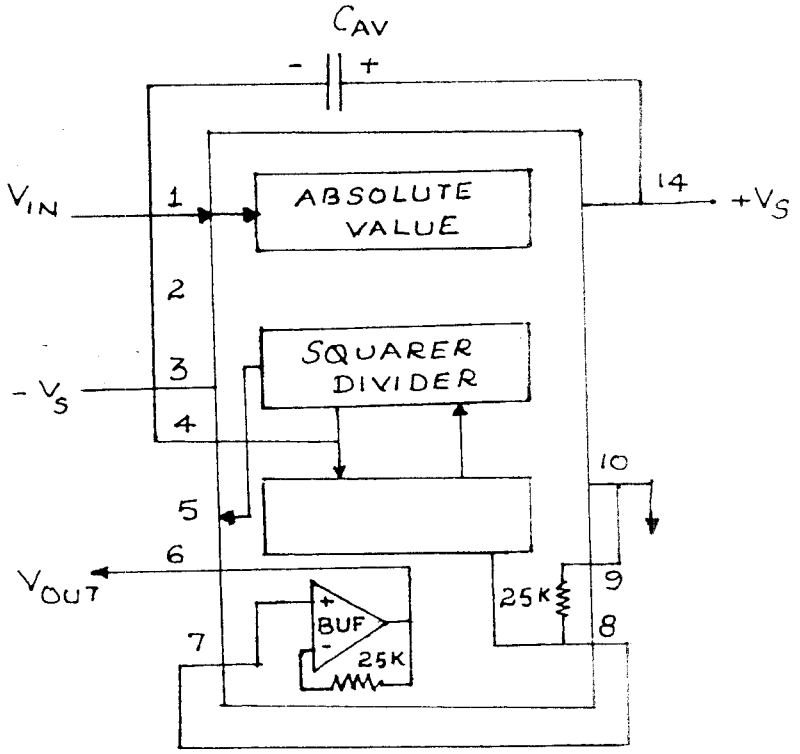


FIG. 3.3 BLOCK DIAGRAM AND STANDARD RMS CONNECTION OF AD 536 A

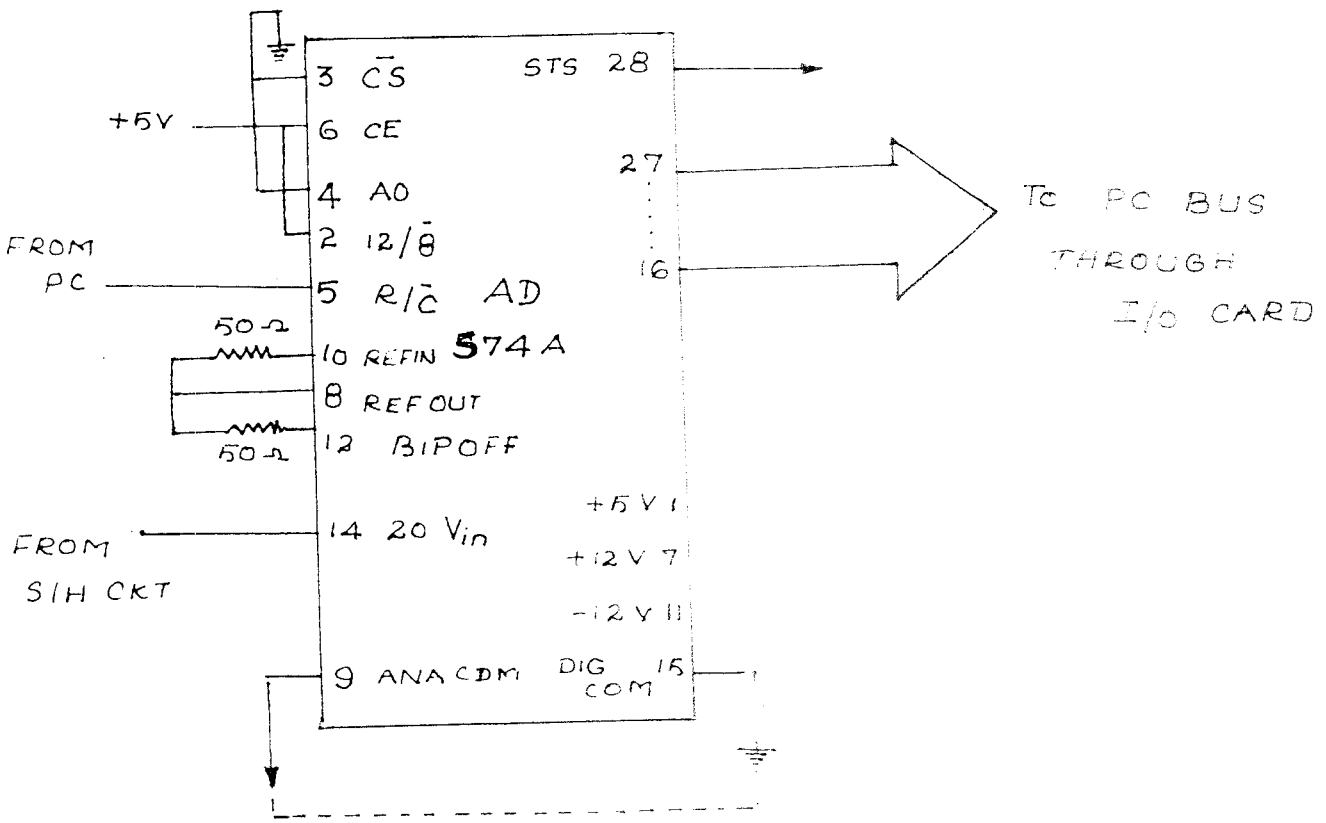


FIG 3.4a BIPOLAR CONNECTIONS FOR AD 574

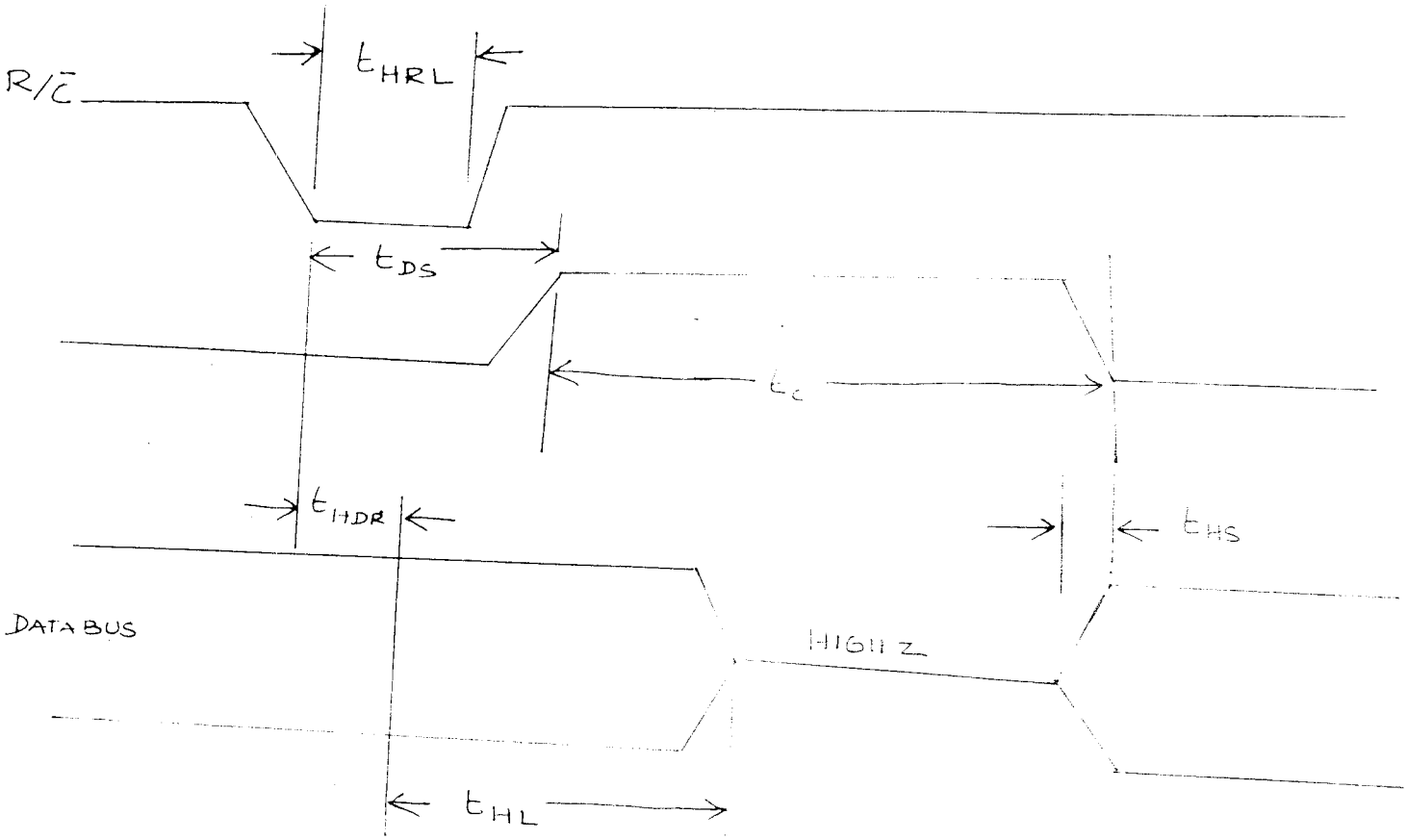
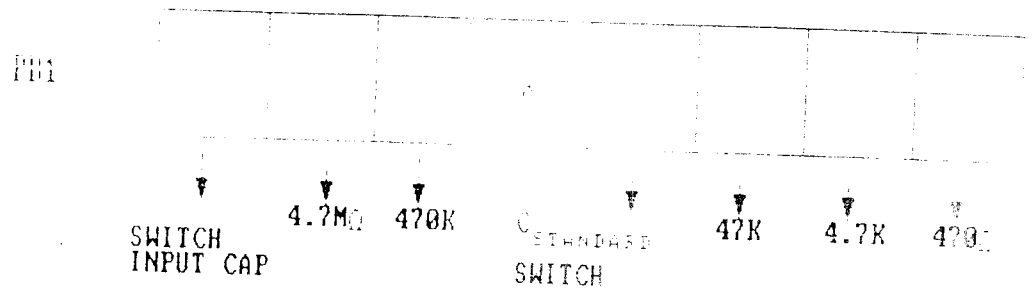
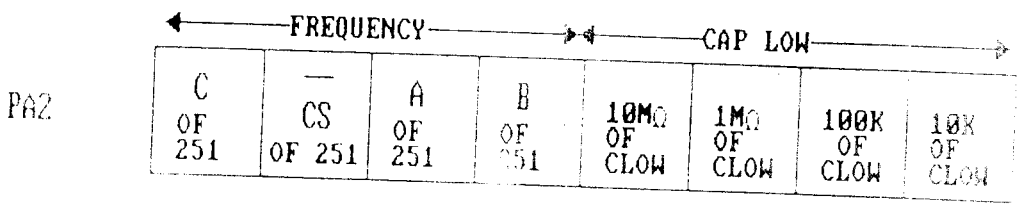


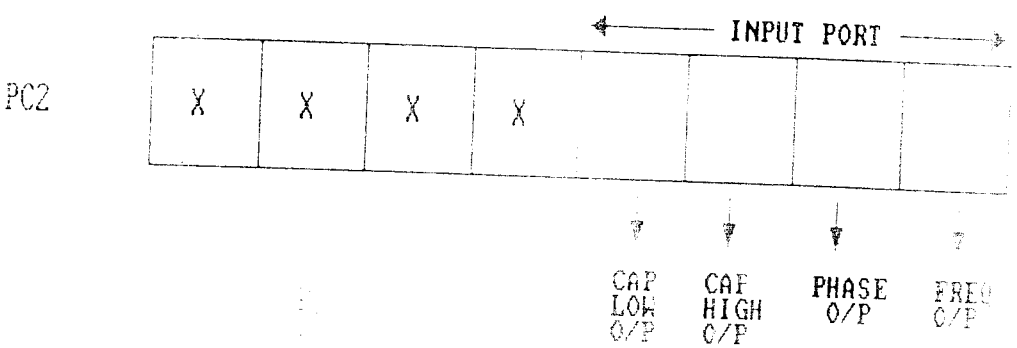
FIG 3.4.b "STAND ALONE" TIMING DIAGRAM FOR ADC 574

SYMBOL	PARAMETER	MIN	TYP	MAX	UNITS
t_{HRL}	LOW R/C PULSE WIDTH	350	-	-	nS
t_{DB}	STS DELAY FROM R/C	-	-	500	nS
t_{HDR}	DATA VALID AFTER R/C LOW	25	-	-	nS
t_{HL}	OUTPUT FLOAT DELAY	-	110	150	nS
t_{HS}	STS DELAY AFTER DVALID	300	-	1000	nS
t_C	CONVERSION TIME (12 BIT CYCLE)	15	-	35	μS

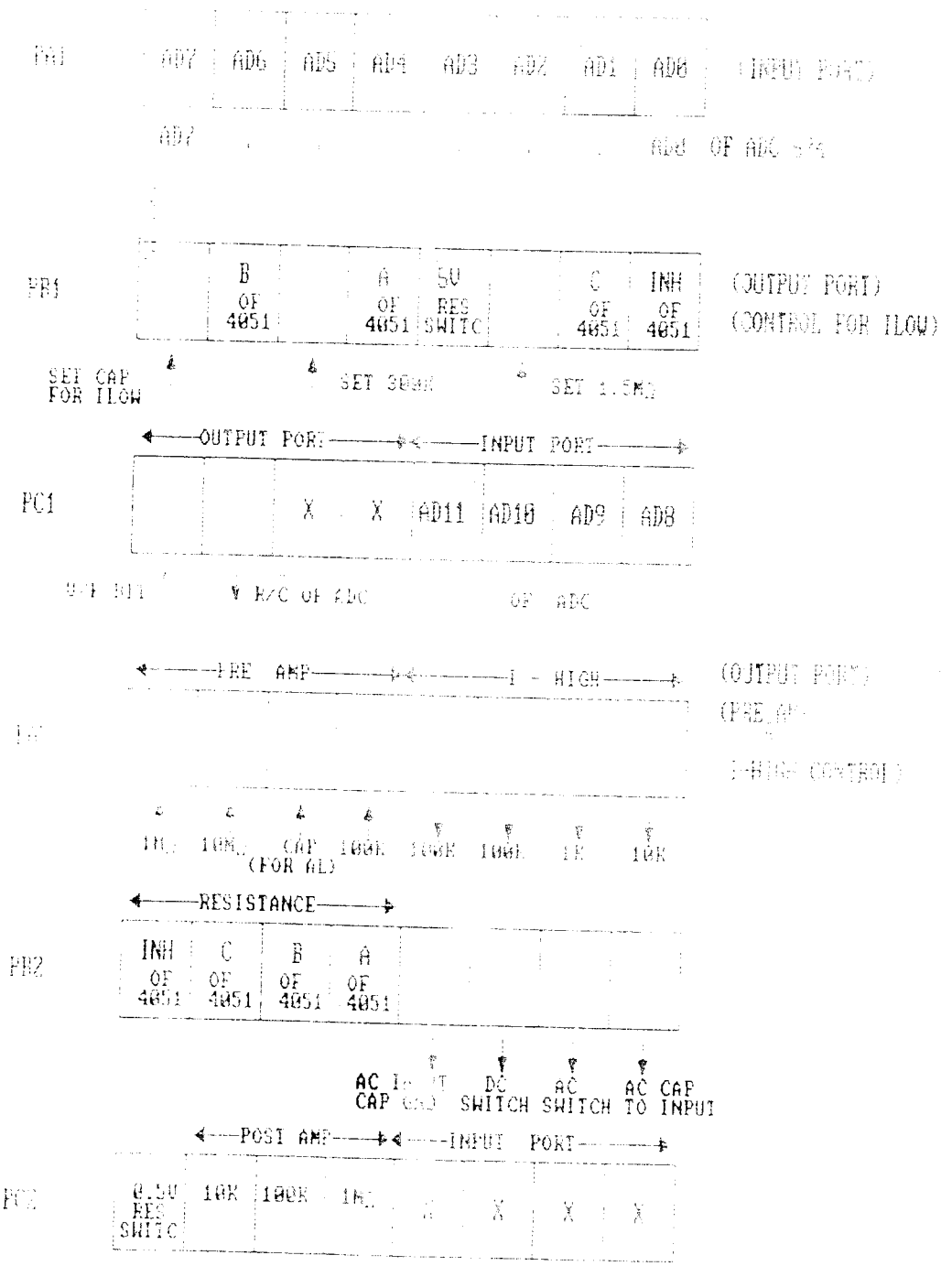
IF O/F = 1







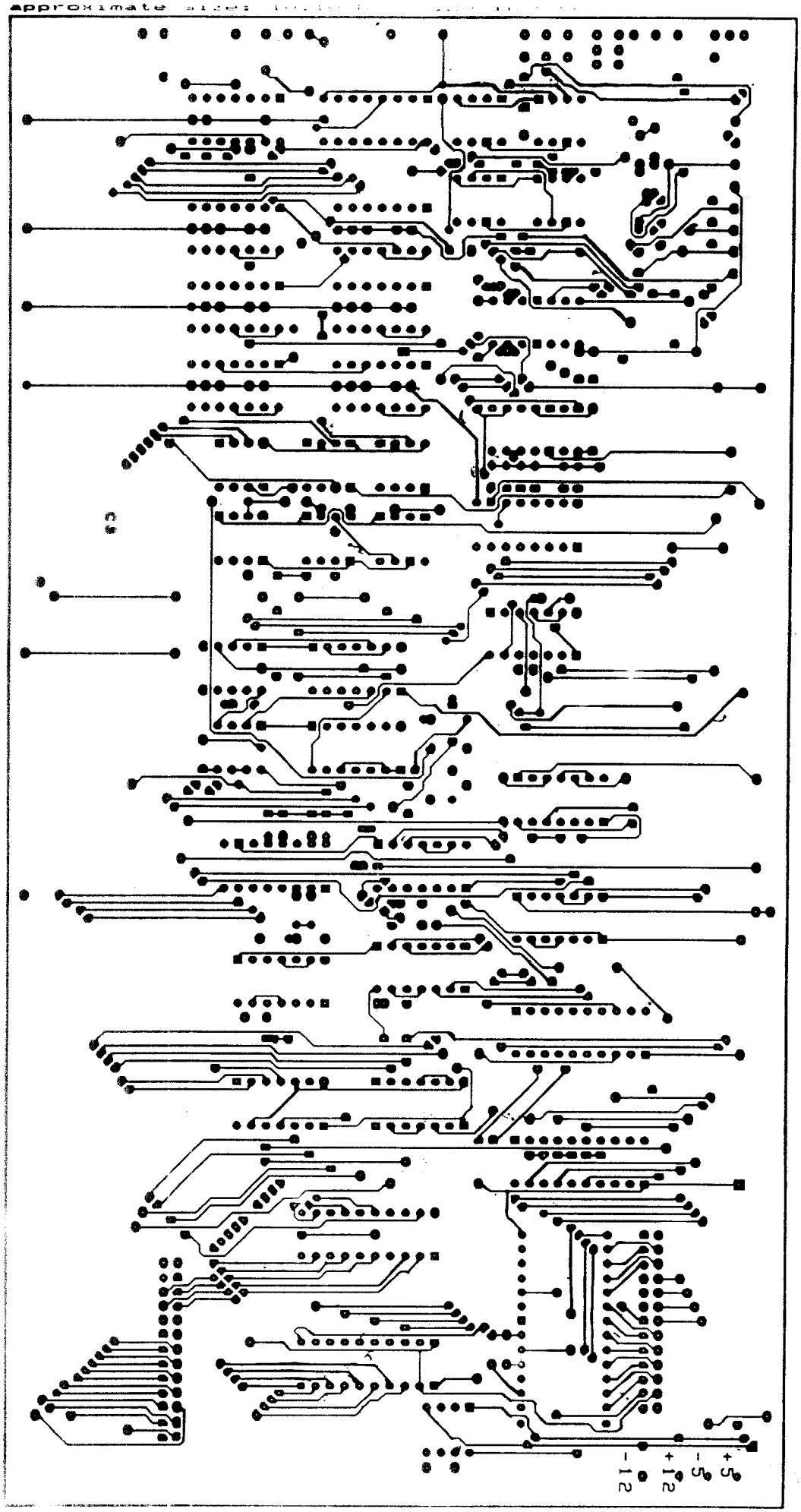
CONT... BIT ALLOCATION TO PO



BIT ALLOCATION TO PORTS

CONT...

maha
v1.2 r2 holes: 11
approximate size:



Decoupling grounding and Power system layout

CHAPTER -IV

DECOUPLING , GROUNDING AND POWER SUPPLY LAYOUT

4.1 NEED FOR GROUNDING

For any electronic system to work property, it is necessary that the system be properly grounded . Improper grounding gives rise to a variety of problems such as ground loop interference , frequency instability (at high frequencies) and can hence degrade the performance of the system. For eg.: consider fig. 4-1 . The gain of the circuit is ideally expected to be $-V_{in}\ R_2/R_1$. But , since the ground has an inpendance of Z the actual voltage O/P . V_o is = $-V_{in}\ R_2/R_1 + V_z(1+ R_2/R_1)$ where V_z is the ground impendence as small as possible.

4.2 EFFECTIVE GROUNDING

One important key to effective grounding is to have a careful look at where currents flow. Allowing these currents to share a path with a low level signal may result in trouble. An example of improper grounding is shown in fig. 4.2 1 m a. The amplifier drives a load resistor RL . If the resistance between point A&B is around 2 milli ohms (if the length is around 15 cm). This value of 8 m is normal , and R_L = 2 k Ω, a 10 volt. Output signal will result in about 40 μ volts between pointsmarked. V . This signal acts in series with the non-inverting input and can result in significant errors. For eg. the typical gain of amplifiers is in the order of 10^6 (open loop)

so that only a 10 μ volt signal is required to produce a 10 volt signal O/P. With this noise value of 40 μ volt, the noise gain error will increase upto 4 times. This degradation will affect circuit performance considerably, especially in measurement (precision) applica tion. Care has been taken to see that such problems do not occur in the test instrument, as far as possible. Another possible problem with the error "ΔV" is that, it might cause oscillations due to positive feed back for large closed lopp gains with R f / R i greater than 250 k.

The recommended scheme of gounding to avoid the sort of trouble mentioned above is shown in Fig. 4-3 This connection will hold good for cascaded loads also, as shown.

4.3 DECOUPLING

4.3.1 Negative Supply Decoupling

With most analog integrated circuits, the decoupling problem is related to the negative supply terminal. It is found that the negative supply resection will approach zero for signals above the closed loop band width. This means that high speed high level circuits can"talk to" low level circuits through the common impedance of the negative supply line. This problem is related to the output stages of the integrated circuits. It may be remembered that thirty to forty centimetres of wire can act like a high Q inductor to add to the normally over dampled supply response. A decoupling capacitor may not always cure the problem. If the decoupling currents flow elsewhere, it may still produce an undesirable glitch.

Fig .4.4 illustrates a possible configuration for 80 negative supply decoupling. This configuration minimizes disturbances of the V and ground busses. The high frequency component of the load current is confined to a loop which doesn't include any part of the ground path. If the capacitor is of sufficient size and quality, it will minimise the glitch on the negative without disturbing input or output signal paths.

4.3.2 Positive Supply Decoupling

The techniques that apply to negative supply decoupling apply equally well to positive supply decoupling, except that the reference should be the positive supply .
Positive supply decoupling is important for circuits that have a compensating Integrator referred to the positive supply. Otherwise it is not of much practical importance.

4.4 DECOUPLING AND GROUNDING USED HERE

4.4.1. Buffer Decoupling

The buffer used here is 74 LS 244. The decoupling is done by a 0.1 μ F disc type ceramic capacitor. This 0.1 μ F cap is quite effective here, since the frequency of data dealt is relatively small.

Power supply lay out for all IC 's have been done so that V_{cc}, Gnd and V_{EE} lines run inbetween the IC's and also s that they are all parallel. All power lines are situated in the same plane. This method is found to reduce the effects of noise due to irregular layout of power supply to a large extent.

4.4.2 Decoupling and Grounding for the RMS-DC Convertor

Both Vcc and V_{EE} have been bypassed to ground to 0.1 μ F ceramic discs so that any high frequency ripple in the power supply may not affect the RMS-DC converter perfor mance. These capacitors are placed closed to the device to make the decoupling effective.

4.4.3 Supply Decoupling and Layout for the ADC

It is critically important that the AD 574 power supplies be well regulated filtered and free from high frequency noise use of noisy supplies will cause unstable output codes to be generated. A few millivolts of noise represents several count error in the 574.

Decoupling capacitors have been used on all power supply pins . A decoupling capacitor of 47 μ F (electrolytic) ; 25 V has been used between Vcc and V_{EE} in paralliel with a 0.1 μ F disc ceramic type. Also the +5 volt. line has been decoupled to ground through a 0.1 μ F disc ceramic capacitor.

The associated analog input circuitry and inter connections have been placed a litter bit far from the ADC logic circuitry. This has found to enhance the ADC performance.

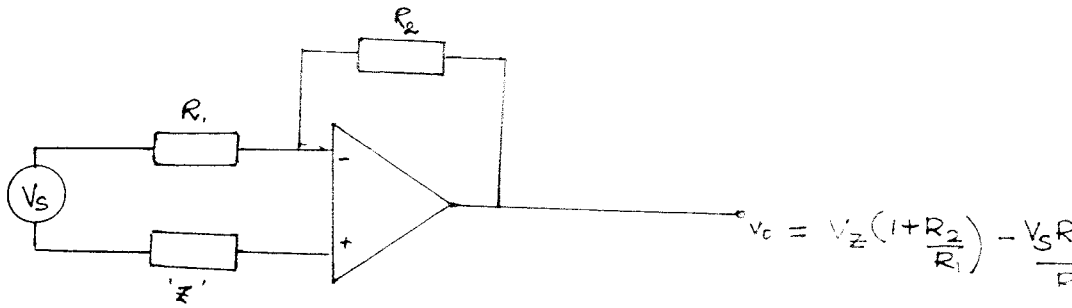


FIG 4.1 GROUND LOOP INTERFERENCE

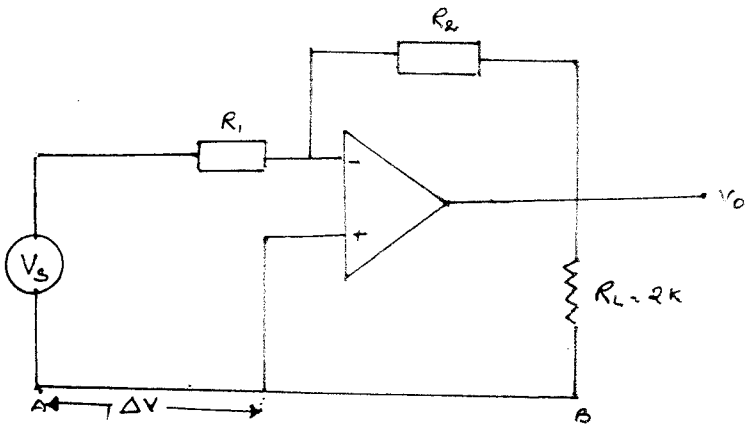


FIG 4.2 THE 'ΔV" PROBLEM

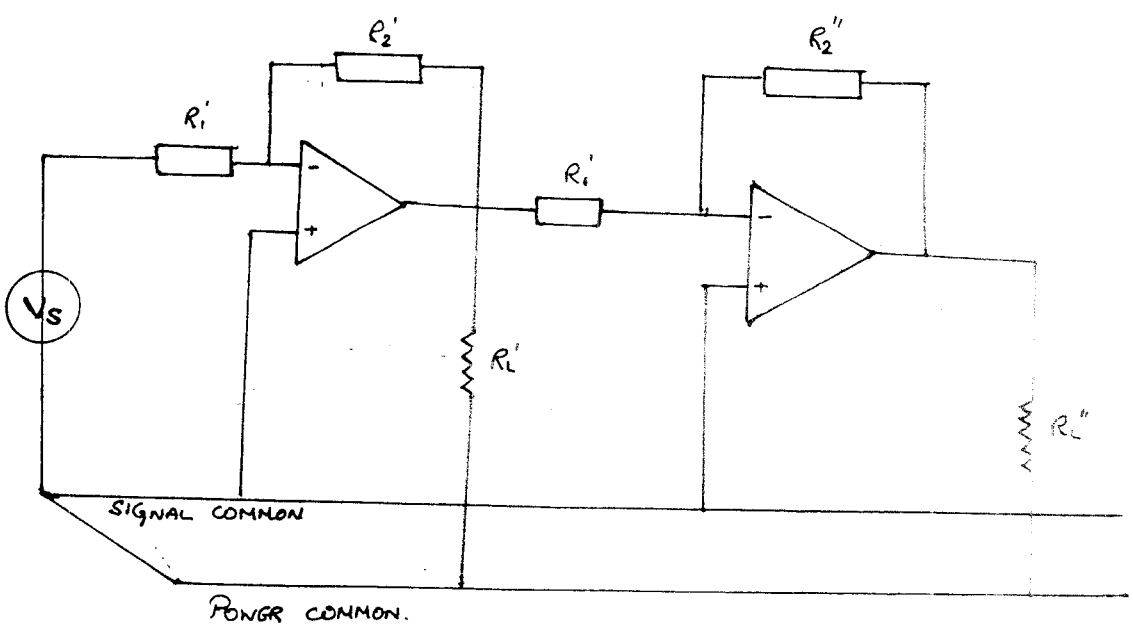


FIG 4.3 SOLUTION TO ABOVE PROBLEM

CHAPTER -V

SOFTWARE

5.1 INTRODUCTION

The software to the project has been written in QUICK BASIC , a fast version of Microsoft's BASIC and in assembly language (8088 assembly).

The purpose of the software is as follows.

(i) To switch ranges and hence made the gain of an OPamp to be programmable.

(ii) To issue convert pulse to the ADC and read the converted data.

(iii) To log data, that has been previously measured.

(iv) To input data from ports and calculate the required results.

(v) To display the result.

QUICK BASIC AND REASONS FOR USING IT

1. It offers a built in, full screen editor, compiler and debugger with pull down menus, dialog boxes and on-line help.
2. It can compile in memory entirely and hence save time.
3. Its editor locates error and highlights them with the cursor.
4. Program can be compiled to optimise speed or efficiency.
5. It supports structured programming with alpha numeric labels, structured logic statements, sub programs and multi line functions , which makes programs easier to read and understand.

6. It supports graphics, BLOAD/BSAVE and event trapping.

7. It has a wide range of debugging facilities.

8 .Dynamic arrays allows us to allocate memory for the array at run time, providing more efficient use of memory.

The above advantages of Quick Basic make it easy for us to program and use it efficiently.

5.3 ASSEMBLY LANGUAGE

The reasons for choosing assembly language are comparatively obvious.

1. It is the fastest and hence can enhance the throughout of the instrument.

2. Functions like bit fiddling, rotating bits are most effectively implemented usin ; assembly language.

3. It can communicate with I/O ports at a very fast rate.

4. The use of MASM (Macro Assembler) and its range of pseudoops, directives have made

it programming in assembly easier and enjoyable.

5. The assembly language acts as the interface to actual hardware and the software.

6. The assembly language provides a cornucopia of interrupts for the user to handle keyboard , video, printer and other devices easily.

5.4 INTERFACING Q BASIC AND ASSEMBLY PROCEDURE

The following are the steps to Interface Q Basic and the Assembler.

1. On exit, the called routine must restore the contents of the BP, SS and DS registers to their initial values.

2. All data declared and referred in the routine should be in the segment DATA and in the group D Group.
3. The called routine must know the number and type of arguments passed. References to the arguments are positive offsets from BP (assuming that the called routine moved the current SP to BP)
4. The called routine must do a far RET n to adjust the stack to the start of the calling sequence.
5. To return values to BASIC , include the variable name that will receive the result in the argument list.

5.5 DC VOLTAGE MEASUREMENT FLOW CHART

The DC voltage measurement flow chart is shown in Fig. 5.1. As shown the gain of the two stages is increased, if the voltage measured lies with -1 and +1 volt. Otherwise, the value is returned to the calling program.

5.6 AC VOLTAGE MEASUREMENT FLOW CHARTS

Here the gain is first set at lowest value. The value is passed to the calling program if the RMS value lies 0.5 and 5 volt. Otherwise the range is varied till the correct value is got by suitably switching the amplifiers. The flowchart is shown in fig. 5.2.

5.7 CURRENT MEASUREMENT FLOW CHART

The current is first converted to a proportional voltage. This voltage is available at the output of the ADC. This value is read and compared. If it is within 0.6 V to 6 volts.

the value is passed, otherwise the next higher gain is chosen by switching the next range. This process has been explained in Fig. 5.3

5.8 RESISTANCE MEASUREMENT FLOW CHART

A reference voltage drives the unknown resistance. The gain depends on the value of the unknown resistance. Two references have been used to increase the range. If the voltage is within 0.5 and 5 volt, it is passed to the calling routine, otherwise the next range is chosen and the value is passed only if the above criterion met. The flow chart is shown in Fig. 5.4

5.9. CAPACITANCE MEASUREMENT FLOW CHART

Here the range is varied till the pulse width produced by the monostable is greater than 10 ms. If it is greater than 10ms the value is measured otherwise the next higher range is chosen and this increases the pulse width. This logic is shown in Fig. 5.5

5.10 FREQUENCY MEASUREMENT FLOW CHART

Here the range ($\frac{a}{b}$10) is varied till the pulse width is greater than 1 ms. If it is the value is measured, otherwise the value of frequency is multiplied and measured till, the proper condition is net. The procedure is shown in Fig 5.6

5.11 PHASE MEASUREMENT FLOW CHART

Here too, the pulse width is measured; First as shown the processor waits for zero cross. Then by measuring on T_{OFF} the phase difference can be measured as shown. The logic is shown in Fig. 5.7

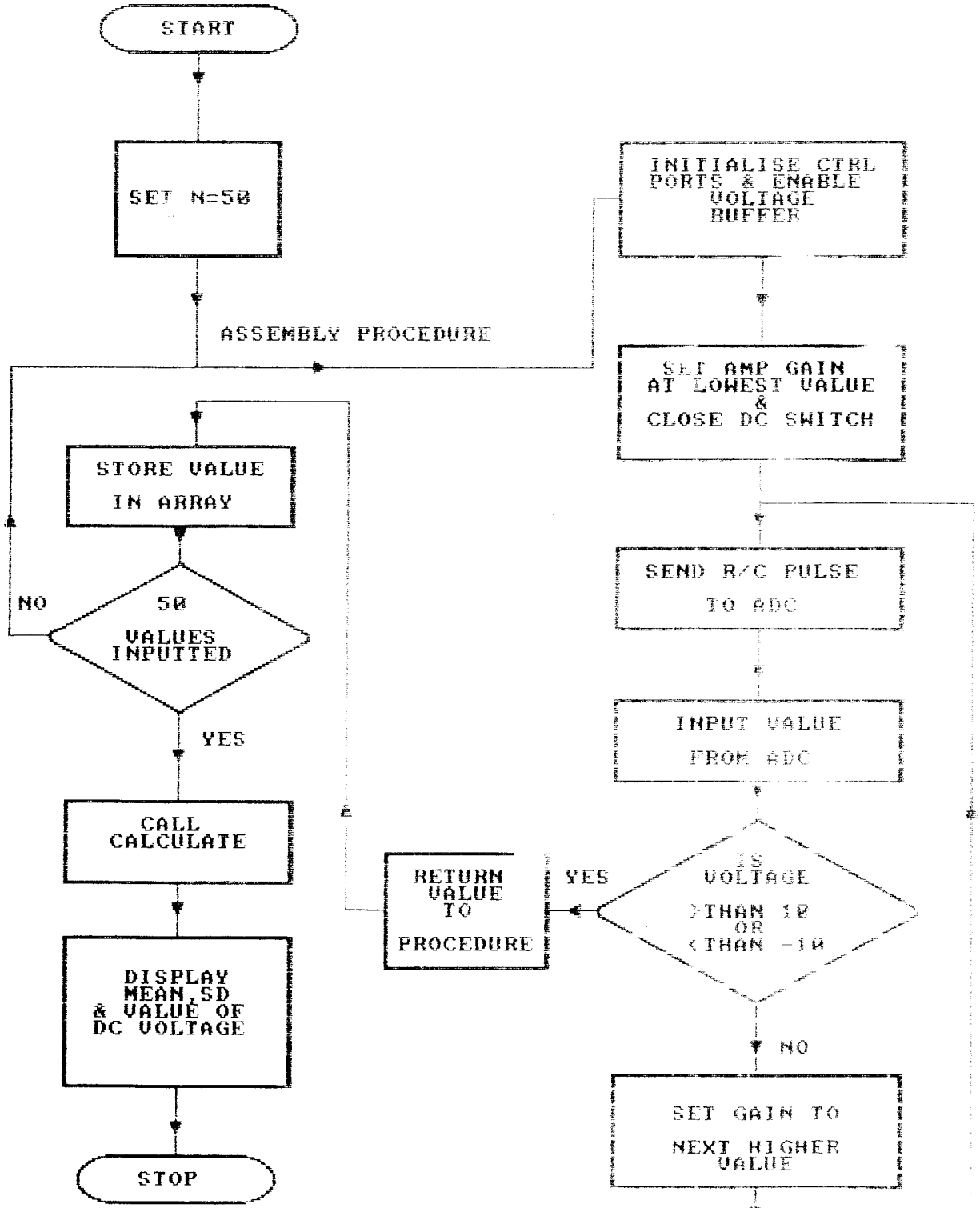


FIG 5.1 D.C. VOLTAGE MEASUREMENT

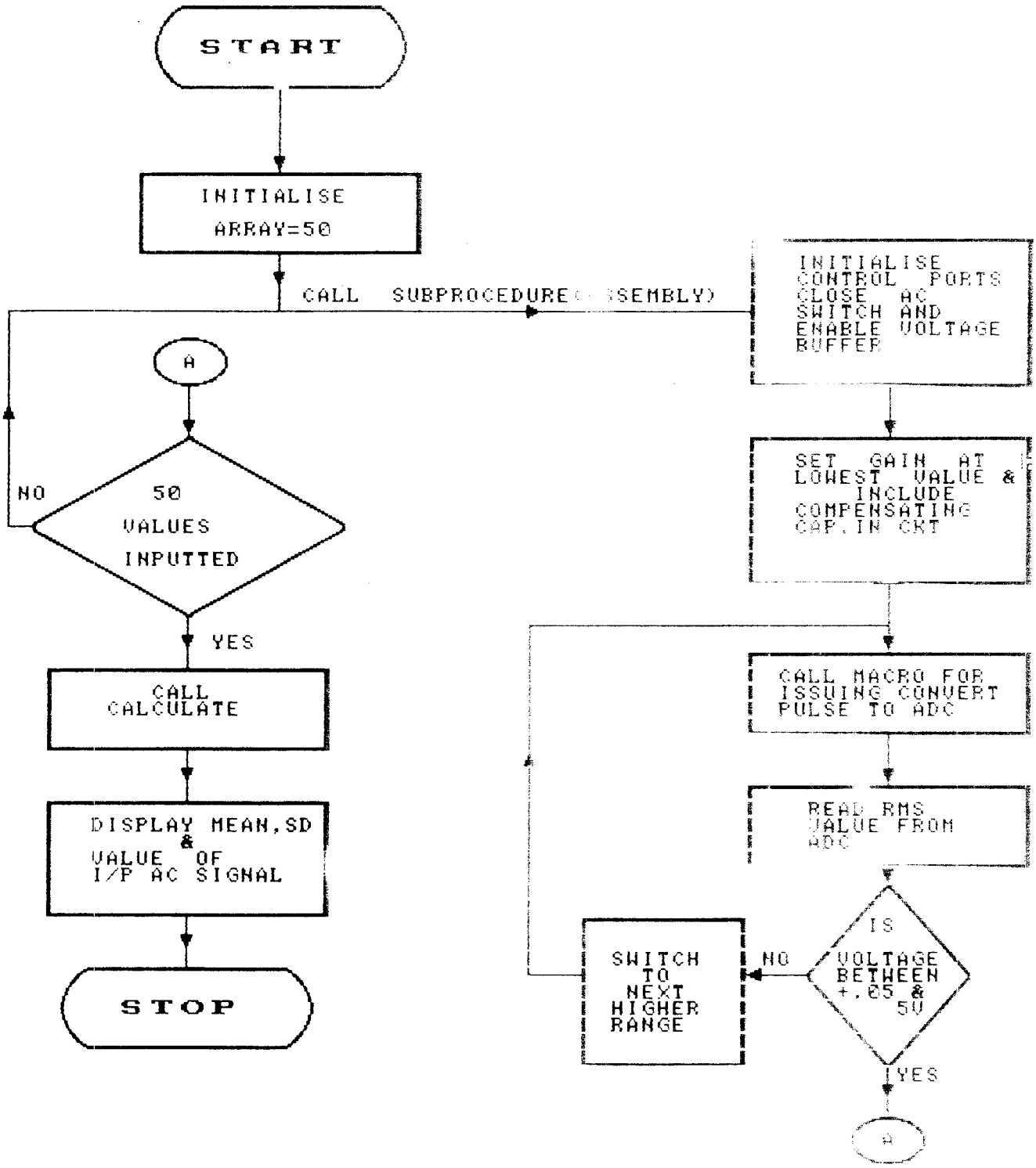


FIG 5.2 A.C. VOLTAGE MEASUREMENT

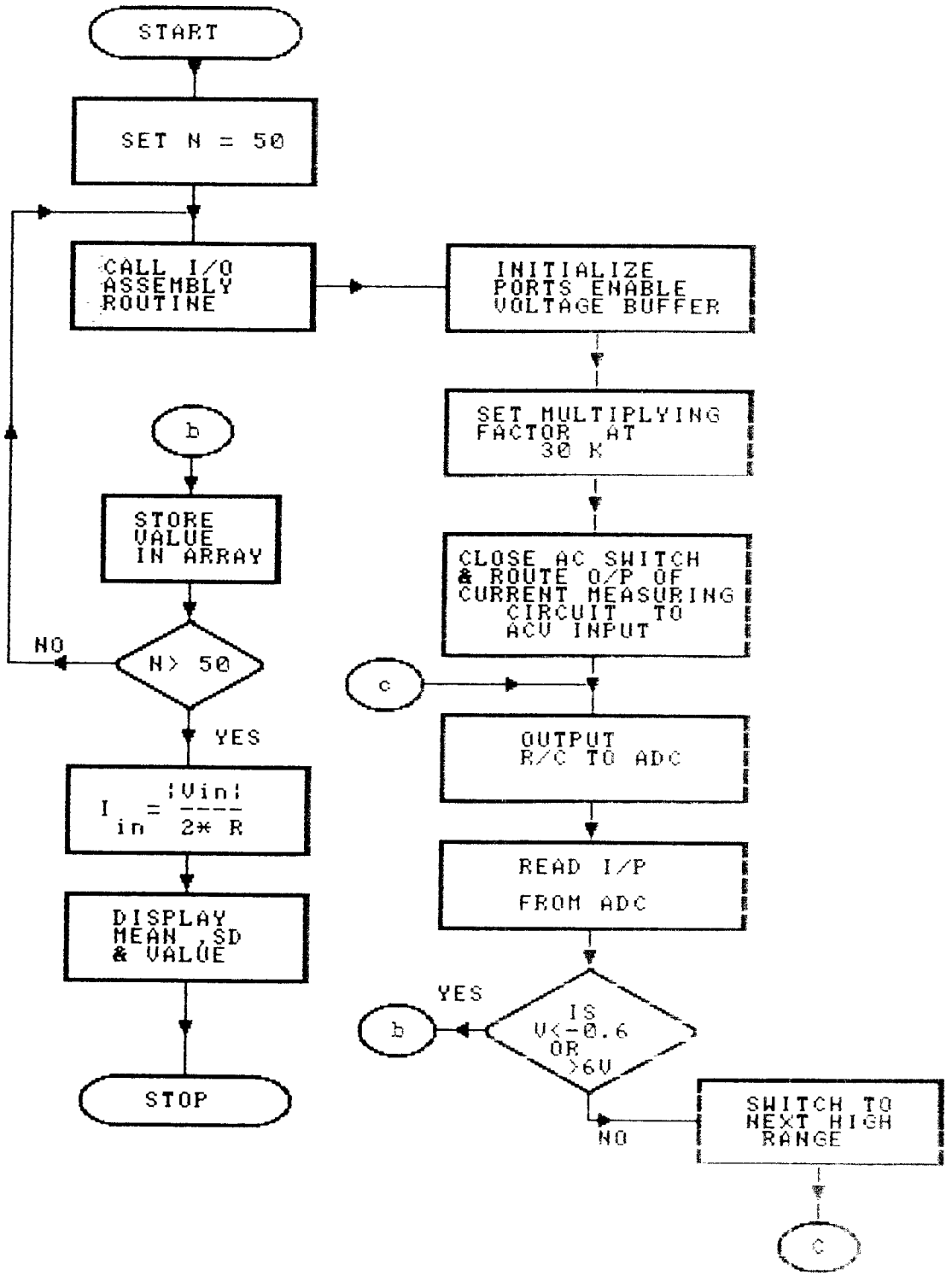


FIG 5.3 CURRENT (LOW & HIGH)

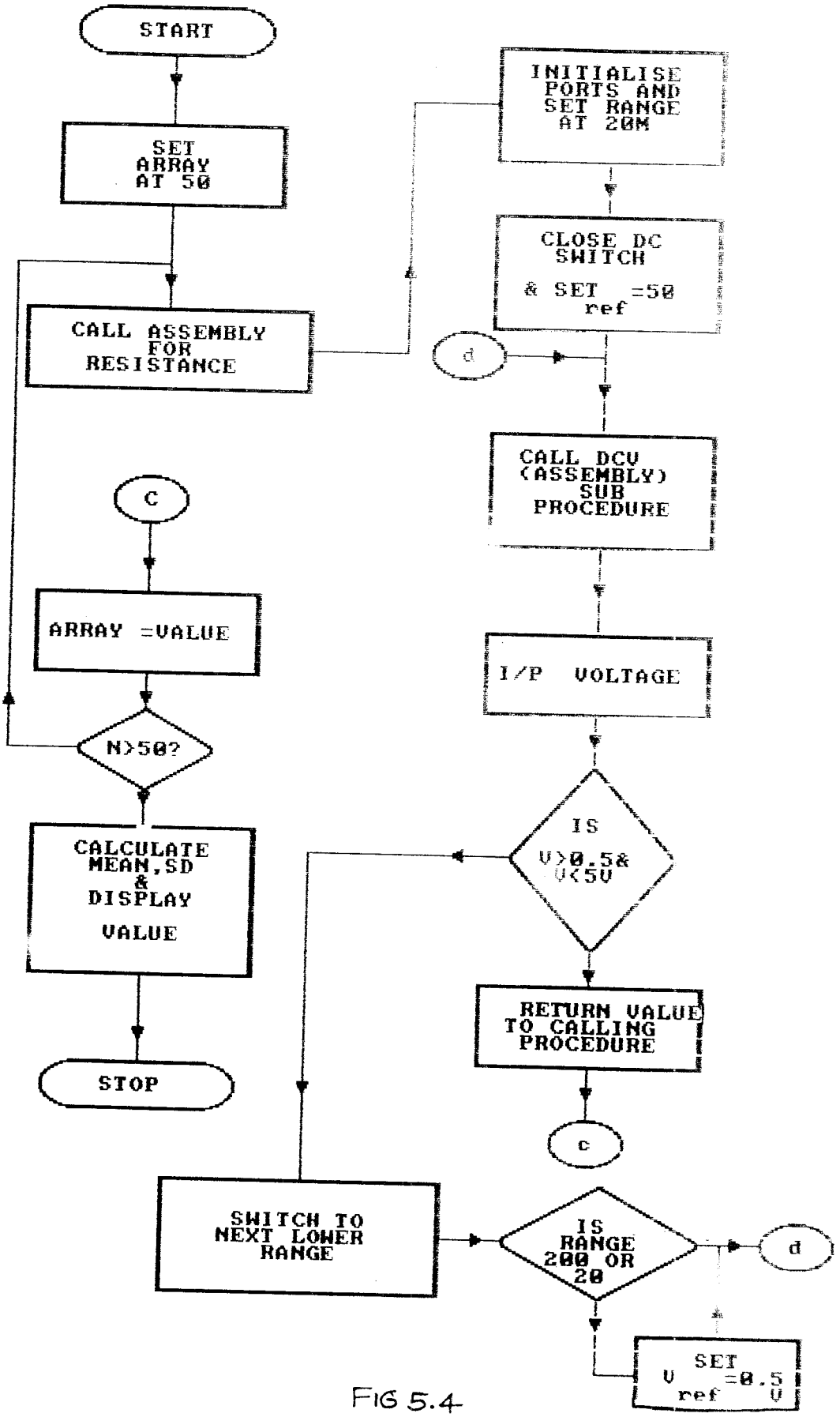


FIG 5.4
RESISTANCE MEASUREMENT

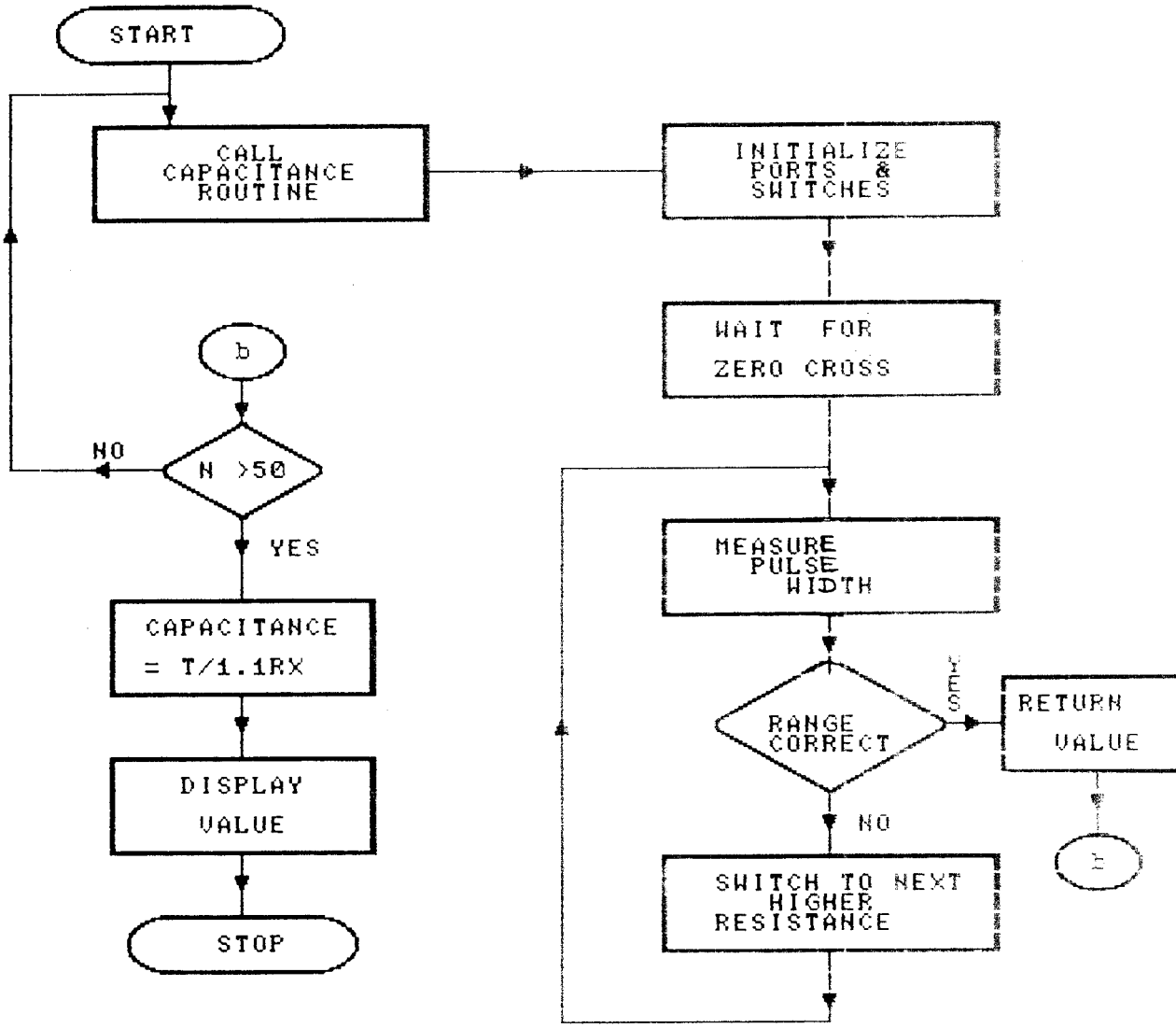


FIG 5.5

CAPACITANCE (LOW & HIGH)

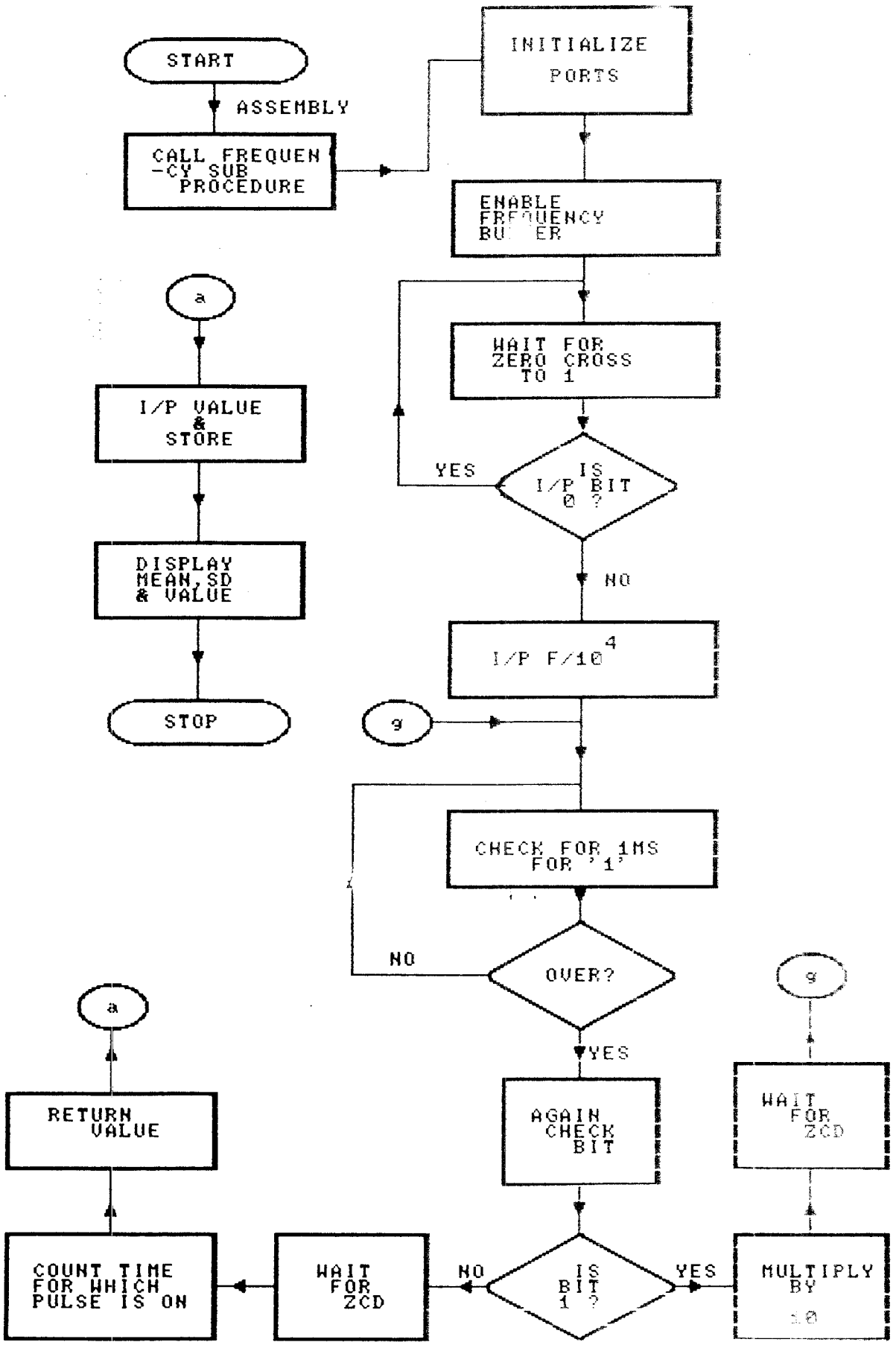


FIG 5.6 FREQUENCY MEASUREMEN

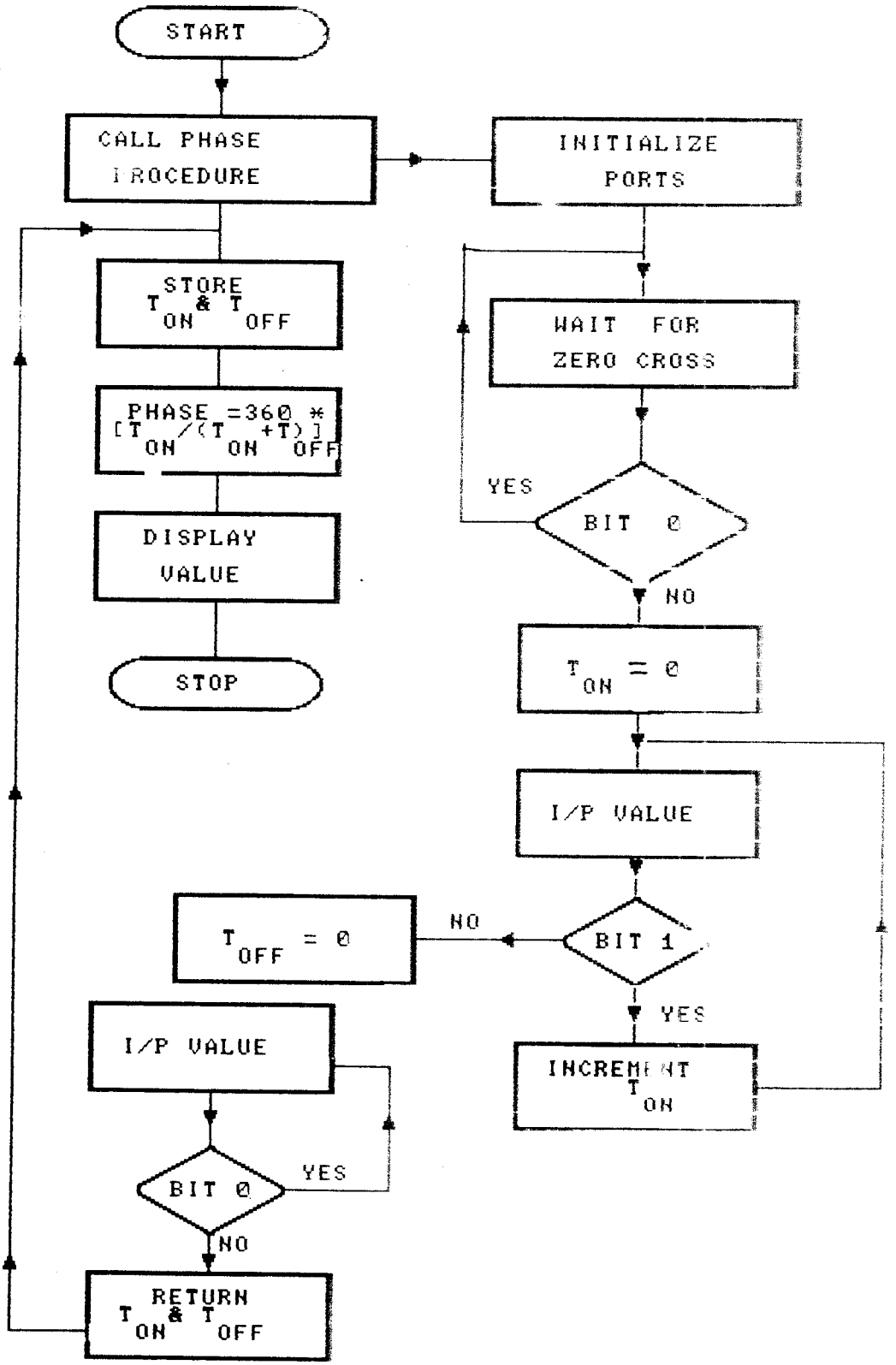


FIG 5.7 PHASE MEASUREMENT

Testing and debugging

CHAPTER -VI

TESTING AND DEBUGGING

The complete circuit, before being drafted on the PCB was tested on the bread board. The accuracy we could obtain was around 1%. This fairly large amount of error was due to a variety of noise problems that creep into wire-wrapped circuits. The circuit was mounted on the PCB and test results showed that the error decreased. This was probably due to better grounding and decoupling provided on the printed circuit board.

The complete software was tested and debugged using break point and single step schemes of debugging. The errors present were suitably removed. The initial problems in interfacing QBASIC and Assembly language was overcome. The debugging tools of QBASIC and CODE VIEW of Macro Assembler (Ver 5.0) were highly useful for debugging.

Conclusion

CHAPTER VII

CONCLUSION

A multifunction test instrument with 0.1% accuracy has been developed and tested successfully .The accuracy can be further improved to 0.01% by a more sophisticated design which has fast and precise Programmable gain amplifiers. These were not used here due to the high cost of these chips.

A logical extension of the instrument would be to incorporate more meaning functions like the inductance and capacitance content in a resistor etc. It can be made into a data acquistion system by transducing the required inputs suitably. In that case one has to go in for a 1 of 16 multiplexor.

References

REFERENCES

1. STEVER ET AL, "seven function systems multimeter offers extended reso lution and scanner capabilities", Hewlett-Packard, Journal Feb.1986.

2. TRAUTNAM & DESJARDIN, "A portable, low cost, High-Performance Digital Multimeter for the HP-IL", Hewlet-Packard, Journal Feb. 1983.

3. JONES ET AL,"Precision DVM has wide dynamic range & High system speed", Hewlet-Packard,Journal April 81.

4. "Digital Multimeter" 3466A Operating and Service Manual, Hewleft-Packard.

5.SERGIO FRANCO, "Analysis & Design of opamps & analog IC'S", Mc-Graw Hill, 1988

6. BOUENS, "Digital Instrumentation", Mc-Graw Hill.

7. D.V. HALL,"Microprocessors and interfacing", Mc-Graw Hill.

8. XT Technical Reference , IBM.

9. TERRY GODFREY, "IBM Microcomputer assembly language", Mc-Graw Hill, 1989.

10. PETER NORTON,"Assembly language programming for the IBM PC".

Appendix

```
            ;This is an assembly language routine which will be called by
            ;QBASIC for dealing with I/O ports & also to do some  bit
            ;fiddling .There are totally 9 far procedures and the calling
            ;language may call any of the routines.The data sent to ports
            ;are done by first dumping the data in an array.Then data can
            ;be obtained by addressing the array.

;_____________________________________________________________________
____


 ;          SOFWARE FOR PCBASED MULTIFUNCTION TEST INSTRUMENT
  ;         ------------------------------------------------
   ;        PROJECTWORK DONE BY
    ;       E.ANAND                 89EEE01
     ;      S.GANESH                89EEE09
      ;     D.KARTHIKEYAN           89EEE13

;GUIDED BY
;           Dr.K.A.PALANISWAMY
;_____________________________________________________________________
____


        data segment word public 'data'

          porta   dw      01f0h,01f8h
          portb   dw      01f1h,01f9h
          portc   dw      01f2h,01fah
          portw   dw      01f3h,01fbh
          worda   db      10h,80h,40h,30h,0a0h,60h,43h
                  db      4ch,49h
                  db      48h,44h,42h,41h,20h,00h,90h
          wordb   db      84h,01h,83h,8ah,09h,14h,24h
                  db      44h,54h,64h,74h,20h,60h,02h
                  db      12h,22h,0f2h,0a0h,80h,60h
                  db      48h,10h,0c0h
          wordc   db      40h,20h,10h,0c0h
          wordw   db      81h,91h


          data    ends

          ;all data  declared and refrenced are
          ;in the segment data and in the group dgroup

          dgroup     group     data

          ;code segment begins here
          code_1 segment public 'code'

                   assume cs:code_1,ds:dgroup

          ;declare all data as global
                  public porta,portb,portc,portw

                  public worda,wordb,wordc,wordw
```

```
public dcv,acv,res,ilo,ihi,clo,chi,freq,phs



dcv     proc    far     ;   far routine
        push    bp      ;   save  bp
        mov     bp,sp   ; set upto addreess of bp
        mov     bx,[bp]+6
;macro for outputting data to 8088 ports

ofn     macro x,m,n
        mov  al,ds:word&x [ &m ]
        mov  dx,word ptr ds:port&x [ &n ]
        endm

;macro for supplying  convert pulse to adc&
;also for inputting  converted digital data
;thro I/Ocardfor measurement of AC &DC
;voltages

a2d_cnvt    macro       f
            local a1,a2
            mov   ax,0
            mov   dx,01f2h
            mov   cx,0003


     a1:    out   dx,ax
            loop  a1
            mov   ax,4000h
            out   dx,ax
            mov   cx,0005h
     a2:    nop
            loop   a2
            mov    dx,01f2h
            in     al,dx
            mov    ah,al
            mov    dx,01f0h
            in     al,dx
            and    ax,0fffh
            call   decide_&f
            endm


;invoke macro for outputting data
;to control ports
            ofn w,1,0
            ofn w,0,1
            ;enable voltage buffer
;in general ofn x,y,z means this:
;x refers to a port name or a word
;name .for example if x =a
;it means that port and worda is
;referenced
;y refers to the word array .Hence
;if y =2 it means  that the third
;element in that array is accessed
;z refers to the port array .for eg
```

```
  ;if z =1 the seecond port is chosen
  ;y refers to t
           ofn   c,3,Ø
           ofn   a,Ø,1
           ofn   b,Ø,1
           ofn   c,Ø,1
  ;send control pulse to adc &read
  ;converted value
          a2d_cnvt     d
  ;switch to next higher range
          ofn a,1,1
          a2d_cnvt     d
          ofn a,2,1
          a2d_cnvt     d
          ofn c,1,1
          a2d_cnvt     d
          ofn c,2,1
          a2d_cnvt     d
   decide_d    proc    near
 ;this procedure checks if the voltage
 ;is  greater than 1 volt or less than
 ;-1 volt

          cmp  ax,Ø8ceh
          jae  target
          cmp  ax,Ø734h
          jbe  target
          ret
   decide_d       endp
   target:   mov [bx],ax
;return digested voltage to
;calling procedure
     pop  bp
     ret   2
 dcv  endp

;procedure   to handle ac voltage switching
;and measurements


acv       proc    far   ;far routine
          public acv
          push bp
          mov bp,sp

;call macro to perform necessary
;switching

          ofn w,1,Ø
          ofn w,Ø,1
          ofn b,1,Ø
          ofn c,Ø,Ø
          ofn a,3,1
          ofn b,2,1
          ofn c,Ø,1
          a2d_cnvt   a
          ofn a,4,1
          a2d_cnvt   a
          ofn a,5,1
          ofn b,3,1
          a2d_cnvt   a
```

```
                ofn c,2,1
                a2d_cnvt   a
        decide_a    proc      near
        ;this procedure checks if the
        ;RMSvalue lies between .5 volts
        ;and 5 volts
                cmp ax,Ø8ceh
                jae trgt
                cmp ax,Øc ØØh
                jbe trgt
          decide_a   endp
              trgt:  nop
                     mov bx,[bp+6]
                     mov [bx],ax
                     pop bp
                     ret 2
              acv    endp

      ;procedure for servicing resistance
      ;routine
                     public res
              res    proc   far
                     push   bp
                     mov    bp,sp
;MACRO "ACN" HAS BEEN CHANGED TO FAR PROC.& IS AT THE END.
                     call   acn
        ;this procedure returns two flags.one is the
        ;range flag &the other is the actual value
                  decidr  macro  x,y
                          local c3
                          mov si,&x
                          cmp ax,Ø3ffh
                          jae c3
                          cmp ax,Ø798h
                          jbe c3
                          jmp again&y

                 c3:      jmp retval
                          endm

        ;invoke macro to switch &find correct
        ;range

                         ofn w,1,Ø
                         ofn w,Ø,1
                         ofn c,Ø,Ø
                         ofn a,2,1
                         ofn b,4,Ø
                         ofn b,5,1
                         ofn c,Ø,1
                         call acn
                         decidr 1,2
            again2:      ofn b,6,1
                         call acn
                         decidr 2,3
            again3:      ofn b,7,1
                         call acn
                         decidr 3,4
            again4:      ofn b,8,1
```

```
                call acn
                decidr 4,5
    again5:     ofn b,9,1
                ofn c,3,1
                call acn
                decidr 5,6
    again6:     ofn b,1Ø,1
                ofn c,3,1
                call acn
                decidr 6,7
    again7:     nop

    retval:     mov[bp+6],si
                mov[bp+8],ax
                pop bp
;return two integers to calling procedure

                ret 4
    res         endp
;procedure to service current
;measurent (low range)

                public ilo
   ilo          proc   far
                push   bp
                mov bp,sp
   il_decid     macro x,y
                local b1
                mov si,&x
                cmp ax,Ø784h
                jbe b1
                cmp ax,Ø332h
                jae b1
                jmp try&y
  b1:           jmp il_ret
                endm
;call macro to switch &find correct range
               ofn w,1,Ø
               ofn w,Ø,1
               ofn a,2,1
               ofn b,Ø,1
               ofn c,1,1
               ofn c,Ø,Ø
               ofn b,11,Ø
               call acn
               il_decid 1,2
  try2:        ofn b,12,Ø
               call acn
               il_decid 2,3
 try3:         ofn b,13,Ø
               call acn
               il_decid 3,4
 try4:         ofn b,14,Ø
               call acn
               il_decid 4,5
 try5:         ofn b,15,Ø
               call acn
               il_decid 5,6
 try6:         ofn b,16,Ø
               call acn
```

```
                il_decid 6,7
   try7:        nop

  ;range flag to [bp+6]
  ;actual value to [bp+8]
  il_ret:       mov [bp+6],si
                mov [bp+8],ax
                pop bp
                ret 4
   ilo          endp


                public  ihi
  ihi           proc    far
                push bp
                mov bp,sp
  ih_decid      macro x,y
                local b2
                mov si,ax
                cmp ax,Ø798h
                jbe b2
                cmp ax,Ø3ffh
                jae b2
                jmp rpt&y
 b2:            jmp ih_ret
                endm
                ofn w,1,Ø
                ofn w,Ø,1
                ofn c,Ø,Ø
                ofn b,1,Ø
                ofn b,Ø,1
                ofn a,6,1
                ofn c,1,1
                call acn
                ih_decid 1,2
     rpt2:      ofn a,7,1
                call acn
                ih_decid 2,3
     rpt3:      ofn a,8,1
                call acn
                ih_decid 3,4
     rpt4:      ofn c,2,1
                call acn
                ih_decid 4,5
    rpt5:       nop


  ih_ret:       mov [bp+6],si
                mov [bp+8],ax
                pop bp
                ret 4
    ihi         endp

                public clo
   clo          proc far
                push bp
                mov bp,sp

  ;this is a macro which detects
  ;the zero crossing of an input
  ;square wave
```

```
     zero_det   macro
                local b6,b7
                mov dx,Ø1fah
 b6:            in al,dx
                and al,Ø8h
                jnz b6
 b7:            in al,dx
                and al,Ø8h
                jz b7
                endm
;this macro checks if the input
;is '1' for more than 1Ø ms
;the value x in the macro is
;the range flag

chk_1Øms        macro  x
                local c8
                mov  si,x
                mov  cx,Ø428h
                mov dx,Ø1fah
 c8:            in al,dx
                and al,Ø8h
                jz measr
                loop c8
                endm

 chk_1Øms  1
                ofn a,1Ø,1
                zero_det
 chk_1Øms  2
                ofn a,11,1
                zero_det
 chk_1Øms  3
               ofn a,12,1
               zero_det
 chk_1Øms  4
  measr:       mov dx,Ø1fah
               mov cx,ØØØØh
  inpt:        inc cx
               in al,dx
               and al,Ø8h
               jnz inpt
               mov [bp+6],si
               mov [bp+8],cx
               pop bp
               ret 4
   clo        endp

              public chi
   chi        proc far
              push bp
              mov bp,sp
;call macros to switch
        dly    macro
               local kill_time
               mov cx,54ebh
  kill_time :  nop
               nop
   loop kill_time
               endm
```

```
    ch_check    macro
                mov cx,0000
                mov dx,01fah
                inc cx
         chk:   in al,dx
                and al,04h
                loopnz chk
                in al,dx
                and al,04h
                jz m1
jmp short_cap
       m1:      jmp measure
                endm

                ofn w,0,0
                ofn w,0,1
                ofn a,2,1
                ofn b,18,1
                ofn c,3,0
                ofn b,14,0
                mov  cx,0000h
                mov dx,01fah
incr:           inc cx
                in al,dx
                and al,04h
                jnz incr
                mov bx,[bp+6]
                mov [bx],cx
                ofn b,21,0
                dly
                ofn b,23,0
    ch_check
 short_cap:  ofn b,21,0
             dly
  measure:    mov cx,0000h
   more:      inc cx
              in al,dx
              and al,04h
     loopnz more
            mov cx,[bp+8]
            pop bp
            ret 4
     chi    endp


             public freq
   freq     proc far
            push bp
            mov bp,sp
   chk_1ms  macro
            local verfy
            mov cx,009ch
            mov dx,01fah
   verfy:   in al,dx
            and al,01h
            jnz verfy
            endm
```

```
;this macro detects zero cross
  zero_det2  macro
  local      input1,input2
             mov dx,01fah
   input1:   inal,dx
             and al,01h
             jnz input1
   input2:   in al,dx
             and al,01h
             jz input2
             endm

;switch and find correct range
             ofn w,0,0
             ofn w,0,1
             ofn c,3,0
             ofn b,17,0
             ofn b,18,0
             ofn a,4,1
             mov dx,01fah
             mov si,0004h
   zero_det
   chk_1ms
              in al,dx
              jnz l1
              jmp rang1
      l1:     jmp f_measr
      rang1: mov si,0003h
              ofn  a,3,1
   zero_det
   chk_1ms
             in al,dx
             and  al,01h
             jnz l2
             jmp rang2
     l2:     jmp f_measr
     rang2: mov si,0002h

             ofn a,15,1
  zero_det
  chk_1ms
             in al,dx
             and al,01h
             jnz l3
             jmp rang3
     l3:     jmp f_measr
     rang3: mov si,0001h
             ofn a,0,1
    f_measr:   zero_det
             mov cx,0000h
     count:inc cx
           in al,dx
           and al,01h
    loopnz count
;cx has the  value of frequency
;si has the  range multiplier
```

```
            mov  [bp+6],si
            mov  [bp+8],cx
            pop  bp
            ret  4
freq        endp
            public phs
phs         proc  far
            ofn w,0,0
            ofn w,0,1
            mov dx,01fah
            zero_det3 macro
   ;macro for detecting zero cross
c1:         in al, dx
            and al, 02h
            jnz c1
c2:         in al,dx
            and al, 02h
            jnz c2
            endm
            zero_det3
            mov cx, 0000h
  one:      inc cx
  ;count time for which input is '1'
            in al, dx
            and al, 02h
            jz zero
            mov si,0000h
   zero:    inc si
  ;count time for which input is '0'
            in al, dx
            and al, 02h
            jz zero
            mov[bp +6],si
  ;si register has zero count
            mov[bp +8],cx
  ;cx register has one count
            pop bp
            ret 4
    phs     endp

     acn      proc   near;macro
                ; local a3,a4
                 mov ax,0
                 mov dx,01f2h
                 mov cx,0003
      a3:        out dx,ax
                 loop a3
                 mov ax,4000h
                 out dx,ax
                 mov cx,0005h
      a4:        nop
                 loop a4
                 mov dx,01f2h
                 in al,dx
                 mov ah,al
                 mov dx,01f0h
                 in al,dx
                 and ax,0fffh
      acn        endp;endm

    code_1  ends
```

```
DECLARE SUB Menu3 ()
DECLARE SUB Voltage ()
DECLARE SUB Resistance ()
DECLARE SUB Phase ()
DECLARE SUB Menu2 (keyin$)
DECLARE SUB Menu1 (keyin$)
DECLARE SUB Menu (keyin$)
DECLARE SUB Cap ()
DECLARE SUB Current ()
DECLARE SUB Frequency ()
DECLARE SUB Measure ()
DECLARE SUB Ilow ()
DECLARE SUB Ihigh ()
DECLARE SUB Help ()
DECLARE SUB Dlog ()
DECLARE SUB Clow ()
DECLARE SUB Calculate ()
DECLARE SUB Chigh ()

SCREEN Ø, Ø, Ø
DIM SHARED array(1ØØ)

DO
  CALL Menu(keyin$)
  SELECT CASE keyin$
    CASE "H"
      CALL Help
    CASE "M"
      CALL Measure
    CASE "D"
      CALL Dlog
    CASE "E"
       EXIT DO
    CASE ELSE
      BEEP
  END SELECT
LOOP
END



SUB Box (tr%, tc%, br%, bc%)
  wd% = bc% - tc% + 1
  LOCATE tr%, tc%: PRINT "╔" + STRING$(wd% - 2, 2Ø5) + "╗"
  FOR r% = tr% + 1 TO br% - 1
    LOCATE r%, tc%: PRINT "║" + SPACE$(wd% - 2) + "║"
  NEXT r%
  LOCATE br%, bc%: PRINT "╚" + STRING$(wd% - 2, 2Ø5) + "╝"
END SUB

SUB Calculate
  FOR j = Ø TO 49
    sum# = sum# + array(j)
    sqsum# = sqsum# + array(k) ^ 2
  NEXT j
  mean# = sum# / 5Ø
  smean# = (mean#) ^ 2
  sd# = SQR((sqsum# / 5Ø) - smean#)
  CLS
  LOCATE 1Ø, 2Ø
```

```
  PRINT "Press any key to continue"
  wait$ = INPUT$(1)
  CLS
END SUB

SUB Dlog
  CLS
  PRINT "Do you want to Save or Load [S/N] : "
  DO
    a$ = INKEY$
  LOOP WHILE INSTR("SsLl", a$) = 0

  offset = VARPTR(array(0))
  IF a$ = "S" OR a$ = "s" THEN
    PRINT "Your data is being logged"
    BSAVE "prog", offset, 400
  ELSEIF a$ = "L" OR a$ = "l" THEN
    BLOAD "prog", offset
    PRINT "Your data is being loaded"
  END IF
  PRINT "Press any key to continue"
  wait$ = INPUT$(1)
  CLS
END SUB

SUB Frequency
  CLS
  FOR I = 0 TO 49
    CALL freq(a%, B%)
    array(I) = a% * (2 * 10 ^ (-7))
    j = B% - 1
    IF B% < 1 OR B% > 4 THEN
      PRINT "Frequency out of range for the instrument"
    END IF
    array(I) = (1 / array(I)) * 10 ^ j
  NEXT I
  CALL Calculate
  PRINT "Press any key to continue."
  wait$ = INPUT$(1)
END SUB

SUB Help
  CLS
  OPEN "temp.ddd" FOR INPUT AS #1
    FOR I = 1 TO 16
      INPUT #1, a$
      PRINT a$
    NEXT I

    PRINT "Press any key to continue"
    DO WHILE INKEY$ = ""
    LOOP
    CLS
    FOR I = 1 TO 18
      INPUT #1, a$
      PRINT a$
    NEXT I

    PRINT "Press any key to continue"
    DO WHILE INKEY$ = ""
    LOOP
```

```
    CLS
    FOR I = 1 TO 17
      INPUT #1, a$
      PRINT a$
    NEXT I

    PRINT "Press any key to continue"
    DO WHILE INKEY$ = ""
    LOOP
    CLS
    FOR I = 1 TO 19
      INPUT #1, a$
      PRINT a$
    NEXT I

    PRINT "Press any key to continue"
    DO WHILE INKEY$ = ""
    LOOP
    CLS
    FOR I = 1 TO 14
      INPUT #1, a$
      PRINT a$
    NEXT I

    PRINT "Press any key to continue"
    DO WHILE INKEY$ = ""
    LOOP
    CLS
    FOR I = 1 TO 24
      INPUT #1, a$
      PRINT a$
    NEXT I

    PRINT "Press any key to return to main menu"
  CLOSE #1
  wait$ = INPUT$(1)
  CLS
END SUB

SUB Ihigh
  CLS
  FOR I = Ø TO 49
    CALL ihi(a%, B%)
    array(I) = (a% - 2048) / 204.7
    j = B% - 1
    array(I) = (-array(I) / 5) * 1Ø ^ j
    IF B% < 1 OR B% > 4 THEN
      PRINT "Current out of range or too small"
    END IF
  NEXT I
  CALL Calculate
END SUB

SUB Ilow
  CLS
  FOR I = Ø TO 49
    CALL ilo(a%, B%)
    array(I) = (a% - 2048) / 204.7
    IF B% = 1 THEN array(I) = -array(I) / 6ØØØ
    IF B% = 2 THEN array(I) = -array(I) / 12ØØØ
    IF B% = 3 THEN array(I) = -array(I) / 6ØØØØ
    IF B% = 4 THEN array(I) = -array(I) / 12ØØØØ
```

```
  LOCATE 20, 25: PRINT "7. (E)xit"
  LOCATE 22, 25: PRINT "Press V,U,R,C,F,P or E"
  DO
    keyin$ = UCASE$(INPUT$(1))
  LOOP WHILE INSTR("VURCFPE", keyin$) = 0
END SUB

SUB Menu2 (keyin$)
  CLS
  LOCATE 15, 20: PRINT "Choose (H)igh or (L)ow"
  DO
    keyin$ = UCASE$(INPUT$(1))
  LOOP WHILE INSTR("HL", keyin$) = 0
END SUB

SUB Menu3
  CLS
  LOCATE 15, 20: PRINT "Choose (D)C or (A)C"
  DO
    keyin$ = UCASE$(INPUT$(1))
  LOOP WHILE INSTR("AD", keyin$) = 0
END SUB

SUB Phase
  CLS
  FOR I = 1 TO 49
    CALL phs(a%, B%)
    array(I) = (a% / (a% + B%)) * 360
  NEXT I
  CALL Calculate
  PRINT "Press any key to continue."
  wait$ = INPUT$(1)
END SUB

SUB Resistance
  CLS
  FOR I = 0 TO 49
    CALL res(a%, B%)
    array(I) = (a% - 2048) / 204.7
    IF B% = 1 THEN array(I) = -array(I) * 2E+07 / 5
    IF B% = 2 THEN array(I) = -array(I) * 2000000 / 5
    IF B% = 3 THEN array(I) = -array(I) * 200000 / 5
    IF B% = 4 THEN array(I) = -array(I) * 20000 / 5
    IF B% = 5 THEN array(I) = -array(I) * 2100 / .5
    IF B% = 6 THEN array(I) = -array(I) * 300 / .5
    IF B% < 1 OR B% > 6 THEN
      PRINT "Over range or resistance too low"
    END IF
  NEXT I
  CALL Calculate
  PRINT "Press any key to continue"
  wait$ = INPUT$(1)
  CLS
END SUB

SUB Voltage
  CALL Menu3(keyin$)
  SELECT CASE keyin$
    CASE "A"
      FOR I = 0 TO 49
```