

# Voice Recognition Robot

P-1336

A Project Report

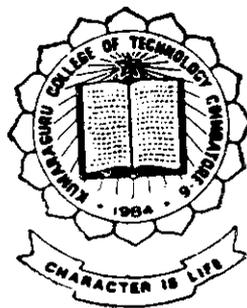
Submitted In partial fulfilment of the requirements  
for the award of the Degree of  
**BACHELOR OF ENGINEERING IN**  
**ELECTRONICS AND COMMUNICATION ENGINEERING**  
of the Bharathiar University, Coimbatore.

Submitted by

V. Anantha Narayanan  
N. Matheshwaran  
T. Natraj  
R. Udhayakumar

Guided by

Ms. A. Vasuki, M.E.



Department of Electronics and Communication Engineering  
Kumaraguru College of Technology

Coimbatore-641 006

1997 - 98

*KUMARAGURU COLLEGE OF TECHNOLOGY*  
COIMBATORE - 641 006

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING  
**CERTIFICATE**

*This is to certify that the Report entitled*

**VOICE RECOGNITION ROBOT**

*has been submitted by*

**MR. V.Ananthanarayanan, N.Matheswaran, T.Nataraj, R.Udaykumar**

*In partial fulfillment of the requirements for the award of Degree of Bachelor of Engineering in the Electronics and Communication Engineering branch of the Bharathiyar University, Coimbatore - 641 006 during the academic year 1997 - '98.*

  
\_\_\_\_\_  
(GUIDE)

\_\_\_\_\_  
(HEAD OF DEPARTMENT)

*Certified that the Candidate was Examined by us in the Project Work Viva –*

*Voce Examination held on \_\_\_\_\_ and the University*

*Register Numbers are \_\_\_\_\_*

\_\_\_\_\_  
(Internal Examiner)

\_\_\_\_\_  
(External Examiner)

# Tamil Nadu State Council for Science and Technology

(Autonomous body under Govt. of Tamil Nadu)



தமிழ்நாடு அறிவியல் தொழில் நுட்ப மாநில மன்றம்

President

Prof. K. ANBAZHAGAN

Hon'ble Minister for Education

Vice-President

Dr. P. RAMASAMY

Member Secretary

Dr. K. DHARMALINGAM

Ref: TNSCST/SPS/VR/97-98/

25.9.97

To

The Principal  
Kumaraguru College of Technology  
Chinnavedampatti (P.O), Coimbatore - 641 006.

Sir,

Sub: Student Project Scheme (1997-98)- Approval and Release  
of grant - Reg.

With reference to the above I am happy to inform that following student project has been approved by the Council for funding. Please find enclosed herewith the cheque towards the grant sanctioned to you as detailed below. A stamped receipt may be sent in acknowledgement of receipt of this grant.

TITLE : Voice Recognition Robot.

SECTOR : Engg. and Tech.

AMOUNT(Rs): 5000

STUD.NAME : V.Ananthanarayanan, T.Nataraj, R.Udayakumar,  
: N.Matheswaran

CODE : EG39

Yours faithfully,

Member Secretary

cc: Miss.Vasuki  
Lecturer  
Dept. of ECE  
Kumaraguru College of Technology  
Chinnavedampatti (P.O), Coimbatore - 641 006.

## **ACKNOWLEDGEMENT**

The authors express their gratitude to the correspondent **Prof.K.Arumugam** and Principal **Dr.S.Subramanian** for the facilities provided for implementing this project.

We are thankful to the **Tamilnadu State Council for Science and Technology** for granting us the scholarship to carry out this project.

Our sincere thanks are to **Prof.M.Ramasamy**, Head of Electronics and Communication Department for constantly encouraging us to successfully complete the project.

We owe a deep sense of gratitude to **Ms.A.Vasuki**, senior lecturer, Department of Electronics and Communication Engineering, for her inspiring guidance, kind advice and encouragement.

We are also grateful to **Prof.S.Jayaraman** and **Mrs.F.T.Vanathi**, of Electronics and Communication Engineering department, PSG College of Technology, for having assisted us.

We offer our sincere thanks all those who have assisted us in one way or other.

## **SYNOPSIS**

The main idea of this project is to recognize the human voice. Basically we drive a robotic arm based on the voice commands. Speech recognition is achieved using PIC 16C57 processor. It is a multi purpose chip, which recognize continuous speech of unlimited vocabulary from low data rate inputs.

The analog voice signal is converted into equivalent pulses. The pulses are then counted. In this project we are recognizing four commands namely: UP, DOWN, TURN LEFT, GO RIGHT.

Firstly the number of pulses for each of the above four commands are taken and stored in PIC. To be more precise we go for large number of samples preferably from the same person. We then determine the expected range of pulses for each of the above four commands.

The ALP actually does the job of finding whether the voice spoken during the experiment, after converting into pulses, falls in any of the above ranges. Accordingly the stepper motor drives the robotic arm structure.

## 1. INTRODUCTION



P-1330

Recognition of speech by a machine has been the goal of research for more than four decades and has inspired such science fiction wonders as *the computer HAL* in Stanley Kuberick's famous movie *2001 - A space Odyssey* and *the robot R2D2* in the George Lucas classic *Star Wars* series of movies. However, inspite of the glamour of designing an intelligent machine that can recognize the spoken word and comprehend its meaning, and inspite of the enormous research efforts spent in trying to create such a machine, we are far from achieving the desired goal of a machine that can understand spoken discourse on any subject by all speakers in all environments.

Generally, there are 3 approaches to speech recognition

1. The acoustic - phonetic Approach.
2. The Pattern recognition Approach.
3. The Artificial Intelligence Approach.

The acoustic - phonetic approach is the speech recognition by a machine whereby the machine attempts to decode the speech signal in a sequential manner based on the observed acoustic features of the signal and the known relations between acoustic features and phonetic symbols.

The pattern recognition method has 2 steps - namely, training of speech patterns and recognition of patterns via pattern comparison.

**FOUNDATION OF VOICE  
RECOGNITION**

## ***2. FOUNDATION OF VOICE RECOGNITION***

### ***2.1. Voice Mechanism - An Introduction***

Sound is a wave phenomenon; i.e., each particle of air vibrates in some fashion and passes on the disturbance to its neighbors. But while this disturbance carries both information and energy to distant places, each bit of air remains always in the vicinity of its original position. In a gas or liquid the local vibration is always parallel to the direction of wave travel; so sound waves are classified as longitudinal waves.

Sound whose frequency lies between 20Hz and 20,000 Hz are audible to human ear. Voice sounds include the vowels of ordinary speech as well as the tone characteristic of the singing voice. The action of the vocal cords produce a saw tooth type of variation in the velocity and correspondingly of pressure in the modulated stream of air. The sound waves, upon analysis into a Fourier series of sinusoidal waves is found to contain a large number of harmonically related frequency components.

### ***2.2 Voice Patterns***

The numerous resonating cavities and orifices of the nose, mouth and the throat form an acoustic network which further modulates the pressure wave of Fig 1.1 (a). Many of these parameters are controllable at will, that is by changing the position of the tongue or altering the configuration of the lips, and thereby a wide variety of voice sounds may be produced. Fig 1.1(b)

shows how the output at the vocal cords may be modified so as to produce the vowel sound “a” as in “father”. This kind of signals may be called periodic signals.

Most sound waves are not sinusoidal. Some, like a steady note played on a flute or a sung vowel, are periodic signals. An oscilloscope display will show the same unit being repeated over and over (Fig 1.2a). After a specific length of time 't' has passed, the same disturbance occurs again; so period and frequencies are still well defined for such a sound. A non-periodic sound signal, a hissing sound will be seen in an oscilloscope (as shown in Fig 1.2b). Such a sound has no single well-defined period or frequency, but should be thought as a mixture of many sounds covering a whole range of frequencies.

Continuously produced voice signals, such as words and sentences are not steady and are not perfectly sinusoidal. An oscilloscope display will show this non-periodicity clearly. These signals do not remain steady for a long time. Fig 1.3 shows the pattern of the signal as displayed on the oscilloscope for various persons for a particular word. The pattern does not repeat itself for different persons because of the difference in pronunciation of the word.

The average speech power emitted by a person at a conversational level is about 10 microwatts, when the power is averaged over a long time interval, i.e., from two to four seconds. When one talks as loudly as possible without straining the vocal cords, this average speech power rises to about 200 microwatts and upon shouting to about 1000 microwatts. By contrast the speech power associated with whispering is about 0.001 microwatt.

### ***2.3. Designing a voice recognition system***

The most important part in creating voice recognition system is to analyze the voice and create templates for comparison with the reproduced voice. Understanding speech has many difficulties, as pronunciation varies for each speaker. Thus the acoustic signal is different for a same word spoken twice. An optimum tolerance has to be given to the recognition system for such variation. This increases the probability of false recognition and thereby reducing the efficiency of the system.

A continuous speech recognition system is highly complex, effect of phones over the consecutive phones blends the acoustic wave patterns, and these patterns are highly difficult to analyze. These patterns are different from those of isolated words. Full understanding of total sentences and intended machine response involves co-operation use of some advanced techniques in areas like acoustics, phonetics, syntax, pragmatics and semantics. Such a

system can be developed only by using some artificial intelligence (AI) and other similar languages, with a computer.

#### ***2.4. Male / Female sound differentiation***

The most significant factor that is often mentioned regarding male and female voice is the difference in fundamental frequency. The fundamental frequency for females lies in the range of 150 to 300 Hz, while it is around 85 to 180 MHz for male voice.

In terms of formants, on an average, five formants are required for male speech and four for female speech. Female vocal tract is shorter than that of male. Formant frequencies are inversely proportional to the vocal tract length. The intensity of the voiced speech generated by a female is about 6 dB less than the voice speech of a male identical condition.

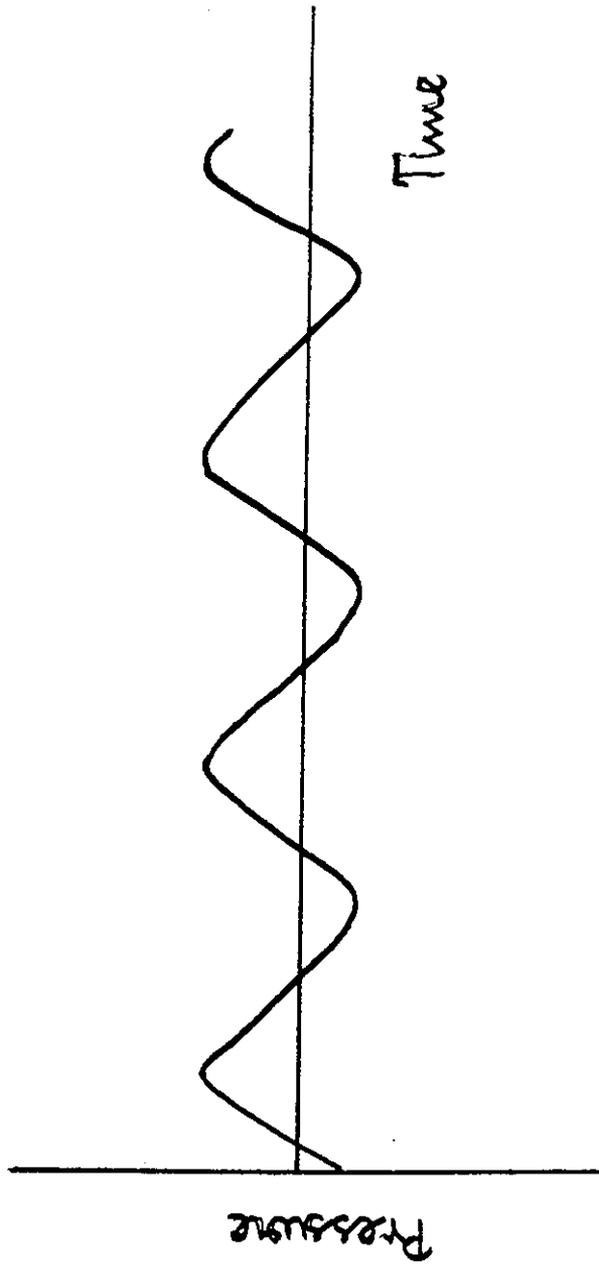
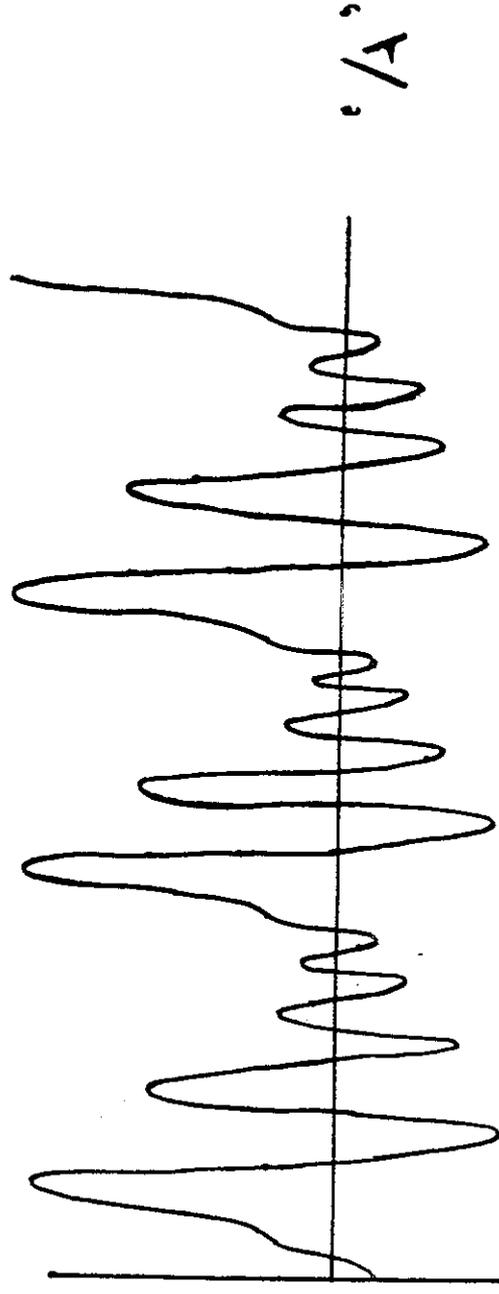


Fig. 1.1.

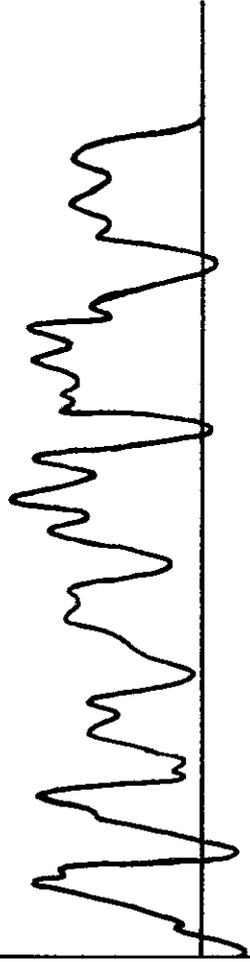
# ACOUSTIC PRESSURE OUTPUT OF VOCAL CORDS



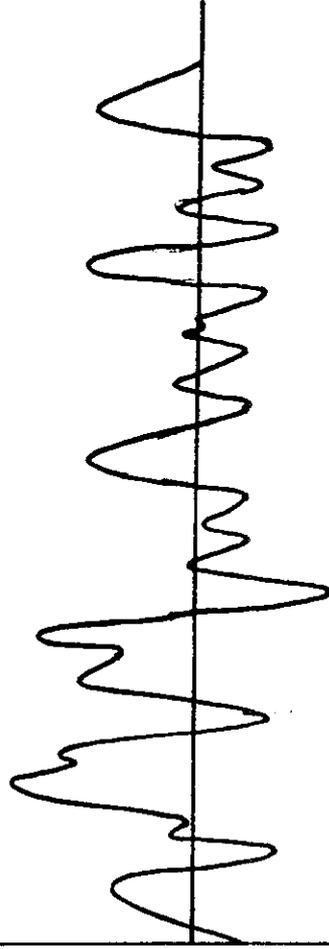
ACOUSTIC PRESSURE OUTPUT  
FOR VOWEL SOUND OF 'A' AS IN FATHER



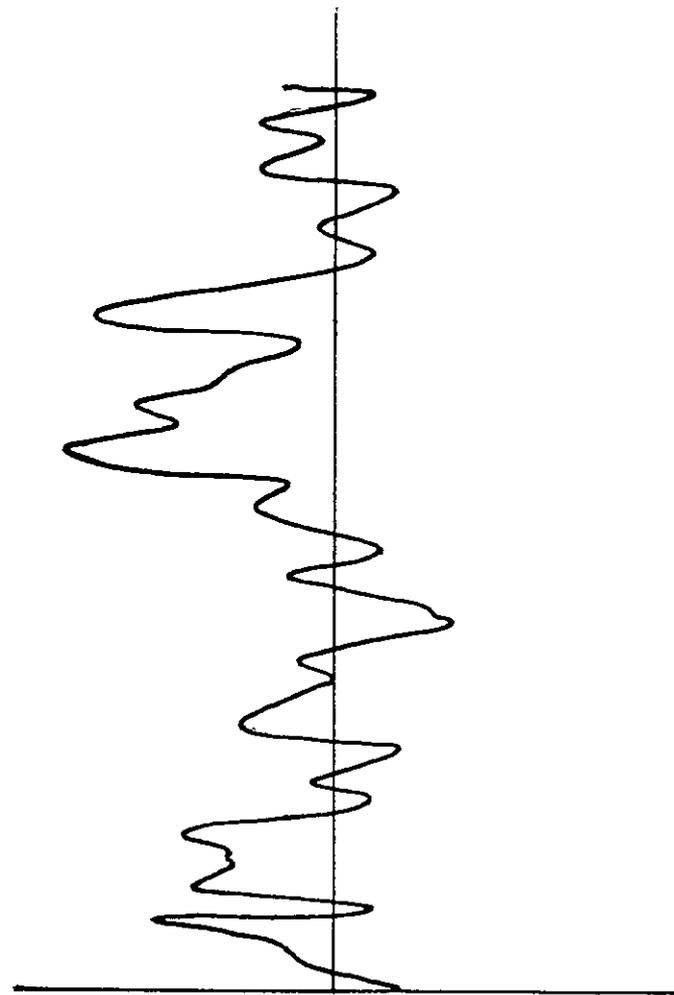
OSCILLOSCOPE DISPLAY FOR  
PERIOD SIGNAL FOR A STEADY VOWEL



OSCILLOSCOPE DISPLAY FOR  
NON-PERIODIC SIGNAL, A HISSING  
SOUND



OSCILLOSCOPE DISPLAY FOR  
THE WORD 'GO'



OSCILLOSCOPE DISPLAY FOR THE  
WORD 'GO'

### 3. PIC16C57 MICROCONTROLLER

#### 3.1. Peripheral Features

The PIC's have a number of not-so-obvious features and qualities that are important:

- **Fully static operation**

Allows you to stop the oscillator and then restart where you left off, with all registers intact. This is useful in applications where power conservation is important.

- **Individual I/O pins are programmable as inputs or outputs.**

Sink current: 25mA per pin, 50 mA per port

Source Current: 20 m A per pin, 40 mA per port

- **2 level hardware stack**

- **Real-Time Clock/Counter (RTCC).**

The RTCC is an 8-bit counter, which can be driven by the RTCC pin or by the PIC's internal instruction clock (OSC/\$). If the external pin is used, the counter can be set to increment on low-to-high or high-to-low transitions. Normally RTCC is driven by either source. For higher count values, though, the prescaler can be used to effectively increase the RTCC to 16 bits. The RTCC signal source and trigger edge is determined by bits in the Option register. See the Microchip PIC16C5x data sheet for details concerning timing characteristics for the RTCC's external input.

- **Prescaler/Postscaler**

This 8-bit counter can be assigned to the RTCC (as a prescaler) or the watchdog timer (as a Postscaler). For simplicity, this counter is normally referred to as the “prescaler”, even when it’s used as a Postscaler.

When assigned to the RTCC, the prescaler is placed between the RTCC and its clock source. The clock signal which would normally increment the RTCC, increments the prescaler. When the prescaler overflows, the RTCC is incremented. Increment ratios from 1:2 - 1:256 can be used, effectively giving you a 16-bit RTCC.

When assigned to the watchdog timer, the prescaler is placed between the watchdog timer and the PIC’s reset circuit. The watchdog timer signal, which would be normally, reset the PIC, increments the prescaler. When the prescaler overflows, the PIC is reset. Delay ratios from 1:1 - 1:128 can be used, allowing the watchdog period to be set from 18ms to several seconds.

The prescaler setup is determined by bits in the Option register.

- **Programmable Oscillator Type**

The PIC can run with any of four oscillator types, as shown below:

LP: Low Power crystal (from DC to 40 kHz)

RC : Resistor & capacitor(from DC to 4 MHz.+/- 13 to 39 percent)

XT: Crystal Resonator (100 kHz - 4 MHz)

HS: High-speed crystal (4 - 20 MHz)

The oscillator type is determined by two EPROM bits, which are normally programmed at the factory. The RC and erasable PIC's however can be user-programmed for any other oscillator types. Keep in mind, though, that RC-type PIC's are only tested for use with RC oscillators.

- **Internal Architecture**

Internally, the PIC is based on a register file concept with separate busses and memories for data and instructions (sometimes called "Harvard Architecture"). The data bus and memory (RAM) are 8-bits wide, while the program bus and memory (EPROM) is 12-bits wide. All PIC instructions and their operands fit into a single 12-bit word, resulting in smaller code and faster execution. PIC programs are typically 35-50 percent smaller programs written for 8-bit processors. And most instructions execute in a single instruction cycle (4 clock cycles); instructions that affect the program counter take an extra instruction cycle, for a total of 8 clock cycles. To further increase speed, the PIC uses overlapping instruction fetch and execution cycles; while one instruction is executed, the following instruction is being read from program memory. Because of its efficiency, the PIC can deliver 5 MIPS execution with a clock frequency of 20 MHz.

- **PIC16C57 Registers.**

The following table shows the various PIC16C5x registers; the function of each register is described in the following pages.

Register	Function
00h	Indirect addressing register
01h	Real-time clock/counter (RTCC)
02h	Program Counter (PC)
-	Stack registers (2)
03h	Status Register
04h	File Select Register (FSR)
05h	I/O Port A
06h	I/O Port B
07h	I/O Port C
-	TRISA
-	TRISB
-	TRISC
	W register
	Option register
08h - 0Fh	General Purpose Register
10h - 1Fh	General Purpose Register (4 banks in PIC16C57).

The following text describes the function of each register. For some of the registers, you will notice the designation “xxh” following the register name. This indicates the address of the register. Registers with no address cannot be addressed directly.

- **Indirect Addressing Register (00h)**

This register does not actually exist. Naming register 00h in an instruction causes the PIC to read the register pointed to it by register 04h (file select register). For example, the instruction “ADD 00h, #05” will add five to register 00h; instead, it will add five to whatever register is pointed to by the address in register 04h.

If register 00h itself is read through register 04h(04h contains “00h”). 00h will be returned. If register 00h is written through register 04h, the PIC will execute a NOP.

- **Real time Clock/Counter (01h:RTCC)**

This is the location of the RTCC. Although its contents may change in response to a clock signal, the RTCC register may be read and written just as any other register.

- **Program Counter (02h : PC)**

The program counter holds the address for the instruction currently being executed. The program counter and its associated two-level stack are 9-11 bits wide, depending on the EPROM size of the PIC being used.

Certain instruction affect the program counter, as shown below:

GOTO (Microchip) and JMP (Parallax) load the lower 9 bits of the program counter. In the PIC16C56 and ‘57, which have more than 512 words of EPROM, the upper two bits of the program counter are loaded with the page select bits from the status register. The Parallax instruction

set includes a convenient instruction, LJMP, which sets the page select bits before executing the jump.

CALL loads the lower 8 bits of the program counter and clears the ninth bit. The program counter + 1 is pushed into the stack. In the '56 and '57, the upper two bits of the program counter are loaded with the page select bits from the status register. The Parallax instruction set LCALL, which sets the page, select bits before executing the jump.

RETLW (Microchip) and RETW (Parallax) load the program counter with the address almost recently pushed on the stack by a CALL instruction.

Instructions, which load a computed value into the program counter, such as JMP PC+ W, load the value into the lower 8 bits. The ninth bit of the program counter is cleared. In the PIC16C56 and '57, the upper bits of the program counter are loaded with the page select bits from the status register.

It should be noted that because the ninth bit of the program counter is cleared by CALL instructions and computed value instructions (such as JMP PC + W), all subroutine calls and computed jumps have their destination in the first 256 locations of any page. (Each page is 512 words).

As you may have noticed when reading the JMP and CALL paragraphs above the program counter may not be loaded as expected when using a PIC with more than 512 words of EPROM. This is because the upper two bits of the program counter are loaded with the page select bits from the

status register. If your program continues into the second page (probably not what you want). To avoid this mistake, make sure to set the page select bits are for the correct page. Or use, Parallax “long” instruction, which do this for you.

- **Stack**

The stack is a pair of registers, which are used for calling and returning from subroutines. The stack is affected by two instructions:

When a CALL is executed, the first stack register is copied into the second register, then the program counter + 1 (the return address) is loaded into the first register. The original contents of the second register are lost. Finally, the program counter is loaded with the subroutine address, at which point execution continues.

When a RETLW (Microchip) or RETW (Parallax) is executed, the first stack register is copied into the program counter, then the second register is copied into the first register. Execution continues at the address loaded from the first stack register.

- **Status Register (03h)**

This register contains the status of the PIC’s arithmetic logic unit (ALU), the reset status, and the page select bits for PIC’s with more than 512 words of EPROM.

The function of each bit in the status register is shown below:

Bit	Function
0	Carry bit(C). Set if an addition or subtraction causes a overflow from the most significant bit of the resultant (Bit 7). Subtraction is included because it is executed by adding the two's complement. Also used to rotate instructions, which rotate the contents of a register and copy the low or high order bit register into the carry bit.
1	Digit carry bit (DC). Set if an addition or subtraction causes an overflow from the 4th low order bit (bit 3). digit carry indicates that more than one hex digit (4 bits) was necessary to accommodate the result.
2	Zero bit (Z). Statement if the result of an arithmetic or logic operation is zero.
3	Power-down bit (PD). Set during power-up or by CLR WDT (clear watch dog) instruction. Cleared by a SLEEP instruction.

- 4 Time-out bit (TO). Set during power-up, by CLR WDT, or by SLEEP. Cleared by a watchdog time-out.
- 5&6 Page select bits (PA0, PA1). In the '54 and '55, these are unused. In the '56, bit 5 selects program page 0 or 1 (bit 6 unused). In the '57 both bits select page 0, 1, 2 or 3. Each page is 512 words long.
- 7 Unused bit (PA2). Reserved by Microchip for future use.

### ***3.2. PIC Applications***

The main applications of PIC are

#### **1. Interfacing a Matrix Keypad.**

This application note covers the use of matrix-encoded keypads with PIC microcontrollers. It presents an example program in Parallax assembly language for reading a 4 \* 4 keypad.

#### **2. Receiving RS-232 Serial Data.**

This application note presents a simple program for receiving asynchronous serial data with PIC microcontrollers. The example program, written using Parallax assembly language, displays received bytes on a bank of eight LED's.

### **3. Sending RS-232 Serial Data.**

This application note covers transmission of asynchronous serial data using PIC microcontrollers. It presents an example program in Parallax assembly language that transmits a text string serially via RS-232 at a speed upto 19,200 bits per second. Hardware examples demonstrate the use of a popular serial line driver, as well as a method for using the PIC's output to directly drive a serial line.

### **4. Reading Rotary Encoders.**

This application note covers the use of incremental rotary encoders with PIC microcontrollers. It presents an example program in Parallax assembly language for reading a typical encoder and displaying the results as an up/down count on a seven-segment LED display.

### **5. Producing Sound & Music.**

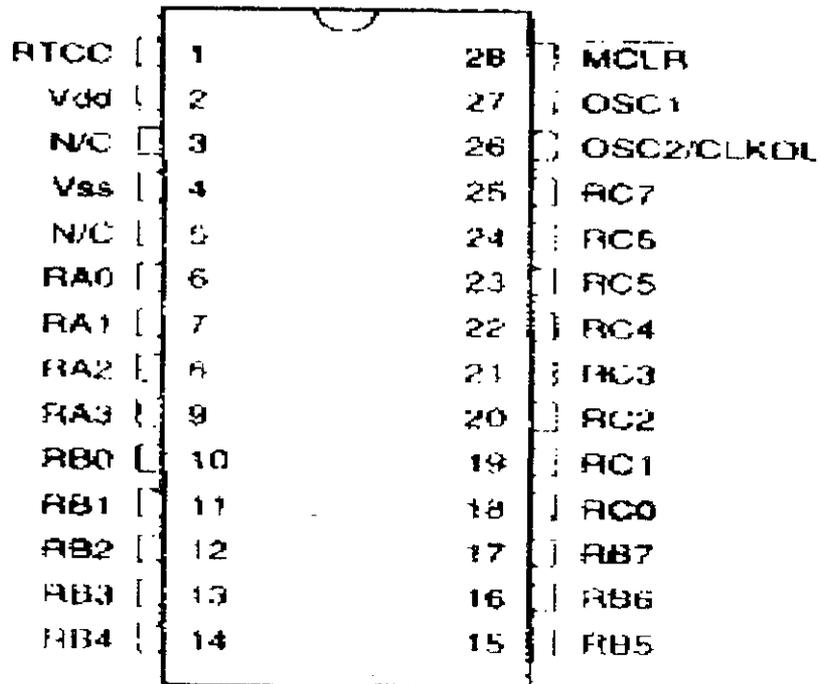
This application note presents a subroutine for production tones by specifying values for frequency and duration. The example program plays a brief tune through a speaker. The circuit uses two PIC outputs driven differentially to produce a 10-volt peak-to-peak signal from a single-ended power supply.

### **6. Driving a LCD display.**

This application note shows how to interface PIC microcontrollers to common Hitachi -Liquid Crystal Display (LCD) modules. The program, in

Parallax assembly language writes text to display reads display status and creates custom character patterns.

## PIC16C55, -57

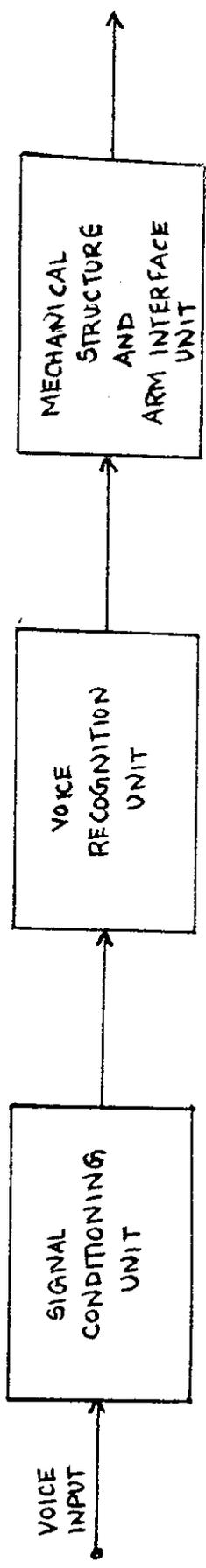


**PDIP  
CERDIP  
SOIC**

PIN:

FUNCTION

RA0 - RA3	IO PORT A
RB0 - RB7	IO PORT B
RC0 - RC7	IO PORT C
RTCC	REAL TIME CLK/COUNTER INPUT
OSC1	OSCILLATOR IP
OSC2/CLK OUT	OSCILLATOR OP
VDD	POWER SUPPLY
VSS	GROUND



# BLOCK DIAGRAM

## ***4. HARDWARE ORGANISATION***

### ***4.1. Block Diagram Representation***

The above block diagram gives an overview of the project.

It consists of the following blocks.

- Signal conditioning unit
- Voice recognition unit
- Mechanical structure

As shown in the above block diagram, the voice signal is given as the input to the signal conditioning unit. In this unit the voice signal is amplified and then filtered using a proper filter and then the signal is being converted into a pulse waveform with the help of switching transistor.

The signal which is converted into the pulse waveform is now fed to the Voice Recognition Unit. This unit consists of a PIC (Programmable Interrupt Controller) which has got an inbuilt timer, a counter, and various other parts which have already been discussed in chapter 2. The signal output from the signal-conditioning unit is given as the input to this unit. Here the pulses, fed as the input, are being counted by the timer-counter and it is compared to that of the already stored data's. If the data, which is coming as the input, and the data, which is stored as template in the PIC, are the same then an output is obtained which is fed as the input to the mechanical structure block.

The output of the Voice Recognition Unit is fed as the input to the mechanical structure which consists of two stepper motors and a robo arm which is interfaced with that of the stepper motor driver circuit. This circuit consists of a buffer through the output of the Voice Recognition Unit is fed.

#### ***4.2. Signal Conditioning Unit***

Voice signal from the microphone is of very low volt signal, in the order of millivolts. It is amplified by using op-amp 741. First the voltage signal from the microphone is given to the second pin of IC 741. It is inverting input pin then the third pin is connected to the ground. The feedback resistor is to vary the gain. The gain of the amplifier is given below.

$$A_v = - R_f / R_1$$

Where  $R_f$  = feedback resistance

$R_1$  = input resistance

We design our amplifier to give two to three volt output. It is in the inverting output mode. So that we have to design a unity gain feedback amplifier this must be inverting amplifier. The output of the first amplifier is given to the second pin of the another 741 third pin is connected to the ground, the sixth pin i.e., the output pin is connected to the second pin.

Then we need a 5v pulse to give input to the processor. So we have to make the voice signal as in the pulse form. So that we decide to go for switching transistor.

The output from the unity feedback amplifier is given to the base of the switching transistor. The collector is connected to the 5v supply and the emitter is connected to the ground. When we give a pulse to the switching transistor's base the collector will give output pulse of 5v amplitude the transistor base requires minimum of 0.7v to conduct.

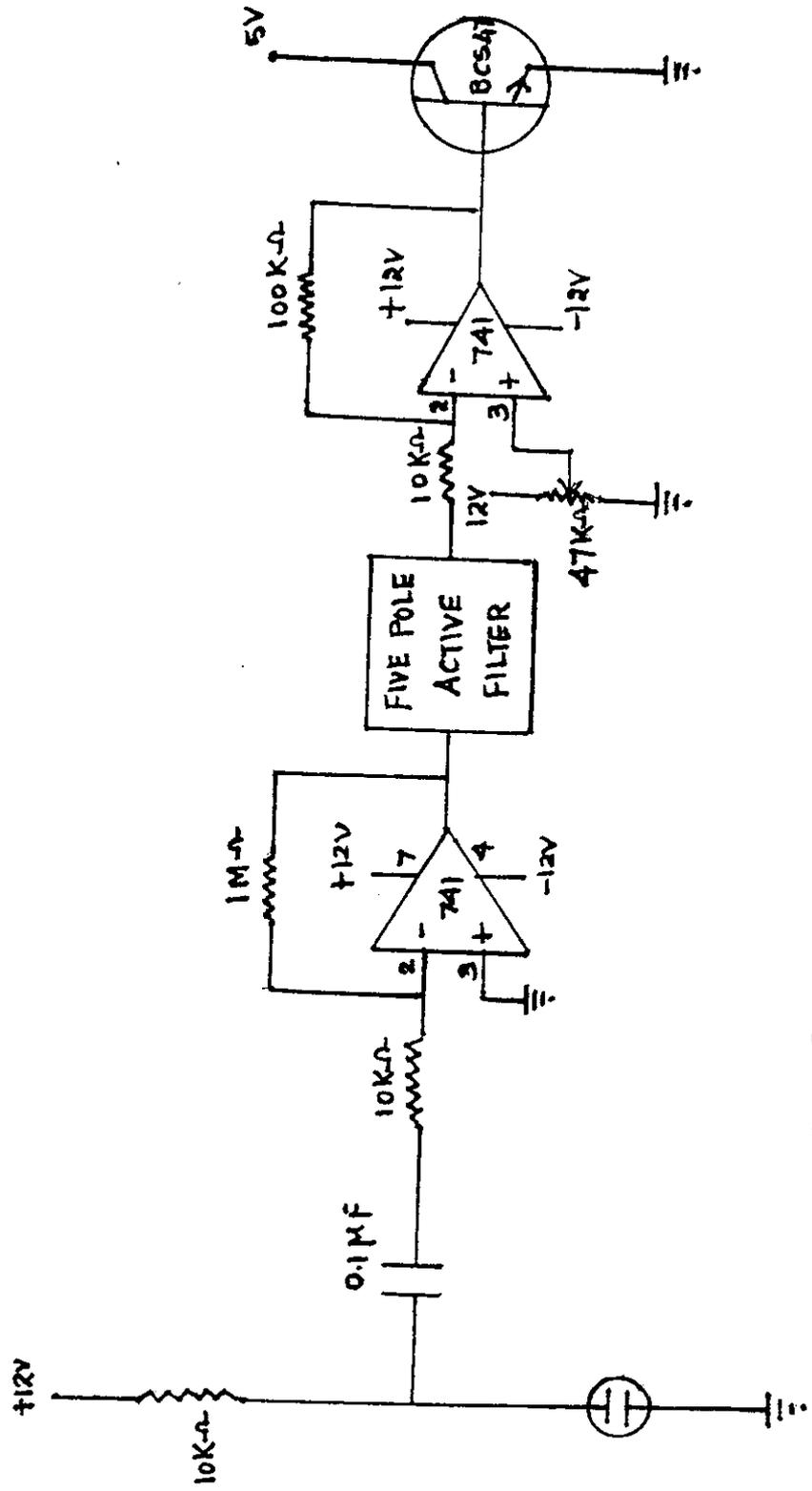
### ***4.3. Voice Recognition Unit***

Then the pulse is given to the I/O line of the processor. In the processor we made a program to count the pulse. According to the count the stepper motor will run in either clockwise direction or in the anticlockwise direction. According to the count the processor will give the output pulse through its I/O line then the signal from the I/O line is connected to the stepper motor driving card.

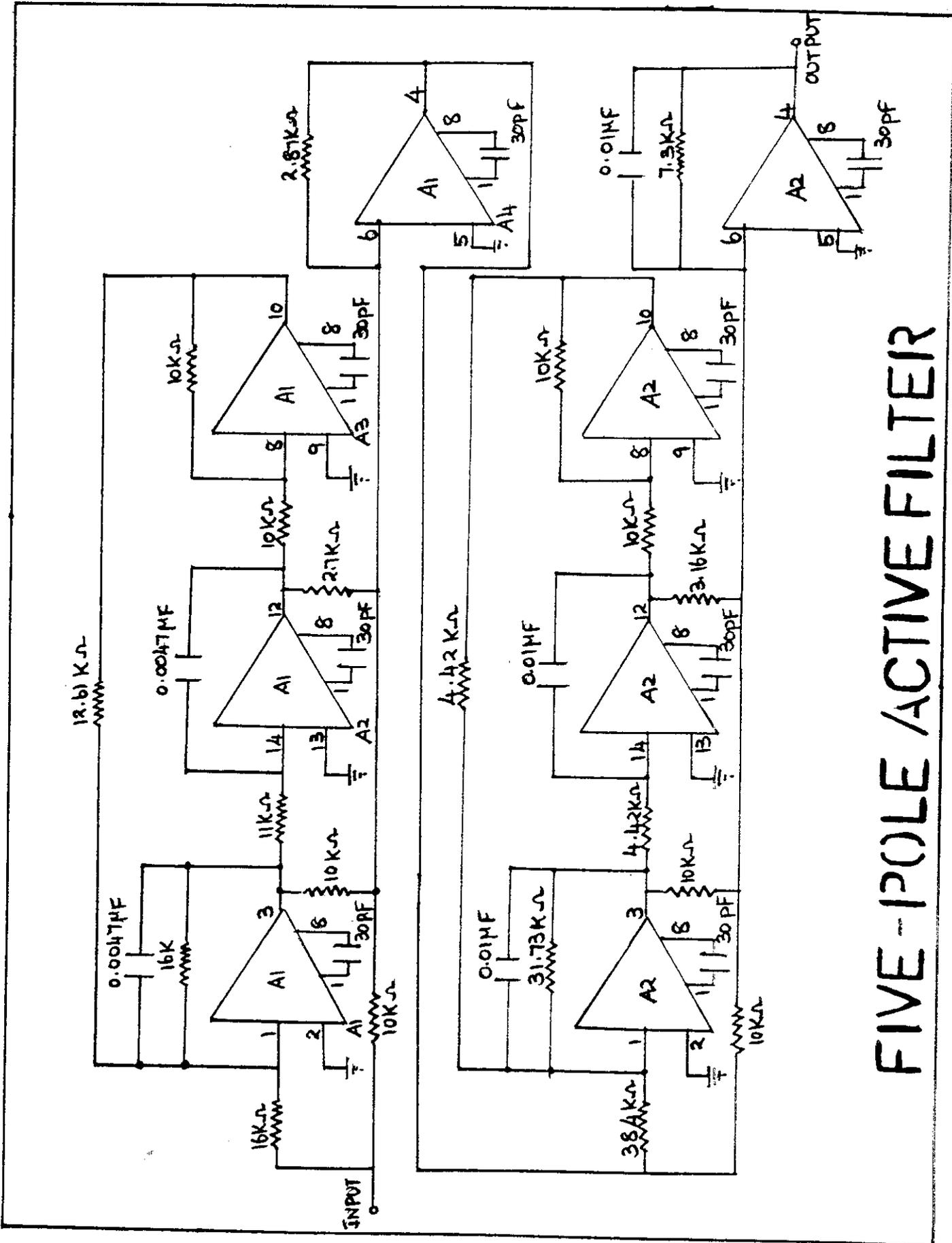
The four I/O lines are needed for a single stepper motor. Two phases must be on at the same time to run the stepper motor. The working principle of the stepper motor driving card is given in the mechanical structure, which is described in the next topic.

#### ***4.4. Power supply for the processor***

The processor board consumes a current of 440 mA. For this a 0 - 12, amps Transformer is used, whose output is rectified using Full-wave rectifier and then filtered using a 2200 micro farad capacitor and given to a 7805 regulator IC. The IC is provided with suitable heat sink to dissipate the heat. The Output of 7805 is given to the processor board.

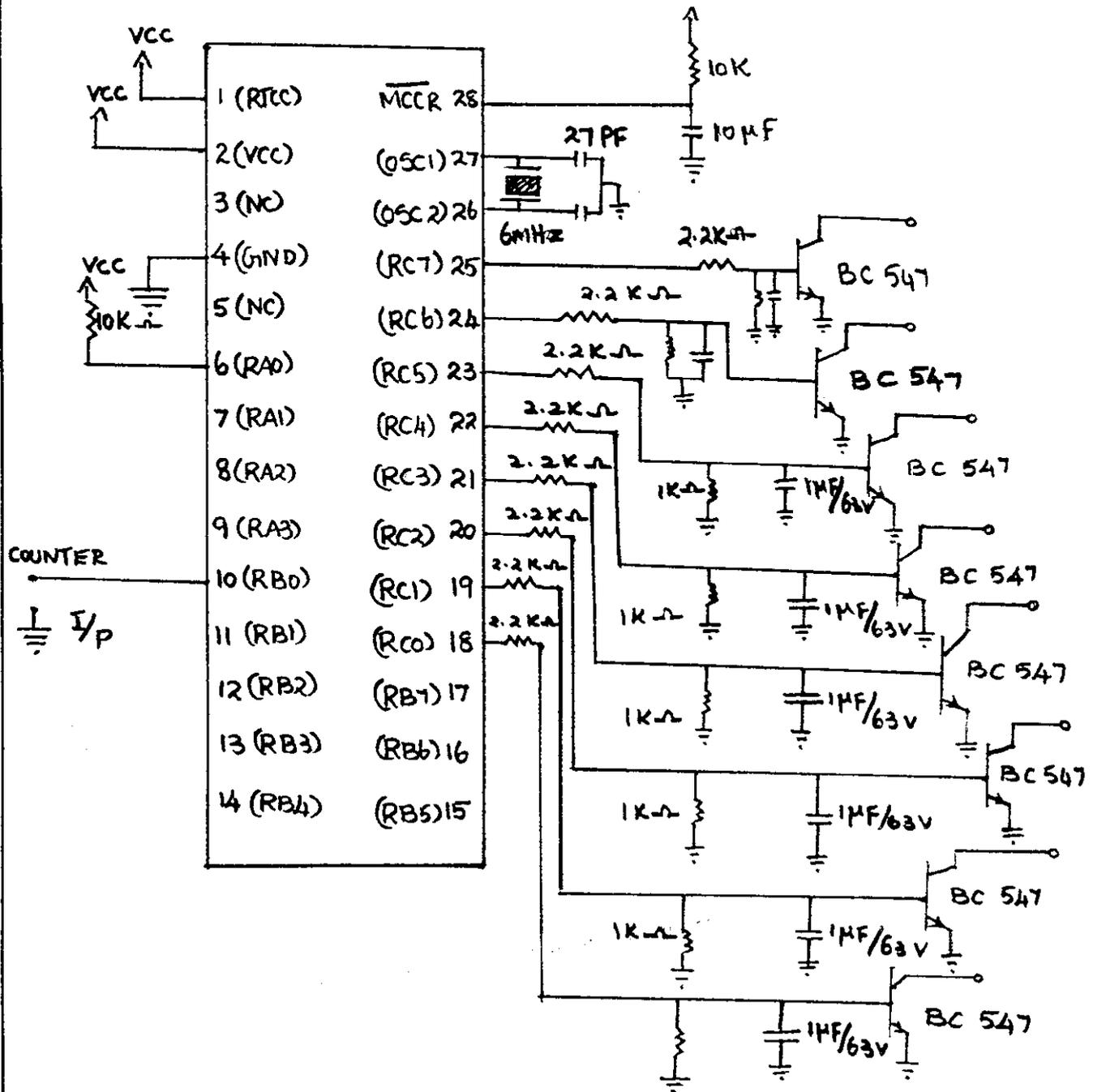


# SIGNAL CONDITIONING UNIT

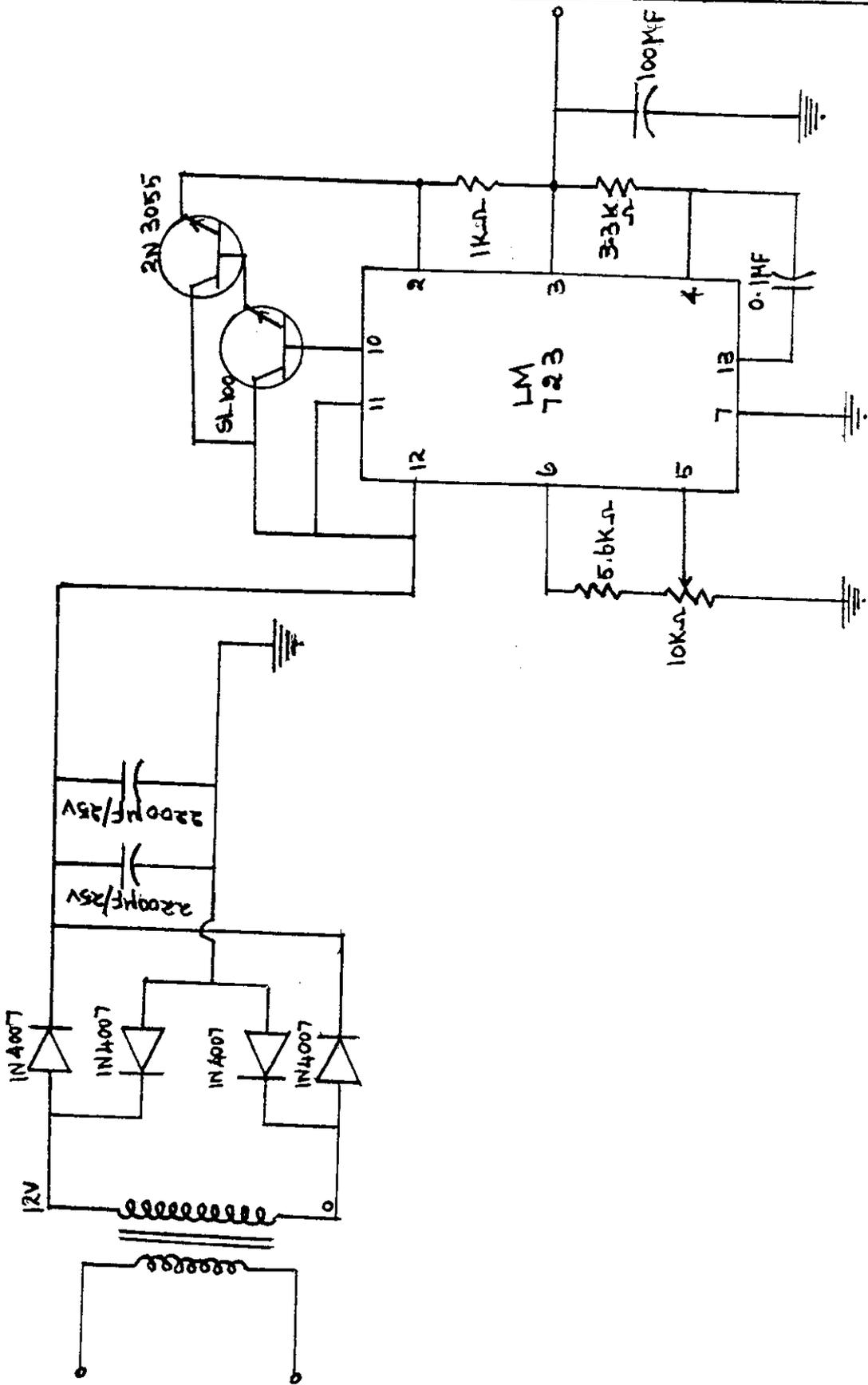


# FIVE-POLE ACTIVE FILTER

# VOICE RECOGNITION UNIT



# POWER SUPPLY



## ***5. MECHANICAL STRUCTURE***

### ***5.1. Introduction***

An industrial robot is a re-programmable, multifunction manipulator designed to move, materials, parts, tools etc through variable programmed motions for performing a variety of tasks. In present day robots, the most obvious anthropomorphic characteristic is the robot's mechanical arm, which is used for performing various industrial tasks. These capabilities permit the robot to perform a variety of tasks in the industry.

Robots are particularly useful in a wide variety of industrial applications, such as material handling, painting, welding, inspection and assembly. However the most impressive fact is that the current research effort focuses on creating a "smart" robot that can "see", "hear", "touch" and make decisions.

Automation and robotics are two closely related technologies. Industrial robots, as other manufacturing systems, are advanced automation systems that use computers or other control devices as the integral part of their control.

Robotics is an applied engineering science that has been referred to as a combination of machine design, control theory, microelectronics, computer programming, artificial intelligence, human factors and production theory.

Advancements in technology will enlarge the scope of the applications of robotics.

To describe the technology of the robot, variety of technical features about the way the robot is constructed and the way it operates must be defined. Robots work with sensors, tools and grippers. The programming of the robot works in various ways.

### ***5.2. Voice Controlled Robot***

The common feature of all robots is their ability to repeat a programmed sequence of operations as long as necessary. However, these robots are unable to sense and respond to any changes in their environment. For example, if the robot was programmed to grip at a certain point, the robot will always close its gripper jaws at that point, even if a part is not there or if an obstacle is inserted in the robot's path, the robot will collide with it and will not move around the collision to avoid collision.

In order for robots to operate effectively in a changing environment, they must be equipped with sensors and must have some degree of Artificial Intelligence (AI). Such robots are called intelligent robots.

Such an intelligent robot must have accurate and tactile sensors to monitor the environment continuously. The sensors include proximity sensors, visual sensors, acoustic sensors, etc.,

One such robot is our VOICE CONTROLLED ROBOT which can respond to changes in the environment upon commands from it's user. The level of sophistication of the robot depends on how well the voice recognition system works.

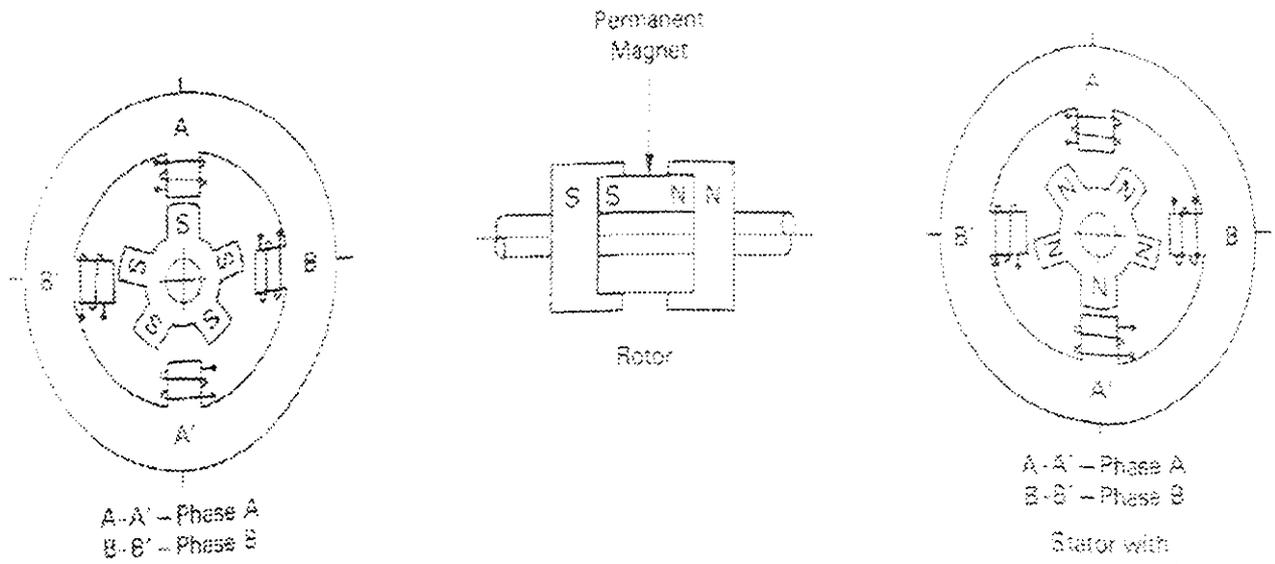
For example, a robot control system would make use of the words, "up", "down", "turn left", "go right", etc. In the subsequent use of the system, the operator would repeat these words into the system a number of times so that the speech recognition system could understand each word in sequence. For each word or set of words, the robot would accomplish some corresponding action or task.

The advantage of voice programming is that the human operator would be able to command, communicate with the robot more quickly and in a way that is more easier and natural to the operator.

Uses for voice controlled robots include applications in hazardous environments like nuclear power plants where direct human contact is not possible. These robots are also used in household purposes and for helping handicapped people, particularly the blind. In applications where the robot cannot be programmed and direct human contact is not possible, the ultimate choice would be the voice-controlled robot.

### 5.3. Stepper Motor

#### 5.3.1. Introduction



The increasing trend towards digital control of machines and process functions has generated a demand for mechanical devices capable of delivering incremental motions of predictable accuracy. The answer to this question was found with advent of stepper motors. The stepper motor is a digital device, which converts electrical pulses into proportionate mechanical movement. Each movement of stepper motor is made up of a series of discrete individual steps. The motor provides clockwise or anticlockwise rotation and thus is ideally suited for our application.

#### 5.3.2. Distinguishing features

Now the question that comes to mind is that what is the difference between a stepper motor and a conventional ac or dc motor. Conventional ac

or dc motors have a free turning shaft. The stepper motor shaft rotating is incremental. The basic feature of stepper motor is that upon being energized it will move and come to rest after some number of steps in strict accordance with the digital input commands provided, therefore allows control of load's velocity and direction. The repeatability is very good. The only system error introduced by the stepper motor is its single step error, which is a small percentage of one step and is generally less than 5%. Maximum dynamic torque in a stepper occurs at low pulse rates therefore it can easily accelerate a load. Once the required position is reached and the command pulse ceases the shaft stops without the need for brakes. The actual rotational movements or step angles of shaft are from 1.8-90 deg., depending on particular motor chosen. Thus this robo arm employs a stepper motor, which has a step angle of 1.8 deg., a stream of 200 pulses will give an angular displacement of 360 deg., or one complete rev.

### **5.3.3. Types of stepper motor**

Stepper motors are divided into 3 principle types or classes, each with distinct construction and performance characteristics.

1. Variable reluctance
2. Permanent magnet
  - I. PM - hybrid

The permanent magnet stepper motor is the most commonly used type. It consists of a multiphased stator and a two-part, permanent- magnet rotor. Both of these structures are toothed (shown in fig). The opposite ends of the rotor are north and south poles of a PM with the teeth at these ends being offset by half a tooth inch. The PM stepper motors we use are operated in the microstep mode.

Most often the microstep is determined by dividing the angular distance of a full step by an integral power of 2(eg., 2,4,8,16 or 32; this produces the smallest computational burden on the stepper motor control). Although microstepping mode requires considerably complex switching circuitry to implement so the cost is quite a bit higher than that required for full-step operation, its use generally produces smoother low-speed operation of the motor. In a robot application, this is an important consideration since oscillation at the desired final point is usually unacceptable.

Characteristics of the permanent magnet Stepper motor can be represented in the Table form.

CHARACTERISTICS	PM MOTOR
I. Motor	Magnetized
II. Rotor Position	Depends on Stator excitation polarity
III. Rotor inertia	High due to magnet
IV. Mechanical response	Not as good ( due to high inertia )
V. Inductance	Low due to Rotor offset
VI. Electrical response	Faster current rise ( due to low inductance )

Table 5.1.

#### 5.3.4. Stepper motor interface

The stepper motor has four windings, which are to be excited in sequence for normal operation. The primary voltage waveforms for this are obtained from the ports of PIC 16C 57. The purpose of interfacing is to translate these voltage levels to a level compatible with the stepper motor.

Peripheral device in this system is the stepper motor. It has rated supply of 3 volts and a current rating of 3.1 amps. Per phase for normal operation .the control signals obtained from the PIC 1657 output has an amplitude of 5 volts and has a square wave nature. This waveform is to be translated to a waveform of peak value 3 volts to the stepper motor.

#### **5.4. Necessity of high input impedance stages**

The output pins of PIC can source only a few tens a milliampere from +5 supply and sink only 1 or 2 milliamps to gnd. For controlling high power devices, buffering the post devices with TTL buffers such as 7406 Hex Non inverting is essential. The 7406 has open collector output, so one has to pull up each output to +3 volts. Each of the buffers can sink as much as 40 milliamps to the gnd. So the 7406 buffers are employed to increase the input impedance of the interface circuit to limit the current loading on the PIC to a few milliamps.

#### **5.5. Interface Circuit**

The interface circuit is connected to input output port of the PIC via the buffer. Each phase of the stepper motor is driven by three transistors SL100, TIP122 and 2N3055.(shown in fig.) which translates the low value input to high level output.

##### **5.5.1. Operation**

The control waveform switches the voltage supply across the winding of the stepper motor. The interface consists of a darlington pair. The input transistors SL100 are the low power high frequency transistors and output transistors are power transistors capable of handling high current that is to be delivered to the motor. The output transistor is operated with the winding of

stepper motor as load in the collector circuit. When a high level is present at output of the port the transistor SL100 draws some base current from the port typically of the order of few milliamps. To keep this base current from loading the input output port, open collector buffer is provided.

This base current gets amplified in SL100 and emitter current of SL100 will be  $h_{fe}+1$  times the base current. The emitter current of SL100 is fed to the base of transistor TIP122 for further amplification. The result is that the pair acts as a single transistor with current gain equal to  $[h_{fe}+1]*h_{fe2}$  which is approx. equal to  $h_{fe1}*h_{fe2}$ . The darlington pair is employed in order to obtain an output collector current of TIP122 conforming to required value while at the same time maintaining the input base current of SL100 to a safe optimum value.

### **5.5.2. The Free Wheeling Diode.**

The reverse biased free wheeling diode (1N4007) is connected across the coil of the motor. This diode suppresses the transients in the winding by providing leakage path in the direction opposite to the normal current in the winding.

This diode is a very important component when one is driving an inductive load as a stepper motor. The basic principle of an inductor is that it fights a change of current through it. When a voltage is applied to the coil by turning on the transistors it takes a while for the current to start flowing. This does not cause any major problem. However when the transistor is turned off the

collapsing magnetic field in the inductor keeps a current flowing for a while. This current cannot flow through the transistor because it is cut off. Instead this current develops a voltage across the inductor or produces an inductive kick. It is usually large enough to break down the transistor in the absence of the diode.

When the coil is conducting the diode is reverse biased. As soon as induced voltage reaches 0.7 volt this diode turns on and the supplies return path for induced current. In effect voltage across inductor is clamped at 0.7 volt which saves the transistor.

### 5.5.3. Interface Board Design

Current sourcing of PIC 16C57 is of the order of 2 milliamperes. This current is used to drive the base of SL100. The collector current of TIP122 is 3.1 amps for a torque of 3 kgf-cm.

Since  $I_c = h_{fe} * I_b$

Assuming an optimum value of current gain,

$I_{b2}$  of TIP122 =  $I_c / h_{fe} = 3.1 / 50 = 0.062$  amps.

This is the emitter current of SL100.

Therefore  $I_b$  of SL100 =  $62 \text{ mA} / (50+1) = 0.121 \text{ mA}$ .

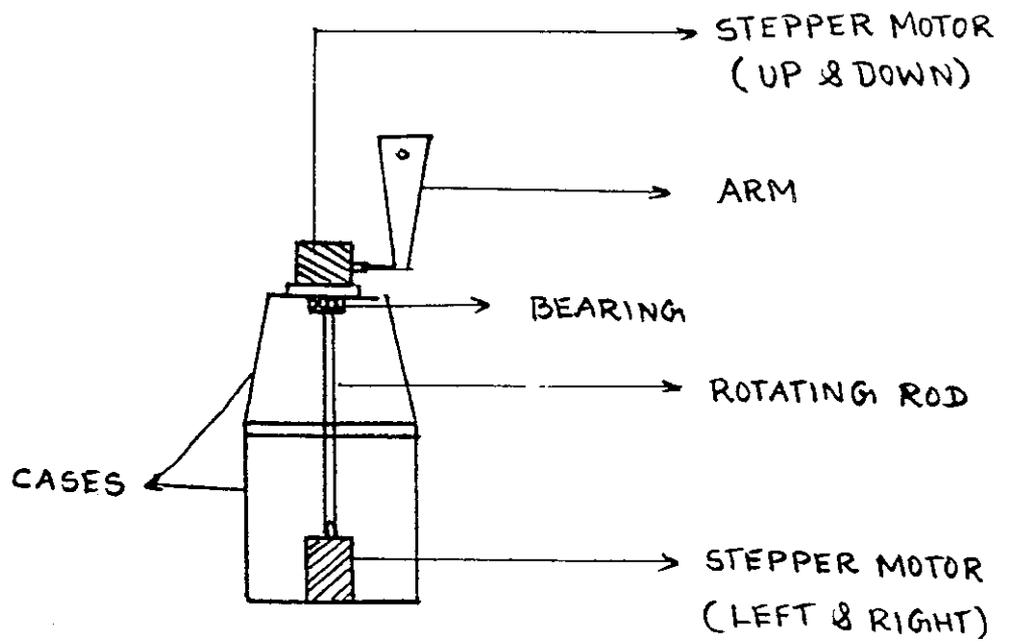
Since  $I_e = (h_{fe}+1) * I_b$ ,

Now  $V_{cc} - 2V_{be} = I_{b1} * R_1$

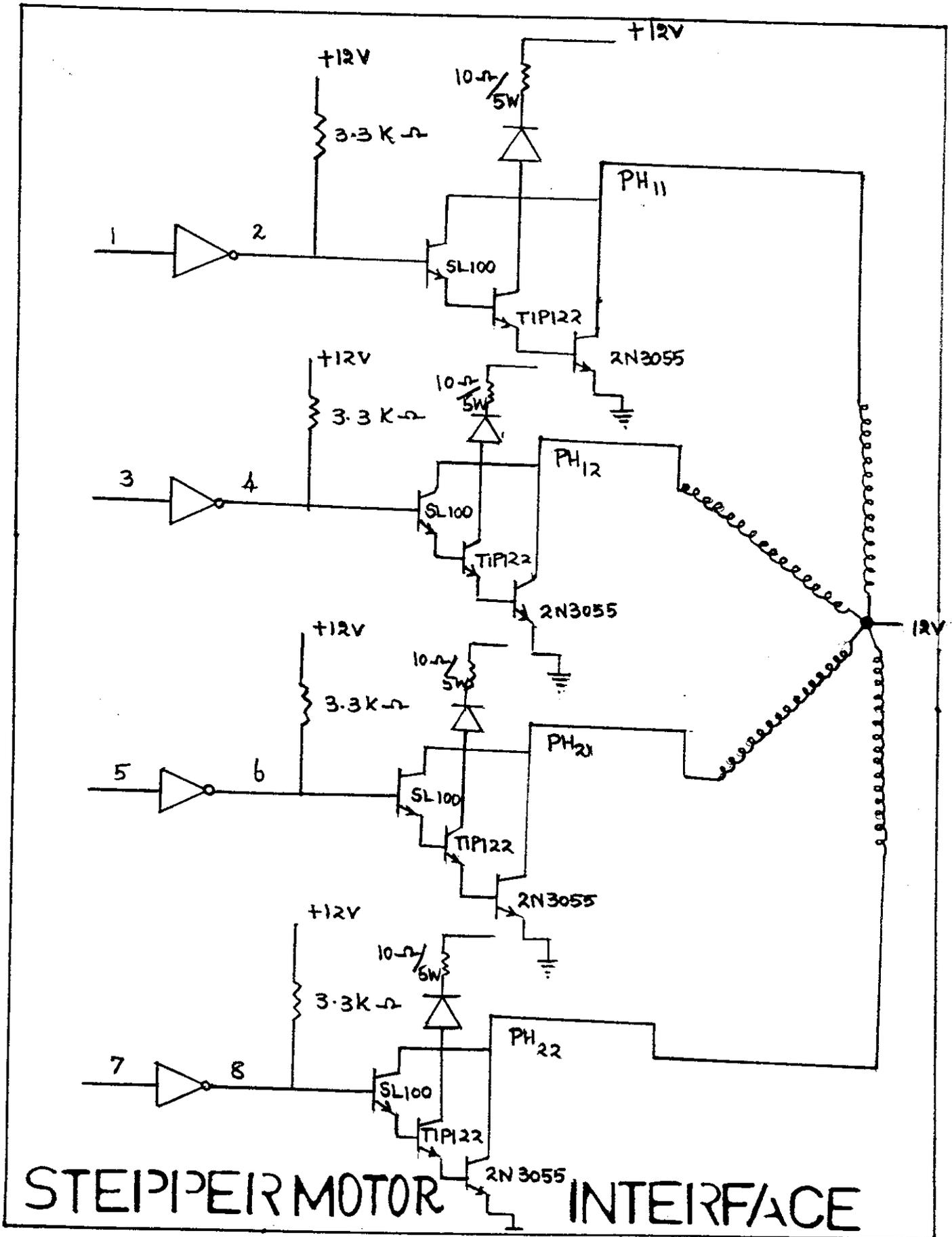
$R_1 = (4 - 2*0.7) / 0.121 \text{ mA} = 2.2 \text{ kilo ohms}$ .

#### **5.5.4 Robotic Arm**

The robotic arm, we have designed basically consists of two stepper motors. One for the left-right motion and the other for up-down motion. The first stepper motor which takes care of left-right motion of the arm forms the base (enclosed in a case). A rotating rod whose one end is fixed to the base motor, other end to the bearing is responsible for the rotary movement. The other motor which takes care of the up-down motion of the arm is placed horizontally above the base motor's case as shown in the figure. When the base motor is operated the left-right motion is achieved and the operation of the horizontally placed motor leads to the up-down motion.



# ROBOTIC ARM



## ***6. APPLICATIONS OF ROBOTS***

Robots are employed in a wide assortment of applications in industry and in other fields. The better quality of products achieved by robot operations is considered as the most important contribution of robots.

Some of the important industrial applications of robot are as follows:

- I. Material handling and machine loading and unloading operations: In these applications, the robots are used to move material from one part to another.
- II. Processing applications such as spot welding, arc welding and spray painting: The robot manipulates a tool to accomplish some manufacturing process.
- III. Assembly and inspection: Here the robots make use of sensors to gauge and measure quality characteristics of the manufactured product.

Non manufacturing applications include:

- Under sea operations.
- Coal mining.
- Space applications.
- Hospitals.
- Military.

## *7. CONCLUSION*

The robot developed by us is capable of recognizing four commands at present. This project is implemented as a minimal version, which could be upgraded to a machine capable of recognizing several commands from different people in any environmental conditions. This robot promises to be a very useful machine in the industrial field for voice-operated systems. From this project, we have been able to achieve a one-to-one correspondence between Man & Machine. Time, which is an important factor, is given much importance here. Nearly 50% of the time is saved through direct voice communication when compared to manual operations.

The constructional features of the robot shows its simplicity in design without sacrificing much of its precision.

The PIC16C57 has been used as the backbone of the robot with its powerful programming capabilities.

## ***BIBLIOGRAPHY***

1. *PIC16CXX APPLICATIONS HANDBOOK 1.0, PARALLAX INC.,*
2. *PIC16/17CXX DEVELOPOMENT TOOLS  
USING PARALLAX SOFTWARE AND HARDWARE TOOLS*
3. *THEORY AND APPLICATIONS OF DIGITAL SIGNAL PROCESSING -  
L.R.RABINER & BERNARD GOLD*
4. *DIGITAL CODING OF WAVE FORMS  
- JAYANTH & NOLL*
5. *ROBOTICS  
EDITED BY E.P.POPOV & E.I.YUREVICH*
6. *ROBOTIC ENGINEERING – AN INTEGRATED APPROACH  
- RICHARD D.KLAFTER., THOMAS.A. CHMIELEWSKI.,  
MICHAEL NEGIN*
7. *INDUSTRIAL CONTROL ELECTRONICS  
- JACOB*

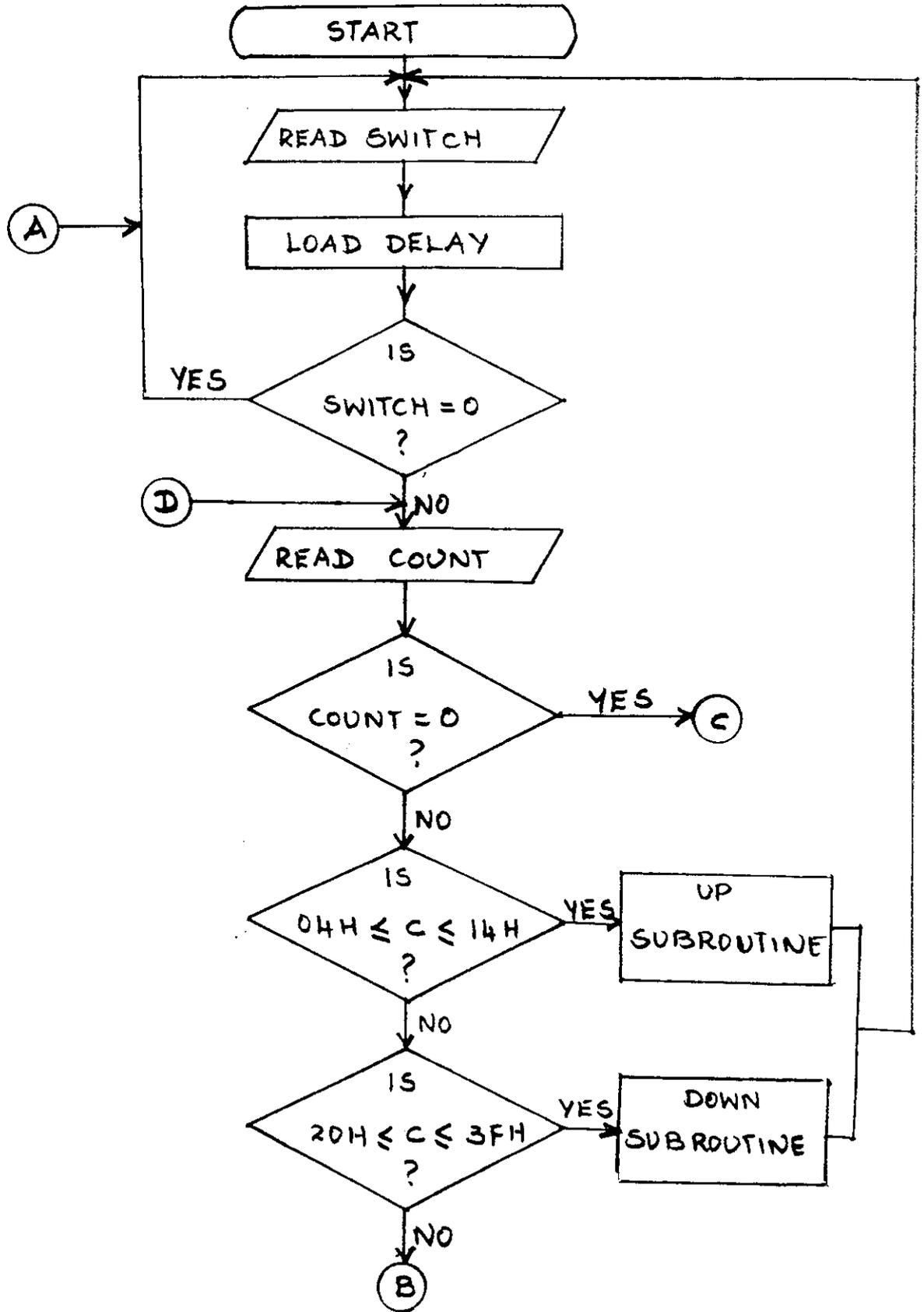
# PIC16Cxx Instruction Set

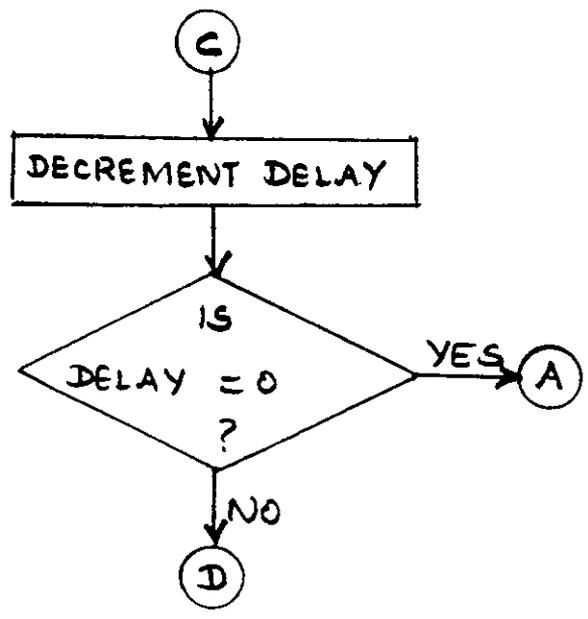
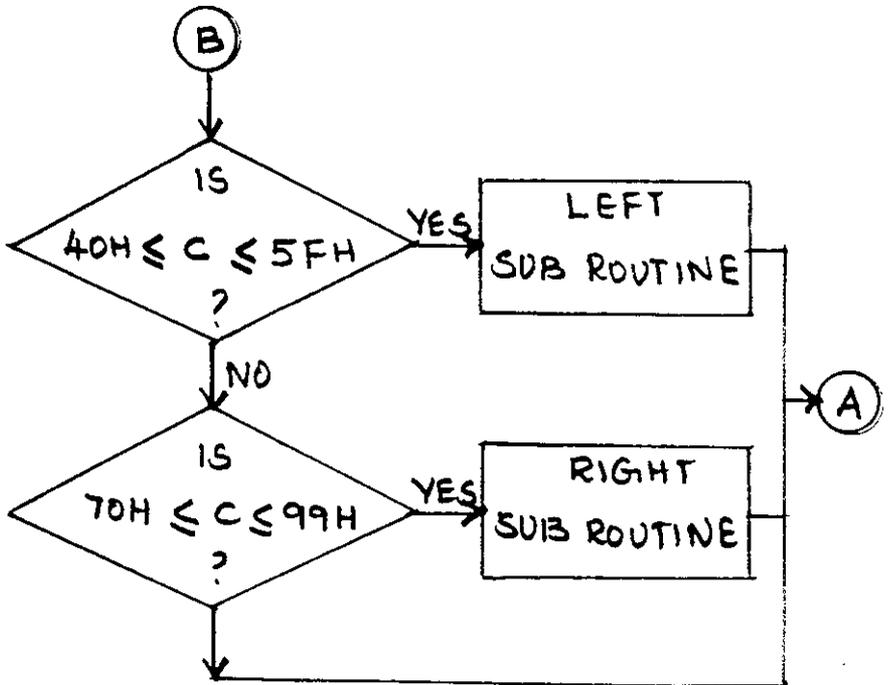
## Parallax Instruction Set

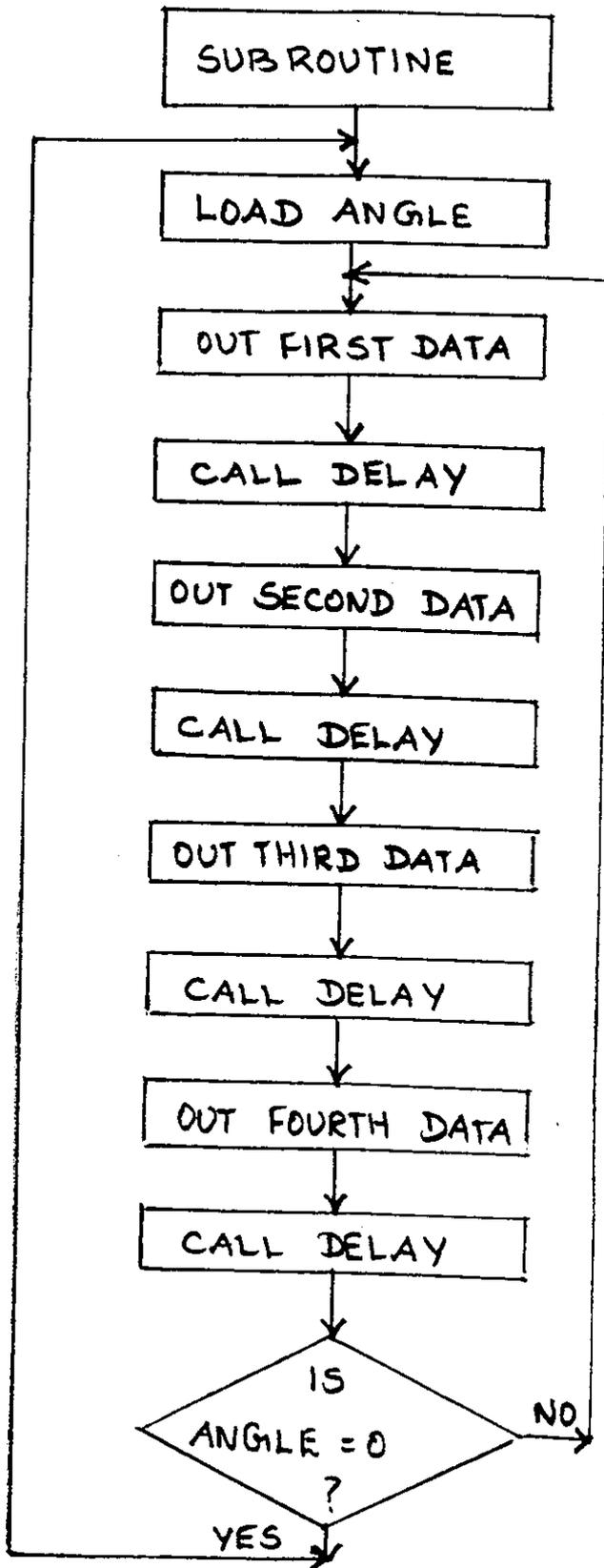
ADD	fr,#lit	CSNE	fr,#lit	MOVB	bit1,/bit2
ADD	fr1,fr2	CSNE	fr1,fr2	MOVSZ	W,++fr
ADD	fr,W	DEC	fr	MOVSZ	W,--fr
ADD	W,fr	DECSZ	fr	NEG*	fr
ADDB*	fr,bit	DJNZ	fr,addr9	NOP	
AND	fr,#lit	IJNZ	fr,addr9	NOT	fr
AND	fr1,fr2	INC	fr	NOT	W
AND	fr,W	INCSZ	fr	OR	fr,#lit
AND	W,#lit	JB	bit,addr9	OR	fr1,fr2
AND	W,fr	JC	addr9	OR	fr,W
CALL	addr8	JMP	addr9	OR	W,#lit
CJA	fr,#lit,addr9	JMP	PC+W	OR	W,fr
CJA	fr1,fr2,addr9	JMP	W	RET	
CJAE	fr,#lit,addr9	JNB	bit,addr9	RETW	lit,lit,...
CJAE	fr1,fr2,addr9	JNC	addr9	RL	fr
CJB	fr,#lit,addr9	JNZ	addr9	RR	fr
CJB	fr1,fr2,addr9	JZ	addr9	SB	bit
CJBE	fr,#lit,addr9	LCALL*	addr11	SC	
CJBE	fr1,fr2,addr9	LJMP*	addr11	SETB	bit
CJE	fr,#lit,addr9	LSET*	addr11	SKIP	
CJE	fr1,fr2,addr9	MOV	fr,#lit	SLEEP	
CJNE	fr,#lit,addr9	MOV	fr1,fr2	SNB	bit
CJNE	fr1,fr2,addr9	MOV	fr,W	SNC	
CLC		MOV	OPTION,#lit	SNZ	
CLR	fr	MOV	OPTION,fr	STC	
CLR	W	MOV	OPTION,W	STZ	
CLR	WDT	MOV	!port_fr,#lit	SUB	fr,#lit
CLRB	bit	MOV	!port_fr,fr	SUB	fr1,fr2
CLZ		MOV	!port_fr,W	SUB	fr,W
CSA	fr,#lit	MOV	W,#lit	SUBB*	fr,bit
CSA	fr1,fr2	MOV	W,fr	SWAP	fr
CSAE	fr,#lit	MOV	W,/fr	SZ	
CSAE	fr1,fr2	MOV	W,fr-W	TEST	fr
CSB	fr,#lit	MOV	W,++fr	XOR	fr,#lit
CSB	fr1,fr2	MOV	W,--fr	XOR	fr1,fr2
CSBE	fr,#lit	MOV	W,<<fr	XOR	fr,W
CSBE	fr1,fr2	MOV	W,>>fr	XOR	W,#lit
CSE	fr,#lit	MOV	W,<>fr	XOR	W,fr
CSE	fr1,fr2	MOVB	bit1,bit2		

\* These instructions are not available in PASMx.

'MAIN PROGRAM'







```
SWITCH EQU RA.0
COUNTER EQU RB.0
STEP EQU RC
COU EQU 08H
DLY0 EQU 09H
DLY1 EQU 0AH
DLY2 EQU 0BH
COU1 EQU 0CH
```

```
DEVICE PIC16C57,XT_OSC,WDT_OFF
```

```
RESET START
```

```
START MOV !RA,#0000 0001B
      MOV !RB,#0000 0001B
      MOV !RC,#0000 0000B
      CLR COU
      CLR COUNTER
      CLR SWITCH
      CLR STEP
      CLR DLY0
      CLR DLY1
      CLR DLY2
      CLR COU
      CLR COU1
      MOV STEP,#0000 0001B
```

```

CHK      JB SWITCH, CHK
        CALL DELAY
        CALL DELAY
CHK1     JB COUNTER,CHK1
        MOV DLY0,#1111 1111B
AD1      MOV DLY1,#1111 1111B
AD2      JB COUNTER,AD3
        DJNZ DLY1,AD2
        DJNZ DLY0,AD1
        JMP AD4
AD3      INC COU
        MOV STEP,COU
        JMP CHK1
AD4      MOV STEP,#0000 0001B
CHK3     JB SWITCH,CHK3
        CALL DELAY
        CALL DELAY
        CJE COU,#4,STEP1
        JNC XX
        CJE COU,#22,STEP1
        JNC STEP1
        CJE COU,#32,STEP2
        JNC XX
        CJE COU,#63,STEP2
        JNC STEP2

```

```
      CJE COU, #80, STEP3
      JNC XX
      CJE COU, #101, STEP3
      JNC STEP3
      CJE COU, #112, STEP4
      JNC XX
      CJE COU, #153, STEP4
      JNC STEP4
XX     MOV STEP, #1111 1111B
      CALL DELAY
CHK4   JB SWITCH, CHK4
      JMP CHK1
DELAY  MOV DLY1, #1000 0000B
XXZ    MOV DLY2, #1000 0000B
XXJ    DJNZ DLY2, XXJ
      DJNZ DLY1, XXZ
      RET
STEP1  MOV DLY0, #0001 1001B
MO1    JMP MOTF
S1     DJNZ DLY0, MO1
      CLR COU
      JMP XX
STEP2  MOV DLY0, #0001 1001B
MO2    JMP MOTF1
```

```
S2      DJNZ DLY0,MO2
        CLR COU
        JMP XX

STEP3   MOV DLY0,#0110 0100B
MO3     JMP MOTF2

S3      DJNZ DLY0,MO3
        CLR COU
        JMP XX

STEP4   MOV DLY0,#0110 0100B
MO4     JMP MOTF3

S4      DJNZ DLY0,MO4
        CLR COU
        JMP XX

MOTF    MOV STEP,#1111 1010B
        CALL DELAY
        MOV STEP,#1111 0110B
        CALL DELAY
        MOV STEP,#1111 0101B
        CALL DELAY
        MOV STEP,#1111 1001B
        CALL DELAY
        JMP S1

MOTF1   MOV STEP,#1111 1001B
        CALL DELAY
```

```
MOV STEP,#1111 0101B
CALL DELAY
MOV STEP,#1111 0110B
CALL DELAY
MOV STEP,#1111 1010B
CALL DELAY
JMP S2
MOTF2 MOV STEP,#1010 1111B
CALL DELAY
MOV STEP,#0110 1111B
CALL DELAY
MOV STEP,#0101 1111B
CALL DELAY
MOV STEP,#1001 1111B
CALL DELAY
JMP S3
MOTF3 MOV STEP,#1001 1111B
CALL DELAY
MOV STEP,#0101 1111B
CALL DELAY
MOV STEP,#0110 1111B
CALL DELAY
MOV STEP,#1010 1111B
CALL DELAY
JMP S4
```