PC BASED VOICE CONTROLLED ROBOT

PROJECT REPOR

SUBMITTED B

M. PADMARA

M. PRASA

V. RAGAVENDARA

K. S. SRINIVASA RAGHAVA

GUIDED B

Mrs. R. LATHA, B.

IN PARTIAL FULFILMENT OF THE REQUIREMEN

FOR THE AWARD OF THE DEGREE O

BACHELOR OF ENGINEERING I

ELECTRONICS & COMMUNICATION ENGINEERIN

OF THE BHARATHIAR UNIVERSI

2000 - 2001

Department of Electronics & Communication Engineering

Kumaraguru College of Technology

Coimbatore - 641 006

Kumaraguru College of Technology

Coimbatore - 641006

Department of Electronics and Communication Engineering

Certificate

This is to certify that this project entitled

PC Based Voice Controlled Robot

has been submitted by

M. PRASAD, M. PADMARAJ, V. RAGAVENDRAN, K.S. SRINIVASA RAGHAVAN

In partial fulfillment of the requirements for the award of Degree of Bachelor of Engineering in the Electronics and Communication Engineering Branch of the Bharathiar University, Coimbatore - 641046 during the Academic Year 2000-01

R. Lak
9/3/01

(Guide)

9/3

(Head of Department)

Certified that the candidate was examined by us in the Project Work Viva-Voce Examination held on 12/3/2001.

University Register Number

9727D0222, 9727D0218, 9727D0224, 9727D0235

12/3

(Internal Examiner)

12-03-2k1

(External Examiner)

ACKNOWLEDGEMENTS

We express our sincere thanks to our beloved Principal **Dr.K.K.Padmanabhan** B.Sc(Engg), M.Tech., Ph.D for all the facilities provided in carrying out this project.

We also take privilege to thank **Prof.M.Ramaswamy** M.E., MIEEE(USA), MIE, MISTE, C.Engg(I), MBMESI, Head of Department of Electronics and Communication Engineering for his constant encouragement throughout the completion of the project.

We also express our gratitude and indebtedness to our respected guide **Mrs.R.Latha** B.E. for her full-fledged technical guidance and suggestions in carrying out this project.

We would also like to thank our Senior Lecturer **Mr.Thiagarajan** for lending a helping hand towards the completion of our project. We would also like to thank **Mr.Raja** B.E for his external guidance for completing this project.

Finally we thank all the faculty members of ECE Department, our friends and our parents for the valuable help rendered to us.

CONTENTS

1.SYNOPSIS

In a world progressing with information super highway, our lives have become sophisticated and not to mention, comfortable. This has been largely due to computers and robotics. Robots are entering into nearly all walks of life. They can be used in many applications right from doing household chores to inspection and testing in Industrial units.

Our project deals with the control of a robot by means of voice command .The core idea is the transformation of human voice command into a discrete set of digital pulses, which in turn is processed and finally instructs the ROBOT to perform a specific action .The arm of the ROBOT is connected with the stepper motor. The stepper motor is driven by means of a stepper card, a special card that controls the operation of the stepper motor. As the motor rotates, the arm is set into motion in accordance with the command.

The processes are controlled by a PC, which is interfaced with the robot with the support of peripheral device like 8255A. The use of a PC makes the control and operations more flexible and versatile.

2.INTRODUCTION

2.1 WHAT IS A ROBOT ?

When we talk about ROBOTS, many people will think of those machines in the science fiction films. They have a man-like shape, behave like a human being and speak understandable languages.

The word Robot was first devised by Czech author Karel Capek and simply meant work or servitude. Nowadays, this is a pretty apt description of most Robots. In manufacturing industries, Robots are used to perform repeated and tedious jobs. They work with high accuracy and without complaints. In exploration, Robots are sent to hazardous areas, to deep sea, or even other planets to perform the jobs which no human could easily do.

2.2 CLASSIFICATION OF ROBOTS

The Robots can be classified according to the types of control, capabilities, configuration and mobility.

2.2.1 Based on types of control

- *Point – to- Point* Robots move from one point to another but cannot stop at arbitrary intermediate points.
- *Continuous point* Robots can move to a prescribed number of points along a path and can stop at arbitrary intermediate points.
- *Computed trajectory* Robots can move along the path specified algebraically.
- *Servo-Controlled* Robots have some means of sensing the current position and feeding this sensed position back so that a prescribed path can be followed by them.

2.2.2 Based on capability

- *Sequence-controlled* Robots are machines which go through a fixed sequence of actions according to the instructions.
- *Computer trajectory* Robots follow a specified path between the starting and finishing points. Some numerically – controlled machines are of this type.
- *Adaptive* Robots can react to their environment using their sensors. The performance is optimized by adjusting controls to changing parameters.
- *Intelligent* Robots are provided with sensors to study and model the environment, thereby generating a knowledge base. They perform with the help of an expert system provided along with the Robot. The knowledge base is constantly updated so as to better the performance with the passage of time.

2.2.3 Based on configuration

- A robot may articulate it's joints in the cartesian co-ordinate system, in which case, it is known as *Cartesian* Robot.
- A *Cylindrical* Robot works with the cylindrical co-ordinate system.

2.2.4 Based on mobility

- Robots may be fixed in the workspace or mounted on a short track. These are called as robots of *Fixed* type.
- *Mobile* Robots are mounted on wheels, powered by batteries and guided by non-contact type tracks. Sometimes, they are guided by personnel using a tele-operator.
- *Walking* Robots have legs which can move about in difficult terrains. Some have the ability to climb up and down the slopes and staircases. These Robots are powered by batteries or engines, and are operated remotely via a radio link by the computer.

Our Robot comes under the category of *Fixed Robot*

3. STEPPER MOTOR

3.1 INTRODUCTION

Stepper motors are being widely used in simple position control systems in the open loop and closed loop modes. A digital computer or a microprocessor can directly drive a stepper motor using binary numbers. The stepper motor exhibits the characteristics of a synchronous motor. It produces a steady torque at a given speed.

3.2 CONSTRUCTIONAL FEATURES

A stepper motor could be either of the reluctance type or of the permanent magnet(PM) type. A PM stepper motor has salient magnetic poles and a reluctance stepper motor has unmagnetized salient poles. The basic two-phase stepper motor consists of two pairs of stator poles. Each of the four poles has it's own winding. The excitation of any one winding generates a north pole(N) and a south pole(S) gets induced at the diametrically opposite side.

The four pole structure is continuous with the stator frame and the magnetic field passes through the cylindrical stator annular ring. The rotor magnetic system has two end faces. The left face is permanently magnetized as 'south pole' and right face a 'north pole'. The south pole structure may have three or five pole faces and north pole structure possesses similar pole faces. The north pole structure is twisted with respect to south pole structure so that a south pole comes precisely between two north poles.

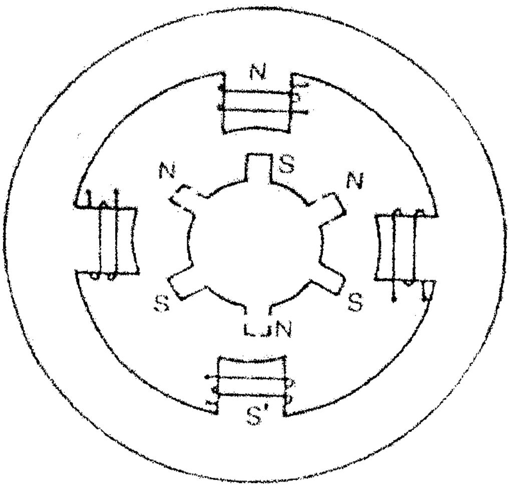


Fig 3.1 Stepper Motor – Cross-sectional view

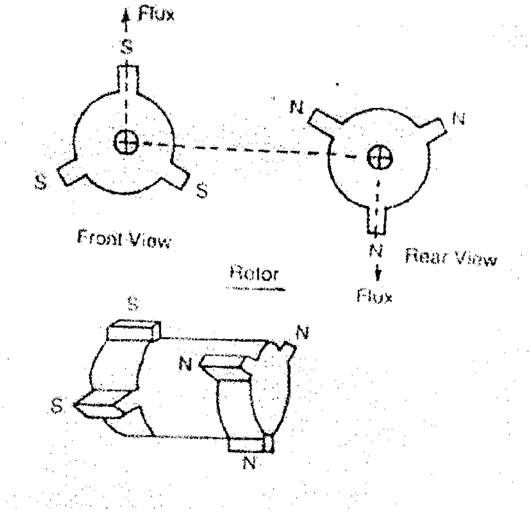


Fig 3.2 Typical Stepper Motor

3.3 PARAMETERS OF STEPPER MOTOR

Certain parameters of stepper motors are defined in this section.

1. Step angle : This refers to the angle through which the motor moves for one step.

2. Stepping rate: The number of steps executed per second is known as stepping rate. If the stepping pulses are given at a rate greater than the maximum allowable rate, then the motor would miss many steps.
3. Holding Torque: This is equal to the external torque which must be supplied to break away from its equilibrium position, when a stator winding remains excited.
4. Dynamic Torque: The torque produced by the stepper motor under a constant stepping rate is known as the dynamic torque.
5. Detent torque: This is the torque required to over come the residual magnetism and reluctance torque under the unexcited condition. This torque enables the motor to hold a load even when the stator windings are de-energized.
6. Holding torque/Inertia ratio: High performance motors have a large holding torque or holding ratio. Such motors have a length greater than the diameter.
7. Step response: When a single step is executed by the motor, the rotor exhibits a decaying oscillatory response. Mechanical dampers may be employed to reduce the settling time.
8. Ramping: The process of controlling the switching frequency or stepping rate to accelerate a motor from zero to full speed as well as to decelerate it from maximum speed to zero without losing steps is called ramping.
9. Step size: The north pole structure is offset with respect to the south pole structure by one pole pitch. In an arrangement with four stator poles and three pairs of rotor poles, there exists twelve possible stable positions in which a south pole of the rotor can lock with the north pole of stator. It is clear that the step size is (360/12) = 30 degrees (mechanical).

The two phase scheme is often used for 'stepping' a stepper motor which is explained in the next section.

P - 1376

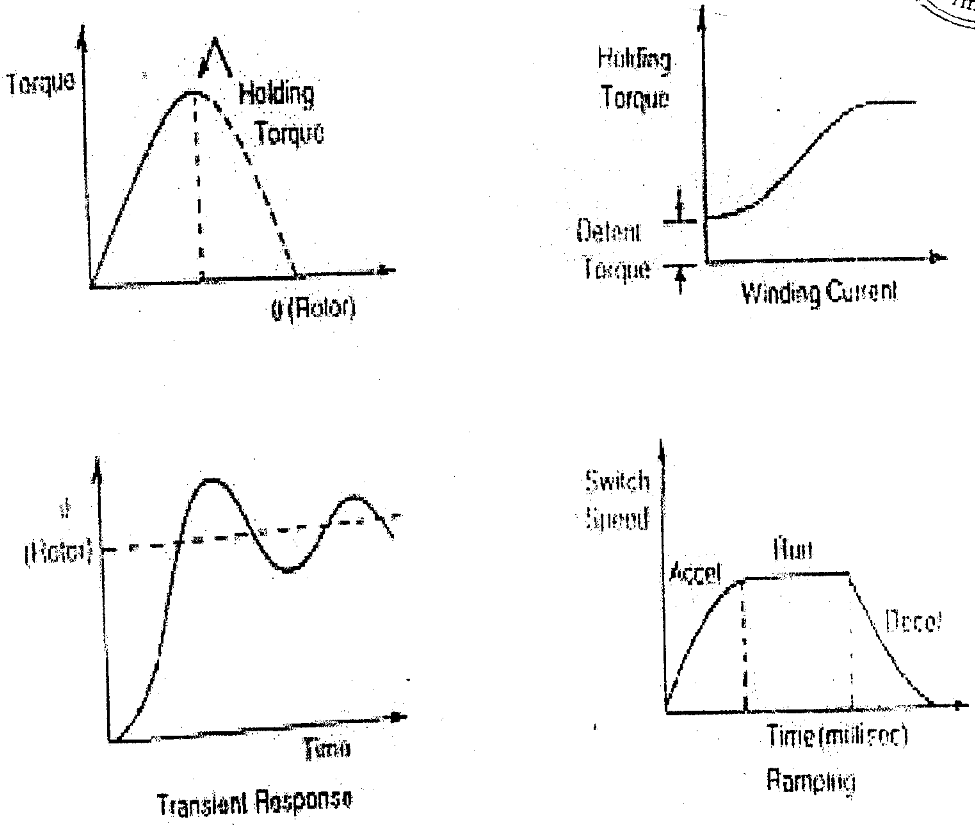


Fig 3.3 Characteristics of Stepper Motor

3.4 TWO-PHASE SCHEME:

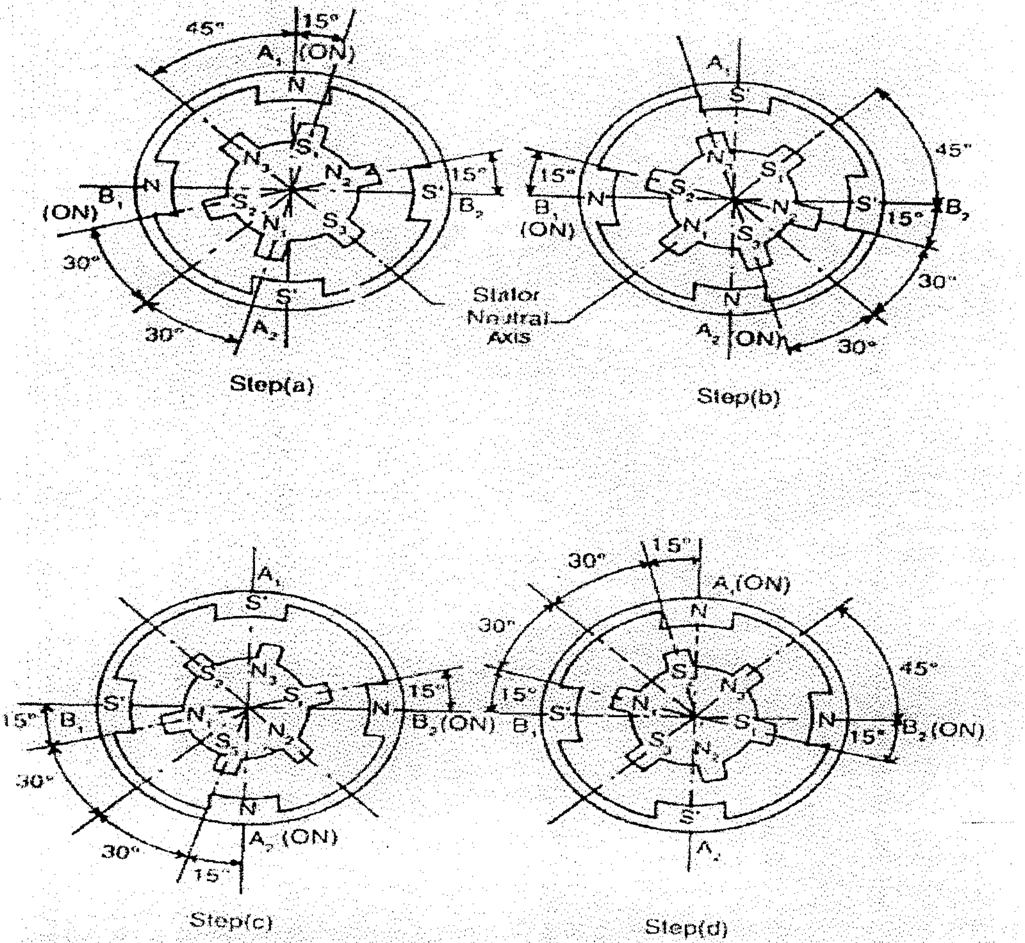


Fig 3.4 Two-phase scheme of a Stepper Motor

In this scheme, any two adjacent stator windings are energized. There are two magnetic fields active in quadrature and none of the rotor pole faces can be in direct alignment with the stator poles. A partial but symmetric alignment of rotor poles is of course possible .

Typical equilibrium conditions of the rotor when the windings on successive stator poles are excited are illustrated in figure 3.4. In step(a) A1 & B1 are energized.

The pole-face S1 tries to align itself with the axis of A1(N) and the pole-face S2 with B1(N) . The north pole N3 of the rotor finds itself in the neutral zone between A1(N) & B1(N). S1 & S2 of the rotor, position themselves symmetrically with respect to the two stator north poles.

Next, when B1 & A2 are energized, S2 tends to align with B1(N) & S3 with A2(N). Of course again under equilibrium conditions, only partial alignment is possible and N1 finds itself in the neutral region midway between B1(N) & A2(N) [step (b)] . In step(c), A2 and B2 are on. S3 & S1 tend to align with A2(N) & B2(N), respectively, with N2 in the neutral zone. Step (d) illustrates the case when A1 & B2 are on.

A total of twelve steps are required to move the rotor by 360 degrees(mechanical). The switching sequence is given in the table below.

Anticlockwise						Clockwise					
Step	A1	A2	B1	B2	Hex	Step	A1	A2	B1	B2	Hex
1	1	0	0	1	09	1	1	0	1	0	0A
2	0	1	0	1	05	2	0	1	1	0	06
3	0	1	1	0	06	3	0	1	0	1	05
4	1	0	1	0	0A	4	1	0	0	1	09

3.5 NEED FOR STEPPER MOTOR:

The increasing trend towards digital control of the machines and process functions has generated a demand for mechanical devices capable of delivering incremental motion of predictable accuracy. The answer to this question was found with advent of stepper motors. The repeatability of these motors is very good. The only system error introduced is its single step error, which is a small percentage of one step and is generally less than 5%. The actual rotational movements or step angles of shaft are from 1.8 to 90 degrees depending on the particular motor chosen. Our Robot employs stepper motors which have a step angle of 1.8 degrees. A stream of 200 pulses will give an angular displacement of 360 degrees or one complete revolution.

4. INTERFACING

4.1 WHY PC?

The personal computer (PC), revolution has created an insatiable market for faster, easier-to-use, and more powerful components and peripherals. Advances in technology have led to multimedia-capable computers that can listen to human voice and can control some devices. Hence we decided to go for a PC based voice-controlled Robot.

Every application requires that the robot interact with something in the execution of its programmed task. Interfacing the robot and related equipment involves the transmission of information in two directions.

Applying a robot involves an interface between the robot and the application depending on the limitations within the robot. This interface could be a mere start/stop contact or it could be a data communications channel with an external device or high-level compiler. A computer-controlled robot allows for the full spectrum of interfacing from switch closure to data communication and beyond. It is very flexible and lots of applications may be framed easily to cater our needs. Voice input is becoming a more common form of medical and legal dictation.Voice controlled PC can find a wide range of applications in the field of medicine, defense etc.

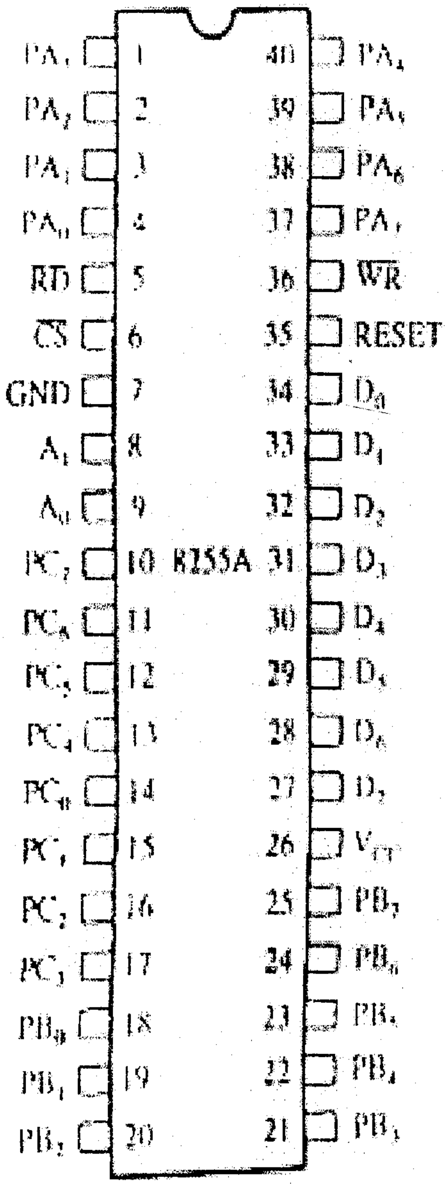
4.2 I/O CARD DESCRIPTION

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

The 8255A has 24 I/O pins that can be grouped primarily in two 8-bit parallel ports, A and B, with the remaining 8 bits as port C. The eight bits of port C can be used as individual bits or be grouped in two 4-bit ports : C_{upper} and C_{lower}. The functions of these ports are defined by writing a control word in the control register.

4.3 BLOCK DIAGRAM OF THE 8255A

Figure 4.1 shows the pin diagram of 8255A along with the pin details. The block diagram in fig 4.2 shows two 8-bit ports(A & B), two 4-bit ports(C_u and C_l), the data bus buffer, the control logic. Figure 4.3 shows a simplified but expanded version of the internal structure, including a control register. This block diagram includes all the elements of a programmable device ; port C performs functions similar to that of the status register in addition to providing handshake signals. There are 8 data lines for writing data bytes to a port or control register and for reading bytes from a port or status register under the control of the RD(low) and WR(low) lines.



Pin Names

D_7-D_0	Data Bus (Bidirectional)
RESET	Reset Input
$\overline{CS}$	Chip Select
$\overline{RD}$	Read Input
$\overline{WR}$	Write Input
A_0, A_1	Port Address
PA_7-PA_0	Port A (Bit)
PB_7-PB_0	Port B (Bit)
PC_7-PC_0	Port C (Bit)
V_{CC}	+ 5 Volts
GND	0 Volts

Fig 4.1 Pin diagram of 8255A

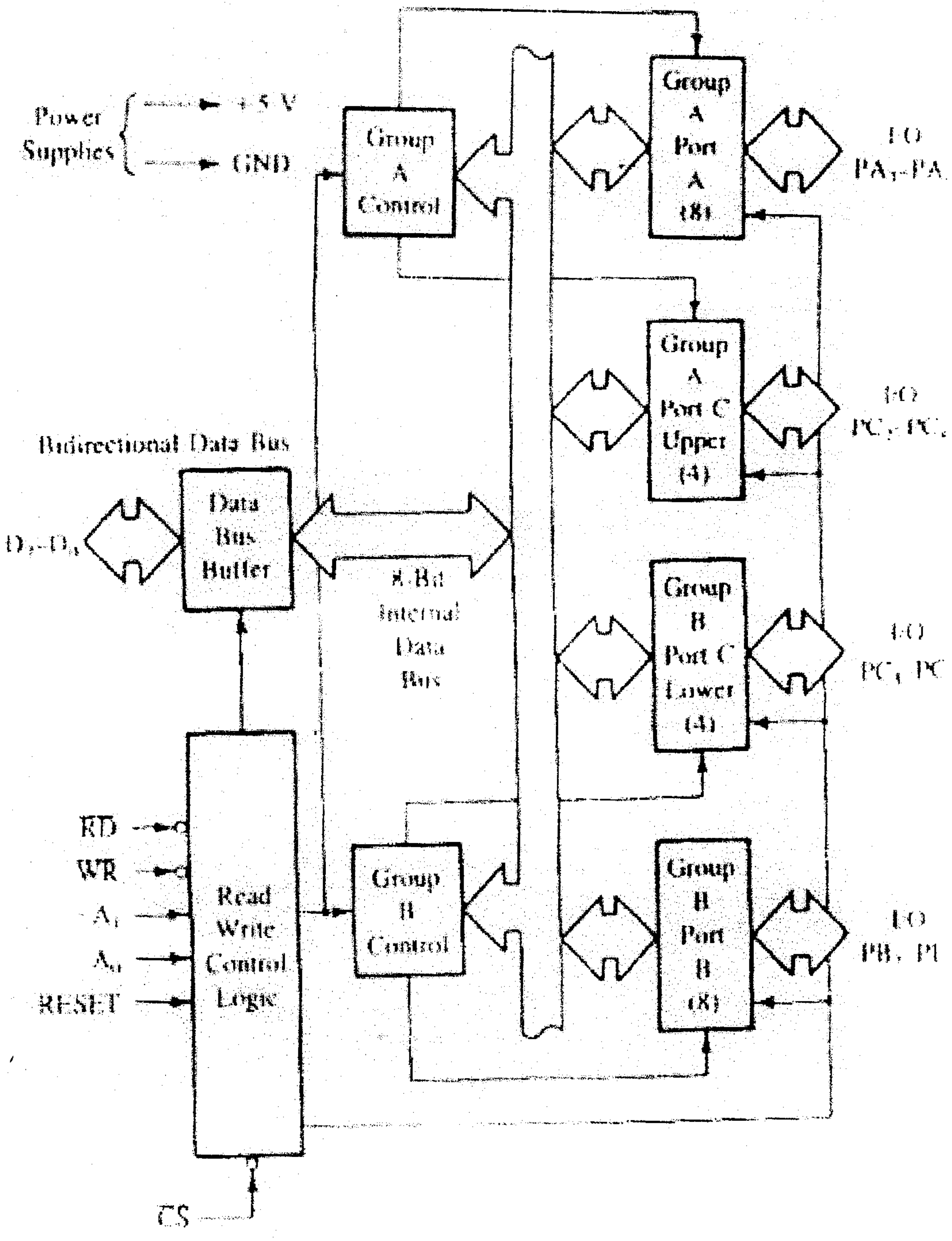


Fig 4.2 Block Diagram of 8255A

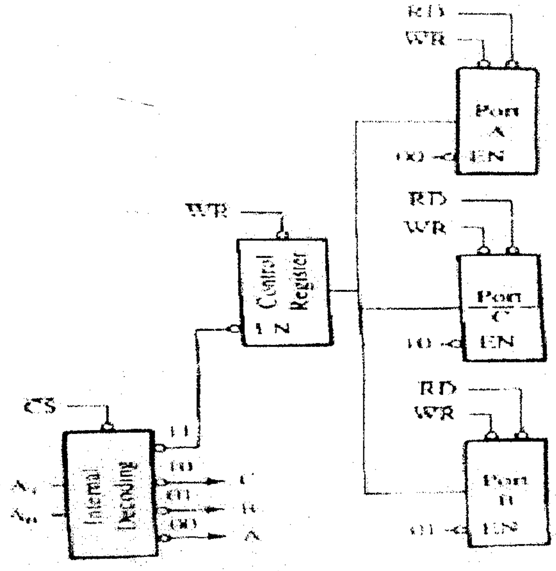


Fig 4.3 Expanded diagram of control logic and I/O ports

4.4 I/O MODES IN 8255A

There are basically three I/O modes in 8255A which are explained below.

MODE 0 : When you want to use a port for simple input or output without handshaking, you initialize that port in mode 0. If both port A and port B are initialized in mode 0, then the two halves of port C can be used together as a additional 8-bit port, or they can be used individually as two 4-bit ports.

MODE 1: When you want to use port A or port B for handshake (Strobed) input or output operation, you initialize that port in mode1. In this mode, some of the pins of port C function as handshake lines. In the handshake mode, two types of I/O data transfer can be implemented. : Status check and interrupt.

MODE 2 : In this mode, port A can be setup for bidirectional data transfer using handshake signals from port C, and port B can be setup either in mode 0 or mode 1.

4.5 CONTROL LOGIC AND CONTROL WORD

The CS(low) signal is the master chip select, A_o and A_1 specify one of the I/O ports or the control register as given below:

CS(low)	A_1	A_0	Selected
0	0	0	Port A
0	0	1	Port B
0	1	0	Port C
0	1	1	Control Register
1	x	x	8255A not selected

Figure 4.4 shows a register called the control register. The contents of this register, called the control word, specify an I/O function for each port. This register can be accessed to write a control word when A_0 and A_1 are at logic 1.

74LS138 is a widely used 1-out-of-8 binary decoder. It has three input lines and eight active low output lines. It requires three enable inputs : two are active low and one is active high.

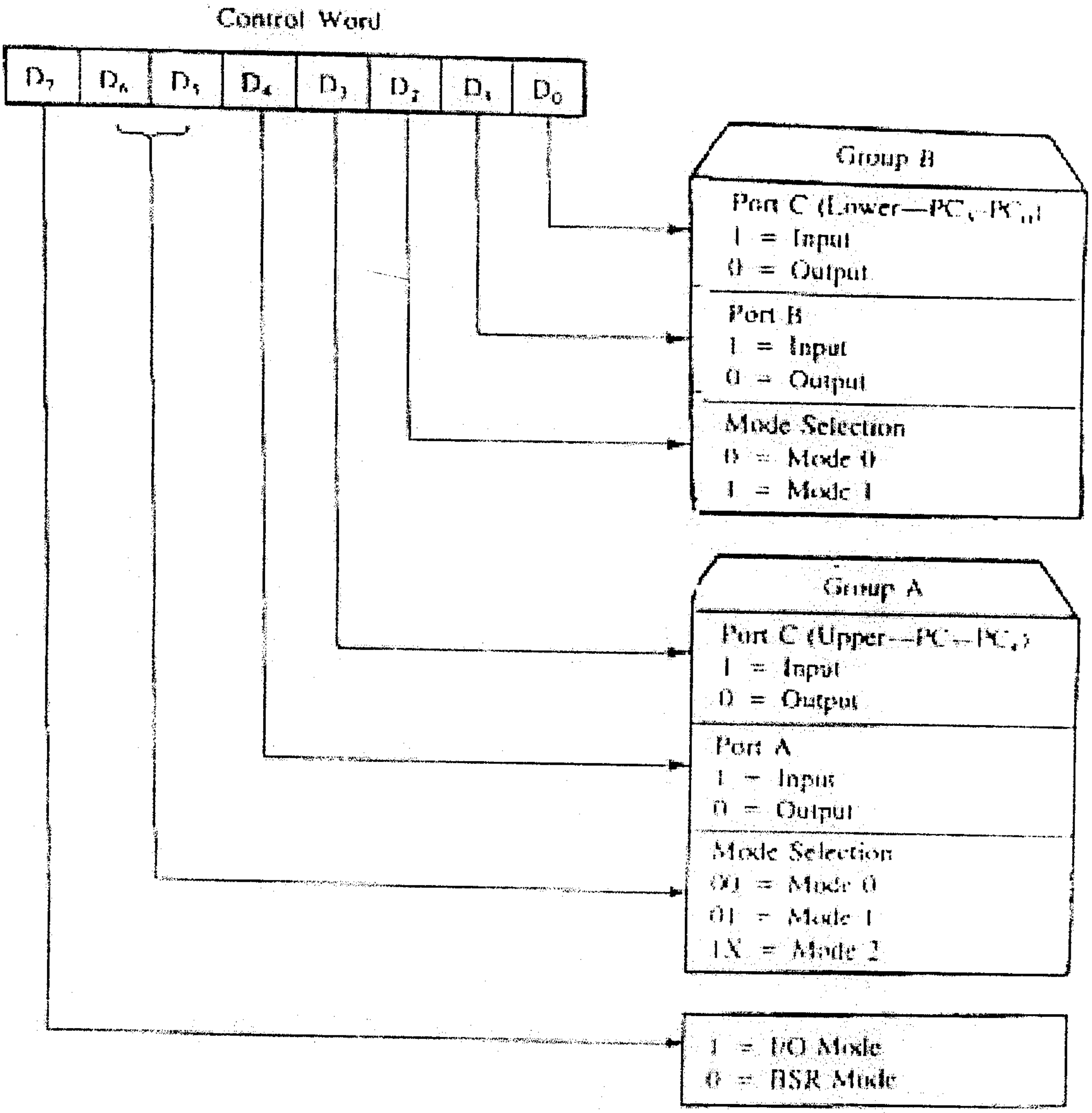


Fig 4.4 Control Word Format of 8255A

4.6 DECODERS

The decoder is a logic circuit that identifies each combination of the signals present at its input. For example, if the input to a decoder has two binary lines, the decoder will have four output lines. The two lines can assume four combinations of input lines-00,01,10,11-with each combination identified by the output lines 0 to 3. If the input is 11, the output line 3 will be at logic 1 and others will remain at logic 0. This is called decoding. A decoder is a commonly used device in interfacing I/O peripherals and memory.

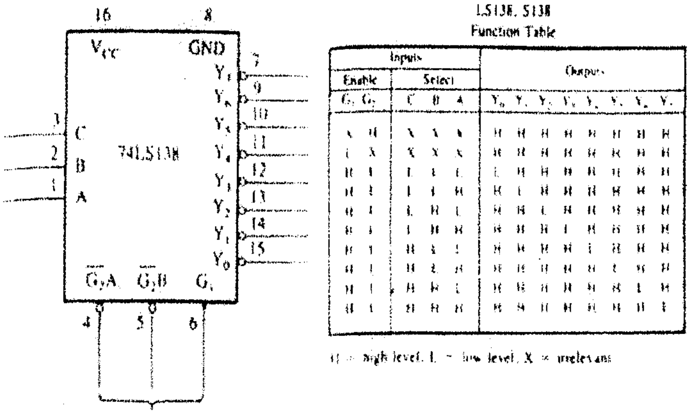
Here in our project we use 74LS138 Decoder.

The main features of 74LS138 are:

- Demultiplexing capability.
- Multiple input enable for easy expansion.
- Ideal for memory chip select decoding.
- Direct replacement for Intel 3205.

The 74LS138 decoder accepts three binary weighted inputs (A_0,A_1,A_2) and when enabled, provides eight mutually exclusive, active LOW outputs(0 – 7). The device features three enable Inputs : two active LOW (E_1,E_2) and one active HIGH (E_3). Every output will be high unless E_1 and E_2 are LOW and E_3 is HIGH. This multiple enable function allows easy parallel expansion of the device to a 1-of-32 (5 lines to 32 lines) decoder with just four '138s and one inverter.

The device can be used as an eight output demultiplexer by using one of the active LOW Enable Inputs as the Data Input and the remaining Enable Inputs as strobes. Enable Inputs not used must be permanently tied to their appropriate active HIGH or active LOW state.



LS138, S138
Function Table

Inputs					Outputs							
Enable		Select										
G_1	G_2	C	B	A	Y_0	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

H = high level, L = low level, X = irrelevant

Fig 4.4 74LS138 Details

4.7 I/O CARD ADDRESSES

74LS138 is used for address selection. 8255A is used for I/O operations as already described. The Addresses used are as follows:

Port A	-	220 H Output(PA_0 to PA_7)
Port B	-	221 H Unused
Port C	-	222 H Input(PC
Control Register	-	223 H

4.8 BIDIRECTIONAL BUFFERS(TRANSCEIVERS) :

Bi-directional buffers are used in the I/O card to enable data transfer between PC and 8255 and vice versa.

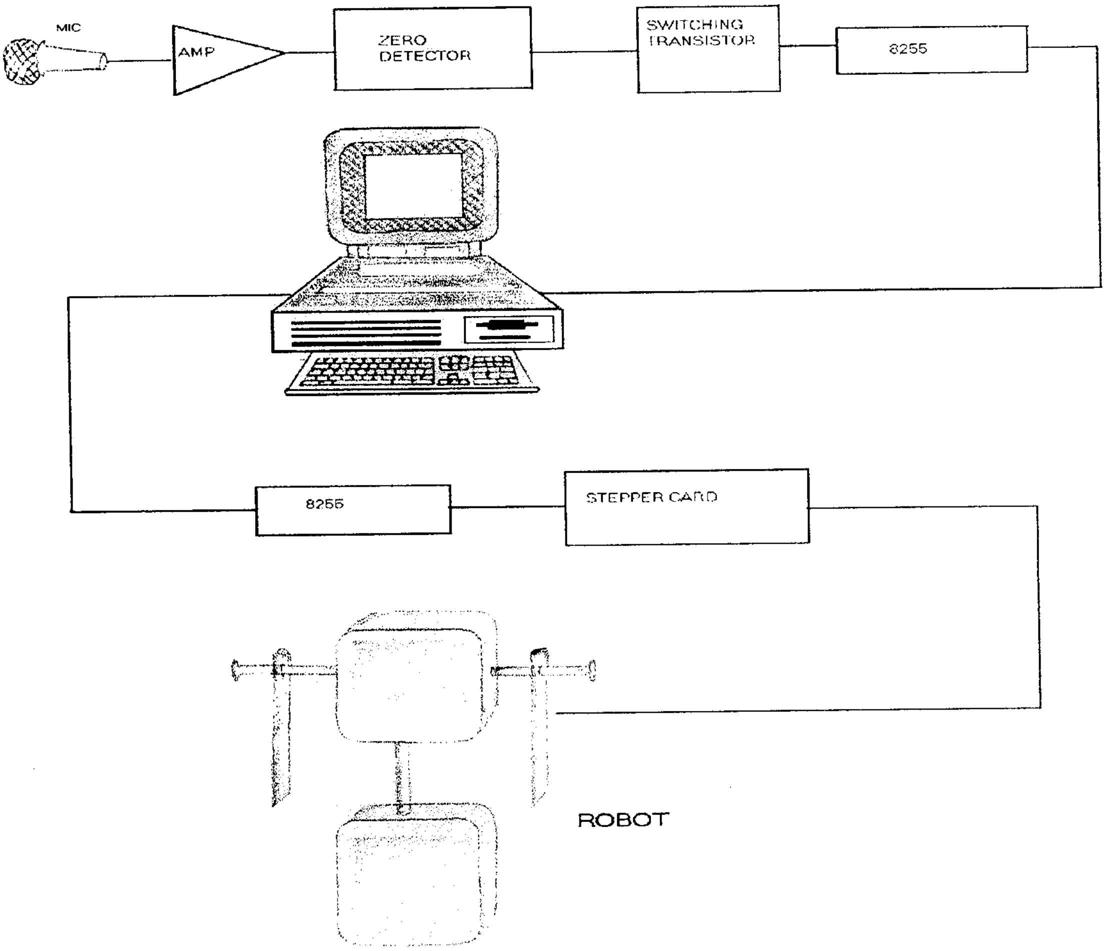
In our project we use 74LS245 transceiver. The main features of this transceiver are:

- Octal bi-directional bus interface.
- 3-state buffer outputs.
- PNP inputs for reduced loading.
- Hysteresis on all data inputs.

The 74LS245 is an octal transceiver featuring non – inverting 3-state bus compatible outputs in both send and receive directions. The outputs are all capable of sinking 24mA and sourcing up to 15mA, producing very good capacitive drive characterisitics. The device features a Chip Enable (CE) input for easy cascading and a Send / Receive (S/R) input for direction control. All data inputs have hysteresis built in to minimize AC noise effects.

5.DESCRIPTION OF HARDWARE

5.1 BLOCK DIAGRAM OF 'PC BASED VOICE CONTROLLED ROBOT'



In fulfilling the objective of transforming the human voice into action, the major requirement is to convert the voice into an electrical signal. So this first step is accomplished with the help of a carbon microphone. The microphone used in the project has impedance of 600 Ω and is dynamic. When the user gives a command, the sound wave impinges on the carbon granules present inside the microphone. These carbon granules undergo a change in its resistance and cause the current to flow through it. The magnitude of the current is proportional to the intensity of the sound waves impinging on the carbon granules.

The output of the carbon microphone is of the order of a few milliwatts. This power is not sufficient to drive other circuits. So it is amplified by means of Common Emitter (CE) amplifiers. Two stages of CE amplifiers are employed for this purpose. The output of the power amplifier is a sine wave with high positive and negative peaks. The sine wave cannot be applied as it is to the port of a programmable peripheral interface. The ports of a Programmable Peripheral Interface (PPI) are based on TTL logic. Hence it can distinguish the input to be logic 0 or logic 1 only if it lies within a certain voltage range. So the sine waves must be converted to pulses by means of a zero crossing detector (i.e. a Comparator).

A comparator is a circuit, which compares the signal voltage applied at one input of an operational amplifier with a known reference voltage at the other input. It is basically an open loop operational amplifier with output $\pm V_{sat}$ ($= V_{cc}$) with the ideal transfer characteristics as shown in fig (a).

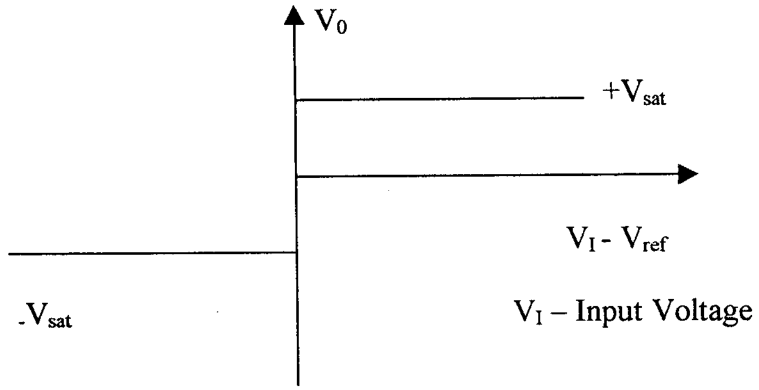


Fig (a) Transfer Characteristics

There are basically two types of comparators

1) Non- Inverting comparator

2) Inverting comparator

In our project, Zero crossing detector is actually an Inverting comparator.

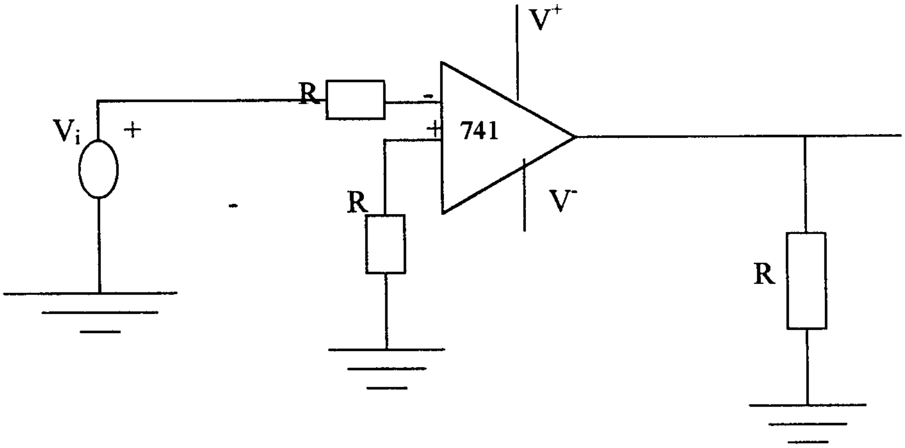


Fig (b) ZERO CROSSING DETECTOR

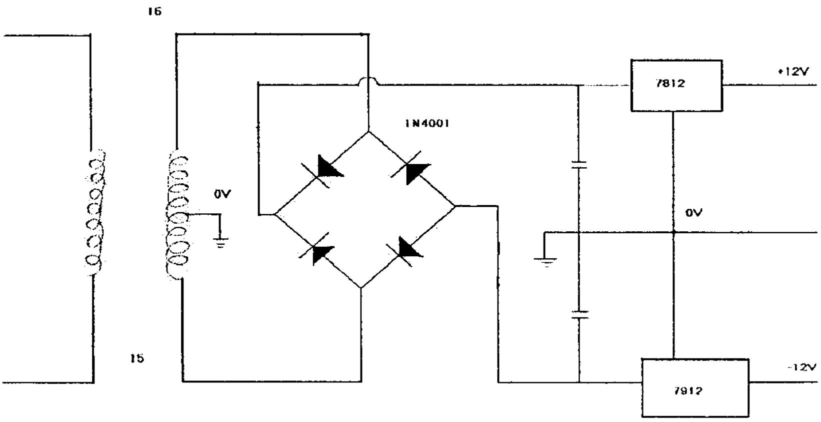
A fixed reference voltage of zero volts is applied to the non-inverting terminal and a time varying input which is the output of the amplifier stage is applied to the inverting terminal of the operational amplifier. The operational amplifier has a supply voltage of +12V and –12V. When the input voltage V_I is greater than the reference voltage the output voltage is at $+V_{sat}$. When the input voltage is less than the V_{ref} the output voltage is at - V_{sat}.

The output of the zero crossing detector must be converted into a form wherein $+V_{sat}$ is recognized as logic 1 and - V_{sat} as logic 0. To accomplish this a switching transistor is employed. This transistor has a supply voltage ranging from zero to –5 volts. It acts as an inverter. When the input to the base of the transistor is at logic 0 the base-collector terminal becomes open circuited. So the supply voltage drops across the collector terminal thus producing an output of 5 volts corresponding to logic 1. When the input to the base of transistor is 1, the collector-base and emitter-base junctions are forward biased as a result of which the transistor saturates. This produces an output voltage of 0.2 volts corresponding to logic 0.

The output bits of the switching transistor are fed as inputs to the port B of 8255. Port B is operated in mode 0. This Programmable Peripheral Interface (PPI) is interfaced with a PC. A stepper motor is also interfaced with the PC. The PC has a program which when executed counts the number of pulses arriving at port B. The number of pulses counted is compared with that which is stored in the program. When it falls within a certain range of the stored pulses, a series of commands are executed to transfer the appropriate bits to the output port (port A) of the PPI so as to enable the stepper motor to step through an angle of 1.8 degrees. An arm of the fixed Robot is connected with the shaft of the stepper motor. As it rotates through the required step angle , the arm is set into motion.

5.2 POWER SUPPLY REQUIREMENTS

Our project incorporates two transformers. One is needed to provide an AC supply to the stepper motor and the other is needed to provide a DC supply to the transistor and the comparators.



The transformer that supplies s DC voltage to the comparators, first steps down the AC supply of 230V from the primary side to 15V in the secondary side. The 15V AC supply is rectified by means of a Full Wave Rectifier. The rectifier has very high efficiency. Two diodes D1 and D3 conduct during the positive half cycles of the input supply and the remaining diodes D2 and D4 conduct during the negative half cycles of the input supply resulting in a constant output voltage with less ripples. The comparators require a DC voltage of ±12V for a biasing the input differential amplifier present within the operational amplifier IC.

The DC voltage sometimes fluctuates because of noise voltage induced in the mains. These voltage fluctuations may affect the normal working of the devices. So to maintain the DC voltage at a constant value a voltage regulator is employed. Two voltage regulators are needed, one for supplying a positive potential and the other for providing the negative potential to the operational amplifier. We use a 7812 voltage regulator to provide a positive potential of +12V and 7912 voltage regulator to provide a negative potential of –12V for the operational amplifier. The voltage regulator has three terminals. One is a common ground terminal, the second is a input terminal to which the output of the rectifier is fed and the third is an output terminal from which a regulated voltage is obtained. A capacitor is usually connected between input terminal and ground to cancel the inductive effects due to long distribution leads. The output capacitor improves the transient response

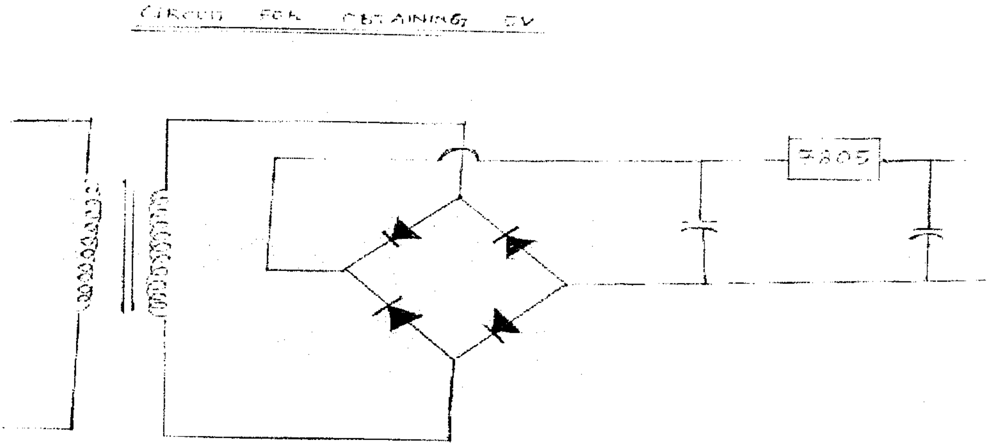


Fig 5.2 Circuit for Obtaining 5V

The switching transistor requires a supply of +5V. To obtain this the transformer steps down the supply from the mains to 9V in the secondary. This is then rectified by means of a bridge rectifier. The output of this rectifier is then fed to a 7805 voltage regulator, which produces a regulated voltage of 5V, this can now be used for biasing the switching transistor.

5.3 ADD-ON CARD

The add-on card is used for interfacing the stepper motor with the PC. The add-on card comprises of

- A comparator(74688)
- Bi-directional buffer(74245)
- Uni-directional Buffer(74244)
- Decoder(74138)
- Programmable Peripheral Interface(8255 PPI)

The block diagram of Add-on card is shown in Fig 5.3

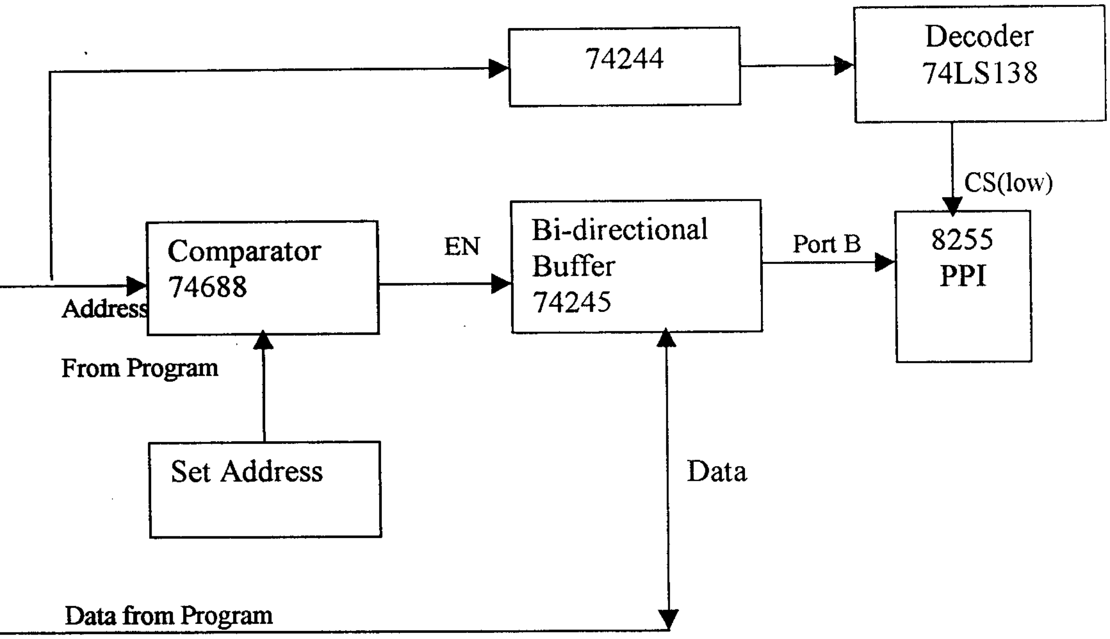


Fig. 5.3 BLOCK DIAGRAM OF ADD-ON CARD

The add-on card is inserted into an ISO (International Standards Organization) slot present within the PC. The add-on card has an address, which is set by the user. The address is represented by means of 10 bits A_0 to A_9. Bits A_9 and A_8 are set to one and the rest of the bits are set to either zero or one. Thus the address of the slot will be anywhere in the range 300 to 3FF. The address of the slot in which the add-on card is inserted is compared with that of the address stored in the program. When a match is found, an enable signal is sent to the bi-directional buffer from the comparator. When it is enabled the data from the output port is transferred to the bi-directional buffer. Meanwhile, the address of the slot is transferred from the add-on card to the Uni-directional buffer. This is then decoded using a decoder and enables the Programmable Peripheral Interface (8255). As soon as the Programmable Peripheral Interface is enabled the data from the bi-directional buffer is transferred to port B of 8255.

5.4 STEPPER MOTOR CARD

The Robot is operated by means of the stepper motor. The stepper motor steps through an angle in accordance with the bits that are transferred from the output port (Port A) of the Programmable Peripheral Interface to the stepper card.

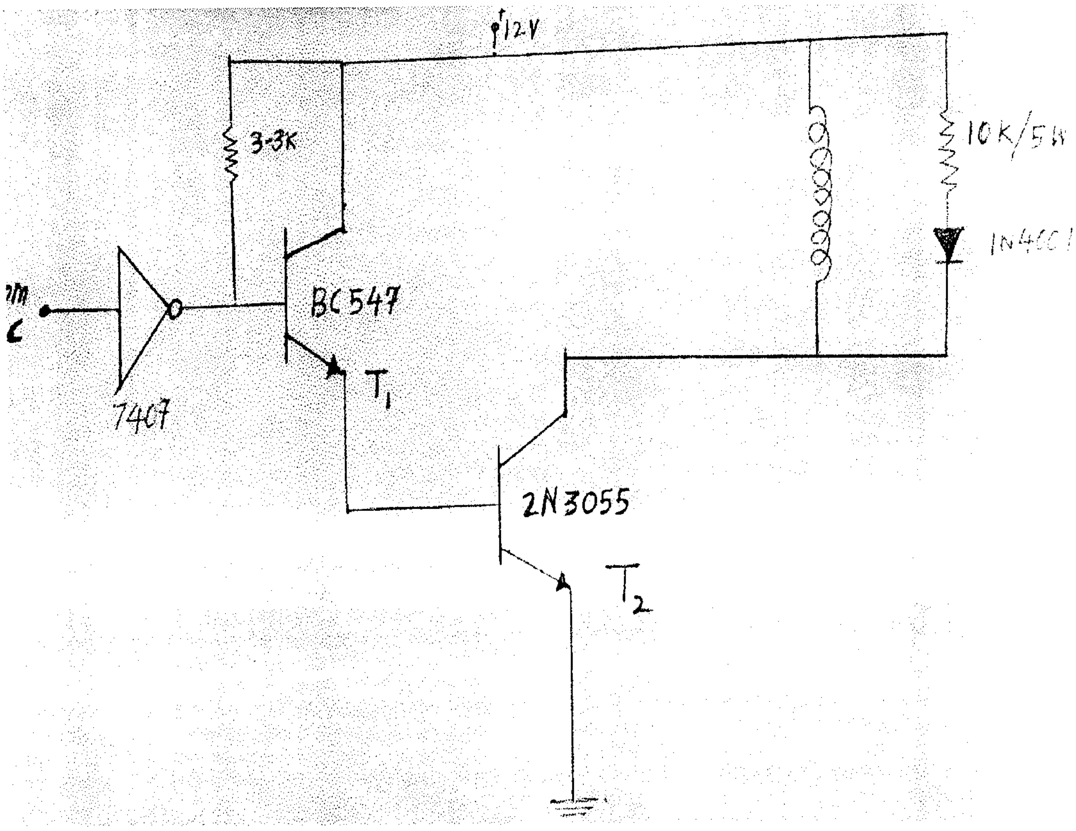


Fig 5.4 Circuit Diagram for interfacing Stepper Motor

Since two stepper motors are employed, port A is subdivided into two groups each consisting of four bits. The lower order bits $PA_0 - PA_3$ are transferred to one stepper card and the higher order bits PA_4 – PA_7 are transferred to another stepper card. Sometimes the bits from port A has to be transmitted through long cables. So to increase the driving capability an inverting buffer 7407 is used. The bits are inverted and given as inputs to the base of a Darlington transistor. Since large current is needed to drive the stepper motor the Darlington transistor is used because of its high current gain. The stepper motor is connected at the collector terminal. The diode is connected in parallel with the stepper motor. When the input to the base of the transistor T_1 the collector-base junction gets open-circuited as a result of which current flows through the windings of the stepper motor from the supply. When the input to the base of T_1 is low, it is driven into saturation. All the current passes through T_1. The presence of diode 1N4001 across the winding is to avoid flyback. Flyback is the phenomenon, which occurs when high current levels are switched in inductors. The diode provides a path for the flyback current to drain.

6.SOFTWARE

6.1 ALGORITHM

An algorithm is an explanation of the logic involved in a program in words .Here the algorithm to control the stepper motor is explained .

Step1:

Initialize the ports of 8255 by writing control word format.

Step2 :

Get the voice pulse through LSB of port B .

Step3 :

Start with checking for negative part of the pulse (bit zero).

Step4 :

If the bit is not zero repeat step 3 , else check next for the positive pulse (bit one).

Step5 :

If bit one is detected , increment a count, else start fromstep3.

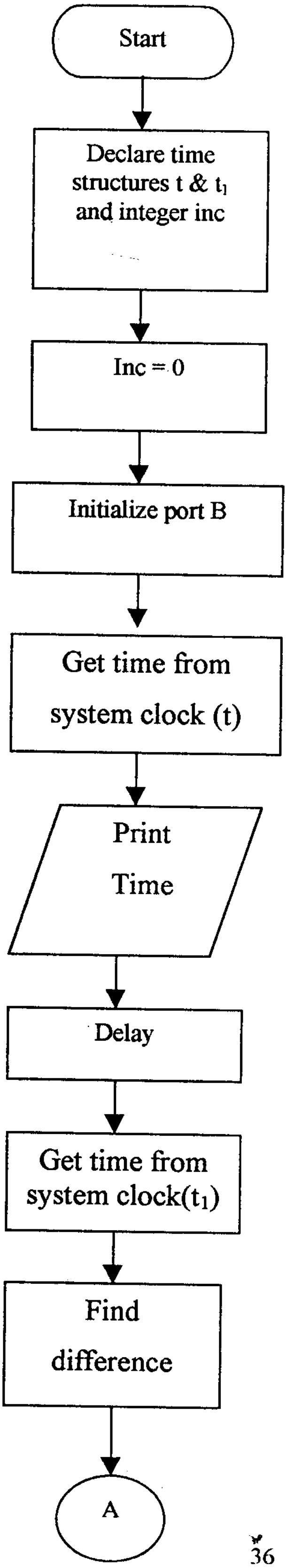
Step6 :

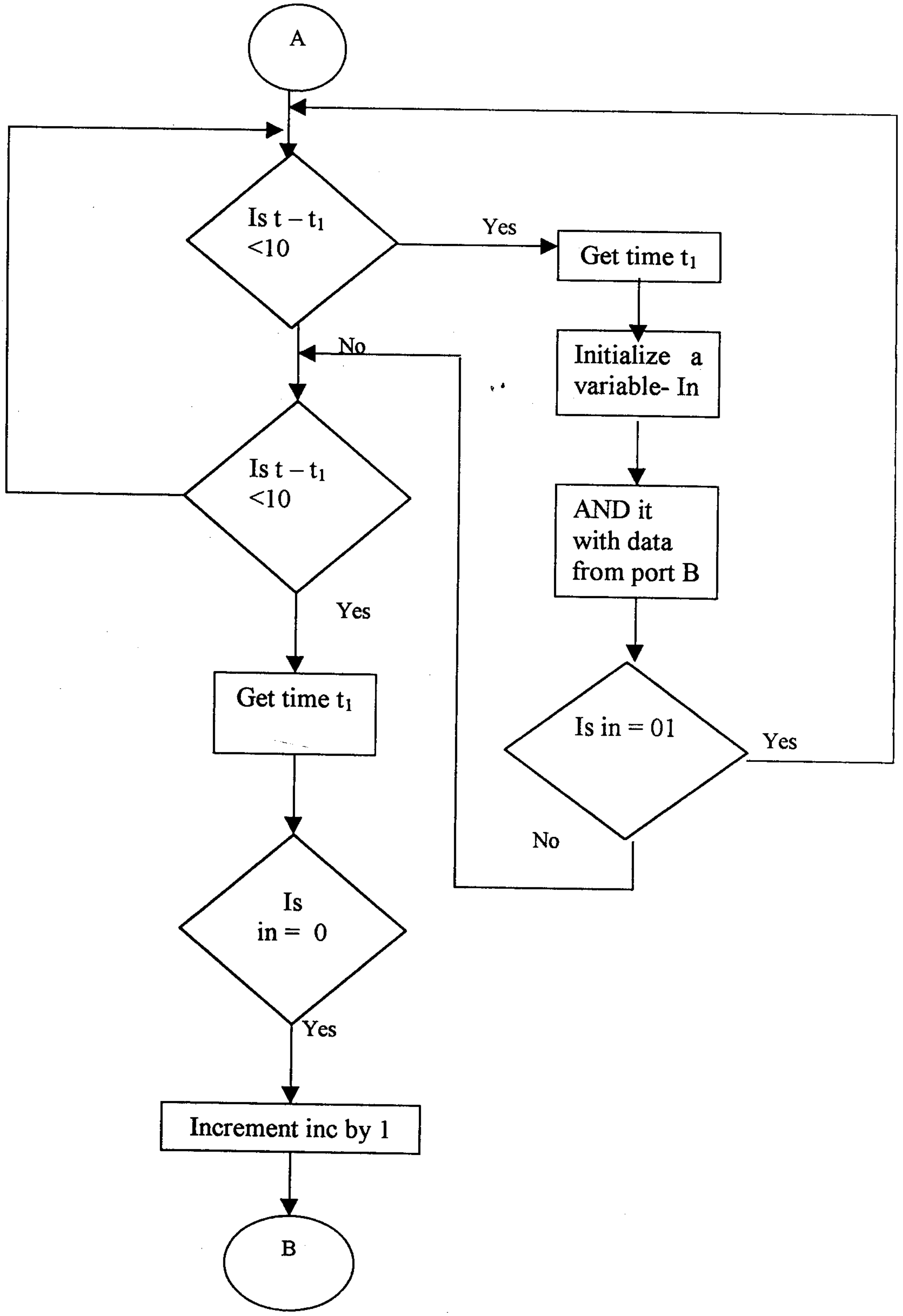
Repeat steps 3 to 5 until all the voice pulse is read through the port.

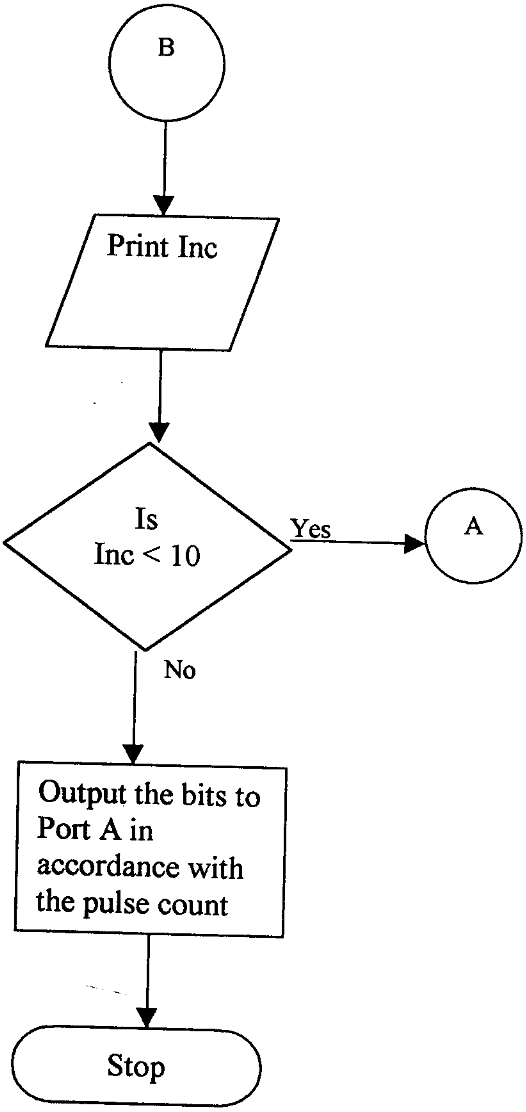
Step7 :

Get the count and depending upon the count give the command to the stepper motor to act , through the port A (4 data bits each to the 2 stepper motor).

6.2 FLOWCHART


Start
Declare time structures t & t_1 and integer inc
Inc = 0
Initialize port B
Get time from system clock (t)
Print Time
Delay
Get time from system clock(t_1)
Find difference
A


A
Is t – t1 <10
Yes
Get time t1
Initialize a variable- In
No
Is t – t1 <10
AND it with data from port B
Yes
Get time t1
Is in = 01
Yes
No
Is in = 0
Yes
Increment inc by 1
B


B
Print Inc
Is
Inc < 10
Yes
A
No
Output the bits to Port A in accordance with the pulse count
Stop

6.3 CODING

```
#include <stdio.h>

#include <dos.h>

#include <math.h>

#include <conio.h>

int main(void)

{

   clrscr();

   struct  time t;

   struct time t1;

   int in,Inc=0;

   outport(0x303,0x82);

 aa:

   Inc=0;

 // clrscr();

   gettime(&t);

   printf("The current time is: %2d:%02d:%02d.%02d\n",

              t.ti_hour, t.ti_min, t.ti_sec, t.ti_hund);

   printf("\n\n");

   gettime(&t1);
```

```
ze0:
  while(abs(t.ti_sec - t1.ti_sec) < 10)
   {
         gettime(&t1);
         in=0x01 & inport(0x301);
         if(in == 0x01)
                {
                 goto ze0;
                }
         else
             goto ze1;
   }
 ze1:
          while(abs(t.ti_sec- t1.ti_sec) < 10)
          {
                 gettime(&t1);
                 in=0x01 & inport(0x301);
                 if(in == 0x00)
                 {
                 goto ze1;
                 }
          Inc++;
          goto ze0;  }
printf("%d",Inc);
if(Inc<=10)
{
goto aa;
}
if(Inc<=100)
{
in=0;
xx:
outport(0x300,0x05);
delay(10);
outport(0x300,0x09);
delay(10);
outport(0x300,0x0a);
delay(10);
outport(0x300,0x06);
delay(10);
in++;
if
(in==10)
{
```

```
goto aa;
}
goto xx;
}
if(Inc<=200)
{
in=0;
xxx:
outport(0x300,0x06);
delay(10);
outport(0x300,0x0a);
delay(10);
outport(0x300,0x09);
delay(10);
outport(0x300,0x05);
delay(10);
in++;
if
(in==100)
{
goto aa;
}
goto xxx;
```

```
}
if(Inc<=300)
{
in=0;
yy:
outport(0x300,0x50);
delay(10);
outport(0x300,0x90);
delay(10);
outport(0x300,0xa0);
delay(10);
outport(0x300,0x60);
delay(10);
in++;
if
(in==100)
{
goto aa;
}
goto yy;
}
if(Inc<=400)
{
```

```
in=0;
yyy:
outport(0x300,0x60);
delay(10);
outport(0x300,0xa0);
delay(10);
outport(0x300,0x90);
delay(10);
outport(0x300,0x50);
delay(10);
in++;
if
(in==100)
{
goto aa;
}
goto yyy;
}
if(Inc<=500)
{
```

```
in=0;

zz:

outport(0x300,0x00);

delay(10);

outport(0x300,0x00);

delay(10);

outport(0x300,0x00);

delay(10);

outport(0x300,0x00);

delay(10);

in++;

if

(in==100)

{

goto aa;

}

goto zz;

}

}
```

7. APPLICATION

A Project developed will be incomplete if the application of the developed product is not mentioned. As the robot is voice controlled it tends to replace the existing robots that are not voice controlled. This robot could be an indispensable object in more ways than one as follows:

- It could be used as one of the robot when developed further that sings songs, plays games, tells nursery rhymes, recites poems and can even wake you in the morning and guide you through an exercise routine when command it through your voice.
- Navigation Chair for severely disabled people could be improved by using this robot so that the disabled could run his/her chair by giving voice command.
- This robot when developed into a mobile robot with voice programming could replace all the robots used at present for order picking, inspection, paint coating etc that are now pre-programmed manually by hand.
- Another important aspect is that any robot developed could be restricted to certain applications. But through this voice controlled robot the area of application could be expanded to a large extent.

8. CONCLUSION

The project was a effort rewarding success .The connection between the external hardware and the PC was established and robot obeyed the voice command perfectly to the expectations.

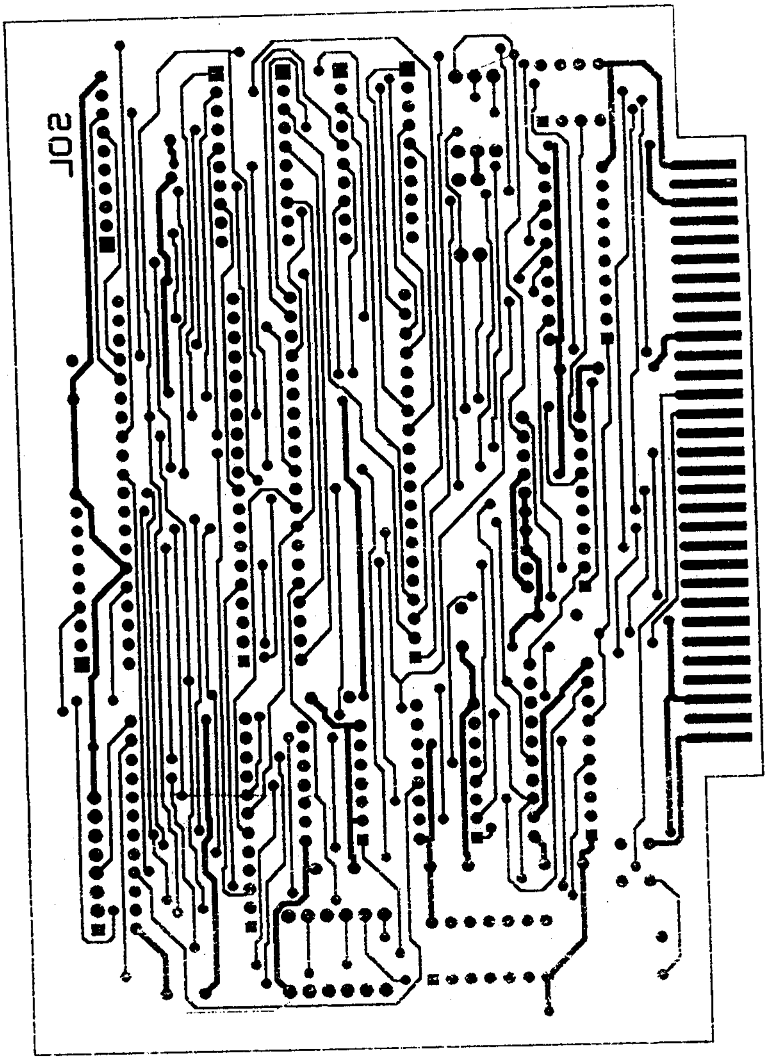
We had various levels of testing of the robot, different commands being given and their pulse count were found out .As a result of that, and the level of understandability of the robot for voice command is recognized.

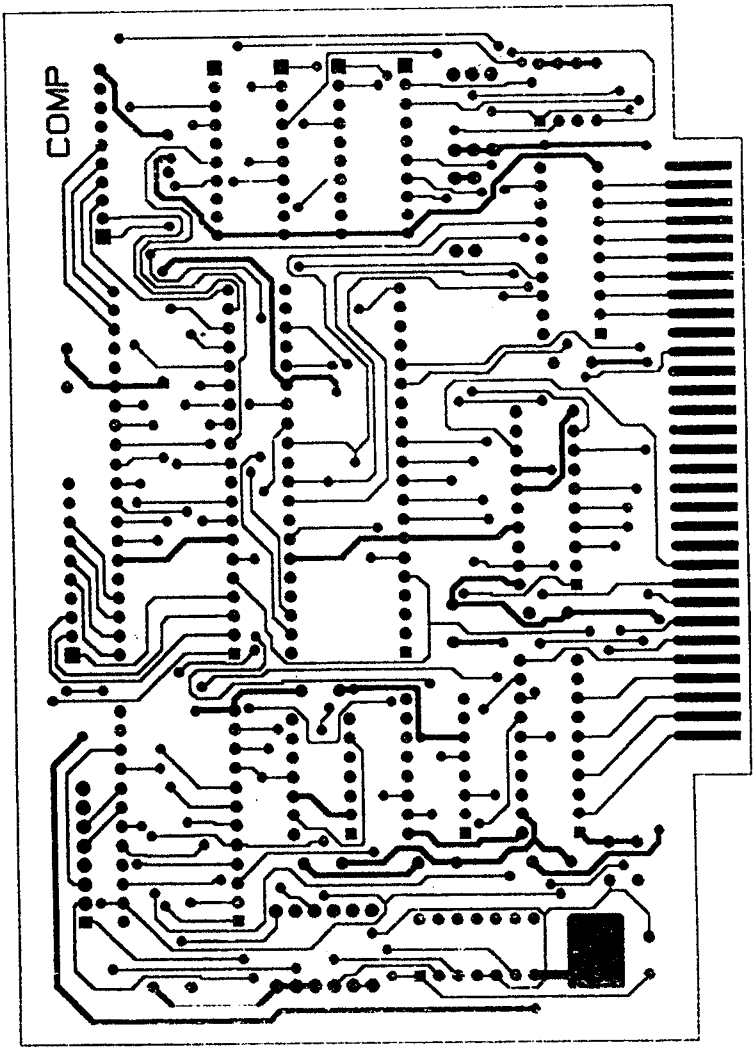
Our project forms a basic foundation for many of the much advanced research projects. Further improvements may be done. As we have seen there are tremendous implications of the project and it would be quite promising.

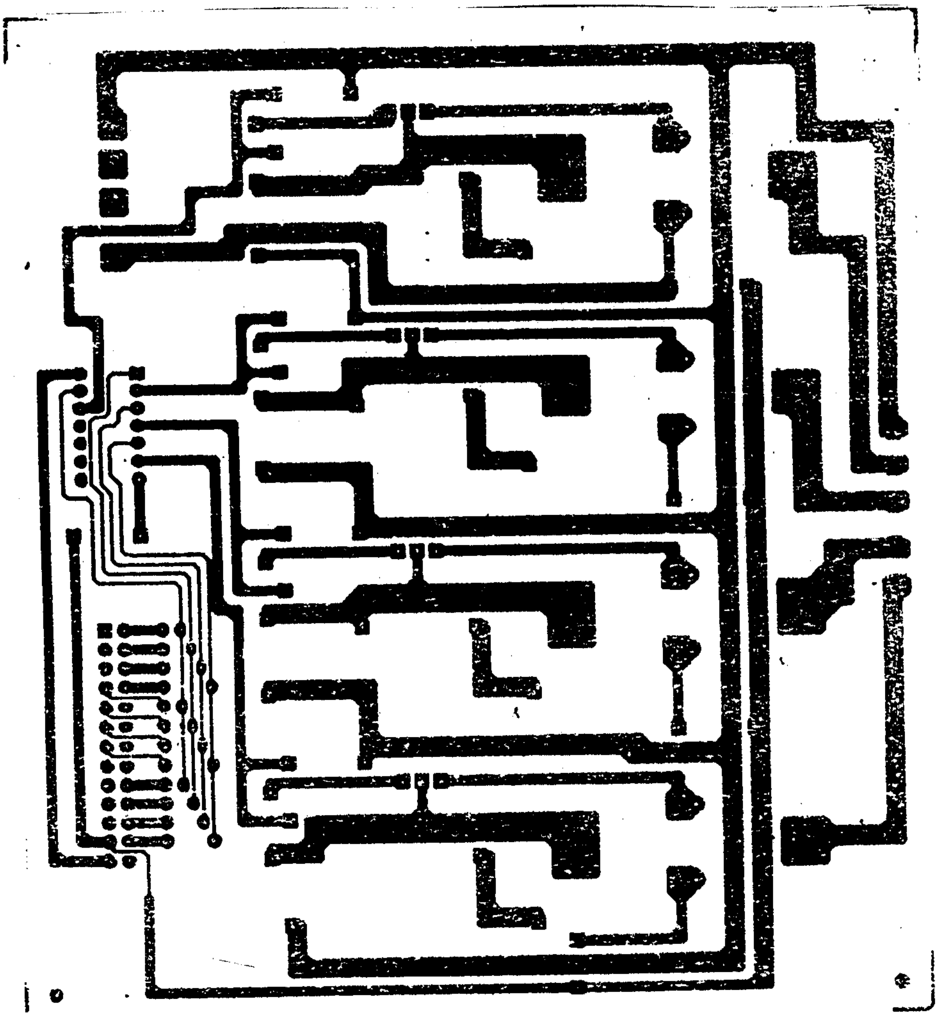
In its relative infancy, the state of the art of robotics applications is, in some ways, paralleling the development of digital computers. It is to be assumed that over the next few years, non-traditional robotics applications will begin to appear which will, in part, contribute to the development of the factory of the future or home maid of the future.

9.BIBLIOGRAPHY

➢ Microprocessor Interfacing	-	Douglas Hall
➢ Microprocessor architecture, programming & applications	-	R.S.Gaonkar
➢ Control Systems	-	Benjamin Kuo
➢ A Text-Book of Electrical Technology	-	B.L.Theraja, A.L.Theraja
➢ Programming with C++	-	Robert Lafore
➢ Introduction to Robotics	-	Fred Martin
➢ Electronic Principles	-	Malvino
➢ Linear Integrated Circuits	-	Roy Choudhary
➢ Electronic Devices and Circuits	-	Robert Boylestad
➢ PC complete	-	Peter Kuhns


JO2


COMP



74LS138, S138
Decoders/Demultiplexers

1-Of-8 Decoder/Demultiplexer
Product Specification

ic Products

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74LS138	20ns	6.3mA
74S138	7ns	49mA

ATURES

- **emultiplexing capability**
- **ultiple input enable for easy xpansion**
- **eal for memory chip select ecoding**
- **irect replacement for Intel 3205**

SCRIPTION

'138 decoder accepts three binary
ghted inputs (A_0, A_1, A_2) and when
abled, provides eight mutually exclu-
e, active LOW outputs ($\overline{O}_0 - \overline{O}_7$). The
ice features three Enable Inputs: two
ive LOW ($\overline{E}_1$, $\overline{E}_2$) and one active
GH (E_3). Every output will be HIGH
ess E_1 and E_2 are LOW and E_3 is
GH. This multiple enable function al-
vs easy parallel expansion of the de-
e to a 1-of-32 (5 lines to 32 lines)
coder with just four '138s and one
verter.

e device can be used as an eight
tput demultiplexer by using one of the
tive LOW Enable inputs as the Data
put and the remaining Enable inputs as
robes. Enable inputs not used must be
rmanently tied to their appropriate
ctive HIGH or active LOW state.

ORDERING CODE

PACKAGES	COMMERCIAL RANGE $V_{CC} = 5V \pm 5\%$; $T_A = 0°C$ to $+70°C$
Plastic DIP	N74S138N, N74LS138N
Plastic SO	N74LS138D, N74S138D

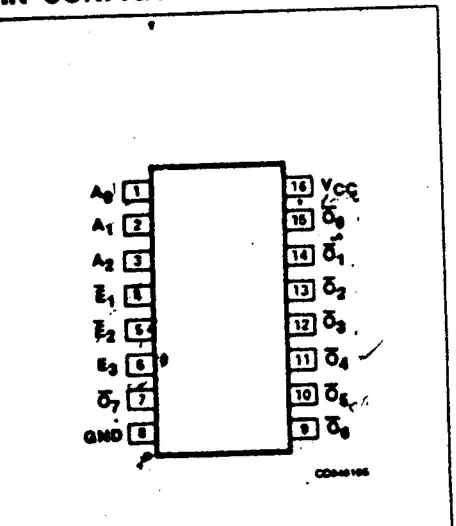
NOTE:
For information regarding devices processed to Military Specifications see the Signetics Military Pro
Data Manual.

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

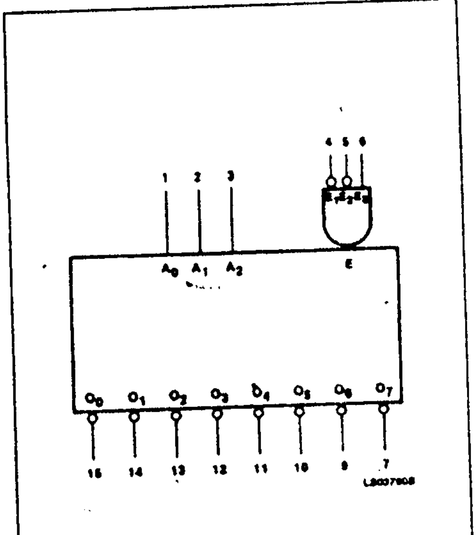
PINS	DESCRIPTION	74S	74LS
All	Inputs	1Sul	1LSul
All	Outputs	10Sul	10LSul

NOTE:
Where a 74S unit load (Sul) is 50μA I_{IH} and −2.0mA I_{IL}, and a 74LS unit load (LSul) is 20μA I_{IH} and −
I_{IL}.

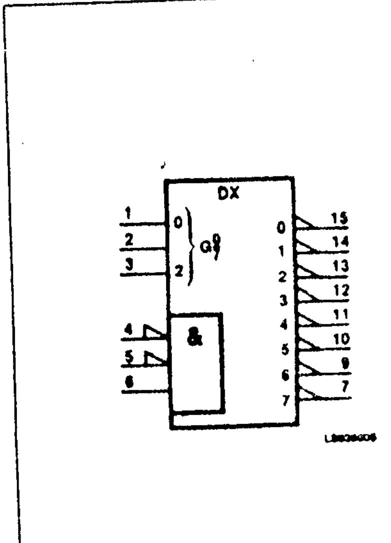
IN CONFIGURATION



LOGIC SYMBOL



LOGIC SYMBOL (IEEE/IEC)



74LS245
Transceiver

Octal Transceiver (3-State)
Product Specification

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

FEATURES

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

DESCRIPTION

The 74LS245 is an octal transceiver featuring non-inverting 3-State bus compatible outputs in both send and receive directions. The outputs are all capable of sinking 24mA and sourcing up to 15mA, producing very good capacitive drive characteristics. The device features a Chip Enable (CE) input for easy cascading and a Send/Receive (S/R) input for direction control. All data inputs have hysteresis built in to minimize AC noise effects.

ORDERING CODE

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

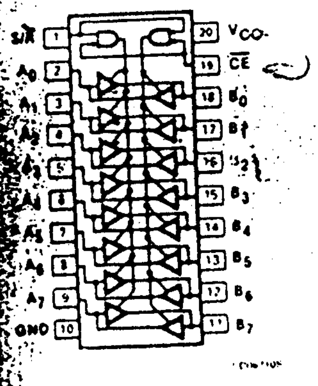
NOTE:
For information regarding devices processed to Military Specifications, see the Signetics Military Pro Data Manual.

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

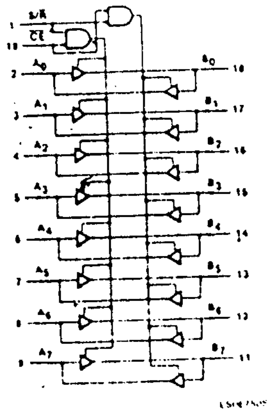
PINS	DESCRIPTION	74LS
All	Inputs	1LSul
All	Outputs	30LSul

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

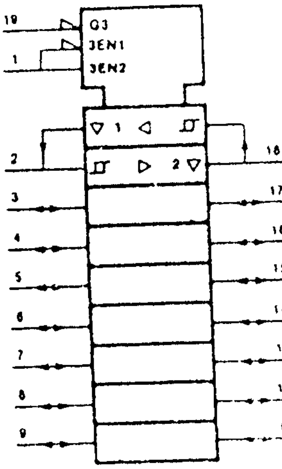
PIN CONFIGURATION



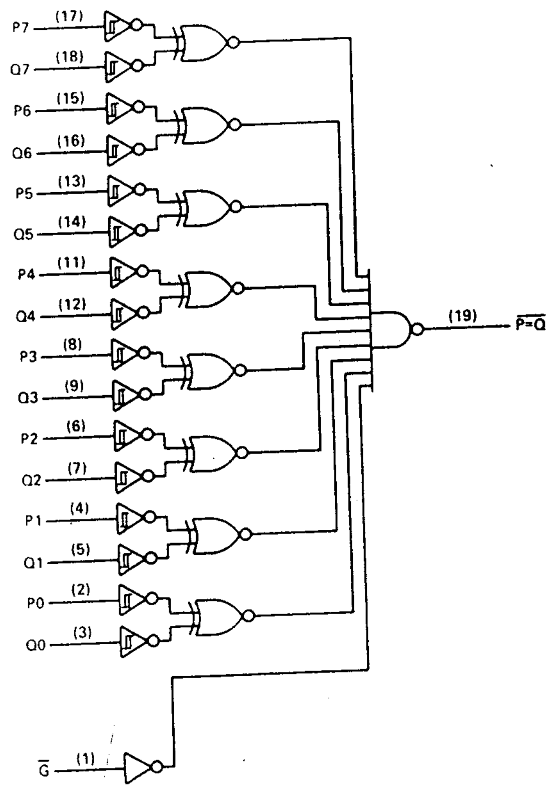
LOGIC SYMBOL



LOGIC SYMBOL (IEEE/IEC)



agram (positive logic)



Pin numbers shown on logic notation are for DW, J or N packages

atings over operating free-air temperature range (unless otherwise noted)

e Note 1) . . . . 5.

nputs of 'LS682 and 'LS683 7

other inputs 7

oltage: 'LS683, 'LS685, 'LS687, 'LS689 . . . .

temperature range: SN54LS682 thru SN54LS689 . . . . −55°C to 125

SN74LS682 thru SN74LS689 . . . . 0°C to 70

ure range . . . . −65°C to 150

e with respect to network ground terminal

ended operating conditions

	SN54LS' MIN	SN54LS' NOM	SN54LS' MAX	SN74LS' MIN	SN74LS' NOM	SN74LS' MAX	UNIT
voltage, V_{CC}	4.5	5	5.5	4.75	5	5.25	V
output current, I_{OH}			−400			−400	µA
output current, I_{OL}			12			24	mA
free-air temperature, T_A	−55		125	0		70	°C

al characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS†		SN54LS' MIN	SN54LS' TYP‡	SN54LS' MAX	SN74LS' MIN	SN74LS' TYP‡	SN74LS' MAX	UNIT
High-level input voltage				2			2			V
Low-level input voltage						0.7			0.8	V
V_{T-} Hysteresis	P or Q inputs	V_{CC} = MIN			0.4			0.4		V
Input clamp voltage		V_{CC} = MIN,	I_I = −18 mA			−1.5			−1.5	V
High-level output voltage		V_{CC} = MIN, V_{IL} = V_{IL}max,	V_{IH} = 2 V, I_{OH} = −400 µA	2.5			2.7			V
Low-level output voltage		V_{CC} = MIN, V_{IH} = 2 V, V_{IL} = V_{IL}max	I_{OL} = 12 mA		0.25	0.4		0.25	0.4	V
			I_{OL} = 24 mA					0.35	0.5	
Input current at maximum input voltage	Q inputs, 'LS682	V_{CC} = MAX,	V_I = 5.5 V			0.1			0.1	mA
	All other inputs	V_{CC} = MAX,	V_I = 7 V							
High-level input current		V_{CC} = MAX,	V_I = 2.7 V			20			20	µA
Low-level input current	Q inputs, 'LS682	V_{CC} = MAX,	V_I = 0.4 V			−0.4			−0.4	mA
	All other inputs					−0.2			−0.2	
Short-circuit output current		V_{CC} = MAX,	V_O = 0	−20		−100	−20		−100	mA
Supply current	'LS682	V_{CC} = MAX,	See Note 2		42	70		42	70	mA
	'LS684				40	65		40	65	
	'LS686				44	75		44	75	
	'LS688				40	65		40	65	

ypical values are at V_{CC} = 5 V, T_A = 25°C.

more than one output should be shorted at a time, and duration of the short-circuit should not exceed one second.

2: I_{CC} is measured with any $\bar{G}$ inputs grounded, all other inputs at 4.5 V, and all outputs open.

hing characteristics, V_{CC} = 5 V, T_A = 25°C

AMETER#	FROM (INPUTS)	TO (OUTPUT)	TEST CONDITIONS	'LS682 MIN	'LS682 TYP	'LS682 MAX	'LS684 MIN	'LS684 TYP	'LS684 MAX	'LS686 MIN	'LS686 TYP	'LS686 MAX	'LS688 MIN	'LS688 TYP	'LS688 MAX	UNIT
LH	P	$\overline{P=Q}$	R_L = 667 Ω, C_L = 45 pF, All other inputs low, See Note 3		13	25		15	25		13	25		18	27	ns
HL					15	25		17	25		20	30		20	30	
LH	Q	$\overline{P=Q}$			14	25		16	25		13	25		18	27	ns
HL					15	25		15	25		21	30		20	30	
LH	$\bar{G}$, $\bar{G}$1	$\overline{P=Q}$									11	20		12	18	ns
HL											19	30		13	20	
LH	P	$\overline{P>Q}$			20	30		22	30		19	30				ns
HL					15	30		17	30		15	30				
LH	Q	$\overline{P>Q}$			21	30		24	30		18	30				ns
HL					19	30		20	30		19	30				
LH	$\bar{G}$2	$\overline{P>Q}$									21	30				ns
HL											16	25				

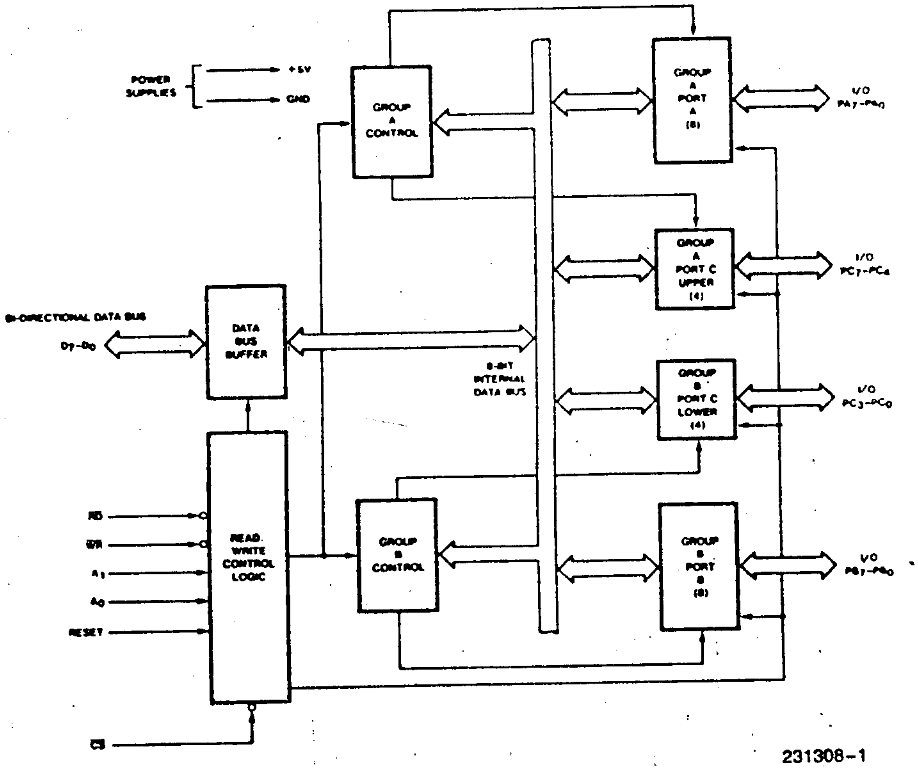
LH ≡ propagation delay time, low-to-high-level outputs; t_{PHL} ≡ propagation delay time, high-to-low-level output.

TE 3: See General Information Section for load circuits and voltage waveforms.

8255A/8255A-5
PROGRAMMABLE PERIPHERAL INTERFACE

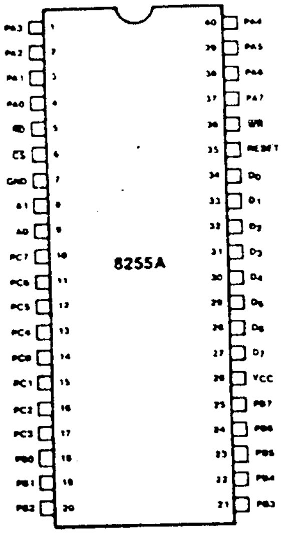
- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel Microprocessor Families
- Improved Timing Characteristics
- Direct Bit Set/Reset Capability Easing Control Application Interface
- Reduces System Package Count
- Improved DC Driving Capability
- Available in EXPRESS
  - —Standard Temperature Range
  - —Extended Temperature Range
- 40 Pin DIP Package or 44 Lead PLCC
  (See Intel Packaging: Order Number: 231369)

The Intel 8255A is a general purpose programmable I/O device designed for use with Intel microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.



231308–1

Figure 1. 8255A Block Diagram



231308–2

Figure 2. Pin Configuration

November 1987

A FUNCTIONAL DESCRIPTION

eral

255A is a programmable peripheral interface levice designed for use in Intel microcomputer ns. Its function is that of a general purpose I/O onent to interface peripheral equipment to the computer system bus. The functional configu- of the 8255A is programmed by the system are so that normally no external logic is neces- o interface peripheral devices or structures.

Bus Buffer

3-state bidirectional 8-bit buffer is used to inter- the 8255A to the system data bus. Data is mitted or received by the buffer upon execution out or output instructions by the CPU. Control s and status information are also transferred gh the data bus buffer.

d/Write and Control Logic

function of this block is to manage all of the nal and external transfers of both Data and rol or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

($\overline{CS}$)

Chip Select. A "low" on this input pin enables the communication between the 8255A and the CPU.

($\overline{RD}$)

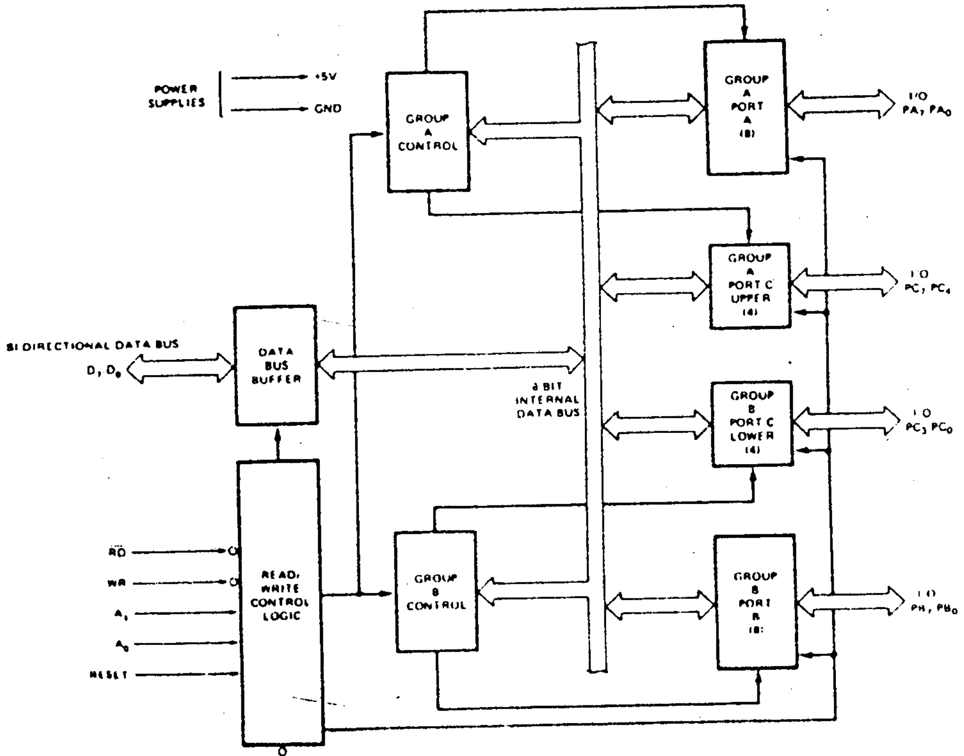
Read. A "low" on this input pin enables the 8255A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255A.

($\overline{WR}$)

Write. A "low" on this input pin enables the CPU to write data or control words into the 8255A.

(A_0 and A_1)

Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus (A_0 and A_1).



8255A BASIC OPERATION

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	**Input Operation (READ)**
0	0	0	1	0	Port A → Data Bus
0	1	0	1	0	Port B → Data Bus
1	0	0	1	0	Port C → Data Bus
					Output Operation (WRITE)
0	0	1	0	0	Data Bus → Port A
0	1	1	0	0	Data Bus → Port B
1	0	1	0	0	Data Bus → Port C
1	1	1	0	0	Data Bus → Control
					Disable Function
X	X	X	X	1	Data Bus → 3-State
1	1	0	1	0	Illegal Condition
X	X	1	1	0	Data Bus → 3-State

(RESET)

Reset. A "high" on this input clears the control register and all ports (A, B, C) are set to the input mode.

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 8255A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 8255A.

Each of the Control blocks (Group A and Gro accepts "commands" from the Read/Write C Logic, receives "control words" from the in data bus and issues the proper commands to i sociated ports.

Control Group A—Port A and Port C upper (C

Control Group B—Port B and Port C lower (C

The Control Word Register can **Only** be writte No Read operation of the Control Word Regi allowed.

Ports A, B, and C

The 8255A contains three 8-bit ports (A, B, a All can be configured in a wide variety of fun characteristics by the system software but ea its own special features or "personality" to enhance the power and flexibility of the 8255

Port A. One 8-bit data output latch/buffer a 8-bit data input latch.

Port B. One 8-bit data input/output latch/buf one 8-bit data input buffer.

Port C. One 8-bit data output latch/buffer a 8-bit data input buffer (no latch for input). Th can be divided into two 4-bit ports under the control. Each 4-bit port contains a 4-bit latch can be used for the control signal outputs and signal inputs in conjunction with ports A and

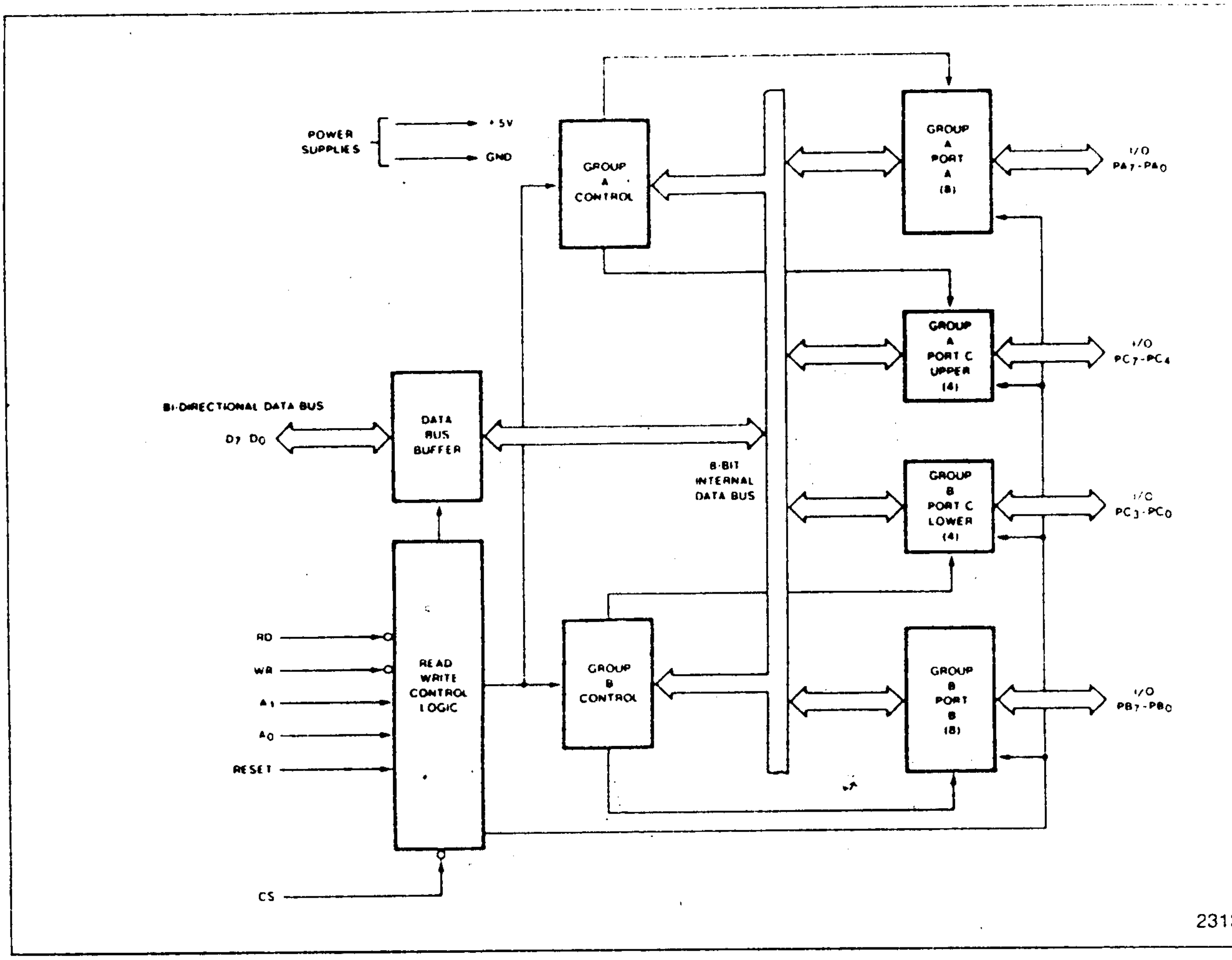
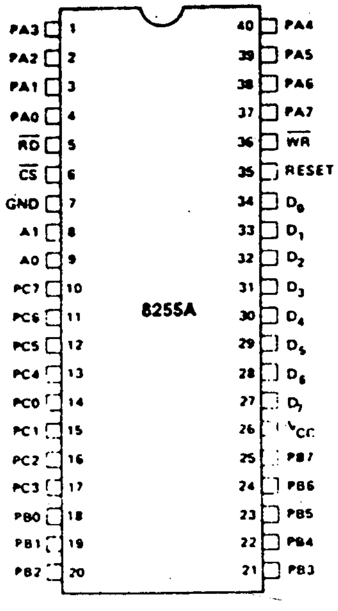


Figure 4. 8225A Block Diagram Showing Group A and Group B Control Functions

Pin Configuration



231308-5

Pin Names

D_7-D_0	Data Bus (Bi-Directio
RESET	Reset Input
$\overline{CS}$	Chip Select
$\overline{RD}$	Read Input
$\overline{WR}$	Write Input
A0, A1	Port Address
PA7–PA0	Port A (BIT)
PB7–PB0	Port B (BIT)
PC7–PC0	Port C (BIT)
V_{CC}	+5 Volts
GND	0 Volts

8255A OPERATIONAL DESCRIP

Mode Selection

Mode 0—Basic Input/Output

Mode 1—Strobed Input/Output

Mode 2—Bi-Directional Bus

When the reset input goes "high" all ports will be set to the input mode (i.e., all 24 lines will be in the high impedance state). After the reset is removed the 8255A can remain in the input mode with no additional initialization required. During the execution of the system program any of the other modes may be selected using a single output instruction. This allows a single 8255A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

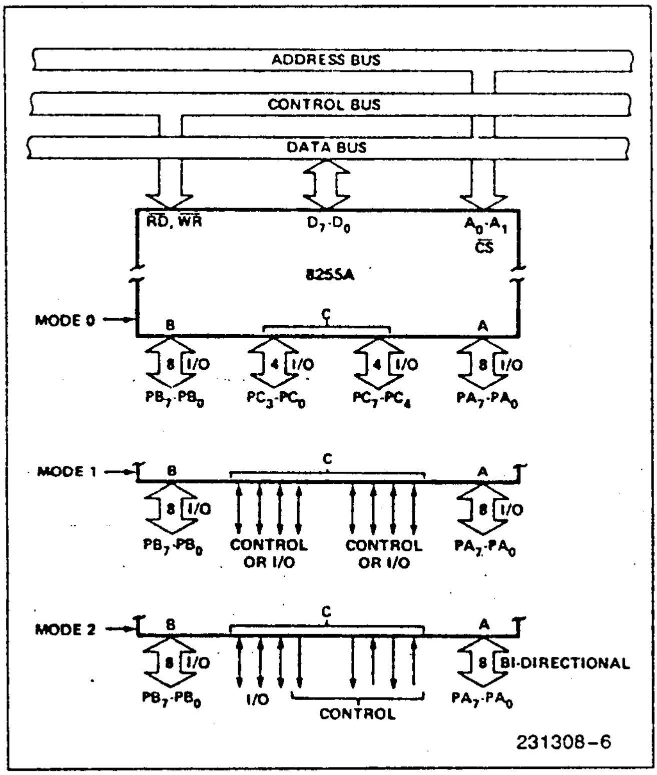


Figure 5. Basic Mode Definitions and Bus Interface

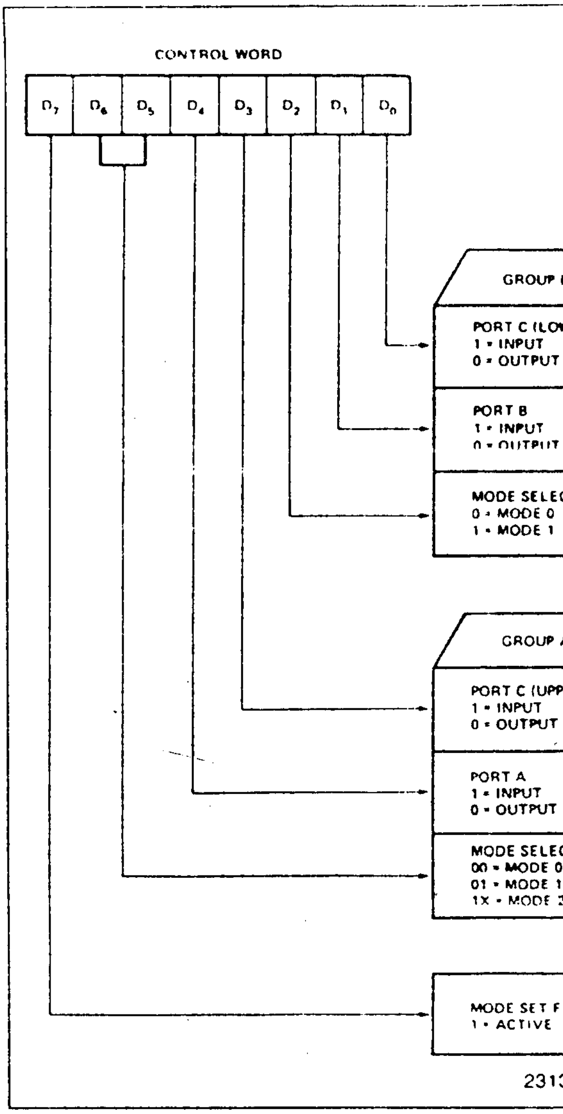


Figure 6. Mode Definition Format

The mode definitions and possible mode c tions may seem confusing at first but after a review of the complete device operation a logical I/O approach will surface. The desig 8255A has taken into account things such cient PC board layout, control signal definitio layout and complete functional flexibility to almost any peripheral device with no extern Such design represents the maximum use available pins.

Single Bit Set/Reset Feature

Any of the eight bits of Port C can be Set c using a single OUTput instruction. This fea duces software requirements in Control-base cations.

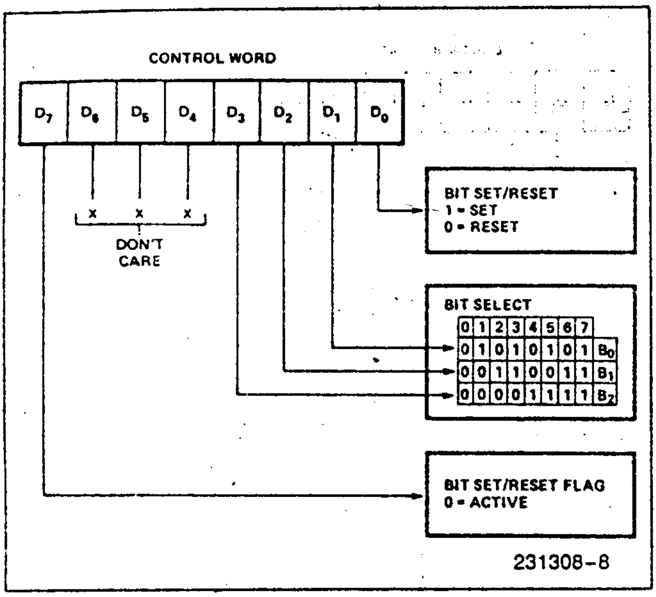


Figure 7. Bit Set/Reset Format

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were data output ports.

Interrupt Control Functions

When the 8255A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the Programmer to disa allow a specific I/O device to interrupt the CP out affecting any other device in the interrupt ture.

INTE flip-flop definition:

(BIT-SET)—INTE is set—Interrupt enable

(BIT-RESET)—INTE is RESET—Interrupt disa

NOTE:
All Mask flip-flops are automatically reset mode selection and device Reset.

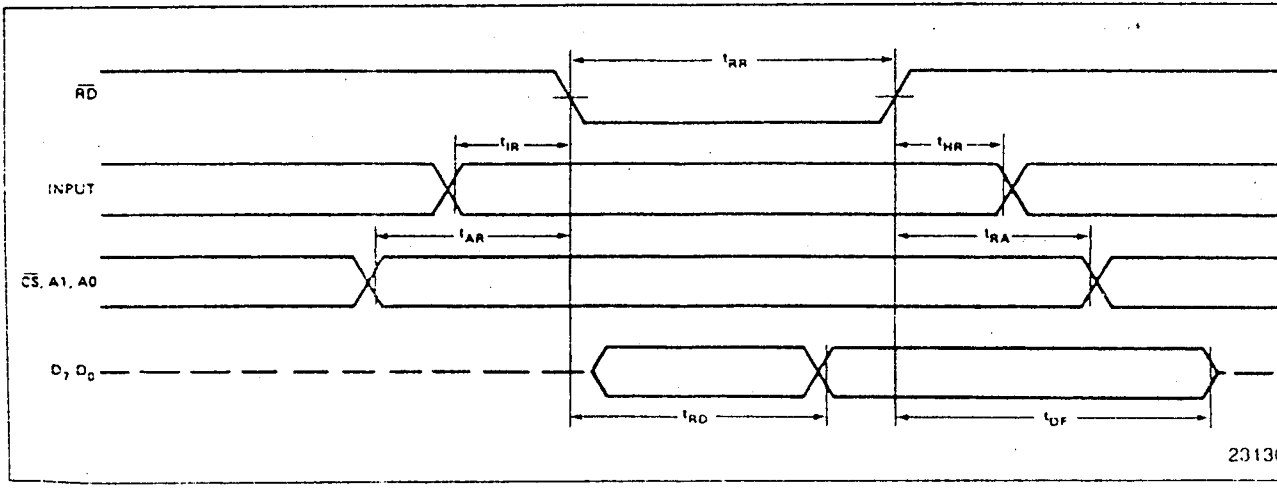
Operating Modes

MODE 0 (Basic Input/Output). This function figuration provides simple input and output tions for each of the three ports. No "handsh is required, data is simply written to or read specified port.

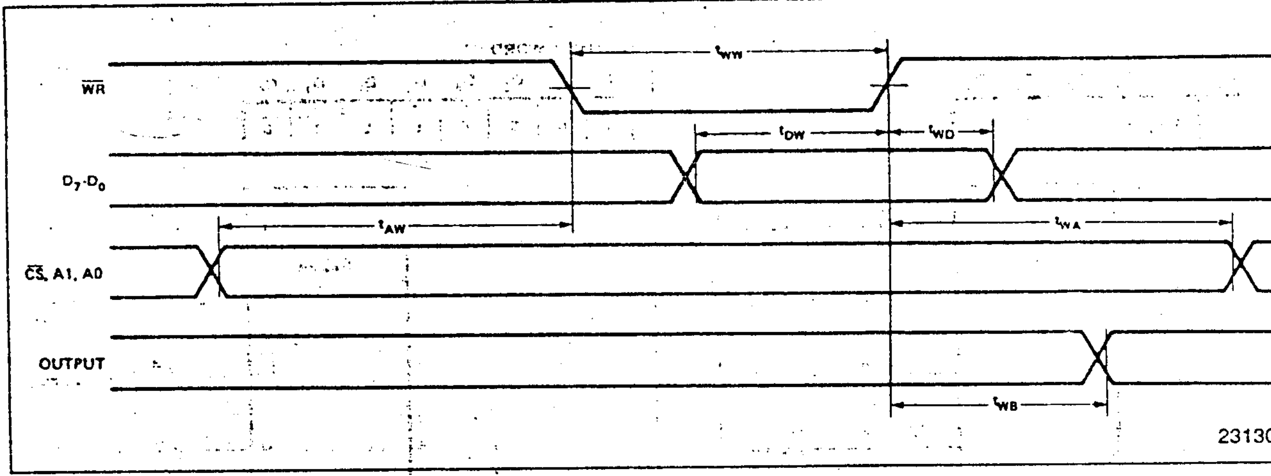
Mode 0 Basic Functional Definitions:

- Two 8-bit ports and two 4-bit ports.
- Any port can be input or output.
- Outputs are latched.
- Inputs are not latched.
- 16 different Input/Output configurations a sible in this Mode.

MODE 0 (BASIC INPUT)



MODE 0 (BASIC OUTPUT)



MODE 0 PORT DEFINITION

A		B		Group A			Group B	
D4	D3	D1	D0	Port A	Port C (Upper)	#	Port B	Po (Lo
0	0	0	0	OUTPUT	OUTPUT	0	OUTPUT	OUT
0	0	0	1	OUTPUT	OUTPUT	1	OUTPUT	INF
0	0	1	0	OUTPUT	OUTPUT	2	INPUT	OUT
0	0	1	1	OUTPUT	OUTPUT	3	INPUT	INF
0	1	0	0	OUTPUT	INPUT	4	OUTPUT	OUT
0	1	0	1	OUTPUT	INPUT	5	OUTPUT	INF
0	1	1	0	OUTPUT	INPUT	6	INPUT	OUT
0	1	1	1	OUTPUT	INPUT	7	INPUT	INI
1	0	0	0	INPUT	OUTPUT	8	OUTPUT	OUT
1	0	0	1	INPUT	OUTPUT	9	OUTPUT	INI
1	0	1	0	INPUT	OUTPUT	10	INPUT	OUT
1	0	1	1	INPUT	OUTPUT	11	INPUT	INI
1	1	0	0	INPUT	INPUT	12	OUTPUT	OUT
1	1	0	1	INPUT	INPUT	13	OUTPUT	IN
1	1	1	0	INPUT	INPUT	14	INPUT	OU
1	1	1	1	INPUT	INPUT	15	INPUT	IN