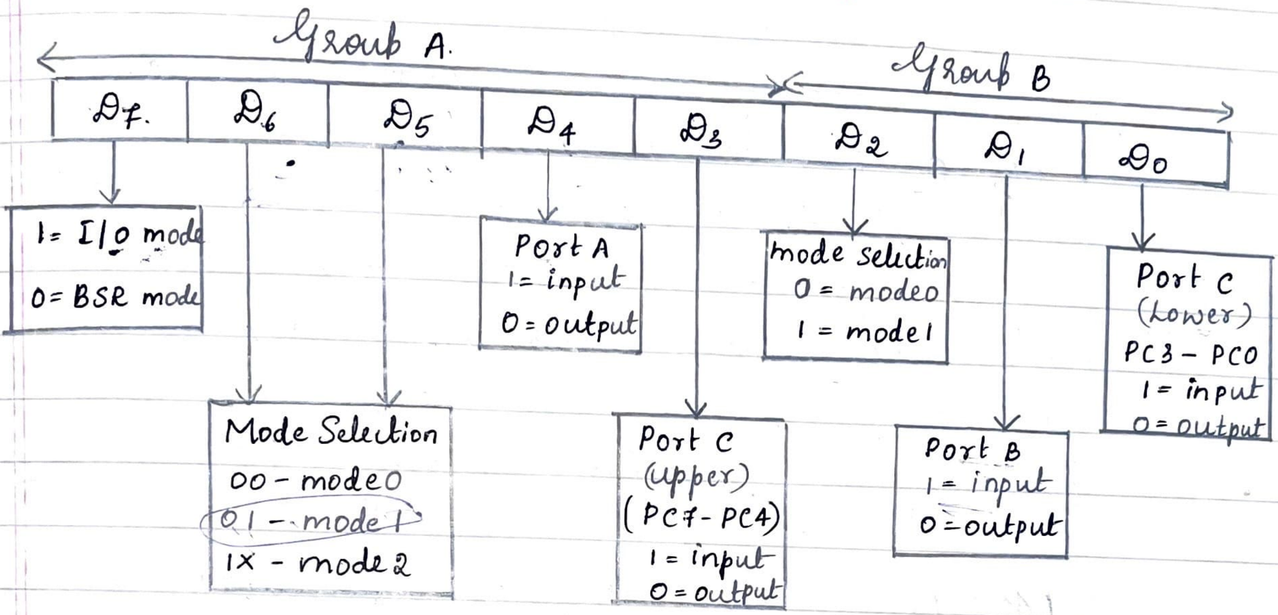
Module - 5.
Interfacing

Q1 Explain control word format of 8255. (I/O mode of 8255) [Jan 2018 / July 2018 (6M)]



8255 control word format (I/O mode)

- Control word format also called I/O selector is used to select I/O mode for 8255
- Intell calls mode0 as the basic I/O mode.
- In mode 0, any ports such as A, B or C can be programmed as input or output.
- Given port cannot be input/output at the same time.
- 8255 can be programmed in any of the 4-modes, by sending a byte to the control register of 8255.
- To find the address of each of ports A, B, C & control register we use the method called as "mapping" the I/O port.

Jan 2018 (8M)

2) Explain the various modes of 8255. & find the control word for the following.

⟹ i) All ports of A, B, & C are o/p ports (mode0)

ii) PA = IN, PB = Out, PCL = Out & PCH = Out

→ Various modes of 8255 :- mode 0, 1, 2 & BSR

1) **Mode 0** :- i) It is also called as simple I/O mode.

ii) In this mode any ports like, A, B, C & Control register can be used as input or output.

iii) In this mode, either all bits are out or all bits are in

iv). Most of the applications involving 8255, use mode 0

2) **Mode 1** :- i) In this port A & port B can be used as input / output port with handshaking capabilities.

ii) Handshaking capabilities are provided by the bits of port C.

3) **Mode 2** :- i) In this mode port A can be used as bidirectional I/O port with handshaking capabilities, provided by port C

ii) Port B can be used as simple I/O mode or handshaking model.

4) **BSR** (bit set / reset) mode :- i) In this mode only the individuals bits of port C can be programmed.

ii) It can be also used for enabling / disabling interrupt

⇒ i) All ports of A, B, C are o/p ports mode 0.

D7	D6	D5	PA	PCU (PCH)	D2	PB	PCL

mode selection (D6, D5)

1	0	0	0	0	0	0	0

= 10000000 = 80h

ii) PA = in, PB = out, PCL = out, PCH - out mode0

D7	D5 & D6		PA	PCH	D2	PB	PCL.
1	0	0	1	0	0	0	0

= 10010000 = 90h

---

3) Explain the features of 8255.

features of 8255 are:-

i) 8255 A is a widely used, programmable, parallel I/O device.

ii) It is compatible with all Intel and most other microprocessors

iii) It is completely TTL compatible.

iv) It has three 8 bit ports : port A, B & C, which are arranged in threegroups of 24 pins.

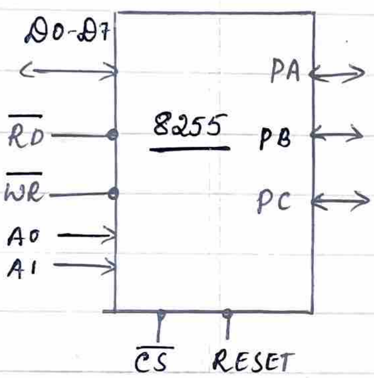
v) 8255 can operate in 3 I/O modes: i) mode0 ii) mode 2 iii) mode 3.

vi) It can be programmed to transfer data under various conditions, from simple I/O to interrupt I/O

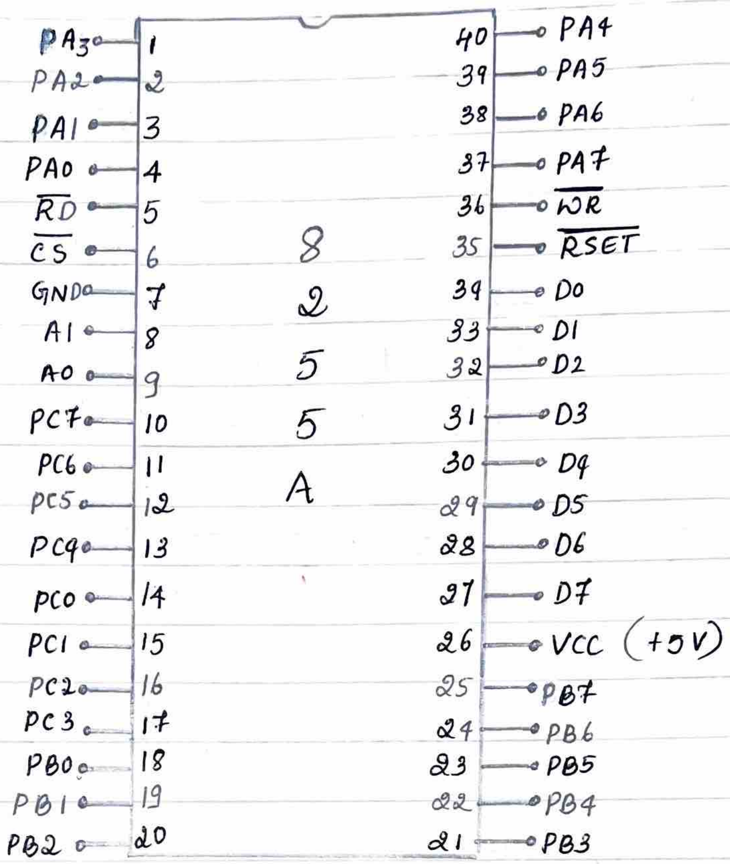
vii) Its BSR mode allows to set/reset the bits of port C.

viii) All I/O pins of 8255 has 2.5 mA DC driving capacity

4. Draw the pin diagram of 8255 chip & explain the signals.



8255 block diagram



Pin diagram of 8255.

PA0-PA7 (port A) :- These are 8 bit I/o port pins of port A, used to send data to the output device and to recieve the data from input device.

PB0- PB7 (port B) :- These are 8 bit I/o (bidirectional) pins of port B, used to send data to the o/p device & recieve data from the i/p device.

PC0- PC7 (port C) :- These 8 bit directional I/o pins of port C can be further devided into 2 groups: PCL and PCU. These groups can individually transfer data in or out when programmed for simple I/O.

$\overline{RD}$ and $\overline{WR}$:- These 2 active-low controls signals are inputs to the 8255. using this pins CPU can read / write, data on the ports.

A0 and A1 :- Along with $\overline{RD}$ & $\overline{WR}$, these A0 & A1 signals control the selection of control/ status word registers.

$\overline{CS}$ (chip select) :- This is an active low input which can be used for data transfer operation b/w CPU & 8255.

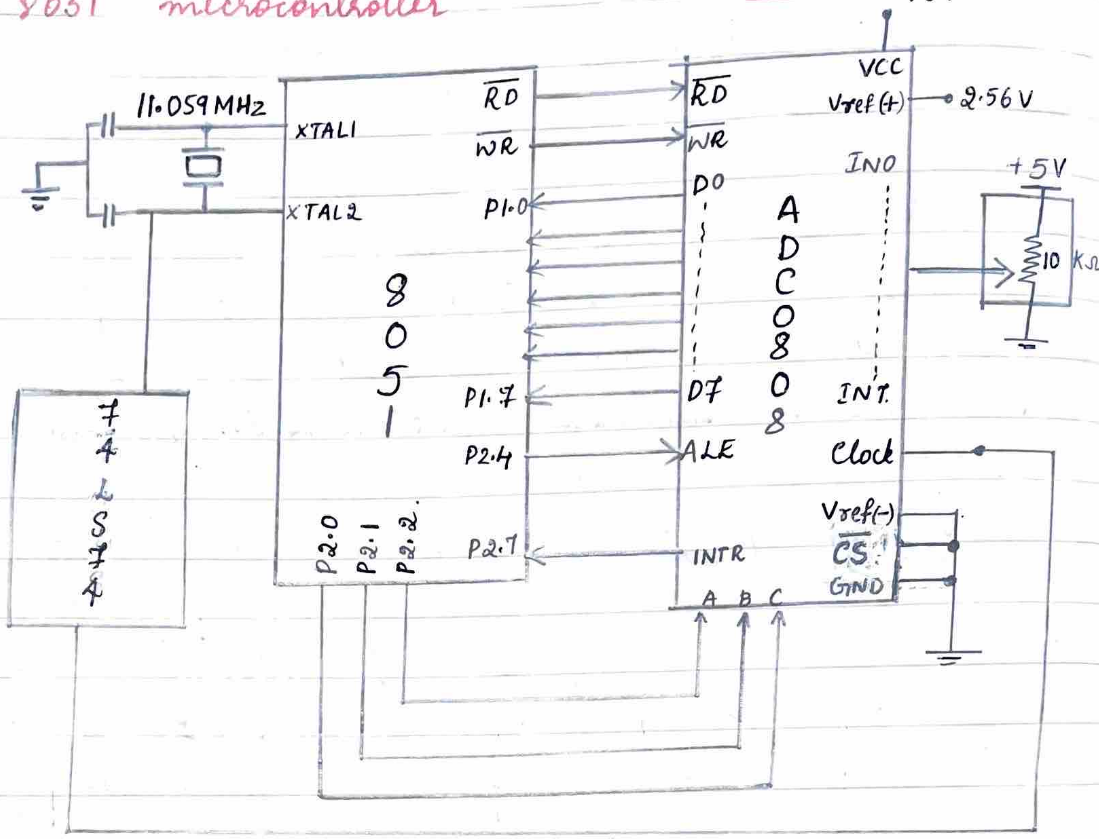
$D_0 - D_7$ (databus) :- These bidirectional, tristate databus lines are connected to system databus.

RESET :- This is an active high input into 8255. , it is used to clear the control register & all ports are initialized as input ports.

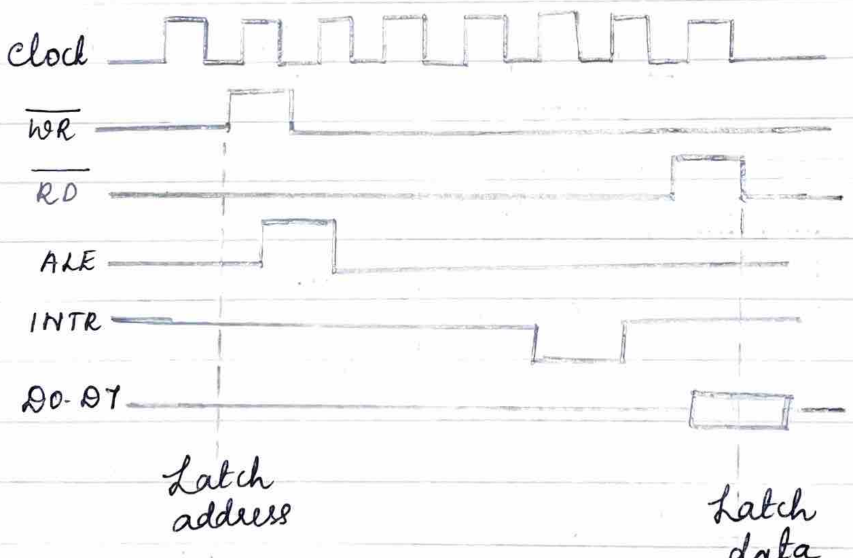
A1	A0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	Operation
					input operation (read)
0	0	0	1	0	Port A to databus
0	1	0	1	0	Port B to data bus
1	0	0	1	0	Port C to databus.
					Output operation (write)
0	0	1	0	0	databus to port A
0	1	1	0	0	" port B
1	0	1	0	0	" port C
1	1	1	0	0	" control register
					Disable function
X	X	X	X	1	data bus tri state
1	1	0	1	0	Illegal condition
X	X	1	1	0	data bus tri state

Signal selection

5) Explain the steps to interface ADC 0808 (ADC 0809) to the 8051 microcontroller



ADC 0808 interfacing with 8051

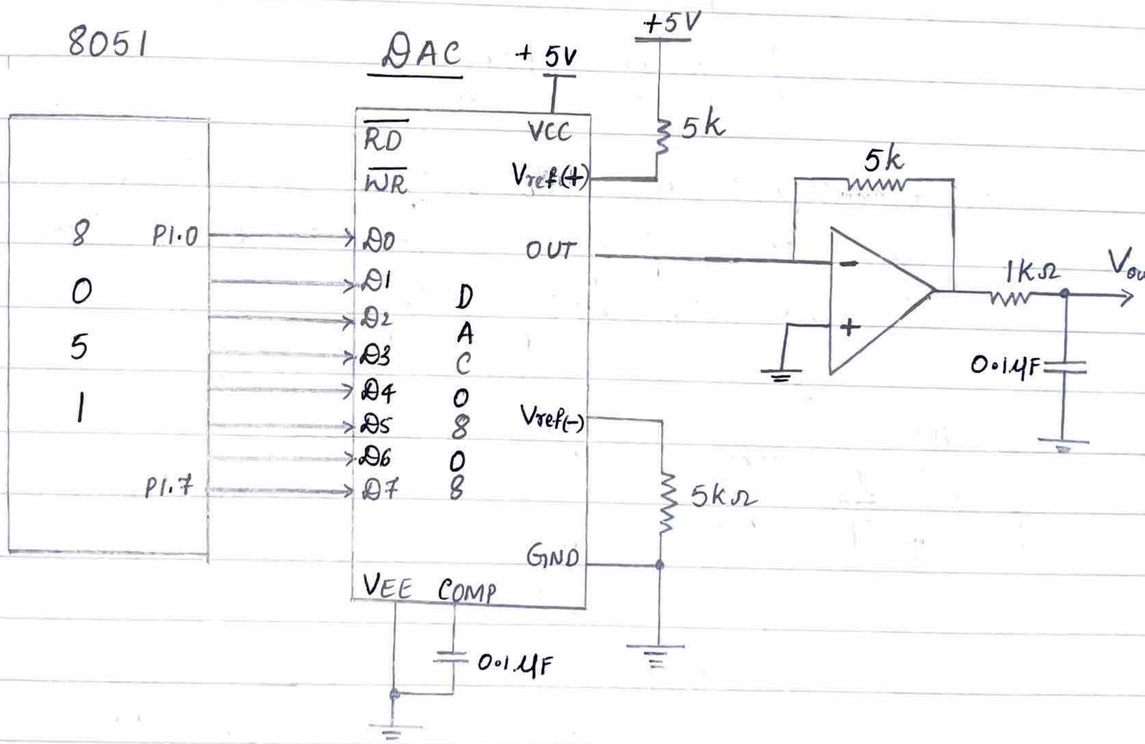


Steps to interface ADC 0808 to 8051

i) Select an analog channel by providing bits to A, B and C addresses

ii) Activate ALE pin
iii) Activate SC (start conversion) by L-H pulse.
iv) Monitor EOC (end of conversion) to see whether conversion is finished
v) Delay size depends on the external clock connected to clock pin
vi) Activate OE (output enable) to read data out of the ADC chip.

✗ 6) Explain with a diagram the interfacing of DAC 0808 with 8051 chip



DAC 0808 interfacing with 8051

- DAC - Digital to analog converter is a device widely used convert digital pulses to analog signals.
- In DAC 0808 the digital inputs are converted to

(I_{out}) and by connecting a resistor to the Iout pin we convert the result to voltage.

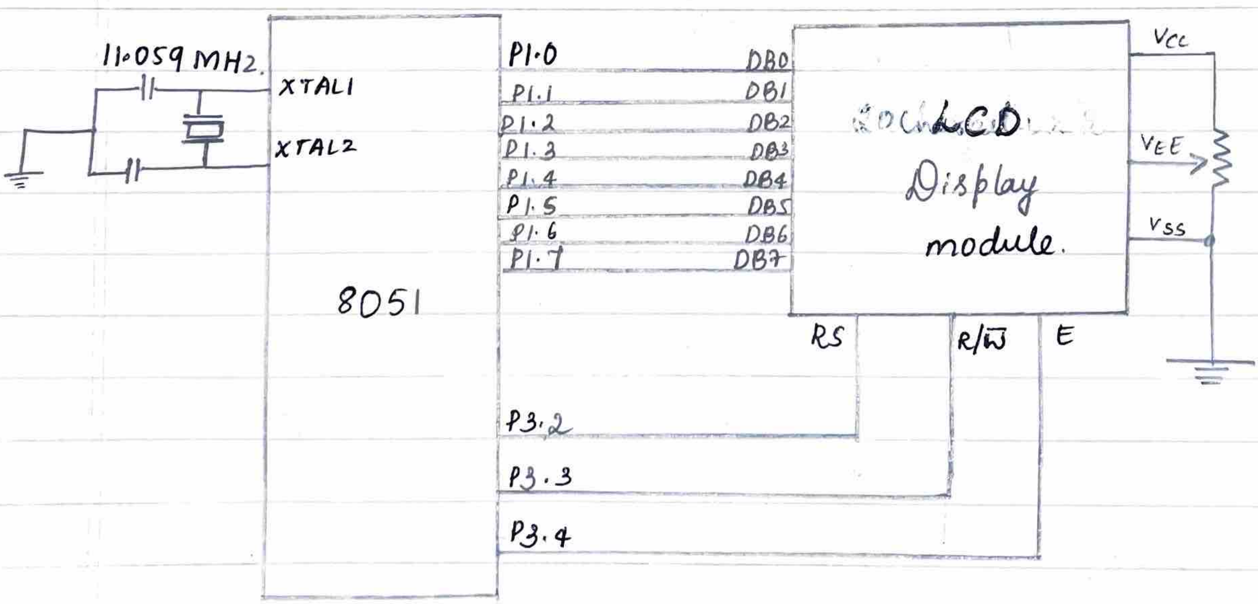
- Total current Iout is given by,

$$I_{out} = I_{ref}\left(\frac{D_7}{2} + \frac{D_6}{4} + \frac{D_5}{8} + \frac{D_4}{16} + \frac{D_3}{32} + \frac{D_2}{64} + \frac{D_1}{128} + \frac{D_0}{256}\right)$$

where D_0 is the LSB & D_7 is the MSB. Iref is the input current to pin 14. Generally Iref is 2mA.

- Iref is isolated by connecting it to an op-amp such as the 741 IC. with Rf = 5kΩ for feed back resistor
- Assuming R = 5kΩ, by changing the binary input the o/p voltage changes.

7) Interface an LCD display to 8051 and write an 8051 C program to send letters 'M', 'D' & 'L' to LCD using delays.



Interfacing LCD with 8051

Date / Page

Program:-

```
#include <reg51.h>
sfr lcd = 0x90;
sbit RS = P2^0;
sbit RW = P2^1;
sbit EN = P2^2;
void maien ()
{
lcdm (0x38);
Ms delay (250);
lcdm (0x0E);
Ms delay (250);
lcdm(0x01);
Ms delay (250);
lcdm (0x06);
Ms delay (250);
lcdm (0x86);
Ms delay (250);
data ('M');
Ms delay (250);
data ('O');
Msdelay (250);
data ('E');
}
Void lcdm (unsigned char)
{
lcd = value;
rs = 0;
rw = 0;
en = 1;
Ms delay (1);
en=0;
} return;
```

```
void data (unsigned char value)
{
   lcd = value;
   rs = 1;
   rw = 0;
   en = 1;
   Ms delay (1);
   en = 0;
   return;
}
void Ms delay (unsigned int itime)
{
   unsigned int i, j;
   for ( i=0; i< itime ; i++)
   for ( j=0; j< itime; j++)
}.
```

8 * Write 8051-C program to display message "good luck" in LCD interfacing (Same as previous program)

9 * Interface LED & write a program to send "WELCOME" to LCD display using delays

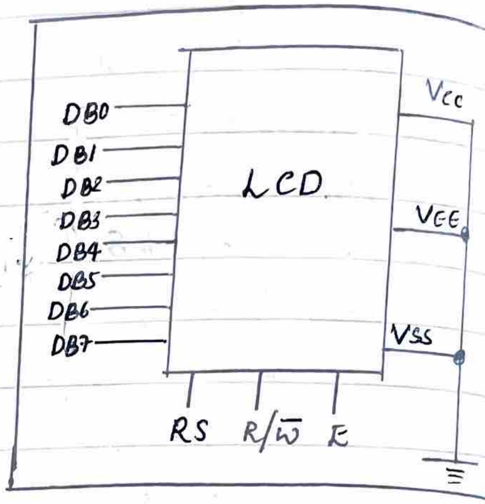
ALP- LCD

```
org 0000h
mov 81h , # 30h
mov A, # 38h
Lcall loop
mov A, #0Eh
Lcall loop
```

```
        mov A, #01;
        L call loop.
        mov A, #06
        L call loop
        mov A, #86
        Lcall loop.
        mov A', #'W'
        L call display
        mov A, # 'E'
        Lcall display
        mov A, # 'L'
        Lcall display
        mov A, # 'C'
        L call display
        mov A, #'O'
        L call display
        mov A, #'M'
        L call display
        mov A, #'E'
        Sjmp here
loop:   L call up
        mov P1, A.
        clr P3.2
        clr P3.3
        clr P3.4
        set. P3.4
up:     Clr P3.4
        clr P3.2
        mov P1, #offh.
read:   setb P3.3
        set b P3.4
        JB P1.7, read.
        clr P3.4
display: Lcall ready
        mov P1, A.
        setB P3.2.
        Clr P3.3.
        set B P3.4
        Clr P3.4.
        RET.
```

* Pin discription for LCD.

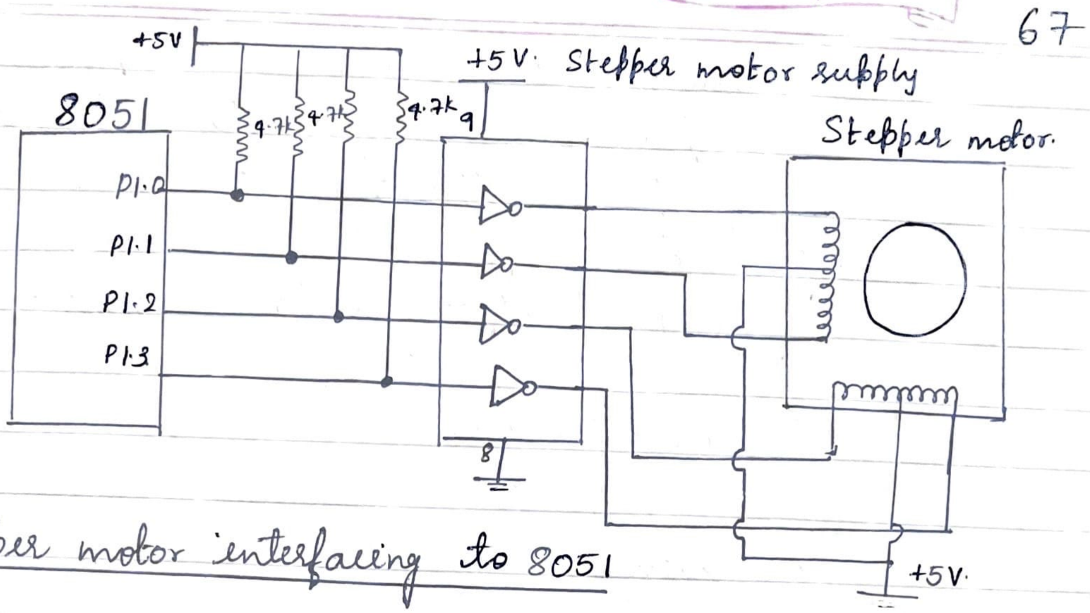
1) VSS — Ground pin.
2) VCC — +5V power supply
3) VEE — Power supply to control contrast.
4) RS — Register select i/p pin
   RS = 01 → data register is selected
   RS = 0 → command register is selected.
5) R/$\overline{W}$ — Input pin which allows the user to write data to the LCD & read data from LCD.
6) E — This input pin is used by LCD to latch information
7) DB_0-DB_7 — I/O pins — These 8-bit data bus pins are used to send or to read information for LCD.



---

10 * Write 8051 ALP to rotate a stepper motor 64° in clockwise direction. The motor step angle is 2°. Use 4 step sequence & draw the schematic diagram step per revolution = 180° No of rotor tooth = 45. movement per 4 step sequence = 8°

Given:- step angle = 2°
one revolution = 360°
movement per 4 step = $4 \times 2° = 8°$
steps per revolution = $360/2 = 180°$
To rotate motor 64° = $\frac{64}{8} = 8$ (counts)



Stepper motor interfacing to 8051

ALP

```
      org 0000h
      mov R2, #08h
up:   mov A, #01h
      mov P2, A
      A call delay
      mov A, #02h
      mov P2, A
      A call delay
      mov A, #04h
      mov P2, A
      A call delay
      mov A, #08h
      mov P2, A.
      A call delay
      DJNZ R2, up.
      mov R3, #100
here: mov R4, #255
loop: DJNZ R4, loop
      DJNZ R3, here
      End.
```

```
#include <reg51.h>
void delay(unsigned int);
void main (void)
{
unsigned char x;
for (x=0; x<8; x++)
{
P2 = 0x01;
delay(1);
P2 = 0x02;
delay(1);
P2 = 0x04;
delay(1);
P2 = 0x08;
delay(1);
}
}
void delay (unsigned int, time)
{
unsigned int i, j;
for (i=0; i< 200; i++)
for (j=0; j< 1275; j++)
}
```

11 * A switch is connected to pin P2.7 write a C program to monitor the status of sw & perform the following

1) If sw=0 stepper motor rotates in clockwise
2) If sw=1 stepper motor rotates in counter clock[wise]

```
# include < reg51.h>
sbit SW= P2^7;
Void main (void)
{
 sw=1;
 while(1)
 {
  if (sw==0)
  {
   P1 = 0x66;
   Ms delay ( );
   P1 = 0xCC;
   Ms delay ( );
   P1 = 0x99;
   ms delay ( );
   P1= 0x33 ;
   ms delay ( );
 }
 else
  {
   P1= 0x66;
   ms delay ( );
   P1= 0x33;
   Ms delay ( );
   P1= 0x99;
   Ms delay ( );
   P1= 0xCC;
   Ms delay ( );
  }
 }
Void Ms delay (unsigned int time)
{
 unsigned int i,j;
 for (i= 0; i< time ; i++)
 for (j=0; j< 1275; j++)
}
```

Date / Page

12 * Consider 8 bit ADC Assume $V_{ref} = 5V$ Calculate 8 bit digital o/p when $V_{in} = 3V$.

$V_{ref} = 5V$

$V_{in} = 3V$.

WKT, $D_{out} = \frac{V_{in}}{\text{Step size}}$

$\text{Step size} = \frac{V_{ref}}{256} = \frac{5}{256.} \doteq 19.5mV$

$D_{out} = \frac{3}{19.5m} = 153.6 \, A. \simeq 154A$

$\therefore D_{out} = 9Ah$

---

13 * For a given ADC 0808 we have $V_{ref} = 2.56V$ calculate $D_0 - D_7$ output (D_{out}) if analog input is 1.7V

$V_{ref} = 2.56V$

$V_{in} = 1.7V$

WKT, $D_{out} = \frac{V_{in}}{\text{Step size.}}$

$\text{Step size} = \frac{V_{ref}}{256} = \frac{2.56}{256} = 10mA$

$\therefore D_{out} = \frac{1.7}{10m} = 170$ (in decimal)

$\therefore D_7 - D_0 \Rightarrow 1010\ 1010.$ (binary)

---

14 * Explain about stepper motor interface with diagram & also write a C program if motor takes 90° steps to complete one revolution & show the calculation.

⇒ total no of steps = 90 / revolution

⇒ Step Angle = $\frac{360°}{\text{total no of steps / revolution}}$

⇒ Step Angle = $\frac{360}{90}$ = 4°

⇒ One revolution = 360°

⇒ Steps per 4 revolution = 4 × 4 = 16° (sequence)

⇒ No of counts = $\frac{\text{total steps}}{\text{sequences}} = \frac{90}{16}$ = 6 count

```
#include <reg51.h>
void Ms delay (unsigned int);
void delay (void)
{
 unsigned char x;
 for (x= 0; x< 5; x++);
}
```

(Same previous program) Q NO - 10

15. Write a program to produce square wave & triangular wave using DAC interface

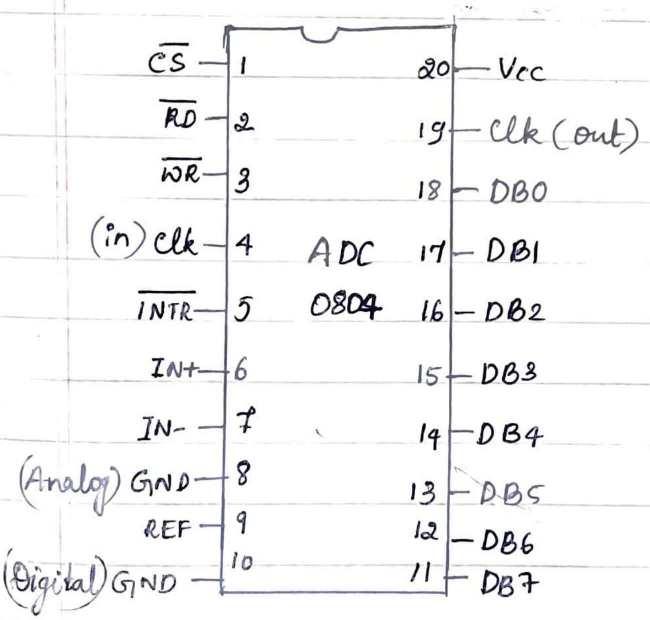
Square wave

```
        org 0000h
        mov SP, #08h.
repeat: mov P1, #offh
        L call delay.
        mov P1, #00h
        L call delay
        Ljmp repeat
delay:  mov R0, # offh
back:   DJNZ R0, back.
        RET.
```

Triangular wave

```
        org 0000h
        mov SP, #08h
        mov R0, # 00h
repeat: mov P1, R0.
        inc R0.
        cjne R0, # offh, repeat
loop:   mov P1, R0.
        DJNZ. R0, loop.
        Ljmp repeat.
```

16. Explain internal Architecture of AD0804 also Draw its timing diagram.



Features

* 8 bit successive ADC.
* Conversion time 100 µs
* Access time 135 ns.
* It has on chip clock generator
* It operates of 5 V power supply

Timing diagram to convert analog data to digital