Important Instructions to examiners:

1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills.
4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q.1. A) Attempt any THREE of the following: 12M

(a) Write any four features of 8051 micro controller.

Ans: (Any 4 Features: 1 mark each)

8051 microcontroller features:

- Eight-bit CPU with register A (the accumulator) and B.
- Sixteen-bit program counter(PC) and data pointer (DPTR).
- Eight-bit program status word (PSW).
- Eight-bit stack pointer (SP).
- Internal ROM or EPROM (8051) to 4K (8051).
- Internal RAM of 128 bytes:
  - Four register banks, each containing eight registers.
  - Sixteen bytes, which may be addressed at the bit level.
  - Eighty bytes of general-purpose data memory.
- Thirty-two input/ output pins arranged as four 8-bit ports: P)-P3.
- Two 16-bit timer/counter: T) and T1.
- Full duplex serial data receiver/ transmitter : SBuf.
- Control registers: TCON,TMOD,SCON,PCON,IP,and IE.
- Two external and three internal interrupt sources.

- Oscillator and clock circuits.

(b) State function of following pins of 8051 microcontroller.

1) **$\overline{\text{PSEN}}$**
2) **ALE**
3) **$\overline{\text{EA}}$**
4) **$\overline{\text{INTO}}$**

Ans: (1 Mark each)

1. $\overline{\text{PSEN}}$

It is active low output control signal used to activate enable signal of external ROM/ EPRM . it is activated every six oscillator periods while reading the external memory.

2. ALE: (Either Diagram or Explanation)

There are two ALE pulses per machine cycle. The Ale pulse, which is primarily used as a timing pulse for external memory access, indicates when every instruction byte is fetched.

3. **$\overline{\text{EA}}$** :

It is active low output control signal. When EA = 1, μc accesses internal and external program memory when EA =μc accesses only external program memory.

4. $\overline{\text{INTO}}$

It is external interrupt 0, active low pin.

It is external hardware interrupt I/P signal. Through this user, programmer or peripheral interrupts to microcontroller. This is connected to port pin p3.2

(c) Describe 'sbit' and 'sfr' data types used in C programming for 8051.

Ans: (2 Mark each)

sbit

With typical 8051 applications, it is often necessary to access individual bits within an SFR. The C51 compiler makes this possible with the **sbit** data type. The **sbit** data type allows you to access bit-addressable SFRs. For example:

```
sbit EA = 0xAF;
```

This declaration defines EA to be the SFR bit at address 0xAF. On the 8051, this is the *enable all* bit in the interrupt enable register.

sfr Types:

SFRs are specified as **sfr** . For example:

sfrP0 = 0xFF00; /* Port-0, address FF00h */

sfr P1 =0xFF04; /* Port-1, address FF04h */

Names for **sfr** types are defined using the same method as other C variables. In the above example, **P0** and **P1**are the SFR name declarations. Any symbolic name may be used in an **sfr** declaration.

The address specification after the equal sign ('=') must be a numeric constant.

Data type	Bits	Bytes	Value Range
sbit	1		0 or 1
sfr	8	1	0.255

Note: Any relevant example should be given marks

(d) List control signals of LCD display and state their functions.

Ans: (List signals 1 Mark, Each pin function 1 Mark)

List : RS, R/W, EN

Functions :

1) **RS:** RS is the register select pin used to write display data to the LCD (characters), this pin has to be high when writing the data to the LCD. During the initializing sequence and other commands this pin should low.

2) **R/W:** Reading and writing data to the LCD for reading the data R/W pin should be high (R/W=1) to write the data to LCD R/W pin should be low (R/W=0)

3) **EN:** Enable pin is for starting or enabling the module. A high to low pulse of about 450ns pulse is given to this pin.

B) Attempt any ONE of the following: 6M

(a) Draw the organization of data memory(RAM) of 8051 and describe in brief.

Ans: (Diagram 4 Mark(1 M for working register, 1 M for General purpose, 2 M for bit Addressable) Explanation 2 Mark)

Diagram:

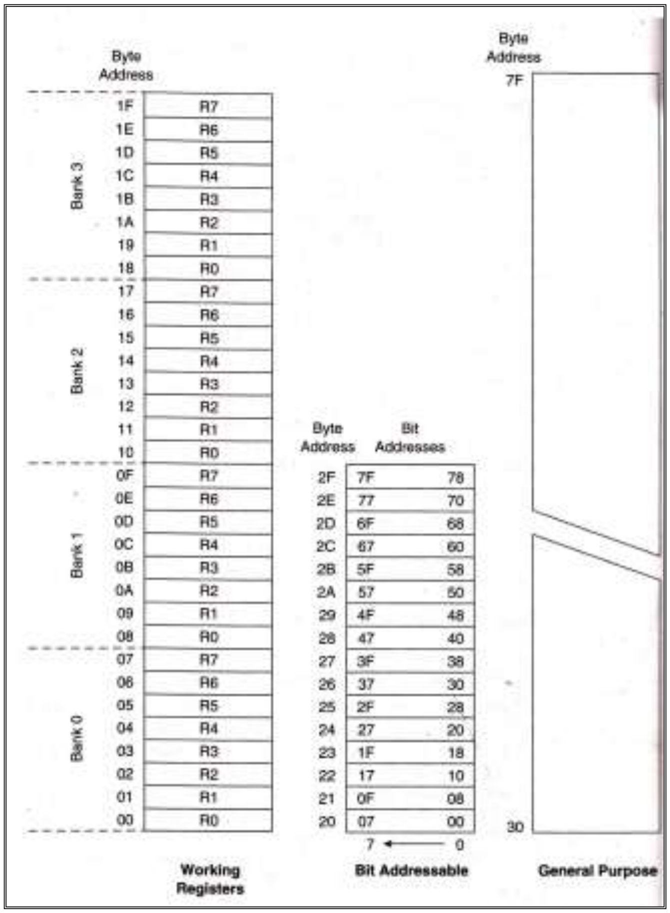


Fig. Internal RAM organization

Explanation:

8051 has 128-byte internal RAM. Which is organized into 3 distinct areas.

- Thirty-two bytes from address 00h to 1Fh that make up 32 working registers organized as four banks of eight registers each. The four register banks are numbered 0 to 3 and are made up of eight registers named R0 to R7.
- A bit-addressable area of 16 bytes occupies RAM byte addresses 20h to 2Fh, forming a total of 128 addressable bits.an addressable bit may be specified by its bit address 00h to7Fh.
- A general-purpose RAM area above the bit area, from 30h to 7Fh.
- A general-purpose RAM area above the bit area, from 30h to 7 Fh, addressable as bytes.

(b) Describe following instructions with reference to their function and addressing mode:

1) **ADD A,R1**
2) **SUBB A,R0**
3) **ORL A, # 30H**

Ans: (Each instruction 2 Mark, 1 M for function, 1 M for addressing mode)

1) **ADD A,R1**

Add contents of R1 to Accumulator and store the results into accumulator

OR

$(A) + (R1) \longrightarrow A$

Addressing mode: Register

2) **SUBB A,R0**

Subtract contents of R0,contents of carry flag from accumulator and store the results into accumulator.

$(A)- (R0)-cy \longrightarrow A$

Addressing mode: Register

3) **ORL A, # 30H**

Logically Or contents of accumulator with given data and store the results in accumulator.

Addressing mode: Immediate

Q2. Attempt any Two of the following: 16M

a) Write assembly language program to find largest number of an array containing 16 numbers. Store this number in the internal RAM location 50H. write appropriate comments.

Ans: (Correct Program – 6 Marks And Comments – 2 Marks)

PROGRAM :

To find largest number of an array containing 16 numbers

```
;           16 numbers are stored from 40H internal RAM location

        ORG     0000H               ;START FROM 0000H
        MOV     R0,#40H             ;LOAD THE STARTING ADDRESS IN R0
        MOV     R7,#15              ;LOAD THE COUNTER IN R7
        MOV     A,@R0               ;TAKE NUMBER IN A AFROM ARRAY
        MOV     50H,A               ;STORE THE NUMBER AT 50H
UP:     INC     R0                  ;GO FOR NEXT NUMBER
        MOV     A,@R0               ;TAKE THE NEXT NUMBER
        CJNE    A,50H,CHECK_BIG     ;COMPARE THE NUMBERS
UP1:    DJNZ    R7,UP               ;DEC THE COUNTER BY 1
H:      SJMP    H                   ;TERMINATE THE PROGRAM

CHECK_BIG:
        JNC     EXCHANGE            ;FIND THE BIG NUMBER
        SJMP    UP1

EXCHANGE  :                         ;SAVE LARGEST NUMBER IN 50H
        MOV     50H,A
        SJMP    UP1
```

NOTE: Program may change. Student can also use the other logic.

Please check the logic and understanding of students.

b) Draw interfacing diagram of 7 segment display with Port 1 of 8051 micro controller. Write C program to display BCD numbers from 0 to 9.

Ans: (Interfacing Diagram – 4 Marks And Correct Program – 4 Marks)

Interfacing Diagram:

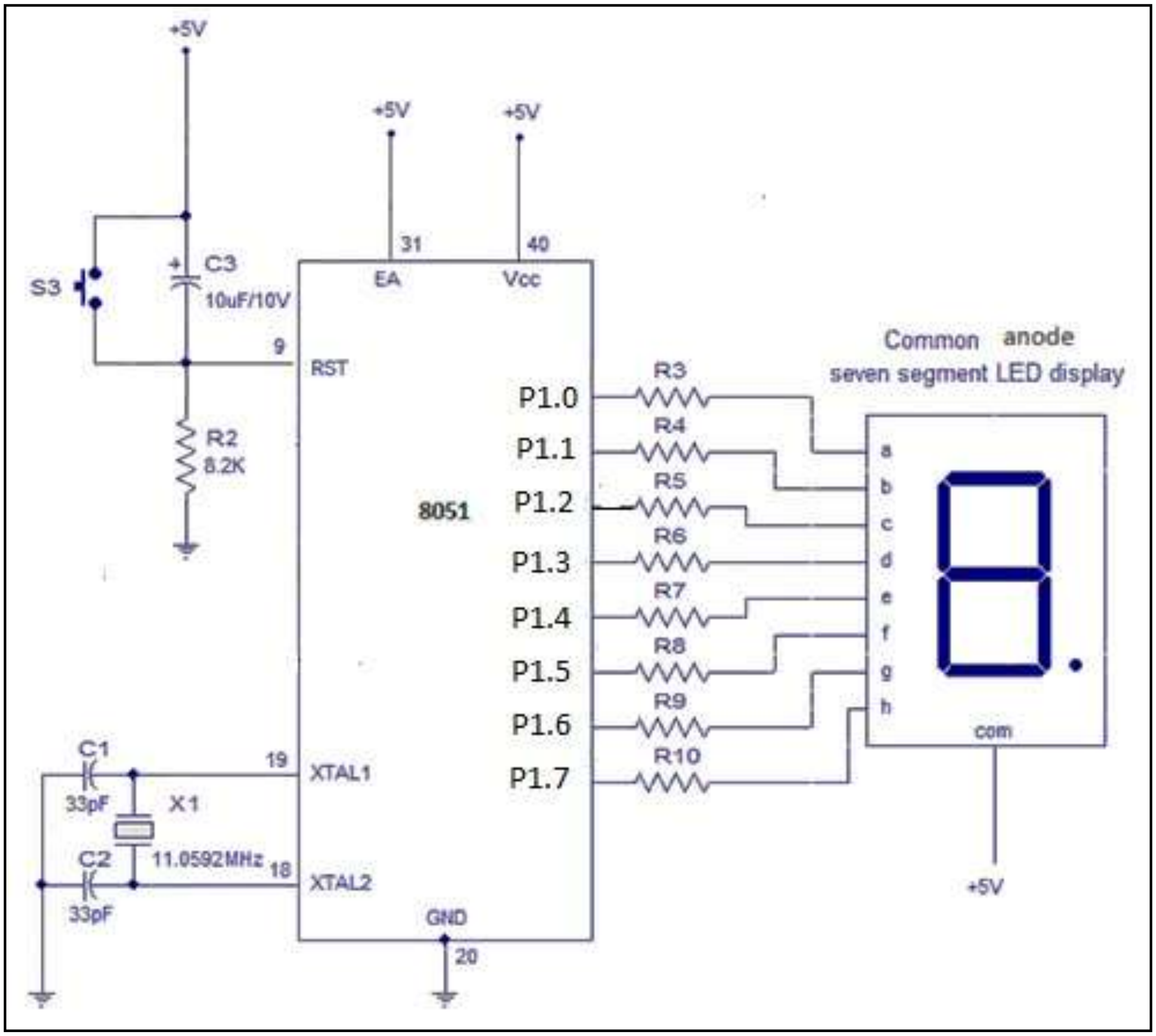


Fig. Interfacing diagram of 7 segment

Program:

```
// C language program for 7 Segment display interfacing

#include <Intel\8052.h>

#include <standard.h>

/*SEVEN SEGMENT DISPLAY

        DP      G       F       E       D       C       B       A

NO      D7      D6      D5      D4      D3      D2      D1      D0

0       0       1       0       0       0       0       0       0       =40H

1       0       1       1       1       1       0       0       1       =79H

2       0       0       1       0       0       1       0       0       =24H

3       0       0       1       1       0       0       0       0       =30H

4       0       0       0       1       1       0       0       1       =19H

5       0       0       0       1       0       0       1       0       =12H

6       0       0       0       0       0       0       1       0       =02H

7       0       1       1       1       1       0       0       0       =78H

8       0       0       0       0       0       0       0       0       =00H

9       0       0       0       1       0       0       0       0       =10H


A = P1.0        B = P1.1        C = P1.2        D = P1.3

E = P1.4        F = P1.5        G = P1.6        DP= P1.7

*/

void main ()

{

        P1 = 0xFF;              //DISPLAY OFF

        while(1)

        {
                P1 = 0x40;              //DISPLAY 0
                delay_ms(1000);
                P1 = 0x79;              //DISPLAY 1
                delay_ms(1000);
                P1 = 0x24;              //DISPLAY 2
                delay_ms(1000);
                P1 = 0x30;              //DISPLAY 3
                delay_ms(1000);
                P1 = 0x19;              //DISPLAY 4
                delay_ms(1000);
                P1 = 0x12;              //DISPLAY 5
                delay_ms(1000);
                P1 = 0x02;              //DISPLAY 6
                delay_ms(1000);
                P1 = 0x78;              //DISPLAY 7
                delay_ms(1000);
                P1 = 0x00;              //DISPLAY 8
                delay_ms(1000);
                P1 = 0x10;              //DISPLAY 9
                delay_ms(1000);
        }
}
```

NOTE: Program may change. Student can also use the other logic.

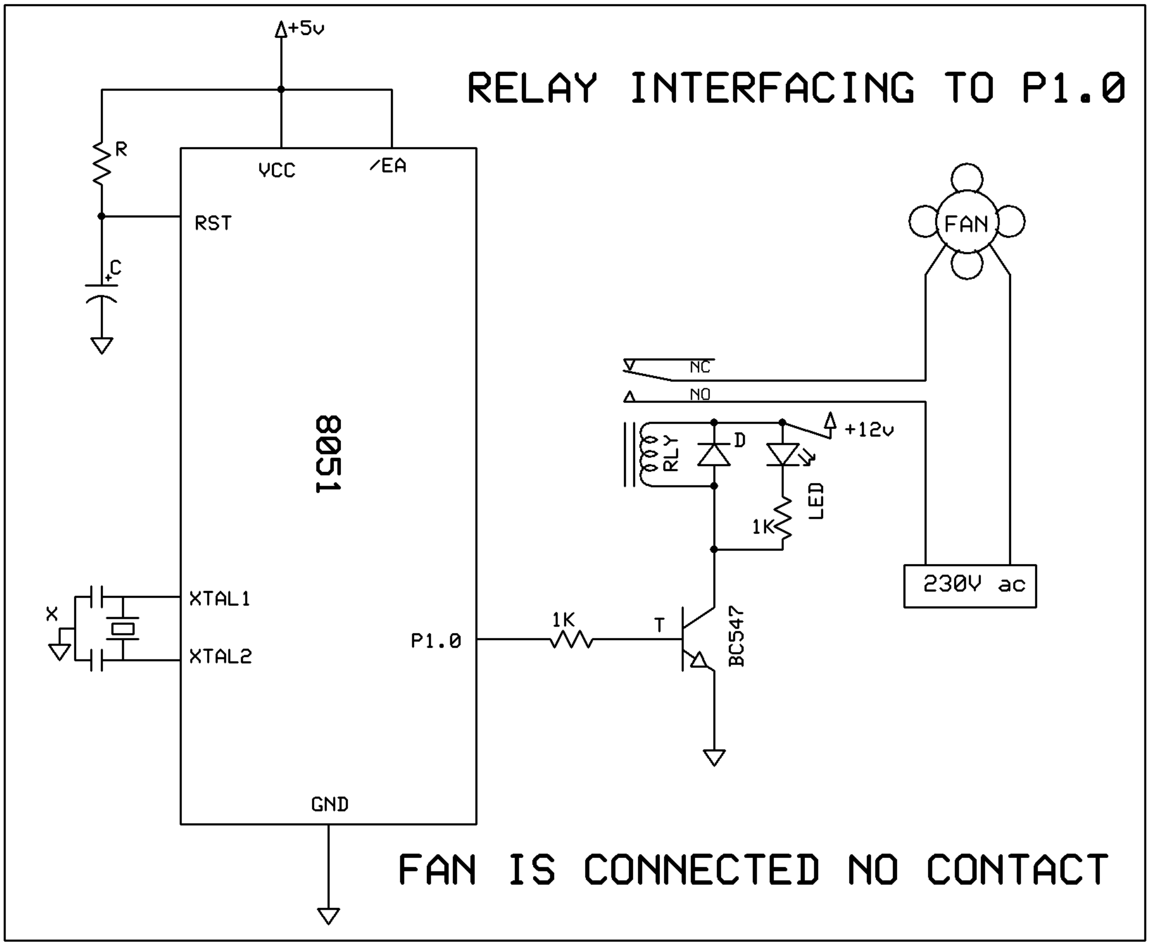
Please check the logic and understanding of students.

c) Draw interfacing diagram where P1.0 pin of 8051 microcontroller is used to control relay contact in turn; controls the fan connected to 230V. Describe operation of this circuit.

Ans: (Interfacing Diagram – 4 Marks And Operation of circuit – 4 Marks)

Any other suitable diagram should be considered.

Interfacing Diagram :



Operation of circuit:

- Fan is operating on 0 to 230Vac and Microcontroller is operating on 0 to 5 Vdc.
- We can't connect FAN directly to microcontroller; therefore we have to connect RELAY in between FAN and Microcontroller.
- RELAY is electromechanical switch. It operates on 12Vdc. Therefore we can't connect RELAY directly to the microcontroller.
- To drive the relay we are using relay driver i.e transistor BC547.

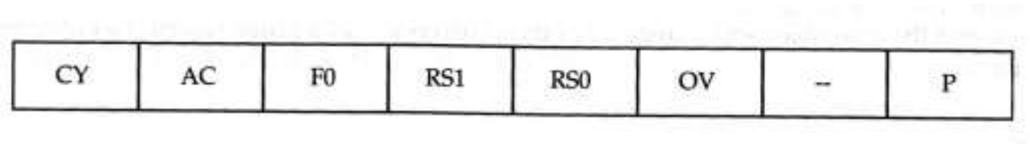
- Microcontroller will operate transistor, Transistor will operate relay and relay will operate fan.
- When P1.0 is logic HIGH i.e 5V, transistor turns ON, Relay turns ON. The NO contact will become NC. And FAN will turns ON.
- When P1.0 is logic LOW i.e 0V, transistor turns OFF, Relay turns OFF. The NO contact remains NO. And FAN will turns OFF.
- LED is used to indicate Relay is ON or OFF.
- Instructions to turn ON and OFF the FAN are as follow:
  - SETB P1.0 is used to turn ON the Relay as well as FAN.
  - CLR P1.0 is used to turn OFF the Relay as well as FAN.

Q.3 Attempt any FOUR: 16 M

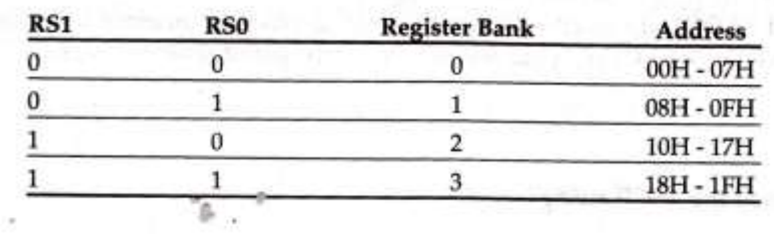
(a) Draw the format of PSW register of 8051 and state the functions of each bit.

Ans: (Format 2 Marks, Function 2 marks)

Format of PSW register of 8051:



CY	AC	F0	RS1	RS0	OV	--	P



RS1	RS0	Register Bank	Address
0	0	0	00H - 07H
0	1	1	08H - 0FH
1	0	2	10H - 17H
1	1	3	18H - 1FH

Functions:

CY, the carry flag-

This flag is set whenever there is a carry out from the D7 bit. This flag bit is affected after an 8-bit addition or subtraction.

AC, the auxiliary carry flag-

If there is a carry from D3 to D4 during an ADD or SUB operation, this bit is set; otherwise, it is cleared. This flag is used by instructions that perform BCD (binary coded decimal) arithmetic.

P, the parity flag-

The parity flag reflects the number of 1s in the A (accumulator0 register only. If the A register contains an odd number of 1s, then P=1. Therefore, P=0 if A has an even number of 1s.

OV, the overflow flag-

This flag is set whenever the result of a signed number operation is too large causing the high-order bit to overflow into the sign bit.

(b) Write alternate functions of port 3 of 8051.

Ans: (each function ½ Mark)

P3 Bit	Function	Pin
P3.0	RxD (Serial data receive pin.)	10
P3.1	TxD (Serial data transmit pin.)	11
P3.2	$\overline{\text{INT0}}$ (External interrupt 0.)	12
P3.3	$\overline{\text{INT1}}$ (External interrupt 1).	13
P3.4	T0 (Clock input for counter 0)	14
P3.5	T1 (Clock input for counter 1)	15
P3.6	$\overline{\text{WR}}$ (Write pin used for external RAM.)	16
P3.7	$\overline{\text{RD}}$ (Read pin used for external RAM.)	17

(c) Describe the following C program for 8051.

Ans: (4 marks comments)

Program: (1 Mark for each comment)

```
# include <reg51.h>…………………..Include header file
Void main (void)
     {
     Unsigned char z; …………………Initialize ‘z’ as a char
     For (z=0; z<=8; z++)……………..increment ‘z’ from 0 till 8
     P1=z;       ………………………...send value of ‘z’ to port 1.
     }           once z > 8……………..end of the program.
```

(d) Compare EEPROM and Flash memory (any four points).

Ans: (each point 1 Mark)

EEPROM:	**Flash memory:**
1.Data is erased using electric current	1.The erasure is caused by flower- nordheium tunneling in which electrons pierce through a thin dielectric material which remove an electronic charge from a floating gate associated with each memory cell.
2.Data is within/ programmed at the byte level.	2.It is done at the block (group of bytes) level.
3.Slower updation (time required 5 ms)	3.Faster (flashes within 10 µs)
4.Program / erase cycle refers to the no. of times that chip can be erased and programmed before it becomes unusable EPROM 1000 times.	4.Flash 100,000 times.
5.No. of cells / unit is less package density is less.	5. No. of cells / unit is less package density is more.

(e) Draw interfacing diagram of 8 key connected to P0 of 8051 and label it.

Ans: (4 Mark for correct diagram)

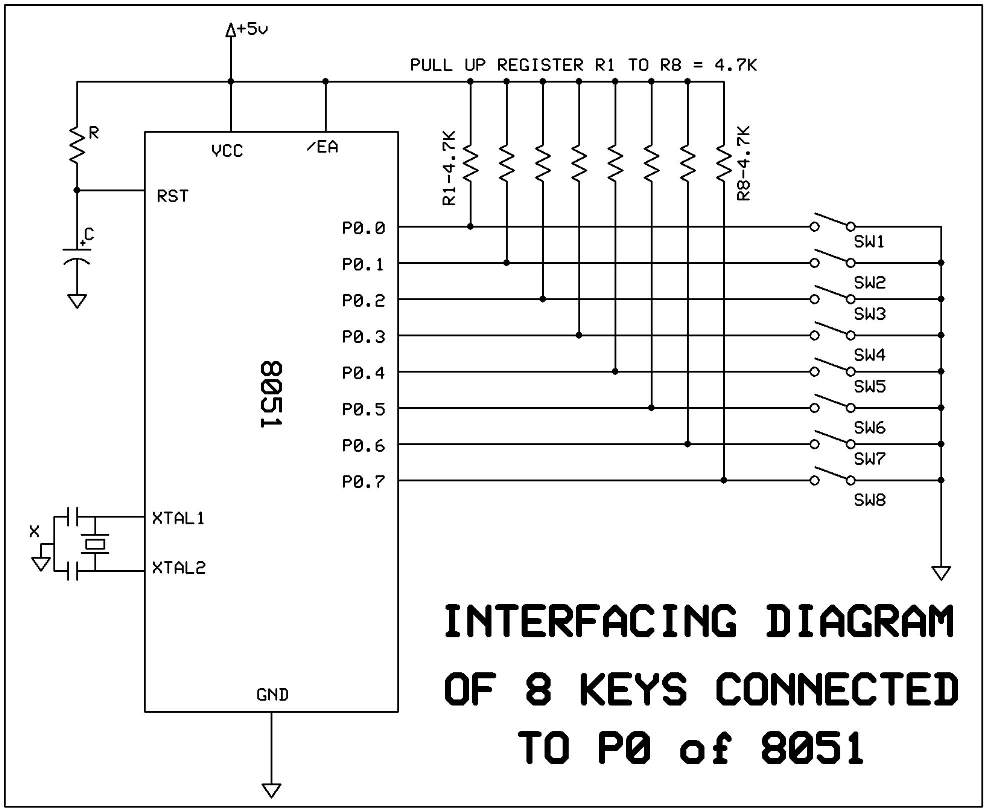


Fig. Key connected to P0

Note. We are using P0 therefore Pull up register is compulsory.

Q4. A) Attempt any three of the following: 12M

a) Draw the interfacing diagram of stepper motor to Port 1. Use ULN2003 driver IC.

Ans: (Interfacing diagram- 4Marks)

Stepper Motor Interfacing Diagram:

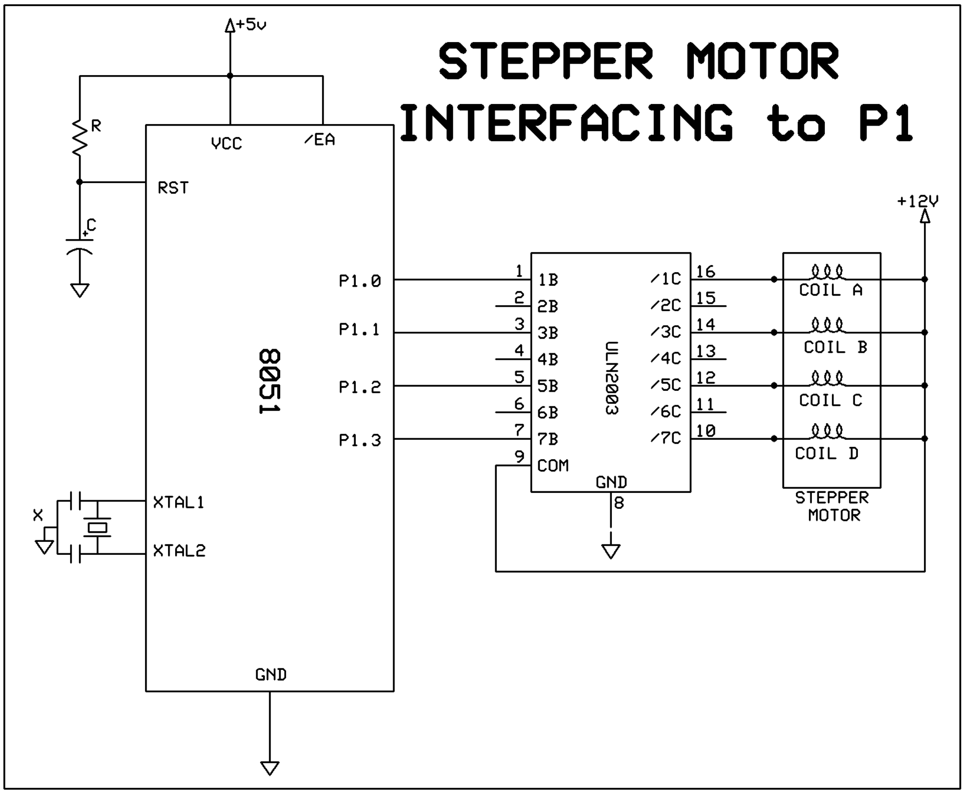


Fig. Stepper motor to Port 1

b) Describe bitwise Left / Right Shift operator used in 8051 C programming with example.
Ans:(Left Shift operator with example – 2 Marks And Right Shift operator with example – 2 Marks)

Bitwise Left Shift Operator in C:

4) It is denoted by <<
5) Bit Pattern of the data can be shifted by specified number of Positions to Left
6) When Data is Shifted Left , trailing zero’s are filled with zero.
7) Left shift Operator is Binary Operator [Bi – two]

Example:

Syntax : Bitwise Left Shift Operator
[variable]<<[number of places]

P0=0x3C << 2

After execution of this instruction
Shift number 2 times to left:
3C=0011 1100
F0=1111 0000
So,
P0=0xF0

Bitwise Right Shift Operator in C:

- It is denoted by >>
- Bit Pattern of the data can be shifted by specified number of Positions to Right
- When Data is Shifted Right , leading zero's are filled with zero.
- Right shift Operator is Binary Operator [Bi – two]

Example:

Syntax : Bitwise Right Shift Operator
[variable]>>[number of places]
P0=0x3C >> 2
After execution of this instruction
Shift number 2 times to Right:
3C=0011 1100
0F=0000 1111
So,
P0=0x0F

NOTE: Example May change. Please check the logic and understanding of students.

c) Compare microprocessor and micro controller(any 4 points).

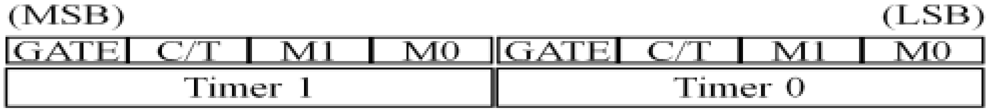
Ans: Comparison : (Any 4 points 4 Marks)

Sr.No	***Microprocessor***	***Microcontroller***
1	Microprocessor don't have inbuilt ROM and RAM	Micro controller has inbuilt ROM and RAM
2	Microprocessor don't have inbuilt Timer and Counter	Micro controller has inbuilt Timer and Counter
3	Microprocessor don't have inbuilt UART	Micro controller has inbuilt UART
4	Microprocessor don't have inbuilt I/O PORTs	Micro controller has inbuilt 4 I/O PORTS
5	It has many instructions to move data between memory and CPU.	It has one or two instructions to move data between memory and CPU.
6	It has one or two bit handling instructions.	It has many bit handling instructions.
7	Access times for memory and I/O devices are more.	Less access times for built-in memory and I/O devices.
8	Microprocessor based system requires more hardware.	Micro controller based system requires less hardware reducing PCB size and increasing the reliability.

d) Draw the format of TMOD and describe each bit.

Ans:- (TMOD Format – 2 Marks And Each bit description – 2 Marks)

TMOD format:



(MSB) (LSB)

GATE	C/T	M1	M0	GATE	C/T	M1	M0
Timer 1				Timer 0			

TMOD
(89h) SFR :

Bit	**Name**	**Explanation of Function**	**Timer**
7	GATE1	When this bit is set the timer will only run when INT1 (P3.3) is high. When this bit is clear the timer will run regardless of the state of INT1.	1
6	C/T1	When this bit is set the timer will count events on T1 (P3.5). When this bit is clear the timer will be incremented every machine cycle.	1
5	T1M1	Timer mode bit	1
4	T1M0	Timer mode bit	1
3	GATE0	When this bit is set the timer will only run when INT0 (P3.2) is high. When this bit is clear the timer will run regardless of the state of INT0.	0
2	C/T0	When this bit is set the timer will count events on T0 (P3.4). When this bit is clear the timer will be incremented every machine cycle.	0
1	T0M1	Timer mode bit	0
0	T0M0	Timer mode bit	0

The
modes of operation are:

TxM1	**TxM0**	Timer Mode	Description of Mode
0	0	0	13-bit Timer.
0	1	1	16-bit Timer
1	0	2	8-bit auto-reload
1	1	3	Split timer mode

Q4. B) Attempt any one : 6M

a) Describe following branching instructions:

1) DJNZ R0,UP

2) CJNE @R1, #80H, LOOP

3) JB P1.5,Here

Ans:- (Each instruction description – 2 Marks)

1) DJNZ R0,UP

Decrement R0 register and if it is not zero then jump to UP lable.

DJNZ instruction decrements the value of register R0 by 1. If the new value of register is not 0 the program will branch to the address indicated by relative address UP. If the new value of register is 0 program flow continues with the instruction following the DJNZ instruction.

2) CJNE @R1, #80H, LOOP

Compare the data (indirectly Pointed by R1) with immediate number 80H. If both numbers are not equal then jump to LOOP label

CJNE compares the value of *operand1i.e* the data (indirectly Pointed by R1) and *operand2, i.e 80H* and branches to the indicated relative address LOOP, if *operand1* and *operand2* are not equal.

If the two operands are equal program flow continues with the instruction following the CJNE instruction.

3) JB P1.5, Here

If port pin P1.5 is set then jump to the address pointed by label Here.

JB branches to the address indicated by *reladdr-Here* if the bit indicated by *bit addrP1.5* is set. If the bit is not set program execution continues with the instruction following the JB instruction.

b) Compare 8051 micro controller with 8052, with reference to

1) **On chip ROM**
2) **On chip RAM**
3) **Timers**
4) **Interrupt Sources.**

Ans: **(Any 1 point 1 ½ Marks*)***

Sr. No.	**Parameter**	**8051**	**8052**
1	**On chip ROM**	4 KB	8 KB
2	**On chip RAM**	128 Byte	256 Byte
3	**Timers**	2	3
4	**Interrupt Sources**	5	6

Q.5.Attempt any TWO **16**

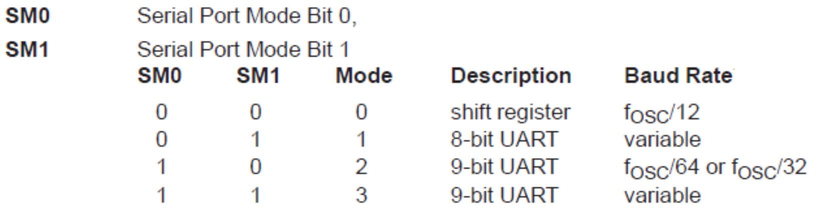
a) Draw the format of SCON register and describe function of each bit. State the importance of SMOD bit when it is set. If XTAL =12 MHZ, Calculate baud rate for

TH1= -12 and SMOD=1.

ANS: (SCON format:2 marks , SCON bit explanation :2 marks , Significance of SMOD bit :2 marks , Baud rate calculation: 2 marks)

SERIAL PORT CONTROL REGISTER :SCON

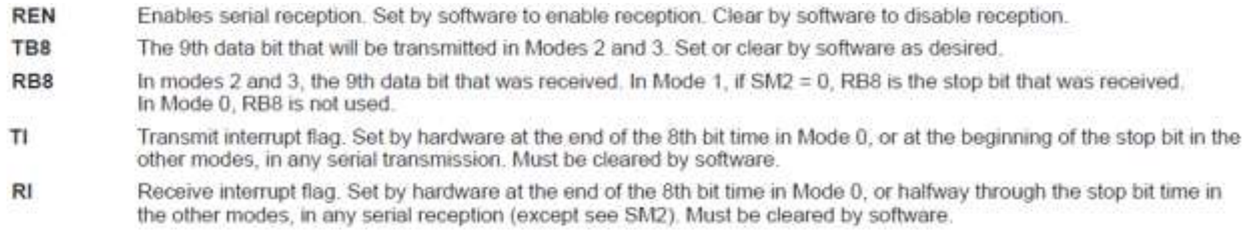
SM0	**SM1**	**SM2**	**REN**	**TB8**	**RB8**	**TI**	**RI**



SM0 Serial Port Mode Bit 0,
SM1 Serial Port Mode Bit 1

SM0	**SM1**	**Mode**	**Description**	**Baud Rate**
0	0	0	shift register	$f_{OSC}/12$
0	1	1	8-bit UART	variable
1	0	2	9-bit UART	$f_{OSC}/64$ or $f_{OSC}/32$
1	1	3	9-bit UART	variable

SM2 enables the multiprocessor communication feature in Modes 2 and 3. In Modes 2 or 3, if SM2 is set to 1 than RI will not be activated if the received 9th date bit(RB8) is 0. In Mode 1, if SM2 = 1 then RI will not be activated if a valid stop bit was not received. In Mode 0 SM2 should be 0



REN	Enables serial reception. Set by software to enable reception. Clear by software to disable reception.
TB8	The 9th data bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired.
RB8	In modes 2 and 3, the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used.
TI	Transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software.
RI	Receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software.

Significance of SMOD bit:

SMOD is PCON.7 bit , which when set to 1 doubles the serial communication baud rate. When 8051 is powered up, SMOD is 0.

$$\textbf{Baud Rate} = \frac{2^{SMOD}}{32} \ \textbf{X (Timer 1 Overflow Rate)}$$

Baud rate calculation for Fosc=12 MHZ , TH1 = -12 , SMOD =1 :

$$\textbf{Baud Rate} = \frac{2^{SMOD}}{32} \ \textbf{X (Timer 1 Overflow Rate)}$$

$$= \frac{2^{SMOD}}{32} \ \text{X} \ \frac{\text{Oscillator Frequency}}{12 \text{ X } [256 - (\text{ TH1 })]}$$

FOR SMOD=1,

Baud Rate = (2/ 32) X (12 MHZ / (12X[256-TH1]))

Baud Rate = (62500 HZ / [256 – TH1])

Given, TH1 = -12 = f4h =244

Baudrate = (62500 HZ / [256-244]) = 62500 / 12

Baudrate = 5208 bps

(b) Write assembly language program for 8051 to generate square wave of 10 KHZ on port pin P1.7 Assume XTAL =12MHZ , use Timer0 to generate delay.(show delay calculation with comments)
ANS: (delay calculation: 2 marks , Program : 6 marks)

Frequency = 10 khz
Therefore Time period T = 1 / 10KHZ = 0.1 ms
Therefore TON = TOFF = 0.1ms / 2 = 0.05 ms = 50 usec

Required time delay = (12 / Fosc) x number of increments (N)
50 usec = (12 / 12MHZ) x number of increments (N)
50usec = 1 usec x N
N = 50

Using TIMER 0 in MODE 1,

COUNT = 2^{16} - N

COUNT =65536 – 50 =65486 = FFCE H

Assembly language program :

ORG 0000H

```
         MOV TM0D,#01H ;TIMER 0, MODE 1
    AGAIN:MOV TH0,#0FFH ;Load higher byte of count
          MOV TL0,#0CEH  ; load lower byte of count
          SETB TR0       ;start timer 0
    HERE: JNB TF0,HERE ;CHECK IF TF0 IS SET
         CPL P1.7    ; Complement P1.7
         CLR TR0      ;  STOP TIMER 0
         CLR TF0     ; CLEAR TF1
         SJMP AGAIN  ;REPEAT AGAIN
           END
```

OR

Using TIMER 0 in MODE 2 (8 bit AUTO RELOAD timer mode),

COUNT = 2^{8} - N

COUNT = 256 – 50 =206 = CE h

Therefore TH0 = TL0 = CE h

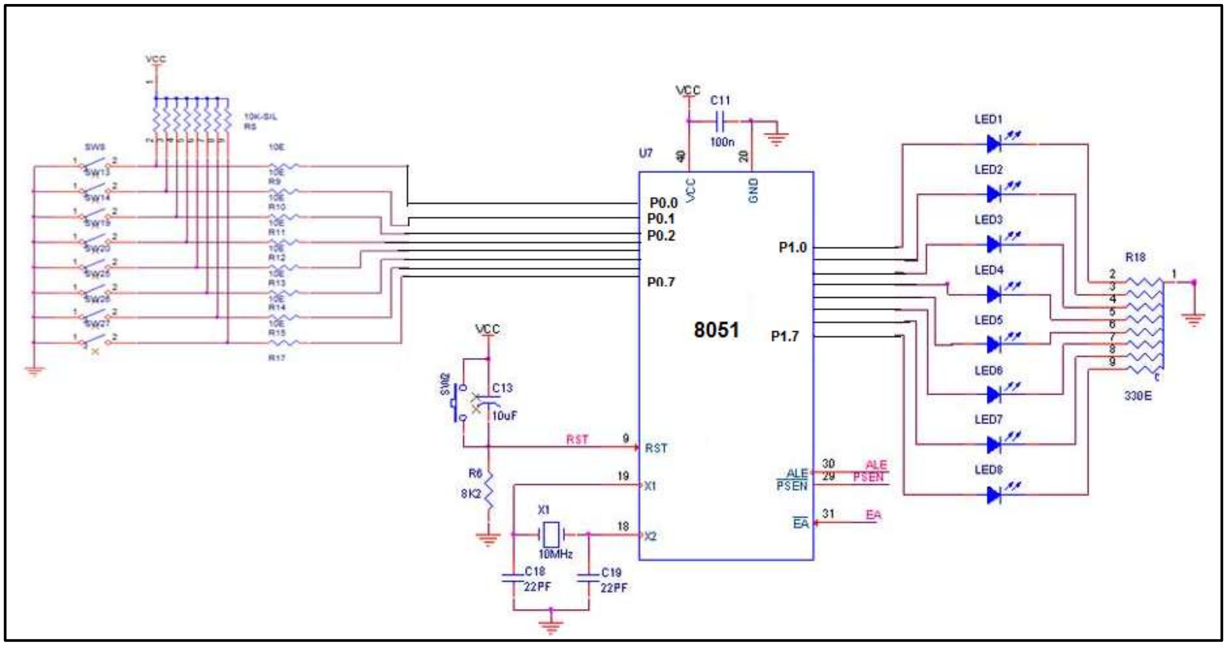
Program:

```
    MOV TMOD,#02H ;Timer0, Mode 2
    MOV TH0,#0CEh ;Load higher byte of count
    SETB TR0      ; start timer0
 HERE:JNB TF0,HERE ;Check if TF0 is set
    CPL P1.7   ;COMPLEMENT P1.7
    CLR TF0      ; CLEAR TF0
    SJMP HERE ;REPEAT
     END
```

Note: (Any other correct program logic can be used)

C) Draw interfacing diagram to interface 8 switches to PORT 0 and 8 LEDs to PORT 1 Write program for 8051 in C language rom read switch status and display it on LEDs

ANS: (Diagram : 4 marks , Program : 4marks)
Any otherc suitable diagram and program related to that should be considered.



(Note : LEDs can be connected in common anode mode also)

'C' Language program:

```
#include <reg51.h>
void main(void)
{
     unsigned char mybyte;
     P0=0xFF; //make Port0 input port
     P1=0X00; // make Port1 output port
     while (1)
     {
     mybyte=P0; //get a byte from P0
     P1= ~ mybyte; //send compliment of it to P1
     }
}
```

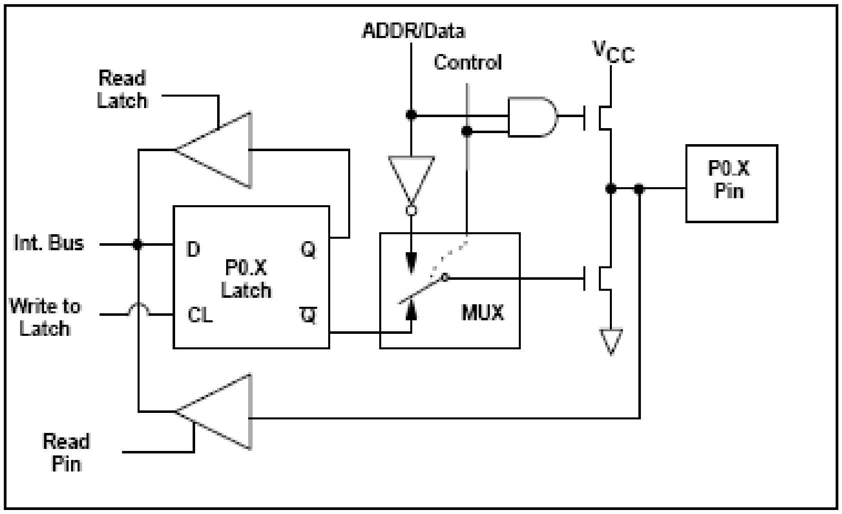
Note: When switch is not pressed port pin of P0 status is logic 1 and when switch is pressed port pin of P0 is logic 0. So we have to complement the status of P0 i.e mybyte variable.

Q.6) Attempt any FOUR **16**

a) Draw diagram of port 0 of 8051 and label it. Write the process to read port 0 pin status.

ANS: (neat labeled Diagram : 2 marks , Explanation: 2 marks)

PORT 0 diagram:

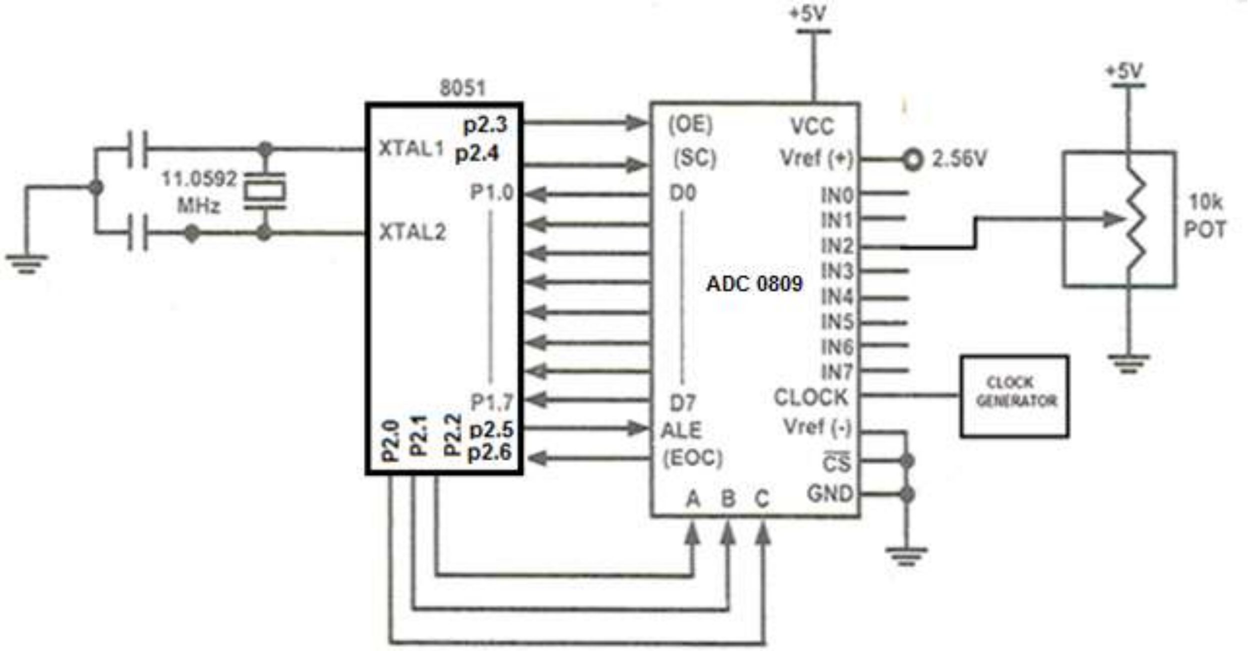


Process to read pin status:

When the port is used as an input port, '1' must be written to the latch. In this situation both the output MOSFETs are 'off'. Hence the output pin floats in high impedance state, and pin is connected to input buffer . Data on the pin can be read by asserting “Read pin “ signal when read pin instruction is executed

b) Draw labeled diagram interfacing diagram of ADC0809 with 8051 microcontroller

Ans: (neat labeled diagram : 4 marks)



C) State addressing modes of 8951. Describe any two addressing modes with example

ANS: Stating addressing modes: 1 mark , explanation of any two addressing modes with one example each : 3 marks (one example each :1/2 mark)

Following are addressing modes of 8051

1. **Immediate addressing mode**
2. **Register addressing mode**
3. **Direct addressing mode**
4. **Indirect / register indirect addressing mode**
5. **Indexed addressing mode**
6. **Register specific addressing mode**

1. **Immediate addressing mode**
   The value of a constant i.e. operand is specified in instruction itself. The data is part of instruction. The immediate data must be preceded by the “#” sign

   MOV A, #64H

   MOV DPTR,#6080H
2. **Register addressing mode**
   Registers are used to hold the operands (data to be operated on). The register banks, containing registers R0 through R7, can be accessed by this addressing mode

   MOV A, R3

ADD A,R4

3. **Direct addressing mode**

In direct addressing the operand is specified by an 8-bit address field in the instruction. i.e. in this mode data is in internal RAM location whose address is specified in the instruction . Only internal Data RAM and SFRs can be directly addressed.

MOV R2, 20H
ADD A,30H
MOV 30H,45H
MOV A , 50H

4. **Indirect / register indirect addressing mode**

In indirect addressing the instruction specifies a register which contains the address of the operand. Registers are used to point towards data. Both internal and external RAM can be indirectly addressed. For internal RAM Only registers R0 or R1 can be used as memory pointers . For external RAM ,16-bit "data pointer" register, DPTR can be used

MOV A,@R0
ANL A,@R1
ADD A,@R0

5. **Indexed addressing mode**

Only program Memory can be accessed with indexed addressing, and it can only be read. This addressing mode is intended for reading look-up tables in Program Memory. A 16-bit register (data pointer/program counter – DPTR/PC) holds the base address of the lookup table and the accumulator holds an 8-bit displacement or index value. The sum of these two registers forms the effective address for the instruction

MOVC A,@A+PC
MOVC A,@A+DPTR

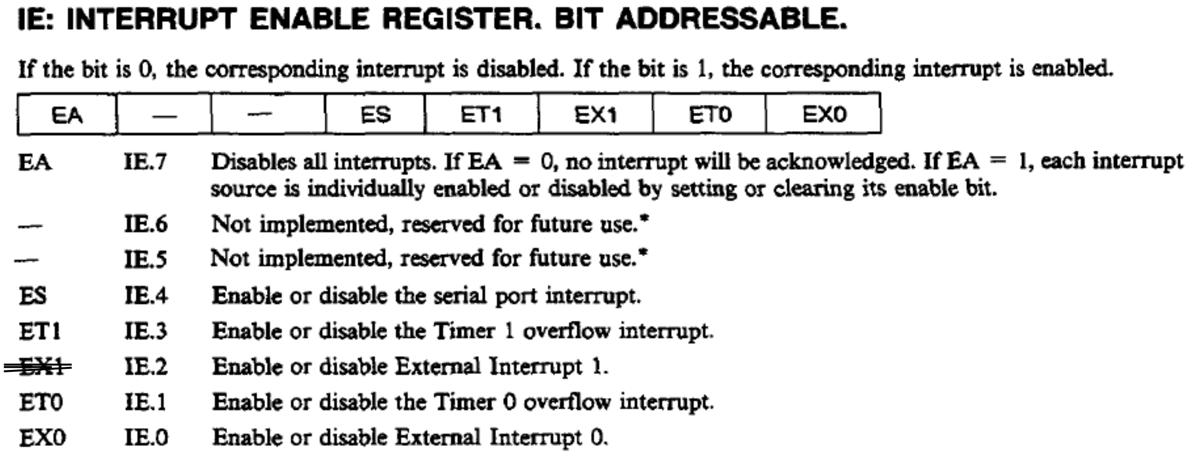
6. **Register specific addressing mode**

Some instructions are specific to a certain register. For example, some instructions always operate on the Accumulator, or Data Pointer, etc., so no address byte is needed to point to it. The opcode itself does that. Instructions that refer to the Accumulator as A assemble as accumulator specific opcodes.

RLC A
RRC A
SWAP A
DAA

d) Draw format of IE register and describe each bit

ANS: format : 2 marks , bit explanation: 2 marks



IE: INTERRUPT ENABLE REGISTER. BIT ADDRESSABLE.

If the bit is 0, the corresponding interrupt is disabled. If the bit is 1, the corresponding interrupt is enabled.

EA	—	—	ES	ET1	EX1	ET0	EX0

EA IE.7 Disables all interrupts. If EA = 0, no interrupt will be acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.

— IE.6 Not implemented, reserved for future use.*

— IE.5 Not implemented, reserved for future use.*

ES IE.4 Enable or disable the serial port interrupt.

ET1 IE.3 Enable or disable the Timer 1 overflow interrupt.

~~EX1~~ IE.2 Enable or disable External Interrupt 1.

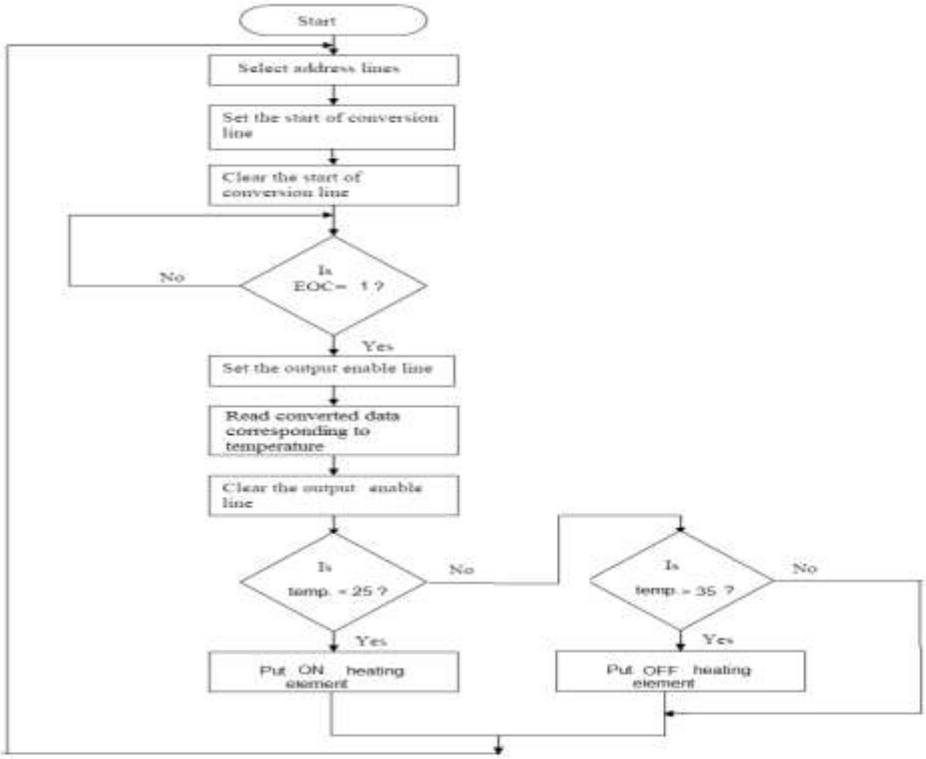
ET0 IE.1 Enable or disable the Timer 0 overflow interrupt.

EX0 IE.0 Enable or disable External Interrupt 0.

e) Assuming temperature control system using LM35 as temperature sensor and ADC 0809. Draw flow chart of this system to keep temperature within
25 ° C to 35° C

ANS: correct flow chart : 4 marks

(Note: Any other correct logic can be used)


Start
Select address lines
Set the start of conversion line
Clear the start of conversion line
No
Is EOC= 1 ?
Yes
Set the output enable line
Read converted data corresponding to temperature
Clear the output enable line
Is temp. = 25 ?
No
Is temp. = 35 ?
No
Yes
Put ON heating element
Yes
Put OFF heating element