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.

i)Compare between Microprocessor and Microcontroller (any FOUR points)

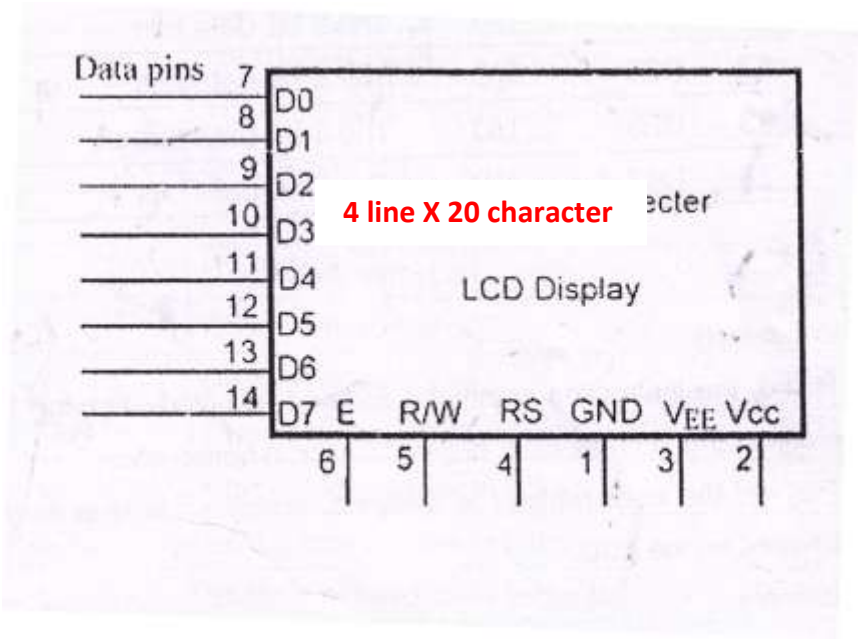
Ans: (any FOUR points 1 mark each)

Sr. No.	Microcontroller	Microprocessor
1.	I/o ports are available.	I/o ports are not available.
2.	Inbuilt timer is present.	No inbuilt timer is present.
3.	Inbuilt RAM and ROM.	Do not have inbuilt RAM and ROM.
4.	Inbuilt serial ports are present.	Inbuilt serial ports are not present.
5.	Separate memory to store program and data.	Program and data are stored in the same memory.
6.	Boolean processor is present.	Boolean processor is not present.
7.	More multifunction pins.	Less multifunction pins.

ii) Draw the pin diagram of 20 × 4 LCD display .what is the function of RS,EN and R/W pins.

Ans) (Diagram 1 mark, function 1 mark each)

Diagram:



Functions:

RS(register select)

If RS=0, the instruction command code register is selected , allowing the user to send the command such as clear display, cursor at home, etc.

If RS=1, the data register is selected ,allowing the user to send data to be displayed on the LCD.

R/W (read/write)

- R/W input allows the user to write information to the LCD or read information from it.
- R/W=1 when reading ;R/W=0 when writing.

EN(enable)

- The enable pin is used by the LCD to latch information presented to its data pin.
- When data is supplied to its data pin, a high-to-low pulse must be applied to this in order for the LCD to latch in the data presented at the data pins.

  This pulse must be a minimum of 450 ns wide

iii) What are the different data types used in C? Also give their value range.

Ans) (any FOUR data types 1 mark each)

Data types used in C:

i)Unsigned character

range :– 0-255

2)Signed character

range :- (-128+0+127)

3)Unsigned integer

Range :- 0-65535(0000-FFFFH)

4)signed integer

range ;- (-32768 to 32767)

5) Bit

Range RAM bit addressable only

6)SFR

Range RAM addresses 80 –FFH only

7) Sbit

Range SFR bit addressable only

iv) Draw and explain reset circuit used for 8051 microcontroller

Ans) (Diagram 1 mark ,Explanation 3 mark)

Diagram:

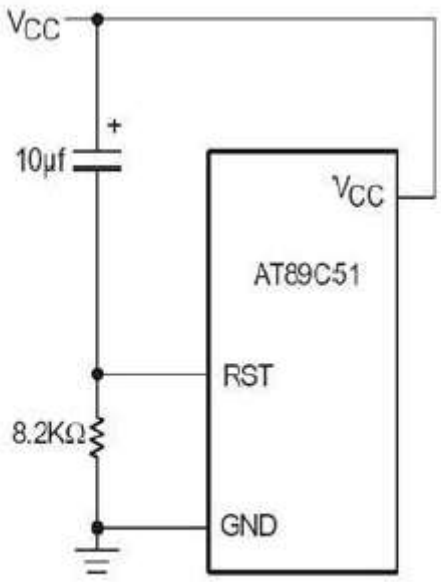


Figure Power-On Reset Circuit

Power-On Reset

For CMOS devices, the external resistor can be removed because the RST pin has an internal pulldown. The capacitor value can then be reduced to 1 µF in Figure .

When power is turned on, the circuit holds the RST pin high for an amount of time that depends on the capacitor value and the rate at which it charges. To ensure a valid reset, the RST pin must be held high long enough to allow the oscillator to start up plus two machine cycles.

On power-up, V_{CC} should rise within approximately 10 ms. The oscillator start-up time depends on the oscillator frequency. For a 10 MHz crystal, the start-up time is typically 1 ms. For a 1 MHz crystal, the start-up time is typically 10 ms.

With the given circuit, reducing V_{CC} quickly to 0 causes the RST pin voltage to momentarily fall below 0V. However, this voltage is internally limited and will not harm the device.

.

Q. 1 **b. Attempt any ONE of the following.**

i) Describe the timer modes of 8051 microcontroller

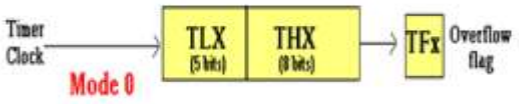
Ans) ($1^{1/2}$ mark each mode)

Operating modes of Timer:

The timer may operate in any of the four modes that are determined byM1 and M0 bit in TMOD register.

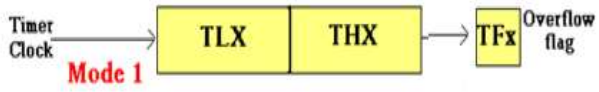
M1	M0	Mode	Operating Mode	
0	0	**0**	**13-bit timer** mode	8-bit THx + 5-bit TLx (x= 0 or 1)
0	1	**1**	**16-bit timer** mode	8-bit THx + 8-bit TLx
1	0	**2**	**8-bit auto reload**	8-bit auto reload timer/counter;
1	1	**3**	**Split timer mode**	

MODE 0: 13 Bit Timer:



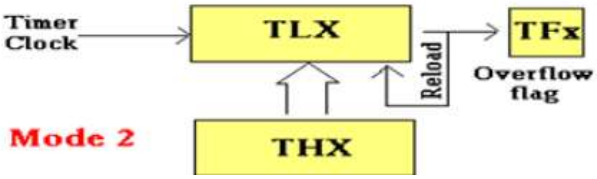
- Mode 0 is a 13-bit timer mode that provides compatibility with the 8051's predecessor, the 8048.
- It is not generally used in new designs.
- The timer high-byte (THx) is cascaded with the five least-significant bits of the timer low-byte (TLx) to form a 13-bit timer.

Mode -1: 16-Bit Timer Mode



- Mode 1 is 16-bit timer mode
- The clock is applied to the combined high and low timer registers (TLx/THx).
- An overflow occurs on the FFFFH-to-0000H transition of the count and sets the timer overflow flag.
- Overflow flag is the TFx bit in TCON

Mode -2: 8 bit Auto Reload Mode

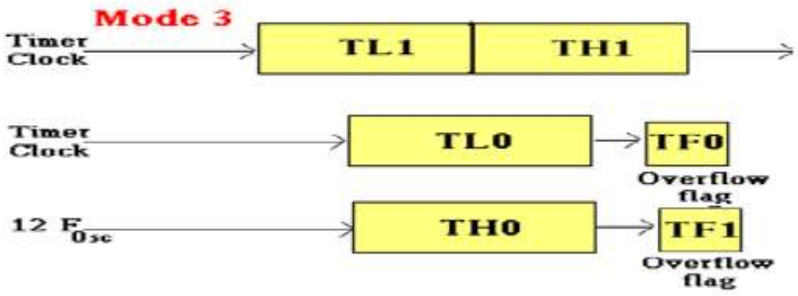


.8-bit auto-reload mode

• The timer low-byte (TLx) operates as an 8-bit timer while the timer high-byte (THx) holds a reload value.

• When the count overflows from FFH, not only is the timer flag set, but the value in THx is loaded into TLx.

Mode -3: Split Timer Mode



- Mode 3 is the split timer mode and is different for each timer.
- Timer 0 in mode 3 is split into two 8-bit timers.
- **TL0** and **TH0** act as separate timers with overflows setting the **TF0** and **TFI** bits respectively.

.Timer 1 is stopped in mode 3 but can be started by switching it into one of the other modes. The only limitation is that the usual Timer I overflow flag, TF1, is not affected by Timer 1 overflows, since it is connected to TH0.

• The 8051 appears to have a third timer.

• Timer 1 can be turned on and off by switching it out of and into its own mode 3. It can still be used by the serial port as a baud rate generator.

ii) List the addressing modes of 8051 microcontroller with examples each.

Ans) (Listing of modes 1/2M each, Example 1/2M each)

Addressing modes of 8051:

1) Register addressing mode : E.g. MOV A, R_7

2) Direct addressing mode:E.g. MOV A,54 OR MOV A,SFR

3) Register indirect addressing mode: E.g. MOV A, @R_P Where $R_P = R_0$ or R_1

4) Immediate addressing mode: E.g. MOV A, #45H

5) Register specific addressing mode : DA A OR RLA ,RRA

6) Indexed addressing mode : E.g MOVC A,@A+PC or MOVC A,@A+DPTR

Q 2. Attempt any FOUR of the following.

a) Classify the instruction set of microcontroller 8051 Give one example of each.

Ans) (Classification 1 mark each ,Example 1 mark each, Any four type)

Classification of instruction of 8051:

1. Data transfer instructions.

   MOV R0, direct.

2. Arithmetic instruction.

   ADD A,@R0

3. Logical Instruction

   ANL A,# data

4. Branch Instructions

   SJMP radd.

5. Bitwise instruction

   CPL C

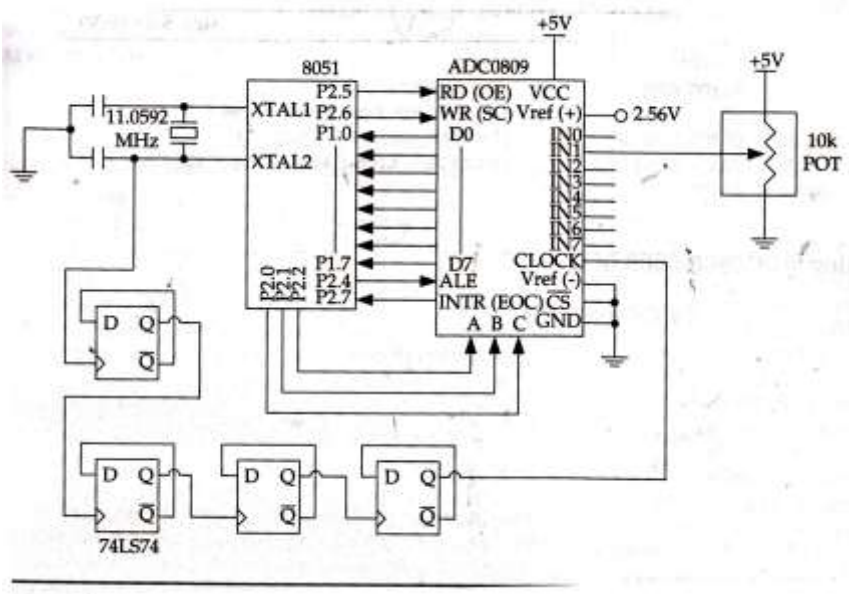
b) Draw the interfacing diagram of ADC 0809 with microcontroller 8051. Write C language program

to generate 50 Hz sq. wave with crystal freq.=12MHz.

Ans) (Diagram 4 M , Program : 4 M)

Note:(Any other program logic to generate square wave should be given marks)

Diagram:



Note: diagram with direct clock can be given full marks.

Program to generate square wave of 50 Hz :

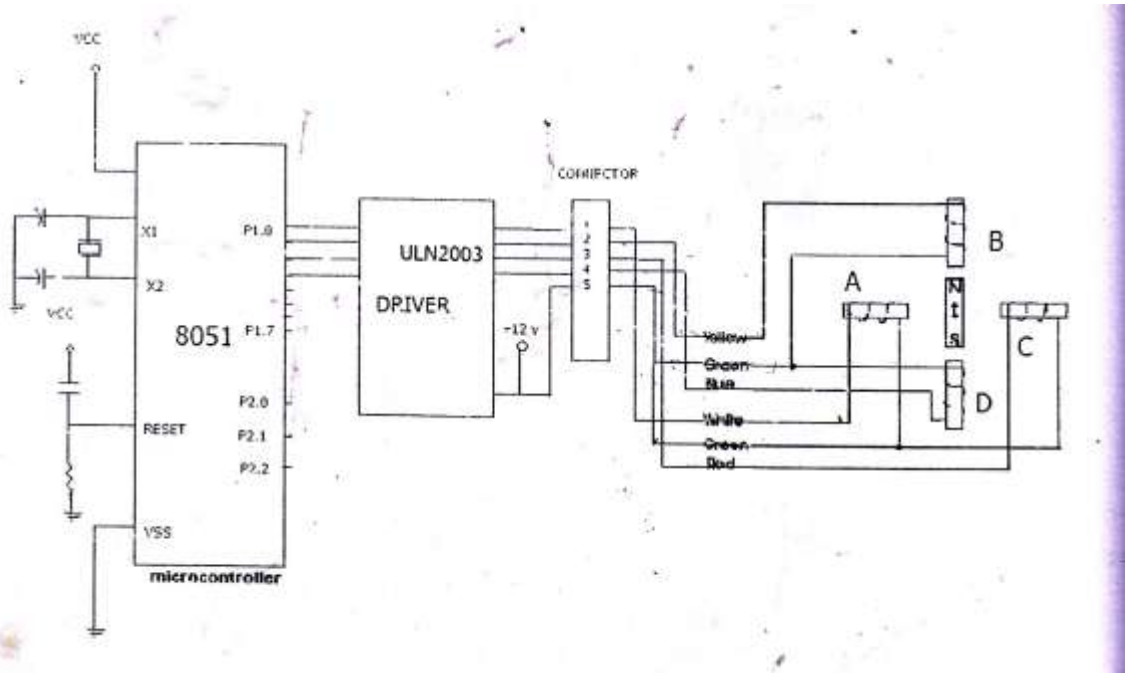
```
#include<reg51.h>
Void T0Delay(void);
Void main(void)
{
While(1)
{
P1=0x55;
T0Delay();
P1=0xAA;
T0Delay();
}
```

```
}

Void T0Delay()

{

TMOD=0x01;

TL0=0x0F0;

TH0=0x0D8;

TR0=1;

While(TF0==0);

TR0=0;

TF0=0;

}
```

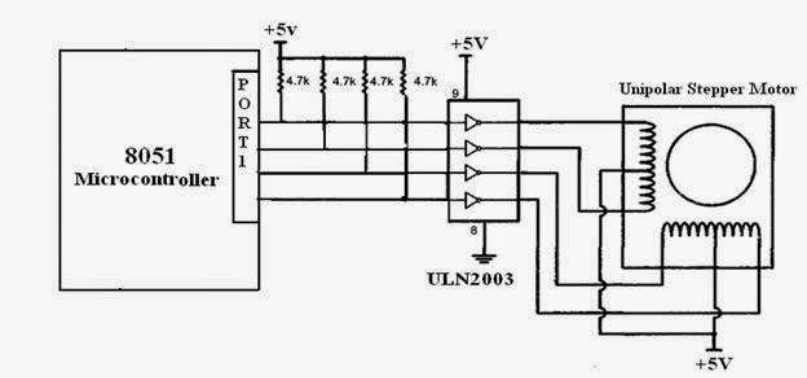
C. Draw the interfacing diagram of stepper motor with 8051 microcontroller. Draw the flow chart for rotating stepper motor in clockwise direction. (Program not expected)

Ans) (Diagram 4 M , Flowchart 4 M)

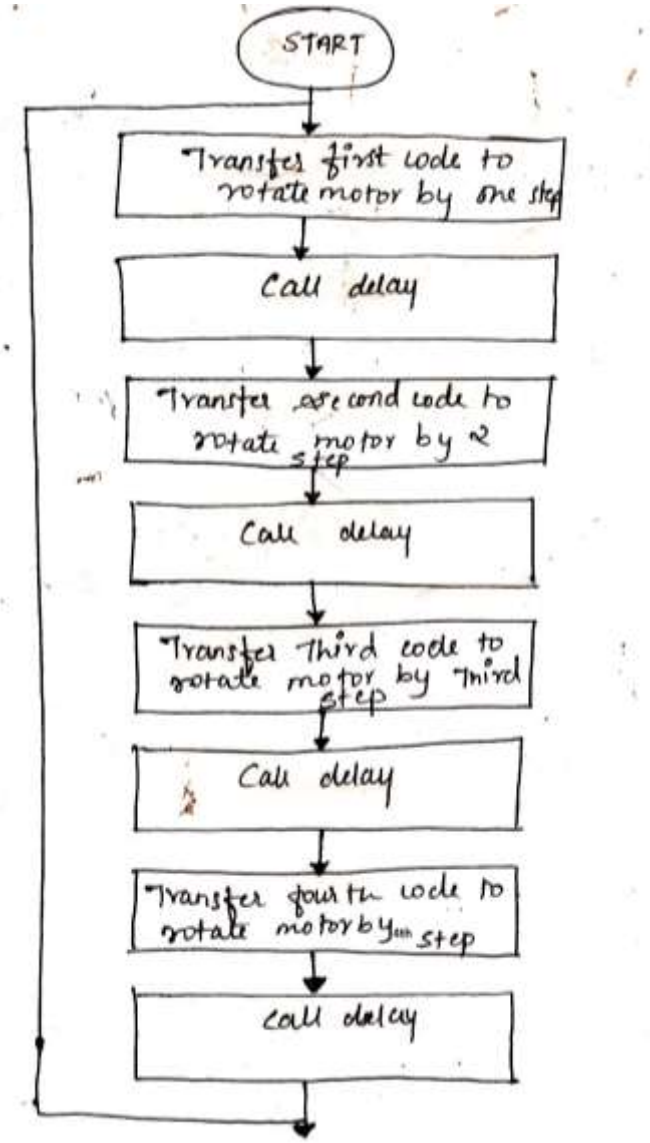
Diagram:



'OR'



Flow chart:



Q 3. Attempt any FOUR of the following.

a) Draw the format of PSW SFR and Describe the function of each bit

Ans)(Format – 2M, Explanation – 2M)

PSW register –

Format:-

D_7 D_0

Cy	AC	F0	RSI	RS0	0V	X	P

Bit7: Carry/Borrow bit: When two 8-bit operands are added, the result may exceed 8-bit and the 9th bit is copied in the carry bit. During subtraction, if the borrow occurs, the carry bit is set and otherwise, it is cleared.

Bit 6: Auxiliary carry/borrow bit: This bit indicates a carry from lower nibble during 8 bit addition. If AC flag is set, it means there is a carry from 3rd to 4th bit position.

Bit5: F0: Flag 0 is available to user for general purpose.

Bit 4-3: RS1:RS0: Register Bank selects bits

RS1	RS0	Bank selected
0	0	Register Bank 0
0	1	Register Bank 1
1	0	Register Bank 2
1	1	Register Bank 3

Bit 2: OV: Over flow flag is used to detect errors in signed arithmetic operations. When two signed numbers are added, it the result exceeds the destination, overflow flag is set, else it is reset.

Bit1: Undefined flag

Bit 0: Parity Flag: P flag is set, if the result contains an odd number of 1 bit, else it is reset, if the result contains an even number of 1 bit.

b) Compare between 8051 and 8052 microcontroller.

Ans: (any 4 points each 1 mark)

Feature	8051	8052
ROM(bytes)	4K	8K
RAM(bytes)	128	256
Timers	2	3
I/O pins	32	32
Serial Port	1	1
Interrupts	6	8
Watchdog timer	No	No

c) Write C program to continuously toggle all bits of port O with same delay.

Ans) (4 M for correct logic)

```
#include <reg51.h>

Void MSDelay(unsigned int) ;

Void main(void)
{
While(1)  //repeat forever

{

P0=0x55;

MSDelay(250) ;

P0=0xAA;

MSDelay(250);

}
}

Void MSDelay(unsigned int itime)

{
Unsigned int i, j;

For(i=0;i<itime;i++)
```

```
For(j=0;j<itime;j++) ;

}
```

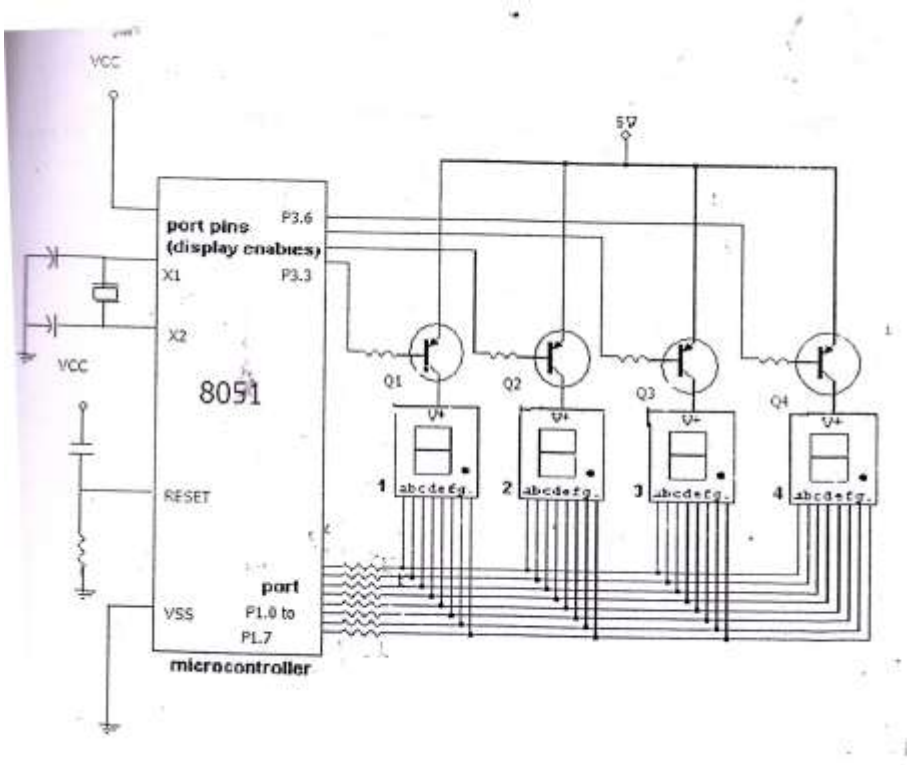
d) Compare between CISC RISC and machines.

Ans) (any 4 points each 1 mark)

RISC	CISC
Reduced instruction set computer	Complex instruction set computer.
Instruction set is simple and limited.	Instruction set is very large.
Instruction set is not flexible, hence the program is long.	Instruction set is flexible, hence the program is short
Provides few addressing modes normally register.	Provides many addressing modes
Large memory is required	Less memory required
Provides large number of registers	No. of registers are less.
Process architecture & control unit is simple	Processor architecture and control unit is complicated.
Instruction are shorter, hence execution speed is fast.	Instructions are lengthy hence execution speed is slow.
External memory is accessed rarely	External memory is accessed frequently
Each instruction required few bus cycle.	Each instruction requires many bus cycle.
e. g. ARM, ATMEL, AVR, MIPS, PIC, POWER PC, etc.	Interx86, Motorola, 68000 series.

e) Draw the interfacing diagram of four common cathode 7 segment display connected in multiplexed mode with 8051 microcontroller.

Ans) Diagram :



Q.4 A) Attempt ant THREE of the following.

i) Draw the interfacing diagram of temperature measurement using LM35, ADC0808 with microcontroller 8051.

Ans) (Correct diagram : 4marks)

Diagram:

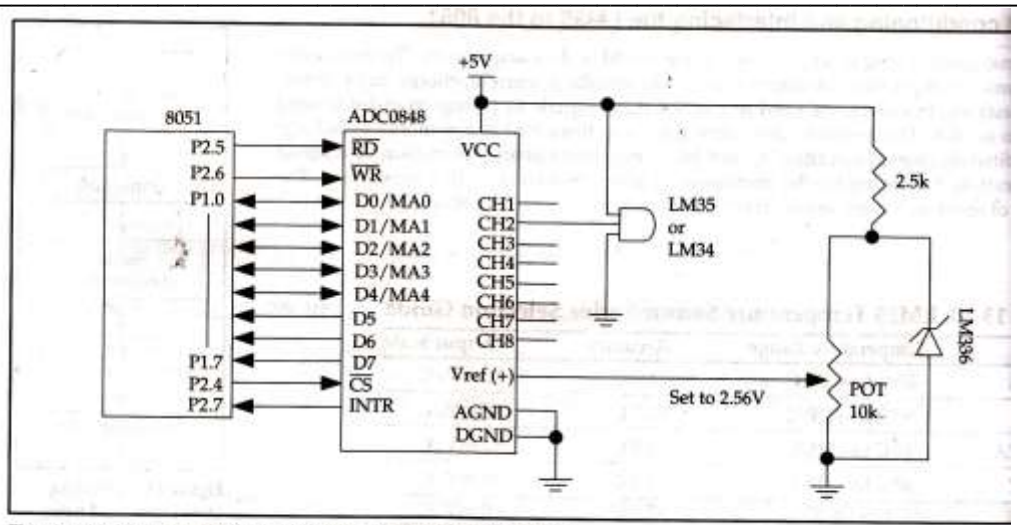


Figure 13-21. 8051 Connection to ADC0848 and Temperature Sensor

ii) Write instruction to perform following task using C operators:

Ans) (2 marks each)

P0= 0x56 >>4; shift 4 times to right

P0= 0x56 <<4 ; shift 4 times to left

iii) Compare between EPROM and flash memory.

Ans) (any 4 points each 1 mark)

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

iv) List the alternate function of port 3.

Ans) (½ MARK EACH)

P3.0-RXD (Receive serial data)
P3.1-TXD (Transmit serial data)
P3.2-$\overline{INT_O}$ (External Hardware Interrupt 0)
P3.3-$\overline{INT_1}$ (External Hardware Interrupt 1)
P3.4-T_O (External pulses given to counter0)
P3.5-T_1 (External pulses given to counter1)
P3.6-$\overline{WR}$ (Active low Write signal enables writing into external data memory)
P3.7-$\overline{RD}$ (Active low Read signal enables reading external data memory)

Q 4 B) Attempt any ONE of the following

i) Describe the functions of following instructions of 8051 Microcontroller.

1. **SWAP A**
2. **DIV AB**
3. **RLA**
4. **XCH A,RO**
5. **SETB C**
6. **DA A**

Ans) (1 mark each for every instruction)

1) SWAP A

SWAP nibbles within the accumulator .This instruction interchanges the low order and high order nibbles of the accumulator.

EG. A = 05H

SWAP A

A = 50H

2) DIV AB

This instruction divides the content of Register A by the contents of register B. Quotient is stored in register A & remainder is stored in register B .

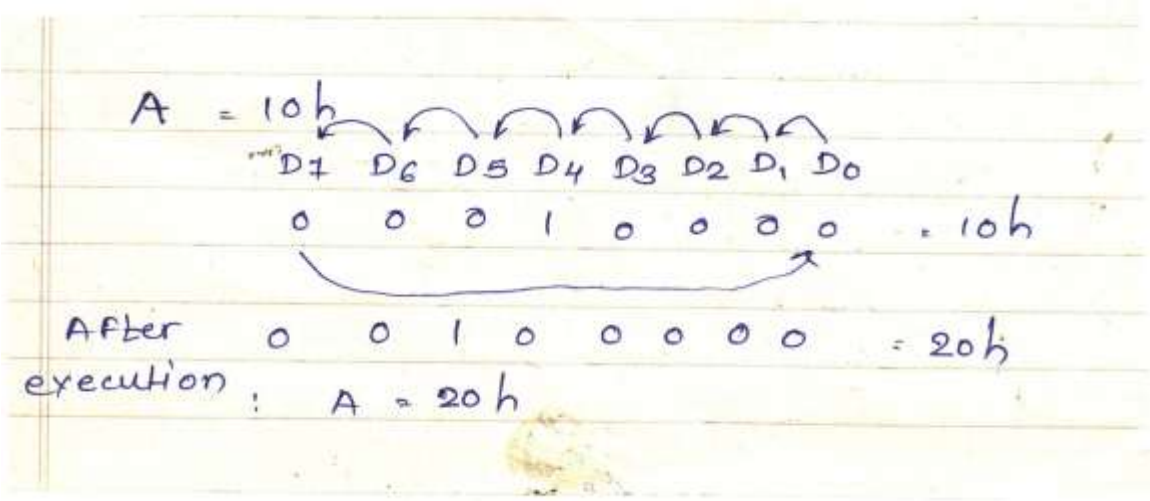
(Dividend is in Register A , Divisor is in Register B)

Flags affected : CARRY FLAG (C) = 0 & OVER FLOW FLAG (OV) =X

3) **RLA**

Rotate the contents of Accumulator to LEFT without including Carry flag.

For e.g A=10h



4) **XCH** A , R0.

(A) ⟷ (R0)

Exchange the content of A with register R0.

For e.g. A=23h, R0=78h

After execution: A=78h, R0=23h

5) **SETBC**

Set the carry flag. After the execution CY=1.

6) DA A

Decimal Adjust Accumulator

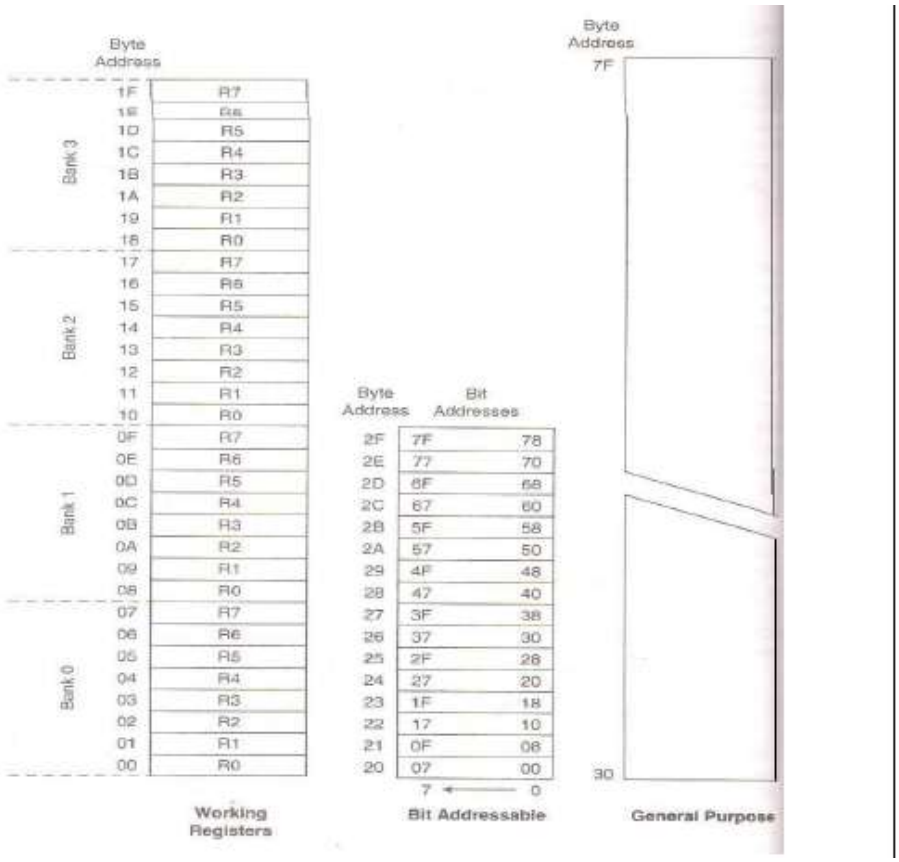
Only after the addition of two BCD numbers this instruction can be used. Result stored in Accumulator is converted into equivalent BCD form.

For eg. A=0Ch

After execution A=12 BCD

ii) With suitable diagram describe the memory organization of internal program and data memory.

Ans) (1 mark for ROM explanation, 3 marks for RAM explanation, 2 marks for Internal RAM diagram)



Internal RAM of 8051 is organized in to three different areas.

1) Working registers
2) bit / byte addressable
3) general purpose

1) Working registers
First 32 bytes from address 00H to 1FH are organized as four register banks.
The four register banks are named as Bank 0, Bank 1, Bank 2 & Bank 3.
Each register bank consist of eight register denoted as R_0 to R_7. each register is of 8-bits.
The register bank is selected by setting or resetting bits. RS_O & RS_1 of PSW register.

2) bit / byte addressable

The memory location area from 20H – 2FH, total 1B memory locations are used for bit-bytes addresses. One memory location as 8-bits.Each bit has its own address. Therefore one memory location has 8- addresses. Hence 16-memory location that is 16×8=128 addresses. Therefore bit addresses of this area varies from OOH-7FH.Thus single 128 bits can be stored in this area.

4) general purpose

the remaining memory location of RAM from 30H-7F can be used as general purpose data memory where data results are stored. These area can be used as stack memory.

8051 also consists of 4KB of internal ROM from the address 000h to FFFh where codes can be stored.

Q.5 Attempt any TWO of the following:

a)Write 'C' program to transfer message "INDIA" serially at baud rate 4800bps, 8-bit data, 1 stop bit, assume crystal frequency 11.0592 MHz.

Ans) (Baud rate calculation : 2 marks , Correct program with comments : 6 marks)

Baud rate calculation: (for 4800)

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

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

Upon reset SMOD =0

Baud Rate = (1 / 32) X (11.0592 MHZ / 12[256-TH1])

Baud Rate = 28800 HZ / [256 – TH1]

4800 = 28,800 / [256 – TH1]

TH1= FA H = -6

'C' PROGRAM :

```
#include<reg51.h>
void main(void)
{
        unsigned char msg[]= "INDIA";
        int i;
        TMOD=0X20;// timer 1,mode 2
        TH1= - 6 ;// count for 4800 BAUD
        SCON=0x50; // 8 bit ,1 stop bit (serial mode 1)
        TR1=1;//start timer 1
        for (i=0;i<5;i++)
        {
        SBUF = msg[i];//send message "INDIA" serially
        while(TI==0);//wait for transmission to complete
        TI=0;//clear TI flag for next transmission
        }
}
```

NOTE: (ANY OTHER CORRECT PROGRAM LOGIC SHOULD ALSO BE CONSIDERED AND GIVEN MARKS)

b) Write assembly language program to transfer array of ten numbers stored in memory location 50H to memory location 70H.

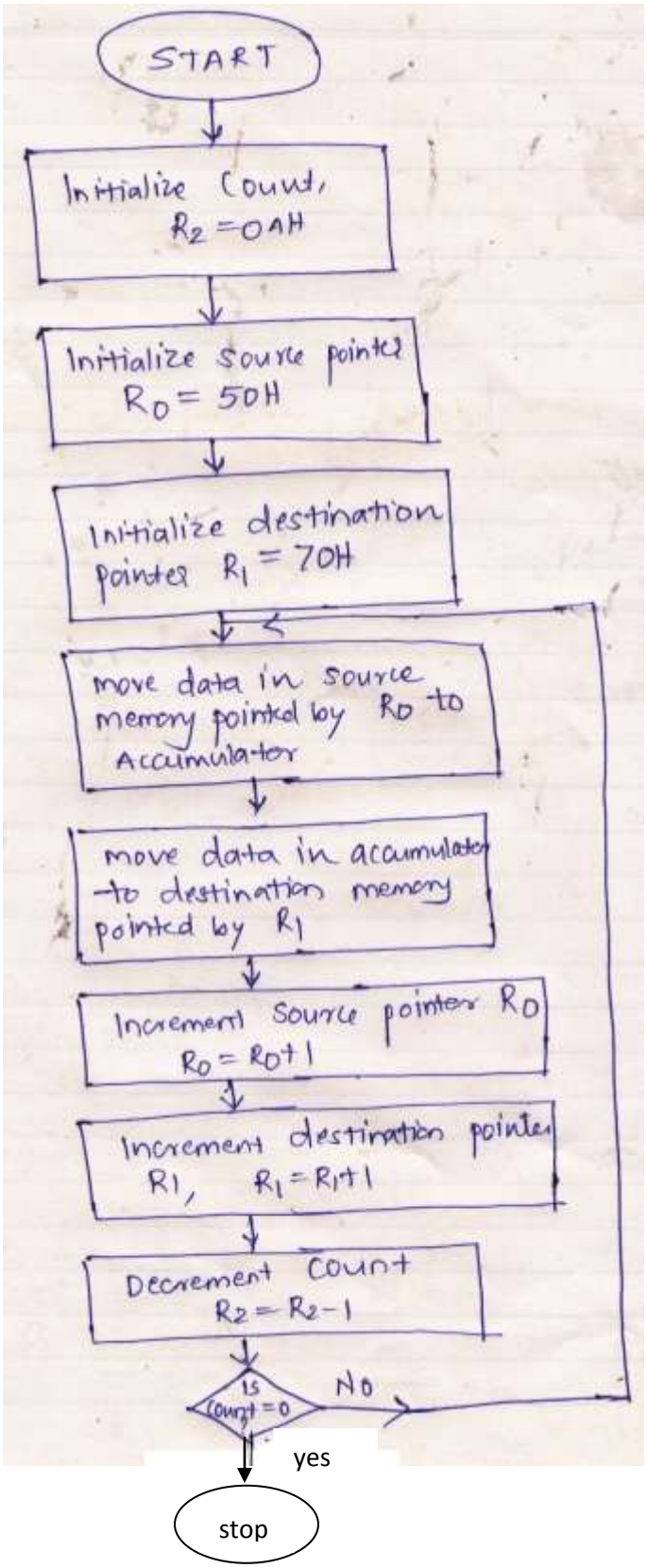
Ans) (ALGORITHM/FLOWCHART : 4 MARKS

PROGRAM : 4 MARKS

Algorithm:

1. Start
2. Initialize count R2=0AH
3. Initialize source pointer R0 with starting address 50h
4. Initialize destination pointer R1 with starting address 70h
5. Move number pointed by R0 to accumulator
6. Move number from accumulator to destination address pointed by R1
7. Increment source pointer R0
8. Increment destination pointer R1
9. Decrement count & if count not equal to zero repeat step 5 to step 9
10. Else stop

FLOW CHART:


START
Initialize Count, R2 = 0AH
Initialize source pointer R0 = 50H
Initialize destination pointer R1 = 70H
move data in source memory pointed by R0 to Accumulator
move data in accumulator to destination memory pointed by R1
Increment Source pointer R0 R0 = R0+1
Increment destination pointer R1, R1 = R1+1
Decrement Count R2 = R2-1
Is Count = 0
NO
yes
stop

Assembly language program:

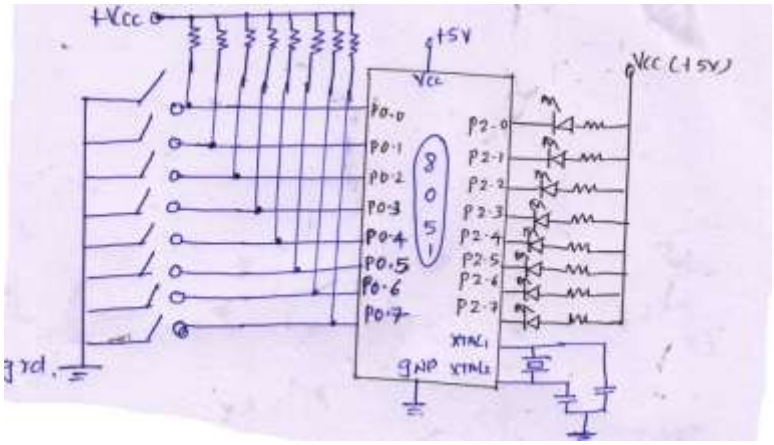
```
        MOV R2,#0AH ;  load count 10 in register R2
        MOV R0,#50H;   load source address in register R0
        MOV R1,#70H ;  load destination address in register R1
UP:     MOV A , @R0 ;  take source data pointed by R0 in Accumulator
        MOV @R1 , A ; Move data in accumulator to address pointed by R1
        INC R0          ; Increment source pointer R0
        INC R1          ; Increment destination  pointer R1
        DJNZ  R2 , UP  ;Decrement count and jump UP if count = 0
HERE:  SJMP HERE       ; terminate program
```

Note: (ANY OTHER CORRECT PROGRAM LOGIC SHOULD ALSO BE CONSIDERED AND GIVEN MARKS)

c) Draw interfacing diagram to connect 8 LED's on port 2 and 8 switches on port 0. Write C program to read the status of switches and send to port 2.

Ans) (Interfacing diagram : 4 marks ,'C' Program : 4 marks)

Interfacing diagram



'C' PROGRAM

```
#include<reg51.h>
void main(void)
{
        unsigned char MYBYTE;
        P0=0XFF;      //SET PORT0 AS INPUT
        While(1)
        {
        MYBYTE=P0;     //GET SWITCH STATUS FROM PORT 0
        P2=MYBYTE;    //SEND DATA BYTE TO PORT 2
        }
}
```

Q.6. Attempt any FOUR of the following:

a) Draw and describe IP SFR format for 8051 microcontroller.

Ans) (IP FORMAT : 2 MARKS, DESCRIPTION :2 MARKS)

- **The IP (INTERRUPT PRIORITY) SFR FORMAT**

MSB **LSB**

-	-	-	PS	PT1	PX1	PT0	PX0

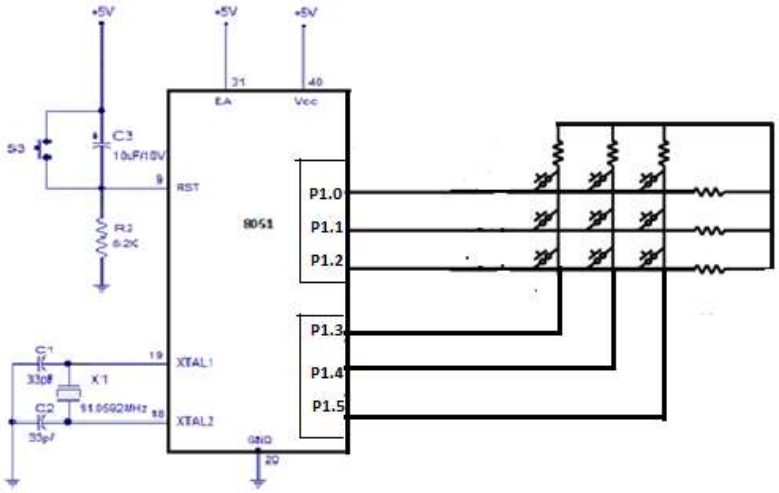
DESCRIPTION:

Bit	Name	Description
IP.7	-	Reserved
IP.6	-	Reserved
IP.5	-	Reserved
IP.4	PS	Serial Port Interrupt priority
IP.3	PT1	Timer 1 Interrupt priority (TF1)
IP.2	PX1	External Interrupt 1 priority (INT1)
IP.1	PT0	Timer 0 Interrupt priority (TF0)
IP.0	PX0	External Interrupt 0 priority (INT0)

Priority bit =1 , assigns higher priority , Priority bit 0 = assigns lower priority

b) Draw interfacing diagram to interface 3 X 3 key matrix to 8051 microcontroller.

Ans) (CORRECT DIAGRAM : 4 MARKS (ANY PORT CAN BE USED))



c) Describe the following assembler directives with one examples.

i. **ORG**
ii. **DB**
iii. **EQU**
iv. **END**

Ans) (Each directive explanation with example : 1 mark each)

1. **ORG :** ORIGIN
   The ORG directive is used to indicate the beginning of the address. The number that comes after ORG is the address from where program will begin. The number can be either in hex and decimal
   e.g. ORG 1000H
   It indicates that program shall start from memory address 1000h

2. **DB : Define Byte**
   It is used to define the 8-bit data. It is used to write the value after DB , into the program memory. When DB is used to define data, the numbers can be in decimal, binary, hex, ASCII formats
   e.g. ORG 1000H
   MYDATA: DB 20,21
   After execution of this , location 1000h=20 & 1001h = 2**1**

3. **EQU:** EQUATE
   It is used to define a constant without occupying a memory location The EQU directive assigns a constant value to a label. When the label appears in the program, its constant value will be substituted for the label.
   e.g. COUNT EQU 10
   MOV R2, #COUNT

When the instruction is executed , register R2 is loaded with value 10.

4. **END:** This indicates to the assembler the end of the source (asm) file . The END directive is the last line of an 8051 program. Means that in the program anything after the END directive is ignored by the assembler

e.g.
```
MOV A , #20
ADD A , #10
END
```

d) Describe the dual role of port 0 of microcontroller 8051.

Ans) (DIAGRAM: 2 MARKS ,EXPLAINATION :2MARKS)

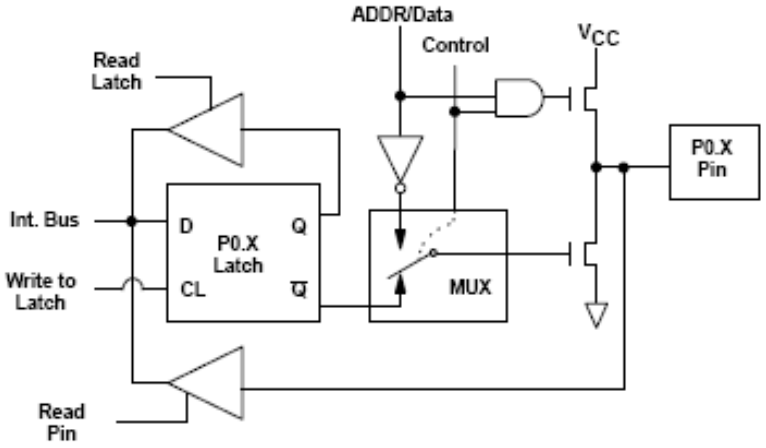
Port 0 can be used as

a) Simple I/O Port and

b) Bidirectional low order address and data bus (AD0 – AD7) for external memory access.

Port -0 has 8 pins (P0.0-P0.7).

The structure of a Port-0 pin is shown in fig below



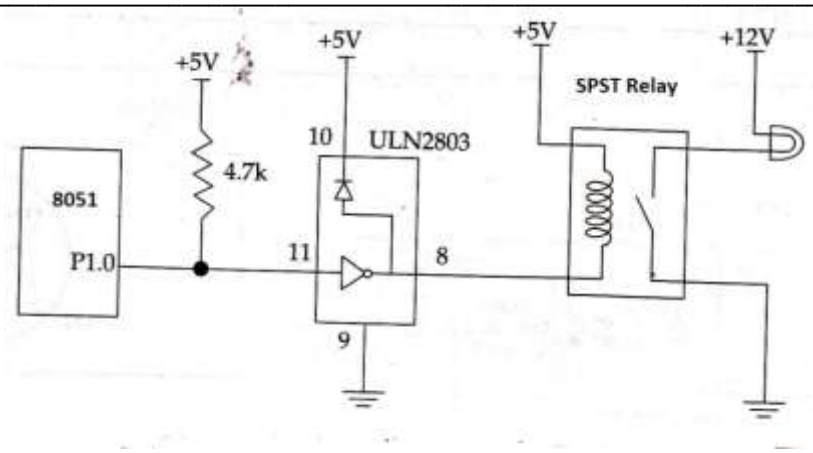
Port0 pin circuit

When the control is '0', the port can be used as a normal bidirectional I/O port. When the port is used as an input port, '1' must be written to the latch and to output 0 on the pin '0' must be written to the latch.

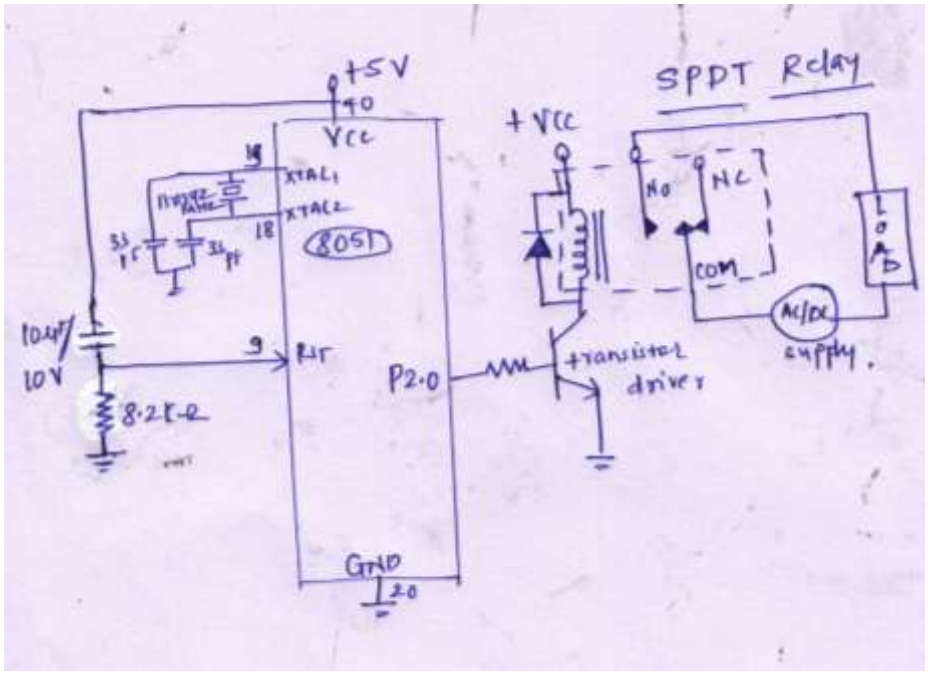
When the control is '1', the port can be used as address/data bus(AD0 – AD7)for external data transfer.

e) Draw interfacing diagram to interface relay with 8051 microcontroller.

Ans) (CORRECT DIAGRAM : 4 MARKS)


+5V
+5V
+5V
+12V
SPST Relay
10
ULN2803
4.7k
8051
P1.0
11
8
9

'OR'


+5V
40
Vcc
SPDT Relay
+Vcc
XTAL1
XTAL2
18
8051
NO
NC
COM
LOAD
AC/DC supply.
10μF/10V
9
RST
P2.0
transistor driver
8.2kΩ
GND
20