WINTER– 14 EXAMINATION

Subject Code: **17534** **Model Answer**

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.

1 A) Attempt any three: **12 Marks**

i) What is a bus? Describe the different types of buses used in 8051 microcontroller.

Ans: - 1 Mark- definition of Bus, 3 Mark -1x3(three Bus types-explanation)

A Bus is a set of physical connection used for communication between CPU and peripherals.

There are three types of buses Address Bus, Data Bus and Control Bus

(1) Address Bus

- The address bus is unidirectional over which the microcontroller sends an address code to the memory or input/output. The size of the address bus is specified by the number of bits it can handle.
- The more bits there are in the address bus, the more memory locations a microcontroller can access. A 16-bit address bus is capable of addressing (64k) addresses.

(2) Data Bus

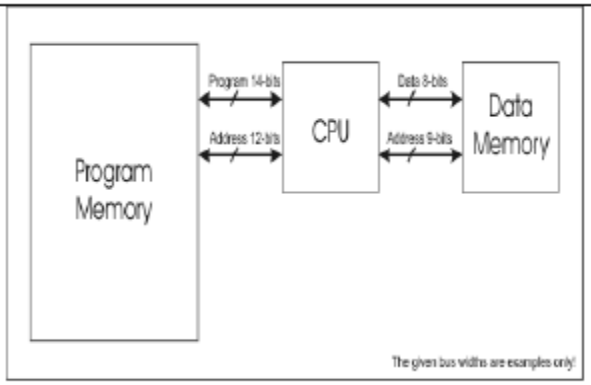
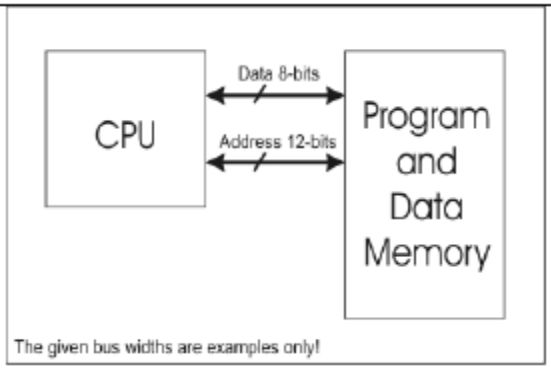
- The data bus is bi—directional on which data or instruction codes are transferred into the microcontroller or on which the result of on operation or computation is sent out from the microcontroller to the memory or input/output.
- Depending on the particular microcontroller, the data bus can handle 8-bit or 16-bit data.

(3) Control Bus:

- The control bus is used by the microcontroller to send out or receive timing and control signals like read and write in order to co- ordinate and regulate its operation and to communicate with other devices i.e. memory or input/output.

ii) State the difference between Harvard and Von Neumann architectures with suitable diagram

Ans: - Harvard Architecture diagram -- 1 Mark ,
Van Neumann's Architecture diagram -- 1 Mark
Any 2 difference points --2 marks

Sr. No	Harvard Architecture	Van Neumann's Architecture
1.		
2.	The Harvard architecture uses physically separate memories for their instructions and data.	The Van Neumann's architecture uses single memory for their instructions and data.
3.	Requires separate & dedicated buses for memories for instructions and data	Requires single bus for instructions and data.
4.	Its design is complicated	Its design is simpler.
5.	Instructions and data can be fetched simultaneously as there is separate buses for instructions and data which increasing operation bandwidth.	Instructions and data have to be fetched in sequential order limiting the operation bandwidth.

iii) Draw the format of PSW register of 8051 microcontroller and explain the function of any two flags.

Ans:- Flag register format --2 Marks ,Explanation --2 Marks(any two flags)

CY	AC	F0	RS1	RS0	OV	--	P

CY	PSW.7	Carry Flag.
AC	PSW.6	Auxiliary carry flag.
F0	PSW.5	Available to the user for general purpose.
RS1	PSW.4	Register bank selector bit 1.
RS0	PSW.3	Register bank selector bit 0.
OV	PSW.2	Overflow flag.
--	PSW.1	User- definable bit.
P	PSW.0	Parity flag. Set/cleared by hardware each instruction cycle to indicate and Odd/ even number of 1 bit in the accumulator.

1. CY: the carry flag.

This flag is set whenever there is a carry out from the D7 bit.
The flag bit is affected after an 8 bit addition or subtraction.
It can also be set to 1 or 0 directly by an instruction such as "SETB C" and CLR C" where "SETB C" stands for "set bit carry" and "CLR C" for "clear carry".

2. AC: the auxiliary carry flag

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

3. F0: Available to the user for general purposes.

4. RS0, RS1: register banks selects bits

These two bits are used to select one of the four register banks n internal RAM in the table. By writing zeroes and ones to these bits, a group of registers R0- R7 can be used out of four registers banks in internal RAM

RS1	RS0	Space in RAM
0	0	Bank 0 (00H- 07H)
0	1	Bank 1 (08H-0FH)
1	0	Bank2 (10H-17H)
1	1	Bank3 (18H-1FH)

.

5.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. In general, the carry flag is used to detect errors in unsigned arithmetic operations. The overflow flag is only used to detect errors in signed arithmetic operations.

6. P: Parity flag

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

iv) With the help of ADD instruction explain:

a) Direct addressing mode

b) Indirect addressing mode

c) Register addressing mode

d) Immediate addressing mode.

Ans 1 mark for each addressing mode.

1) Direct Addressing mode

ADD A,add

A ←A+(add)

ADD A,12H

The contents of memory location specified by 8 bit direct address will be logically ADDED bit by bit with the contents of accumulator result is stored in accumulator.

2)Indirect addressing mode

ADD A,@Ri

A ←A+(Ri)

ADD A,@R0

The content of memory location whose address is specified by Ri(R0/R1) will be logically added bit by bit with contents of accumulator. Result is stored in accumulator.

3) Register addressing mode

ADD A ,Rn

A ←A+Rn

ADD A,R2

The contents of specified register Rn (R0-R7) will be logically ADDED bit by bit with the contents of accumulator result stored in Accumulator.

4) Immediate addressing mode

ADD A,#data

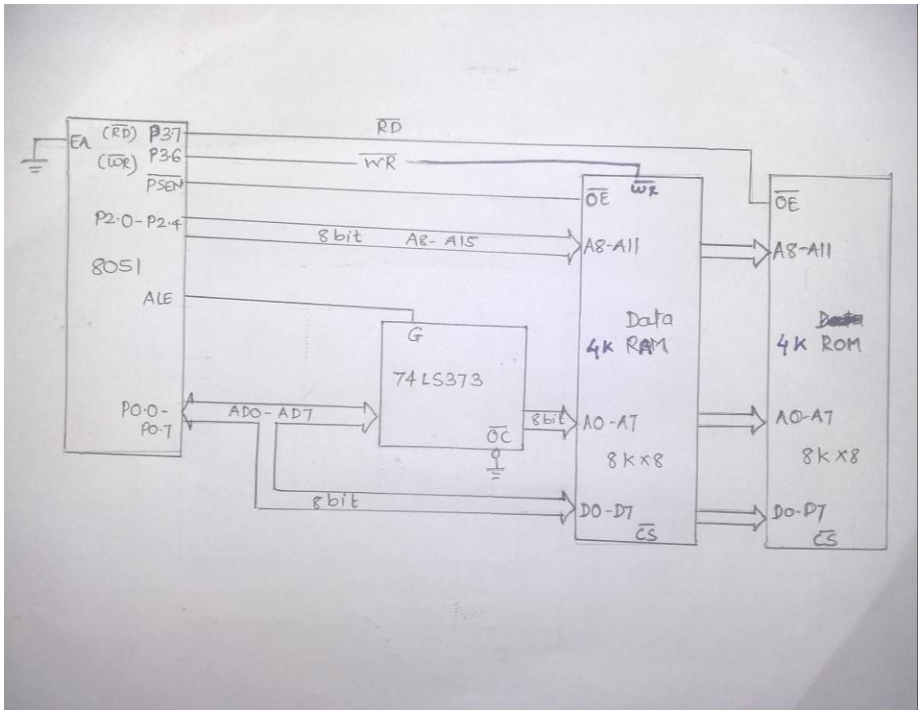
A ←A+#data

ADD A,#23H

Immediate data is logically ADDED bit by bit with contents of accumulator and result is stored in accumulator.

v) Draw the interfacing diagram of 4K bytes of RAM and 4K bytes of EPROM to 8051 microcontroller.

Ans:- 4 mks diagram



B) Attempt any one: 06 Marks

i) Write an assembly language program for 8051 microcontroller to add five 8 bit nos. stored in internal RAM from 20H onwards. Store the result at 30H

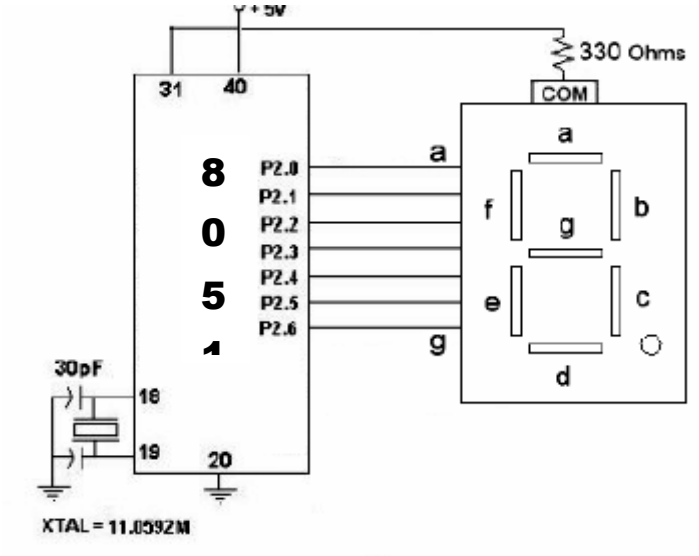
Ans:- 6 Marks- Any correct program with comments.

Program for addition of five 8 bit nos.

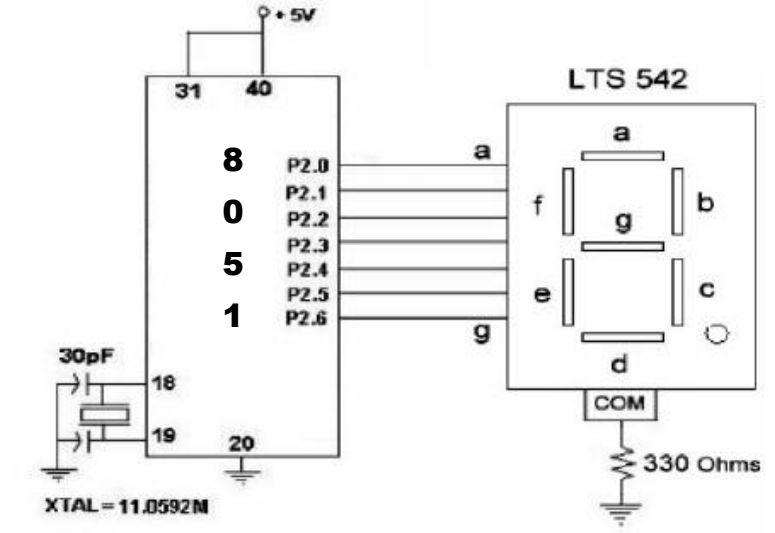
```
CLR PSW.3; Select register Bank 0
CLR PSW.4;
MOV Ro, #05H; Initialize byte counter
MOV R1, #20H ; Initialize memory pointer
MOV A, # 00H; Clear Accumulator
UP: ADD A @R1; Add accumulator with number from array
INC R1; Increment memory pointer
DJNZ R0, UP; Decrement byte counter,
        ; if byte counter ≠ 0
        ; Then go to UP
MOV30H, A; Store result in internal memory
HERE: SJMP HERE; Stop
```

ii) Draw the interfacing diagram of seven segment display to 8051 microcontroller .Write an Assembly Language Program to display 'g' on seven segment display.

Ans:- 4 Marks –Diagram, 2 Marks—Any correct program



For Common anode display



For Common cathode display

(**Any other relevant diagrams should also be considered correct.)**

Program(for common cathode display)

MOV A,#6F H ;load 7 segment code of g into A

MOV P2,A ;0ut code of g to port 2

HERE : SJMP HERE

Q2) Attempt any four: **16 Marks**

a) Draw the internal RAM memory organization of 8051 microcontroller.:

Ans:- **Diagram: 4Marks**

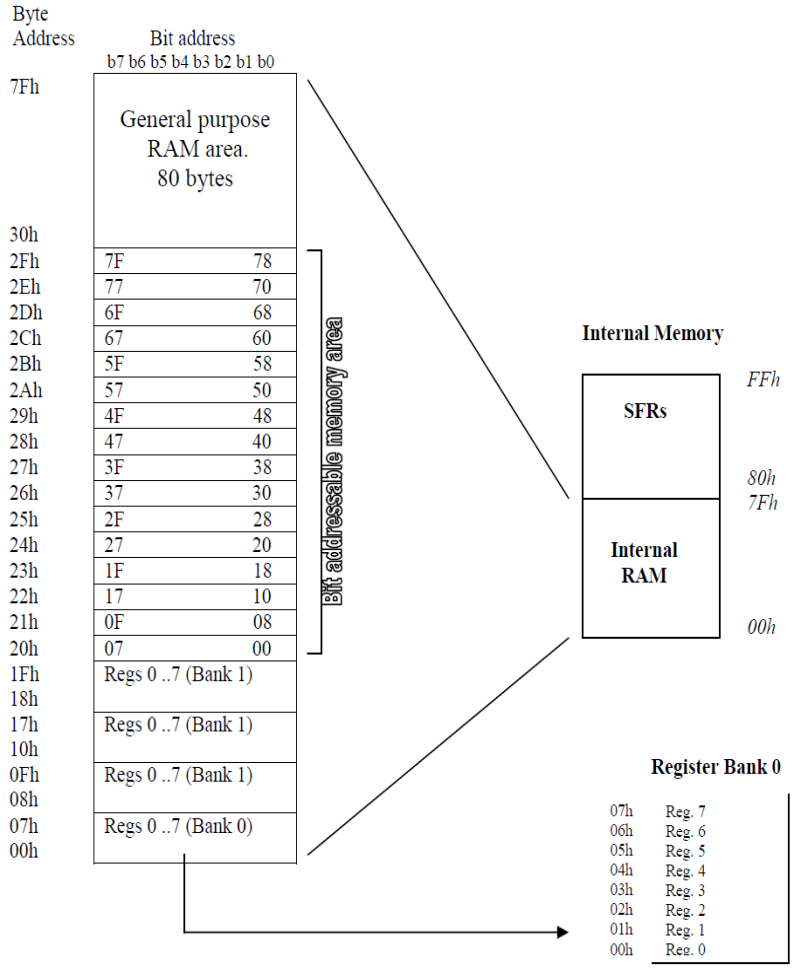
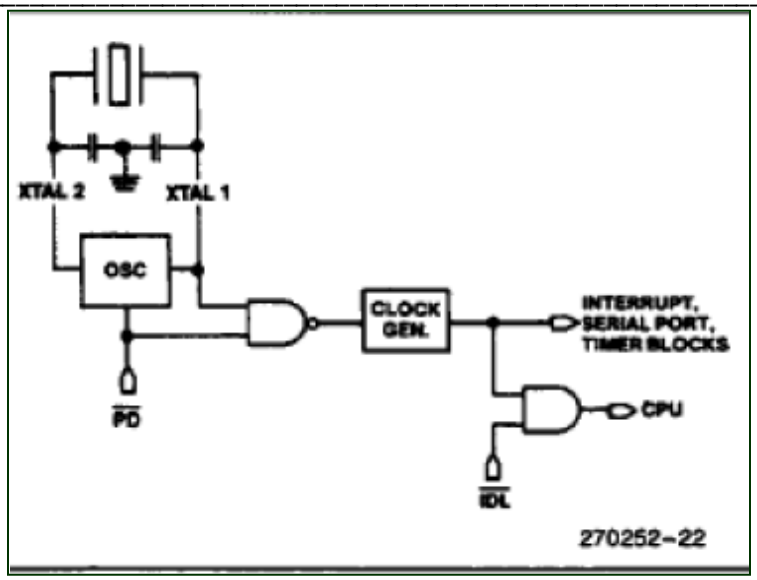


Fig:- Internal memory organization

b) Describe the power saving modes of 8051 microcontroller.

Ans : Diagram : 1 mark, PCON format: 1mark, Explanation : 2 marks



Format of PCON:

PCON: POWER CONTROL REGISTER. NOT BIT ADDRESSABLE.

SMOD	—	—	—	GF1	GF0	PD	IDL

SMOD Double baud rate bit. If Timer 1 is used to generate baud rate and SMOD = 1, the baud rate is double when the Serial Port is used in modes 1, 2, or 3.
— Not implemented, reserved for future use.*
— Not implemented, reserved for future use.*
— Not implemented, reserved for future use.*
GF1 General purpose flag bit.
GF0 General purpose flag bit.
PD Power Down bit. Setting this bit activates Power Down operation in the 80C51BH.
IDL Idle Mode bit. Setting this bit activates Idle Mode operation in the 80C51BH.

IDLE MODE

In the Idle mode, the internal clock signal is gated off to the CPU, but not to the Interrupt, Timer, and Serial Port functions.

The CPU status is preserved in its entirety, the Stack Pointer, Program Counter, Program Status Word, Accumulator, and all other registers maintain their data during Idle. The port pins hold the logical state they had at the time idle mode was activated. ALE and PSEN hold at logic high levels.

There are two ways to terminate the idle mode.

i) Activation of any enabled interrupt will cause PCON.O to be cleared and idle mode is terminated.

ii) Hard ware reset: that is signal at RST pin clears IDEAL bit IN PCON register directly. At this time, CPU resumes the program execution from where it left off.

POWER DOWN MODE

An instruction that sets PCON.1 causes that to be the last instruction executed before going into the Power Down mode. In the Power Down mode, the on-chip oscillator is stopped. With the clock frozen, all

functions are stopped, but the on-chip RAM and Special Function Register are maintained held. The port pins output the values held by their respective SFRS. ALE and PSEN are held low.
Termination from power down mode: an exit from this mode is hardware reset.
Reset defines all SFRs but doesn't change on chip RAM

c) State any four important features of 8051 microcontroller.

Ans. **Any four features -- 4 marks**
Features of 8051 micro controller are as follows:-

1) 8- bit data bus and 8- bit ALU.

2) 16- bit address bus – 64KB of RAM and ROM.

3) On- chip RAM -128 (256) bytes (" Data Memory")

4) On- chip ROM – 4 KB ("Program Memory")

5) Four 8-bit bi- directional input/output ports Four 8-bit bi- directional input/ output ports.

6) Programmable serial ports i.e. One UART (serial port)

7) Two 16- bit timers- Timer 0& Timer 1

8) Six interrupts are available: Reset, Two interrupts Timers i.e. Timer 0 and Timer 1, Two external hardware interrupts- INT0 and INT1, Serial communication interrupt for both receive and transmit

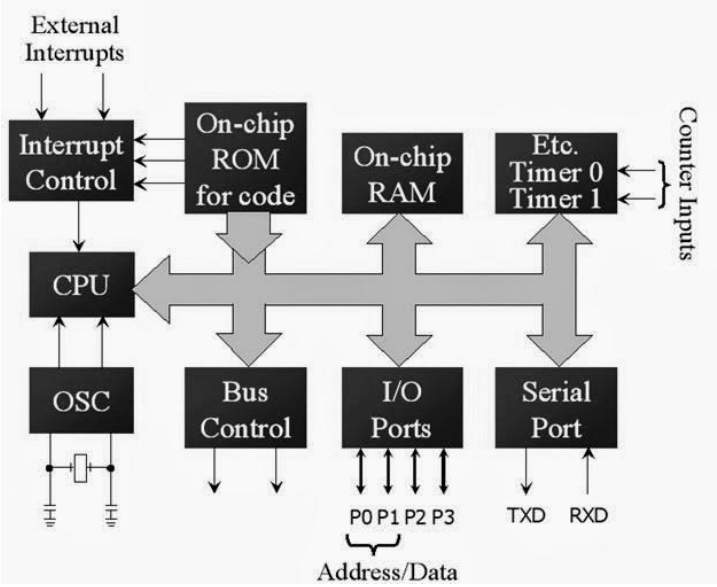
d) Compare 8031,8051,8751(any four points)

Ans:- 1x4 Marks for each difference

specification	**8031**	**8051**	**8751**
On chip data memory	**128 byte**	**128 byte**	**256 byte**
On chip program memory	**ROM less**	**4K ROM**	**4K EPROM**
Number of 16 bit timer/counter	**2**	**2**	**3**
Number of vectored interrupts	**5**	**5**	**5**
Full duplex serial I/O	**1**	**1**	**1**
On chip peripherals	**UART**	**UART**	**UART**
No of I/o lines	**32**	**32**	**32**
Speed MHz	**12**	**12**	**12**

e) Draw the block diagram of 8051 microcontroller.

Ans:- 4 Marks for correct diagram



f) Distinguish between microprocessor and microcontroller (any four points)

Ans. **(1x4 Marks for each point)**

Sr. No	Parameter	Microprocessor	Microcontroller
1.	No. of instructions used	Many instructions to read/ write data to/ from external memory.	Few instruction to read/ write data to/ from external memory
2.	Memory	Do not have inbuilt RAM or ROM.	Inbuilt RAM or ROM
		Program and data are stored in same memory.	Separate memory to store program and data
3.	Registers	Microprocessor contains general purpose registers, Stack pointer register, Program counter register	Microcontroller contains general purpose registers, Stack pointer register, Program counter register additional to that it contains Special Function Registers (SFRs) for Timer , Interrupt and serial communication etc.
4.	Timer	Do not have inbuilt Timer.	Inbuilt Timer
5.	I/O ports	I/O ports are not available requires extra device like 8155 or 8255.	I/O ports are available
6.	Serial port	Do not have inbuilt serial port, requires extra devices like 8250 or 8251.	Inbuilt serial port
7.	Multifunction pins	Less Multifunction pins on IC.	Many multifunction pins on the IC
8.	Boolean Operation	Boolean operation is not possible directly.	Boolean Operation i.e. operation on individual bit is possible directly
9.	Applications	General purpose, Computers and Personal Uses.	Single purpose(dedicated application), Automobile companies, embedded systems, remote control devices.

Q.3 Attempt **any four:** **16 Marks**

a) Describe the function of editor, assembler, compiler and linker.

Ans:- **1mark for each**

Assembly language programming tools:
1) Editor
2) Assembler
3) Compiler
4) Linker

1) Editor: An editor is a program which helps you to construct your assembly language program in right format so that the assembler will translate it correctly to machine language. So, you can type your program using editor. This form of your program is called as source program and extension of program must be .asm or .src depending on which assembler is used.

The DOS based editor such as EDIT , Wordstar, and Norton Editor etc. can be used to type your program.

2) Assembler: An assembler is programs that translate assembly language program to the correct binary code for each instruction i.e. machine code and generate the file called as Object file with extension .obj and list file with extension .lst extension.

Some examples of assembler are ASEM-51, Keil"s A51, AX 51 and C51, Intel PL/M-51 etc.

3) Compiler: Instructions in assembly language are represented in the form of meaningful abbreviations, and the process of their compiling into executable code is left over to a special program on a PC called compiler.

4) Linker: A linker is a program, which combines, if requested , more than one separately assembled object files into one executable program, such as two or more programs and also generate .abs file and initializes it with special instructions to facilitate its subsequent loading the execution.

Some examples of linker are ASEM-51 BL51, Keil u Vision Debugger, LX 51 Enhanced Linker etc.

b) Write an assembly language program to transfer five bytes from source block to destination block. Assume source block address is 10H and destination block address is 20H.

Ans:- **4 mks- proper program**

```
ORG 0000H                ;Program from 0000H
MOV R3, #05H             ;Initialize Byte counter
MOV R0, #10H             ;Initialize memory pointer for source array
MOV R1, #20H             ;Initialize memory pointer for destination array
                         ; therefore R0 ---> Source pointer R1 ---> destination pointer
UP: MOV A, @R0           ; Read number from source array
MOV @R1, A                     ; Write number to destination array
INC R0                   ;Increment source memory pointer by 1
INC R1                   ; Increment destination memory pointer by 1
DJNZ R3, UP              ;Decrement byte counter by 1
                         ; Is it zero? No, jump to UP
HERE: SJMP HERE
END                      ; Stop
```

c) Describe the function of following instructions of 8051 microcontroller.

1. MOVX A, DPTR
2. SWAP A
3. MUL AB
4. MOV A,R0

Ans:- **1mark for each instruction**

1. MOVX A, DPTR

Description: This instruction moves the contents of the external RAM memory pointed by (or stored in) DPTR to accumulator.
No of bytes: 1 byte
Addressing mode: register

```
Example:    MOV DPTR, # 2000H      ; DPTR = 2000H(external RAM
                                     address)
            MOV A, @DPTR           ; 2000H = 0BH
                                   ;A = 0BH
```

2. SWAP A

Description: This instruction exchanges bits 0-3 of the Accumulator with bits 4-7 of the Accumulator. This instruction is identical to executing "RR A" or "RL A" four times.
No of bytes: 1 byte
Addressing mode: register specific

```
Example:    MOV  A, #59H       ;A= 59H
             SWAP A            ; A= 95H
```

3. MUL AB

Description: Multiples the unsigned value of the Accumulator by **t**he unsigned value of the "B" register. The least significant byte of the result is placed in the Accumulator and the most-significant-byte is placed in the "B" register.The **Carry Flag (C)** is always cleared. The **Overflow Flag (OV)** is set if the result is greater than 255 (if the most-significant byte is not zero), otherwise it is cleared.

No of bytes: 1 byte
Addressing mode: register or register specific

```
Example:    MOV A, #5
            MOV B, # 7
            MUL AB          ; A= 35=23H, B=00
```

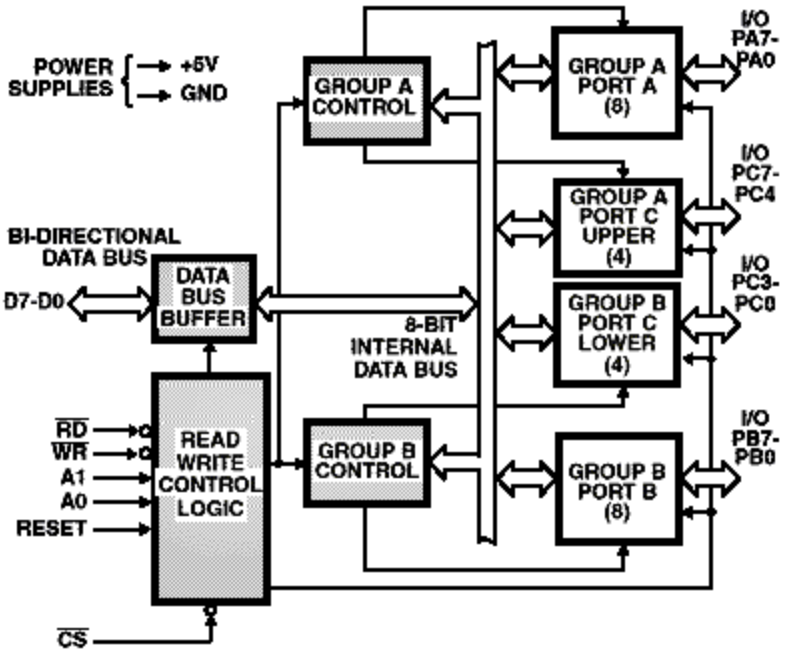
4. MOV A, R0

Description: This instruction moves the contents of the register R0 to accumulator.
No of bytes: 1 byte
Addressing mode: register

```
Example:    MOV R0, #35H    ; R0 = 35H
            MOV A, R0       ; A = 35H
```

d) Draw the block diagram of 8255

Ans:- **4 mks diagram**



e) Draw the format of SCON register of 8051 microcontroller and explain the function of each bit.

Ans: - (**SCON register format -- 2 Marks , Explanation of each bit -- 2 Marks)**

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

SM0	SCON.7	Serial port mode specifier
SM1	SCON.6	Serial port mode specifier.
SM2	SCON.5	Used for multiprocessor communication (Make it 0.)
REN	SCON.4	Set/ cleared by software to enable/ disable reception.
TB8	SCON.3	Not widely used.
RB8	SCON.2	Not widely used
TI	SCON.1	Transmit interrupt flag. Set by hardware at the beginning of the stop Bit in mode 1.Must be cleared by software.
RI	SCON.0	Receive interrupt flag. Set by hardware halfway through the stop bit time in mode 1.Must be cleared by software.

Note: Make SM2, TB8 and RB8 = 0.

SM0	SM1	
0	0	Serial Mode 0
0	1	Serial Mode 1, 8-bit data, 1 stop bit, 1 start bit
1	0	Serial Mode 2
1	1	Serial Mode 3

SM2: SM2 is the D5 bit of the SCON register.
This bit enables the multiprocessing capability of the 8051. Make SM2= 0 since we are not using the 8051 in a multiprocessor environment.

REN: The REN (receive enable) bit is D4 of the SCON register. The REN bit is also referred to as SCON.4 since SCON is a bit addressable register.

When the REN =1, it allows the 8051 to receive data on the RxD pin of the 8051. As a result if we want the 8051 to both transfer and receive data, REN must be set to 1.
By making REN=0, the receiver is disabled. Making REN=1 or REN=0 can be achieved by the instructions "SETB SCON.4" and "CLR SCON.4", respectively.
This bit can be used to block any serial data reception and is an extremely important bit in the SCON register.

TB8: TB8 (transfer bit 8) is bit D3 of SCON. It is used for serial modes 2 and 3. We make TB8=0 since it is not used in our applications.

RB8: RB8 (receive bit 8) is bit D2 of the SCON register. In serial mode 1, this bit gets copy of the stop bit when an 8 bit data is received. This bit (as is the case for TB8) is rarely used anymore. In all our applications we will make RB8=0. Like TB8, the RB8 bit is also used in serial modes 2 and 3.

TI: TI (transmit interrupt) is bit D1 of the SCON register.
This is an extremely important flag bit in the SCON register.
When the 8051 finishes the transfer of the 8 bit character, it raises the T1 flag to indicate that it is ready to transfer another byte. The TI bit is raised at the beginning of the stop bit.

RI: RI(receive interrupt) is the D0 bit of the SCON register. This is another extremely important flag in the SCON register. When the 8051 receives data serially via RxD, it gets rid of the start and stop bits and places the byte in the SBUF register. Then it raises the RI flag bit to indicate that a byte has been received and picked up before it is lost. RI is raised halfway through the stop bit.

Q.4 a) Attempt **any Three** **12 Marks**

i) With the help of suitable diagram describe the serial communication modes of 8051 microcontroller.

Ans:- :**1mark for each mode**

8051 micro controller communicate with another peripheral device through RXD and TXD pin of port3.controller have four mode of serial communication. This four mode of serial communication are below.

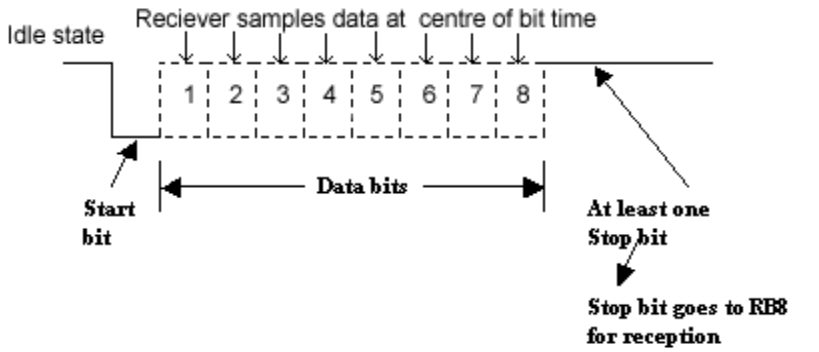
1. Serial data mode 0-fixed baud rate.
2. Serial data mode 1-variable baud rate.
3. Serial data mode 2 -fixed baud rate.
4. Serial Data mode 3 -variable baud rate.

1. **Serial Data Mode-0 (Baud Rate Fixed)**

In this mode, the serial port works like a shift register and the data transmission works synchronously with a clock frequency of f_{osc} /12. Serial data is received and transmitted through RXD. 8 bits are transmitted/ received aty a time. Pin TXD outputs the shift clock pulses of frequency f_{osc} /12, which is connected to the external circuitry for synchronization.The shift frequency or baud rate is always 1/12 of the oscillator frequency.

2. **Serial Data Mode-1 (standard UART mode)(baud rate is variable)**

In mode-1, the serial port functions as a standard Universal Asynchronous Receiver Transmitter (UART) mode. 10 bits are transmitted through TXD or received through RXD. The 10 bits consist of one start bit (which is usually '0'), 8 data bits (LSB is sent first/received first), and a stop bit (which is usually '1'). Once received, the stop bit goes into RB8 in the special function register SCON. The **baud rate is variable**.



$$f_{baud} = \frac{2^{SMOD}}{32} X \frac{fosc}{12 X [256-(TH1)]}$$

3. Serial Data Mode-2 Multiprocessor (baud rate is fixed)

In this mode 11 bits are transmitted through TXD or received through RXD.The various bits are as follows: a start bit (usually '0'), 8 data bits (LSB first), a programmable 9 th (TB8 or RB8)bit and a stop bit (usually '1'). While transmitting, the 9 th data bit (TB8 in SCON) can be assigned the value '0' or '1'. For example, if the information of parity is to be transmitted, the parity bit (P) in PSW could be moved into TB8.On reception of the data, the 9 th bit goes into RB8 in 'SCON', while the stop bit is ignored. The baud rate is programmable to either 1/32 or 1/64 of the oscillator frequency.

$$f_{baud} = (2^{SMOD}/64) f_{osc}$$

4. Serial Data Mode-3 - Multi processor mode(Variable baud rate)

In this mode 11 bits are transmitted through TXD or received through RXD.The various bits are: a start bit (usually '0'), 8 data bits (LSB first), a programmable 9 th bit and a stop bit (usually '1'). Mode-3 is same as mode-2, except the fact that the baud rate in mode-3 is variable (i.e., just as in mode-1).

$$f_{baud} = (2^{SMOD}/32) * (f_{osc} / 12 (256-TH1))$$

ii) Draw the format of PCON register of 8051 microcontroller and describe the function of each bit.

Ans:- **(PCON register format -- 2 Marks, Explanation of each bit -- 2 Marks)**

Register PCON controls processor power down, sleep modes and serial data baud rate. Only one bit of PCON is used with respect to serial communication. The seventh bit (b7)(SMOD) is used to generate the baud rate of serial communication.

Address: 87H

b7							b0
SMOD	—	—	—	GF1	GF0	PD	IDL

SMOD: Serial baud rate modify bit
GF1: General purpose user flag bit 1
GF0: General purpose user flag bit 0
PD: Power down bit
IDL: Idle mode bit

iii) Write an assembly language program to send message "WELCOME" serially at 9600 baud rate continuously.

Ans:- **4 mks for proper program**

```
MOV TMOD, #20H          ; timer 1, mode2
        MOV TH1,#-3             ; 9600 baud rate
        MOV SCON, #50H          ; 8-bit data,1 stop bit, REN enabled
        SETB TR1                ; Start timer 1
AGAIN:  MOV A, #"W"             ; transfer "W"
        ACALL MESSAGE           ; Some delay
        MOV A, #"E"             ; transfer "E"
        ACALL MESSAGE
        MOV A, #"L"             ; transfer "L"
        ACALL MESSAGE
        MOV A, #"C"             ; transfer "C"
        ACALL MESSAGE
        MOV A, #"O"             ; transfer "O"
        ACALL MESSAGE
        MOV A, #"M"             ; transfer "M"
        ACALL MESSAGE
        MOV A, #"E"             ; transfer "E"
        ACALL MESSAGE
        SJMP AGAIN

MESSAGE: MOV SBUF, A;
        JNB T1, HERE;
        CLR T1;
        RET
```

iv) Write an assembly language program to multiply two 8-bit numbers stored in internal RAM locations 10H and 11H.Store the result at 12H and 13H.

Ans:-: **4 mks for proper program**

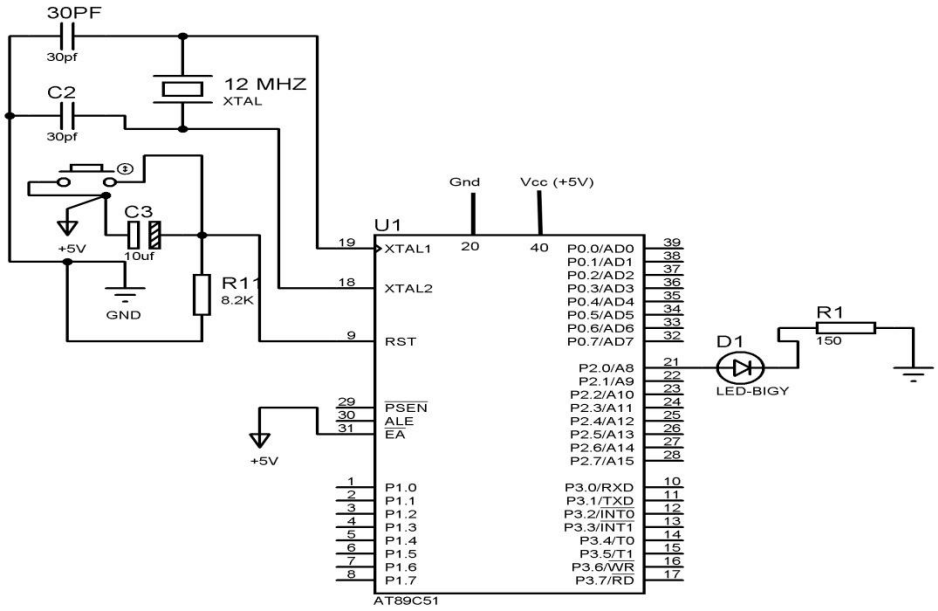
```
        MOV 10H, # 02H          ; store first 8-bit no. in 10H
        MOV 11H, #03H           ; store second 8-bit no. in 11H
        MOV A, 10H              ; move first number to A
        MOV B, 11H              ; move second number to B
        MUL AB                  ; multiply the numbers
        MOV 12H, A              ; move LSB to 12H
        MOV 12H, B              ; move MSB to 13H
HERE:   SJMP HERE
```

b) Attempt **any one** **06 Marks**

i) Draw the interfacing diagram of 8 LEDs to port 2 of 8051 microcontroller. Write an assembly language program to make LED ON and OFF after certain delay.

Ans:-Diagram- 3 marks , Program- 2marks- Prog -1 mark Delay (any delay generating program)

(Note: LEDs should be connected to all the 8 lines of Port 2)




```
        MOV A, # FFH          ; Store FFH in A
BACK:   MOV P2, A             ; move FF to P2 to glow all the LEDs
        ACALL DELAY           : wait for some time
        CPL A                 ; turn off the LEDs
        SJMP BACK
Delay Program:
        AGAIN: MOV R3, #255
        DJNZ R3, AGAIN
        RET
```

ii) Write an assembly language program to find largest number out of ten numbers stored in internal RAM location s 40H onwards, store the result at 50H.

Ans:-

```
        MOV R1, 0AH           ; initialize the counter
        MOV R0, #40H          ; initialize the memory pointer
        DEC R1                ; decrement counter by one
        MOV A,@R0             ; load number in accumulator
        MOV B, A              ; move that number to register B
UP:     INC R0                ; increment the memory pointer
        MOV A,@R0             ; read the next number in A
```

```
        CJNE A, B, DOWN      ; compare the first two numbers, if not equal
                               go to DOWN
        AJMP NEXT            ; else go to NEXT
DOWN:   JC NEXT              ; if number in A is greater then go to NEXT
        MOV B, A             ; else move the number in register B
NEXT:   DJNZ R1, UP          ; decrement the counter by one, if count ≠ 0,
                               then go to UP
        INC R0               ; increment the memory pointer
        MOV A,B
        MOV 50H, A           ; store result at memory location 50H
HERE:   SJMP HERE
```

Q.5 Attempt **any four** **16 Marks**

a) Draw the format of IE register of 8051 microcontroller and describe the function of each bit.

Ans:-(IE **register format -- 2 Marks ,Explanations of each bit -- 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	—	ET2	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.*
ET2	IE.5	Enable or disable the Timer 2 overflow or capture interrupt (8052 only).
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.

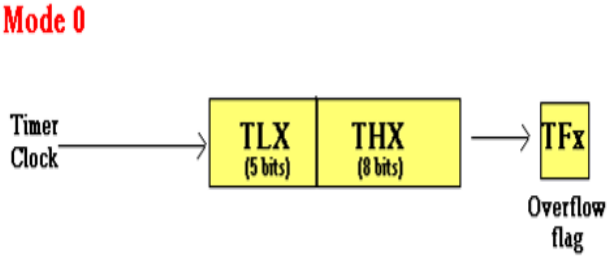
*User software should not write 1s to reserved bits. These bits may be used in future MCS-51 products to invoke new features. In that case, the reset or inactive value of the new bit will be 0, and its active value will be 1.

b) With the help of neat diagram describe the timer modes of 8051microcontroller.

Ans:- **1mark for 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.

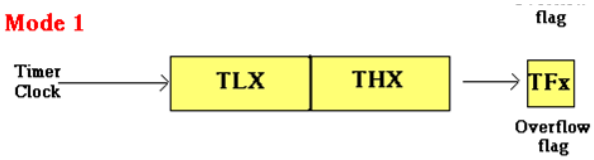
Mode 0:



In mode0 the register THX is used as 8 bit counter and TLX is used as 5 bit counter. The pulse i/p is divided by $(32)_{10}$so that TH counts. Hence original oscillator frequency is divided by $(384)_{10}$.The timer flag is set when THX rolls

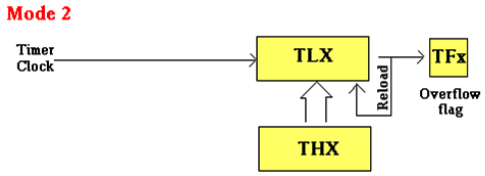
over from FF to 00H.

Mode 1:



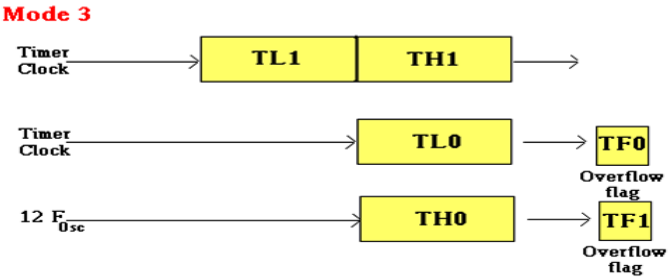
It is similar to Mode 0 except TLX is configured as a full 8-bit counter. Hence pulse input is divided by 256_{10} so that TH counts the timer flag is set when THX rolls over from FF to 00H.

Mode 2



In this mode only TLX is used as 8-bit counter. THX is used to hold the value which is loaded in TLX initially. Everytime TLX overflows from FFH to 00H the timer flag is set and the value from THX is automatically reloaded in TLX register.

Mode 3



In this mode, timer 0 becomes two completed separate 8-bit timers. TL0 is controlled by gate arrangement of timer 0 and sets timer 0 flag when it overflows. TH0 receives the timer clock under the control of TR1 bit and sets TF1 flag when it overflows. Timer 1 may be used in mode 0, 1 and 2 with one important exception that no interrupt will be generated by the timer when the timer 0 is using TF1 overflow flag.

c) Write an assembly language program for 8051 microcontroller to generate a square wave of 1KHz on P1.5. Assume crystal frequency as 11.0592MHz.

Ans:- Calculations --- 1Mark , Delay--- 1Mk, Program ---2 Marks

Crystal frequency= 12 MHz

I/P clock = $(11.059 \times 10^6)/12 = 1000000 = 921.58$KHz

$T_{in} = 1.085\mu$ sec

For 1 kHz square wave

$F_{out} = 1$ KHz

$T_{out} = 1/ 1X\ 10^3$

$T_{out} = 1000\mu$ sec

Consider half of it = $T_{out} = 500\mu$ sec

$N = T_{out} / T_{in} = 500/1.085 = 460.82$

$65536-461 = (65075)_{10} = (FE33)_{16}$

Program:-

```
      MOV TMOD, # 01H          ; Set timer 0 in Mode 1, i.e., 16 bit timer
L2:   MOV TLO, # 33H           ; Load TL register with LSB of count
      MOV THO, # FEH           ; load TH register with MSB of count
      SETB TRO                 ; start timer 0
L1:   JNB TFO, L1              ; poll till timer roll over
      CLR TRO                  ; stop timer 0
      CPL P1.5                 ; complement port 1.5 line to get high or low
      CLR TFO                  ; clear timer flag 0
      SJMP L2                  ; re-load timer with count as mode 1 is not auto reload
```

d) Write an assembly language program for 8051 microcontroller to receive 10 bytes of data serially and save them in accumulator. Assume baud rate is 4800.

Ans: Proper program 4 mks

```
      MOV R2, # 0AH            ; initialize the counter for 10 bytes
      MOV TMOD, #20H           ; timer 1, mode 2
      MOV TH1, #-6             ;       4800 baud
      MOV SCON, # 50H          ; 8-bit data, I stop bit, REN enabled
      SETB TR1                 ; start timer1
HERE: JNB RI HERE              ; wait for the character to come in
      MOV A, SBUF              ; save incoming byte in A
      MOV P1,A                 ; send to port 1(This instruction is optional)
      CLR RI                   ; get ready to receive next byte
      DJNZ R2, HERE            ; check if all ten bytes received, if not then wait else stop.
      END
```

e) List the I/O ports of 8051 microcontroller and describe the alternative functions of ports.

Ans: list 2 mks, alternate functions ½ mks each

There are four ports available with 8051 microcontroller as,

1. Port 0
2. Port 1
3. Port 2
4. Port 3

1. Port 0: It can be used as
   a) Simple input/output
   b) Bidirectional low order address / data bus for external memory.
2. Port 1: It does not have any dual function just used as simple input/output port.
3. Port 2: It can be used as
   a) Simple input/output port
   b) the alternative use is to supply a higher order address byte in conjunction with the port 0 lower order byte to address external memory.
4. Port 3: It can be used as

a) Simple input/output port

b) Alternate functions of port 3 are as given below

Pin	Name	Alternate Function
P3.0	RXD	Serial input line
P3.1	TXD	Serial output line
P3.2	$\overline{INT0}$	External interrupt 0
P3.3	$\overline{INT1}$	External interrupt 1
P3.4	T0	Timer 0 external input
P3.5	T1	Timer 0 external input
P3.6	$\overline{WR}$	External data memory write strobe
P3.7	$\overline{RD}$	External data memory read strobe

Q.6 Attempt **any four** **16 Marks**

a) Draw the format of TCON register of 8051 microcontroller and state the functions of each bit.

Ans: **(Format: 2 Marks Explanation: 2 Marks)**

TCON: TIMER/COUNTER CONTROL REGISTER. BIT ADDRESSABLE.

TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1(TCON. 7)- Timer 1 overflow flag. Set by hardware when the Timer/Counter 1 overflows. Cleared by hardware as processor vectors to the interrupt service routine.

TR1(TCON. 6)- Timer 1 run control bit. Set/cleared by software to turn Timer/Counter 1ON/OFF.

TF0(TCON. 5)- Timer 0 overflow flag. Set by hardware when the Timer/Counter 0 overflows. Cleared by hardware as processor vectors to the service routine.

TR0(TCON. 4)- Timer 0 run control bit. Set/cleared by software to turn Timer/Counter 0 ON/OFF.

IE1(TCON. 3) - External Interrupt 1 edge flag. Set by hardware when External Interrupt edge is detected. Cleared by hardware when interrupt is processed.

IT1(TCON. 2) - Interrupt 1 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt.

IE0(TCON. 1)- External Interrupt 0 edge flag. Set by hardware when External Interrupt edge detected. Cleared by hardware when interrupt is processed.

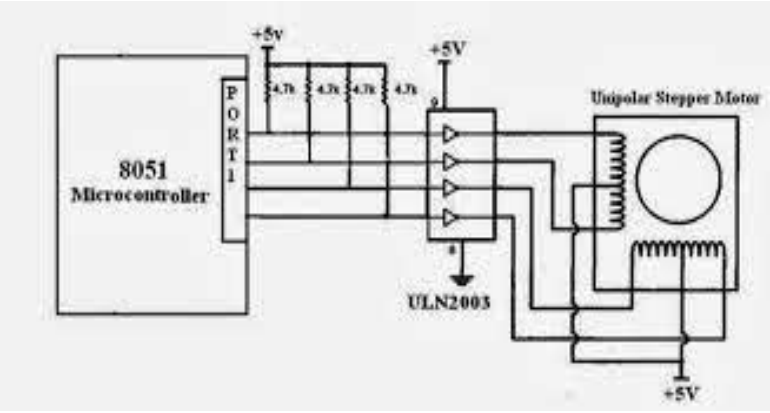
IT0(TCON. 0)- Interrupt 0 type control bit. Set/cleared by software to specify falling edge/low level triggered External Interrupt.

b) List the priority of interrupts of 8051 microcontroller with respective interrupt destinations.
Ans: 4 mks for proper answer

Interrupt Source	Vector address	Interrupt priority
External Interrupt 0 –INT0	0003H	1
Timer 0 Interrupt	000BH	2
External Interrupt 1 –INT1	0013H	3
Timer 1 Interrupt	001BH	4
Serial Interrupt	0023H	5

All the 5 interrupts of 8051 has got different priorities. Interrupts are serviced according to it's priority order. From the table above, you can see that INT0 has the highest priority of 1 and Timer 0 comes next with priority value 2. The order of priority works like this – consider a case where two interrupts are raised at the same time – one from INT0 and another from Timer 1 interrupt. In such a case, processor would serve the interrupt according to it's priority. In our case INT0 is of high priority (priority order 1)and Timer 1 interrupt is of low priority (priority order 4). So processor will execute ISR of INTO first and then later, after finishing ISR of INT0, processor will begin executing ISR of Timer 1 interrupt. (explanation not compulsory)

c) Draw the interfacing of stepper motor with 8051 microcontroller.
Ans: (4- marks for diagram, any other appropriate diagram should be consider)



d) Write an assembly language program for 8051 microcontroller to turn ON LED connected to P1.7 pin on the occurrence of INT0 and turn OFF LED after some delay.
Ans: 3marks- Program , 1-mark Delay(any delay generating program)

```
        MOV A, # 80H              ; Store 80H in A
        HERE: JNB P3.2 HERE       ; wait for INT0 interrupt
BACK: MOV P1, A                   ; move 80 to P1 to glow the LED connected
                                    to P1.7
        ACALL DELAY               : wait for some time
        MOV A, #00H               ; turn off the LED
         SJMP BACK
```

Delay Program

```
AGAIN: MOV R3, #255
       DJNZ R3, AGAIN
       RET
```

e) Describe any four selection factors of microcontroller.

Ans: (1 mark for each criterion – any four should be considered)

The selection of microcontroller depends upon the type of application. The following factors must be considered while selecting the microcontroller.

1. Word length: The word length of microcontroller is either 8, 16 or 32 bit. As the word length increases, the cost, power dissipation and speed of the microcontroller increases.
2. Power dissipation: It depends upon various factors like clock frequency, speed, supply voltage, VLSI technology etc. For battery operated embedded systems, we must use low power microcontrollers.
3. Clock frequency: The speed of an embedded system depends upon the clock frequency. The clock frequency depends upon the application.
4. Instruction Set: On the basis of instructions microcontrollers are classified into two categories 1. CISC 2. RISC.
   CISC system improves software flexibility. Hence it is used in general purpose systems.
   RISC improves speed of the system for the particular applications.
5. Internal resources: The internal resources are ROM, RAM, EEPROM, FLASH ROM, UART, TIMER, watch dog timer, PWM, ADC, DAC, network interface, wireless interface etc. It depends upon the application for which microcontroller is going to be used.
6. I/O capabilities: The number of I/O ports, size and characteristics of each I/O port, speed of operation of the I/O port, serial port or parallel ports. These are the considerations needed to ascertain.