PANIMALAR INSTITUTE OF TECHNOLOGY

(JAISAKTHI EDUCATIONAL TRUST)

CHENNAI 602 103

DEPARTMENT OF ECE

III YEAR – VI SEMESTER

# EC8691- MICROPROCESSORS AND MICROCONTROLLERS

QUESTION BANK

**EC8691 MICROPROCESSORS AND MICROCONTROLLERS L T PC 3 0 0 3**

**OBJECTIVES:**

- To understand the Architecture of 8086microprocessor.
- To learn the design aspects of I/O and Memory Interfacingcircuits.
- To interface microprocessors with supportingchips.
- To study the Architecture of 8051microcontroller.
- To design a microcontroller basedsystem

**UNITI THE8086 MICROPROCESSOR 9**

Introduction to 8086 – Microprocessor architecture – Addressing modes - Instruction set and assembler directives – Assembly language programming – Modular Programming - Linking and Relocation - Stacks - Procedures – Macros – Interrupts and interrupt service routines – Byte and String Manipulation.

**UNITII 8086 SYSTEM BUS STRUCTURE 9**

8086 signals – Basic configurations – System bus timing –System design using 8086 – I/O programming – Introduction to Multiprogramming – System Bus Structure – Multiprocessor configurations – Coprocessor, Closely coupled and loosely Coupled configurations – Introduction to advancedprocessors.

**UNITIII I/OINTERFACING 9**

Memory Interfacing and I/O interfacing - Parallel communication interface – Serial communication interface – D/A and A/D Interface - Timer – Keyboard /display controller – Interrupt controller – DMA controller – Programming and applications Case studies: Traffic Light control, LED display , LCD display, Keyboard display interface and Alarm Controller.

**UNITIVMICROCONTROLLER 9**

Architecture of 8051 – Special Function Registers(SFRs) - I/O Pins Ports and Circuits - Instruction set - Addressing modes - Assembly language programming.

**UNIT VINTERFACINGMICROCONTROLLER 9**

Programming 8051 Timers - Serial Port Programming - Interrupts Programming – LCD & Keyboard Interfacing - ADC, DAC & Sensor Interfacing - External Memory Interface- Stepper Motor and Waveform generation - Comparison of Microprocessor, Microcontroller, PIC and ARM processors

**TOTAL: 45 PERIODS**

**TEXT BOOKS:**


1. Yu-Cheng Liu, Glenn A.Gibson, -Microcomputer Systems: The 8086 / 8088 Family - Architecture, Programming and Design‖, Second Edition, Prentice Hall of India, 2007. (UNIT I- III)
2. Mohamed Ali Mazidi, Janice GillispieMazidi, RolinMcKinlay, -The 8051 Microcontroller and Embedded Systems: Using Assembly and C‖, Second Edition, Pearson education, 2011. (UNIT IV-V)


**REFERENCES:**


1. Doughlas V.Hall,-Microprocessors and Interfacing, Programming and Hardware,TMH,2012
2. A.K.Ray, K.M.Bhurchandi, "Advanced Microprocessors and Peripherals" 3rd edition, Tata McGrawHill,2012

## COURSE OUTCOMES

### At the end of course, students will have ability to

| | |
|---|---|
| **CO1** | Analyse the architecture, instruction set of microprocessor and execute programs based on 8086 microprocessor. |
| **CO2** | Design aspects of I/O and Memory Interfacing circuits. |
| **CO3** | Design and interface I/O circuits. |
| **CO4** | Apply and analyse the architecture, instruction sets of Microcontroller for developing assembly language programs. |
| **CO5** | Design and implement 8051 microcontroller based systems |
| **CO6** | Acquire knowledge of PIC and ARM processors |

# UNIT-I
# THE 8086 MICROPROCESSOR

## PART-A

**1. What is microprocessor? R**

A microprocessor is a multipurpose, programmable, clock-driven, register-based electronic device that reads binary information from a storage device called memory, accepts binary data as input and processes data according to those instructions, and provides result as output.

**2. What is Accumulator? R**

The Accumulator is an 8-bit register that is part of the arithmetic/logic unit (ALU). This register is used to store 8-bit data and to perform arithmetic and logical operations. The result of an operation is stored in the accumulator. The accumulator is also identified as register A.

**3. What is stack?(EE2354April/May2013) R**

The stack is a group of memory locations in the R/W memory that is used for temporary storage of binary information during the execution of a program

**4. What is a subroutine program? R**

A subroutine is a group of instructions written separately from the main program to perform a function that occurs repeatedly in the main program. Thus subroutines avoid the repetition of same set of instructions in the main program.

**5. Define addressing mode. U**

Addressing mode is used to specify the way in which the address of the operand is specified within the instruction.

**6. Define instruction cycle. U**

It is defined as the time required to complete the execution of an instruction.

**7. Write a program to add a data byte located at offset 0500H in 2000H segment to another data byte available at 0600H in the same segment and store the result at 0700H in the same segment. C**

```
MOV AX, 2000H; initialize DS with value
MOVDS, AX; 2000H
MOV AX, [500H]; Get first data byte from 0500H offset
ADD AX, [600H]; Add this to the second byte from 0600H
MOV [700H], AX; store AX in 0700H
HLT; Stop.
```

**8. What are the different types of addressing modes of 8086 instruction set? [April/May 2019] [April/May 2018](Nov/Dec2013) (Apr/May 2015) R**

The different addressing modes are:

i. Immediate
ii. Direct
iii. Register
iv. Register indirect
v. Indexed
vi. Register relative
vii. Based indexed

viii. Relative based indexed.

**9. What are the different types of instructions in 8086 microprocessor? (May/jun2011) R**

The different types of instructions in 8086 microprocessor are:

i. Data copy / transfer instructions
ii. Arithmetic and logical instructions
iii. Branch instructions
iv. Loop instruction
v. Machine control instruction
vi. Flag manipulation instruction
vii. Shift and rotate instruction
viii. String instruction

**10. What is assembly level programming? R**

A program called assembler is used to convert the mnemonics of instruction and data into their equivalent object code modules. The object code modules are further converted into executable code using linker and loader programs. This type of programming is called
assembly level programming.

**11. What is a stack? R**

Stack is a top-down data structure, whose elements are accessed using a pointer that is implemented using the SS and SP registers. It is a LIFO data segment.

**12. How is the stack top address calculated? A**

The stack top address is calculated using the contents of the SS and SP register. The contents of stack segment (SS) register is shifted left by four bit positions (multiplied by(0h)) and the resulted 20-bit content is added with the 16-bit offset value of the stack pointer(SP) register.

```
SS    -     5000H
SP    -     2050H
            SS    -        0101 0000 0000 0000
      10H * SS    -      0101 0000 0000 0000 0000
            SP    -        0010 0000 0101 0000
      Stack- top         0101 0010 0000 0101 0000
      Address              5    2    0    5    0
```

**13. What are macros? R [April/May 2019]**

Macros are small routines that are used to replace strings in the program. They can have parameters passed to them, which enhances the functionality of the micro itself.

**14. How are constants declared? A**

Constants are declared in the same way as variables, using the format:

**Const–Label EQU 012h**

When the constants label is encountered, the constant numeric value is exchanged for the string.

**15. Write an assembly language program for a 16-bit increment and will not affect the contents of the accumulator. C**

```
MACRO inc16variable; Increment two bytes starting at "variable"
Local INC16 End
INC variable; Increment the low 8 bits PUSH ACC
```

```
MOV A        variable; Are the incremented low 8 bits = 0?
JNZ INC 16 End
INC variable + 1
Inc16 End;Yes–increment the upper 8 bits
POP ACC
END MAC
```

**16. What will happen if a label within a macro is not declared local? R**

If a label within a macro is not declared local, then at assembly time, there will be two types of errors:

I. The first will state that there are multiple labels in the source.

II. The second will indicate that jump instructions don't know which one to use.

**17. Write an assembly language program to load the accumulator with a constant value. C**

```
MACRO invert value
if (value==0)
MOV A, #1
else
clr A
end if
END MAC.
```

**18. What is the difference between the microprocessor and microcontroller? R**

Microprocessor does not contain RAM, ROM and I/O ports on the chip. But a microcontroller contains RAM, ROM and I/O ports and a timer all on a single chip.

**19. What is assembler?(NOV/DEC2014) R**

The assembler translates the assembly language program text which is given as input to the assembler to their binary equivalents known as object code. The time required to translate the assembly code to object code is called access time. The assembler checks for syntax errors & displays them before giving the object code.

**20. What is loader? R**

The loader copies the program into the computer's main memory at load time and begins the program execution at execution time.

**21. What is linker? R**

A linker is a program used to join together several object files into one large object file. For large programs it is more efficient to divide the large program modules into smaller modules. Each module is individually written, tested & debugged. When all the modules work they are linked together to form a large functioning program.

**22 .Explain ALIGN & ASSUME.(Nov/Dec 2010,April/may2011) U**

The ALIGN directive forces the assembler to align the next segment at an address divisible by specified divisor. The format is ALIGN number where number can be 2,4, 8 or 16. Example ALIGN 8.

The ASSUME directive assigns a logical segment to a physical segment at any given time. It tells the assembler what address will be in the segment registers at execution time.

Example ASSUME CS: code, DS: data, SS: stack

**23. Explain PTR & GROUP. U**

A program may contain several segments of the same type. The GROUP directive collects them under a single name so they can reside in a single segment, usually a data segment. The format is Name GROUP Seg-name,…..Seg-name

PTR is used to assign a specific type to a variable or a label. It is also used to override the declared type of a variable.

**24. Explain PROC & ENDP (April/May 2010) U**

PROC directive defines the procedures in the program. The procedure name must be unique. After PROC the term NEAR or FAR are used to specify the type of procedure. Example FACT PROC FAR.

ENDP is used along with PROC and defines the end of the procedure.

**25. Explain SEGMENT & ENDS. U**

An assembly program in .EXE format consists of one or more segments. The starts of these segments are defined by SEGMENT and the end of the segment is indicated by ENDS directive. Format Name SEGMENT.

**26**. **Define SOP(Nov/Dec2010) R**

The segment override prefix allows the programmer to deviate from the default segment

Eg : MOV CS: [BX] , AL

**27. Define variable**. R

A variable is an identifier that is associated with the first byte of data item. In assembly language statement: COUNT DB 20H, COUNT is the variable.

**28. What are procedures? R**

Procedures are a group of instructions stored as a separate program in memory and it is called from the main program whenever required. The type of procedure depends on where the procedures are stored in memory. If it is in the same code segment as that of the main program then it is a near procedure otherwise it is a far procedure.

**29. Explain the linking process. U**

A linker is a program used to join together several object files into one large object file. The linker produces a link file which contains the binary codes for all the combined modules. It also produces a link map which contains the address information about the link files. The linker does not assign Absolute addresses but only relative address starting from zero, so the programs are relocatable& can be put anywhere in memory to be run.

30. **Compare Procedure & Macro.(April/May2011) A/E**

| **Procedure** | **Macro** |
|---|---|
| Accessed by CALL & RET instruction during program execution | Accessed during assembly with name to macro when defined |
| Machine code for instruction is put only Once in the memory | Machine code is generated for instruction each time when macro is called |

| | |
|---|---|
| With procedures less memory is required | With macro more memory is required |
| Parameters can be passed in registers, memory locations or stack | Parameters passed as part of statement Which calls macro |

**31. What is the maximum memory size that can be addressed by 8086? (April/May 2014) (Nov/Dec 2014) R**

In 8086, a memory location is addressed by 20 bit address and the address bus is 20 bit address and the address bus is 20 bits. So it can address up to one megabyte ($2^{20}$) of memory space.

**32. How many data lines and address lines are available in 8086? R**

Address lines= 20 bit address bus
Data lines= 16 bit data bus

**33. What information is conveyed when Qs1 and Qs0 are 01? R**

Qs1 and Qs0 are output signals that reflect the status of the instruction queue. When Qs1 and Qs0 are 01 , then queue has first byte of an opcode.

**34.What is the addressing mode of MOV AX, 55H (BX) (SI) ? R**

MOV AX, 55H (BX) (SI) – Base Indexed memory addressing mode.

**35.What are the 8086 interrupt types? [Apr/May 2015] R**

Dedicated interrupts
- Type 0: Divide by zero interrupt
- Type 1: Single step interrupt
- Type 2:Nonmaskable interrupt
- Type 3: Breakpoint
- Type 4: Overflow interrupt

Software interrupts: Type 0-255

**36. What is interrupt service routine?[Nov/Dec2018][NOV/DEC 2011] R**

Interrupt means to break the sequence of operation. While the CPU is executing a program an interrupt breaks the normal sequence of execution of instructions & diverts its execution to some other program. This program to which the control is transferred is called the interrupt service routine.

**37. Calculate the physical address for fetching the next instruction to be executed, in 8086? A**

The physical address is obtained by appending four zeros to the content present in CS register and then adding the content of IP register with the above value .

For example, assuming the content of
CS = 1200 H
IP = 0345 H
CS= 0001 0010 0000 0000 0000
0000 0011 0100 0101
---
0001 0010 0011 0100 0101 – Physical address=12345 H

**38. If the execution unit generates effective address of 43A2 H and the DS register contains 4000 H. What will be the physical address generated by the BIU? What is the Maximum Size of the data segment? A**

Effective Address 43A2H
Physical Address 40000H

443A2H

Maximum size of the DS is $2^{16}$

39. **Calculate the physical address, when segment address is 1085H and effective address is 4537H. [Nov/Dec 2015] A**

Segment address - 1085H
Effective address - 4537H
---
Physical address - 14D87H
---

**40. Show how the 2 byte INT instruction can be applied for debugging. [Nov/Dec2015] C**

INT type

The INT instruction is used as a debugging and in case where single stepping provides more detail then is wanted, by inserting INT instructions at key points called break points.

**41. List the flags of 8086.[April/May 2019] [May/June 2016] [April/May 2021] R**

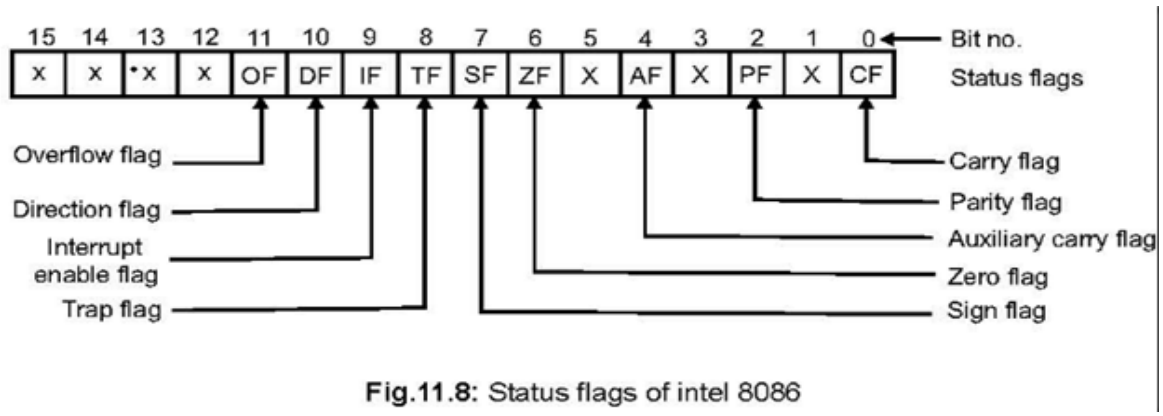



**Fig.11.8:** Status flags of intel 8086

- OF - Overflow Flag. Set if signed number exceeds capacity of result. ...
- DF - Direction Flag. Set by user to indicate a direction (0=forward, 1=backward) ...
- IF - Interrupt Flag. Set by user to disable hardware interrupts temporarily. ...
- TF - Trap Flag. Used by debuggers.
- SF - Sign Flag. ...
- ZF - Zero Flag. ...
- AF - Aux. ...
- PF - Parity Flag.

**42.The offset address of a data is (341B)H and the data segment register value is (1234)H. What is the physical address of data?[APR/MAY 2017]**

The physical address is obtained by appending four zeros to the content present in DS register and then adding the offset address of a data with the above value .

DS = 1234 H
Offset = 341B H
DS= 0001 0010 0011 0100 0000
0011 0100 0001 1011
---
01 1 0111 0101 1011 – Physical address=1575B H

**43.Define stack register.[APR/MAY 2017] R**

In 8086, the main stack register is called stack pointer - SP. The stack segment register (SS) is usually used to store information about the memory segment that stores the call stack of currently executed program. SP points to current stack top. By default, the stack grows downward in memory, so newer values are placed at lower memory addresses. To push a value to the stack, the PUSH instruction is used. To pop a value from the stack, the POP instruction is used.

**44. List the modes of operation in 8086. [NOV/DEC 2017] R**

(i). Minimum mode.

(ii). Maximum mode.

**45. Define Macros. [NOV/DEC 2017] R**

Macro is a group of instructions. The macro assembler generates the code in the program each time where the macro is 'called'. Macros can be defined by MACRO and ENDM assembler directives. Creating macro is very similar to creating a new opcode that can be used in the program. Macro sequences execute faster than procedures because there are no CALL and RET instructions to execute. The assembler places the macro instructions in the program each time when it is invoked.

**46. What are Byte and String Manipulations?[Nov/Dec2018] R**

A String is s series of data byte or word available in memory at consecutive locations. It is either referred as byte string or word string. Their memory is always allocated in a sequential order. Instructions used to manipulate strings are called string manipulation instructions.

REP, MOVSB/MOVSW, CMPSB/CMPSW, SCASB/SCASW, STOSB/STOSW, LODSB/LODSW.

**47. Define Stack pointer.[April/May 2019] [April/May 2018] R**

In 8086, the main stack register is called stack pointer - SP. The stack segment register (SS) is usually used to store information about the memory segment that stores the call stack of currently executed program. SP points to current stack top. By default, the stack grows downward in memory, so newer values are placed at lower memory addresses. To push a value to the stack, the PUSH instruction is used. To pop a value from the stack, the POP instruction is used.

**48. Given that (Bx) = 0158 (DI) = 10A5 Displacement = 1B57 (DS) = 2100. Determine the effective address and physical address for the following addressing modes.**

**[April/May 2019] A/E**

**(a) Register Indirect**

**(b) Relative based indexed.**

Physical Address (PA) = [Segment Address (SA) x 10H] + Effective Address (EA)

**a)Register Indirect**

Effective Address = Pointer Register / Index Register (BP/DI/SI)

EA = DI = 10A5

Physical Address (PA) = [DSx10] + EA = [2100x10] + 10A5

= 21000+10A5 = **220A5 H**

**(b) Relative based indexed**

Effective Address = BX + DI + Displacement

= 0158+10A5+1B57

0158 = 0000 0001 0101 1000

10A5 = 0001 0000 1010 0101

1B57 = 0001 1011 0101 0111

---

0010 1101 01010100 = **2D54H (EA)**

Physical Address (PA) = [DSx10] + EA = [2100x10] + 2C54

= 21000+2C54 = 23C54

**PA = 23D54H**

**49. List the types of interrupts in 8086. R [April/May 2021]**

The 8086 has two hardware interrupt pins, i.e. NMI and INTR. NMI is a non-maskable interrupt and INTR is a maskable interrupt having lower priority. One more interrupt pin associated is INTA called interrupt acknowledge.

**50. What are the advantages of using the MOVS and CMPS instructions over the MOV and CMP instructions while working with strings? (Nov/Dec 2021) U**

The string instructions MOVS and CMPS are more advantages than the MOV and CMP instructions. The advantages are

1. They are only 1 byte long.
2. Both operands are memory operands.
3. The auto indexing operations will decrease overall processing time.

**51. If a data segment begins at address 2400H, what is the address of the last location in the segment? (Nov/Dec 2021) E**

Segment address = 2400H

Segment starting address = 24000H

Segment ending address = starting address +Memory chip size -1

= 24000H+FFFFH = 33FFFH

## PART-B

1. (a) Write an assembly language program in 8086 to search the largest data in the array. (6**) C** **(April/May 2011)**
   (b) Explain the various status flags in 8086. (6) U **(Nov/Dec2011)**
2. (a) Explain the following assembler directive in 8086. (8) U **(April/May2013)(Apr/May2015)** i. ASSUME ii. EQU iii. DW iv. DD
3. (a)Write short notes on Macro.(6) C **(April/May 2012)**
   (b) Explain the function of assembler directives. (10**)** U **(April/May2011) (Nov/Dec 2014)**
4. Explain the architecture of 8086 **(16)**U **(Nov/Dec2014), (April/May2011) [Nov/Dec2015]**
5. (a) Explain the register organization of 8086 **(10)**U **(April/May2013)**
   (b) Explain the pin diagram of 8086 (6) U
6. Discuss the instruction set of 8086 in detail (8) U **(April/May 2011)**
7. Explain the interrupt and types?(8) U **(Nov/Dec2010) (April/May 2011)**
8. Explain briefly about the internal hardware architecture of 8086 microprocessor with a neat diagram. (13) U **[April/May 2019][April/May 2018](Apr/May 2015)**
9. Write an assembly language program in 8086 to convert BCD data – binary data.(6) C **(Apr/May 2015)**
10. Explain briefly about Interrupt handling process in 8086. (8) U **(Apr/May 2015)**
11. Explain in detail about the interrupts and interrupt service routines of 8086. (16) U **[Apr/May 2015] [Nov/Dec 2015]**
12. i) Explain the Data transfer, arithmetic and branch instructions with examples.(9) U
    (ii) Write an 8086 ALP to find the sum of numbers in an arry of 10 element. (7) C **[May/June 2016]**
13. Define interrupts and their types. Write in detail about interrupt service routine. (16) C **[May/June 2016]**
14. Draw and explain the architecture of 8086 with neat diagram. (13) U **[April/May 2017]**
15. Describe the interrupts of 8086 and its types with service routine. (13) U **[April/May 2017]**
16. Draw the architecture and explain the functional units of 8086. (13) U **[Nov/Dec 2017]**
17. Describe the interrupts of 8086 and its types with service routines. (13) U **[Nov/Dec 2017]**
18. Explain in detail about the interrupts and interrupt service routines of 8086. (13) U **[April/May 2018]**
18. For 8086 Microprocessor what are the instruction set and assembler directives? (13) U**[Nov/Dec2018]**
19. Explain the various addressing modes of 8086. (13) U **[Nov/Dec2018] (April/May2011) (Nov/Dec 2014) [Nov/Dec2015]**
20. What is Interrupt and interrupt routine. Explain interrupt sequence for 8086 Microprocessor and interrupt pointers. (13) U **[April/May 2019]**

21. How would you show your understanding of internal hardware architecture of 8086 microprocessor with neat diagram? U **[April/May 2021]**

22. What are the various addressing modes of 8086 microprocessor with examples? R **[April/May 2021]**

**23.** i) Write an 8086 ALP to find the sum of numbers in an array of 10 elements U **[April/May 2021]**

ii) Write an 8086 ALP to find the largest number and smallest number in an array. U

24. Express the categories under which the instructions in the instruction set of the 8086 microprocessor are grouped. Explain the operation of any two instructions in each group. **(Nov/Dec 2021) U**

25. State the advantages of modular programming and illustrate the process by which the modules assembled separately are linked together and programs are prepared for execution. **(Nov/Dec 2021) U**

## ASSIGNMENT QUESTIONS

1. What is memory segmentation? Describe physical memory organization. How to calculate the physical memory address. **(Understand)**
2. Write the Assembly Language Program(ALP) to move a block of data without overlap**. (Create)**
3. Write the Assembly Language Program(ALP) to perform sorting of an array in ascending order**. (Create)**

## UNIT II
## 8086 SYSTEM BUS STRUCTURE

### PART A

**1. Differentiate between minimum and maximum mode.**

**[April/May 2018](April/May2010) A/E**

| | Minimum mode | Maximum mode |
|---|---|---|
| i. | A processor is in minimum mode when MN/MX pin is strapped to +5V. | A processor is in maximum mode when MN/MX is grounded. |
| ii. | All the control signals are given out by microprocessor chip itself. | The processor derive the status signals S2 , S1 and S0. Another chip called bus controller derives control signals using this status information. |
| iii. | There is a single microprocessor. | There may be more than one microprocessor. |

**2. Give any four pin definitions for the minimum mode. (Nov/Dec2008) C**

| Symbol | | Description |
|---|---|---|
| i. | INTA | Indicates recognition of an interrupt request. Consists of two negative going pulses in two consecutive bus cycles. |
| ii. | ALE | Outputs a pulse at the beginning of the bus cycle and to indicate an address available on address pins. |
| iii. | HLDA | Outputs a bus grant to a requesting master. |
| iv. | HOLD | Receives bus requests from bus masters. |

**3. What are the pins that are used to indicate the type of transfer in minimum mode? R**

The M/IO, RD, WR lines specify the type of transfer. It is indicated in the following table:

| M/IO | RD | WR | |
|---|---|---|---|
| 0 | 0 | 1 | I/O Read |
| 0 | 1 | 0 | I/O Write |
| 1 | 0 | 1 | Memory read |
| 1 | 1 | 0 | Memory write. |

**4. What are the functional parts of 8086 CPU? R**

The two independent functional parts of the 8086 CPU are:

i. Bus Interface Unit (BIU) : BIU sends out addresses, fetches instruction from memory, reads data from ports and memory and writes data to ports and memory.

ii. Execution Unit (EU):EU tells the BIU where to fetch instructions or data, decodes instructions and executes instructions.

**5. What is the operation of S0, S1 and S2 pins in maximum mode? R**

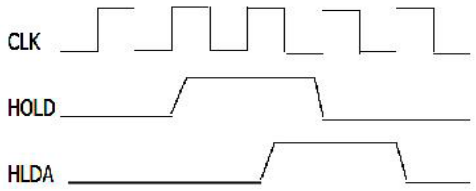
S2, S1, S0 indicates the type of transfer to take place during the current bus cycle.

| $S_2$ | $S_1$ | $S_0$ | | |
|---|---|---|---|---|
| 0 | 0 | 0 | - | Interrupt acknowledge |
| 0 | 0 | 1 | - | Read I/O port |
| 0 | 1 | 0 | - | Write I/O port |
| 0 | 1 | 1 | - | Halt |
| 1 | 0 | 0 | - | Instruction fetch |
| 1 | 0 | 1 | - | Read Memory |
| 1 | 1 | 0 | - | Write Memory |
| 1 | 1 | 1 | - | Inactive. |

**6. Give any four pin definitions for maximum mode. C**

| Symbol | Description |
|---|---|
| QS1, QS0 | Reflects the status of the instruction queue. This status indicates the activity in the queue during the previous clock cycle. |
| $\overline{\text{LOCK}}$ | Indicates that the bus is not to be relinquished to other potential bus masters. |
| $\overline{\text{RQ/GT1}}$ | For inputting bus requests and outputting bus grants. |
| $\overline{\text{RQ/GT0}}$ | Same as $\overline{\text{RQ/GT1}}$ except that a request on $\overline{\text{RQ/GT0}}$ has higher priority. |

**7. Draw the bus request and bus grant timings in minimum mode system. A**




Bus request and bus grant timings in minimum

**8. What is the purpose of a decoder in EU? R**

The decoder in EU translates instructions fetched from memory into a series of actions, which the EU carries out.

**9. Give the register classification of 8086.(Nov/Dec2012) C**

The 8086 contains:

i. General purpose registers: They are used for holding data, variables and intermediate results temporarily.

ii. Special purpose registers: They are used as segment registers, pointers, index register or as offset storage registers for particular addressing modes.

**10. What are general data registers? R**

The registers AX, BX, CX and DX are the general data registers.

| | | |
|---|---|---|
| AX | AH | AL |
| BX | BH | BL |
| CX | CH | CL |
| DX | DH | DL |

- L and H represents the lower and higher bytes of particular register.
- AX register is used as 16-bit accumulator.
- BX register is used as offset storage for forming physical addresses in case of certain

- addressing modes.
- CX register is used as a default counter in case of string and loop instructions.
- DX register is used as an implicit operand or destination in case of a few instructions.

**11. Give the different segment registers.(April/May2012) C**

The four segment registers are:

i. Code segment register: It is used for addressing a memory location in the code segment ofthe memory, where the executable program is stored.

ii. Data segment register: It points to the data segment of the memory, where data is resided.

iii. Extra segment register: It also contains data.

iv. Stack segment register: It is used for addressing stock segment of memory. It is used tostore stack data.

**12. What are pointers and index registers? R**

IP, BP and SP are the pointers and contain offsets within the code, data and stack segments respectively. SI and DI are the index registers, which are used as general purpose registers and also for offset storage in case of indexed, based indexed and relative based indexed addressing modes.

**13. How is the physical address calculated? Give an example**. R

The physical address, which is 20-bits long is calculated using the segment and offset registers, each 16-bits long. The segment address is shifted left bit-wise four times and offset address is added to this to produce a 20 bit physical address.

```
Eg:   segment address - > 1005H
      Offset address  - >  5555H
      Segment address  - > 1005H  - >   0001 0000 0000 0101
      Shifted by 4 bit position  - > 0001 0000 0000 0101 0000
      Offset address  - >      +            0101 0101 0101 0101

      Physical address  - >          0001 0101 0101 1010 0101
                                       1    5    5    A    5
```

**14. What is meant by memory segmentation? R**

Memory segmentation is the process of completely dividing the physically available memory into a number of logical segments. Each segment is 64K byte in size and is addressed by one of the segment register.

**15. What are the advantages of segmented memory? R**

The advantages of segmented memory are:

i. Allows the memory capacity to be 1Mbyte, although the actual addresses to be handled are of 16-bit size.

ii. Allows the placing of code, data and stack portions of the same program in different parts of memory for data and code protection.

iii. Permits a program and/or its data to be put into different areas of memory, each times program is executed i.e., provision for relocation may be done.

**16. What is pipelining? R**

Fetching the next instruction while the current instruction executes is called pipelining.

**17. What are the two parts of a flag register? R**

The two parts of the 16 bit flag register are:

i. Condition code or status flag register: It consists of six flags to indicate some condition produced by an instruction.

ii. Machine control flag register: It consists of three flags and are used to control certain operations of the processor

**18. Draw the format of 8086 flag register.(April/May2011) (April/May2019)A**

8086 flag register:

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| U | U | U | U | OF | DF | IF | TF | SF | ZF | U | AF | U | PF | U | CF |

U-Undefined

CF - Carry flag
PF - Parity flag
AF - Auxiliary flag
ZF - Zero flag
SF - Sign flag
TF - Single step trap flag
DF - Direction flag
IF - Interrupt enable flag
OF - Overflow flag

**19. Explain the three machine control flags. U**

i. **Trap flag:** If this flag is set, the processor enters the single step execution.

ii**. Interrupt flag:** If this flag is set, the maskable interrupts are recognized by the CPU, otherwise they are ignored.

iii. **Direction flag:** This is used by string manipulation instructions. If this flag bit is„0", the string is processed from the lowest to the highest address i.e., auto incrementing mode. Otherwise, the string is processed from highest address to lowest address, i.e., auto decrementing mode.

**20. What are the three groups of signals in 8086?(Nov/Dec2009) R**

The 8086 signals are categorized in three groups.

They are:

i. The signals having common functions in minimum and maximum mode.

ii. The signals having special functions for minimum mode.

iii. The signals having special functions for maximum mode.

**21. What are the uses of AD15 – AD0 lines? R**

AD15 – AD0 are time multiplexed memory I/O address and data lines. Address remains on the lines during T1 state, while data is available on data bus during T2, T3, Tw and T4 states. These lines are active high and float to a tristate during interrupt acknowledge and local bus hold acknowledge cycles.

**22. What is the operation of RD signal? R**

RD is an active low signal. When it is low, it indicates the peripherals that the processor is performing a memory or I/O read operation.

**23.Give the function of i. Ready and ii. INTR signal.(May/Jun 2013) C**

i. Ready signal: It is an acknowledgement from slow devices of memory that they have completed data transfer. The signal is synchronized by 8284 A clock generator to give ready input to 8086. The signal is active high.

ii. INTR signal: It is a level triggered input. This is sampled during the last cycle of each instruction to determine the availability of the request. If any interrupt request is pending, the processor enters the interrupt acknowledge cycle. This can be internally masked by resetting the interrupt enable flag. The signal is active high and internally synchronized.

**24. What is the operation performed when TEST input is low? R**

When the TEST input is low, execution will continue, else, the processor remains in an idle state.

**25. What is NMI (Non-Maskable Interrupt)? R**

NMI is an edge-triggered input, which causes a type 2 interrupt. It is not maskable internally by software and transition from low to high initiate the interrupt response at the end of the current instruction. This input is internally synchronized.

**26. What is the purpose of clock input? R**

The clock input provides the basic timing for processor operation and bus control activity. It is an asymmetric square wave with 33% duty cycle. The range of frequency varies from5MHz to 10MHz.

**27. What is the function of $MN/\overline{MX}$ pin?[april/may2011] [April/May 2021] R**

The logic level at $MN/\overline{MX}$ pin decides whether processor operates in minimum or maximum mode.

$MN/\overline{MX}$ =0 Maximum Mode

$MN/\overline{MX}$ =1 Minimum Mode

**28. What happens when a high is applied to RESET pin? R**

When a high is given to RESET pin, the processor terminates the current activity and starts executing from FFFF0H. It must be active for at least four clock cycles. It is internally synchronized.

**29. What will happen when a DMA request is made, while the CPU is performing a memory or I/O cycles? [Nov/dec2011] R**

When a DMA request is made, while the CPU is performing a memory or I/O cycles, it will request the local bus during T4 provided:

i. The request occurs on or before T2 state of the current cycle.

ii. The current cycle is not operating over the lower byte of a word.

iii. The current cycle is not the first acknowledge of an interrupt acknowledge sequence.

iv. A lock instruction is not being executed.

**30. What is multiprogramming?[April/May 2018][APR/MAY 2017][Nov/Dec 2015] R**

If more than one process is carried out at the same time, then it is known as multiprogramming. Another definition is the interleaving of CPU and I/O operations among several programs is called multiprogramming. To improve the utilization of CPU and I/O devices, we are designing to process a set of independent programs concurrently by a single CPU. This technique is known as multiprogramming.

**31. Write the advantages of loosely coupled system over tightly coupled systems? C**

1. More number of CPUs can be added in a loosely coupled system to improve the system performance
2. The system structure is modular and hence easy to maintain and troubleshoot.
3. A fault in a single module does not lead to a complete system breakdown.

**32. What is the different clock frequencies used in 80286? R**

Various versions of 80286 are available that run on 12.5MHz, 10MHz and 8MHz clock frequencies.

**33. Define swapping in? R**

The portion of a program is required for execution by the CPU, it is fetched from the secondary memory and placed in the physical memory. This is called 'swapping in' of the program.

**34. What are the different operating modes used in 80286? R**

The 80286 works in two operating modes

1. Real addressing mode
2. Protected virtual address mode.

**35. What are the CPU contents used in 80286? R**

The 80286 CPU contains almost the same set of registers, as in 8086

- Eight 16-bit general purpose register
- Four 16-bit segment registers
- Status and control register
- Instruction pointer.

**36. What are the signals used in 8086 maximum mode operation? R**

Qs1, Qs0, s0, s1, s2, LOCK, RQ/GT1, RQ/GT0 are the signals used in 8086 maximum mode operation.

**37. Write the size of physical memory and virtual memory of 8086 microprocessor. C**

Physical addresses are formed when the left shifted segment base address is added to the offset address. The combination of segment register base addresses and offset address is the logical address in memory.

Size of physical memory=220=1MB Size of virtual memory=216=64 KB

**38. List the advantages of using segment registers in 8086. C**

- It allows the memory addressing capacity to be 1MB even though the address associated with individual instruction is only 16-bit.
- It facilitates use of separate memory areas for program , data and stack.
- It allows the program to be relocated which is very useful in multiprogramming.

**39. Explain the BHE and LOCK signals of 8086. U**

- *BHE (Bus High Enable):* Low on this pin during first part of the machine cycle indicates that at least one byte of the current transfer is to be made on higher byte AD15-AD8.
- *LOCK:* This signal indicates that an instruction with a LOCK prefix is being executed and the bus is not to be used by another processor.

**40. What are the two modes of operations present in 8086? [may/june2007] [April/May 2021] R**

i. Minimum mode (or) Uniprocessor system

ii. Maximum mode (or) Multiprocessor system

**41. What are the functions of status pins in 8086? R**

S2 S1 S0
0 0 0 ---- Interrupt acknowledge
0 0 1 ---- Read I/O
0 1 0 ---- Write I/O
0 1 1 ---- Halt
1 0 0 ---- Code access
1 0 1 ---- Read memory
1 1 0 ---- Write memory
1 1 1 ---- inactive

S4 S3
0 0 --I/O from extra segment
0 1 --I/O from Stack Segment
1 0 --I/O from Code segment
1 1 --I/O from Data segment
S5 --Status of interrupt enable flag
S6 --Hold acknowledge for system bus
S7 --Address transfer.

**42. What are the three classifications of 8086 interrupts?[ MAY/JUNE-2006] R**

(1) Predefined interrupts,

(2) User defined Hardware interrupts,

(3) User defined software interrupts.

**43. What are the differences between maximum mode and minimum mode [NOV/DEC2003 ]R**
**Minimum mode R (APR/MAY 2021)**

1 A processor is in minimum mode when MN /MX pin is strapped to +5v

2. All control signals are given out by microprocessor chip it self

3. There is a single micro processor

**Maximum mode**

1. A processor is in maximum mode when MN /MX is grounded

2. The processor derive the status signals S2, Sl and So. Another chip called bus controller derives control signals using this status information.

3. There may be more than one microprocessor

**44. What is Coprocessor? [NOV/DEC 2007] [APR/MAY2011] R**

The coprocessor is a processor which specially designed for processor to work under the control of the processor and support special processing capabilities. Example : 8087 which has numeric processing capability and works under 8086.

**45. What are the basic multiprocessor configurations? R**

1. Closely Coupled configuration

2. Loosely coupled configuration

**46. Differentiate External verses Internal Bus. A/E [MAY/JUNE 2016]**

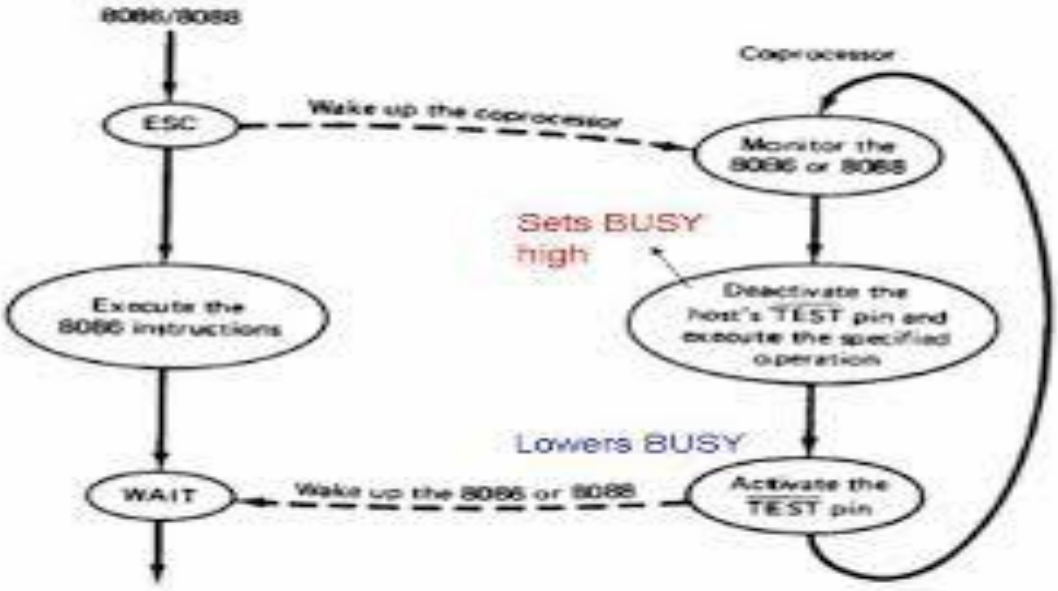
**Internal Data Bus:** The internal data bus only works inside a CPU that is internally. It is able to communicate with the internal cache memories of the CPU. Since they are internally placed they are relatively quick and are now affected by the rest of the computer.

**External Data bus:** This type of bus is used to connect and interface the computer to its connected peripheral devices. Since they are external and do not lie within the circuitry of the cpu they are relatively slower. The 8088 processor in itself contains a 16-bit internal data bus coupled with a 20-bit address register. This allows the processor to address to a maximum of 1 MB memory.

**46. Compare closely coupled and loosely coupled configurations.**
**[April/May 2019] [NOV/DEC 2011][May/June 2016] A/E**

| Closely coupled | Loosely coupled |
|---|---|
| 1. Single CPU is used | 1. Multiple CPU modules are used |
| 2. It has local bus only | 2. It has local as well system bus |
| 3. No system memory or IO | 3. It has system memory and IO, shared |
| 4. Data rate is High. | 4. Data rate is low. |
| 5. It experiences more conflicts. | 5. It does not encounter memory conflict. |
| 6. More expensive. | 6. Low expensive. |
| 7. Efficient for high-speed or real-time processing. | 7. Efficient when tasks running on different processors has minimal interaction. |
| 8.No bus arbitration logic required. | 8. Bus arbitration logic required among the CPU modes. |

**47. Schematically show,how synchronization is made between 8086 and its coprocessor. [NOV/DEC 2015] A/E**




48. **Write some example for advanced processors**. R **(NOV/DEC 2017)**

Pentium processors, 80486, 80386, 80286

49. **What is the need of LOCK signal.R[Nov/Dec 2017]**

This signal indicates that an instruction with a LOCK prefix is being executed and the bus is not to be used by another processor.

**50. Define system bus.R[Nov/Dec 2018]**

The system bus works by combining the functions of the three main buses: namely, the data, address and control buses. Each of the three buses has its separate characteristics and responsibilities.

51. **When is co-processor used?R[Nov/Dec 2018]**

A coprocessor is a computer processor used to supplement the functions of the primary processor (the CPU). Operations performed by the coprocessor may be floating point arithmetic, graphics, signal processing, string processing, cryptography or I/O interfacing with peripheral devices. By offloading processor-intensive tasks from the main processor, coprocessors can accelerate system performance. Coprocessors allow a line of computers to be customized, so that customers who do not need the extra performance do not need to pay for it.

**52. Write the advantages of loosely coupled system over tightly coupled systems. [APR/MAY 2019]**

1. More number of CPUs can be added in a loosely coupled system to improve the system performance
2. The system structure is modular and hence easy to maintain and troubleshoot.
3. A fault in a single module does not lead to a complete system breakdown.

**53. State the function of ALE signal in 8086. [NOV/DEC 2019].**

To demultiplex address and data lines using external latch.

**54. Draw the timing diagram for an interrupt acknowledgement on an 8086system. (Nov/Dec 2021) R**

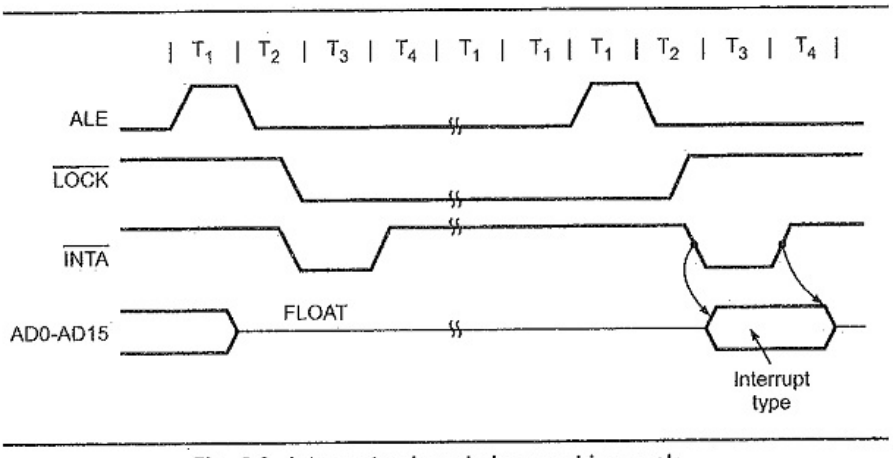



Fig. 9.3 Interrupt acknowledge machine cycle

**55. List the features of multi core processors. (Nov/Dec 2021) A**

1. L2 and L3 cache is shared in all types of multi core processors in order to improve performance.
2. Each core in a multi core processor has its own independent connection to reserved

RAM.

3. Applications such as gaming and video editing benefit from having multi core processors available.
4. Multiple single-core processors use less power and generate less heat than a multicore processor does.
5. Multicore processors can execute instructions more quickly than single-core processors can.

## PART-B

1. (a) Draw and explain the maximum mode of 8086 (12)U **(April/May 2012)**
   (b) List the advantages of multiprocessor system (4) C
2. (i) Show the pin configuration and function of signals of 8086 microprocessor. (8) A/E **(April/May2011)**
   (ii) Show the memory organization and interfacing with 8086 microprocessor. Explain how the memory is accessed.(8)A/E **(April/May 2011)**
3. (a) Explain the functions of (8) U **(Nov/Dec2012)**
   i. HLDA
   ii. RQ/GT0
   iii. DEN
   iv. ALE
   (b) Draw and explain the minimum mode of 8086 (8**)**U**(Nov/Dec2014) [Nov/Dec2015]**

4. (a) Draw and explain the block diagram of minimum mode of operation. (12)U **(NOV/Dec2011)**
   (b)Write notes on addressing memory (4) C **(May/June2014)**
5. Define the bus cycle and minimum mode read and write bus cycles with propertiming diagram (16) R **(April/May2013)**
6. (a) Draw the input and output timing diagram of maximum mode of operation in8086 (10) A/E
   (b) Explain the addressing capabilities of 8086 (6) U **(Nov/Dec2013)**
7. Discuss the maximum mode configuration of 8086 with a neat diagram. Mention the functions of the various signals. (16) U **(Apr/May 2015)**
8. Compare closely coupled configuration with loosely coupled configuration (8**) A/E** **(Apr/May 2015)**
9. Write a assembly language program to check whether the given string is palindrome or not. (8)C **(Apr/May 2015)**
10. Explain the bus interface unit and execution unit of 8086 microprocessor. (8) U **(Nov/Dec 2014)**
11. Describe the sequence of signals that occurs on the address bus, the control bus and the data bus when a simple microcomputer fetches an instruction. (8) R **(Nov/Dec 2014)**
12. Write an assembly language program to multiply two 16 bit numbers to give 32 bit result. (8) C **(Nov/Dec 2014)**
13. Describe the conditions which cause the 8086 to perform type 0 and type 1 interrupt. (8) R **(Nov/Dec 2014)**
14. Define loosely coupled system. Explain the schemes used for establishing priority. (16) U **[NOV/DEC 2015]**
15. Explain in detail about the system bus timing of 8086.(16) U **[May/June 2016]**
16. Explain the following: U **[May/June 2016]**
    (i) Multiprocessor system (4)

(ii) Coprocessor (4)
(iii) Multiprogramming (4)
(iv) Semaphore (4)

17. Explain the system bus structure of 8086. Draw the timing diagram for interrupt acknowledgement cycle. (13) U **[Nov/Dec 2017][April/May 2017]**

18. Explain the closely looped configuration with neat diagram. (13) U **[April/May 2017]**

19. Explain the loosely looped configuration with neat diagram. (13) U **[Nov/Dec 2017]**

20. Discuss the maximum mode configuration of 8086 with a neat diagram. Mention the functions of various signals. (13) U **[April/May 2018]**

21. Discuss about the multiprocessor configurations of 8086. (13) U **[April/May 2018]**

22. Distinguish between closely coupled and loosely coupled multiprocessor configurations. (13) A/E**[Nov/Dec 2018]**

23. What do you understand from system bus structure? Explain. (13) U **[Nov/Dec 2018]**

24. With neat block diagram, explain the architecture of 8086 in maximum mode configuration. Also explain the Bus timing diagram for input and output transfer on a maximum mode. (13) U**[April/May 2019]**

25. Explain the interrupt system based on multiple 8259 with necessary block diagram. (13) U **[April/May 2019]**

26. Draw the pin diagram of 8086 processor and explain all the signals. R **[April/May 2021]**

27. i) Explain in detail about closely coupled configurations. **U [April/May 2021]**
ii) Discuss on loosely coupled configurations in detail R

**28.** Examine the effectiveness of the minimum mode and maximum mode of operations in 8086 in detail. **U [April/May 2021]**

**29.** Differentiate between minimum and maximum mode of operation of 8086 microprocessor, and sketch the maximum mode configuration. **(Nov/Dec 2021) A**

**30.** Differentiate closely coupled and loosely coupled configurations of 8086 based multiprocessing systems, and illustrate the general concepts underlying the schemes for establishing priority in a loosely coupled multiprocessor system. **(Nov/Dec 2021)A**

## ASSIGNMENT QUESTIONS

1. Write a 8086 assembly language program to check whether the given string Palindrome or not. **(Create)**
2. Explain in detail about closely coupled and loosely coupled configuration. What are the relative advantages and disadvantages? **(Understand)**

## UNIT III

## I/O INTERFACING

## PART-A

**1. What is memory mapped I/O?(Nov/Dec 2014) R**

This is one of the techniques for interfacing I/O devices with μp. In memory mapped I/O, the I/O devices assigned and identified by 16-bit addresses. To transfer the data between MPU and I/O devices memory related instructions (such as LDA, STA etc.) and memory control signals (MEMR, MEMW) are used.

**2. What is I/O mapped I/O?(April/May 2013) R**

This is one of the techniques for interfacing I/O devices with μp. In I/O mapped I/O, the I/O devices assigned and identified by 8-bit addresses. To transfer the data between MPU and I/O devices I/O related instructions (IN and OUT ) and I/O control signals (IOR, IOW) are used.

**3. What is simplex and duplex transmission? R**

Simplex transmission: data are transmitted in only one direction. Duplex transmission: data flow in both directions. If the transmission goes one way at a time, it is called half duplex; if it goes both way simultaneously, then it is called full duplex.

**4. Define Baud.(EE2354May/June2012) R**

The rate at which the bits are transmitted, bits per second is called Baud.

**5. What are the signals available for serial communication? R**

SID – serial input data

SOD – serial output data

**6. What is USART? R**

It is a programmable device. Its function and specification for serial I/O can be determined by writing instructions in its internal registers. The Intel 8251A USART is a device widely used in serial I/O.

**7. Write the features of 8255A.(Nov/Dec 2013) C**

The 8255A has 24 I/O pins that can be primarily grouped primarily in two 8-bit Parallel ports: A and B, with eight bits as port C. The 8-bits of port C can be used as two 4-bit ports: C UPPER CU and CLOWER CL.

**8. What is BSR mode? R**

All functions of 8255 are classified according to 2 modes. In the control word, if D7 = 0, then it represents bit set reset mode operation. The BSR mode is used to set or reset the bits in port C.

**9. What is mode 0 operation of 8255?[April/May 2019] (Nov/Dec2011) R**

In this mode, ports A and B are used as two simple 8-bit I/O ports and port C as two 4-bit ports. Each port can be programmed to function as an input port or an output port. The input/ output features in mode 0 as follows:

i. outputs are latched

ii. inputs are not latched

iii. ports do not have handshake or interrupt capability.

**10. What are the modes of operation supported by 8255? R**

i. Bit set reset mode (BSR)

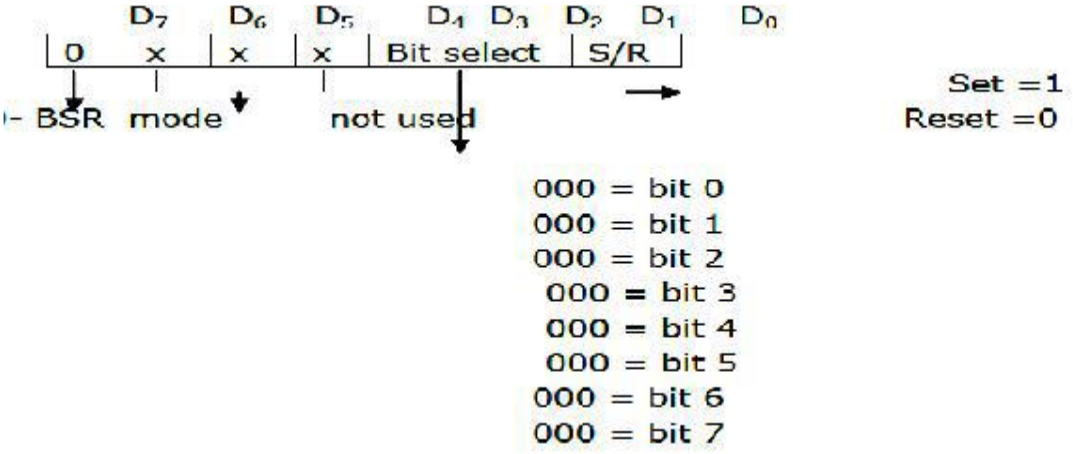
ii. I/O mode

Mode 0

Mode1

Mode2

**11. Write the control word format for BSR mode. C**




**12. What is ADC and DAC? R**

The electronic circuit that translates an analog signal into a digital signal is called analog-to-digital converter(ADC).

The electronic circuit translates a digital signal into an analog signal is called Digital-to-analog Converter(DAC).

**13. Define conversion time. R**

It is defined as the total time required to convert an analog signal into a digital output. It is determined the conversion technique used and by the propagation delay in various circuits.

**14. What are the functions to be performed by µp while interfacing an ADC? R**

i. Send a pulse to the START pin.

ii. Wait until the end of conversion

iii. Read the digital signal at an input port

**15. Write the different types of ADC. C**

i. Single slope ADC

ii. Dual slope ADC

iii. Successive approximation ADC

iv. Parallel comparator type ADC

v. Counter type ADC

**16. What is resolution time in ADC? R**

It is defined as a ratio of change in value of input voltage Vi, needed to change the digital output by 1 LSB. If the full scale input voltage required to cause a digital output of all 1‟s is ViFS. Then the resolution can be given as

Resolution = ViFS / (2n-1)

**17. List the functions performed by 8279.(april/may2009**) C

i. It has built-in hardware to provide key debounce.
ii. It provides a scanned interface to a 64 contact key matrix.
iii. It provides multiplexed display interface with blanking and inhibit options.
iv. It provides three input modes for keyboard interface.

**18. What is key debounce?(Nov/Dec2014) R**

The push button keys when pressed, bounces a few times, closing and opening the contacts before providing a steady reading. So reading taken during bouncing may be faulty. Therefore the microprocessor must wait until the key reach to steady state. This is known as key debounce.

**19. What are the operating modes in 8279?[April/May 2019] [Nov/dec2013] R**

i. Scanned keyboard mode
ii. Scanned sensor matrix
iii. Strobed input

**20. What is N-key rollover? [Nov/Dec2013], [April/may2012] R**

In N-key rollover each key depression is treated independently from all others. When a key is depressed, the denounce logic is set and 8279 checks for key depress during next two scans.

**21. Find the program clock command word if external clock frequency is 2 MHz. U**

Prescalar value = (2 x106) / (100 x 103) = (10100)2
Therefore command word = (00110100)2

**22. What is multiple interrupt processing capability? R**

Whenever a number of devices interrupt a CPU at a time, and id the processor is able to handle them properly, it is said to have multiple interrupt processing capability.

**23. What is hardware interrupt? R**

An 8086 interrupt can come from any one of three sources. One source is an external signal applied to the nonmaskable interrupt (NMI) input in or to the interrupt (INTR) input pin. An interrupt caused by the signal applied to one of these input is referred to as a hardware interrupt

**24. What is software interrupt? R**

The interrupt caused due to execution of interrupt instruction is called software interrupt.

**25. What are the two types of interrupts in 8086? R**

The two types of interrupts are:
i. **External interrupts:** In this, the interrupt is generated outside the processor.
Example: Keyboard interrupt.
ii. **Internal interrupts:** It is generated internally by the processor circuit or by the execution of an interrupt instruction. Example: Zero interrupt, overflow interrupt.

**26. What is the purpose of control word written to control register in8255? [APRIL/MAY2011] R**

The control words written to control register specify an I/O function for each I.O port. The bit D7 of the control word determines either the I/O function of the BSR function.

**27. What is memory mapping?[NOV/DEC 2007] R**

The assignment of memory addresses to various registers in a memory chip is called as memory mapping.

**28. What are the modes of operations used in 8254?[April/May 2018] (Apr/May 2015) R**

Each of the three counters of 8254 can be operated in one of the following six modes of operation.

1. Mode 0 (Interrupt on terminal count)
2. Mode 1 (Programmable mono shot)
3. Mode 2 (Rate generator)
4. Mode 3 (Square wave generator)
5. Mode 4 (Software triggered strobe)
6. Mode 5 (Hardware triggered strobe)

**29. List the operating modes of 8255A and 8237A. [NOV/DEC 2015] C**

8255 has 2 modes.

1. I/O mode-Multiprocessor
   - Mode 0
   - Mode 1
   - Mode 2
2. Bit Set-Reset mode (BSR)

8237 has several modes. They are,

- Single mode
- Burst mode
- Block mode
- Demand mode
- Cascade mode

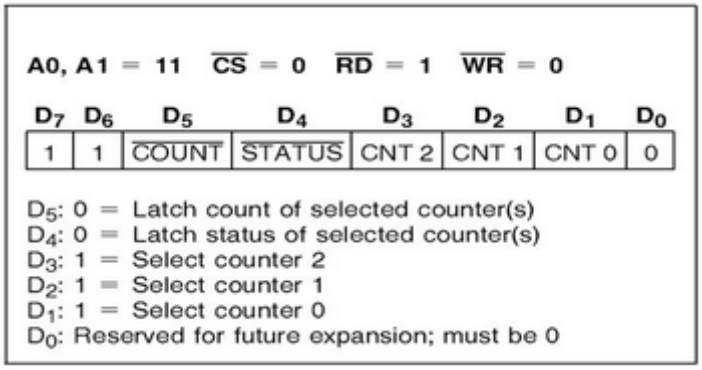
**30. What freq. transmit clock (Txc') is required by an 8251 in order for it to transmit data at 4800 baud with a baud rate factor of 16. U [NOV/DEC 2015]**

T=209us

**31. What is keydebouncing? R [May/June 2016]**

When the key is depressed and released, the contact is not brokenpermanently. In fact, the key makes and breaks the contacts several times for afew milliseconds before the contact is broken permanently.

32. **Draw the format of read back command register of 8254. R (APR/MAY 2017)**




33. **Write a 16 bit delay program of 8086. C [April/May 2018][APR/MAY 2017]**

```
        MOV BX, FFFFH
Back:   DEC BX
        JNZ back
        RET
```

**34. What are the handshaking signals used in Mode – 2 configuration of 8255? R [NOV/DEC 2017]**

$PC_3$-$PC_7$ signals are used for handshaking purpose in mode – 2.

**35. How the DMA operation performed with 8086? A/E [NOV/DEC 2017]**

In this technique external device is used to control data transfer. External device generates address and control signals required to control data transfer and allows peripheral device to directly access the memory. Hence this technique is referred to as Direct Memory Access (DMA) and external device which controls the data transfer is referred to as **DMA controller.**

**36. Why is memory interfacing required? R [NOV/DEC 2018]**

**Memory requires** some signals to read from and write to registers. Similarly the microprocessor transmits some signals for reading or writing a data. In simple words, the primary function of a **memory interfacing** circuit is to aid the microprocessor in reading and writing a data to the given register of a **memory** chip.

**37. What are the differences between LED display and LCD display? A/E [NOV/DEC 2018]**

**30. <u>LED Display:</u>**

- LEDs use light emitting diodes.
- The light emitting diodes can be placed either behind the screen or around its edges.
- The difference in lights and in lighting placement has generally meant that LED display can be thinner than LCDs.
- LED Display run with greater energy efficiency and can provide a clearer, better picture than the general LCD Displays.
- The brightness level is very good for LEDs
- LEDs Consume more power than LCDs.

- LEDs have wide viewing angle. The viewing angle is 150 degree.

**LCD Display:**

- LCDs use fluorescent lights.
- The fluorescent lights in an LCD display are always behind the screen.
- LCDs take advantage of a phenomena known as polarization. Polarization is the direction in which the light wave is oscillating, or swinging back and forth at the same speed. Light comes out of the backlight unpolarized. It then passes through one polarizer, which makes all the light oscillate the same way.
- LCDs have moderate brightness level.
- LCD Consumes very less power.
- The viewing angle for LCD is 100 degree.

**31. What is mode 0 operation of 8255? (APR/MAY 2019) R**

In this mode, ports A and B are used as two simple 8-bit I/O ports and port C as two 4-bit ports. Each port can be programmed to function as an input port or an output port. The input/ output features in mode 0 as follows:

i. outputs are latched
ii. inputs are not latched
iii. ports do not have handshake or interrupt capability.

**32. What are the signals available for serial communication (APR/MAY 2021) R**

The four basic SPI signals (MISO, MOSI, SCK and SS), Vcc and Ground are the part of data communication. So it needs 6 wires to send and receive data from slave or master. Theoretically, the SPI can have unlimited number of slaves. The data communication is configured in SPI registers.

**33.** Define conversion time. **(APR/MAY 2021) R**

The task of the analog-to-digital converter (ADC) is the inverse of the digital-to-analog converter: to convert an analog input signal into a numerical digital value.

**34. State the important functions of an I/O interface. (Nov/Dec 2021) R**

**35. Calculate the resolution of an 8-bit A/D converter assuming the voltage range of the input as –10V to +10V. (Nov/Dec 2021) E**

Vrange= Vmax-Vmin
Vmax = 10V, Vmin = -10V
Vrange = 10 =- (-10) = 20
n=8
Resolution = R= Vrange/ $2^n$ = $20/2^8$ = 20/256 =0.078125

**PART B**

1. Draw the block diagram of 8279 and explain the function of each. (16) U **(Nov/Dec 2014)(Nov/Dec2010)**
2. With the help of neat diagram explain how 8251 is interfaced with 8086 and used for serial Communication. (16) U **(May/June2013)**
3. Discuss the silent feature of 8259 and explain the block diagram of 8259- programmable interrupts controllers (16) U **(April/May 2013&2012)**

4. (a) Describe the various modes of operation in 8253 programmable internal timer. (8) R **(Nov/Dec2010)(Nov/Dec 2014)**

(b) Explain the operation of DMA controller 8237 (8) U **(EE2354 May/June 2014)**

5. (a)Draw and explain the interfacing of cascaded 8259 with 8086. (10) U **(Nov/Dec2013)**

(b) Explain in detail with the modes of operation of 8255 (6) U

6. Discuss the various operating modes of 8253 timer with necessary control words (16) U

7. (i) Explain the operation of 8255 PPI Port A programmed as input and output in Mode 1 with necessary handshaking signals. **(8) U** **(April/May2011)**

(ii) Show and explain the ADC interfacing with 8086 microprocessor. (8). U **(April/May2011)**

8. With functional block diagram, explain the operation and programming of 8251 USART (Serial communication Interface) in detail. (16) U **(April/May2011)[NOV/DEC 2015]**

9. Explain how D/A and D/A interfacing is done with 8086 with an application**.(10) U** **(Apr/May 2015)**

10. What is DMA? Explain the DMA based data transfer using DMA controller. **(6) U** **(Apr/May 2015)**

11. Draw the block diagram of traffic light control system using 8086.**(8) A/E(Apr/May 2015)**

12. Write the algorithm and assembly language program for traffic light control system. **(8) C** **(Apr/May 2015)**

13.Draw the block diagram of programmable Interrupt controller (8259) and explain its operations**. U** **[NOV/DEC 2015]**

14. Explain in detail about DMA controller with its diagram. (16) U

**[April/May 2018][May/June 2016]**

16. Draw and Explain the functional diagram of parallel communication interfacing chip. (13) U **[April/May 2017]**

17. Explain the need of DMA controller with its functional diagram. (13) U **[April/May 2017]**

18. Develop a 8086 based system to display the word HELLO for every 2ms in the common cathode seven segment LED display and check how many times the word display for one hour. (15) C **[April/May 2017]**

19. Draw and explain the functional diagram of 8251. (13) U **[Nov/Dec 2017]**

20. Draw and explain the functional diagram of keyboard and display controller.(13)U **[Nov/Dec 2017]**

21. Develop a 8086 based system with 128 RAM and 4K ROM, to display the word HAPPY for every 2ms in the common anode seven segment LED display. Explain the delay timings. (15) C

**[Nov/Dec 2017]**

22. Draw the block diagram and explain the operation of USART. (13) U **[April/May 2018]**

23. Draw the block diagram of traffic light control system using 8086. Write the algorithm and ALP for traffic light control system. (15) U **[April/May 2018]**

24. How are D/A and A/D Interfaces used? Explain. (13) A/E **[Nov/Dec 2018]**

25. What are Interrupt controller and DMA controller? Explain. (13) U **[Nov/Dec 2018]**

26. Draw the Block diagram and explain the operations of 8251 serial communication interface. (13) U **[April/May 2019]**

27. Explain in detail about interfacing of four LCD digits to 8086. (13) U **[April/May 2019]**

28. Draw the block diagram of 8279 and explain the function of each. R **[April/May 2021]**

29. Explain the operation of DMA controller 8237 with neat diagrams. U **[April/May 2021]**

30. Describe the operating modes and control words of Programmable Peripheral Interface (8255). Also specify the handshaking signals and their functions if port A of 8255 is setup as input port in mode 1. **(Nov/Dec 2021) R**

31. Discuss in detail the working of an independent DMA controller and its data transfer modes. Also, show the general organization of a one channel DMA controller and its principal connections. **(Nov/Dec 2021) R**

32. Use a keyboard / display controller (8279) to interface a 64-key keyboard and an eight digit seven-segment display. Also discuss the working principle of 8279 with appropriate commands needed to be given before the CPU sends the characters to be displayed to the 8279 and before inputting data from the FIFO memory. **(Nov/Dec 2021) A**

## ASSIGNMENT QUESTIONS

1. What is Control Word? Determine the control word for the following configuration of 8255:- **(Create)**

Port A – Output
Mode of port A – Mode 1
Port B – Output
Mode of port B – Mode 0
Port C lower (pins PC0 – PC2) – Output

2. Explain why serial data transfer is mostly preferred over parallel data transfer. Give reasons? Discuss the types of serial communication? **(Understand)**

3. Design a hardware interfacing circuit for interfacing 8251 with 8086, set the 8251A in synchronous mode as a transmitter and receiver with even parity enable, 2 stop bits, 8-bit character length, frequency 160KHz and baud rate 10K. Write an ALP to transmit 100 bytes of data string starting at location 2000:5000H. **(Create)**

4. Give the description about the following**. (Understand)**

a. RS – 232
b. IEEE-488 GPIB
c. Prototyping and troubleshooting

## UNIT-IV

## MICROCONTROLLER

### Part-A

**1. What are the special function register?(EE2354 April/May2012) R**

The special function register are stack pointer, index pointer (DPL and DPH), I/O port addresses, status(PSW) and accumulator.

**2. What are the uses of accumulator register? R**

The accumulator registers (A and B at addresses OEOh and OFOh, respectively) are used to store temporary values and the results of arithmetic operations.

**3. What is PSW?(EE2354 Nov/Dec2011) R**

Program status word (PSW) is the set of flags that contains the status information and is considered as one of the special function register.

**4. What is stack pointer (sp)? (EE2354 April/May2011) R**

Stack pointer (SP) is a 8 bit wide register and is incremented before the data is stored into the stack using PUSH or CALL instructions. It contains 8-bit stack top address. It is defined anywhere in the on-chip 128-byte RAM. After reset, the SP register is initialized to 07. After each write to stack operation, the 8-bit contents of the operand are stored onto the stack, after incrementing the SP register by one. It is not a top-down data structure. It is allotted an address in the special function register bank.

5. **What is data pointer (DTPR)?(Nov/Dec2010) R**

It is a 16-bit register that contains a higher byte (DPH) and lower byte (DPL) of a 16-bit external data RAM address. It is accessed as a 16-bit register or two 8-bit registers. It has been allotted two addresses in the special function register bank, for its two bytes DPH and DPL.

**6. Why oscillator circuit is used? R**

Oscillator circuit is used to generate the basic timing clock signal for the operation of the circuit using crystal oscillator.

**7. What is the purpose of using instruction register? R**

Instruction register is used for the purpose of decoding the opcode of an instruction to be executed and gives information to the timing and control unit generating necessary signals for the execution of the instruction.

**8. Give the purpose of ALE/PROG signal. (May/June2014) C**

ALE/PROG is an address latch enable output pulse and indicates that valid address bits available on the respective pins. The ALE pulses are emitted at a rate of one-sixth of the oscillator frequency. The signal is valid only for external memory accesses.
It may be used for external timing or clockwise purpose. One ALE pulse is skipped during each access to external data memory.

9**. Explain the two power saving mode of operation.(April/May2011) U**

The two power saving modes of operation are:

- **Idle mode**: In this mode, the oscillator continues to run and the interrupt, serial port and timer blocks are active, but the clock to the CPU is disabled. The CPU status is preserved. This mode can be terminated with a hardware interrupt or hardware reset signal. After this, the CPU resumes program execution from where it left off.

- **Power down mode:** In this mode, the on-chip oscillator is stopped. All the functions of the controller are held maintaining the contents of RAM. The only way to terminate this mode is hardware reset. The reset redefines all the SFRs but the RAM contents are left unchanged.

**10. Differentiate between program memory and data memory. A/E**

**Program Memory**

i. It stores the programs to be executed.

ii. It stores only program code which is to be executed and thus it need not be written, so it is implemented using EPROM It stores the data, line intermediate results, variables and constants required for the execution of the program.

**Data Memory**

The data memory may be read from or written to and thus it is implemented using RAM.

**11. What are addressing modes? R**

The various ways of accessing data are called addressing modes.

**12. Give the addressing modes of 8051? [NOV/DEC 2018] (April/May 2011) C**

There are six addressing modes in 8051.They are

- Direct addressing
- Indirect addressing
- Register instruction
- Register specific (register implicit)
- Immediate mode
- Indexed addressing

**13. What is direct addressing mode? R**

The operands are specified using the 8-bit address field, in the instruction format. Only internal data Ram and SFRS can be directly addressed. This is known as direct addressing mode.

Eg: Mov R0, 89H

**14. What is indirect addressing mode? R**

In this mode, the 8-bit address of an operand is stored in a register and the register, instead of the 8-bitaddress, is specified in the instruction. The registers R0 and R1 of the selected bank of registers or stack pointer can be used as address registers for storing the 8-bit addresses. The address register for 16-bit addresses can only be „data pointer" (DPTR).

Eg: ADD A, @ R0.

**15. What is meant by register instructions addressing mode? R**

The operations are stored in the registers R0 – R7 of the selected register bank. One of these eight registers(R0 – R7) is specified in the instruction using the 3-bit register specification field of the opcode format. A register bank can be selected using the two bank select bits of the PSN. This is called as register instruction addressing mode

Eg: ADD A, R7.

16. **What is immediate addressing mode?(April/May2013) R**

An immediate data ie., a constant is specified in the instruction, after the opcode byte.

Eg: MOV A, #100

The immediate data 100 (decimal) is added to the contents of the accumulator. For specifying a hex number,it should be followed by H. These are known as immediate addressing mode.

17. **What is indexed addressing? [May/June2014] R**

This addressing mode is used only to access the program memory. It is accomplished in 8051 for look-up table manipulations. Program counter or data pointer are the allowed 16-bit address storage registers, in this mode of addressing. These 16-bit registers point to the base of the look-up table and the ACC register contains a code to be converted using the look-up table. The look-up table data address is found out by adding the contents of register ACC with that of the program counter or data pointer. In case of jump instruction, the contents of accumulator are added with one of the specified 16-bit registers to form the jump destination address.

Eg: MOV C, A @ A + DPTP
JMP @ A + DPTR

18. **List the five addressing modes of 8051 microcontroller.(Nov/Dec2010) C**

The five addressing modes are,

I. Immediate addressing
II. Register addressing
III. Direct addressing
IV. Register indirect addressing
V. Indexed addressing.

19. **MOV R4, R7 is invalid. Why? R**

The movement of data between the accumulator and Rn (for n = 0 to 7) is valid. But movement of data between Rn register is not allowed. That is why MOV R4, R7 is invalid.

20. **WHAT IS SFR?(Nov/Dec2014) R**

In the 8051 microcontroller registers A, B, PSW and DPTR are part of the group of registers commonly referred to as special function registers (SFR).

21. **What are the two main features of SFR addresses? R**

The following two points should be noted SFR addresses.

- The special function registers have addresses between 80H and FFH. These addresses are above 80H, since the addresses 00 to 7FH are addresses of RAM memory inside the 8051.
- Not all the address space of 80 to FH is used by the SFR. The unused locations 80Hto FFH are reserved and must not used by the 8051 programmer.

22**. What is the difference between direct and register indirect addressing mode? R**

Loop is most efficient and is possible only in register indirect addressing whereas looping is not direct addressing mode.

**23 List out some compare instructions.(EE2354May/June2014) C**

The compare instructions are:

a. CJNE
b. CLR
c. CPL

24 **Write a program to save the accumulator in r7 of bank 2. C**

```
CLR PSW – 3
SETB PSW – 4
MOV R7, A.
```

25**. What are single bit instructions? Give example. R**

Instructions that are used for single bit operation are called single bit instructions.

Examples: SETB bit
CLR bit
CPL bit

26. **Write a program to save the status of bits p1.2 and p1.3 on ram bit locations 6 and 7 respectively. C**

```
MOV C, P1.2; save status of P1.2 on CY
MOV O6, C; save carry in RAM bit location 06
MOV C, p1.3; save status of p1.3 on CY
MOV 07, C; save carry in RAM bit location 07.
```

27. **Write a program to see if bits 0 and 5 of register b r1. If they are not, make them so and save it in r0.(Nov/Dec2011) C**

```
JNB OFOH, NEXT – 1; JUMP if B.0 is low
SET BOFOH; Make bit B.0 high
NEXT – 1:JNB OF5H, NEXT – 2; JUMP if B.5 is low
SETB OF5H; Make B.5 high
NEXT – 2: MOV R0, B; Save register B.
```

**28. Mention the size of DPTR and Stack Pointer in 8051 microcontroller.(April/May 2011), May/June2014. C**

DPTR and SP are 16 bit register.

**29. What is the operation of the given 8051 microcontroller instructions: XRL A, direct (April/May 2011) R**

XRLA, Direct Exclusive OR operation with A register content and Direct value

**30. List the features of 8051 microcontroller? (May/June2013) C**

The features are

- Single supply +5 volt operation using HMOS technology.
- 4096 bytes program memory on chip(not on 8031)
- 128 data memory on chip.
- Four register banks.
- Two multiple mode, 16-bit timer/counter.
- Extensive Boolean processing capabilities.
- 64 KB external RAM size
- 32 bidirectional individually addressable I/O lines.
- 8 bit CPU optimized for control applications.

**31. Compare Microprocessor and Microcontroller.**
**[NOV/DEC 2018] [NOV/DEC 2006,2011] [ Nov/Dec 2021] A/E**

| Sl.No | Microprocessor | Microcontroller |
|---|---|---|
| 1 | Microprocessor contains ALU, general purpose registers, stack pointer, program counter, clock timing circuit and interrupt circuit. | Microcontroller contains the circuitry of microprocessor and in addition it has built in ROM, RAM, I/O devices, timers and counters. |
| 2 | It has many instructions to move data between memory and CPU. | It has one or two instructions to move data between memory and CPU. |
| 3 | It has one or two bit handling instructions. | It has many bit handling instructions. |
| 4 | Access times for memory and I/O devices are more | Less access time for built-in memory and I/O devices |

| | | |
|---|---|---|
| 5 | Microprocessor based system requires more hardware | Microcontroller based system requires less hardware reducing PCB size and increasing the reliability. |

**32. Name the five interrupt sources of 8051?[MAY/JUNE2007] [APRIL/MAY2008] C**

The interrupts are:
Vector address
• External interrupt 0: IE0: 0003H
• Timer interrupt 0: TF0: 000BH
• External interrupt 1: IE1: 0013H
• Timer Interrupt 1: TF1: 001BH
• Serial Interrupt
Receive interrupt: RI: 0023H
Transmit interrupt: TI: 0023H

**33. List the 8051 instructions that affect the overflow flag. C**

ADD, ADDC, DIV, MUL, SUBB

**34. List the 8051 instructions that always clear the carry flag. C**

CLR C, DIV, MUL

**35. List the 8051 instructions that affect all the flags.[NOV/DEC 2007] C**

ADD, ADDC and SUBB

**36. What are the different types of ADC?[APR/MAY2008 NOV/DEC 2011] R**

The different types of ADC are successive approximation ADC, counter type ADC flash type ADC, integrator converters and voltage to- frequency converters.

**37. What is the necessity of interfacing DAC with microcontroller? (Nov/Dec 2014) R**

In many applications, the microcontroller has to produce analog signals for controlling certain analog devices. Basically, the microcontroller can produce only digital signals. In order to convert the digital signal to analog signal a Digital to Analog Converter has to be employed.

**38. Mention the number of register banks and their addresses in 8051?[NOV/DEC'15] C**

There are 4 register banks. They are Bank0, Bank1, Bank2& Bank3.
RAM locations from 00 to 07H for bank 0
RAM locations from 08 to 0FH for bank 1
RAM locations from 10 to 17H for bank 2
RAM locations from 18 to 1FH for bank 3

**39. What is the jump range? [NOV/DEC'15] R**

AJMP addr11 (Absolute Jump) – Within 2K bytes of program memory.
LJMP addr16 (Long Jump) -Within 64K bytes of program memory.
SJMP Rel.addr (Short Jump) –128 to +127 of program memory.

**40. What are the different ways of operand addressing in 8051?[May/June 2016] R**

The five addressing modes are,

1. Immediate addressing
2. Register addressing
3. Direct addressing
4. Register indirect addressing

5. Indexed addressing.

**41.Which port used as multifunction port? List the signals. R(APR/MAY 2017)**

Port 3 is used as multi function port.

PORT 3 ALTERNATE FUNCTIONS :

| P3 BIT | FUNCTION | PIN |
|---|---|---|
| P3.0 | RXD | 10 |
| P3.1 | TXD | 11 |
| P3.2 | $\overline{\text{INT0}}$ | 12 |
| P3.3 | $\overline{\text{INT1}}$ | 13 |
| P3.4 | TO | 14 |
| P3.5 | TI | 15 |
| P3.6 | $\overline{\text{WR}}$ | 16 |
| P3.7 | $\overline{\text{RD}}$ | 17 |

**42. Illustrate the CJNE instruction. R (APR/MAY 2017)**

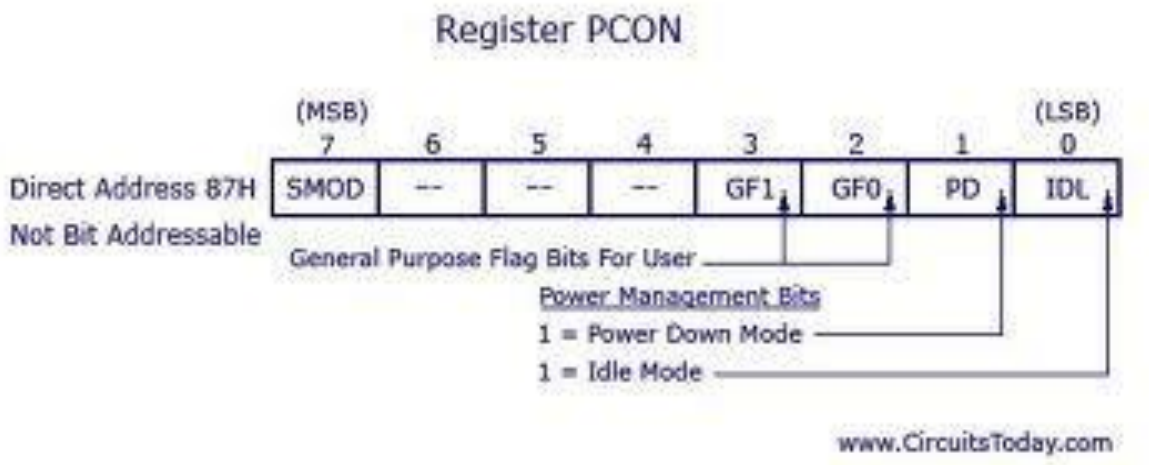
Compare and Jump if Not Equal – CJNE

– Compare the magnitude of the two operands and jump if they are not equal.
  - The values are considered to be unsigned.
  - The Carry flag is set / cleared appropriately.

- CJNE A, direct, rel
- CJNE A, #data, rel
- CJNE Rn, #data, rel
- CJNE @Ri, #data, rel

**43. How to set 8051 in idle mode? R [NOV/DEC 2017]**

The microcontroller 8051 can be driven to idle mode by setting IDL bit of PCON register. When idle mode is activated the clock signal is stopped to CPU (ALU), but the clock signal is supplied to interrupt, timer and serial port blocks. The idle mode can be terminated either by an interrupt or by hardware reset.

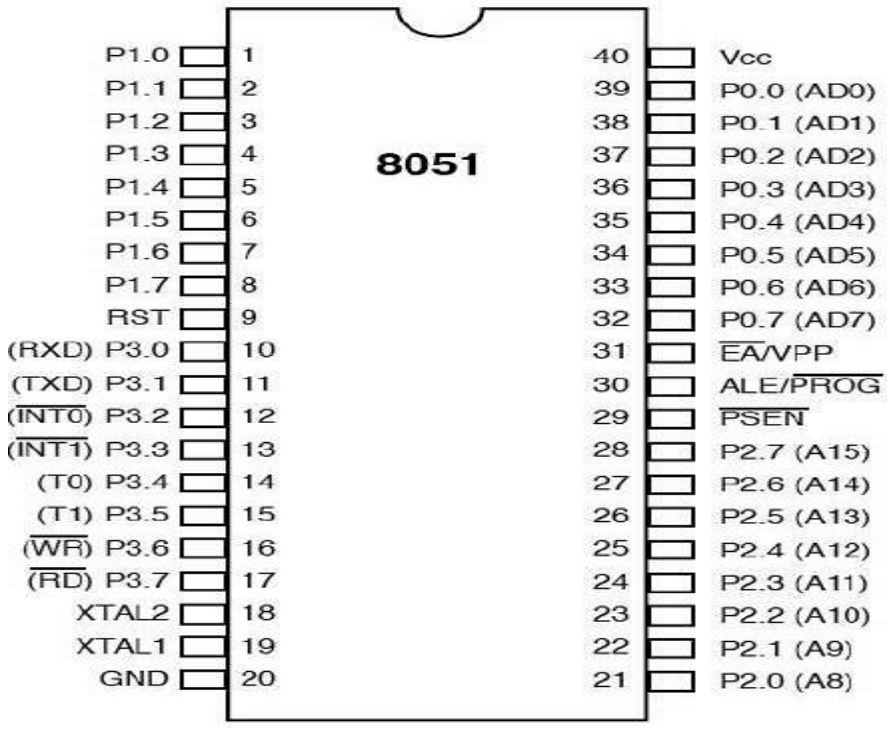



**44. Illustrate the DJNZ instruction. R [NOV/DEC 2017]**

In 8051, the loop action is performed by the instruction "DJNZ reg,label". In this instruction, the register is decremented; if it is not zero, it jumps to the target address referred to by the label. Prior to the start of the loop the register is loaded with the counter for the number of repetitions. In this instruction both the register decrement and the decision to jump are combined into a single instruction.
Ex:

```
        MOV A, #00
        MOV R2, #05
AGAIN:  ADD A, #25
        DJNZ R2, AGAIN
        MOV R5, A
```

**45. Draw the pin diagram of 8051. (Apr/May 2018)**




**46. what are bit manipulation instructions? Give two examples.R[April/May 2018]**

Boolean or Bit Manipulation Instructions will deal with bit variables. We know that there is a special bit-addressable area in the RAM and some of the Special Function Registers (SFRs) are also bit addressable.

Examples of Bit Manipulation instructions are:

These instructions can perform set, clear, etc.

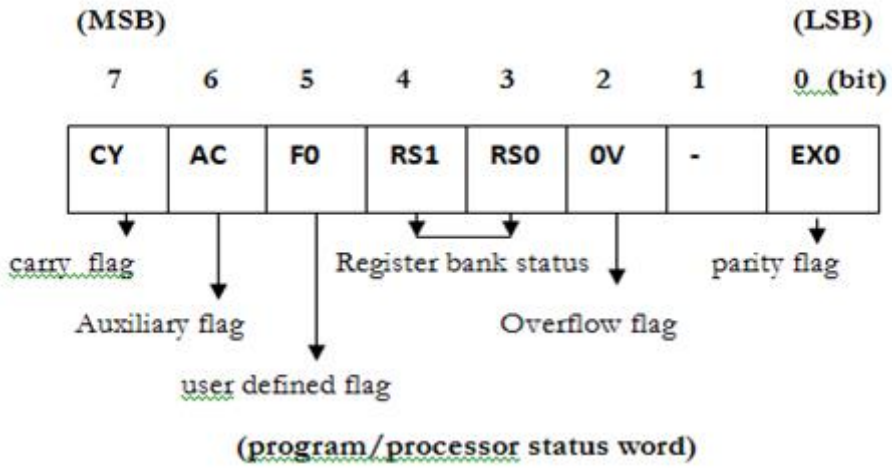
CLR,SETB,MOV,JC,JNC,JB,JNB,JBC,ANL,ORL,CPL.

CLR C ; CLR bit
SETB C ; SETB bit
CPL C ; CPL bit
MOV C,bit
MOV bit,C
ANL C,bit
ORL C,bit

**47. Which bits of the PSW are responsible for selection of the register banks?**

**R[April/May 2019]**

PSW Register bit 3 and 4 are responsible for register bank selection.




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

**48. Differentiate ARM processor and 8051 Microcontroller.**

| ARM PROCESSOR | 8051 MICROCONTROLLER |
|---|---|
| • ARM executes almost all the instruction in only one cycle Ex:1 cycles for execution of BNEQ in ARM | • 8051 micro controller takes more than one cycles in almost all the instruction except register transfer Ex:3 cycles for execution of DJNZ in 8051 |
| • RISC based architecture | • CISC based architecture |
| • based on load store architecture | • 8051 can access memory directly |

**49. What is the difference between PIC micro-controller and 8051 microcontroller?**

8051 MICROCONTROLLER has no in built A/D Converters but PIC has it. 8051 MICROCONTROLLER based on CISC architectuture while PIC based on RISC archirecture. 8051 has 250 instruction which take 1 to 4 machine cycle to execute while PIC has nearly 40 instruction.

**50. What is the difference between 8051 and 8086?**

8051: 16 bit Microcontroller

on chip ROM( 8KB) and On chip RAM (128 bytes)
two 16 bit timer/counter.
four 8-bit ports for input/output
fully duplex serial receiver/transmitter.
no prefetching of instruction.
16 address pins

8086: 16 bit Microprocessor

No on chip memory.
memory is divided into two banks to increase the processing speed.
prefetching of 6 bytes of instruction in a queue.
20 address pins

**51.** Write a program to mask the 0th and 7th bit using 8051. **R (APR/MAY 2021)**

```
ORG 00H
MOV A, #FFH ; A = FFH = 255D = 11111111B
ANL A, #7E H ; 126 D= 7E H= 01111110B
END
```

Input: FFH, Output: 7EH

**52.** Write a program to find the 2's complement using 8051. **R (APR/MAY 2021)**

```
ORG 0000H
MOV DPTR, #0030H
MOVX A, @DPTR
CPL A
ADD A, #01H
MOV DPTR, #0090H
MOVX @DPTR, A
MOV PCON, #02H
```

**53. Write an assembly language program to clear the accumulator and then add '3' to the accumulator ten times. (Nov/Dec 2021) C**

```
        MOV A,#0    ; A=0, clear ACC
        MOV R2,#10 ; load counter R2=10
Again: ADD A,#03    ; add 03 to ACC
```

```
DJNZ R2, AGAIN    ; repeat until R2=0 (10 times)
MOV R5, A         ; save A in R5
```

**54. Why do 55H and AAH is used to test the ports of 8051? (Nov/Dec 2021) U**

55H and AAH are used to test the ports of 8051 because complementing 55H (01010101) turns it into AAH (10101010). By sending 55H and AAH to a given port continuously, we toggle all the bits of that port.

**PART-B**

1. With the necessary diagram of control word format, explain the various operating modes of timer in 8051microcontroller **(EE2354May/June2014) U**
2. With the help of neat diagram explain the memory organization of 8051 microcontroller **(April/May2011)(Nov/Dec2014) U**
3. With neat sketch explain the architecture/ functional block diagram of 8051 microcontroller. **(Nov/Dec2010), (April/May2010)** (Apr/May 2015) (8) (**Nov/Dec2014) U**
4. Draw the Pin Diagram of 8051 and explain the function of various signals**. (Nov/Dec2010) U**
5. List the various Instructions available in 8051 microcontroller and explain. **(EE2354 May/June2014) C**

   Data Transfer Instructions. **(Nov/Dec2014)**

   Boolean variable Manipulation Instructions **(May/June 2013)**
6. (i) Explain the Data transfer instructions and Program control instructions of 8051 microcontroller. (8) **(April/May2011) U**

   (ii) Write an assembly language program based on 8051 microcontroller instruction set to perform four arithmetic operations on 2, 8 bit data. (8) **(April/May2011) C**
7. Discuss about the organization of Internal RAM and Special function registers of 8051 microcontroller in detail. (16) **(April/May2011) U**
8. Explain the arithmetic and control instructions of 8051microcontroller.(10) **(April/May2015)U**
9. (i) Explain the Interrupt structure with the associated registers in 8051 microcontroller. (8) **(April/May2011) (Nov/Dec2014) U**
10. Explain the TMOD function register and its timer modes of operations.(8) **(April/May2015) U**
11. Explain in detail about the special function register of 8051 in detail. **[NOV/DEC'15](8) U**
12. Explain the different addressing modes of 8051**. [NOV/DEC'15] U**
13. Give PSW of 8051 and describe the use of each bit in PSW. **[NOV/DEC'15] U**
14. Describe the functions of the following signals in 8051. RST, EA, PSEN and ALE**.U [NOV/DEC'15]**
15. Explain the architecture of 8051 with its diagram. (16) U **[May/June 2016]**

16. Write an 8051 ALP to create a square wave of 66% duty cycle on bit 3 of port 1. (16) C **[May/June 2016]**

17. Write the available special function registers in 8051. Explain each register with its format and functions. (13) U **[NOV/DEC 2018][April/May 2017]**

18. (i) Discuss the type of addressing mode with suitable example in 8051. (8) U

**[April/May 2017]**

(ii) Write an 8051 assembly language program to multiply the given number 48H and 30H. (5) C **[April/May 2017]**

19. Describe the architecture of 8051 with neat diagram. (13) U

**[April/May 2018][Nov/Dec 2017]**

20. Discuss the ports and its circuits of 8051. (13) U **[Nov/Dec 2017]**

21. How input/output Pins and Ports help in a circuit of a Microcontroller? (13) A/E

**[Nov/Dec 2018]**

22. How Microprocessor and Microcontrollers are different from computer based controllers? (13) A/E **[Nov/Dec 2018]**

23. How Microprocessor and Microcontrollers can help to control a process or a Machine tool? (13) A/E **[Nov/Dec 2018]**

24. (i) Explain in detail about the 8051 register banks and stack. (8) U **[April/May 2019]**

(ii) Show the code to push R5,R6 and A onto the stack and then pop them back into R2,R3 and B, where register B = register A, R2 = R6 and R3 = R5. (5) A/E **[April/May 2019]**

25. Briefly explain about the various addressing modes of 8051 with one example. (13) U **[April/May 2018]** [**April/May 2019]**

26. With neat sketch explain the architecture of 8051 microcontroller. R [**April/May 2021]**

27. i) Explain the different addressing modes of 8051. U **[April/May 2021]**

ii) List the various instructions available in 8051 microcontroller. U

28. With a functional block diagram, briefly discuss the architecture of the 8051 microcontroller. **(Nov/Dec 2021)**

29. (i) Discuss on the different types of addressing modes supported by the8051 microcontroller with examples. (10)

(ii) State the merits and demerits of accessing memory using various addressing modes. (3) **(Nov/Dec 2021)**

## ASSIGNMENT QUESTIONS

1. Explain the logical instructions in 8051 with an example. **(Evaluate)**
2. Explain the functions of I/O Ports in 8051. **(Evaluate)**

## UNIT-V
## INTERFACING MICROCONTROLLERS

# PART-A

**1. What is a serial data buffer? R**

Serial data buffer is a special function register and it initiates serial transmission when byte is written to it and if read, it reads received serial data. It contains two independent registers internally. One of them is a transmit buffer, which is a parallel-in serial-out register. The other is a receive buffer, which is a serial-in parallel-out register

2. **What are timer registers? R**

Timer registers are two 16-bit registers and can be accessed as their lower and upper bytes. TLO represents the lower byte of the timing register 0, while THO represents higher bytes of the timing register 0. Similarly, TLI and THI represent lower and higher bytes of timing register 1. These registers can be accessed using the addresses allotted to them, which lie in the special function registers address range, i.e., 801 H to FF.

**3. What is the use of timing and control unit? R**

Timing and control unit is used to derive all the necessary timing and control signals required for the internal operation of the circuit. It also derives control signals that are required for controlling the external system bus.

**4. When are timer overflow bits set and reset? R**

The timer overflow bits are set when timer rolls over and reset either by the execution of an RET instruction or by software, manually clearing the bits. The bits are located in the TCON register along with timer run control (TRn) bits.

**5. Explain the mode (0 and1) operation of the timer.(April/May2012) U**

The operations are as follows:

• Timer mode 0 and 1 operations are similar for the 13 bit (mode) or 16 bit (mode 1) counter. When the timer reaches the limits of the count, the overflow flag is set and the counter is reset back to zero.

• The modes 0 and 1 can be used to time external events.

• They can be used as specific time delays by loading them with an initial value before allowing them to execute and overflow.

**6. What is the different modes in which timer 2 can operate? R**

The two different modes in which Timer 2 operates are.

i**. Capture mode**: Timer 2 operates as free running clocks, which saves the timers value on each high to low transition. It can be used for recording bit lengths when receiving Manchester-encoded data.

ii. **Auto-reload mode**: When the timer overflows, value is written into TH2/TL2 registers from RCA P2H/RCA P21 registers. This
feature is used to implement a system watch dog timer.

**7. What is the use of a watch dog timer? R**

A watching timer is used to protect an application in case the controlling microcontroller begins to run amok and execute randomly rather than the preprogrammed instructions written for the application.

**8. Define interrupt. R**

Interrupt is defined as request that can be refused. If not refused and when an interrupt request is acknowledged, a special set of routine or events are followed to handle the interrupt.

**7. What are the steps followed to service an interrupt? R**

The steps followed are:

I. Save the context register information.
II. Reset the hardware requesting the interrupt.
III. Reset the interrupt controller.
IV. Process the interrupt.
V. Restore the context information.
VI. Return to the previously executing code.

**8. How can 8051 be interrupted? A/E**

There are five different ways to interrupt 8051. Two of these are from external electrical signals. The other three are caused by internal 8051 I/O hardware operations.

**9. Give the format of the interrupt enable register. (april/ May2013) U**

The format of the interrupt enable register is,

EA--ES ET1 EX1 ET0 EX0

The register is used to enable or disable all 8051 interrupts and to selectively enable or disable each of the five different interrupts.

- EA: Disables all interrupts
- Es: Enables or disable the serial port interrupt.
- ET1: Enable or disable the timer 1 overflow interrupt.
- EX1: Enable or disable external interrupt 1.
- ET0: Enable or disable the timer 0 overflow interrupt.
- EX0: Enable or disable external interrupt 0.

**10. What is meant by nesting of interrupts? R**

Nesting of interrupts means that interrupts are re-enabled inside an interrupt handler. If another interrupt request codes in, while the first interrupt handler is executing, processor execution will acknowledge the new interrupt and jump to its vector.

**11. How is the 8051 serial port different from other micro controllers? Nov/Dec2013. A/E**

The 8051 serial port is a very complex peripheral and able to send data synchronously and asynchronously in a variety of different transmission modes.

**12. Explain synchronous data transmission. U**

• In synchronous mode (mode 0), the instruction clock is used.
• Data transfer is initiated by writing to the serial data port address.
• Txd pin is used for clock output, while Rxd pin is for data transfer.
• When a character is received, the status of the data transfer is monitored by polling the RI-n bit in serial control register (SCON).

**13. Give an application for synchronous serial communication. U**

An application for synchronous serial communication is RS–232.

**14. When is an external memory access generated in 8051? A/E [April/May 2019]**

In 8051, during execution the data is fetched continuous. Most of the data is executed out of the 8051‟sbuilt-in control store. When an address is outside the internal control store, an external memory access is generated.

**15. Give the priority level of the interrupt sources in 8051.**

**[April/May 2018](Nov/Dec2010) U**

Interrupt source Priority within a level
IE0 (External INT0)
TF0 (Timer 0)
IE 1 (External INT 1)
TF 1 (Timer 1)
RI = TI (Serial port) Highest
.
.
Lowest

**16. What is the use of stepper motor? R**

A stepper motor is a device used to obtain an accurate position control of rotating shafts. A stepper motor employs rotation of its shaft in terms of steps, rather than continuous rotation as in case of AC or DC motor.

**17. What is meant by key bouncing? R**

Microprocessor must wait until the key reach to a steady state; this is known as Key bounce.

**18. Explain the operating mode0 of 8051 serial ports? U**

In this mode serial enters &exits through RXD, TXD outputs the shift clock.8 bits are transmitted/received:8 data bits(LSB first).The baud rate is fixed at 1/12 the oscillator frequency.

**19. Explain the operating mode2 of 8051 serial ports? April/May 2009&Nov/Dec2008) U**

In this mode 11 bits are transmitted(through TXD)or received(through RXD):a start bit(0), 8 data bits(LSB first),a programmable 9th data bit ,& a stop bit(1).ON transmit the 9th data bit (TB* in SCON)can be assigned the value of 0 or 1.Or for eg:, the parity bit(P, in the PSW)could be moved into TB8.On receive the 9thdatabit go in to the RB8 in Special Function Register SCON, while the stop bit is ignored. The baud rate is programmable to either 1/32or1/64 the oscillator frequency.

**20. Explain the mode3 of 8051 serial ports? (april/May2008) U**

In this mode,11 bits are transmitted(through TXD)or received(through RXD):a start bit(0), 8 data bits(LSB first),a programmable9th data bit ,& a stop bit(1).In fact ,Mode3 is the same as Mode2 in all respects except the baud rate. The baud rate in Mode3 is variable. In all the four modes, transmission is initiated by any instruction that uses SBUF as a destination register. Reception is initiated in Mode0 by the condition RI=0&REN=1.Reception is initiated in other modes by the incoming start bit if REN=1.

**21. Write a program to mask the 0th&7thbit using8051? C**

```
MOV A,#data
ANL A,#81
MOV DPTR,#4500
MOVX @DPTR,A
LOOP SJMP LOOP
```

**22. Write about CALL statement in 8051? C**

There are two subroutine CALL instructions. They are

*LCALL(Long CALL)

*ACALL(Absolute CALL)

Each increments the PC to the 1stbyte of the instruction & pushes them in to the stack.

**23. Write a program to find the 2's complement using 8051? C**

```
MOV A,R0
CPL A
INC A
```

**24. Define baud rate. [May/June 2016] R**

Baud rate is used to indicate the rate at which data is being transferred.
Baud rate = 1/Time for a bit cell.

**25. Mention the features of serial port in mode 0. [NOV/DEC'15] U**

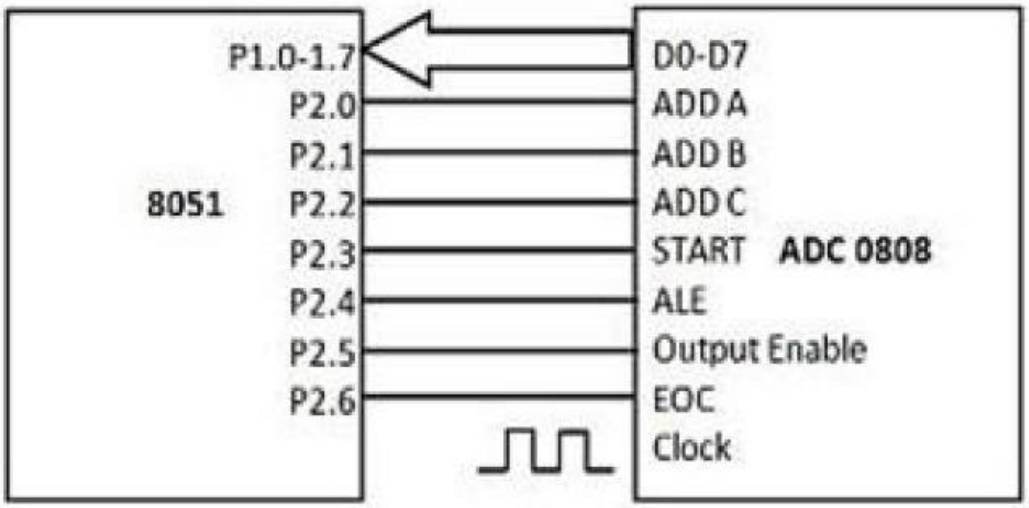
In this mode serial enters and exits through RXD, TXD outputs the shiftclock. 8 bits are transmitted /received 8 data bits first (LSB first).The baudrate is fixed at 1/12 the oscillator frequency.

**26. Which register is used for serial programming in 8051 microcontroller? Illustrate it. [APR/MAY'15] A/E**

SBUF Register (Serial Buffer):
SBUF is an 8-bit register for serial communication in 8051.For a byte of data to be transferred via TxD line and holds the byte of data when it is received by 8051's RxD line.
SCON Register (Serial Control):
SCON is an 8 bit register used to program the start bit,stop bit and data bits of data framing among other things.

| SM0 | SM1 | SM2 | REN | TB8 | RB8 | T1 | R1 |
|---|---|---|---|---|---|---|---|

**27. How is A/D convertor interfaced with 8051? [NOV/DEC'15] A/E**




**28. Compare polling and interrupt.[May/June 2016] A/E**

The 8051 microcontroller can do only one task at a time. In polling, the microcontroller continuously checks each port one by one according to the priority assigned to the ports, and if any device requires service, then it provides it. In interrupt, when the device requires service, it sends the request to microcontroller and the controller then provides service to it.

So essentially, the difference is that in polling, microcontroller has to check continuously whether any device is asking for request, while in interrupt the device itself sends the request and

the controller satisfies it. And because microcontroller is freed from the task of checking each port, it can do other work.

29. **List the 8051 interrupts with priority.**R **(APR/MAY 2017)[NOV/DEC 2017]**

| Highest to Lowest Priority | |
|---|---|
| External Interrupt 0 | (INT0) |
| Timer Interrupt 0 | (TF0) |
| External Interrupt 1 | (INT1) |
| Timer Interrupt 1 | (TF1) |
| Serial Communication | (RI + TI) |

**30.What are the types of sensors used for interfacing? R**

**[April/May 2018](APR/MAY 2017)**

Temperature, Humidity, Light, Acceleration, Force, Frequency, Flow, Pressure, Torque, Proximity, Displacement.

**31. Give two examples of sensors and state its uses. R [NOV/DEC 2017]**

**Temperature Sensor**

A device which gives temperature measurement as an electrical signal is called as temperature sensor. This electrical signal will be in the form of electrical voltage and is proportional to the temperature measurement.

**Ultrasonic Sensor**

The principle of ultrasonic sensor is similar to sonar or radar in which interpretation of echoes from radio or sound waves to evaluate the attributes of a target by generating the high-frequency-sound waves (around 40kHz). The transducer used for converting energy into ultrasound or sound waves with ranges above human hearing range is called an ultrasonic transducer.

**32. How to program 8051 Timers? A/E [NOV/DEC 2018]**

The 16-bit register of Timer 0 is accessed as low byte and high byte. The low byte register is called TLO (Timer 0 low byte) and the high byte register is referred to as THO (Timer 0 high byte).

Timer 1 is also 16 bits, and its 16-bit register is split into two bytes, referred to as TLl (Timer 1 low byte) and TH1 (Timer 1 high byte). These registers are accessible in the same way as the registers of Timer 0.

Both timers 0 and 1 use the same register, called TMOD, to set the various timer operation modes. TMOD is an 8-bit register in which the lower 4 bits are set aside for Timer 0 and the upper 4 bits for Timer 1. In each case, the lower 2 bits are used to set the timer mode and the upper 2 bits to specify the operation.

**33. What are the types of ADC? R [NOV/DEC 2018]**

Flash ADC
Sigma-delta
ADC
Dual slope converter
Successive approximation converter
ADC 0803, ADC 0804, ADC 0805 – CMOS 8-bit Successive approximation ADC
ADC 0808, ADC 0809 – Monolithic CMOS 8-bit Successive approximation ADC

**34. What are the different modes in which timer 2 can operate? R [April/May 2019]**

The two different modes in which Timer 2 operates are.

i. **Capture mode**: Timer 2 operates as free running clocks, which saves the timers value on each high to low transition. It can be used for recording bit lengths when receiving Manchester-encoded data.

ii. **Auto-reload mode**: When the timer overflows, value is written into TH2/TL2 registers from RCA P2H/RCA P21 registers. This feature is used to implement a system watch dog timer.

**30. When is an external memory access generated in 8051? (APR/MAY 2019)**

In 8051, during execution the data is fetched continuously. Most of the data is executed out of the 8051‟s built-in control store. When an address is outside the internal control store, an external memory access is generated.

**31. Define the operating model 0 of 8051 serial ports. U (APR/MAY 2021)**

Mode 0 is a synchronous communication system. Through the RxD pin is transmit data, while the sync pulses are located at TxD. For this mode, the transmission speed is fixed to the oscillator frequency ÷ 12.

**32. Give the different types of ADC. R (APR/MAY 2021)**

- Successive Approximation (SAR) ADC.
- Delta-sigma (ΔΣ) ADC
- Dual Slope ADC.
- Pipelined ADC.
- Flash ADC.

**33.Which register has the SMOD bit, and what is its status when the 8051**

**ispowered up? (Nov/Dec 2021)**

The SMOD Bit in **the PCON Register** is used to control the Baud Rate of the Serial Port.It is 8-bit register. When 8051 is powered up, **SMOD is zero**.By setting the SMOD, baud rate can be doubled. If SMOD = 0 (which is its value on reset), the baud rate is 1/64 the oscillator frequency.

## PART-B

1. With neat sketch explain the functions of 8251. **(Nov/Dec2011) U**
5. With neat sketch explain the function of A/D converter**.** **(Nov/Dec 2014) U**
6. With neat sketch explain the function of D/A converter. **(Nov/Dec 2014) U**
7. (i) Explain the interfacing of Keyboard/Display with 8051 microcontroller. (8) **(April/May2011**) U
8. (i) Explain in detail the modes of operation of Timer unit in 8051 microcontroller. (8) **(April/May2011) U**

   (ii) Explain the Stepper motor control using 8051 microcontroller.(8)**(April/May2011) (Nov/Dec2014) U**
10. Using 8051 timer/Counter write a program for generating square wave of 100 ms and 50% duty cycle and Explain. **(April/May2011)(Nov/Dec2014)(Nov/Dec2010) C**
1112. Describe the different modes of operation of timers/counters in 8051 with its associated register. **(April/May2015) (10) R**
13. How does one interface a 16 x 2 LCD display using 8051 microcontroller. **(April/May2015) (6). A/E**
14. Draw the diagram to interface a stepper motor with 8051 microcontroller and explain. Write a 8051 assembly language program to run the stepper motor in both forward and reverse direction with delay. (16). **(April/May2015). C**
15. How to interface an LCD display with microcontroller? Explain how to display a character using LCD display. **(Nov/Dec 2014). (8) A/E**

**16.** Draw the schematic for interfacinga stepper motor with 8051microcontroller and write 8051 ALP for keypad scanning**.** **[NOV/DEC'15][APR/MAY'15] C**

**17.** With a neat circuit diagram explain how a 4x4 keypad is interfaced with 8051 microcontroller and write 8051 ALP for keypad scanning.**[NOV/DEC'15][MAY/JUNE'13] U**

18. Draw the diagram to interface a stepper motor with 8051 microcontroller and Write its ALP to run the stepper motor in both forward and reverse direction with delay.(16) C **[April/May 2018][May/June 2016]**

19. Explain 8051 serial port programming with examples. (16) U **[May/June 2016]**

20. Write a program for generation of unipolar square waveform of 1 KHz frequency using Timer 0 of 8051 in mode 0. Consider the system frequency as 12MHz. (13) C **[April/May 2017]**

21. Demonstrate the interfacing of the stepper motor with 8051 and explain its interfacing diagram and develop program to rotate the motor in clock wise direction. (13) U **[April/May 2017]**

22. Develop 8051 based system design having 8Kbyte RAM to generate the triangular wave using DAC. (15) U **[April/May 2017]**

23. Illustrate the serial communication in 8051, with its special function register. (13) U **[Nov/Dec 2017]**

24.(i) Interface the ADC converter with 8051 and explain with neat diagram. (7) U

(ii) Write the assembly language program to execute the ADC conversion. (6) C **[Nov/Dec 2017]**

25. Design a circuit to generate 12 MHz frequency for a system. Write a program for generation of unipolar square waveform of 1 KHz frequency using Timer 0 of 8051 in mode 0. (15) C

**[Nov/Dec 2017]**

26. Write and explain, What is known as Serial Port programming? (13)U**[Nov/Dec 2018]**

27. What are Sensor interfacing and External memory interfacing? Explain(13)U[**Nov/Dec 2018]**

28. Describe the different modes of operation of timers/counters in 8051 with its associated registers. U (13) **[April/May 2018]**

29. Describe how to program and interface an LCD to an 8051 using Assembly language programming. (13) U **[April/May 2019]**

30. Draw and explain the DAC interfacing using 8051. (13) U **[April/May 2019]**

31. (i) Discuss what happens if interrupts INT0,TF0, and INT1 are activated at the same time. Assume priority levels were set by the power-up reset and that the external hardware interrupts areedge-triggered. (7) U **[April/May 2019]**
(ii) With necessary diagrams explain how to interface LM35 temperature sensor and then discuss the issue of signal conditioning. (8) U **[April/May 2019]**

32.(i) Discuss the number of pin sets aside for addresses in each of the following memory chips.

(1) 16K * 4 DRAM and (2) 16 K * 4 SRAM. (4) U **[April/May 2019]**

(ii) Briefly explain about the interfacing of 8051 with external data ROM.

(11) U[April/May 2019]

33. With neat sketch block diagram of interfacing 64KB of External RAM and 64KB of External ROM with the 8051 Microcontroller. **U [April/May 2021]**

34. Draw and explain the ADC interfacing using 8051.R **[April/May 2021]**

35. In a certain application, 256K bytes of NV-RAM are needed to store data collected by an 8051 microcontroller. Show the connection of an 8051 to a single 256K × 8 NV-RAM chip. **(Nov/Dec 2021) A**

36. (i) Differentiate RET and RETI instructions. Explain why RET cannotbe used as the last instruction of an ISR instead of RETI. (6)

(ii)Illustrate the options available with Timer Mode (TMOD) registerof 8051. (7)
**(Nov/Dec 2021) U**

37. Interface a temperature sensor to the 8051 microcontroller through an analog to digital converter and outline the steps to be followed to get data from analog input of ADC into the microcontroller. **(Nov/Dec 2021) C**

**38.**

## ASSIGNMENT QUESTIONS

1. Draw the circuit diagram to interface LCD with microcontroller and explain how To display the data using LCD. **(Understand)**
2. How are the timers of 8051 used to produce time delays in timer mode? **(Remember)**