CAPITAL UNIVERSITY - KODERMA

MICROPROCESSOR AND MICROCONTROLLER - ASSIGNMENT

Name : Arshad Nazir

Electrical and Electronics Engineering

Signature:

Date :

Part 1

Q1. Identify the difference between Macro and subroutine.

Ans. Macros can only be used in the program they are defined in and only after the definition is expandedat compilation/generation.

Subroutines (FORM) can be called from both the program they are defined in and other programs.

A MACRO is more or less an abbreviation for some lines of code that are used more than once or twice.A FORM is a local subroutine (which can be called external). A FUNCTION is (more or less) a subroutine that is called external.

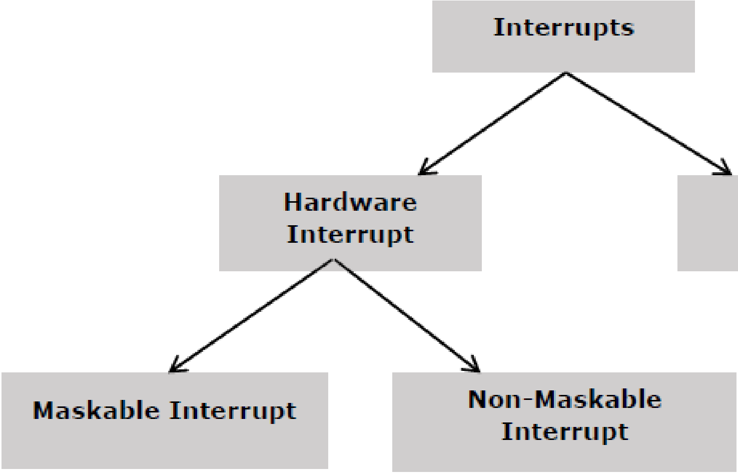
Since debugging a MACRO is not really possible, prevent the use of them (I've never used them, but seenthem in action). If the subroutine is used only local (called internal) use a FORM. If the subroutine is called external (used by more than one program) use a FUNCTION.

1. When a task is to be done repeatedly then it is written as subroutine and this subroutine will be calledeach time to perform that task.
2. Subroutine program will be stored in some memory location and program control will be transferred tothat location each time.
3. whereas in macro the number of instructions will be less than subroutine. Here each time u call a macrothat set of instructions will be inserted in that location.
4. macro doesn't have a return statement while subroutine has.
5. memory requirement for macro is higher.
6. execution time of macro is lesser than subroutine.

Q2. Name the different types of interrupts supported by 8086.

Ans. The interrupt is the method of creating a temporary halt during program execution and allows peripheral devices to access the microprocessor. The microprocessor responds to that interrupt with an ISR (Interrupt Service Routine), which is a short program to instruct the microprocessor on how to handlethe interrupt.

The following image shows the types of interrupts we have in an 8086 microprocessor –



Interrupts

- Hardware Interrupts

A hardware interrupt is caused by any peripheral device by sending a signal through a specified pin to themicroprocessor.

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.

- NMI

It is a single non-maskable interrupt pin (NMI) having higher priority than the maskable interrupt requestpin (INTR)and it is of type 2 interrupt.

When this interrupt is activated, these actions take place –

Completes the current instruction that is in

progress.Pushes the Flag register values onto the

stack.

Pushes the CS (code segment) value and IP (instruction pointer) value of the return address onto the

stack.IP is loaded from the contents of the word location 00008H.

CS is loaded from the contents of the next word location

0000AH.The interrupt flag and trap flag are reset to 0.

- INTR

The INTR is a maskable interrupt because the microprocessor will be interrupted only if interrupts are enabled using set interrupt flag instruction. It should not be enabled using clear interrupt Flag instruction.

The INTR interrupt is activated by an I/O port. If the interrupt is enabled and NMI is disabled, then themicroprocessor first completes the current execution and sends ‘0’ on the INTA pin twice. The first ‘0’means INTA informs the external device to get ready and during the second ‘0’ the microprocessor receives the 8-bit, say X, from the programmable interrupt controller.

These actions are taken by the microprocessor –

First completes the current instruction.

Activates INTA output and receives the interrupt type, say X.

Flag register value, CS value of the return address and IP value of the return address is pushed onto the stack.

IP value is loaded from the contents of word location X ×

4CS is loaded from the contents of the next word

location.

The interrupt flag and trap flag is reset to 0

- Software Interrupts

Some instructions are inserted at the desired position into the program to create interrupts. These interruptinstructions can be used to test the working of various interrupt handlers. It includes –

INT- Interrupt instruction with type number

It is 2-byte instruction. First byte provides the op-code and the second byte provides the interrupt typenumber. There are 256 interrupt types under this group.

Its execution includes the following steps –

The flag register value is pushed onto the stack.

CS value of the return address and IP value of the return address is pushed onto the

stack.IP is loaded from the contents of the word location 'type number' × 4

CS is loaded from the contents of the next word

location.Interrupt Flag and Trap Flag is reset to 0

The starting address for type0 interrupt is 000000H, for type1 interrupt is 00004H similarly for type2 is 00008H and ……so on. The first five pointers are dedicated interrupt pointers. i.e. –

TYPE 0 interrupt represents division by zero situation.

TYPE 1 interrupt represents single-step execution during the debugging of a program.

TYPE 2 interrupt represents a non-maskable NMI interrupt.

TYPE 3 interrupt represents break-point

interrupt.TYPE 4 interrupt represents overflow

interrupt.

The interrupts from Type 5 to Type 31 are reserved for other advanced microprocessors, and interruptsfrom 32 to Type 255 are available for hardware and software interrupts.

INT 3-Break Point Interrupt Instruction

It is a 1-byte instruction having an op-code is CCH. These instructions are inserted into the program sothat when the processor reaches there, then it stops the normal execution of a program and follows the break-point procedure.

Its execution includes the following steps –

The flag register value is pushed onto the stack.

CS value of the return address and IP value of the return address is pushed onto the

stack.IP is loaded from the contents of the word location 3×4 = 0000CH

CS is loaded from the contents of the next word

location.Interrupt Flag and Trap Flag is reset to 0

INTO - Interrupt on overflow instruction

It is a 1-byte instruction and their mnemonic INTO. The op-code for this instruction is CEH. As the namesuggests it is a conditional interrupt instruction, i.e. it is active only when the overflow flag is set to 1 andbranches to the interrupt handler whose interrupt type number is 4. If the overflow flag is reset then, the execution continues to the next instruction.

Its execution includes the following steps –

Flag register values are pushed onto the stack.

CS value of the return address and IP value of the return address is pushed onto the

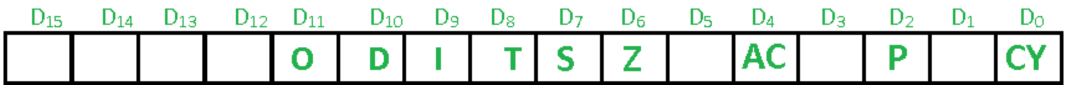
stack.IP is loaded from the contents of word location 4×4 = 00010H

CS is loaded from the contents of the next word

location.Interrupt flag and Trap flag are reset to 0

Q3. List the flag register in 8086.

Ans. The Flag register is a Special Purpose Register. Depending upon the value of the result after any arithmetic and logical operation, the flag bits become set (1) or reset (0).



D_{15}	D_{14}	D_{13}	D_{12}	D_{11}	D_{10}	D_9	D_8	D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
				O	D	I	T	S	Z		AC		P		CY

Figure – Format of the flag register

There are a total of 9 flags in 8086 and the flag register is divided into two types:

(a) Status Flags – There are 6 flag registers in the 8086 microprocessor which become set(1) or reset(0) depending upon condition after either 8-bit or 16-bit operation. These flags are conditional/status flags. 5 of these flags are the same as in the case of the 8085 microprocessor and their working is also the same asin the 8085 microprocessor. The sixth one is the overflow flag.

The 6 status flags are:

- Sign Flag (S)

- Zero Flag (Z)
- Auxiliary Cary Flag (AC)
- Parity Flag (P)
- Carry Flag (CY)

(b) Control Flags – The control flags enable or disable certain operations of the microprocessor. There are3 control flags in the 8086 microprocessor and these are:

- Directional Flag (D) – This flag is specifically used in string instructions.

If the directional flag is set (1), then access the string data from the higher memory location towards thelower memory location.

If the directional flag is reset (0), then access the string data from the lower memory location towards thehigher memory location.

- Interrupt Flag (I) – This flag is for interrupts.

If the interrupt flag is set (1), the microprocessor will recognize interrupt requests from the peripherals.

If the interrupt flag is reset (0), the microprocessor will not recognize any interrupt requests and will ignore them.

- Trap Flag (T) – This flag is used for on-chip debugging. Setting the trap flag puts the microprocessor into a single-step mode for debugging. In single stepping, the microprocessorexecutes an instruction and enters into single step ISR.

If the trap flag is set (1), the CPU automatically generates an internal interrupt after each instruction, allowing a program to be inspected as it executes instruction by instruction.

If the trap flag is reset (0), no function is performed.

Q4. What are the assembler directives there in 8086?

Ans. The 8086 instructions are categorized into the following main types. 1. Data Copy / Transfer Instructions 2. Arithmetic and Logical Instructions 3. Shift and Rotate Instructions 4. Loop Instructions 5.Branch Instructions 6. String Instructions 7. Flag Manipulation Instructions 8. Machine Control Instructions.

The 8086 instructions are categorized into the following main types.

1. Data Copy / Transfer Instructions
2. Arithmetic and Logical Instructions
3. Shift and Rotate Instructions

4. Loop Instructions

5. Branch Instructions

6. String Instructions

7. Flag Manipulation Instructions

8. Machine Control Instructions

Q5. Outline the different types of addressing modes of 8086

Ans. The way of specifying data to be operated by an instruction is known as addressing modes. This specifies that the given data is immediate data or an address. It also specifies whether the given operand isregister or register pair.

Types of addressing modes:

- Register mode – In this type of addressing mode both the operands are registers.

Example:

MOV AX, BX

XOR AX, DX

ADD AL, BL

- Immediate mode – In this type of addressing mode the source operand is a 8-bit or 16-bit data.The destination operand can never be immediate data.

Example:

MOV AX, 2000

MOV CL, 0A

ADD AL, 45

AND AX, 0000

Note that to initialize the value of the segment register a register is required.

MOV AX, 2000

MOV CS, AX

- Displacement or direct mode – In this type of addressing mode the effective address is directly given in the instruction as displacement.

Example:

MOV AX, [DISP]

MOV AX, [0500]

- Register indirect mode – In this addressing mode the effective address is in SI, DI or

BX.Example: Physical Address = Segment Address + Effective Address

MOV AX, [DI]

ADD AL, [BX]

MOV AX, [SI]

- Based indexed mode – In this, the effective address is the sum of the base register and indexregister.

Base register: BX, BP

Index register: SI, DI

The physical memory address is calculated according to the base register.

Example:

MOV AL, [BP+SI]

MOV AX, [BX+DI]

- Indexed mode – In this type of addressing mode the effective address is the sum of the indexregister and displacement.

Example:

MOV AX, [SI+2000]

MOV AL, [DI+3000]

- Based mode – In this, the effective address is the sum of base register and displacement.

Example:

MOV AL, [BP+ 0100]

- Based indexed displacement mode – In this type of addressing mode, the effective address is the sum of the index register, base register and displacement.

Example:

MOV AL, [SI+BP+2000]

- String mode – This addressing mode is related to string instructions. In this, the value of SI andDI are auto-incremented and decremented depending upon the value of the directional flag.

Example:

MOVS B MOVS

W

- Input/Output mode – This addressing mode is related to input-output operations.

Example:

IN A, 45

OUT A, 50

- Relative mode –

In this, the effective address is calculated with reference to the instruction pointer.

Example:

JNZ 8 bit address

IP=IP+8 bit address

Micro Processors

Part-1

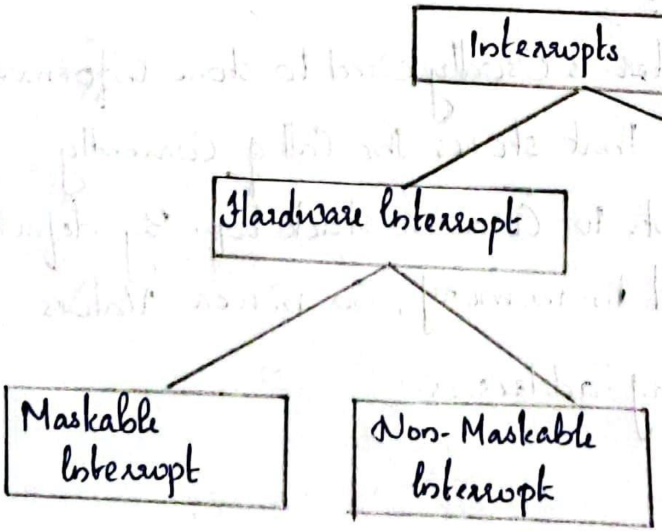
1. Identify the difference between Macro and Subroutine?

A. Macro:- Macro Can be Called only in the program it is defined. Macro Can have a parametres. Macro Can be Called only after its definition. Macro Is Used when Same thing Is done In a programe a number of times.

Subrootine:- Subroutine Can be Called from other programs also. Can have any number of parameters. This Is not true for Subroutine. Subroutine Is Used for modularization.

2. Name the different types of Interrupts Supported by 8086?

A The following Image shows the types of Interrupts we have In a 8086 microprocessor -



Hardware Interrupts

Hardware Interrupt Is ~~Used~~ Caused by any peripheral device by Sending a Signal through a Specified pin to microprocessor.

Software Interrupts

Some Instructions are Inserted at the desired position Into the program to Create Interrupts. These Interrupt Instructions Can be Used to test the Working of Various Interrupt handlers.

3. List the flag register In 8086?

A flag Register Is One of the Special register. The flag bits are changed to 0 or 1 depending Upon the Value of result after arithmetic Or logical Operations.

8086 has 16-bit flag register, and there are 9 Valid flag bits. We Can divide the flag bits Into two Sections. The status flags and the Control flags.

4. Define stack Segment register?

A The stack Segment register is Usually Used to store Information about the memory Segment that stores the Call 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.

5. What are the assembler directives there In 8086?

A. a) The DB directive

b) The DW directive

c) The DD directive

d) The struct (or STRUC) and ends directives (Counted as One)

e) The EQU directive

f) The COMMENT directive

g) ASSUME

h) EXTERN

i) GLOBAL

j) SEGMENT

k) OFFSET

l) PROC

m) GROUP

n) INCLUDE

Part-2A

2. Examine the Various addressing modes available In 8086. Explain each mode with an Example?

A The way of Specifying data to be Operated by an Instruction is known as addressing modes. This Specifies that the given data is an Immediate data or an address. It also Specifies whether the given Operand is register or register pair.

Types of addressing modes :-

* Register mode - In this type of addressing mode both the Operands are register.

Example : MOV AX, BX
XOR AX, DX
ADD AL, BL

* Immediate mode - In this type of addressing mode the Source Operand is a bit Or 16 bit data. Destination Operand Can be never Immediate data.

*. Displacement or direct mode - In this type of addressing mode the effective address is directly given Instruction as displacement.

Example : MOV AX, (DISP)
MOV AX, (0500)

*. Register Indirect mode - In this addressing mode the effective address is in SI, DI or BX.

Example - Physical address = Segment Address + Effective Address,
MOV AX, (DI)
MOV AI, (BX)
MOV AX, (SI)

* Based Indexed mode - In this the effective address is Sum of based register and Index register.

Base register : BX, BP

Index register : SI, DI

The physical memory address is Calculated according to the base register

Example : MOV AI, (BP+ SI)
MOV AI, (BX + DI)

* Indexed mode - In this type of addressing mode the effective address is Sum of Index register and displacement.

Example : MOV AX, (SI+2000)

MOV AI, (DI+3000)

* Based mode - In this effective address is the Sum of base register and displacement.

Example : MOV AI, [BP + 0100]

* Based Indexed displacement mode - In this type of addressing mode the effective address is the Sum of Index register, base register, displacement.

Example : MOV AI, [SI+ BP+ 2000]

* string mode - This addressing mode is related to string Instructions. In this the Value of SI and BI are auto Incremented and decremented depending Upon the Value of directional flag.

Example : MOVS B

MOVS W

* Input /Output mode - This addressing mode is releated with Input Output Operations

Example : IN A, 45

OUT A, 50

* Relative mode - In this effective address is Calculated with reference to Instruction pointer.

Example : JNZ 8 bit address

IP = IP+8 bit address

1. 1) Write an 8086 ALP to Convert BCD data to Binary data?

A.

```
Data_Seg Segment
BCD DB 25H
; storage for a BCD Value
BIN DB ?
; storage for binary Value
Data_Seg ends

Code_Seg Segment
Assume
 CS: Code_Seg. DS: data_Seg
start:
MOV AX, Data Seg
MOVDS, AX

MOV AH, BCD
MOV BH, AH
AND BH, OFH
AND AH, OFOH
ROR AH, 04
MOV Cl, 10
MOV Al, AH
MOVAX, OOFFH
MUL CL
ADD Al, BH
MOV BIN, Al
MOV AH, 04CH
INT, 21H
CODE_SEG ENDS
END START
```

i) Identify the Condition which Cause the 8086 to perform type 0 and type 1 Interrupt.

A. The starting address for type 0 Interrupt is 000000H, for type 1 Interrupt is 00004H.

Type 0 - Interrupt represents division by zero situation.
Type 1 - Interrupt represents Single-step execution during during the debugging of a program.

Part-2.

1. Define Bus?

A. A bus is a high-Speed Internal Connection. Buses are Used to Send Control Signals and data between the processor and other Components. Three types of bus are Used. Address Bus, Carries memory addresses from Such as primary storage, and Input, Out devices.

2. state about external and Internal bus?

A. An Internal bus enables the Communication between Internal Components, Such as a Video Card and memory. An external bus is Capable of Communicating with external peripherals, Such as USB or SCSI device.

3. How would you explain two modes of Operation in 8086?

A There are two operating modes of operation for intel 8086, namely the minimum mode and maximum mode. when only on 8086 CPU is to be used in a microprocessor system, the 8086 is used in minimum mode of operation.

4. Distinguish the LOCK and TEST signal?

A. LOCK signal will be active until the completion of next instruc

TEST : this examined by a 'WAIT' instruction. If the TEST pin goes low (0), execution will continue, else the processor remains in an idle state. The signal must be active high (1) for atleast four clock cycles.

5. Compose the term. multiprogramming?

A. Multiprogramming is a rudimentary form of parallel processing in which several programs are run at same time on a uniprocessor. Instead the operating system executes part of one program, then part of another, and so on. To the use it appears that all programs are executing at the same time.

part - II A.

5. Write the operations of I/O programming in detail?

A. I/O is one of the 3 main component of a computer system. The responsibility of I/O is to interface with external devices. Depending on their applications, it can be divided into 3 groups: sensory input, control input and data transfer.

Like memory components, I/O components have address and each I/O address usually consist of 8 or 16 bits of data. Since many sensory input and control output signals consist of just one bit of information, a single I/O address may be able to handle multiple input and output signals. Depending on how they are used, one bit of an I/O address may be referred to as an I/O port or set of all bits at a single I/O adress may be referred to as an I/O port.

1. Explain all the pin function of 8086 processor configured in maximum mode?

A. Maximum mode:-

- In thu we can connect more processors to 8086
- 8086 max mode is basically for implemention of allocation of global resources and passing bus control to other coprocessor.
- All processors execute their own program.
- Resources which are common to all processor are known as global resources.
- Resources which are allocated to a particular processor are known as local or private resources.
- when MN/MX' = 0, 8086 works in max mode.
- Clock is provided by 8284 clock generator.

part - III

1. Define the terms A/D and D/A converter?

A. Analog / Digital Converter :- A device that converts continuously varying analog signal from instruments and sensors that monitor conditions, such as sound, movement and temperature into binary code for the computer.

2. List the four display modes of 8279 keyboard and display controller.

A. The four major sections of 8279 are keyboard, scan, display and CPU interferance. Keyboard section: The keyboard section consist of eight return lines RL0 - RL7 that can be used to form the columns of keyboard matrix. It has two additional input: shift and Control / Strobe.

3. Name the applications of programmable interval timer?

A.
- To generate accurate time delay
- As an event counter
- Square wave generator.
- Rate generator.
- Digital one shot.

4. Outline the different peripheral interfacing used with 8086?

A. Static and dynamic memories, vector input table, interupt service routine, introduction to DOS and BIOS interupts, programmable interupt Controller 8259, DMA controller 8257 interfacing with 8086 ~~mode~~ microprocessor.

5. Write the various modes of 8254 timer?

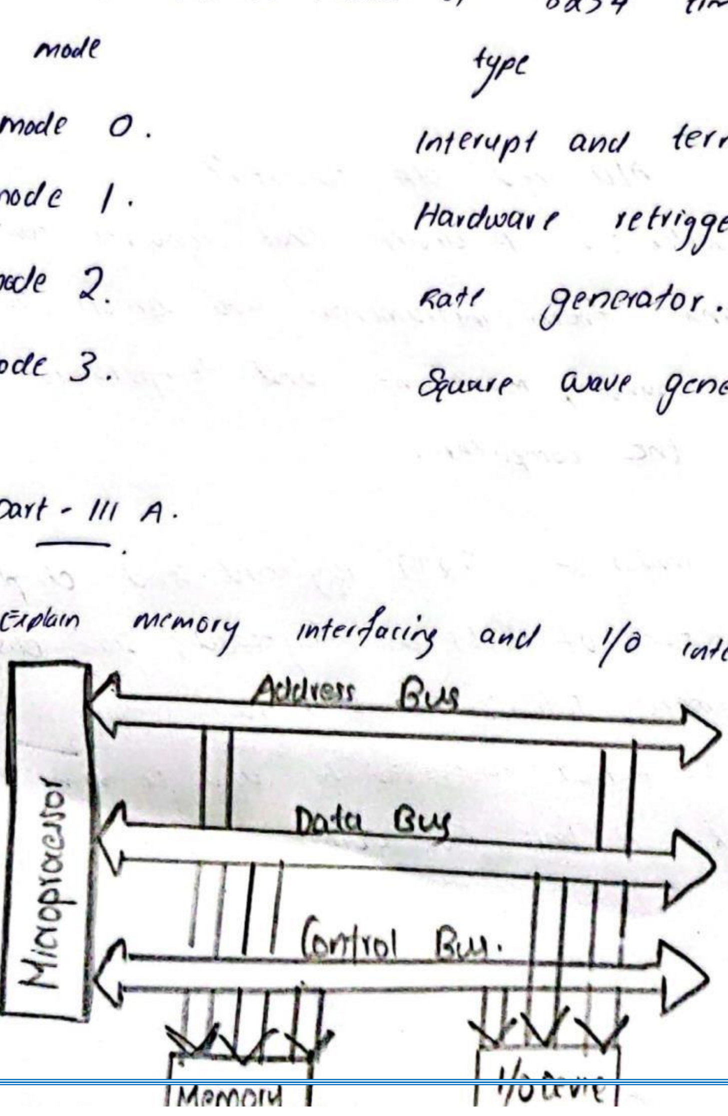
A.

mode	type
mode 0.	Interupt and terminal Count.
mode 1.	Hardware retriggerable one shot.
mode 2.	Rate generator.
mode 3.	Square wave generator.

part - III A.

4. Explain memory interfacing and I/O interfacing in detail?

A.



memory interfacing :-

when we are executing any instruction, the address of memory location or an I/o device is sent out by microprocessor. The corresponding memory chip or I/o device is selected by a decoding circuit. Memory require Some Signals to read from and write to registers and microprocessor transmits Some Signals for reading or writing data.

I/o interfacing :-

Keyboard and displays are used as communication channel with outside world. Therefore it is necessary that we interface keyboard and displays with microprocessor. This is called I/o interfacing. For this type of interfacing, we use latches and buffers for interfacing the keyboards and displays with the microprocessor.

part - IV

1. write the size of memory systems used in 8051 microcontroller?

A. The 8051 has a maxmum of only 256 bytes of internal data memory. this memory includes four register banks, , the 16 bytes of bit addressable memory, the Stack and variable memory. Bank Switching uses upto 4 bytes of the interal data memory for every call that Switches back.

2. Identify the different operand types used in 8051?

A. They are :-

-> Data transfer instructions

-> Arithmetic instructions

-> Logical instructions

-> Boolean or Bit manipulation instructions.

-> program Branching instructions .

3. List the counters available in 8051 ?

A. 8051 has two timers, Timer 0 and Timer 1. They can be used as timers or as event counter. Both Timer 0 and Timer 1 are 16 - bit wide. Since 8051 follows an 8-bit architecture, each 16-bit is accessed as two separate registers of low-byte and high-byte.

4. Label the register bank of 8051?

A. The 8051 microcontroller consist of four register banks, such as Bank 0, Bank 1, Bank 2, Bank 3 which are selected by the program Status word Register. These Register banks are present in the internal RAM memory of 8051 microcontroller and are used to process the data when microcontroller is programmed.

5. Name the numbers of ports, that are bit addressable in 8051?

A. 8051 microcontrollers have 4 I/O ports each of 8 bit, which can be configured as input or output. Hence total 32 input/output pins allow the microcontroller to be connected with the peripheral devices.

part IV-A

1. point out the various instruction sets of 8051 microcontroller and describe any two in detail?

A. The 8051 instruction set is optimized for 8-bit content applica
It has 111 instructions, they are

- 49 single byte instructions.
- 45 two byte instructions.
- 17 three byte instruction

The instruction Set is divided into four groups. they are :-

-> Data transfer Instructions.

-> Arithmetic instructions.

-> Logical instructions

-> Call and Jump instructions.

Arithmetic Instructions:-

operation,

ADD A, <Source byte>

this instruction adds the byte variable indicated to the accumulator. The result is contained in the accumulator.
All the addressing modes can be used for Source:- an immediate number, a register, direct address and indirect address.

Logical Instructions:-

This instruction perform bit wise logical AND operation between the destination and source byte. The result is Stored at destination byte. The Source and destination Support four addressing modes:- register, direct, register-indirect and immediate addressing modes.

3. Outline the I/O ports of 8051 microcontroller in detail?

A. pin can be configured as 1 for input and 0 for output as per logical State.

• Input/Output I/O pin:- All the circuits within the microcontroller must be connected to one of its pin except P0 port because it does not have pull-up resistors built-in.

Input Configuration:- If any pin of this port is configured as an input, then it acts as if it floats, ie, the input has unlimited input resistance and in-determined potential.

Output Configuration:.

When the pin is configured as an input, then it acts as a Open drain, By applying logic 0 to a port bit, the appropriate pin will be connected to ground (0 V); and applying logic 1, the external output will keep on floading.

part - V.

1. What are the type of sensors used for interfacing?

A. Some of the sensors used are:-

→ IR distance sensor.
→ humidity Sensor.
→ Temperature sensor.
→ Thermistor
→ Thermo couple.

2. Mention the advantages of microprocessor based systems design?

A.
- Compact size
- High speed
- Low power consumption
- portable.
- Very Reliable.

3. List out the classifications of stepper motor.

A. There are 3 main types of Stepper motor. They are:-

→ permanent magnet Stepper.
→ Hybrid Synchronous Stepper.
→ Variable reluctance Stepper.

4. How to change the stepper motor direction?

A. To change direction, we need to reverse the current in one Set of coils. At the moment, the two motors are Connected in parallel and turn in same direction. Swap one of coil pair over and we will be able to get two motors turn in opposite direction with same drive Signal.

5. Identify the features of serial port in Mode - 0?

A. Mode 0:-

- Only synchronocy mode.
- Data transferred on RXD clock on TXD
- Clock is fixed at 1/12 of oscillator frequency.

part - VA.