

Microcontrollers

One Mark Questions

Question 1.

What is a microcontroller?

Answer:

Microcontroller is a small computer on a single IC consisting of a relatively simple CPU combined with support functions.

Question 2.

What is an Accumulator?

Answer:

Accumulator is a device which stores a number and which on receipt of another number, adds the two and stores the sum. It can also sense the signal, clear and complement etc.

Question 3.

What is a Program Counter?

Answer:

Program counter is a circuit, which holds the address of a byte in memory. It also specifies the address of the next instruction to be fetched and executed.

Question 4.

What is a Stack pointer?

Answer:

Stack is the area of the memory (RAM) that is used in some instructions to store and retrieve data quickly. Stack pointer is used to hold an internal RAM address. When a program jumps to some branch the address of the next instruction is stored in Stack.

Question 5.

What is an interrupt circuit?

Answer:

It is an electronic circuit, which breaks in the normal flow of system or routine and flow can be resumed from that point at a later time.

Question 6.

What is the need of clock circuit?

Answer:

Clock circuit generates the internal clock pulses and all the internal operations are synchronised with this pulse.

Question 7.

How much is the ROM of microcontroller 8051?

Answer:

Microcontroller 8051 has on chip ROM of 4k bytes.

Question 8.

How much is the RAM of microcontroller 8051?

Answer:

Microcontroller 8051 has 128 bytes of RAM.

Question 9.

How many serial ports are there in microcontroller 8051?

Answer:

Microcontroller 8051 has one serial port for communication.

Question 10.

How many interrupt sources are there in microcontroller 8051?

Answer:

Microcontroller 8051 has six(6) interrupt sources.

Question 11.

What is an addressing mode?

Answer:

Addressing modes specify the way in which the operands are accessed by the instruction.

Question 12.

What is an Operational code (op code)?

Answer:

Operational code (op code) is the short form of expressing the instruction. It consists of mnemonic and operands.

Question 13.

What is the meaning of MOV A, R0?

Answer:

The content of R0 is moved (copied) to register A (accumulator).

Question 14.

What is the meaning of MOV R5, A?

Answer:

Copy the contents of register A(Accumulator) to Register 5.

Question 15.

What is the purpose of register addressing mode?

Answer:

In register addressing mode data is moved between registers A, DPTR, Carry bit and registers R0-R7 are used as part of opcode mnemonics as source or destination.

Question 16.

What is the use of direct addressing mode?

Answer:

In direct addressing mode 128 bytes of internal RAM and SFRs may be addressed directly using single byte address assigned to each RAM location and each SFR.

Question 17.

What is the direct address of working register R5 in register bank 2?

Answer:

The direct address of register R5 in register bank 2 is 10H.

Question 18.

Write the direct address of the Accumulator.

Answer:

The direct address of accumulator register is 00H.

Question 19.

Write the mnemonic for the operation "Save content of A in RAM location 45H".

Answer:

The mnemonic for saving content of register A in location 45H is MOV 45H, A.

Question 20.

What is meaning of MOV A, @R1?

Answer:

The meaning of MOV A, @R1 is to copy the contents of RAM location whose address is stored in R1 into accumulator A.

Question 21.

How do you write the mnemonic for the operation "copy the contents of RAM location whose address is stored in R0 into accumulator A" in an indirect addressing mode?

Answer:

The mnemonic for copy the contents of RAM location whose address is stored in R0 into accumulator A is MOV A, @ R0.

Question 22.

Which sign is used as the mnemonic for immediate data?

Answer:

The mnemonic for immediate data is hash (#).

Question 23.

What is meaning of MOV A, #22H?

Answer:

The meaning of MOV A, #22H is to load value 22H into accumulator A.

Question 24.

Write the mnemonic for the operation "load the decimal value 66 into R3".

Answer:

The mnemonic for loading decimal value 66 into R3 is MOV R3, #66.

Question 25.

What is meaning of MOV DPTR, #1246H?

Answer:

The meaning of MOV DPTR, #1246H is to load 16 bit number (1246H) immediately into register DPTR.

Question 26.

What does 'jump' instruction do?

Answer:

The jump instruction permanently changes the contents of program counter either conditionally or unconditionally.

Question 27.

What does 'call' instruction do?

Answer:

The call instruction temporarily changes the contents of program counter to allow another part of the program to run either conditionally or unconditionally.

Question 28.

What is the description of the mnemonic JNZ and ACALL?

Answer:

JNZ is conditional jump instruction, here contents of PC changes only when the result of operation is non zero. ACALL allows to specify 11-bit address in the instruction and calling subroutine within 2k program memory block, starting from the address of the next instruction is possible.

Question 29.

Why is ROM called non volatile?

Answer:

ROM is called non volatile because it does not depend on electrical power to store the numbers.

Question 30.

Why is RAM called volatile?

Answer:

RAM is called volatile because it depends on electrical power to store the numbers, when power is OFF it loses its information.

Question 31.

What is a program?

Answer:

Computer program is a sequence of operations that the computer is to perform.

Question 32.

Expand EEPROM.

Answer:

Electrical Erasable Programmable Read Only Memory.

Question 33.

Expand SRAM.

Answer:

Static Random Access Memory.

Question 34.

Expand RAM.

Answer:

Random Access Memory or Read write memory.

Question 35.

What is data address?

Answer:

Address in memory that is used by the CPU to read or write is called data address.

Question 36.

What is a machine language?

Answer:

Machine language is one in which information is available only in binary form.

Question 37.

Why is assembly language programming called low level programming?

Answer:

Assembly language is low level programming because, it is machine dependant.

Question 38.

How many bits of binary data can a register A hold?

Answer:

Register A can hold 8 bits of binary data.

Question 39.

How many bits of binary data can a register R hold temporarily?

Answer:

Register R can hold 8 bit of binary data temporarily.

Question 40.

How many bits of address can a register PC hold?

Answer:

Register PC can hold 16- bit of address number.

Question 41.

How much "PAL register PC" can hold code address?

Answer:

Practice Assembly Language (PAL) Register can hold code address from 0000H to FFFFH (64K).

Question 42.

What are directives?

Answer:

Assembly language instructions like ADD and MOV are statements called directives.

Question 43.

What does directive do?

Answer:

Instructions tell the computer what to do, while directives give directions to the assembler.

Question 44.

What is a PIC microcontroller?

Answer:

PIC microcontrollers are a family of microcontroller chips produced by microchip technology.

Question 45.

Expand PIC.

Answer:

PIC is Peripheral Interface Controller.

Question 46.

How many accumulators are there in PIC microcontroller?

Answer:

PIC microcontrollers has one accumulator.

Question 47.

Mention the maximum operating frequency of PIC16F877.

Answer:

Maximum operating frequency of PIC16F877 is 20MHz.

Question 48.

How many times data can be written in PIC16F877?

Answer:

In PIC16F877 microcontrollers 8k ROM memory in flash technology upto 105 times chip can be reprogrammed.

Question 49.

What is the working voltage range of PIC16F877?

Answer:

PIC microcontrollers has working voltage range of 2.0V to 5.5V.

Question 50.

How much is the EEPROM memory?

Answer:

EEPROM memory has 128 bytes.

Question 51.

What is the largest hex value that can be moved into an 8 bit register?

Answer:

FFH is the largest value that can be moved into an 8 bit register.

Two Marks Questions**Question 1.**

Mention two differences between microprocessor and microcontroller.

Answer:

Microprocessor	Microcontroller
Most MPs have very less bit handling instructions	MCs have many bit handling instructions

MPs are intended for general purpose applications	MCs are for a specific application only
---	---

Question 2.

What is an addressing mode? Why is it necessary?

Answer:

The CPU can store data may be in registers, memory or in an external source. The ways by which these data source address are specified are called addressing modes. These are necessary because, the data can be accessed using any one of the modes.

Question 3.

Mention the different opcodes used in 8051.

Answer:

MOV dest, sour; PUSH sour; POP dest; XCH dest, sour; ADD A, Rn etc are some of the opcodes used in 8051.

Question 4.

What are the different parts of memory of 8051?

Ans:

The memory is divided into Internal RAM, Internal SFR, External RAM, internal and External ROM.

Question 5.

What is the main feature of MOV opcode? Give an example.

Answer:

MOV opcode transfers data within the 8051 memory. MOV A, R1; MOV R0, 22H; MOV A, #54H; MOV R2, #40.

Question 6.

What is direct addressing? Explain.

Answer:

In direct addressing mode all 128 bytes of internal RAM and SFRs may be addressed directly using single-byte address assigned to each RAM and SFR registers. MOV R0, 40H save the content of location 40H in R0 register. MOV B, R1 means copy the content of R1 to B.

Question 7.

What is an assembly language?

Answer:

Assembly language fills the gap between low level machine language and high level language. It is written in instruction mnemonics.

Question 8.

Write ALP to represent "load 15H into R2"

Answer:

MOV R2,#15H

Question 9.

Write the summary of unsigned multipliers

Answer:

Multiplication operation use registers A and B as both source and destination addresses for the operation. MUL AB multiplies contents of A and B.

Multiplication	Operand 1	Operand 2	Result
Byte x Byte	A	B	A = lower byte B = High byte

Question 10.

Write the summary of division of unsigned numbers.

Answer:

In the division of unsigned numbers 8051 supports byte over byte only. DIV AB, after division quotient is in A and remainder in B

Division	Numerator	Denominator	Quotient	Remainder
Byte/Byte	A	B	A	B

Question 11.

Mention any two core features of PIC microcontroller.

Answer:

Instruction set simplicity and programmable timer options.

Question 12.

Mention the main features of PIC microcontroller.

Answer:

Operating frequency-0-20MHz, Power supply voltage 2.0 to 5.5V, power saving sleep mode, ADC, Watch dog timer, US ART module etc.

Question 13.

Mention the power supply and power consumption of PIC16F877.

Answer:

Power supply 2.0V to 5.5V and power consumption is 220mA (2.0V, 4MHz), 11mA(2.0V. 32kHz), 50nA standby mode.

Question 14.

Briefly explain Data transfer instruction.

Answer:

In data transfer instructions, data is moved(copied) from the source to destination. Most of the instructions in this group does not affect the PSW flags. MOV, PUSH, POP and XCH are the commonly used instructions in this group.

Question 15.

Briefly explain Arithmetic instruction.

Answer:

Various arithmetic operations are carried out in this group of instructions. ADD, SUB, INC, DEC, MUL, DIV are the basic operations used.

Question 16.

Briefly explain Logical instructions.

Answer:

Logical operations like AND, OR, XOR, complement, rotate are performed by these instructions. One of the operand is always register A.

Question 17.

Briefly explain programming instructions.

Answer:

These are also called branching instructions. Conditional and unconditional branching instructions are used. JMP, AJMP, LJMP, CALL, ACALL are some examples.

Question 18.

What is the function of register A?

Answer:

The register A is used to store the data temporarily and also to hold the result of arithmetic and logical instructions.

Question 19.

What is the function of register R?

Answer:

Register R stores numbers temporarily.

Question 20.

What is the function of register PC?

Answer:

It is a 16 bit register. Program instruction bytes are fetched from locations in memory that are addressed by PC. The PC is automatically incremented after every instruction byte is fetched and may also be altered by certain instructions.

Question 21.

Briefly explain about structure of Assembly Language.

Answer:

Assembly language is a series of lines of assembly language instructions consisting of mnemonics, one or two operands. The four fields of assembly language are: label; mnemonic; operand; comment.

Three Marks Questions**Question 1.**

Mention the reason to write computer instructions in assembly language.

Answer:

- a. To speed computer operation. Programs written in assembly language can be stored compactly, and less time is spent fetching the code. High level languages are converted to code by utility programs named compilers. Because of general nature of high level languages, the compilers often produce excess or over head-code.
- b. To reduce the size of the program. Assembly language requires no extra overhead code because the assembly language programmer is aware of extra needs of the program for any given situation.
- c. To write programs for special situations. Often, particularly when dealing with machine control, no standard programs exist. Robot arms and antilock brakes, for instance, have no standard drivers. It is generally more efficient to write nonstandard driver programs in assembly code, also, when speed of response is critical, assembly-coded programs execute rapidly because of the exact fit of program code to task requirements.
- d. To make economical. Small computer systems such as those that are embedded inside other machines, are often produced in large numbers. Reducing code size also reduces the cost of associated ROM chips.
- e. To better understand how computer operates. In order to fully understand what is going on “under the hood” of the CPU, you should learn to program the CPU in assembly language.

Question 2.

Mention the different programs required ALPs converted into code memory bit charges,

Answer:

The different programs required ALPs converted into code memory bit charges are:

- a. An operating system program, which controls the operation of the personal

computer used for the entire programming process. DOS is the operating system program used by many PCs. Windows is another.

b. A word processing, often called a text editor program. Programs in assembly language mnemonics are written using the text editor and stored on a disk as files that, normally, end in the extension asm. The asm files are intended for the use of another program called the assembler programs. Any text editor that can produce an ASCII (text) file is suitable for writing assembly language, asm files.

c. An assembler program, which takes the. asm assembly language program file and converts it to a machine code, obj file. The assembler converts the ASCII mnemonic text file into an. obj file that contains machine code instructions to the CPU, in binary form.

d. A testing program, which lets you run and test your program under controlled conditions. Testing your program is the most important step of the programming process. To be able to test program one must be able to execute each instruction and see the results. Utility programs that allow the user to test programs are called debuggers or simulators.

Question 3.

Briefly explain the different bits of binary memories of different registers.

Answer:

Figure shows the different bits of binary memories of different registers.

7F	Scratch PAD RAM
30	
2F	Bit addressable RAM
20	
1F	R3
18	
17	R2
10	
0F	R1
08	
07	R0
00	

Question 4.

What are the different fields present in ALP (PAL – Practice Assembly language)?

Answer:

PAL contains three internal registers:

Register A: Performs all operations in the CPU Register B: Stores numbers temporarily

Register PC: Holds the address of the next instruction to be executed in code memory. Register A and R can hold 8 bits of binary data and register PC can hold a 16-bit address number. PAL is called an 8-bit computer because the working registers A and R can hold a 1-byte number. PAL register PC limits the number of code byte addresses to 64k because it can hold code address from address 0000H to address FFFFH.

Question 5.

Write ALP to represent load 15H into R2

Answer:

MOV R2, #15h

Question 6.

Write the description of ALP ADDA, R2

Answer:

The content of register R2 is added with the content of accumulator and the the sum is stored in the accumulator.

Question 7.

Mention the different types of addressing modes.

Answer:

There are four addressing modes in 8051

- Immediate addressing mode
- Register addressing mode
- Direct addressing mode
- Indirect addressing mode

Question 8.

Briefly explain data transfer instructions.

Answer:

These instructions move the content of one register to another one. Data can also be transferred to stack with the help of PUSH and POP instructions.

MNEMONIC	BYTES
MOV A,Rn	1
MOV A,Rx	2
MOV A,@Ri	1
MOV A,#X	2
MOV Rn,A	1
MOV Rn, Rx	2
MOV Rn, #X	2
MOV Rx, A	2
MOV Rx, Rn	2 etc.

Question 9.

Briefly explain arithmetic instructions.

Answer:

These instructions perform several basic arithmetic operations. After execution, the result is stored in the first operand. 8 bit addition, subtraction, multiplication, increment-decrement instructions can be performed in 8051.

Example:

ADDA, Rn

ADDA, #8 bit data

ADDC A, Direct

ADDC A, Rn

ADDC A, @Ri

ADDC A, #8 bit data

SUBB A, Rn

SUBB A, Direct

SUBB A, @Ri

SUBB A, #8 bit data

INC A

INC Rn

INC Direct

INC @Ri

DEC A

DEC Rn

DEC Direct

DEC @Ri

INC DPTR

MUL A B

DIV A B

DA A

Question 10.

Briefly explain logical instructions.

Answer:

These instructions perform logical operations between two register contents on a bit by bit' basis. After execution, the result is stored in the first operand.

Example:

ANL A, Rn

ANL A, Direct

ANL A, @Ri

ANL A, #8 bit data

ANL Direct, A

ANL Direct, #8 bit data

ORLA, Rn

QRL A, Direct

ORLA, @Ri

ORL A, #8 bit data

ORL Direct, A

ORL Direct, #8 bit data

XRL A, Rn

XRL A, Direct

XRL A, @Ri

XRL A, #8 bit data

XRL Direct, A

XRL Direct, #8 bit data

CLR A

CPLA

RLA

RLC A

RR A

RRC A

SWAP A

Question 11.

Briefly explain programming instructions.

Answer:

Assembly language programming consists of writing programs using instruction mnemonics that are specified by the manufacturer. An instruction consists of a label, an instruction mnemonic, operands and operational comments. The instruction mnemonic specifies the action to be taken by the CPU and the operands specify the addresses of data used in the action. Labels are the names given to the address numbers in program code memory.

Question 12.

Mention three key features,

- i. Max operating frequency
- ii. Flash programming memory
- iii. No. of pins of 16F877.

Answer:

i. Operating speed:

- DC – 20 MHz oscillator/clock input
- DC – 200 ns instruction cycle

ii. Flash memory size:

- 100,000 write Flash endurance
- 1,000,000 write EEPROM endurance
- Flash/Data EEPROM retention: > 40 years

iii. No. of pins of 16F877: 28 pins

Question 13.

Mention the value of ROM, RAM and EEPROM of PIC16F877.

Answer:

- 100,000 write Flash endurance
- 1,000,000 write EEPROM endurance
- Flash/Data EEPROM retention: > 40 years

Five Marks Questions**Question 1.**

Compare a microprocessor with a microcontroller

Answer:

Microprocessor	Microcontroller
μ P have many operational codes for moving data and performing operations, but it has only few bit-handling instructions.	μ C have one or two operational codes for moving data from and performing operations, but it has many bit- handling instructions.
μ P is concerned with rapid movement of code and data from external addresses to the chip.	μ C is concerned with rapid movement of bits within the chip.

μP need external digital parts (Hardware) to perform its tasks as computer.	μC can function as computer without addition of external digital parts.
μP are used in general purpose computing devices, supports wide range of applications with general hardware.	μC are used in embedded systems and supports only dedicated applications with customized hardware.

Question 2.

Mention the features of 8051.

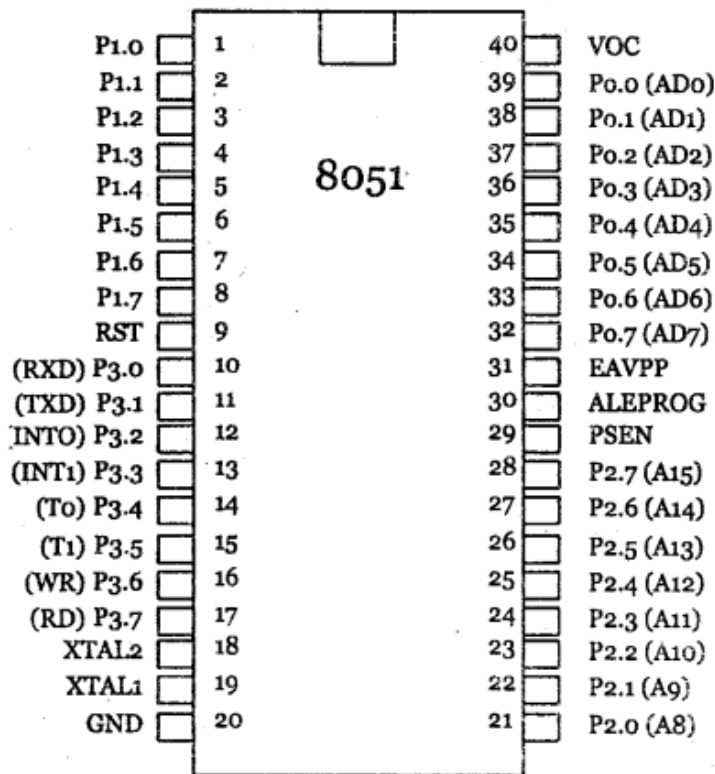
Answer:

1. Eight-bit CPU with registers A (Accumulator) and B.
2. Sixteen-bit program counter (PC) and data pointer(DPTR)
3. Eight-bit program status word (PSW)
4. Eight-bit stack pointer(SP)
5. Internal ROM or EPROM (8751) of 4k bytes
6. Internal RAM of 128 bytes :
 - Four register banks, each containing eight registers
 - Sixteen bytes, which may be addressed at the bit level
 - Eight bytes of general purpose registers
7. Four 8-bit I/O ports (P0-P3)
8. Two 16-bit timer/counter T0 and T1
9. Full duplex serial data receiver/ transmitter : SBUF
10. Control register: TCON, TMOD,SCON,PCON,IP and IE
11. Two external and two internal interrupt sources
12. Oscillator and clock circuits

Question 3.

Draw the pin diagram of 8051

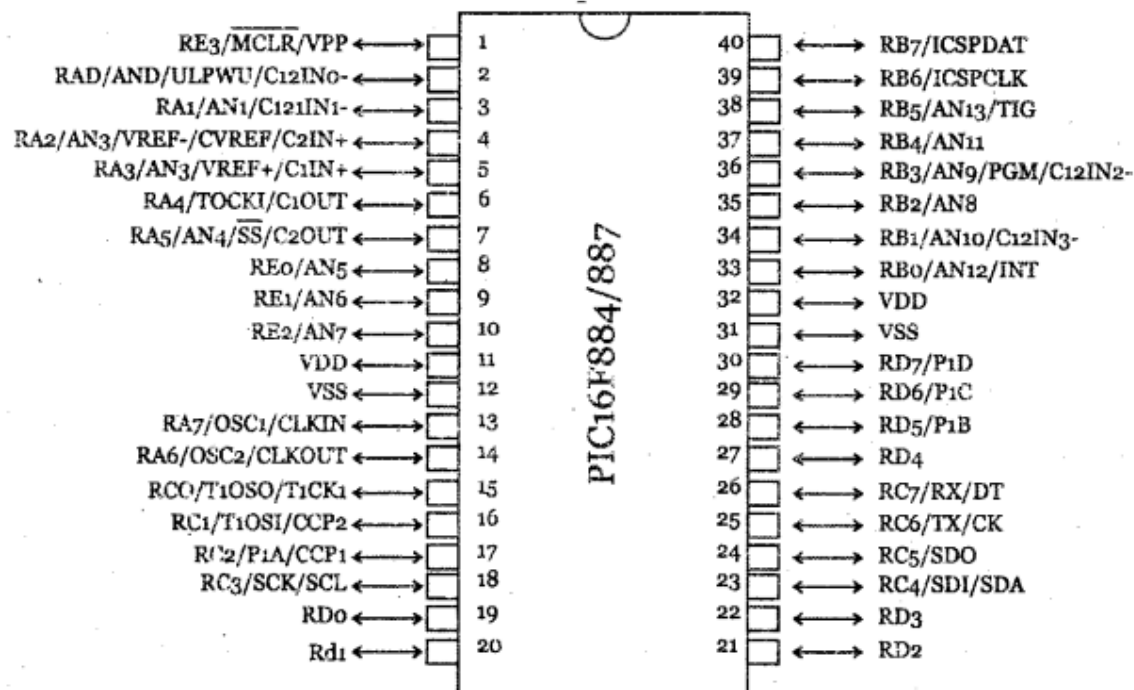
Answer:



Question 4.

Briefly explain the core features of pic microcontroller(PIC 16F887)

Answer:



Question 5.

Briefly explain the steps used in creating a program.

Answer:

The steps to create a program are:

- Program editor is used to type in the program. The editor must be able to produce ASCII file for many assembler and will have the extension “.asm” or “.src”.
- Assembler on receiving the file in “.asm” form converts the instructions into steps to create a program machine code. Assembler will produce an object file “.obj” and list file “.lst”.
- Linker program links one or more object files and produce an absolute object file “.abs”.
- The “.abs” file is fed to object to hex converter(OH) and create an extension “.hex” and is ready to burn ROM.

Question 6.

Briefly explain signed and unsigned subprogram.

Answer:

Subtraction of unsigned numbers: In 8051 SUBB is used to subtract two numbers by using unsigned representation by setting the bit CY=0 or CY=1. Whereas, in signed representation the sum or difference should not be greater than or less than -128 or 127.

Question 7.

Explain the core features of PIC16F887 microcontroller.

Answer:

24/35 I/O pins with individual direction control:

- High current source/sink for direct LED drive
- Interrupt-on-Change pin
- Individually programmable weak pull-ups
- Ultra Low-Power Wake-up (ULPWU)

Analog Comparator module with:

- Two analog comparators
- Programmable on-chip voltage reference (CVREF) module (% of VDD)
- Fixed voltage reference (0.6V)
- Comparator inputs and outputs externally accessible
- SR Latch mode
- External Timer 1 Gate (count enable)

A/D Converter:

- 10-bit resolution and 11/14 channels
- Timer 0: 8-bit timer/counter with 8-bit programmable prescaler

Enhanced Timer 1:

- 16-bit timer/counter with prescaler
- External Gate Input mode
- Dedicated low-power 32 kHz oscillator

Timer2: 8-bit timer/counter with 8-bit period register, prescaler and postscaler

Enhanced Capture, Compare, PWM+ module:

- 16-bit Capture, max. resolution 12.5 ns
- Compare, max. resolution 200 ns
- 10-bit PWM with 1, 2 or 4 output channels, programmable “dead time”, max. frequency 20 kHz
- PWM output steering control

Capture, Compare, PWM module:

- 16-bit Capture, max. resolution 12.5 ns
- 16-bit Compare, max. resolution 200 ns
- 10-bit PWM, max. frequency 20 kHz

Enhanced USART module:

- Supports RS-485, RS-232, and LIN 2.0
- Auto-Baud Detect
- Auto-Wake-Up on Start bit
- In-Circuit Serial Programming™ (ICSP™) via two pins
- Master Synchronous Serial Port (MSSP) module – supporting 3-wire SPI (all 4 modes) and 12C™ Master and Slave Modes with I2C address mask

Question 8.

Briefly explain different addressing modes in the 8051.

Answer:

There are four addressing modes in 8051.

- Immediate addressing mode: the data source is available immediately as a part of instruction itself. Example: MOV A, #8bit data
- Register addressing mode: the source and destination register names are parts of the opcode in instruction. Example: MOV A, R0
- Direct addressing mode: the source and/or destination may be the internal RAM locations. Example: MOV 90h, #50h
- Indirect addressing mode: the source or destination address may be indicated the content of index register. Example: MOV @R1, #40h

Problems with Solutions

Question 1.

Write the instructions to move value 34H into register A and value 3FH into register B, then add them together.

Answer:

```
MOV A, #34h
MOV 0F0, #3Fh
ADD A, 0F0
LCALL 0003h
```

Question 2.

Write the instructions to add the values 16H and CDH, place the result in register R2.

Answer:

```
MOV A, #16h
MOV R2, #CDh
ADD A, R2
MOV R2,A
LCALL 0003 h
```

Question 3.

Add 25H and 34H and put the result in register A.

Answer:

```
MOV A, #25h
ADDA, #34h
LCALL 0003
```

Question 4.

Write a program to add two 8-bit numbers and store it in R6. The numbers are 01EH and 01CH.

Answer:

```
MOV A, #1Eh
ADD A, #1Ch
MOV R6, A
LCALL 0003h
```

Question 5.

Subtract 21H from 30H and write the program and solve.

Answer:

```
MOV A, #30h
```

SUBB A, #21h
LCALL 0003h