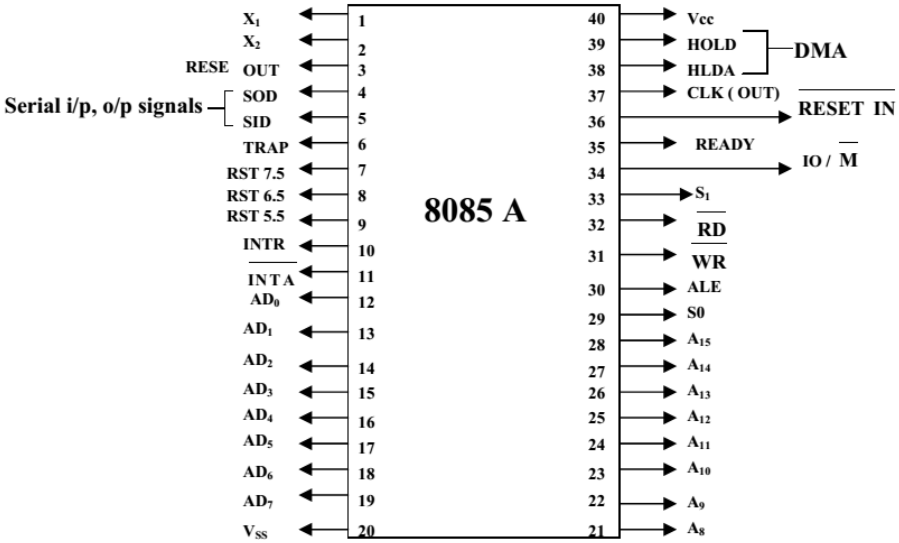
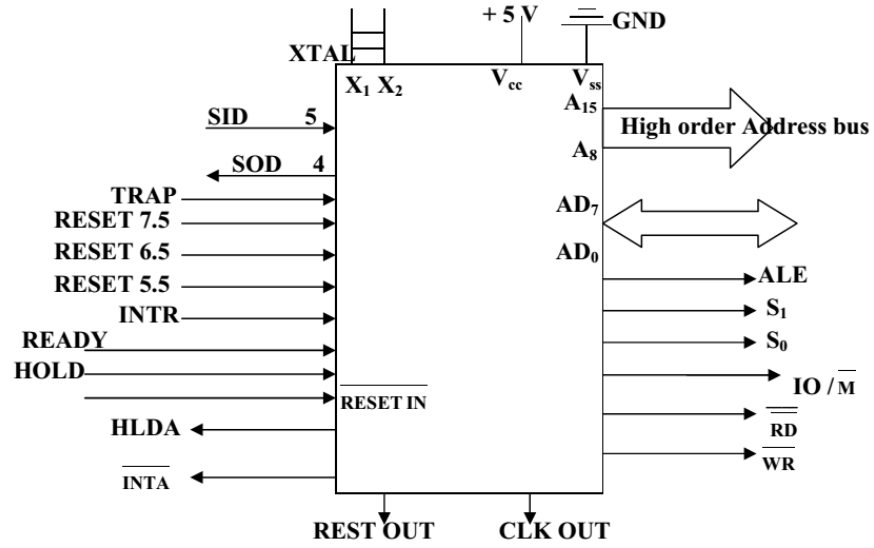
Q.2 a. Draw pin diagram and signal group diagram of 8085 microprocessor. (8)

Answer:



Pin Diagram of 8085

Signal Groups of 8085



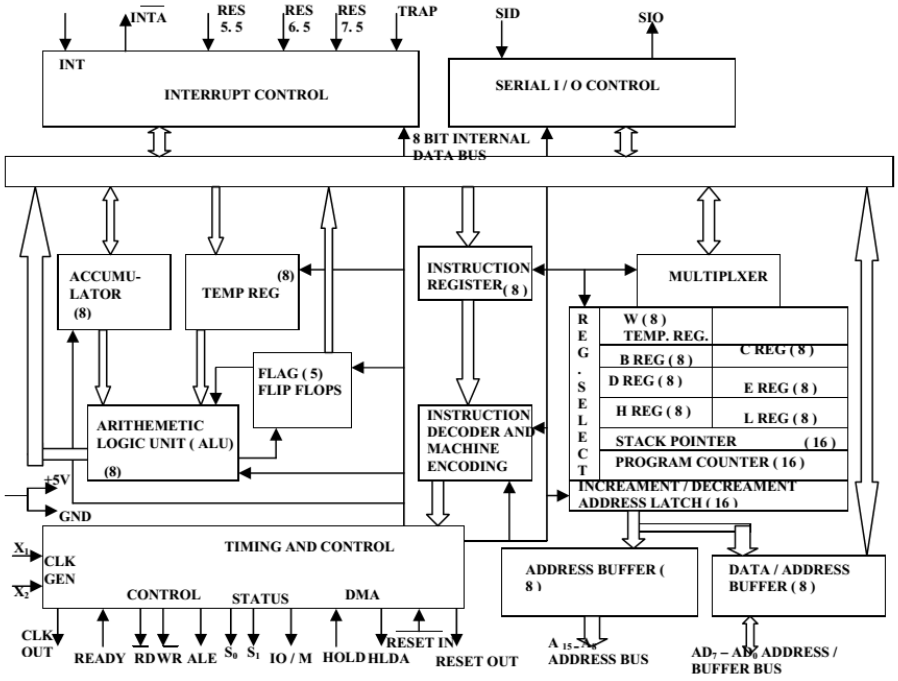
b. List out the various categories of the 8085 instructions. Give examples of the instructions for each group. (8)

Answer:

- Data transfer instructions – MOV, MVI, LXI.
- Arithmetic instructions – ADD, SUB, INR.
- Logical instructions –ANA, XRA, CMP.
- Branch instruction – JMP, JNZ, CALL.
- Stack I/O and Machine control instructions – PUSH, POP, IN, HLT.

Q.3 a. Draw the functional block diagram of 8085 microprocessor. (8)

Answer:



Block Diagram

b. What is the need for input output ports in microcomputer systems? Discuss merits and demerits of i/o mapped with respect to memory mapped i/o in 8085. (8)

Answer:

Need for input output ports in microcomputer systems:

CPU and main memory are very fast as compared to mechanical input /output device. In such cases it is essential that data lines of the computer are not kept engaged for along time during communication with input/output device as the overall speed of the computer system will drastically get reduced so there is need of input/output ports in microcomputer system.

Merits and demerits of input-output mapped with respect to memory mapped input-output in 8085-

• In I/O mapped scheme I/O devices are treated as I/O and memory as memory whereas in memory mapped I/O scheme both I/O and memory are treated as memory.

• Only In and OUT instructions are used for addressing I/O mapped I/O ports but large number of instructions can be used for communication with a memory mapped I/O. All instructions used to access memory can be used to access memory mapped I/O.

• Using I/O mapped I/O only accumulator can communicate with an I/O port. With memory mapped I/O any register can communicate with memory mapped I/O port.

• In memory mapped I/O device address is 20 bits whereas in I/O mapped it is 8/16 bits.

• For I/O mapped total devices that can be shared with 1MB memory is 65535 input and 65535 output devices whereas for 1 MB memory total devices will be A MB as its is shared between memory and I/O.
• Only data transfer operations are possible with direct data from I/O devices whereas in memory mapped I/O both arithmetic/logical operations are also possible on direct data from I/O device.
• In I/O mapped scheme decoding of 8/16 bits address is only required so hardware is less whereas in memory mapped I/O scheme decoding of 20 bits is needed so hardware is more.
• The control signals used in I/O mapped scheme are IOR & IOW whereas in memory mapped I/O it is MEMR and MEMW.

Q.4 a. Write a program to convert a binary number stored at LOC X to its BCD equivalent and display it in the data/addr field. (Assume display subroutine is given to you) (8)

Answer:

PROGRAM	ALGORITHM

START: LDA F100H
MOV B,A
MVI D, 64H
CALL BCD
MOV H,C
MVI D,0AH
CALL BCD
MOV A,C
RLC
RLC
RLC
RLC
ORA B
MOV L,A
CALL UPDAD
HLT
BCD: MVI C,00H
MOV A,B
RPTS: SUB D
JC NC
INR C
JMP RPTS
NC: ADD D
MOV B,A
RET
	STEP 1: Load the number to be converted
	STEP 2: On the basis of successive subtraction find the Co-efficient in BCD form.
	STEP 3: Display the result in Address Field.
	STEP4: Halt the processor

b. Write a program to find the largest number in an array of data using 8085 instruction set. (8)

Answer:

```
LXI H,4200          Set Pointer For Array
    MOV B,M         LOAD THE COUNT
    INX H
    MOV A,M         SET 1ST ELEMENT AS LARGEST DATA
    DCR B           DECREMENT THE COUNT
LOOP: INX H
    CMP M           IF A-REG >M GE TO AHEAD
    JNC AHEAD
    MOV A,M         SET THE NEW VALUE AS LARGEST
AHEAD: DCR B
    JNZ LOOP        REPEAT COMPARISON TILL COUNT =0
    STA 4300        STORE THE LARGEST VALUE AT 4300
    HLT
```

Q.5 a. Discuss the various types of interrupts in the order of lowest to highest priority. (8)

Answer:

Interrupts

- The processor has 5 interrupts. They are presented below in the order of their priority (from lowest to highest):
-
- **INTR** is maskable 8080A compatible interrupt. When the interrupt occurs the processor fetches from the bus one instruction, usually one of these instructions:
- One of the 8 RST instructions (RST_0 - RST_7). The processor saves current program counter into stack and branches to memory location N * 8 (where N is a 3-bit number from 0 to 7 supplied with the RST instruction).
- **CALL** instruction (3 byte instruction). The processor calls the subroutine, addres of which is specified in the second and third bytes of the instruction.
- **RST5.5** is a maskable interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 2CH (hexadecimal) address.
- **RST6.5** is a maskable interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 34H (hexadecimal) address.
- **RST7.5** is a maskable interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 3CH (hexadecimal) address.
- **TRAP** is a non-maskable interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 24H (hexadecimal) address.
- All maskable interrupts can be enabled or disabled using EI and DI instructions. RST 5.5, RST6.5 and RST7.5 interrupts can be enabled or disabled individually using SIM instruction.

b. Explain the various operational modes of 8255 programmable peripheral interface? (8)

Answer:

Operation Description of 8255 peripheral interface:

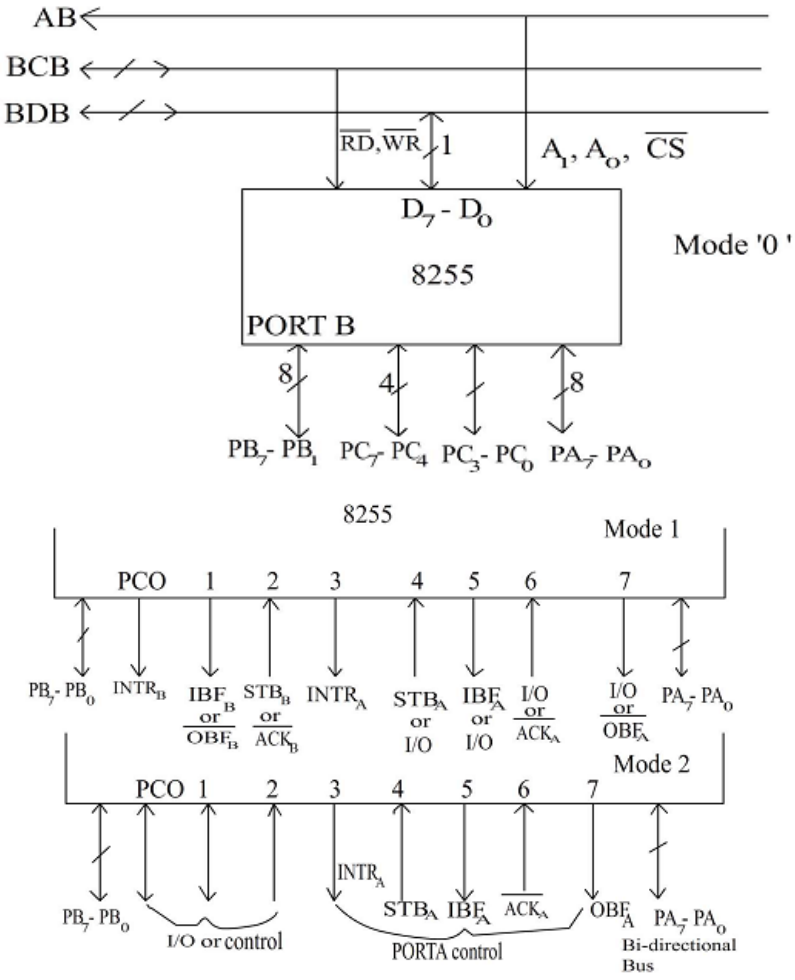
There are three operational modes that can be selected by the system software.

Mode 0: Basic Input/output
Mode 1: Strobes Input/output
Mode 2: Bi-direction bus.

When the reset input goes HIGH all poets are set to mode'0' as input which means all 24 lines are in high impedance state and can be used as normal input. After the reset is removed the 8255A remains in the input mode with no additional initialization. During the execution of the program any of the other modes may be selected using a single output instruction.

The modes for PORT A & PORT B can be separately defined, while PORT C is divided into two portions as required by the PORT A and PORT B definitions. The ports are thus divided into two groups Group A & Group B. All the output register, including the status flip-flop will be reset whenever the mode is changed. Modes of the two group may be combined for any desired I/O operation e.g. Group A in mode '1' and group B in mode '0'.

The basic mode definitions with bus interface and the mode definition format are given in figures below



Q.6 a. Write an 8085 assembly program to evaluate two 4-variables Boolean expression using logic controller interface. (8)

Answer:

21.1.1 EVALUATION OF BOOLEAN EXPRESSION

Write an 8085 assembly language program to evaluate two 4-variable Boolean expressions, using logic controller interface.

Let us say we want to evaluate the following Boolean expressions.

$$X = P\,Q\,\bar{R}\,S + P\,\bar{Q}\,R\,\bar{S} + \bar{P}\,\bar{S} \text{ and } Y = P\,\bar{Q}\,\bar{R}\,\bar{S} + P\,\bar{R}\,S$$

First of all, truth table for the Boolean expressions is written down as shown in the following table.

P	*Q*	*R*	*S*	*X*	*Y*
0	0	0	0	1	0
0	0	0	1	0	0
0	0	1	0	1	0
0	0	1	1	0	0
0	1	0	0	1	0
0	1	0	1	0	0
0	1	1	0	1	0
0	1	1	1	0	0
1	0	0	0	0	1
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	0	0
1	1	0	0	0	0
1	1	0	1	1	1
1	1	1	0	0	0
1	1	1	1	0	0

PQRS inputs are connected to PB3, PB2, PB1, and PB0 of 8255 respectively. X and Y outputs are connected to PA1 and PA0, respectively. In the program that follows, the user is required to keep changing the PQRS inputs manually. The program will be in a loop outputting the X and Y values (displayed using LEDs) corresponding to the PQRS inputs.

The truth table is stored in memory as a look up table, say from location C100H. The following values (corresponding to X and Y in the LS bit positions) are stored in consecutive locations starting from C100H.

02, 00, 02, 00, 02, 00, 02, 00, 01, 01, 02, 00, 00, 03, 00, 00

Program

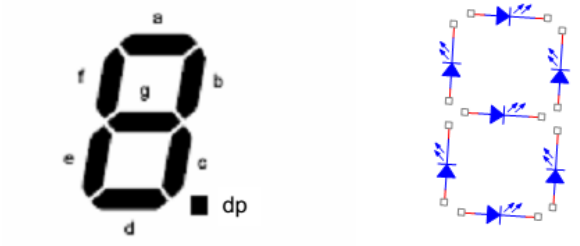
```
;FILE NAME BOOLEAN.ASM

      ORG C100H
TABLE DB 02H, 00H, 02H, 00H, 02H, 00H, 02H, 00H
      DB 01H, 01H, 02H, 00H, 00H, 03H, 00H, 00H

      ORG C000H

PA    EQU D8H
PB    EQU D9H
PC    EQU DAH
CTRL  EQU DBH

      MVI A, 10001010B
      OUT CTRL      ;Configure 8255 ports

LOOP: IN PB
      ANI 0FH       ;Now A will contain PQRS input value

      LXI H, TABLE
      ADD L
      MOV L, A      ;Point HL to proper row in the truth table

      MOV A, M
      OUT PA        ;Output XY from truth table to display
      JMP LOOP
```

Programs Using Interface Modules

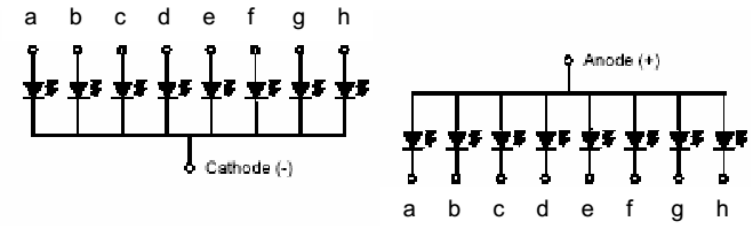
b. Explain the interfacing of 8085 microprocessor with seven segment display. (8)

Answer:

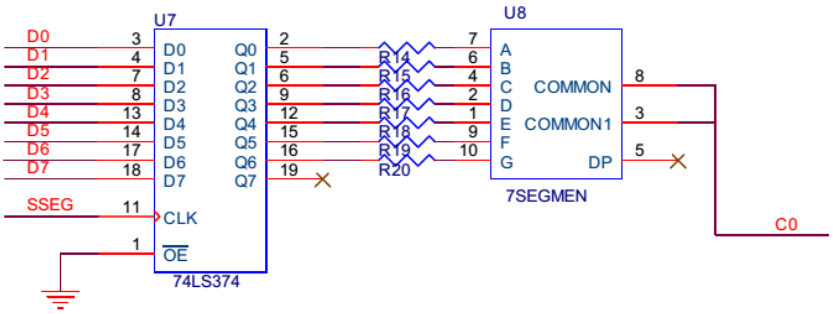
Interfacing of the 8085 Microprocessor System with seven segment display is done through its programmable I/O port 8255. Seven segment displays is often used in the digital electronic equipments to display information regarding certain process.



There are two types of seven segment display; common anode and common cathode. The differences between these two displays are shown in Figure. The internal structure of the seven segment display consists of a group of Light Emitting Diode (LED).



For common cathode, the segment will light up when logic '1' (+V) is supplied and it will light off when logic '0' (OV) is supplied. While for common anode, logic '1' will light off the segment and logic '0' will light up the segment. Therefore to display number '0' on the seven segment display, segment a, b, c, d, e and f must light up. For common cathode, logic '1' should be given to the related segment whereas in the case of common anode, logic '0' should be given to the necessary segment. The following diagram shows the interface for common cathode 7 segments in the I/O board. The 7 segments is connected through 8 bit latch (74LS374) and a resistor (range from 100Ωto 220Ω) in series to each segment.



Q.7 a. What is the benefit of INTEL 8259A- programmable interrupt controller? Name the various operation command words to operate 8259A in various interrupt modes. (8)

Answer:

INTEL 8259A Programmable Interrupt Controller

The 8259A is a programmable interrupt controller designed to work with Intel microprocessor 8080 A, 8085, 8086, 8088. The 8259 A interrupt controller can

1) Handle eight interrupt inputs. This is equivalent to providing eight interrupt pins on the processor in place of one INTR/INT pin.
2) Vector an interrupt request anywhere in the memory map. However, all the eight interrupt are spaced at the interval of either four or eight location. This eliminates the major drawback, 8085 interrupt, in which all interrupts are vectored to memory location on page 00H.
3) Resolve eight levels of interrupt priorities in a variety of modes.
4) Mask each interrupt request individually.
5) Read the status of pending interrupts, in service interrupts, and masked interrupts.
6) Be set up to accept either the level triggered or edge triggered interrupt request.
7) Mine 8259 as can be cascade in a master slave configuration to handle 64 interrupt inputs.
The 8259 A is contained in a 28-element in line package that requires only a compatible with 8259. The main difference between the two is that the 8259 A can be used with Intel 8086/8088 processor. I

Operation Command Words (OCWs): These are the command words which command the 8259A to operate in various interrupt modes. These modes are:

a. Fully nested mode
b. Rotating priority mode
c. Special mask mode
d. Polled mode

b. Discuss the function of following pins in an 8257 DMA controller
(i) IOW (ii) HRQ (iii) HLDA (iv) AEN (8)

Answer:

Functions of pins in 8257DMA Controller:

i. **IOW** -It's an active low pin that is activated by processor to write to an ARC, CR or the control register when 8257 is in slave mode.

ii. **HRQ**- stands for hold request. Active high output connected to HOLD input of 8085.whenever DRQ is active the corresponding DMA channel is enabled and HRQ is activated by 8257.to request processor for granting necessary control of the system bus.

iii. **HLDA**- Stands for HOLD acknowledge. Active high input pin which is connected to HLDA output of 8085.When HLDA becomes active it means that the processor has gone to HOLD state and 8257 is the master of the microcomputer system.

iv**. AEN-**Address enable- Active high output pin .8257 outputs 0 on this pin when 8085 is master of the system. When 8085 goes into HOLD state 8257 gives 1 on this pin indicating that it is mater now.

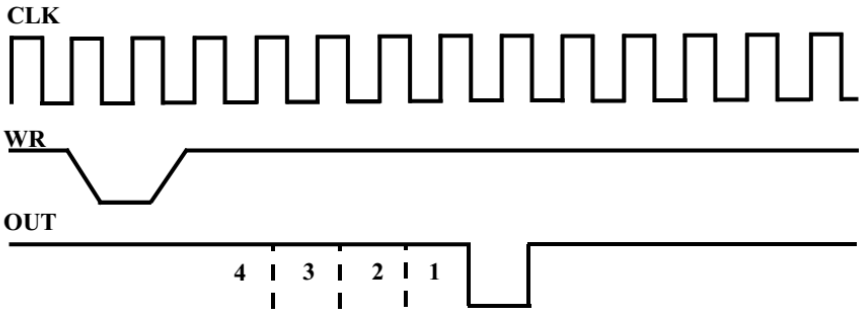
Q.8 a. Explain Mode 4 and Mode 5 operation of 8253 timer? (8)

Answer:

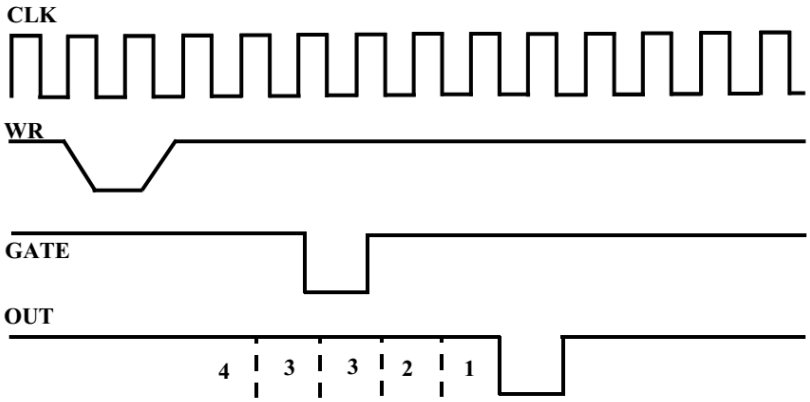
Mode 4 Software Triggered Strobe

In this mode after the count is loaded by the processor the countdown starts. The output goes low for one clock period after the countdown is complete. The countdown can be suspended by making the GATE low .This is also called a software triggered strobe as the countdown is initiated by a program.

GATE is HIGH

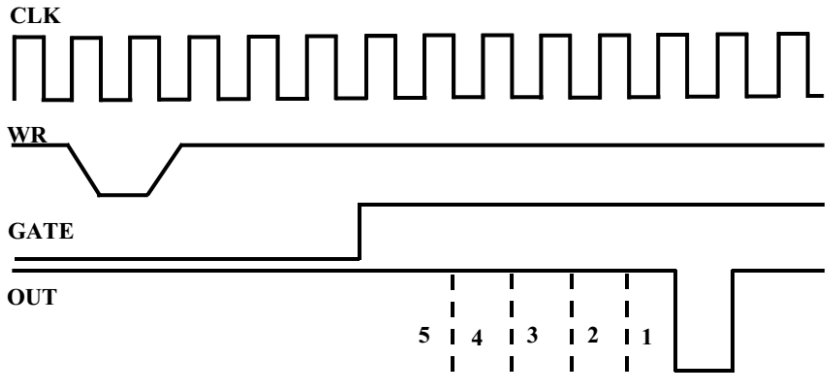


GATE is momentarily LOW



Mode 5 Hardware Triggered Strobe

The count is loaded by the processor but the countdown is initiated by the GATE pulse. The transition from low to high of the GATE pulse enables count down. The output goes low for one clock period after the countdown is complete.



b. Draw and explain wave form on TXD & RXD in synchronous transmission & Reception. (8)

Answer:

26.4 SYNCHRONOUS TRANSMISSION

In synchronous transmission, characters are sent one after another without any gap, synchronized by clock pulses. Before the actual synchronous transmission, the processor writes to 8251 control port one or two synchronization characters, depending on the way the 8251 is configured. Thus the 8251 is aware of the sync (synchronization) characters to be used in the synchronous mode.

First of all, the 8251 sends out the programmed number of sync characters on the TxD pin. This is followed by the assembled data characters. The assembled data characters will not have any start or stop bits. The start and stop bits are not needed any more as one data character follows another without any time gap between them. However, the assembled data characters may have the optional parity bit. Since there are no start and stop bits for each character, the synchronous mode of transmission is faster. It is to be noted that in synchronous mode, a bit is sent out of TxD pin for every TxC* clock. It does not support ×16 and ×64 modes.

The 8251 expects a steady stream of data characters from the processor for transmission on TxD output pin. In case the transmitter becomes empty, indicated by TxE signal becoming active, the 8251 automatically sends out programmed number of sync characters on TxD pin to avoid losing synchronization. The waveform on TxD in synchronous transmission mode is shown in Fig. 26.4.

In synchronous mode, the receiver frequency must match the transmitting frequency. Else, it causes problems in receiving. This is because there is no start bit that ensures synchronization at the beginning of every character.

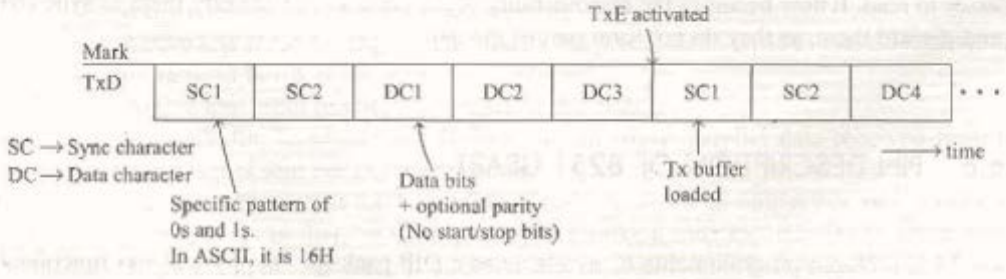


Fig. 26.4 Waveform on TxD in synchronous transmission assuming two sync characters

Waveform on RxD in synchronous receive mode is shown in Fig. 26.5. In synchronous mode, the receiver frequency must match the transmitting frequency. Else, it causes problems in receiving. This is because there is no start bit that ensures synchronization at the beginning of every character.

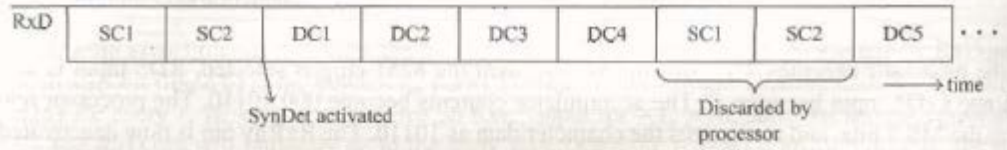


Fig. 26.5 Waveform on RxD in synchronous reception assuming two sync characters

In this mode, character synchronization can be achieved either internally or externally. It depends on bit 6 of the MI, when programmed for synchronous operation. This bit may be called ESD (external sync detect) bit, when programmed for synchronous operation.

If ESD bit = 1, 8251 is prgrammed for external sync detection.
If ESD bit = 0, 8251 is programmed for internal sync detection.

Before any data is received by 8251 in internal sync detect mode, the EH (enter hunt) bit must be set to 1 in the CI. This results in 8251 hunting for programmed number of sync characters on RxD input. The characters received are stored in internal registers that are not accessible to the user. They are compared with the programmed sync characters. Once it detects the sync characters, the 8251 comes out of the hunt mode and activates the SynDet (SD) output. Once the 8251 comes out of the hunt mode it starts receiving characters in the receiver buffer. It is to be noted that in synchronous mode, a bit is received on RxD pin for every RxC* clock. It does not support ×16 and ×64 modes. SD pin is an output pin in internal sync mode. This output is automatically deactivated when the processor reads the status register.

When in external sync mode, the SD pin is an input pin. In this case, external circuit checks for sync characters. Once sync characters are detected, logic 1 is input on the SD input. The 8251 then comes out of the hunt mode and starts receiving characters. The logic 1 on SD pin can be removed after one RxC* cycle.

Once sync characters are detected, the 8251 starts receiving characters in the receive shift register and moves it to the receive buffer for the processor to read. After some data characters are received, if sync characters are again received on the RxD pin, they are received in the receive buffer for the processor to read. It now becomes the responsibility of the processor to identify them as sync characters and discard them, as they do not form part of the data.

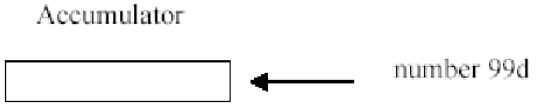
Q.9 a. Discuss the various types of addressing modes available to the 8051 instruction set. Give atleast one example of each. (10)

Answer:

The various addressing modes available to the 8051 instruction set are as follows:

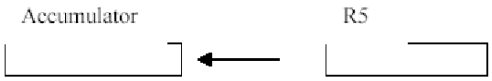
(i) Immediate Addressing

Immediate addressing simply means that the operand (which immediately follows the instruction op. code) is the data value to be used. For example the instruction:
MOV A, #99d



(ii) Register Addressing

One of the eight general-registers, R0 to R7, can be specified as the instruction operand. The assembly language documentation refers to a register generically as Rn. An example instruction using register addressing is:
ADD A, R5; Add register R5 to A (accumulator)

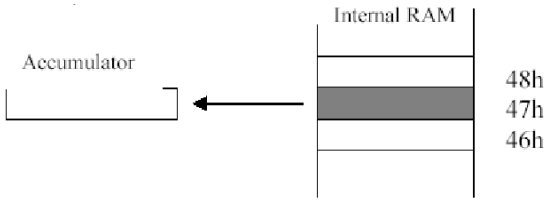


Here the content of R5 is added to the accumulator. One advantage of register addressing is that the instructions tend to be short, single byte instructions.

(iii) Direct Addressing

Direct addressing means that the data value is obtained directly from the memory location specified in the operand. For example consider the instruction:

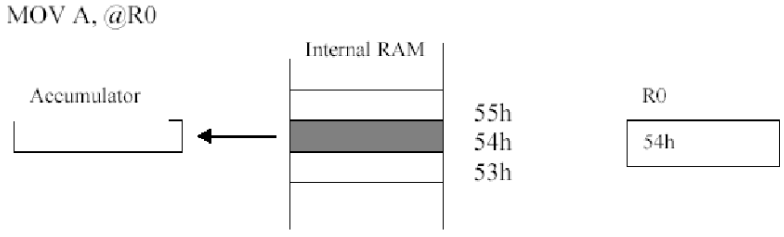
MOV A, 47h



The instruction reads the data from Internal RAM address 47h and stores this in the accumulator. Direct addressing can be used to access Internal RAM, including the SFR registers.

(iv) Register Indirect Addressing

In register indirect addressing, a register is used as pointer to the data. An example instruction, which uses indirect addressing, is as follows:



The @ symbol indicated that the indirect addressing mode is used. R0 contains a value, for example 54h, which is to be used as the address of the internal RAM location, which contains the operand data. Indirect addressing refers to Internal RAM only and cannot be used to refer to SFR registers. Note, only R0 or R1 can be used as register data pointers for indirect addressing.

(v) Indexed Addressing:

With indexed addressing a separate register, either the program counter, PC, or the data pointer DTPR, is used as a base address and the accumulator is used as an offset address. The effective address is formed by adding the value from the base address to the value from the offset address. Indexed addressing in the 8051 is used with the JMP or MOVC instructions. Look up tables are easy to be implemented with the help of index addressing.

Consider the example instruction:
MOVC A, @A+DPTR

MOVC is a move instruction, which moves data from the external code memory space. The address operand in this example is formed by adding the content of the DPTR register to the accumulator value. Here the DPTR value is referred to as the base address and the accumulator value referred to as the index address.

b. Write a program to add two 8 bit BCD numbers using 8051 instructions set. (6)

Answer:

Program to add two 8 bit BCD numbers:

```
MOV R0, #SRC ADDR
MOV R1, #00
MOV A, @R0
INC R0
ADD A, @ R0
DA A
JNC Next
MOV R2, A
MOV A, R1
ADD A, #01
DA A
MOV R1, A
MOV A, R2
MOV 40h, A
MOV 41h, R1
```

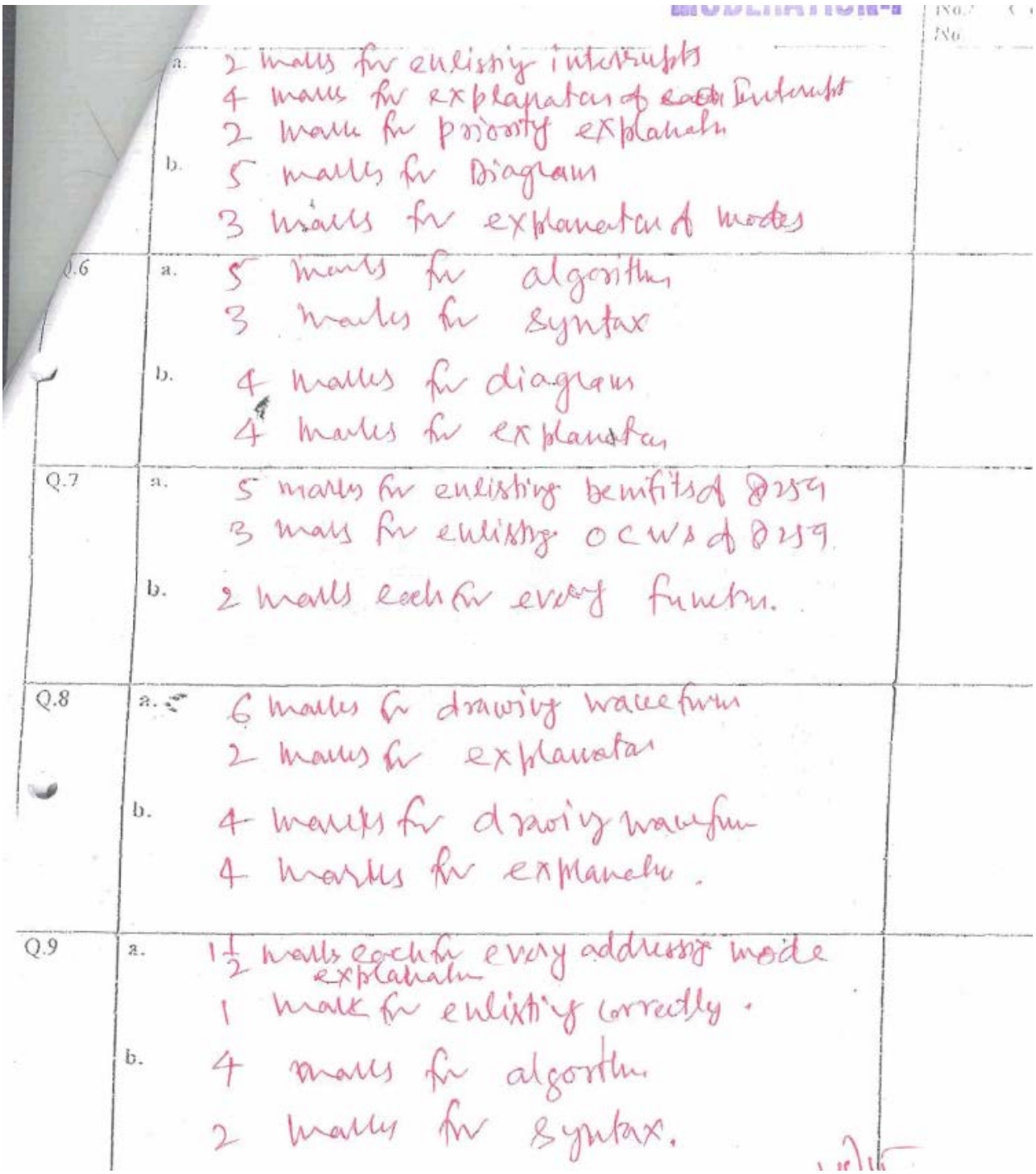
CODE: AE 108/AC/108 /AT108 SUBJECT: Microprocessor & Microcontrollers

(Marking Scheme)

MODERATION-I

Q.			Unit, Text Book, Page Contents No
Q.2	a.	4 marks for Pin diagram 4 marks for signal group diagram	
	b.	4 marks of listing categories 4 marks for examples	
Q.3	a.	6 marks for correct diagram 2 marks for all pins mentioned.	
	b.	2 marks for ~~all pins mentioned~~ Needs of I/O ports 6 marks for merits & demerits	
Q.4	a.	6 marks for algorithm 2 marks for Syntax	
	b.	6 marks for algorithm 2 marks for Syntax	

MODERATION-I



TEXT BOOK

I. The 8085 Microprocessor; Architecture, Programming and Interfacing, K. Udaya Kumar and B. S. Umashankar, Pearson Education, 2008