MICROCONTROLLER - 15EE52

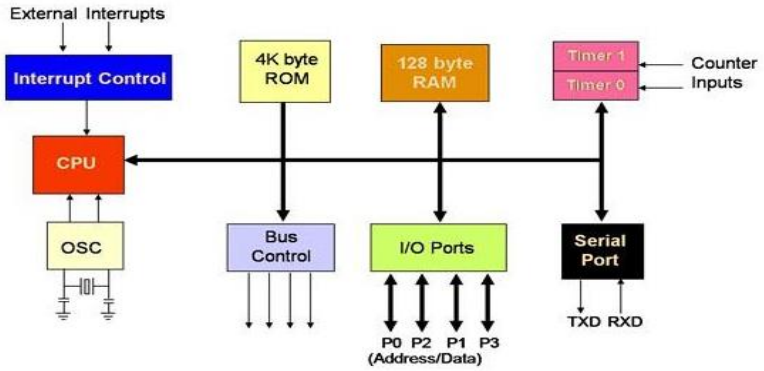
IAT 1 SOLUTION

SEP 2018

1. With a neat diagram, explain the architecture of 8051.

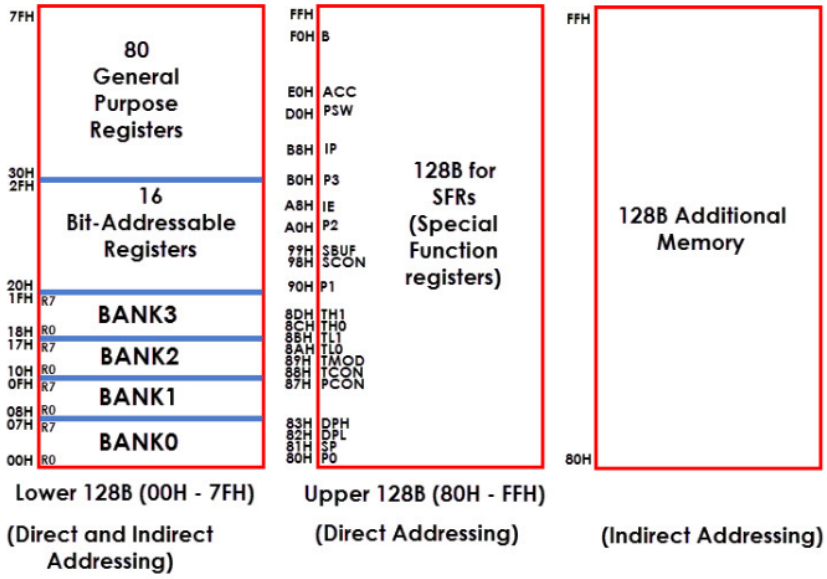
Block Diagram of 8051 Microcontroller

The following illustration shows the block diagram of an 8051 microcontroller –



Elaboration on each block.

2. Explain the internal structure of RAM of 8051 with necessary diagram.



Elaboration on each block.

3. Explain the operations of below 8051 instructions.

i. CJNE A, B, LABEL	ii. RLC A	iii. DIV AB
iv. PUSH B	v. MOV C, P0.1	

i. CJNE A, B, LABEL

Compare A and B and Jump to Label if A≠B.

A-B occurs and the Carry flag will be affected.

ii. RLC A

Contents of the accumulator are rotated along with carry.

iii. DIV AB

Before exectuion : A=Dividend

B=Divider

After execution: A= quotient

B=remainder

iv. PUSH B

Stack pointer is incremented by one and the contents of register B are stored.

v. MOV C, P0.1

Status of D1 bit of P0 port will be copied to the carry flag bit.

Each instruction should be quoted an example.

4. Write an assembly language program to convert ASCII to unpaced BCD and vice versa. Also write ALP to convert ASCII to Decimal and vice versa. Include suitable comments.

ASCII to Unpacked BCD:

MOV A, #'5'

SUBB A, #30H

MOV 50H, A

Unpacked BCD to ASCII:

MOV A, #05H

ADD A, #30H

MOV 50H, A

ASCII to Decimal

```
MOV A,#'2'   ;A=32H
ANL A,#0FH   ;get the lower nibble
MOV R1,#'9'  ;R1=39H
ANL R1,#0FH  ;get the lower nibble
```

– Combined them to the packed BCD.

```
SWAP A       ;become upper nibble A=20H
ORL A,R1     ;packed BCD,A=29H
```

Decimal to ASCII

```
MOV A,#29H
ANL A,#0FH   ;get the lower nibble
```

– The unpacked BCD is tagged with 30H

```
ORL A,#30H   ;make it an ASCII,A=39H '9'
```

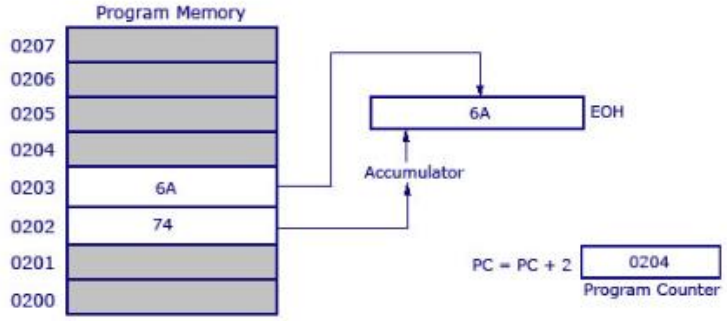
5. List and explain different addressing modes of 8051 with suitable examples.

Addressing Modes

- 8051 provides a total of five distinct addressing modes.
- 1. Immediate
- 2.Register
- 3.Direct
- 4. Register indirect
- 5. Indexed

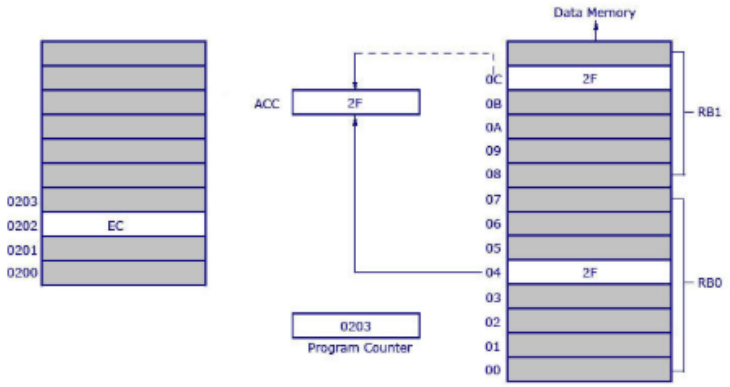
Immediate Addressing Mode

Instruction	Opcode	Bytes	Cycles
MOV A, #6AH	74H	2	1



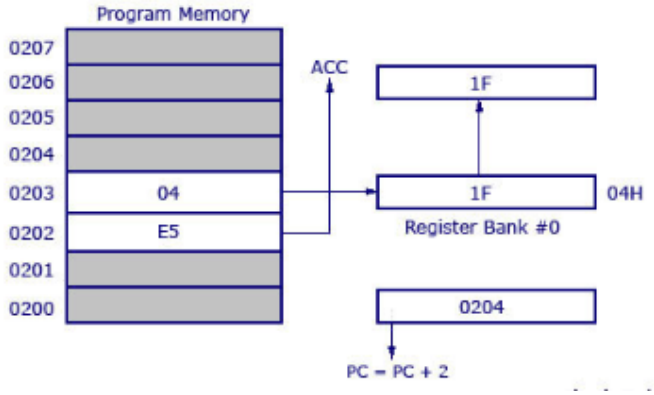
Register Direct Addressing Mode

Instruction	Opcode	Bytes	Cycles
MOV A, R4	ECH	1	1



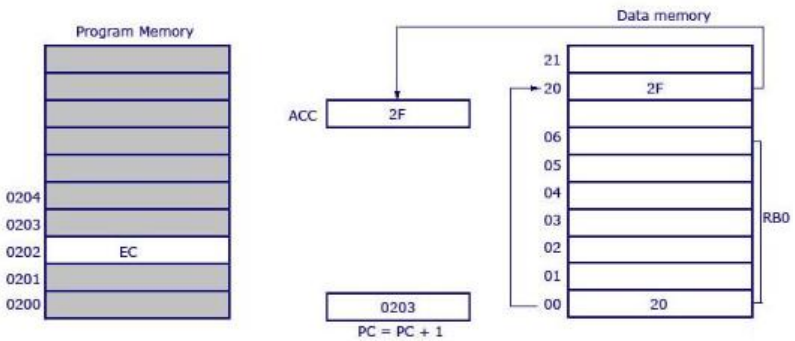
Direct Addressing Mode

Instruction	Opcode	Bytes	Cycles
MOV A, #04H	E5	2	1



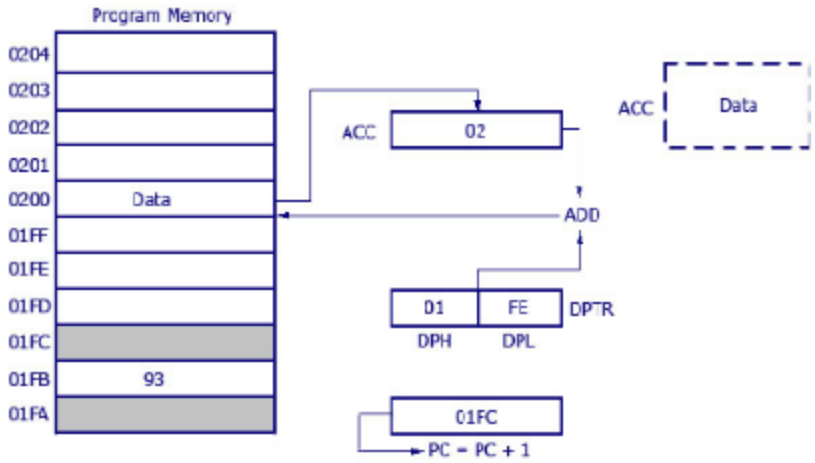
Register Indirect Addressing Mode

Instruction	Opcode	Bytes	Cycles
MOV A, @ R0	E6H	1	1



Indexed Addressing Mode

Instruction	Opcode	Bytes	Cycles
MOVC A,@A +DPTR	93H	1	2



6. Explain the following code with respect to stack.

MOV SP, #10 h

PUSH SP

POP 0E0 h

ADD A, #10 h

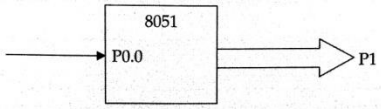
MOV SP, #10 h ; Stack pointer is initialized as 10h

PUSH SP ; Stack pointer content gets incremented by one so 11h will be moved onto stack at 11h

POP 0E0 h ; Stack contents at 11h will be popped to accumulator because its address is 0E0h

ADD A, #10 h ; immediate data 10h gets added to accumulator content 10h and the sum is in the Accumulator i.e., 21h

7. Write a program to bring in data in serial form and send it out in parallel form.



Let us take in data through port pin P0.0 and transmit it through P1.

```
      MOV  R0,#08        ;counter for 8 bits
      SETB P0.0          ;make P0.0 an input port
BACK: MOV  C,P0.0        ;move data from P0.0 into the carry bit
      RRC  A             ;rotate right,the data goes from 'CY' into A
      DJNZ R0,BACK       ;repeat until all 8 bits are moved in
      MOV  P1,A          ;the data is now transferred in parallel to P1
      END
```