FOURTH SEMESTER BTECH DEGREE EXAMINATION, 2017
CS202: COMPUTER ORGANISATION & ARCHITECTURE

Time:3 hrsMax. Marks:100

PART A
(Answer all questions. Each carries 3 marks)

1. **Differentiate between big endian and little endian byte ordering.**
Ans:

- Big-endian and little-endian are terms that describe the order in which a sequence of bytes are stored in computer memory.
- **Big-endian** is an order in which the "big end" (most significant value in the sequence) is stored first (at the lowest storage address).
- **Little-endian** is an order in which the "little end" (least significant value in the sequence) is stored first.
- For example, in a big-endian computer, the two bytes required for the hexadecimal number 4F52 would be stored as 4F52 in storage (if 4F is stored at storage address 1000, for example, 52 will be at address 1001).
- In a little-endian system, it would be stored as 524F (52 at address 1000, 4F at 1001).

Big Endian In big endian, you store the most significant byte in the smallest address. Here's how it would look:

Address	Value
1000	90
1001	AB
1002	12
1003	CD

Little Endian

In little endian, you store the *least* significant byte in the smallest address. Here's how it would look:

Address	Value
1000	CD
1001	12
1002	AB
1003	90

2**. Describe the basic instruction types.**

Ans:

Basic Instruction Types:

Instruction Type	**Syntax**	**Eg**	**Description**
Three Address	Operation Source1,Source2,Destination	Add A,B,C	Add values of variable A ,B & place the result into c.
Two Address	Operation Source,Destination	Add A,B	Add the values of A,B & place the result into B.
One Address	Operation Operand	Add B	Content of accumulator add with content of B.

3. **Give the control sequence for execution of instruction *Add[R3],R1***

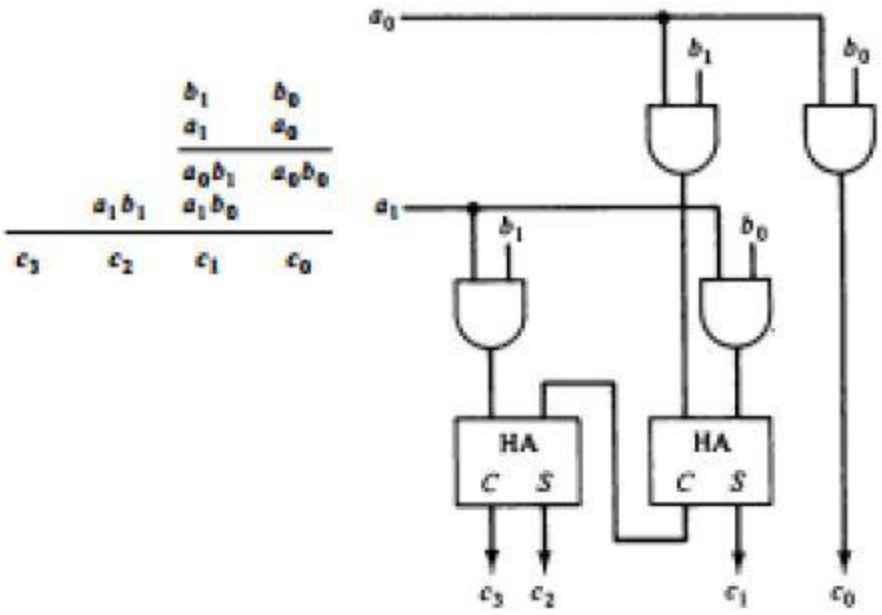
Ans:

Step	Action
1	PC_{out} , MAR_{in} , Read, Select4, Add, Z_{in}
2	Z_{out} , PC_{in} , Y_{in} , WMF C
3	MDR_{out} , IR_{in}
4	$R3_{out}$, MAR_{in} , Read
5	$R1_{out}$, Y_{in} , WMF C
6	MDR_{out} , SelectY, Add, Z_{in}
7	Z_{out} , $R1_{in}$, End

4. Design a 2x2 array multiplier.
Ans:

- The multiplicand bits are b1 and b0, the multiplier bits are a1 and a0, and the product is c3 c2 c1 c0
- The first partial product is formed by multiplying a0 by b1 b0
- The multiplication of two bits such as a0 and b0 produces a 1 if both bits are 1; otherwise, it produces a 0. This is identical to an AND operation and can be implemented with an AND gate.
- As shown in the diagram, the first partial product is formed by means of two AND gates. The second partial product is formed by multiplying a1 by b1 b0 and is shifted one position to the left.
- The two partial products are added with two half-adder (HA) circuits. Usually, there are more bits in the partial products and it will be necessary to use full-adders to produce the sum.
- Note that the least significant bit of the product does not have to go through an adder since it is formed by the output of the first AND gate.

Figure 10-9 2-bit by 2-bit array multiplier.



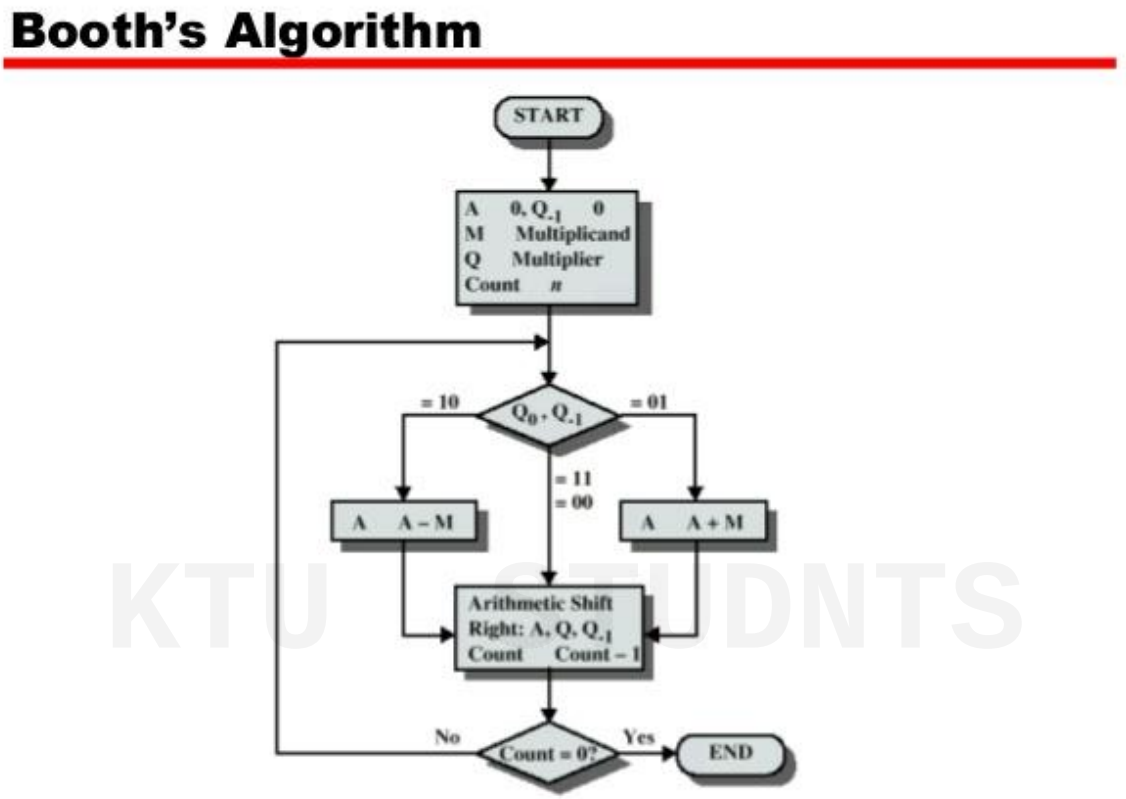
4x3=12

PART B
(Answer any two. Each carries 9 marks)

5. a) **Describe the different addressing modes.** (5)
Ans :

- Immediate mode
- Register mode
- Absolute mode
- Indirect mode
- Index mode
- Base with index
- Base with index and offset
- Relative mode
- Auto-increment mode
- Auto-decrement mode

b) **Give the flow chart for Booth's Multiplication**. (4)
Ans :

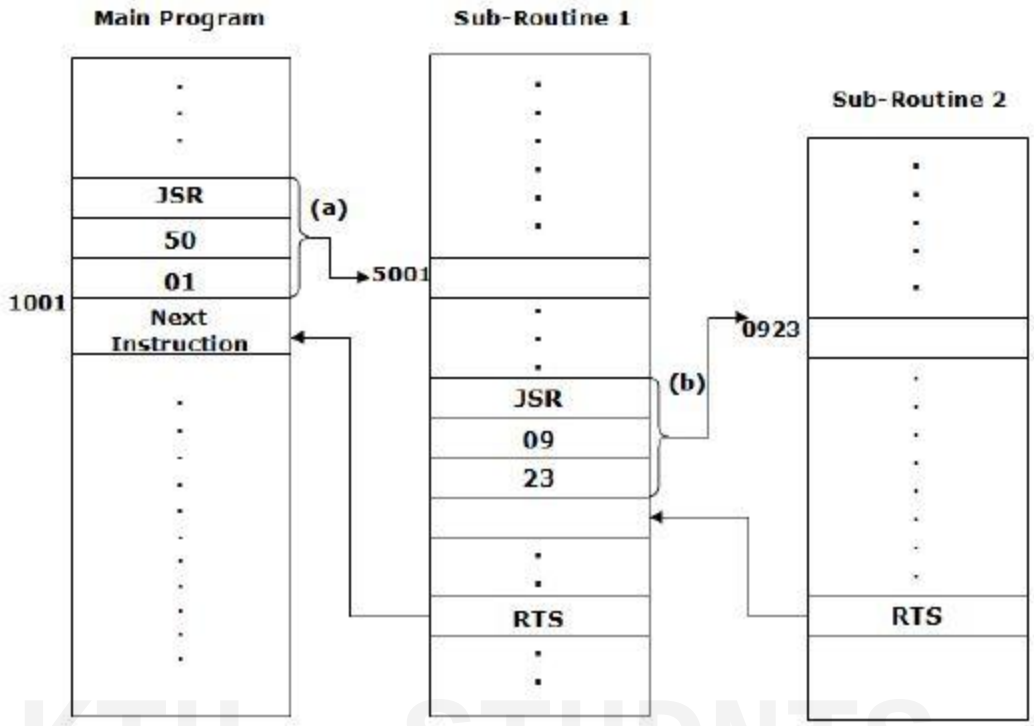


6. **Explain how nested subroutines are processed internally.**
Ans :

- Subroutine nesting, is to have one subroutine call another.
- In this case, the return address of the second call is also stored in the link register, destroying its previous contents.
- Hence, it is essential to save the contents of the link register in some other location before calling another subroutine.
- Otherwise, the return address of the first subroutine will be lost. Subroutine nesting can be carried out to any depth.

- Eventually, the last subroutine called completes its computations and returns to the subroutine that called it.
- The return address needed for this first return is the last one generated in the nested call Memory location Subroutine SUB 1000 first instruction …. …. Return 204 Return sequence. That is, return addresses are generated and used in a last-in-first-out order.
- This suggests that the return addresses associated with subroutine calls should be pushed onto a stack. A particular register is designated as the stack pointer, SP, to be used in this operation.
- The stack pointer points to a stack called the processor stack.
- The Call instruction pushes the contents of the PC onto the processor stack and loads the subroutine address into the PC.
- The Return instruction pops the return address from the processor stack into the PC. Since PC saving and recovery is automatic by the use of stack, it is possible to execute nested subroutines as shown in figure 2.
- In other words, it is possible for a subroutine to call another subroutine.
- As in figure 2, first the subroutine 1 is called from part „a" of main program. At this point, the returning address i.e. 1001 should pe pushed on the top of the return stack. From this subroutine, subrotuine2 is called and jump is made to address 0923 i.e. jump 2.
- At this point, the returning address in subroutine 1 should be pushed onto stack.
- Thus the return stack should contain the returning addresses in the reverse order of the calling order. First „a" is called, and then „b" is called; so returning address of „b" should be on top of stack and below it should be the returning address of „a".

Figure 2: Nested Subroutines referred via main program



7. Explain restoring method of division with an example.

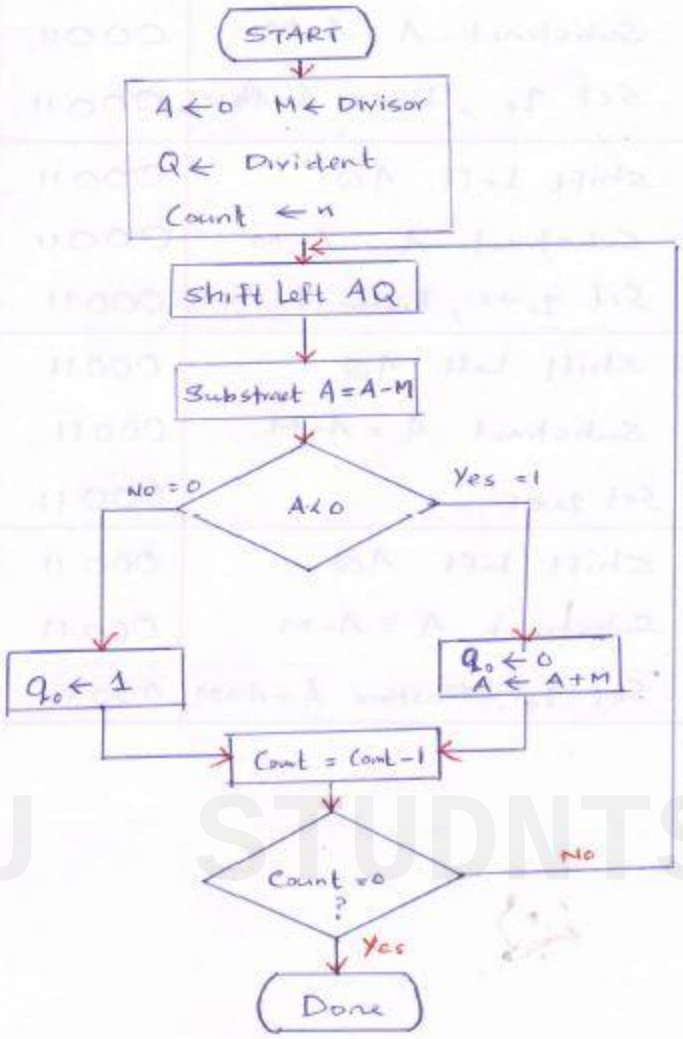
Ans :

Restoring Division

ALGORITHM

1. Shift A and Q Left 1-bit
2. Substract M from A, and place result in A
3. If the sign of A is 1, set q_0 to 0
   and Add M back to A (Restore A)
   - Otherwise set q_0 to 1

FLOWCHART



Example: Divide "8 by 3" using Restoring method

Ans:

8 → 1000 (Divident)

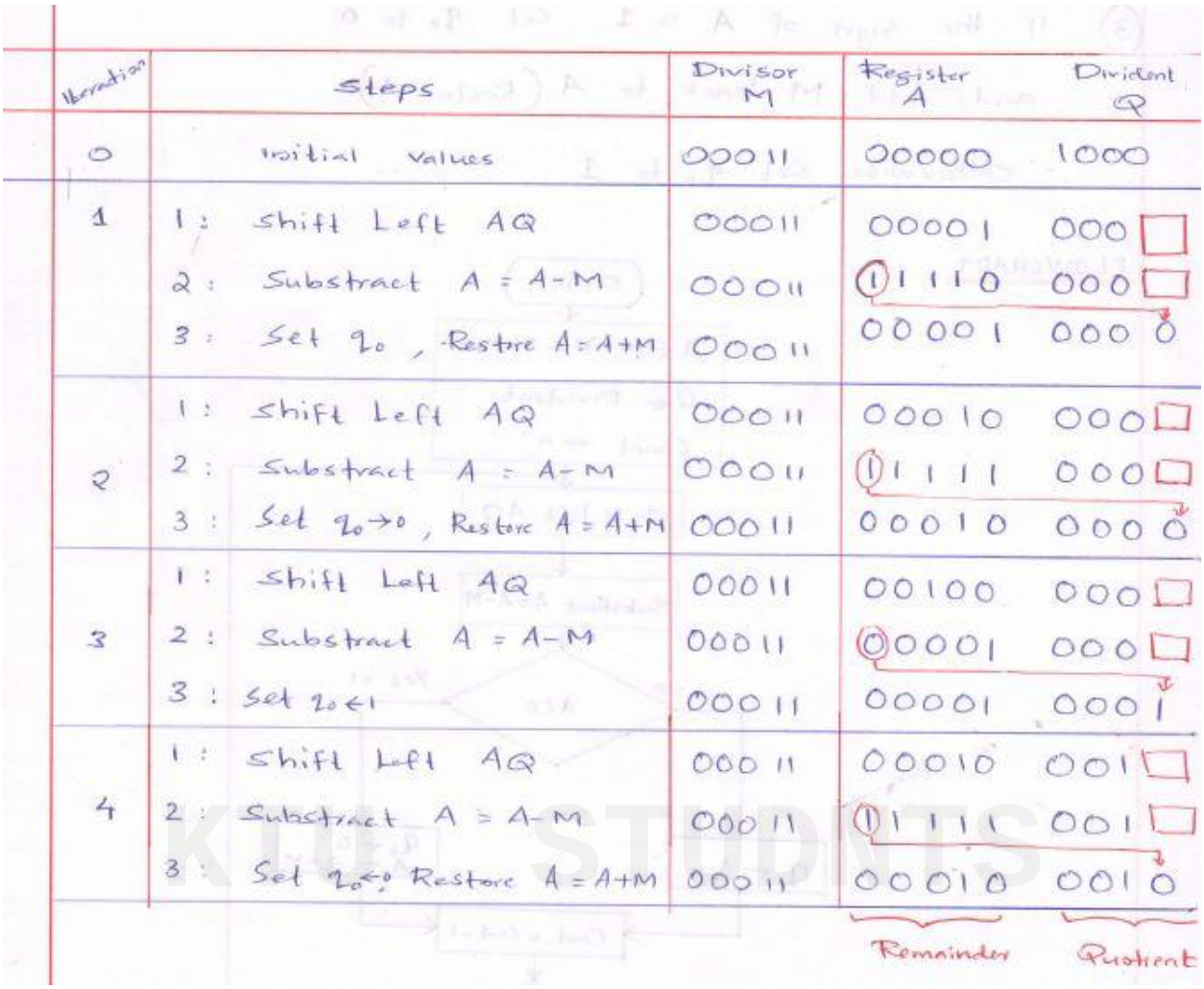
3 → 0011 (Divisor

0010 (Quotient)

0011 | 1000 (Divident)

0010 (Remainder)

$$3 \overline{)\,8\,} \quad 2, \quad 6, \quad 2$$



Iteration	Steps	Divisor M	Register A	Divident Q
0	Initial values	00011	00000	1000
1	1: Shift Left AQ	00011	00001	000
	2: Substract A = A-M	00011	11110	000
	3: Set q_0, Restore A=A+M	00011	00001	0000
2	1: Shift Left AQ	00011	00010	000
	2: Substract A = A-M	00011	11111	000
	3: Set $q_0 \to 0$, Restore A = A+M	00011	00010	0000
3	1: Shift Left AQ	00011	00100	000
	2: Substract A = A-M	00011	00001	000
	3: Set $q_0 \leftarrow 1$	00011	00001	0001
4	1: Shift Left AQ	00011	00010	001
	2: Substract A = A-M	00011	11111	001
	3: Set $q_0 \leftarrow 0$, Restore A = A+M	00011	00010	0010
			Remainder	Quotient

2x9=18

PART C
(Answer all questions. Each carries 3 marks)

8. **Write notes on vectored interrupts.**

Ans :

Vectored Interrupt:

- Here the device requesting an interrupt may identify itself to the processor by sending a **special code over the bus**& then the processor start executing the ISR.
- The code supplied by the processor indicates the starting address of the ISR for the device.
- The code length ranges from 4 to 8 bits.

- The location pointed to by the interrupting device is used to store the staring address to ISR.
- The processor reads this address, called the **interrupt vector**& loads into PC.
- The interrupt vector also includes a new value for the Processor Status Register.

When the processor is ready to receive the interrupt vector code, it activate the interrupt acknowledge (INTA) line.

9. **Differentiate between synchronous and asynchronous buses.**

Ans :

Synchronous bus:

- Transmitter and receivers are synchronized of clock.
- Data bits are transmitted with synchronization of clock.
- Character is received at constant Rate.
- Data transfer takes place in block.
- Start and stop bit are required to establish communication of each character.
- Used in high – speed transmission.

Asynchronous bus:

- Transmitters and receivers are not synchronized by clock.
- Bit‟s of data are transmitted at constant rate.
- Character may arrive at any rate at receiver.
- Data transfer is character oriented.
- Start and stop bits are required to establish communication of each character.
- Used in low – speed transmission.

10. **Briefly explain static memory.**

Ans :

SRAM (**static RAM**) is random access memory (**RAM**) that retains data bits in its memory as long as power is being supplied. Unlike dynamic **RAM** (DRAM), which stores bits in cells consisting of a capacitor and a transistor, **SRAM** does not have to be periodically refreshed.

Advantages:

- Low power consumption
- Simplicity – a refresh circuit is not needed
- Reliability

Disadvantages:

- High Price
- Capacity

11. Describe the LRU algorithm for cache replacement. **4x3=12**

PART D
(Answer any two. Each carries 9 marks)

12. a) **Which are the different bus arbitration schemes**?
Ans :

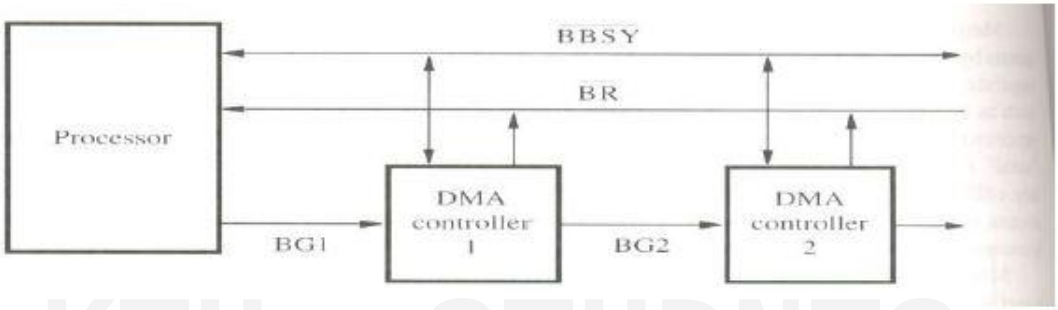
- Centralized arbitration
- Distributed arbitration

Centralized Arbitration:

- Here the processor is the bus master and it may grants bus mastership to one of its DMA controller.
- A DMA controller indicates that it needs to become the bus master by activating the Bus Request line (BR) which is an open drain line.
- The signal on BR is the logical OR of the bus request from all devices connected to it.

- When BR is activated the processor activates the Bus Grant Signal (BGI) and indicated the DMA controller that they may use the bus when it becomes free.
- This signal is connected to all devices using a daisy chain arrangement.
- If DMA requests the bus, it blocks the propagation of Grant Signal to other devices and it indicates to all devices that it is using the bus by activating open collector line, Bus Busy (BBSY).

Fig:A simple arrangement for bus arbitration using a daisy chain



Distributed Arbitration:

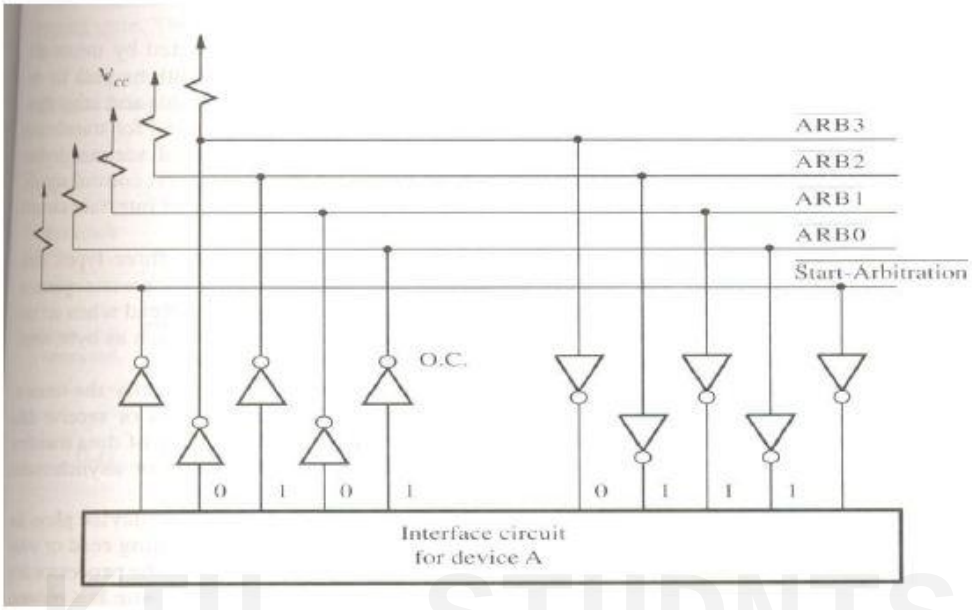
- It means that all devices waiting to use the bus have equal responsibility in carrying out the arbitration process.

- Each device on the bus is assigned a 4 bit id.
- When one or more devices request the bus, they assert the Start-Arbitration signal & place their 4 bit ID number on four open collector lines, ARB0 to ARB3.
- A winner is selected as a result of the interaction among the signals transmitted over these lines.

- The net outcome is that the code on the four lines represents the request that has the highest ID number.

- The drivers are of open collector type. Hence, if the i/p to one driver is equal to 1, the i/p to another driver connected to the same bus line is equal to „0"(ie. bus the is in low-voltage state).

Fig:A distributed arbitration scheme



Eg:

Assume two devices A & B have their ID 5 (0101), 6(0110) and their code is 0111.

Each devices compares the pattern on the arbitration line to its own ID starting from MSB.

If it detects a difference at any bit position, it disables the drivers at that bit position. It does this by placing „0" at the i/p of these drivers.

In our eg. „A" detects a difference in line ARB1, hence it disables the drivers on lines ARB1 & ARB0.

This causes the pattern on the arbitration line to change to 0110 which means that „B" has won the contention.

(5)

b) **Write notes on flash memory** (4)

Ans :

Flash Memory:

- In EEPROM, it is possible to read & write the contents of a single cell.
- In Flash device, it is possible to read the contents of a single cell but it is only possible to write the entire contents of a block.
- Prior to writing,the previous contents of the block are erased.
- Eg.In MP3 player,the flash memory stores the data that represents sound.
- Single flash chips cannot provide sufficient storage capacity for embedded system application.
- There are 2 methods for implementing larger memory modules consisting of number of chips.They are,
- □ Flash Cards
- □ Flash Drives.

Merits:

- Flash drives have greater density which leads to higher capacity & low cost per bit.
- It requires single power supply voltage & consumes less power in their operation.

Flash Cards:

- One way of constructing larger module is to mount flash chips on a small card.
- Such flash card have standard interface.
-
- The card is simply plugged into a conveniently accessible slot.
- Its memory size are of 8,32,64MB.

- Eg:A minute of music can be stored in 1MB of memory. Hence 64MB flash cards can store an hour of music.
- **Flash Drives:**
- Larger flash memory module can be developed by replacing the hard disk drive.
- The flash drives are designed to fully emulate the hard disk.
- The flash drives are solid state electronic devices that have no movable parts.

- **Merits:**
- They have shorter seek and access time which results in faster response.
- They have low power consumption which makes them attractive for battery driven application.
- They are insensitive to vibration.

- **Demerit:**
- The capacity of flash drive (<1GB) is less than hard disk(>1GB).
- It leads to higher cost perbit.
- Flash memory will deteriorate after it has been written a number of times(typically at least 1 million times.)
-

13. **Explain the working of Universal Serial Bus (USB).**

Ans :

- USB, short for **Universal Serial Bus**, is a standard type of connection for many different kinds of devices. Generally, USB refers to the types of cables and connectors used to connect these many types of external devices to computers.

- The Universal Serial Bus standard has been extremely successful. USB ports and cables are used to connect hardware such as printers, scanners, keyboards, mice, flash drives, external hard drives, joysticks, cameras, and more to computers of all kinds, including desktops, tablets, laptops, netbooks, etc. In fact, USB has become so common that you'll find the connection available on nearly any computer-like device such as video game consoles, home audio/visual equipment, and even in many automobiles.
- Many portable devices, like smartphones, ebook readers, and small tablets, use USB primarily for charging. USB charging has become so common that it's now easy to find replacement electrical outlets at home improvement stores with USB ports built it, negating the need for a USB power adapter.
- **USB Versions**
- There have been three major USB standards, 3.1 being the newest:
- USB 3.1: Called *Superspeed*+, USB 3.1 compliant devices are able to transfer data at 10 Gbps (10,240 Mpbs).
- USB 3.0: Called *SuperSpeed USB*, USB 3.0 compliant hardware can reach a maximum transmission rate of 5 Gbps (5,120 Mbps).
- USB 2.0: Called *High-Speed USB*, USB 2.0 compliant devices can reach a maximum transmission rate of 480 Mbps.
- USB 1.1: Called *Full Speed USB*, USB 1.1 devices can reach a maximum transmission rate of 12 Mbps.

- **How USB Works**
- When a computer is powered up and USB devices are connected to a hub, the system will query and request from them the information on how much bandwidth is needed. Enumeration process will then occur where each device is assigned with a unique address. After that, the system will determine what kind of data the USB devices wish to transfer. There will be 4 different modes of transfers:

- Interrupt: used for devices which transfer little amount of data but need fast response (E.g. mouse, keyboard)
- Bulk: used for devices which receive big packet of data (E.g. printer)
- Isochronous: used for devices which requires streaming process (E.g. speaker, webcam)
- Control: short, simple commands to the device, and a status response.

- After all connected devices are enumerated, the computer system will take care of the overall bandwidth and allocate it to different devices according to their transfer mode. Most of the bandwidth will be used for interrupt and isochronous transfer to ensure their requests are guaranteed. Once 90% of bandwidth is taken, the computer will refuse any other transfer from these two modes. Bulk or control

transfer (if available) will then take up the remaining bandwidth amount, which is up to 10%.

USB Main Features

- A maximum of 127 peripherals can be connected to a single USB host controller.
- USB device has a maximum speed up to 480 Mbps (for USB 2.0).
- Length of individual USB cable can reach up to 5 meters without a hub and 40 meters with hub.
- USB acts as "plug and play" device.
- USB can draw power by its own supply or from a computer. USB devices use power up to 5 voltages and deliver up to up to 500 mA.
- If a computer turns into power-saving mode, some USB devices will automatically convert themselves into "sleep" mode

14. a) **Describe the different types of DRAMS**. (5)

Ans :

Synchronous DRAM:

- Here the operations e directly synchronized with clock signal.
- The address and data connections are buffered by means of registers.
- The output of each sense amplifier is connected to a latch.
- A Read operation causes the contents of all cells in the selected row to be loaded in these latches.
- Data held in the latches that correspond to the selected columns are transferred into the data output register, thus becoming available on the data output pins.

Asynchronous DRAM:

- Asynchronous DRAM is the basic type of DRAM on which all other types are based.
- Asynchronous DRAMs have connections for power, address inputs, and bidirectional data lines.
- Although this type of DRAM is asynchronous, the system is run by a memory controller which is clocked, and this limits the speed of the system to multiples of the clock rate.
- Nevertheless the operation of the DRAM itself is not synchronous

Double Data Rate SDRAM(DDR-SDRAM):

- The standard SDRAM performs all actions on the rising edge of the clock signal.

- The double data rate SDRAM transfer data on both the edges(loading edge, trailing edge).
- The Bandwidth of DDR-SDRAM is doubled for long burst transfer.
- To make it possible to access the data at high rate , the cell array is organized into two banks.
- Each bank can be accessed separately.
- Consecutive words of a given block are stored in different banks.
- Such interleaving of words allows simultaneous access to two words that are transferred on successive edge of the clock.

b) **Compare the speed, size and cost of different types of memories**.

Ans :

Characteristics	SRAM	DRAM	Magnetis Disk
Speed	Very Fast	Slower	Much slower than DRAM
Size	Large	Small	Small
Cost	Expensive	Less Expensive	Low price

9x2=18

PART E
(Answer any four. Each carries 10 marks)

15. **Which are the different methods of processor organization?**
Ans :

- Bus organization
- Scratchpad memory
- Accumulator register

Bus organization

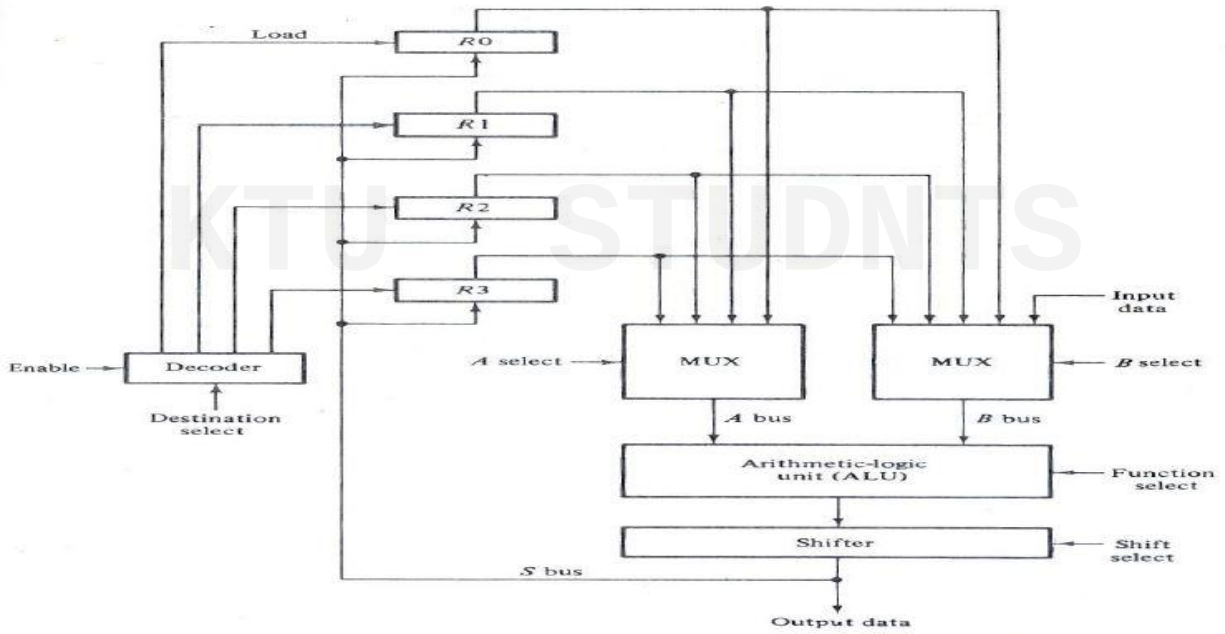


Figure 9-1 Processor registers and ALU connected through common buses

- Each registers is connected two multiplexers(MUX) to form input Buses A and B.
- The selection line of each multiplexers select one registers for the particular bus
- The bus A and B are applied to a common arithmetic and logic unit

- The function selected in the ALU determines the particular operation that is to be performed
- The shift micro operation are implemented in the shifter
- The result of the micro operations goes through the output bus S into input of all registers
- The destination registers that receives the information from the output bus is selected by decoder
- When decoder is enabled, this activates one of the register load input to provide a transfer path between the data on the S bus and the input of selected destination register

The control unit that supervises the processor bus system directs the information flow through the ALU by selecting the various components in the unit. For example, to perform the microoperation:

$$R1 \leftarrow R2 + R3$$

the control must provide binary selection variables to the following selector inputs:

1. MUX A selector: to place the contents of $R2$ onto bus A.
2. MUX B selector: to place the contents of $R3$ onto bus B.
3. ALU function selector: to provide the arithmetic operation $A + B$.
4. Shift selector: for direct transfer from the output of the ALU onto output bus S (no shift).
5. Decoder destination selector: to transfer the contents of bus S into $R1$.

Scratchpad memory

- Registers in a processor unit can be enclosed within a small memory unit
- A small memory when included in a processor unit is scratchpad memory

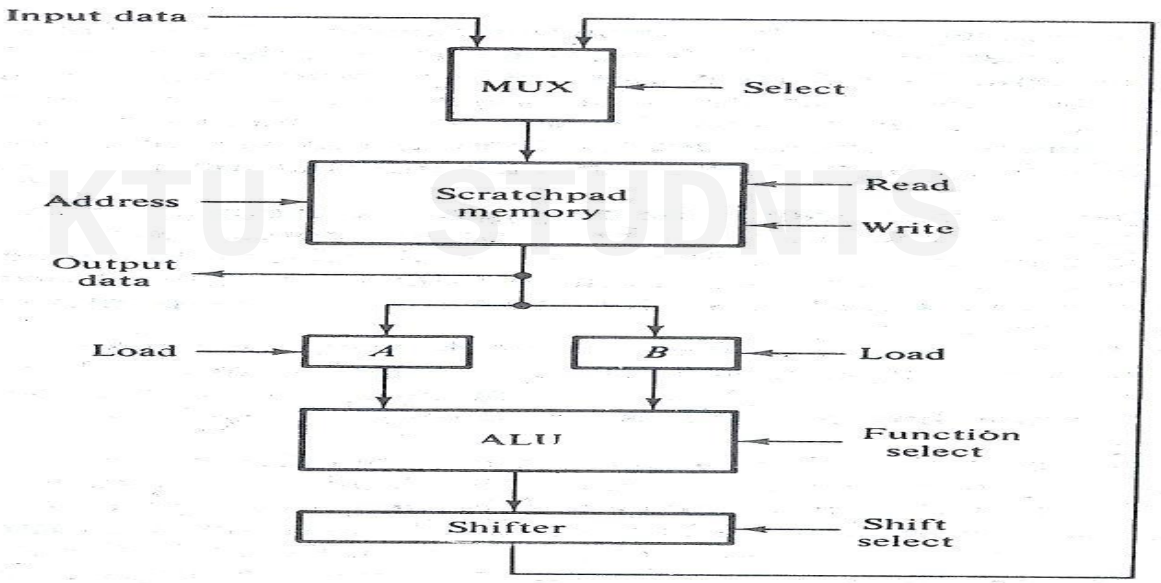


Figure 9-2 Processor unit employing a scratchpad memory

- A source register is selected from memory and load into register A
- A second source register is selected from memory and loaded into B
- The selection is done by specifying the corresponding word address and activating the memory read input
- The information in A and B is manipulated in the ALU and shifter
- The result of the operation is transferred to a memory register by specifying its word address and activating the memory write input control

Accumulator register

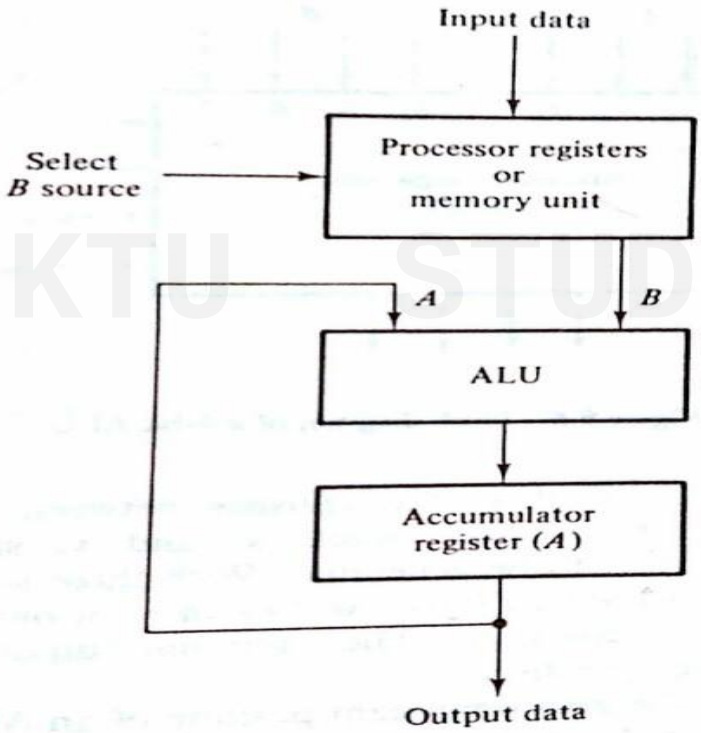


Figure 9-4 Processor with an accumulator register

- Some processor unit distinguishes one register from all others and call it an accumulator register

- In a computer's central processing unit (CPU), an **accumulator** is a register in which intermediate arithmetic and logic results are stored
- Accumulator performs-
  - Parallel load
  - Shift operations
  - Counting
  - Data-processing operations
- Accumulator is a multifunction register that, by itself, can be made to perform all of the micro operations in a processor unit

- The micro operations included in an accumulator depend on the operations that must be included in the particular processor.

TABLE 9-10 List of microoperations for an accumulator

Control variable	Microoperation	Name
p_1	$A \leftarrow A + B$	Add
p_2	$A \leftarrow 0$	Clear
p_3	$A \leftarrow \bar{A}$	Complement
p_4	$A \leftarrow A \wedge B$	AND
p_5	$A \leftarrow A \vee B$	OR
p_6	$A \leftarrow A \oplus B$	Exclusive-OR
p_7	$A \leftarrow \text{shr } A$	Shift-right
p_8	$A \leftarrow \text{shl } A$	Shift-left
p_9	$A \leftarrow A + 1$	Increment
	If $(A = 0)$ then $(Z = 1)$	Check for zero

16. **Explain the design of a 4bit Arithmetic unit with two selection variables, which performs the basic arithmetic functions.**

Ans :

- The basic section of an arithmetic section of an ALU is parallel adder
- By controlling the inputs to the parallel adder , we can obtain different types of arithmetic operation
- Figure demonstrate the arithmetic operations obtaining when one set of inputs to a parallel adder is controlled externally

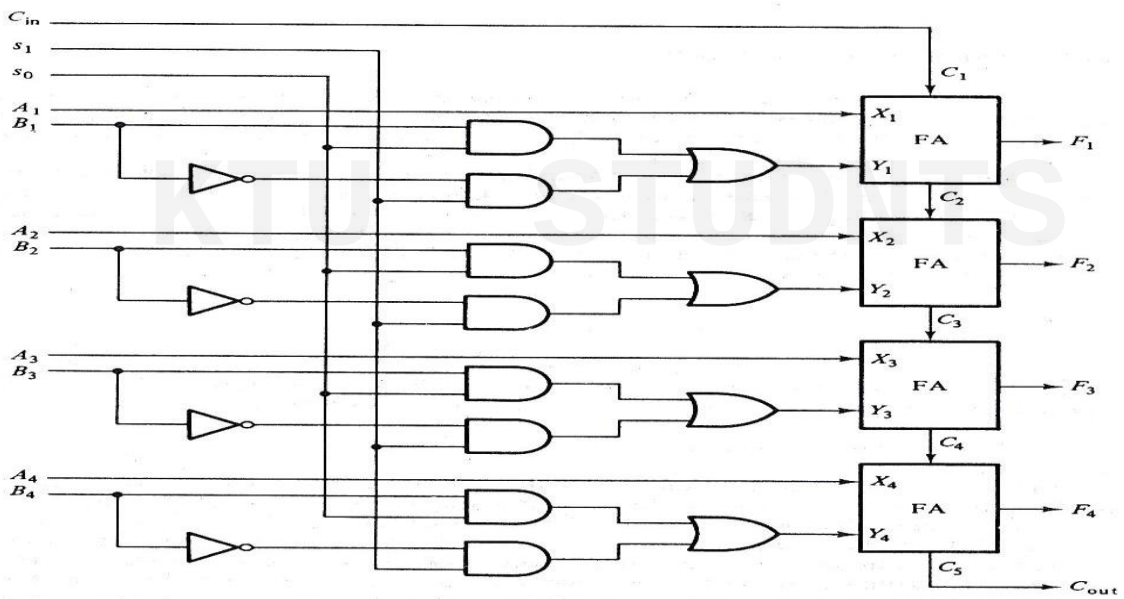


Figure 9-8 Logic diagram of arithmetic circuit

- The design of 4 bit arithmetic circuit that performs 8 arithmetic operations is shown in the figure
- The arithmetic operations implemented in the arithmetic circuit are listed in the table

- The values of the Y inputs are a function of selection variable S_1 and S_0 Adding the value of Y in each case to A plus the C_{in} value gives the arithmetic operations in each case
- The given arithmetic circuit needs a combinational circuit in each case specified by the following Boolean functions
    - Xi=Ai
    - Yi=(Bi*S0) +(Bi'*S1)

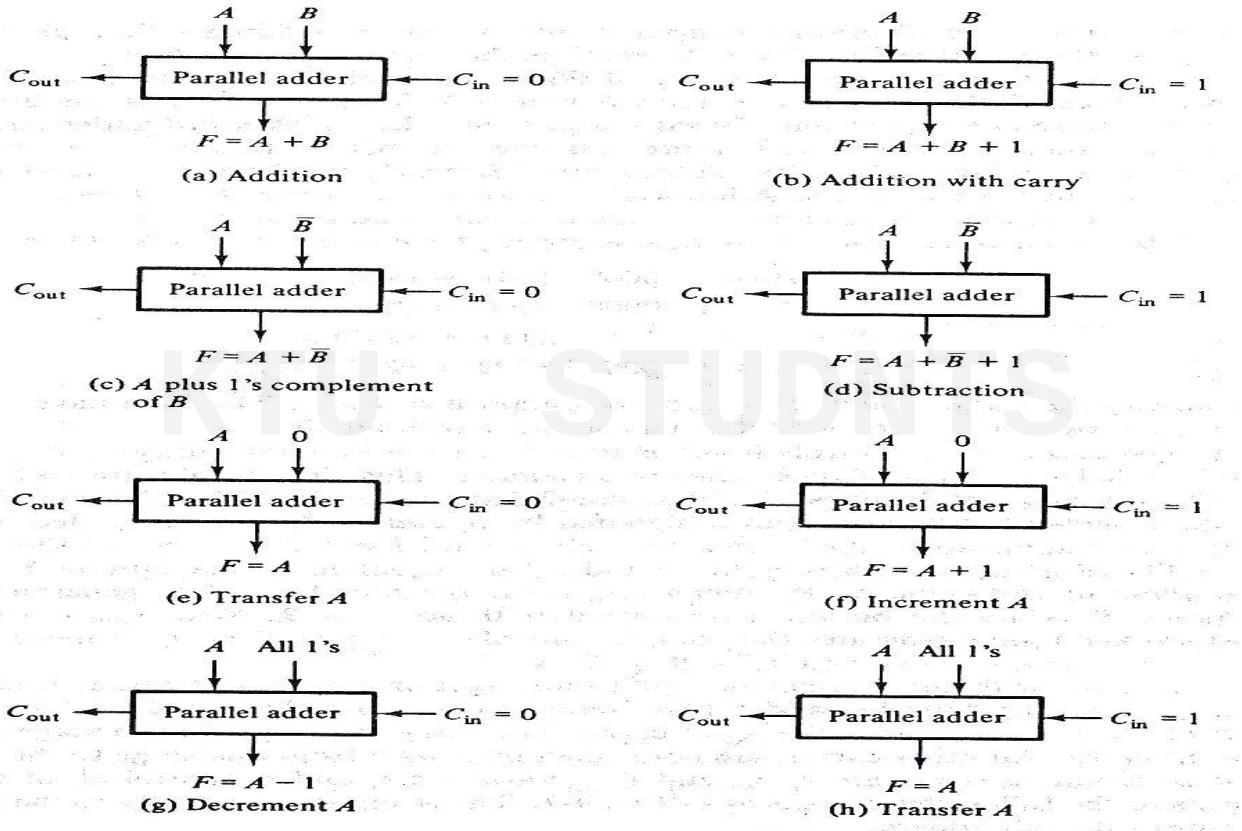


Figure 9-6 Operations obtained by controlling one set of inputs to a parallel adder

TABLE 9-1 Function table for the arithmetic circuit of Fig. 9-8

Function select			Y equals	Output equals	Function
s_1	s_0	C_{in}			
0	0	0	0	$F = A$	Transfer A
0	0	1	0	$F = A + 1$	Increment A
0	1	0	B	$F = A + B$	Add B to A
0	1	1	B	$F = A + B + 1$	Add B to A plus 1
1	0	0	$\overline{B}$	$F = A + \overline{B}$	Add 1's complement of B to A
1	0	1	$\overline{B}$	$F = A + \overline{B} + 1$	Add 2's complement of B to A
1	1	0	All 1's	$F = A - 1$	Decrement A
1	1	1	All 1's	$F = A$	Transfer A

17. a) **Explain the design of status register.**(5)

Ans :

- The status register is the hardware register that contains information about the state of the processor.
- It is sometimes convenient to supplement the ALU with a status Register where these status bit conditions are stored for further analysis
- Status bits conditions are sometimes are called flag bits or condition code.
- The four status bits are symbolized by C,S,V and Z . The bits are set on cleared as a result of an operation performed in ALU .
- C-carry
- V-overflow
- S-sign bit
- Z-zero

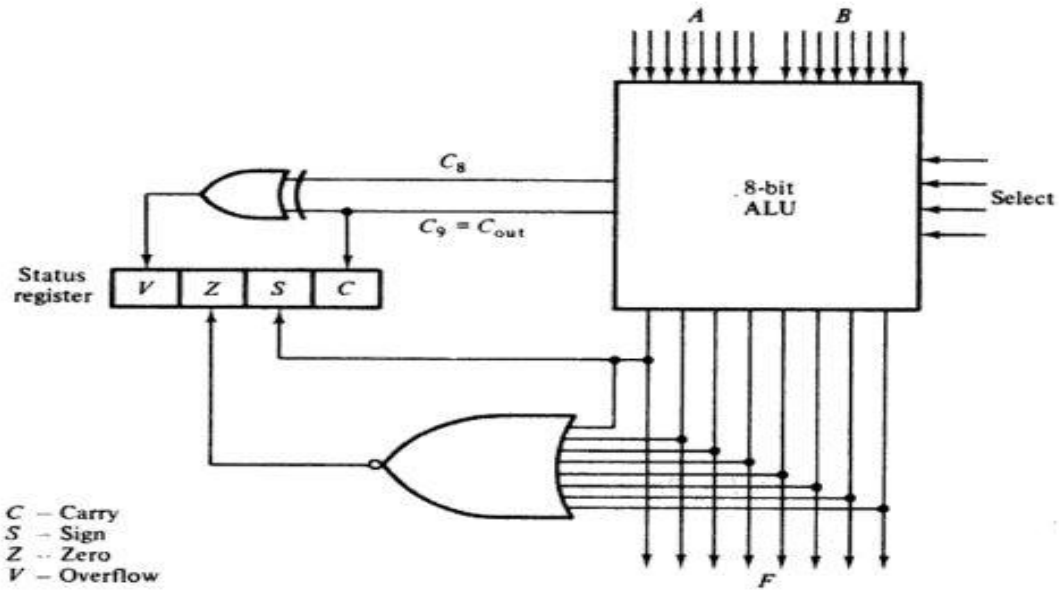


Figure 9-14 Setting bits in a status register

- **C-carry**: Bit C is set if the output carry of the ALU is 1. It is cleared if the output carry is 0.
- **S-sign bit**: Bit S is set if the highest order of the result in the output of the ALU is 1. It is cleared if the highest order bit is 0.
- **Z-zero**: Bit Z is set if the output of the ALU contains 0's and cleared otherwise. Z=1 if the result is zero and Z=0 if result is nonzero.
- **V-overflow**: set if there is any overflow . for a 8 bit ALU, V is set if the result is greater than 127 and less than -128

b) **Give the design of a 4 bit shifter. (5)**
Ans :

- Shift unit Attached to a processor transfers the Output of the ALU onto Output bus
- Shifter may transfer the information directly without a shift (or shift right or left)
- Shifter provides the Shift micro operation not available in an ALU

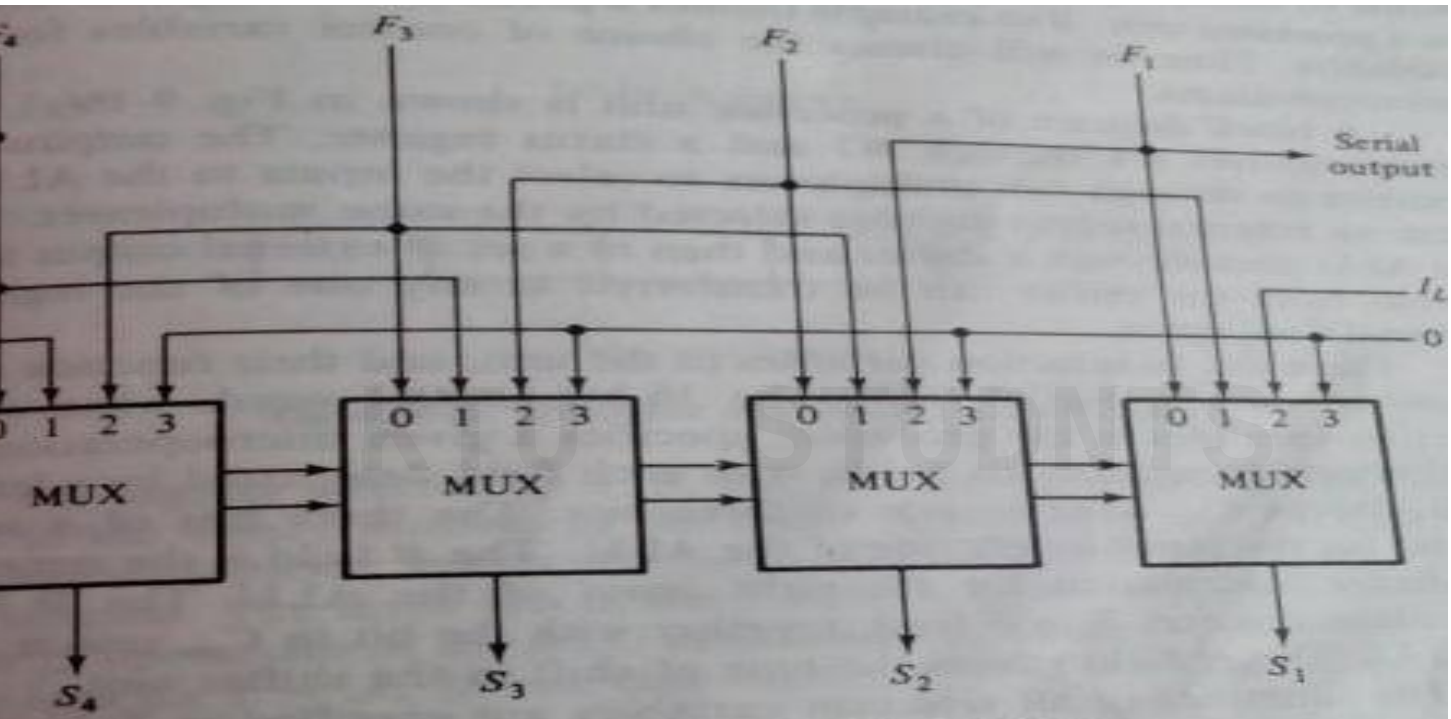


Figure 9-15 4-bit combinational-logic shifter

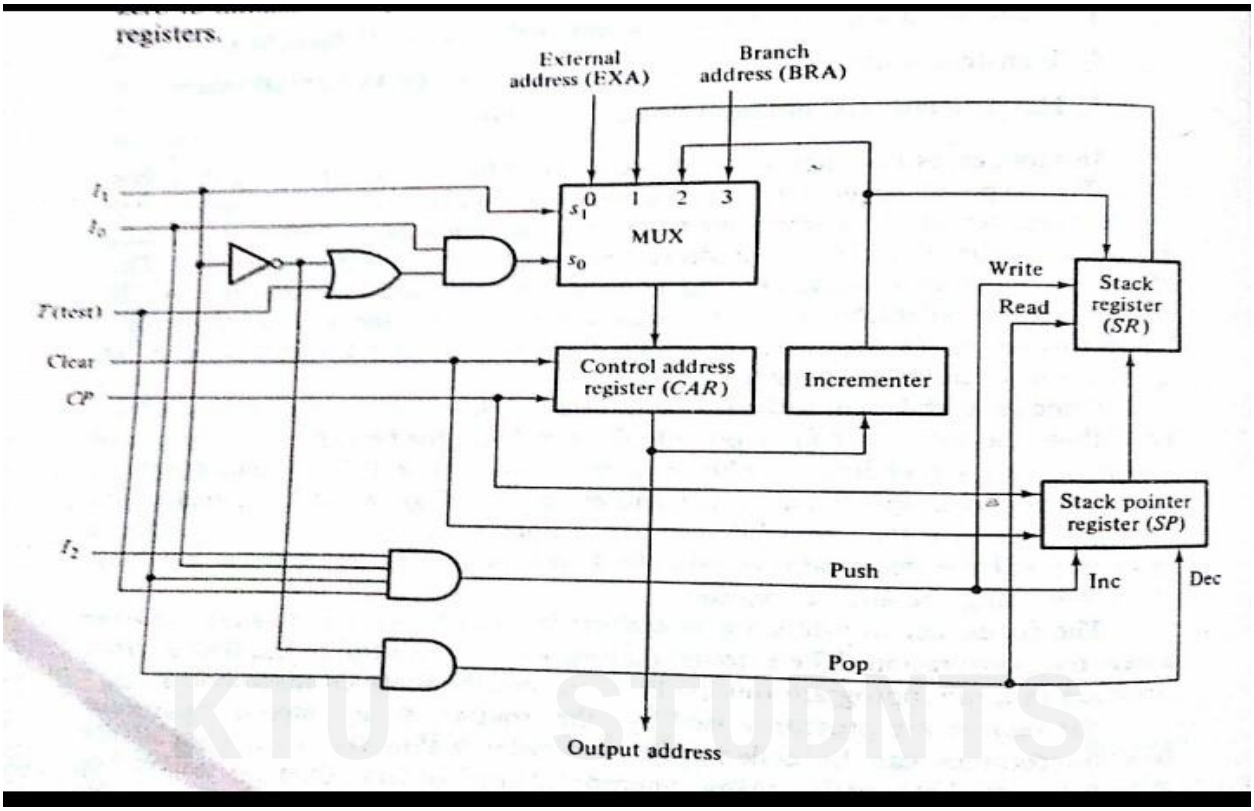
- Shifter is a bidirectional Shift register with parallel load
- The information from the ALU can be transferred to the register in parallel and then shifted to the right or left
- A clock pulse need for the transfer of the shift register and a pulse or shift pulse require to transfer information in shift register to a destination register.
- Hence only clock pulse needed in processor system is for loading the data from O/P bus in to destination register

TABLE 9-7 Function table for shifter

H_0	Operation	Function
0	$S \leftarrow F$	Transfer F to S (no shift)
1	$S \leftarrow \text{shr } F$	Shift-right F into S
0	$S \leftarrow \text{shl } F$	Shift-left F into S
1	$S \leftarrow 0$	Transfer 0's into S

18. **Explain the design of micro program sequencer with an example.**

Ans :

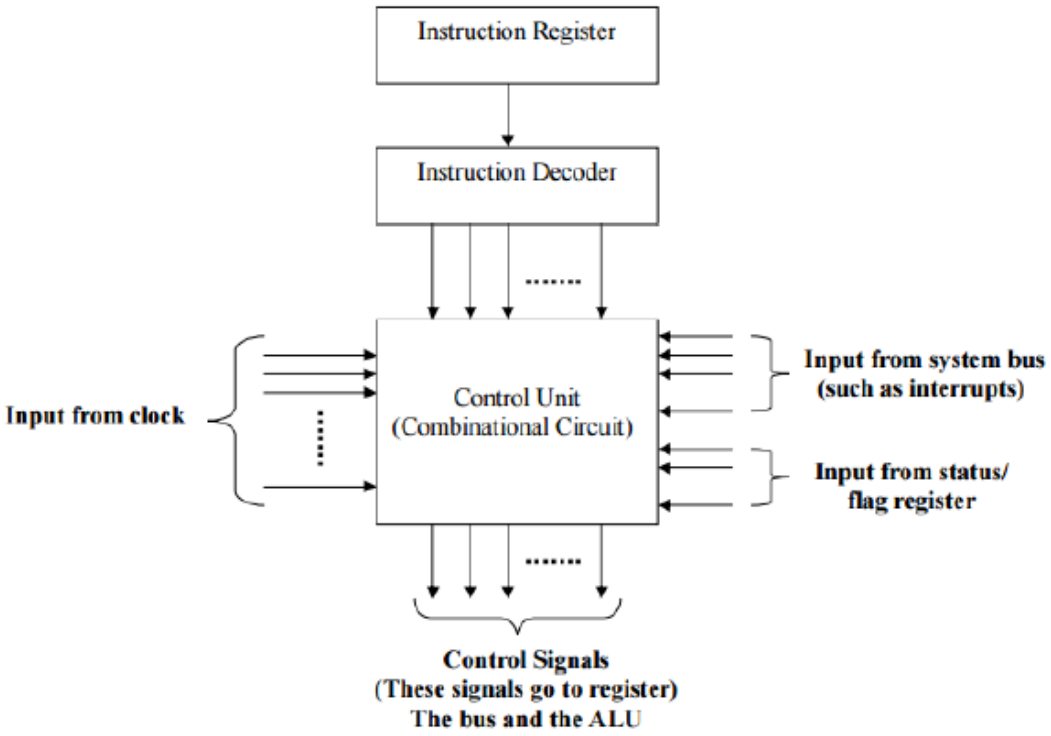


19. **Explain the procedure for designing a hardwired control, using an appropriate example.**

Ans :

- For each instruction, the control unit causes the CPU to execute a sequence of steps correctly.
- In reality, there must be control signals to assert lines on various digital components to make things happen.
- For example, when we perform an Add instruction in assembly language, we assume the addition takes place because the control signals for the ALU are set to "add" and the result is put into the AC.
- The ALU has various control lines that determine which operation to perform.

- The question we need to answer is, "How do these control lines actually become asserted?" We can take one of two approaches to ensure control lines are set properly.
- The first approach is to physically connect all of the control lines to the actual machine instructions.
- The instructions are divided up into fields, and different bits in the instruction are combined through various digital logic components to drive the control lines.
- This is called hardwired control, and is illustrated in figure (1).
- The control unit is implemented using hardware (for example: NAND gates, flip-flops, and counters).We need a special digital circuit that uses , as inputs, the bits from the Opcode field in our instructions, bits from the flag (or status) register, signals from the bus, and signals from the clock. It should produce, as outputs, the control signals to drive the various components in the computer.
- The advantage of hardwired control is that is very fast. The disadvantage is that the instruction set and the control logic are directly tied together by special circuits that are complex and difficult to design or modify



- If someone designs a hardwired computer and later decides to extend the instruction set, the physical components in the computer must be changed.
- This is prohibitively expensive, because not only must new chips be fabricated but also the old ones must be located and replaced

20. a**) Explain the different methods of control organization. (5)**

Ans :

1. One flip-flop per state method
2. Sequence register and decoder method
3. PLA Control
4. Micro program control

One flip-flop per state method

- This method uses one flip-flop per state in the control sequential circuit
- Only one flip-flop is set at any particular time
- All others are cleared
- The advantage of one flip-flop per state method is the simplicity with which it can be designed
- It offers a saving in design effort, an operational simplicity and a potential decrease in the combinational circuit required to implement the complete sequential circuit

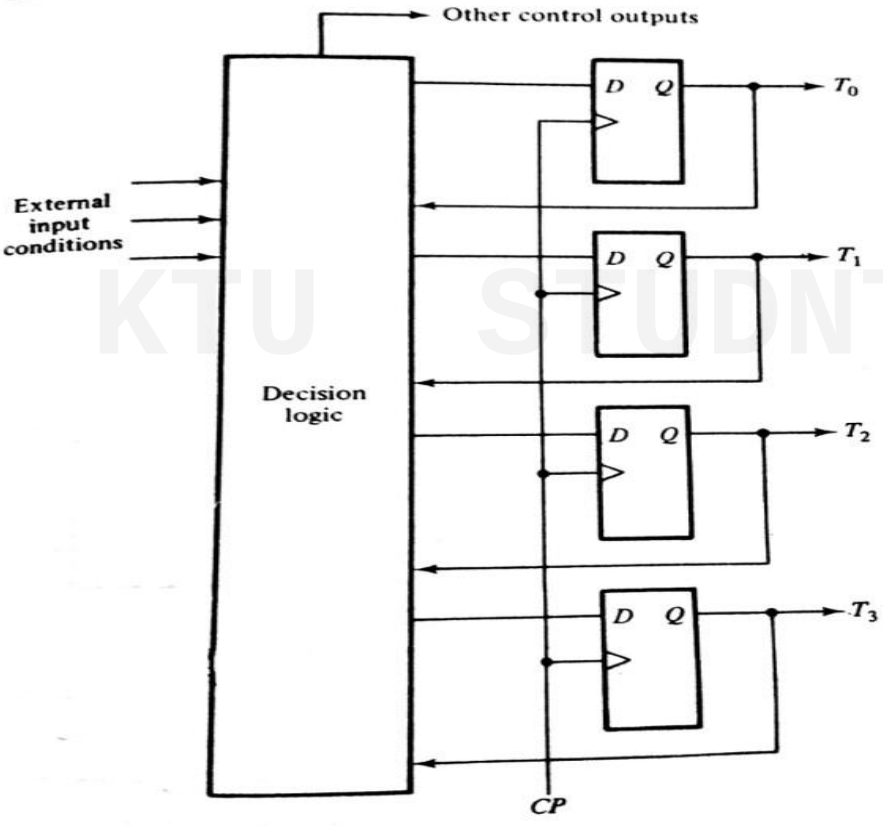


Figure 10-2 Control logic with one flip-flop per state

Sequence register and decoder method

- This method uses a register to sequence the control states

- This register is decoded to provide one output for each state
- For n flip-flop in the sequence register the circuit will have 2n states and the decoder will have 2n outputs
- Example , a 4-bit register can be in any one of 16 states

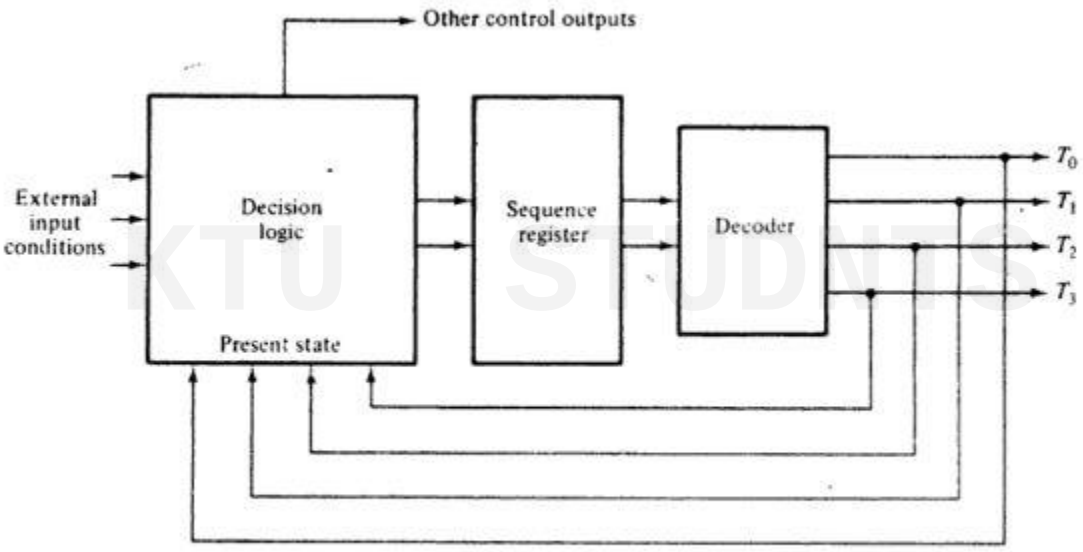


Figure 10-3 Control logic with sequence register and decoder

PLA Control

- The programmable logic array (PLA) control is essentially similar to the sequence register and decoder method except that all combinational circuit are implemented with the PLA, including the decoder and decision logic
- By using a PLA for the combinational circuit, it is possible to reduce number of IC and the number of interconnection wires

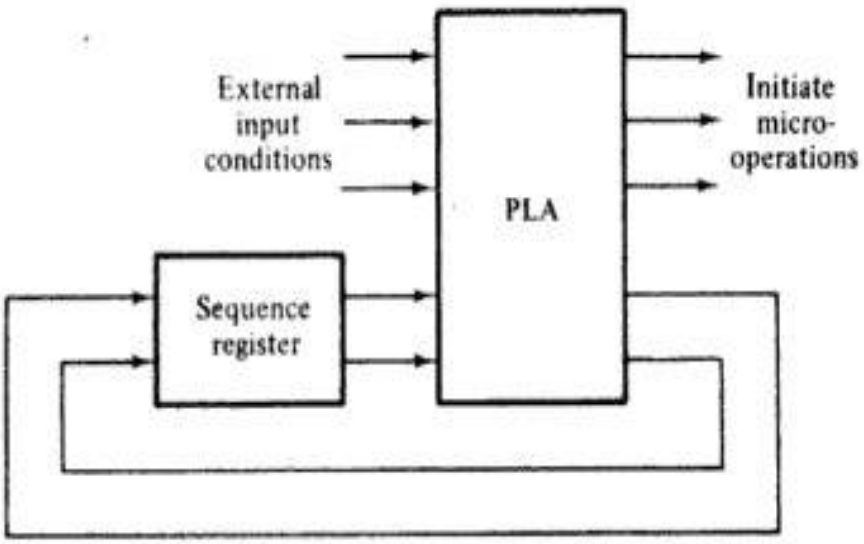
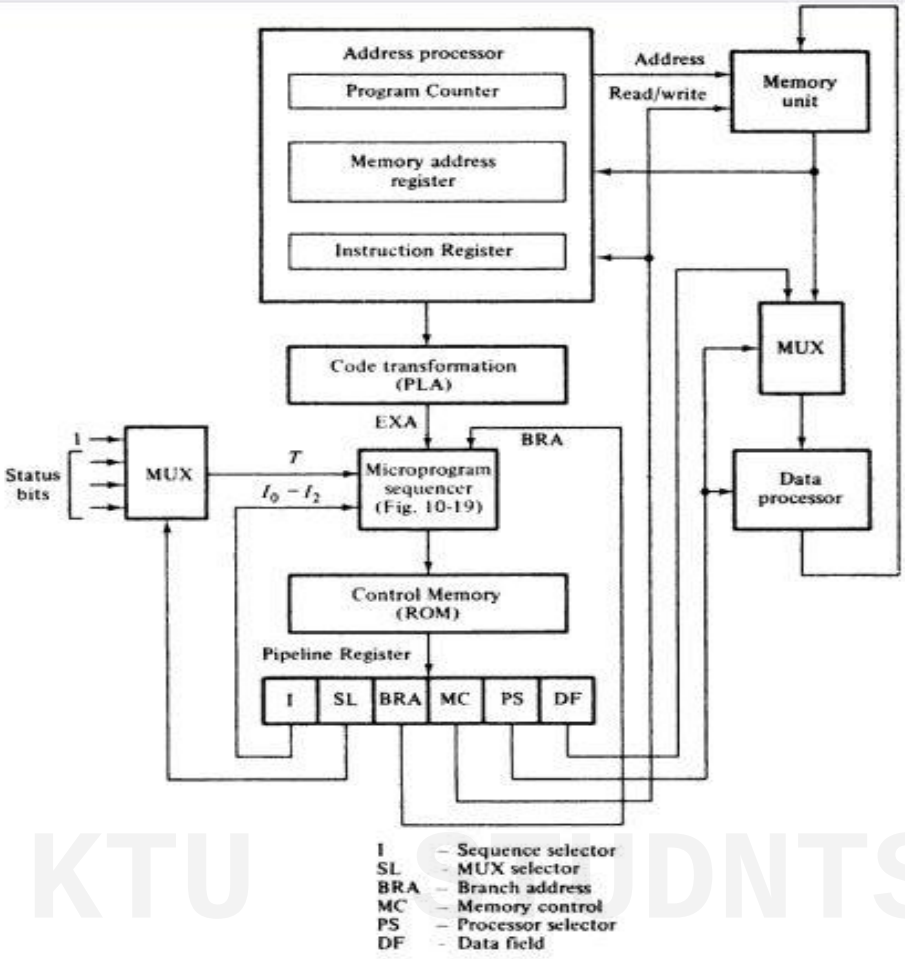


Figure 10-4 PLA control logic

b) **Explain micro programmed CPU organization with the help of a diagram**. (5)

Ans :

- A digital computer consists of a central processor unit (CPU) ,a memory unit, and input-output devices
- The CPU can be classified into two distinct but interactive functional sections.
- One section is the processing section and other is the control section
- A block diagram of micro programmed computer shown above figure



- **It consist of**
- Memory unit
- Two processor unit
- A micro program sequencer
- A control memory
- And few other digital functions

- The memory unit stores the instruction and data supplied by the user through an input device
- The data processor manipulates the data

- The address processor manipulates the address information received from memory
- An instruction extracted from memory during fetch cycle goes into the instruction register
- The micro program control unit consist of the sequencer of
- A **control memory** for storing microinstructions, a multiplexer and a pipeline register
- A **multiplexer** select one of many status and applied it to the T (test) input of sequencer
- The **pipeline register** is not always necessary because output from control memory can go directly to the control input of various unit in the CPU.
- However a pipeline register speeded up the control operation
- It allow next address to be generated and output of control memory to change while the current control word in the pipeline register will initiate the micro operations given by the present micro operations

- The possible micro instruction format for the control memory is illustrated within the pipeline register
- The I filed consist of 3 bit and supplies the input information for the sequencer
- The SL field select a status bit for the multiplexer
- The BRANCH field is the address filed of micro instruction and supplies a branch address to the sequencer
- These three field of micro operation provide information to the sequencer generate the next address and

- the control memory reads the next micro instructions are being executed in the other unit of CPU

✓ The other three filled in the microinstruction are for controlling the micro operations in the processor and memory units

✓ The **memory control(MC)** field controls the address processor and the read and write operations in the memory unit

✓ The **processor field (PS)** control the operations in the data processor unit.

✓ The last field is a **data field (DF)** used to introduce the constant into the processor.

10x4=40