

1. A) Memory is $16\text{TB} = 2^{44}$ bytes and word addressable.

The word is 8 bytes $= 2^3$

No of words in memory $= 2^{44}/2^3 = 2^{41}$

Address bus size is $\log_2 2^{41} = 41$ bits

So the address bus size is at least 41 bits.

1. B) IR is the register that holds the instruction that is currently being executed. Its output is available to the control circuits, which generate the timing signals that control the various processing elements involved in executing the instruction. The Program Counter (PC) keeps track of the execution of a program. It contains the memory address of the next instruction to be fetched and executed. PC points to the next instruction that is to be fetched from the memory. R0 to Rn-1 are general-purpose registers. The two registers, Memory Address Register (MAR) and Memory Data Register (MDR), facilitate communication with the memory. The MAR holds the address of the location to be accessed. The MDR contains the data to be written into or read out of the addressed location.

1. C. A computer with a Harvard architecture has separate memories for storing program and data; a computer with a von Neumann architecture stores program data and instruction data in the same memory.

1.D. The word size is 64 bits = 8 bytes $= 2^3$

Memory is 8GB $= 2^{33}$ bytes

No of words in memory $= 2^{33}/2^3 = 2^{30}$

So the address bus size is $\log_2 2^{30} = 30$ bits

The word size is 64 bits, so the data bus is 64 bits

There are 120 lines on the control bus

System bus size is $30 + 64 + 120 = 214$ lines

1. E. Convert to binary using 2's complement and represent accordingly.

2. I.) The PC is set to the first instruction, that is, the PC is 1000. The content of the PC is transferred to the MAR, and a Read control signal is sent to the memory. After the time required to access the memory elapses, the first instruction of the program is read out to the memory and loaded into MDR. Next, the content of MDR is loaded into IR. The IR holds the instruction in execution, Load LOCA, R1. Now, the instruction is ready to be decoded and executed. Once the instruction is loaded, the PC is updated to the address of the next instruction to be executed, that is, 1004. Since LOCA is a location in memory, the address is sent to the MAR, and a Read cycle is initiated. When the value has been read from the memory to the MDR, it is transferred from the MDR to the R1 register. Now, the next instruction is fetched from memory by sending the instruction address, which is 1004, to MAR, and a Read cycle is initiated. Once the instruction is in MDR, it is sent to the IR. Now, IR holds ADD R1, R0. Since it is an arithmetic operation, ALU is needed, and there are no operands to be read from memory (both operands are in registers), the ADD operation is performed, and the sum is stored in R0.

2.ii.) (i) In auto increment mode

Effective Address = 1205 and Operand = 2803

(ii) In auto decrement mode

Effective Address = 1204 and Operand = 1500

3. I.) i) Branch Target Address = PC(Updated) + Offset

Offset = Branch Target Address - PC(Updated)

= $1012 - 1024$

= -12

ii) PC = 1024 when R1 becomes 0

3li.) **3-address instruction format (2 marks)**

MUL B,C,R₁
 ADD R₁,A,R₂
 MUL E,F,R₁
 SUB D,R₁,R₃
 MUL G,H,R₄
 ADD R₃,R₄,R₅
 DIV R₂,R₅,X

a) **2-address instruction format (2 marks)**

MOV B,R0
 MUL C,R0
 ADD A,R0
 MOV D,R1
 MOV E,R2
 MUL R2,F
 SUB R1,R2
 MOV G,R3
 MUL H,R3
 ADD R2,R3
 DIV R0,R3
 MOV R3,X

4.

$$CPI_A = \frac{(8 \times 1 + 4 \times 3 + 2 \times 4 + 4 \times 3) \times 10^6}{(8 + 4 + 2 + 4) \times 10^6} = 2.22$$

$$MIPS_A = \frac{200 \times 10^6}{2.22 \times 10^6} = 90$$

$$ET_A = \frac{18 \times 10^6 \times 2.22}{200 \times 10^6} = 0.2 \text{ S}$$

$$CPI_B = \frac{(10 \times 1 + 8 \times 2 + 2 \times 4 + 4 \times 3) \times 10^6}{(10 + 8 + 2 + 4) \times 10^6} = 1.92$$

$$MIPS_B = \frac{200 \times 10^6}{1.92 \times 10^6} = 104$$

$$ET_B = \frac{24 \times 10^6 \times 1.92}{200 \times 10^6} = 0.23 \text{ S}$$

B. Even though, machine B has a higher MIPS than machine A, it needs a longer CPU time to execute the similar set of instructions.

5. A. $-16 = 11110000$ (2's Complement of 16 or 00010000)

$-20 = 11101100$ (2's Complement of 20 or 00010100)

$-36 = 111011100$ (Because original number is 2's complement of 11011100 I.e -36)

So, $V = 0$, $S = 1$, $Z = 0$, $C = 1$

B. SAFEPUSH: Compare #1100, SP

Branch ≤ 0 , FULLERROR

Move NEWITEM, $-(SP)$

SAFEPOP Compare #1500, SP

Branch > 0 EMPTYERROR

Move $(SP)+$, ITEM