

# SCHEME OF EVALUATION

# AUTUMN MID SEMESTER EXAMINATION-2023

School of Computer Engineering

Kalinga Institute of Industrial Technology, Deemed to be University

Computer Organization and Architecture [CS21002]

# Time: 1 1/2 Hours Full Mark: 40

***Answer Any four Questions including Question No. 1 which is compulsory. The figures in the margin indicate full marks. Candidates are required to give their answers in their own words as far as practicable and all parts of a question should be answered at one place only.***

**[Assume the Format of the instruction is opcode source, source/ Destination]**

1. Answer all the questions. [ 2 x 5 ]
   1. Distinguish between Von-Neuman and Harvard Architecture.

***Solution –***

1. *Main difference between Von-Neuman and Harvard Architecture is that Von-Neuman architectures have same memory for both program and data but Harvard architectures have separate memories for program and data.* ***[1 mark for this difference]***
2. *Having multiple memories makes Harvard architectures costly compared to Von-Neuman architectures.* ***[1 mark for any other difference]***
   1. How many memory references are required by the processor to execute the following instructions?
      1. ADD R1, (R2), R3
      2. PUSH X [X is a memory location]

***Solution –***

1. *Total 2 memory operations are required. One to fetch instruction from memory and second to fetch data stored at memory location stored in R2 due to register indirect mode.* ***[ 1 mark for correct answer]***
2. *Total 3 memory operations are required.**One to fetch instruction from memory, second to fetch data stored at memory location X and third to push data into stack.* ***[ 1 mark for correct answer]***
   1. Consider the code given below and answer the following questions. All the instructions are 4 bytes in length.
      1. What will be the offset for the instruction BGTZ (Branch greater than zero) to branch to the location labelled LOOP?
      2. What will be the content of PC when R1 becomes zero?

|  |  |
| --- | --- |
| **1000** | **MOV #10,R1** |
|  | **MOV #LOCA,R2** |
|  | **CLEAR R3** |
| **LOOP** | **ADD (R2)+, R3** |
|  | **DEC R1** |
|  | **BGTZ ?** |
|  | **CALL SUB1** |

***Solution –***

|  |  |
| --- | --- |
| **1000** | **MOV#10,R1** |
| **1004** | **MOV#LOCA,R2** |
| **1008** | **CLEARR3** |
| **LOOP(1012)** | **ADD(R2)+,R3** |
| **1016** | **DECR1** |
| **1020** | **BGTZ?** |
| **1024** | **CALLSUB1** |

i) Branch Target Address=PC(Updated)+Offset

Offset=Branch Target Address – PC(Updated)

= 1012-1024

= -12

ii) PC=1024 when R1 becomes 0

* 1. Two processor registers R1 and R2 hold signed operand. If the binary value of the R1 is 1111 1100 and R2 is 1110 1110 then, what will be the content of the register R1 and R2 after the following instructions are executed? [Assume the sizes of the registers are 8bit and numbers are represented in 2's complement format]
     1. AShiftL #2, R1
     2. AShiftR #1, R2

***Solution –***

**i)AShiftL #2, R1** [1]

|  |
| --- |
| 1111 0000 |

R1=

**ii)AShiftR #1, R2** [1]

|  |
| --- |
| 1111 0111 |

R2=

* 1. Discuss the importance of RUN and END control signals in hardwired control unit for an instruction execution.

***Solution –*** [1+1]

**RUN:**It is the control signal which is input to the sequence counter to increment the count(incrementer) ,if RUN=0 then counter stops counting and continue its counting when again RUN becomes 1.

**END**: It is the control signal which is input to the sequence counter to start the counter from 0(Reset) by indicating the end of current instruction and beginning of new instruction.

2.

1. What is addressing mode? Describe the following addressing mode with suitable example. [5]
   1. Immediate ii)Relative iii)Auto-increment iv)Register indirect

**SOLUTION**: Addressing Mode definition [1]

Each addressing mode explanation [1]

1. Write the assembly language code for the given expression using following instruction format: [5]

Z=(A+B-C\*D)/((E+F)\*(H+I-G))

1. RISC instruction format
2. STACK based CPU organization.

**SOLUTION:i) Using RISC instruction format** [2.5]

LOAD C, R1; R1<-[C]

LOAD D, R2; R2<-[D]

MUL R1, R2; R2<-[R1]+[R2]

LOAD A, R3; R3<-[A]

LOAD B, R1; R1<-[B]

ADD R1, R3; R3<-[R1]+[R3]

SUB R1, R3; R3<- R1-R3

LOAD E, R1; R1<- [E]

LOAD F, R2; R2<- [F]

ADD R1, R2; R2<- [R1]+[R2]

LOAD H, R1; R1<- [H]

LOAD I, R4; R4<- [I]

ADD R1, R4; R4<- [R1] + [R4]

LOAD G,R1; R1<-[G]

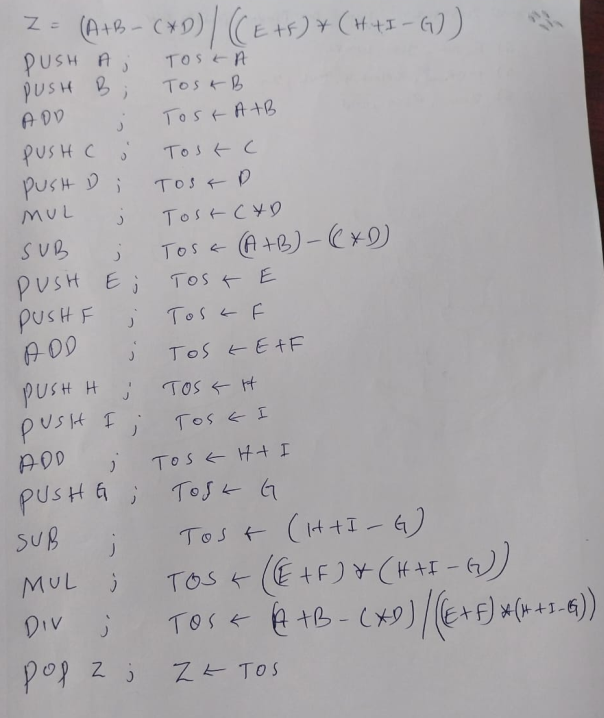
SUB R4, R1; R1<- [R4]-[R1]

MUL R2, R1; R1<- [R2]-[R1]

DIV R3, R1; R1<- [R3]-[R1]

STORE R1, Z; Z<- [R1]

Ii**) Using STACK based CPU organization** [2.5]

****

3.

1. Discuss the factors that affect the performance of a computer. There is a computer system which supports LOAD, STORE, ALU and BRANCH instructions. Suppose the number of clock cycles taken for each load, store, ALU and branch instruction are 5, 4, 3, 3 clock cycles respectively. If a program has 40% ALU instruction, 20% load instruction, 20% store instruction and 20% branch instructions; and the total number of instructions in the program are 5000 instructions. Now if the computer has a clock speed of 3.25 GHz then calculate the overall CPI and time taken to execute the program. [5]

**Solution –** *Factors which affects the performance of a computer are –*

1. ***Clock Speed –*** *Higher clock speed implies higher performance.*
2. ***Operating System –*** *Operating system handles tasks like program scheduling, memory operations, etc. and therefore an efficient OS can improve performance of a computer significantly, reverse of this is also true i.e., inefficient OS can also degrade performance.*
3. ***Number of processors –*** *More processors mean computer can perform more tasks in parallel.*
4. ***I/O Operations –*** *Computer speed is also depends upon I/O operations as I/O operations are usually very slow therefore these should be managed efficiently to take utilize CPU time more effectively.*

***There are other factors also which affects the performance of a computer. We can give 1 marks for factors and 4 marks for numerical part of question.***

*A white board with writing on it

Description automatically generated*

1. Define **Call** and **Compare** instruction. [5]

Initially the stack pointer SP contains 2000, keeping a value 500 in top of the stack. What are the content of PC, SP, and the top of the stack after the following operations in the code given in the table?

* 1. After the subroutine call instruction is executed in the main program?
  2. After the subroutine call instruction is executed in the subroutine SUB1?
  3. After the return from SUB2 subroutine?
  4. After the return from SUB1 subroutine?

|  |  |  |
| --- | --- | --- |
| **Subroutine-1 (SUB1)**  100: AND R11, R12  104: CALL SUB2  108: RETURN | **Main Routine**  200: ADD R1, R2  204: SUB R4, R5  208: CALL SUB1  212: MUL R3, R6 | **Subroutine-2 (SUB2)**  300: OR R8, R9  304: DIV R10, R12  308: RETURN |

**Solution –**

* + 1. **Call Instruction –** Call instruction is a special type of branch instruction which is used to implement subroutines in assembly programming. While executing call instruction, CPU first saves address of next instruction into link register (return address) then updates the PC to point first instruction in subroutine. Call instruction is paired with return instruction which specifies where to return when subroutine is completed.
    2. **Compare Instruction –** Compare instruction is used to compare two values and set appropriate conditional flags. Comparison of two values performed by subtracting one value from another. For example, *compare R1, R2* does R1-R2 and if result is 0 which means both values are equal then *Zero(Z) flag* is set to 1. If the result is negative then *Negative(N)* flag is set to 1.

***[1 mark for call & compare instruction, and 1 mark for each sub question]***

**A close-up of a whiteboard

Description automatically generated**

**A white board with writing on it

Description automatically generated**

A whiteboard with writing on it

Description automatically generated

4.

Explain the Single Bus Datapath in a processor with neat

diagram. Write control signals for the following instructions: [5+5]

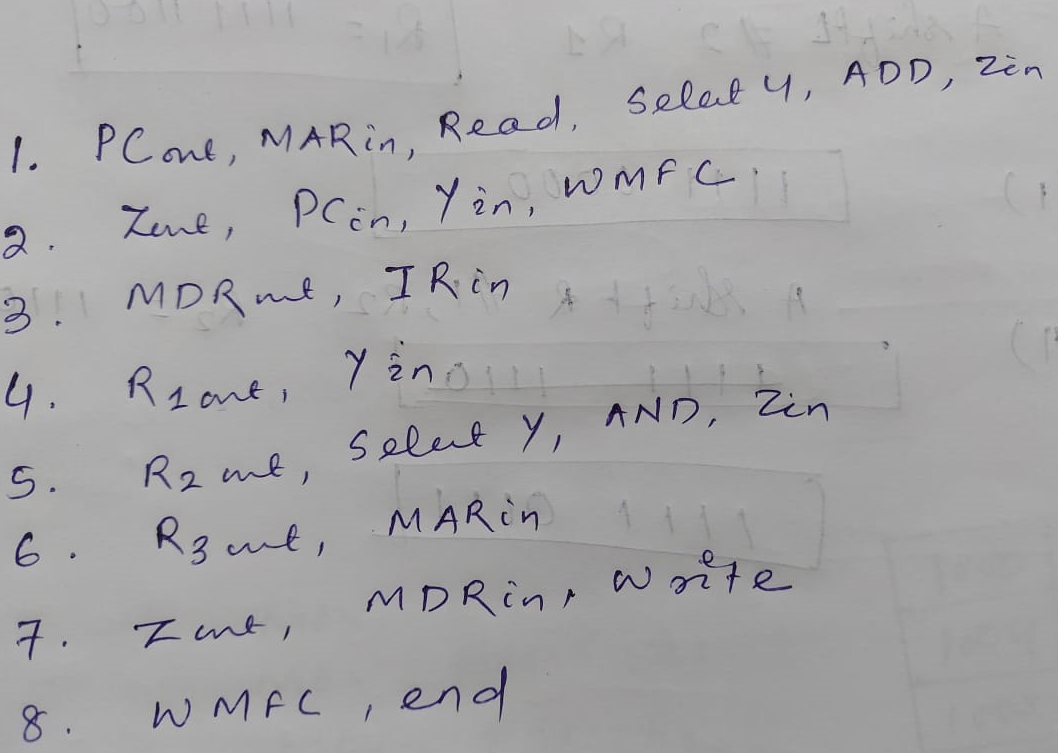
(i)AND R1, R2, (R3)

(ii)STORE R5,LOC

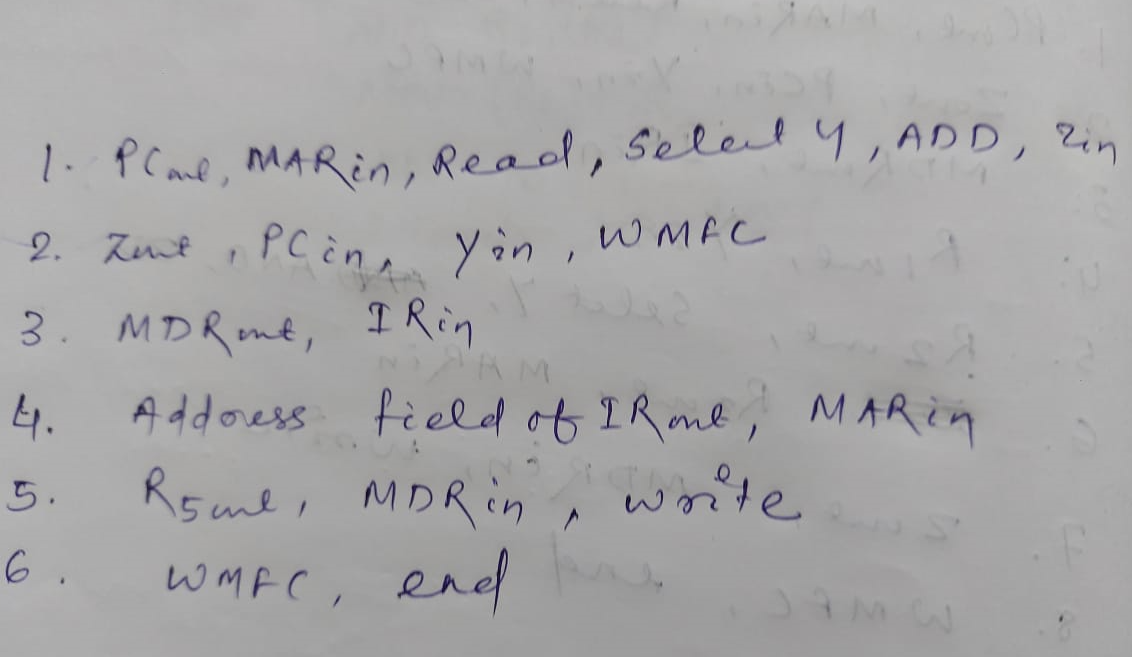
**SOLUTION-**Explanation of Single Bus Data path in a processor.[1]

Neat Diagram [1]

1. AND R1, R2, (R3) [4]



1. STORE R5,LOC [4]



5.

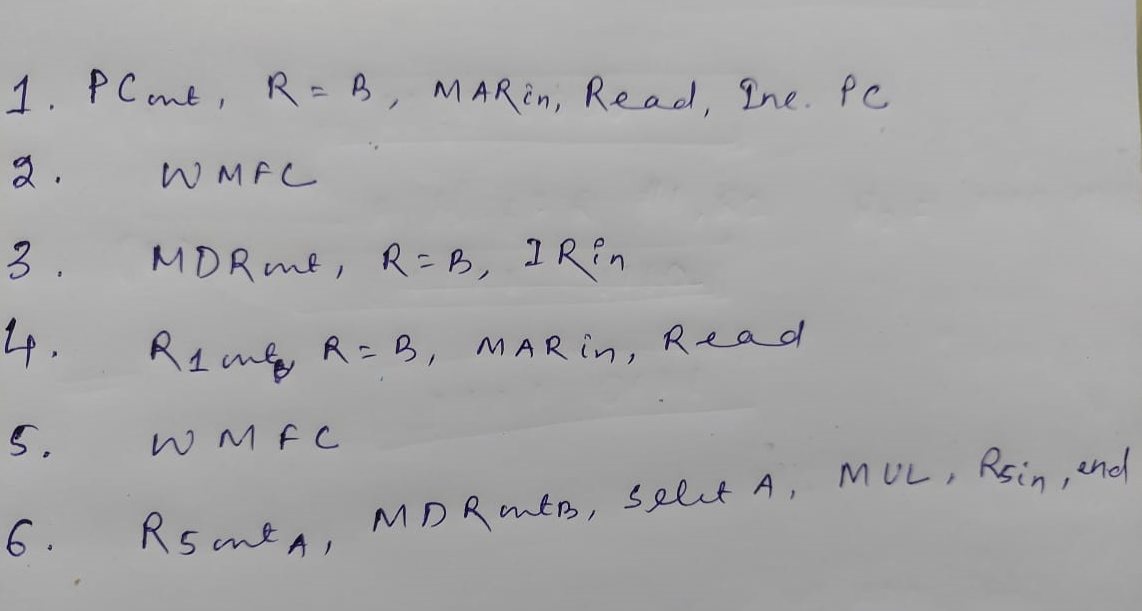
1. Explain the 3-bus architecture inside CPU with neat diagram. Write the control signals for the following instruction. [5]

MUL (R1) , R5

**SOLUTION-**Explanation of 3-bus architecture inside CPU .[1]

Neat Diagram [1]

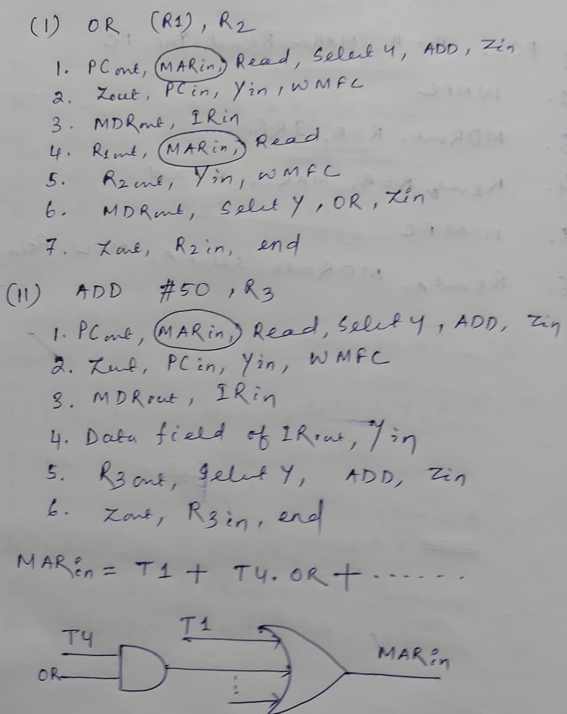
MUL (R1) , R5 [3]



1. Explain the working principle of Hardwired control unit along with neat diagram. Design the control logic and circuit for MARin with reference to the following instructions for single bus CPU Organization. [5]
   1. OR ( R1), R2
   2. ADD #50, R3

**SOLUTION-**Explanation of the working principle of Hardwired control unit . [1]

Neat Diagram [1]

 [3]

\*\*\* Best of Luck \*\*\*