Batch: B2 Roll No.: 16010124107
Experiment / assignment / tutorial No. 03
Grade: AA / AB / BB / BC / CC / CD /DD
Signature of the Staff In-charge with date

TITLE: Study of RISC and CISC Architecture

AIM: Understanding RISC and CISC Architecture

Expected OUTCOME of Experiment: (Mentions the CO/CO's attained)

We will learn about the architecture of RISC and CISC

Books/ Journals/ Websites referred:

1. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, "Computer Organization", Fifth Edition, TataMcGraw-Hill.
2. William Stallings, "Computer Organization and Architecture: Designing for Performance", Eighth Edition, Pearson.
3. Dr. M. Usha, T. S. Srikanth, "Computer System Architecture and Organization", First Edition, Wiley-India.

Pre Lab/ Prior Concepts:

Reduced Set Instruction Set Architecture (RISC)

Complex Instruction Set Architecture (CISC)

RISC Architecture

1. Diagram of RISC Architecture:

2. Brief Explanation of each component

3. RISC Processor Instruction Set Examples with explanation (Any 2)

1. **ADD (Addition) Instruction:**

2. **LW (Load Word) Instruction:**

- **CISC Architecture**

1. Diagram of CISC Architecture:

2. Brief Explanation of each component

3. CISC Processor Instruction Set Examples with explanation (Any 2)

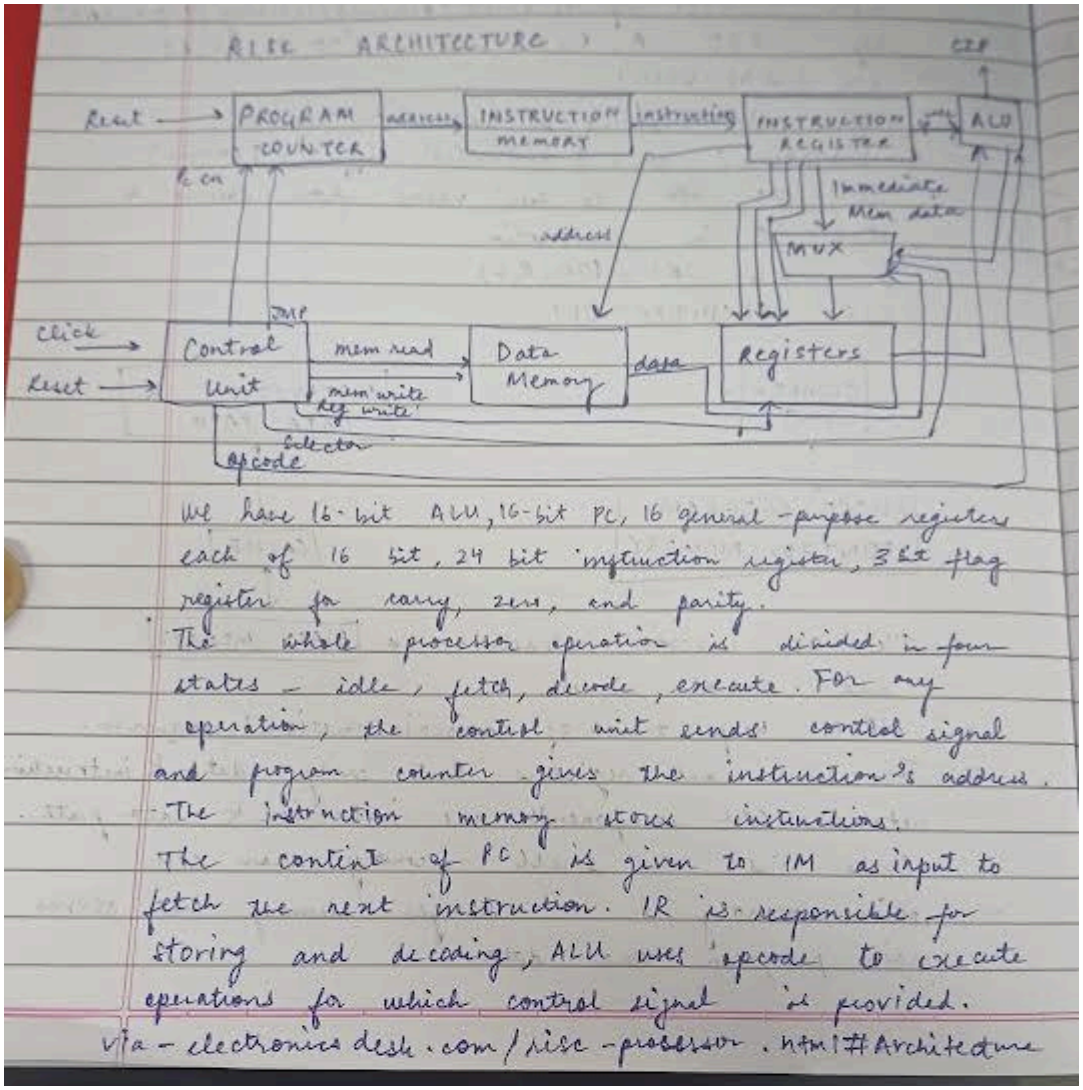
1. **MOV (Move):**

2. **ADD (Addition):**

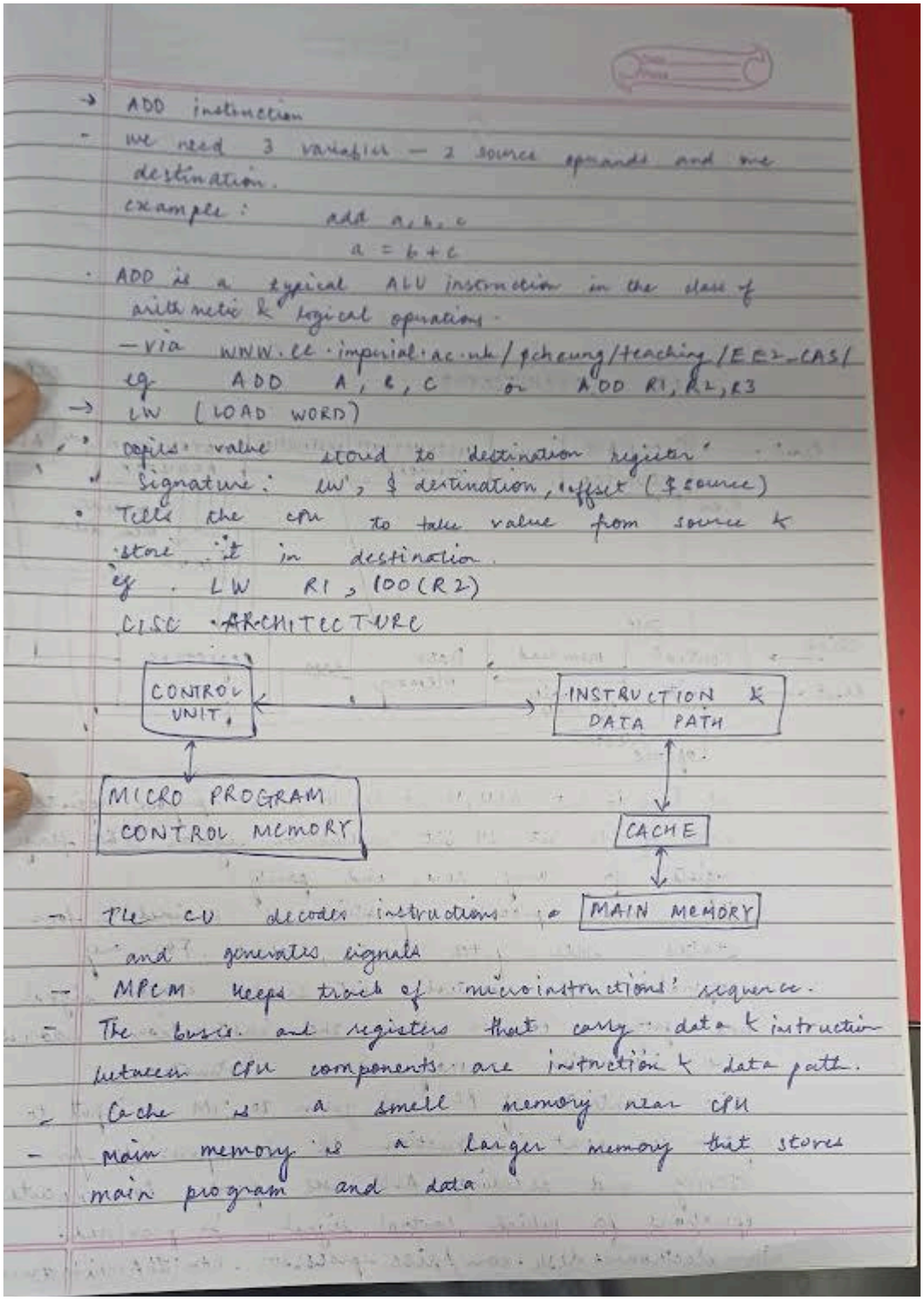
<u>Post Lab Descriptive Questions</u>

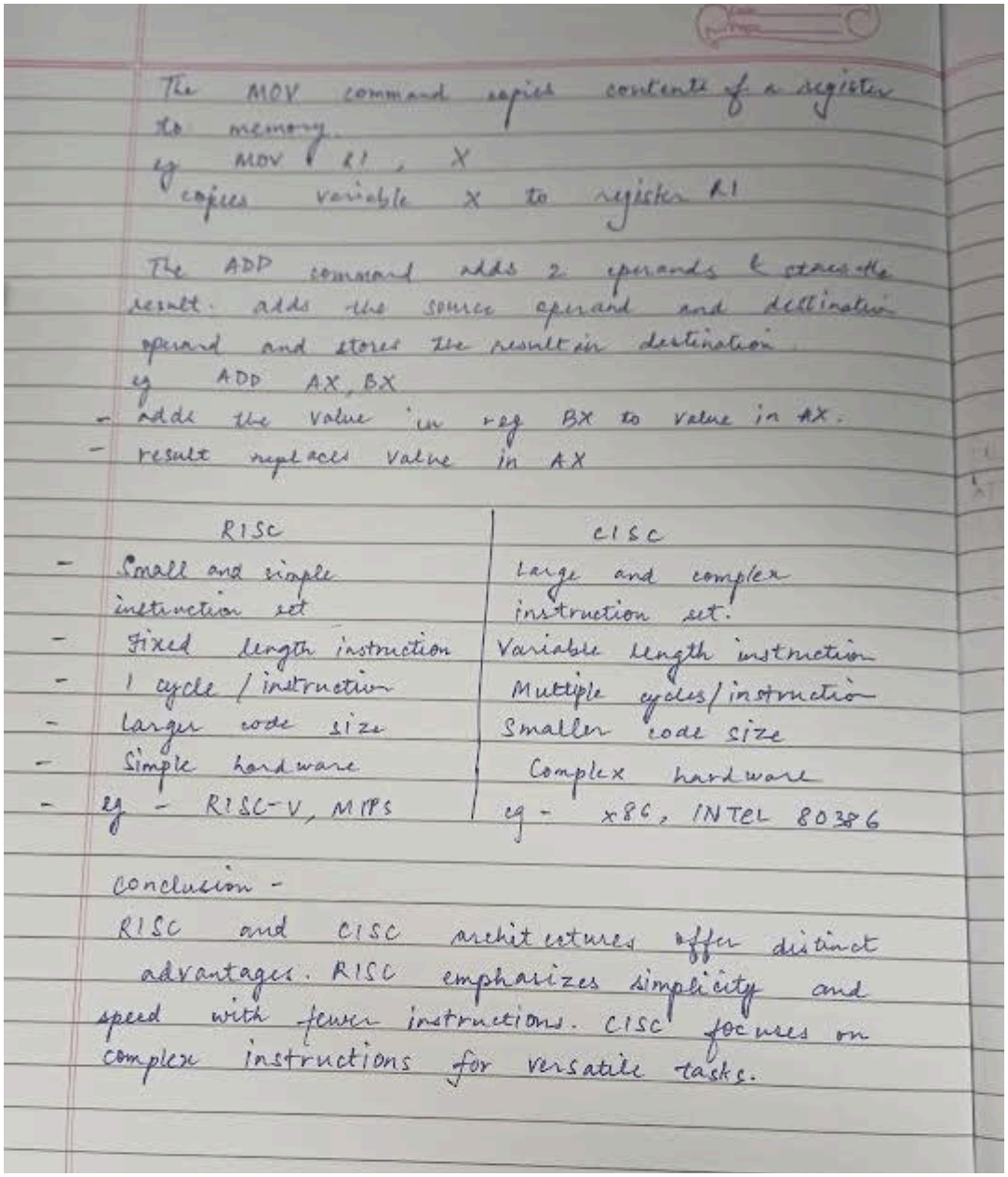
Write a tabular comparative analysis of RISC v/s CISC

Conclusion:



We have 16-bit ALU, 16-bit PC, 16 general-purpose registers each of 16 bit, 24 bit instruction register, 3 bit flag register for carry, zero, and parity.
The whole processor operation is divided in four states - idle, fetch, decode, execute. For any operation, the control unit sends control signal and program counter gives the instruction's address. The instruction memory stores instructions.
The content of PC is given to IM as input to fetch the next instruction. IR is responsible for storing and decoding, ALU uses opcode to execute operations for which control signal is provided.
via - electronicsdesk.com/risc-processor.html#Architecture


ADD instruction
we need 3 variables — 2 source operands and one destination.
example : add a, b, c
a = b + c
ADD is a typical ALU instruction in the class of arithmetic & logical operations.
- via www.ee.imperial.ac.uk/pcheung/teaching/EE2_CAS/
eg ADD A, B, C or ADD R1, R2, R3
LW (LOAD WORD)
copies value stored to destination register
Signature: LW, $ destination, offset ($ source)
Tells the CPU to take value from source & store it in destination.
eg. LW R1, 100(R2)
CISC ARCHITECTURE
CONTROL UNIT
INSTRUCTION & DATA PATH
MICRO PROGRAM CONTROL MEMORY
CACHE
MAIN MEMORY
The CU decodes instructions and generates signals
MPCM keeps track of microinstructions' sequence.
The buses and registers that carry data & instruction between CPU components are instruction & data path.
Cache is a small memory near CPU
main memory is a larger memory that stores main program and data



The MOV command copies contents of a register to memory.
eg MOV R1, X
copies variable X to register R1

The ADD command adds 2 operands & stores the result. adds the source operand and destination operand and stores the result in destination.
eg ADD AX, BX
- adds the value in reg BX to value in AX.
- result replaces value in AX

RISC	CISC
- Small and simple instruction set	Large and complex instruction set.
- Fixed length instruction	Variable length instruction
- 1 cycle / instruction	Multiple cycles/instruction
- Larger code size	Smaller code size
- Simple hardware	Complex hardware
- eg - RISC-V, MIPS	eg - x86, INTEL 80386

conclusion -
RISC and CISC architectures offer distinct advantages. RISC emphasizes simplicity and speed with fewer instructions. CISC focuses on complex instructions for versatile tasks.

Date: 03/10/2025